

# **Experimental methods 2 – Exam**

Complete portfolio

Group 3: Tobias, Jana, Ruta, Riikka and Mie

Date: 03.06.2019

## **Table of content**

Portfolio 1: Regression .....	2
Portfolio 2: Vectors and matrices .....	13
Portfolio 3: fMRI regression.....	28
Portfolio 4: fMRI preprocessing.....	41
Portfolio 5: Model specification .....	59
Portfolio 6: fMRI model estimation .....	72
Portfolio 7: Mixed effects.....	103
Portfolio 8: EEG Stop-Signal-NoGo .....	122
Portfolio 9: fMRI group analysis .....	124
Portfolio 10: PCA factor analysis .....	147

# Portfolio 1: Regression

Group 3: Mie, Riikka, Ruta, Jana, Tobias

February 12, 2019

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

# Loading packages
library(pacman)
p_load(nlme, sjPlot, tidyverse, pastecs)

# Setting up the working directory and reading in the file
setwd("C:/Users/Mie/Desktop/Cognitive science/Experimental methods
2/Assignments/Portfolio 1")
sleepstudy <- read.csv("sleepstudy.csv", header = T, sep = ",")
```

```
str(sleepstudy)
```

```
## 'data.frame': 180 obs. of 4 variables:
## $ X      : int 1 2 3 4 5 6 7 8 9 10 ...
## $ Reaction: num 250 259 251 321 357 ...
## $ Days    : int 0 1 2 3 4 5 6 7 8 9 ...
## $ Subject : int 308 308 308 308 308 308 308 308 308 308 ...
```

## 1: For one participant

### 1.a - Linear regression for reaction time as a function of days of sleep deprivation for one of the participants.

```
# Creating a new dataframe for participant 308 as a subset of the sleepstudy data.
sleepstudy$Subject <- as.factor(sleepstudy$Subject)
participant <- subset(sleepstudy, Subject == "308")
```

```
#The Linear model for reaction time as a function of days of sleep deprivation
#using the new dataframe.
M1 <- lm(Reaction ~ Days, data = participant)
summary(M1)
```

```
##
## Call:
## lm(formula = Reaction ~ Days, data = participant)
##
## 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

```

The linear model for reaction time as a function of days of sleep deprivation shows  $F(1, 8) = 17.12$ ,  $p = .00326$ .

### 1.b - Report degrees of freedom

For the F-value the degrees of freedom are Df = 1 and Df = 8.

### 1.c - At which F-value does a regression with this distribution become statistically significant ( $p < .05$ )?

```

# At a significance level of  $p < .05$  one can find the F-value at which the
# distribution becomes statistically significant by using the qf-function with the
# degrees of freedom from 1.b.
f_critical <- qf(0.95, df1 = 1, df2 = 8)
f_critical
## [1] 5.317655

```

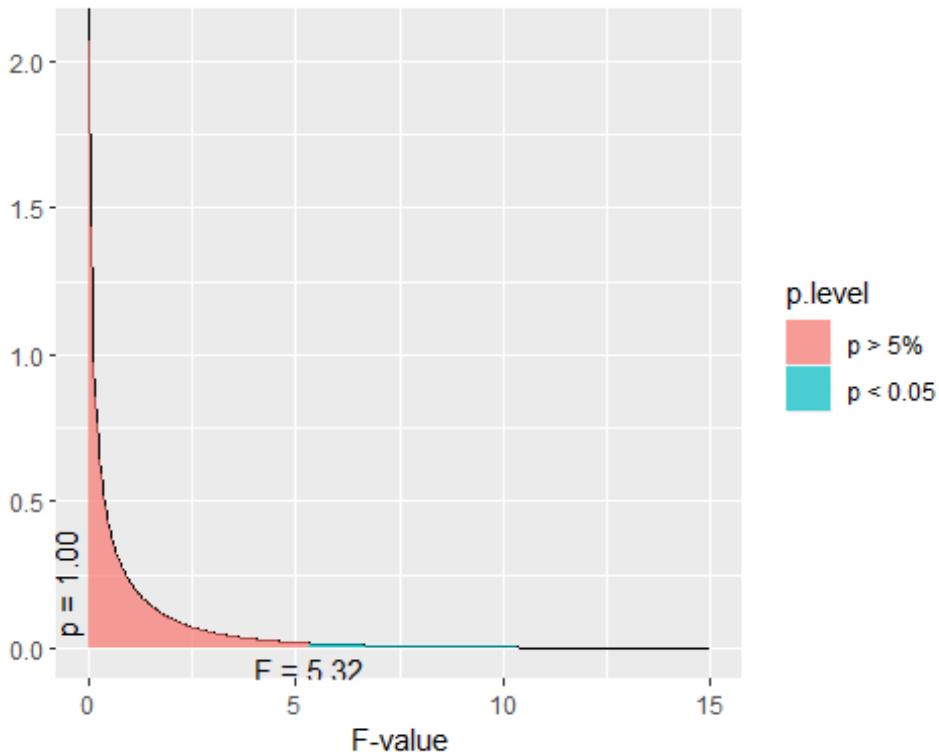
One can therefore read the critical value for F being statistically significant to be 5.32. This can also be seen on the following plot of the F-distribution.

### 1.d - Plot of the F-distribution

```

# To plot the F-distribution we simply use the dist_f function and put in the
# degrees of freedom from 1.b.
dist_f(f=0, deg.f1 = 1, deg.f2 = 8, xmax=15)

```



## 2: For all participants

### 2.a - Coefficients for the regression for reaction time as a function of days of sleep deprivation.

```
# By using the function lmList from the nlme package and specifying the regression
# for reaction time as a function of days a list of the slopes and intercepts for
# each of the participants is generated.
M2 <- lmList(Reaction ~ Days | Subject, data = sleepstudy)
M2_summary <- summary(M2)
M2_summary

## Call:
##   Model: Reaction ~ Days | Subject
##   Data: sleepstudy
##
## Coefficients:
##   (Intercept)
##       Estimate Std. Error t value    Pr(>|t|) 
## 308 244.1927   15.04169 16.23439 2.419368e-34
## 309 205.0549   15.04169 13.63244 1.067180e-27
## 310 203.4842   15.04169 13.52802 1.993900e-27
## 330 289.6851   15.04169 19.25882 1.122068e-41
```

```

## 331 285.7390 15.04169 18.99647 4.646933e-41
## 332 264.2516 15.04169 17.56795 1.236403e-37
## 333 275.0191 15.04169 18.28379 2.303436e-39
## 334 240.1629 15.04169 15.96649 1.135574e-33
## 335 263.0347 15.04169 17.48705 1.946826e-37
## 337 290.1041 15.04169 19.28667 9.653936e-42
## 349 215.1118 15.04169 14.30104 1.983389e-29
## 350 225.8346 15.04169 15.01391 2.939145e-31
## 351 261.1470 15.04169 17.36155 3.943049e-37
## 352 276.3721 15.04169 18.37374 1.402577e-39
## 369 254.9681 15.04169 16.95077 4.023936e-36
## 370 210.4491 15.04169 13.99106 1.253782e-28
## 371 253.6360 15.04169 16.86221 6.656453e-36
## 372 267.0448 15.04169 17.75365 4.373979e-38
## Days
##          Estimate Std. Error    t value   Pr(>|t|)
## 308 21.764702  2.817566  7.7246464 1.741840e-12
## 309  2.261785  2.817566  0.8027444 4.234454e-01
## 310  6.114899  2.817566  2.1702769 3.162541e-02
## 330  3.008073  2.817566  1.0676139 2.874813e-01
## 331  5.266019  2.817566  1.8689956 6.365457e-02
## 332  9.566768  2.817566  3.3954013 8.857738e-04
## 333  9.142045  2.817566  3.2446604 1.462120e-03
## 334 12.253141  2.817566  4.3488388 2.574673e-05
## 335 -2.881034  2.817566 -1.0225257 3.082469e-01
## 337 19.025974  2.817566  6.7526272 3.315759e-10
## 349 13.493933  2.817566  4.7892159 4.115160e-06
## 350 19.504017  2.817566  6.9222924 1.356856e-10
## 351  6.433498  2.817566  2.2833528 2.387301e-02
## 352 13.566549  2.817566  4.8149886 3.683105e-06
## 369 11.348109  2.817566  4.0276282 9.081880e-05
## 370 18.056151  2.817566  6.4084212 1.964766e-09
## 371  9.188445  2.817566  3.2611283 1.385338e-03
## 372 11.298073  2.817566  4.0098697 9.718197e-05
##
## Residual standard error: 25.59182 on 144 degrees of freedom

```

## 2.b - Combination of scatter plot and linear regression lines for each participant

```

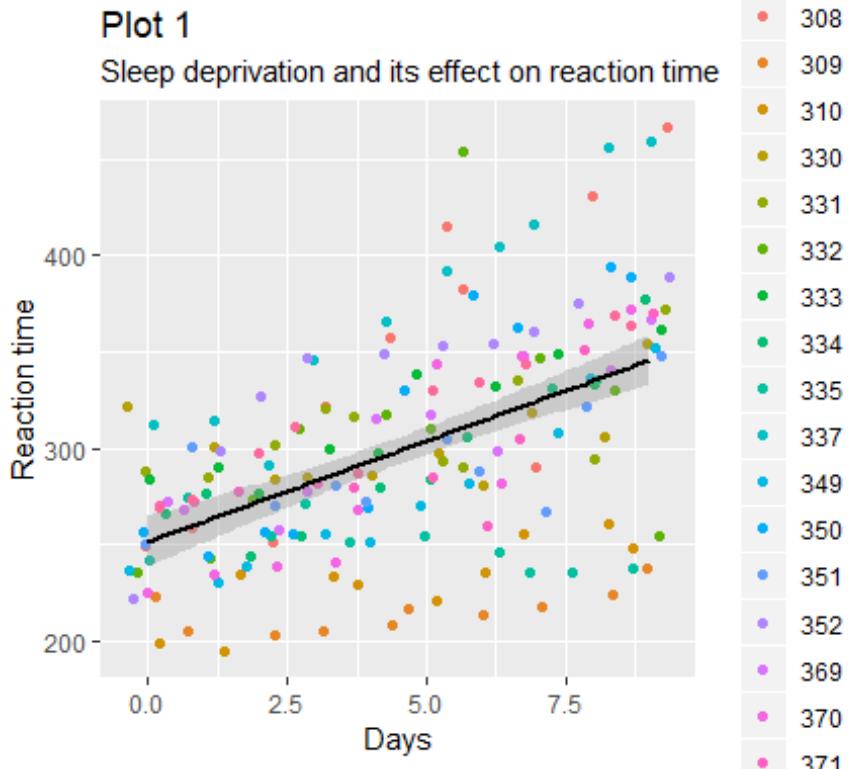
# A scatterplot is created with days on the x-axis and reaction time on the y-axis using the geom_smooth to create a linear regression line by specifying methods as linear model.
ggplot(sleepstudy, aes(Days, Reaction, color = factor(Subject)))+
  geom_point(position = "jitter")+
  geom_smooth(method="lm", color = "Black")

```

```
labs(title = "Plot 1", subtitle = "Sleep deprivation and its effect on reaction time", y = "Reaction time")
```

Plot 1

Sleep deprivation and its effect on reaction time

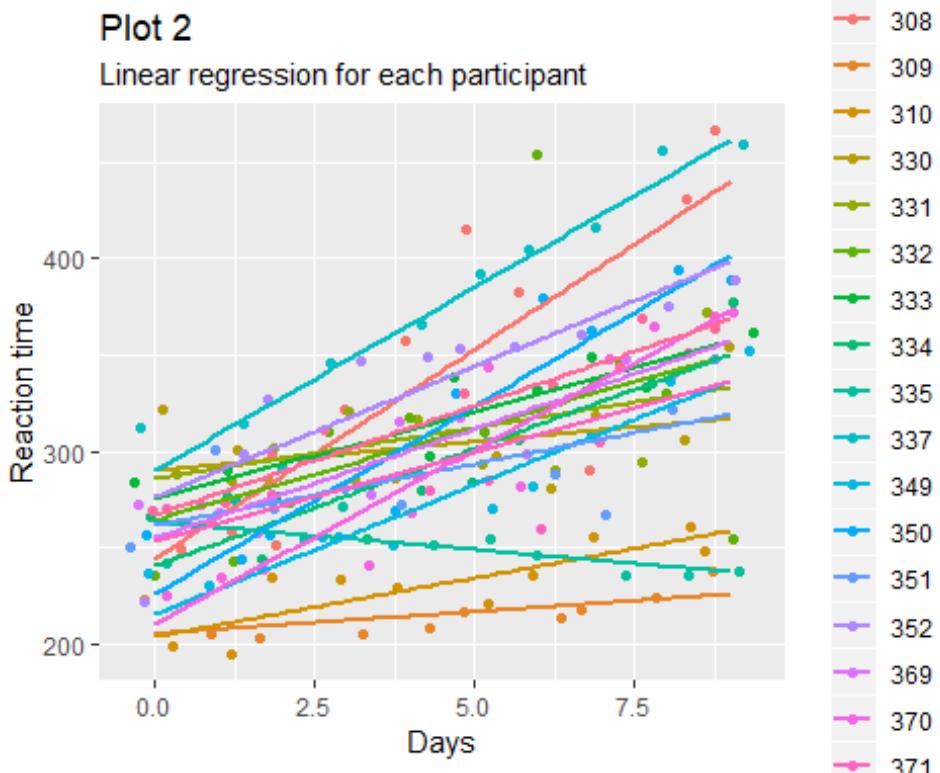


# A similar scatterplot is made but this time with linear regression lines for all participants.

```
ggplot(sleepstudy, aes(Days, Reaction, color = Subject))+
  geom_point(position = "jitter")+
  geom_smooth(method="lm", se=FALSE)+
  labs(title = "Plot 2", subtitle = "Linear regression for each participant", y =
"Reaction time")
```

**Plot 2**

Linear regression for each participant



**2.c - Collect and report the inferential statistics for each participant in a table using t-statistics, including t-value, df and p-value.**

```
# Choosing all subjects once (considering that they all have multiple
# observations) by using the unique function.
subject <- unique(sleepstudy$Subject)
# Creating a new data frame where the first column is the subjects
statistics <- data.frame(subject)
# Using the model summary from question 2.a to get the coefficients
M2_coef <- M2_summary$coefficients

# Adding the t-value to the empty dataframe using the participants (18), 3rd
# column = t-values, 2 = slope (and not intercept).
statistics$t_value <- M2_coef[1:18, 3, 2]

# Adding the p-value to the dataframe using the 4th column.
statistics$p_value <- round(M2_coef[1:18, 4, 2], digits = 4) # Rounding up to 4
# digits to make the p-values easier to interpret.

# Degrees of freedom should be the number of observations (Days) subtracted by 1
# (df = 9-1 = 8).
statistics$df <- 8
```

```

statistics

##   subject    t_value p_value df
## 1      308 7.7246464 0.0000  8
## 2      309 0.8027444 0.4234  8
## 3      310 2.1702769 0.0316  8
## 4      330 1.0676139 0.2875  8
## 5      331 1.8689956 0.0637  8
## 6      332 3.3954013 0.0009  8
## 7      333 3.2446604 0.0015  8
## 8      334 4.3488388 0.0000  8
## 9      335 -1.0225257 0.3082  8
## 10     337 6.7526272 0.0000  8
## 11     349 4.7892159 0.0000  8
## 12     350 6.9222924 0.0000  8
## 13     351 2.2833528 0.0239  8
## 14     352 4.8149886 0.0000  8
## 15     369 4.0276282 0.0001  8
## 16     370 6.4084212 0.0000  8
## 17     371 3.2611283 0.0014  8
## 18     372 4.0098697 0.0001  8

```

## 2.d - How many individual participants display a statistically significant effect of sleep deprivation (p-values uncorrected for multiple comparisons)?

```

# Specifying only the subjects with a statistically significant p-value (p<0.05).
statistics[which(statistics$p_value<0.05),]

##   subject    t_value p_value df
## 1      308 7.724646 0.0000  8
## 3      310 2.170277 0.0316  8
## 6      332 3.395401 0.0009  8
## 7      333 3.244660 0.0015  8
## 8      334 4.348839 0.0000  8
## 10     337 6.752627 0.0000  8
## 11     349 4.789216 0.0000  8
## 12     350 6.922292 0.0000  8
## 13     351 2.283353 0.0239  8
## 14     352 4.814989 0.0000  8
## 15     369 4.027628 0.0001  8
## 16     370 6.408421 0.0000  8
## 17     371 3.261128 0.0014  8
## 18     372 4.009870 0.0001  8

```

As can be seen from the generated dataframe, 14 participants had a significantly different reaction time than the mean reaction time for the population sample.

### 3: Across participants

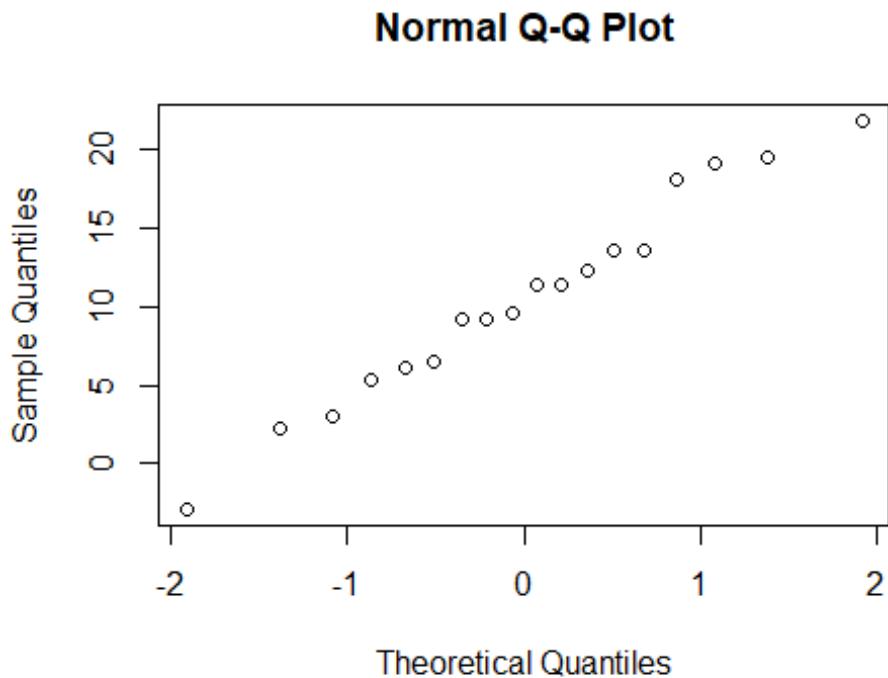
3.a Use the slopes you found for each participant in exercise 2 as a new dataset. Test the hypothesis that the slopes are larger than zero against the null-hypothesis that the slopes are zero (i.e. no differences in response time exist as a function of time).

```
# Making a new data frame for the coefficients using the model 2 from question 2.a
coefficients <- coef(M2)
# Getting the slopes from the data frame
slopes <- coefficients$Days

# Testing the slopes for normality by using statistic descriptions and making a
# qq-plot.
stat.desc(slopes, basic = TRUE, norm = TRUE)

##      nbr.val     nbr.null     nbr.na          min          max
## 18.00000000  0.00000000  0.00000000 -2.88103394 21.76470242
##      range        sum       median         mean       SE.mean
## 24.64573636 188.41114727 10.43242061 10.46728596 1.54578890
## CI.mean.0.95      var      std.dev      coef.var      skewness
## 3.26132949 43.01033961  6.55822687  0.62654511 -0.06529538
## skew.2SE      kurtosis      kurt.2SE      normtest.W      normtest.p
## -0.06087831 -0.82068670 -0.39539921  0.97880715  0.93687026

qqnorm(slopes)
```



```
# Performing the one sample t-test testing whether the slopes are greater than 0
t.test(slopes, mu=0, alternative = "greater")

##
## One Sample t-test
##
## data: slopes
## t = 6.7715, df = 17, p-value = 1.632e-06
## alternative hypothesis: true mean is greater than 0
## 95 percent confidence interval:
## 7.778221      Inf
## sample estimates:
## mean of x
## 10.46729
```

### 3.b: Justify your use of test statistics.

As can be seen in the testing of normality in question 3.a the p-value is 0.94 indicating that the data is normally distributed. This can furthermore be seen on the quantile-quantile plot where the data is approximating a linear relation. Since our data does not violate the assumption of normality we used a one-sample t-test. This made it possible to see if there was a significant difference between the null hypothesis (there not being a difference in reaction time over the course of days of sleep deprivation - slopes = 0) and the slopes from the test data.

### 3.c: Report inferential statistics.

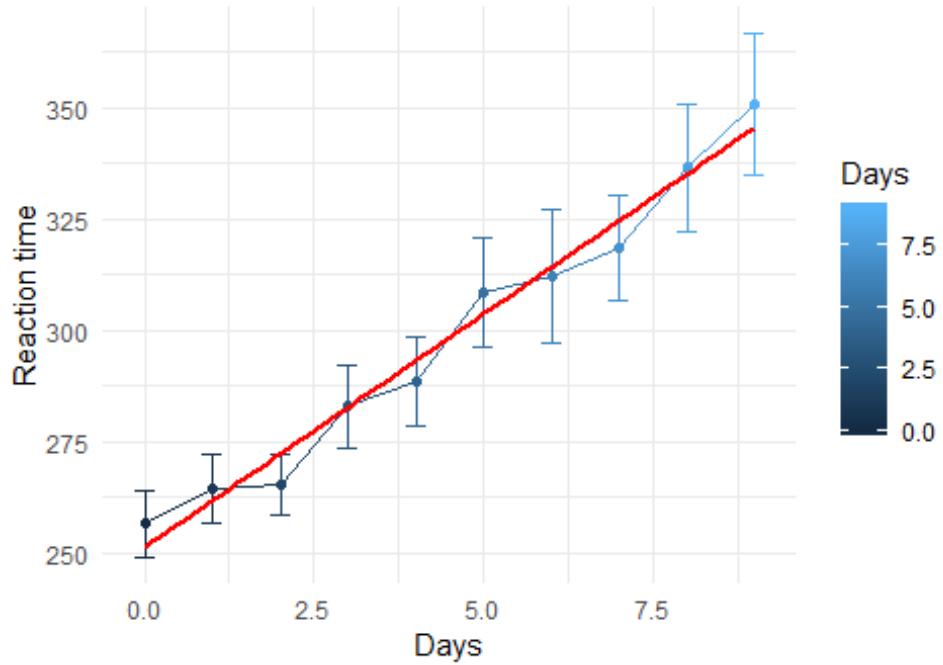
From the t-test and the statistic descriptions in question 3.a, it is evident that sleep deprivation significantly affected the reaction time ( $M = 10.47$ ,  $SD = 6.56$ );  $t(17) = 6.77$ ,  $p < .000001$ .

### 3.d: Make a plot with the mean reaction time and standard error bars for each day across participants and plot the averaged regression line in the same figure.

```
# Loading the package Rmisc
p_load(Rmisc)
summary_sleep <- summarySE(sleepstudy, measurevar = "Reaction", groupvars="Days")
#Summarise reaction time grouped by days of sleep deprivation
ggplot(summary_sleep, aes(x = Days, y = Reaction, color = Days)) +
  geom_errorbar(aes(ymin = Reaction - se, ymax = Reaction + se), width = .3) +
  #using the SE values from the summarySE function to calculate the error bars (ymin and ymax)
  geom_line() + # connecting every datapoint with a line
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") + # The averaged
regression line.
  labs(title = "Plot 3",
       subtitle = "Reaction Time by Days of Sleep Deprivation",
       y = "Reaction time") +
  theme_minimal()
```

### Plot 3

Reaction Time by Days of Sleep Deprivation



## Portfolio 2: Vectors and matrices

Group 3: Mie, Tobias, Jana, Riikka and Ruta

February 13, 2019

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

# Loading packages
library(pacman)
p_load(tidyverse, rgl, jpeg)

# Setting the working directory
setwd("C:/Users/Mie/Desktop/Cognitive science/Experimental methods
2/Assignments/Portfolio 2")

# Creating vectors for exercise 1
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)
```

### 1: Linear regression

**1.a - Make a constant vector of the same length as the data, consisting of ones.**

```
# Creating a constant vector with one repeated 10 times using the rep-function.
constant <- rep(1, 10)
```

**1.b - Report the inner product (aka dot product) of the days vector and the constant vector.**

```
# Calculating the dot product
dot <- sum(Days372*constant)
dot

## [1] 45
```

Evidently, the inner product of the days vector and the constant is 45.

**1.c - What does the dot product say about the possibility of finding an optimal linear regression?**

The dot product tells us about the angle between the two vectors. The closer the dot product is to zero, the better the linear regression fits the data. This also comes from the cosine to the angle being equal to the correlation coefficient  $r$ , where  $r = \cos(0) = 1$ . The smaller the angle the more two vectors are dependent on each other and also more variance in one variable can be explained by the other variable. So, 45 degree angle equals to 50% of explained variance.

### **1.d - Create a 10x2 matrix called X with the days vector and constant vector as columns and use the least**

squares method manually to find the optimal coefficients (i.e. slope and intercept) to reaction time.

```
# Creating a 10x2 matrix consisting of the two vectors
X <- matrix(c(Days372, constant), nrow = 10, ncol = 2)

# Calculating the beta estimates on the base of the X-matrix and the vector for
# the reaction times
beta <- solve(t(X)%%X)%%t(X)%%Reaction372
beta

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

From calculating the beta coefficients using the least squares method we get a slope of 11.298 and an intercept of 267.044.

### **1.e - Check result using lm().**

```
# Using the provided formula where the zero indicates the removal of the default
# constant.
lm(Reaction372~0+X)

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

As can be seen from the linear model, we get the same coefficients (slope = 11.3, intercept = 267).

### **1.f - Subtract the mean of Days372 from the Days372 vector. Replace the days vector with the new vector**

in X and redo the linear regression. Did the coefficients change?

```
# Making a new vector by subtracting the mean of days from the days vector
newDays <- Days372 - mean(Days372)

# Replacing the days vector with the new vector in the X matrix
X2 <- matrix(c(newDays, constant), nrow = 10, ncol = 2)

# Doing the Linear regression
```

```

beta2 <- lm(Reaction372~0+X2)
beta2

##
## Call:
## lm(formula = Reaction372 ~ 0 + X2)
##
## Coefficients:
##   X21     X22
##  11.3   317.9

```

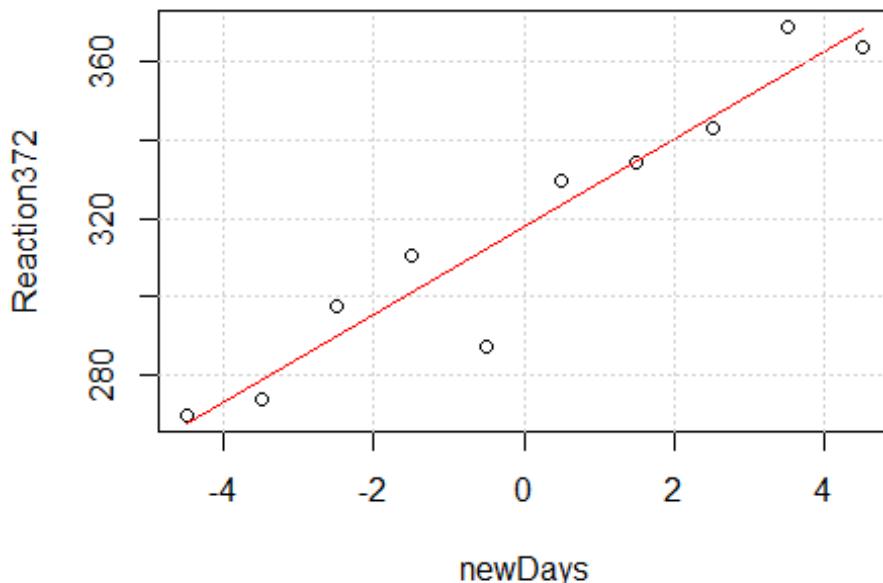
One can see that the slope is the same but that the intercept has changed, which makes sense since the numbers have changed while the interval between them remain.

**1.g - Make a scatter plot with the mean-centered days covariate against response time and add the best fitted line.**

```

# Plotting the data points with the best fitted line having the first index of the
# day vector (newDays[1]) be the intercept (beta[1]) and the last index of the day
# vector (newDays[10]) be the intercept added to the slope when going 9 days out on
# the x-axis (9*beta[1]).
plot(newDays, Reaction372)+
  lines(c(newDays[1],newDays[10]),c(beta[2],beta[2]+9*beta[1])),col="red")+
  grid()

```



```
## integer(0)
```

## 2: Images and matrices

```
# Loading the data  
matrix <- readJPEG("Matrices data.jpg", native = FALSE)
```

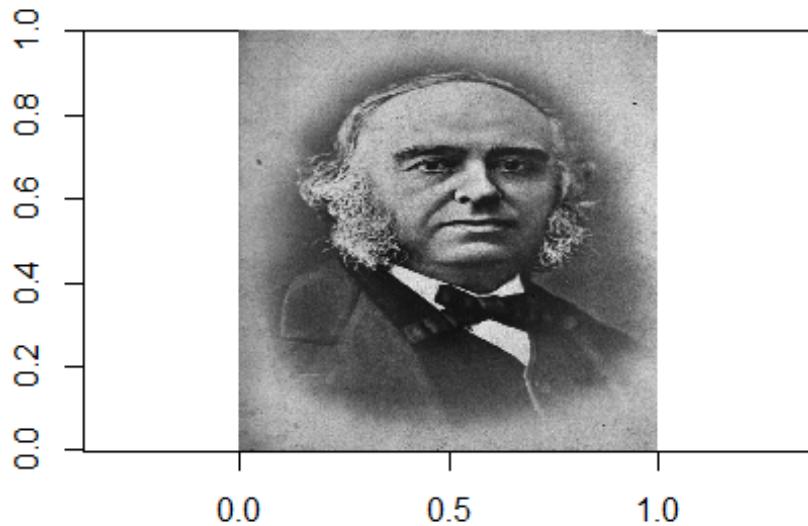
**2.a - Report how many rows and how many columns the matrix has. What are the maximum, minimum and mean pixel values?**

```
# Using the stringr function to check rows and columns  
str(matrix)  
  
## num [1:900, 1:606] 0.365 0.314 0.31 0.345 0.353 ...  
  
# Calculating the maximum, minimum and mean value of the matrix  
max_value <- max(matrix)  
max_value  
  
## [1] 1  
  
min_value <- min(matrix)  
min_value  
  
## [1] 0.0627451  
  
mean_value <- mean(matrix)  
mean_value  
  
## [1] 0.5118474
```

From the stringr output we can see that the matrix have 900 rows and 606 columns. In the matrix, tha maximum value is 1, the minimum value is 0.06 and the mean value is 0.51.

**2.b - Make an image of the loaded matrix. Be sure to rotate the image into the correct orientation. The functions needed are found the in lecture slides. Furthermore, grey scale the picture with gray(1:100/100) - this will color values near 0 black/dark and values near 1 white/light.**

```
# Making a function for the rotation matrix that takes the columns and reverse  
them before transposing the matrix.  
rotate <- function(x) t(apply(x, 2, rev))  
# Reversing the matrix  
matrix2<-rotate(matrix)  
  
# Making the image of the rotated matrix  
image(matrix2, col = gray(1:100/100), asp = 1) # asp = 1, shows the distances  
between the data points in the right scale.
```



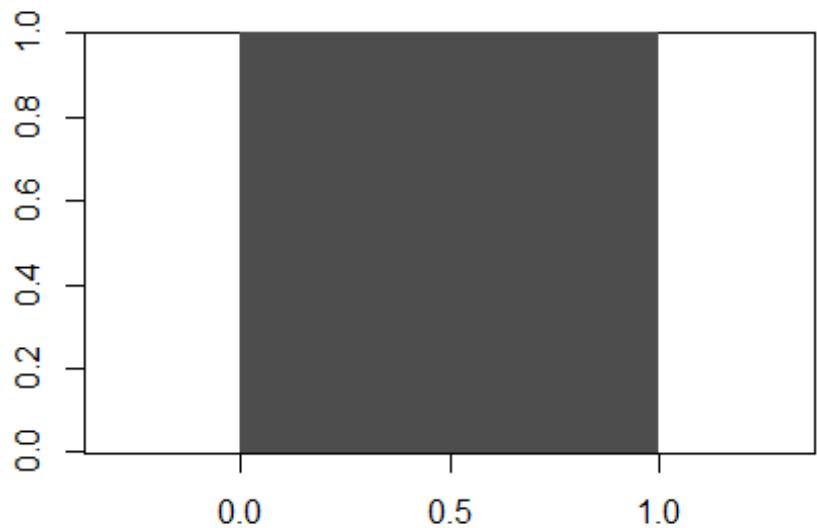
**2.c - Draw an image with the same dimensions as that from 2.b. But this image should be completely black.**

```
# Creating a matrix consisting of zeros
r <- rep(0, 900)
c <- rep(0, 606)
black_matrix <- matrix(c(r, c), nrow = 900, ncol = 606)

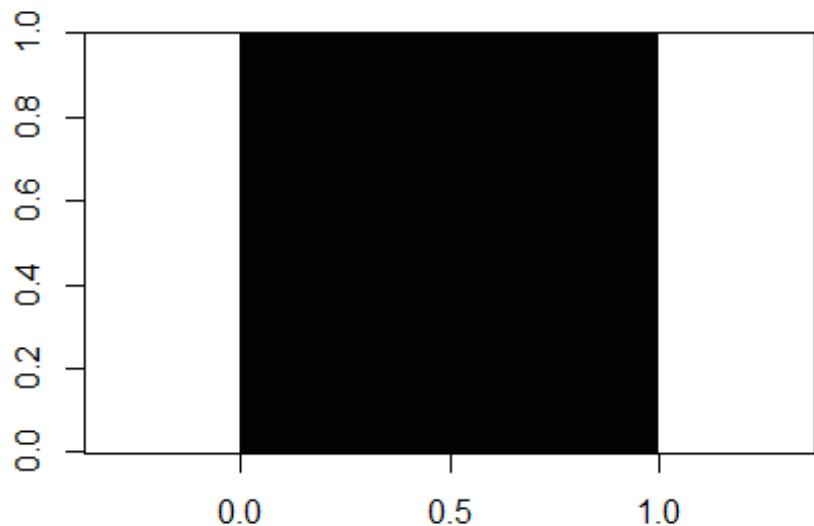
## Warning in matrix(c(r, c), nrow = 900, ncol = 606): data length [1506] is
## not a sub-multiple or multiple of the number of rows [900]

# Plotting the image
image(black_matrix, col = gray.colors(1:100/100), asp = 1)

## Warning in seq.int(from = start^gamma, to = end^gamma, length.out = n):
## first element used of 'length.out' argument
```

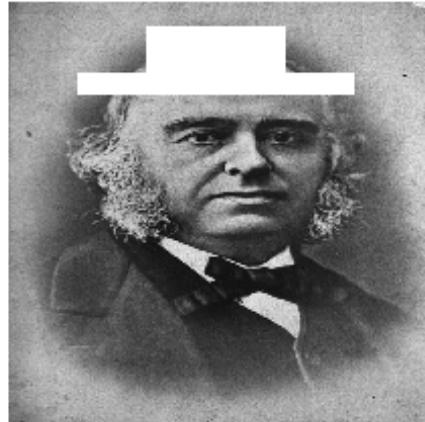


```
# If one wants to make the image to be completely black, one can also use col =  
gray (as opposed to col = gray.colors), though this function does not  
differentiate between all element being 0 or 1.  
image(black_matrix, col = gray(1/100), asp = 1)
```



## 2.d - Draw a white hat on the image from 2.b.

```
# From the original matrix we change the value from row 50 to 150 and from column  
150 to 450 to the value 1.  
matrix[50:150, 200:400] <- 1  
matrix[150:200, 100:500] <- 1  
  
# Then the image is printed using the rotate function and adjusting the scale.  
image(rotate(matrix), col = gray(1:100/100), axes = F, asp = 1)
```

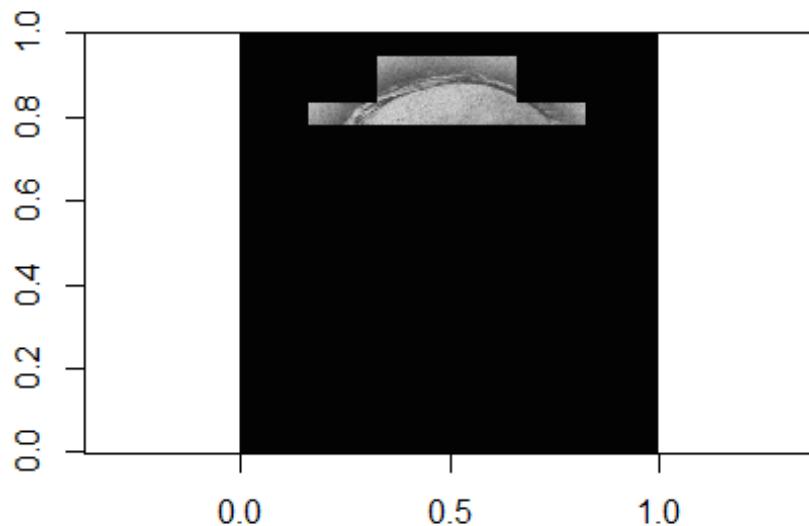


**2.e - Make an image which has the same dimensions as 2.b., and which only contains the parts which was hidden behind the hat in 2.d. The rest should be black.**

```
# Reloading the original matrix
matrix3 <- readJPEG("Matrices data.jpg", native = FALSE)

# Drawing the hat on the black matrix from 2.c
black_matrix[50:150, 200:400] <- 1
black_matrix[150:200, 100:500] <- 1
# Multiplying the black matrix with the real picture element-wise.
behindhat <- black_matrix * matrix3

# Printing the image and using the rotate function.
image(rotate(behindhat), col = gray(1:100/100), asp = 1)
```

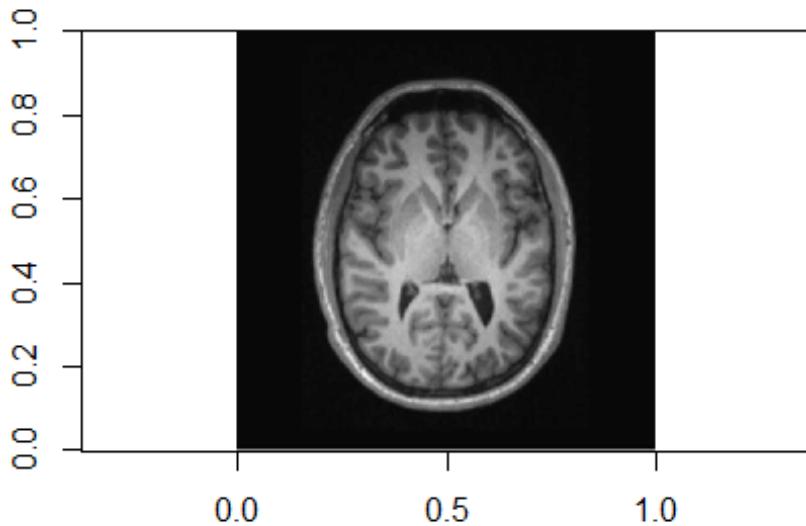


### 3: Brain and matrices

```
# Loading the data  
brain <- readJPEG("Matrices data2.jpg", native = FALSE)
```

#### 3.a - Make an image of the brain.

```
# Making the image of the brain  
image(rotate(brain), col = gray(1:100/100), asp = 1)
```



**3.b - We will attempt to find the interesting areas of this brain image, e.g. only areas with gray matter.**

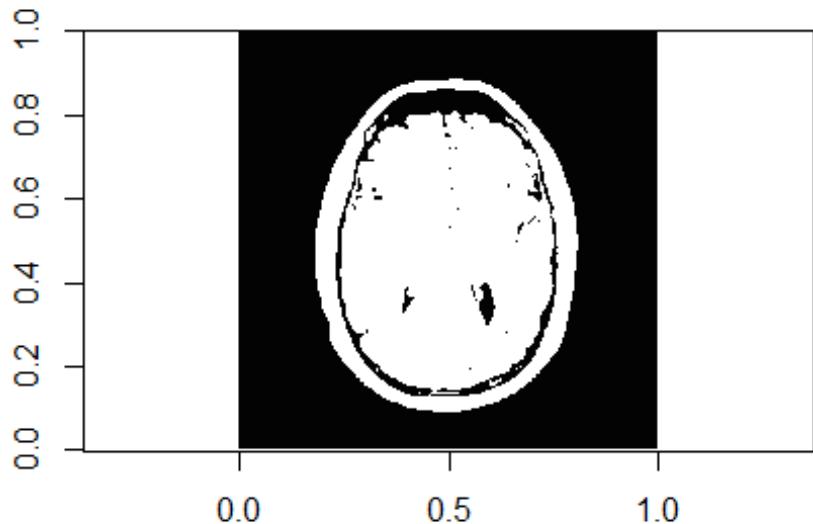
To do this we will create two masks, one that filters all darker areas away, and one that filters the white matter away. The masks will work by having zeros at the areas we want to filter away, and ones at the interesting areas. Thus, the mask will have the intended effect if we do element-wise multiplication it with the brain matrix. Start by making an image which is white (have ones) where the pixel values of the brain image are larger than the mean value of the whole image. Let the image be black (have zeros) everywhere else. Call this matrix mask1.

```
# Finding the mean value of the matrix and calling it mean_brain
mean_brain <- mean(brain)
mean_brain

## [1] 0.1869753

# Making the mask1 matrix consisting of the values from the brain matrix.
mask1 <- brain
# Changing the values bigger than the mean value to be 1.
mask1[mask1>mean_brain] <- 1
# Changing the values equal to or smaller than the mean value to be 0.
mask1[mask1<=mean_brain] <- 0

# To see the result the image of the mask matrix is printed.
image(rotate(mask1), col = gray(1:100/100), asp = 1)
```

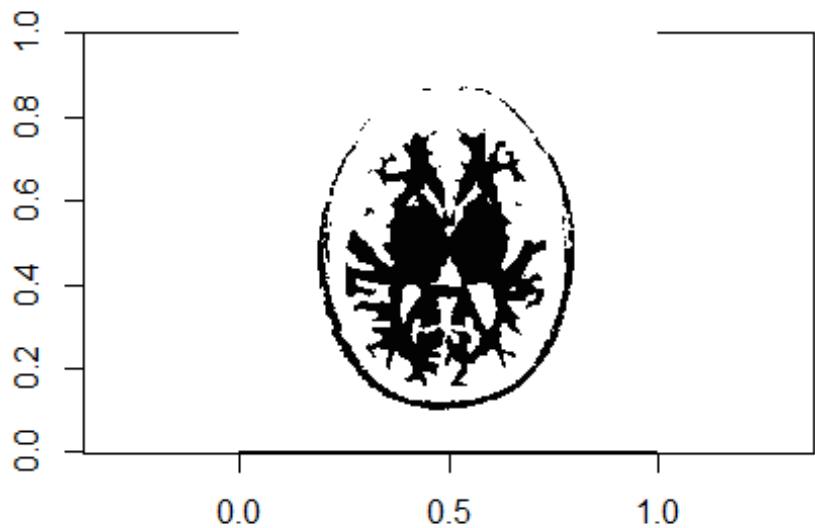


**3.c - Make an image which is white where the pixel values of the brain image are smaller than 2.5 times the mean value of the whole image. Call this matrix mask2.**

```
# Creating a second mask matrix on the base of the brain matrix
mask2 <- brain

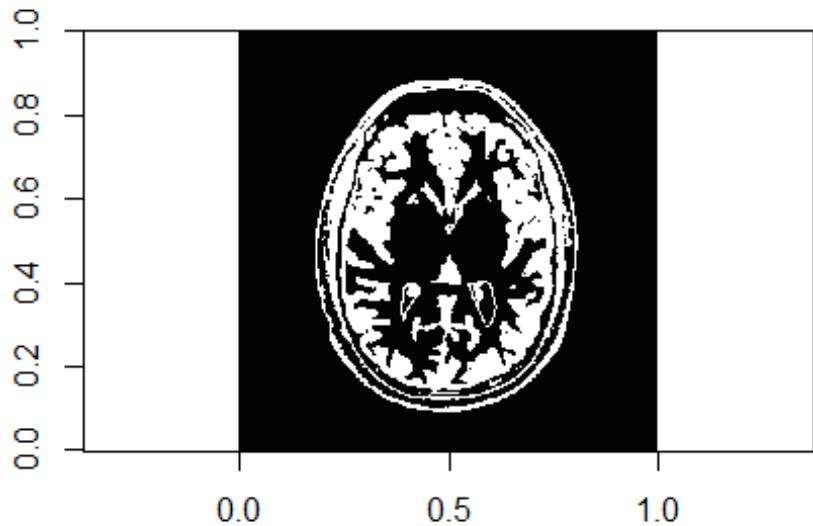
# Changing the elements of the mask2 matrix that are smaller than 2.5 times the
# mean value to be 1.
mask2[mask2<(2.5*mean_brain)] <- 1
mask2[mask2>1] <- 0

# To visualize the mask2 matrix is printed.
image(rotate(mask2), col = gray(1:100/100), asp = 1)
```



**3.d - Convert mask1 and mask2 into one mask with ones where the two masks overlap and zeros everywhere else. What type mathematical procedure can be used to produce this?**

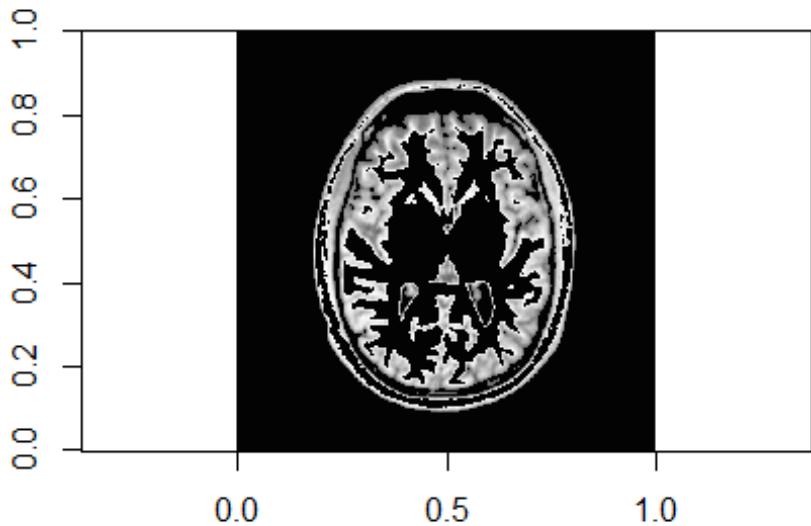
```
# Creating a third mask by doing a element-wise multiplication of mask1 and mask2,
# therefore making the areas that overlap white (X = 1) and the rest black (X = 0).
mask3 <- mask1*mask2
# Making the image to see the result
image(rotate(mask3), col = gray(1:100/100), asp = 1)
```



**3.e - Use the combined mask on the brain image to give you an image with only the image values where the mask has ones, and zeros everywhere else. Did we successfully limit our image to only contain gray matter?**

```
# Applying the combined mask to the brain image by again doing an element-wise multiplication
gray_matter <- brain*mask3

# Making the final brain image
image(rotate(gray_matter), col = gray(1:100/100), asp = 1)
```



### 3.f - Count the number of pixels in the combined mask.

```
# Number of pixels should be equal to the sum of all the elements that are equal to 1.
sum(mask3 == 1)
## [1] 50004
```

So the combined mask contains 50.004 pixels.

## 4: Two equations with two unknowns

### 4.a - In the friday bar, men were three times as likely as women to buy beer. A total of 116 beers were sold.

Women were twice as likely as men to buy wine. 92 glasses of wine were sold. How many men and women attended the Friday bar?

Starting by writing this up as two equations where M = men and W = women.  $W + 3M = 116$  beers  
 $2W + M = 92$  wine

```
# From this two matrices are created, where A contains the amount of sold beers and glasses of wine and B contains the relations of x's and y's from the equations.
```

```
A <- matrix(c(116, 92), nrow = 1, ncol = 2)
B <- matrix(c(1, 3, 2, 1), nrow = 2, ncol = 2)
```

```

# In order to solve the equations, the latter matrix needs to be inverted in order
to divide matrix B from both sides (in reality we're multiplying with the inverse
of matrix B).
invB <- solve(B)

# Finding the values for W and M by multiplying matrix A with the inverse of
matrix B.
C <- A%*%invB
C

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

```

From the calculations, one can infer that 32 women and 28 men attended the friday bar.

## Portfolio 3: fMRI regression

Group 3: Mie, Tobias, Riikka, Ruta and Jana

February 27, 2019

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

library(pacman)
p_load(tidyverse)

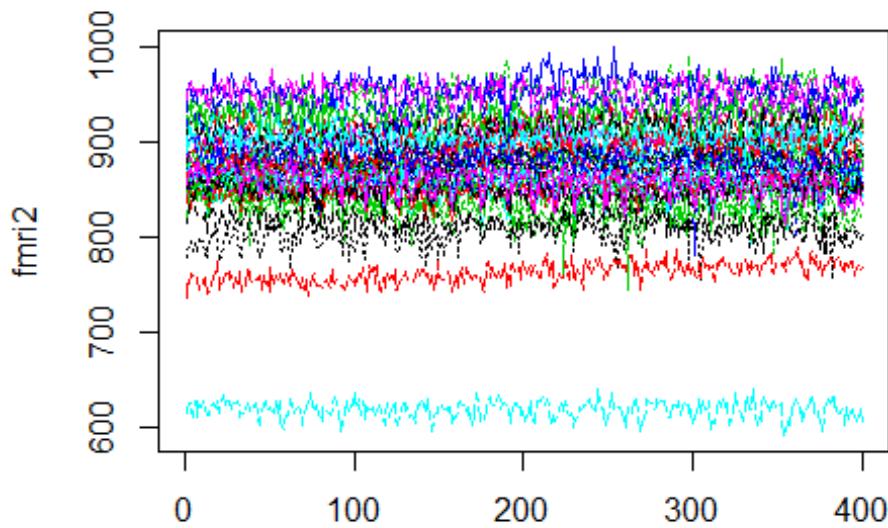
# Setting the working directory
setwd("C:/Users/Mie/Desktop/Cognitive science/Experimental methods
2/Assignments/Portfolio 3")

# Loading in the data
fmri<-as.matrix(read.csv("aud_fmri_data37.csv", header=FALSE))
# Making it a time-series
fmri2<-ts(fmri)
##design
fmrides<-as.matrix(read.csv("aud_fmri_design.csv", header=FALSE))
##making it a time-series
fmrides2<-ts(fmrides)
```

### 1: Make two figures:

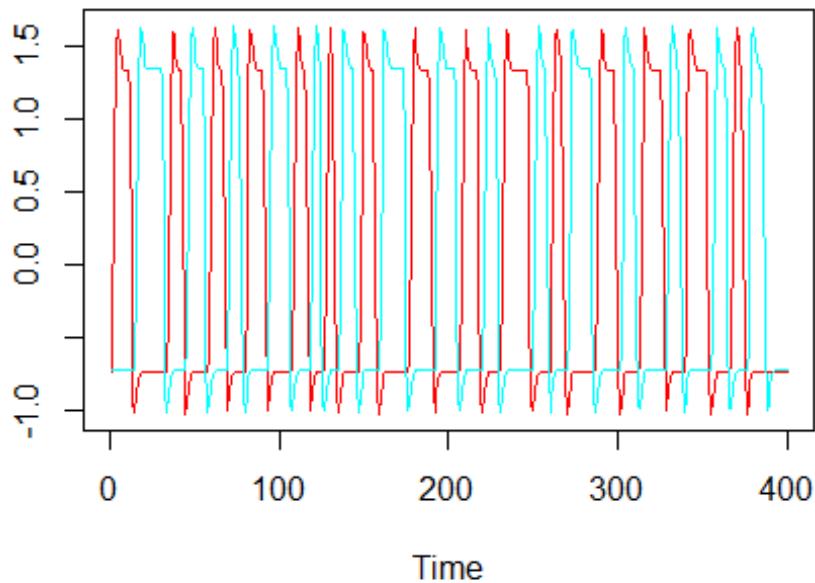
1.a. A figure with lineplots of the data from all participants as a function of time in one figure. Note how much the baseline signal can vary between participants.

```
# Using the ts.plot function to plot the lineplots of the data as a function of
time.
matplot(fmri2, type = "l")
```



### 1.b. A figure lineplots with the model covariates.

```
# Time series plot with the design matrix  
ts.plot(fmrides2, col = rainbow(2))
```



## 2. How many stories did the participants listen to in each condition?

As can be seen in the lineplot of the model covariates is that there were two conditions that each had 15 stories per participant.

## 3: Investigating the model

### 3.a. Are the two model covariates correlated?

```
# Making the fmrides matrix into a dataframe
design <- as.data.frame(fmrides)
# Correlation test of the two model covariates
r <- cor.test(design$V1, design$V2)
r

##
## Pearson's product-moment correlation
##
## data: design$V1 and design$V2
## t = -12.894, df = 398, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.6084775 -0.4697617
```

```
## sample estimates:  
##      cor  
## -0.5428111
```

As can be seen from the output the two model covariates are negatively correlated,  $r(398) = -0.54$ ,  $p < .00001$ . The result of the correlation test therefore indicate that the two covariates have a large effect and this is significant.

### 3.b. Have the covariates been mean-centered?

```
# Checking the means of the covariates to see if they the data is centered around 0.  
  
mean(design$V1)  
  
## [1] -1.34875e-06  
  
mean(design$V2)  
  
## [1] 2.045e-06
```

Evidently, the means of both covariates are very close to 0 and we therefore conclude them to have been mean-centered.

### 4: Please report the percentage of shared variance in the two covariates.

```
# Finding the percentage of shared variance by finding r-squared.  
percentage <- r$estimate[1]^2*100  
percentage  
  
##      cor  
## 29.46439
```

So the two covariates share 29.46 percent of the variance.

### 5: Pick one participant's data set.

```
# Converting fmri into a data frame  
df <- as.data.frame(fmri)  
participant9 <- df$V9  
# Saving the data from participant 9 into a new data frame  
participant <- as.data.frame(participant9)  
participant$time <- c(1:400)
```

### 5.a - Fit the model as it is, including intercept.

```
# Linear model of the participants brain activity predicted by time  
model1 <- lm(V9~fmrides, data = df)  
summary(model1)
```

```

## 
## Call:
## lm(formula = V9 ~ fmrides, data = df)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## -44.041 -9.916  0.461  9.858 38.331 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 952.3775   0.6903 1379.577 < 2e-16 ***
## fmridesV1    2.4223   0.8230   2.943  0.00344 **  
## fmridesV2    0.9929   0.8230   1.206  0.22836    
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 13.81 on 397 degrees of freedom
## Multiple R-squared:  0.02188,   Adjusted R-squared:  0.01695 
## F-statistic:  4.44 on 2 and 397 DF,  p-value: 0.01239

```

### 5.b - Fit the model as it is, excluding intercept.

```

# Linear model without intercept
model2 <- lm(V9 ~ fmrides - 1, data = df)
summary(model2)

## 
## Call:
## lm(formula = V9 ~ fmrides - 1, data = df)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## 908.3  942.5  952.8  962.2  990.7 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## fmridesV1    2.4220   56.9183   0.043   0.966    
## fmridesV2    0.9947   56.9180   0.017   0.986    
## 
## Residual standard error: 954.9 on 398 degrees of freedom
## Multiple R-squared:  4.662e-06,   Adjusted R-squared: -0.00502 
## F-statistic: 0.0009278 on 2 and 398 DF,  p-value: 0.9991

```

### 5.c - Fit only the 1st covariate as a model.

```

# Linear model with the first covariate as a second predictor
model3 <- lm(V9 ~ design$V1, data = df)
summary(model3) # V1 significantly explains the data, V1 estimate = 1.88, p = .00675.

```

```

## 
## Call:
## lm(formula = V9 ~ design$V1, data = df)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -44.009 -10.009   0.124   9.997  37.841 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 952.3775    0.6907 1378.788 < 2e-16 ***
## design$V1     1.8834    0.6916   2.723  0.00675 **  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## Residual standard error: 13.81 on 398 degrees of freedom
## Multiple R-squared:  0.01829, Adjusted R-squared:  0.01583 
## F-statistic: 7.416 on 1 and 398 DF, p-value: 0.00675

```

### 5.d - Fit only the 2nd covariate as a model.

```

# Linear model with the second covariate as a second predictor
model4 <- lm(V9 ~ design$V2, data = df)
summary(model4) # The V2 is not significant, estimate = -0.322, p = 0.645.

## 
## Call:
## lm(formula = V9 ~ design$V2, data = df)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -45.240 -10.179   0.377  10.327  38.391 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 952.3775    0.6970 1366.485 <2e-16 ***
## design$V2     -0.3220    0.6978   -0.461   0.645  
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## Residual standard error: 13.94 on 398 degrees of freedom
## Multiple R-squared:  0.0005345, Adjusted R-squared:  -0.001977 
## F-statistic: 0.2129 on 1 and 398 DF, p-value: 0.6448

```

### 5.e - Fit the 2nd covariate to the residuals from analysis 5.c., the 1st covariate only analysis

```

# Fitting the 2nd covariate to the residuals from model3 in 5.
residuals <- model3$residuals

```

```

model5 <- lm(residuals ~ design$V2)
summary(model5) # V2 estimate: 0.70, p = .31.

##
## Call:
## lm(formula = residuals ~ design$V2)
##
## Residuals:
##    Min     1Q   Median     3Q    Max
## -44.308 -9.989   0.312   9.864  38.345
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.432e-06 6.898e-01  0.000   1.000
## design$V2    7.004e-01 6.907e-01  1.014   0.311
##
## Residual standard error: 13.8 on 398 degrees of freedom
## Multiple R-squared:  0.002577, Adjusted R-squared:  7.05e-05
## F-statistic: 1.028 on 1 and 398 DF, p-value: 0.3112

```

### 5.f - Fit the 1st covariate to the residuals from 5.d., the 2nd covariate only analysis

```

# Fitting the 1st covariate to model4 from 5.d
residuals2 <- model4$residuals
model6 <- lm(residuals2 ~ design$V1)
summary(model6) # V1 estimate: 2.467, p = .014.

##
## Call:
## lm(formula = residuals2 ~ design$V1)
##
## Residuals:
##    Min     1Q   Median     3Q    Max
## -43.999 -9.963   0.104   10.305  37.682
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.304e-06 6.917e-01  0.000   1.000
## design$V1   1.709e+00 6.926e-01  2.467   0.014 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.83 on 398 degrees of freedom
## Multiple R-squared:  0.01506, Adjusted R-squared:  0.01259
## F-statistic: 6.087 on 1 and 398 DF, p-value: 0.01404

```

### 5.g - Does the order in which the predictor variables are fitted to the data matter for the estimates? If it does, what can explain this?

The order of the predictors matter because the predictors overlap in the variance that they explain. So, in model 5 some of the activity that is explained by the predictor V1 could also have been explained by the predictor V2, but since it is not a part of the residual variance we get a different beta estimate than we get when the order is reversed in model 6. The shared variance between V1 and V2 is the baseline condition.

## 6: Group level analyses

**Fit the full model to each of the 37 participants' data and extract the coefficients for each participant. (hint: the full participant data frame can be set as outcome. Alternatively, you can change the data structure and use lmList from assignment 1).**

```
# Fitting the model for each of the participant in the fmri2 time-series dataframe
# using the predictors V1 and V2 from the design dataframe
full_model <- lm(fmri2 ~ design$V1 + design$V2)

# Getting the coefficients for all the participants and transposing it into a
# dataframe.
estimates <- t(full_model$coefficients)
```

**6.a. Test the two individual hypotheses that the set of coefficient from each covariate is different from zero across the whole group (similar to assignment 1). Make a contrast that investigates the difference between the two covariates, i.e. the two types of stories (hint: subtraction).**

```
# Converting the matrix into a dataframe
estimates <- as.data.frame(estimates)
# Renaming the columns for the covariates
colnames(estimates)[2] <- "V1"
colnames(estimates)[3] <- "V2"

# T-test of the coefficients from covariate V1 for all the participants being
# different than 0.
t.test(estimates$V1, mu=0)

##
## One Sample t-test
##
## data: estimates$V1
## 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
```

```

# T-test of the coefficients from covariate V2 for all the participants being
# different than 0.
t.test(estimate$V2, mu = 0)

##
## One Sample t-test
##
## data: estimate$V2
## 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

# Adding the contrast between the covariates to the dataframe
estimate$contrast <- estimate$V1 - estimate$V2

```

### 6.b. Test the hypothesis that the contrast is different from zero across participants.

```

# T-test of the contrast being different from 0.
t.test(estimate$contrast, mu = 0)

##
## One Sample t-test
##
## data: estimate$contrast
## t = 0.69615, df = 36, p-value = 0.4908
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## -0.1287197 0.2632734
## sample estimates:
## mean of x
## 0.06727684

```

### 6.c. Make a bar diagram including the mean effect of the two coefficients and the contrast, including error bars (indicating standard error of mean).

```

# Making 3 different dataframes corresponding to the data for the 2 covariates and
# the contrast between them
fiction <- select(estimate, V1)
factual <- select(estimate, V2)
contrast_df <- select(estimate, contrast)

# Barplot of effects of the different predictors and the contrast with error bars.
ggplot() +
  geom_bar(data = fiction, aes(x=1, y = V1), stat="summary", fun.y=mean,
  fill="pink", color="black") +

```

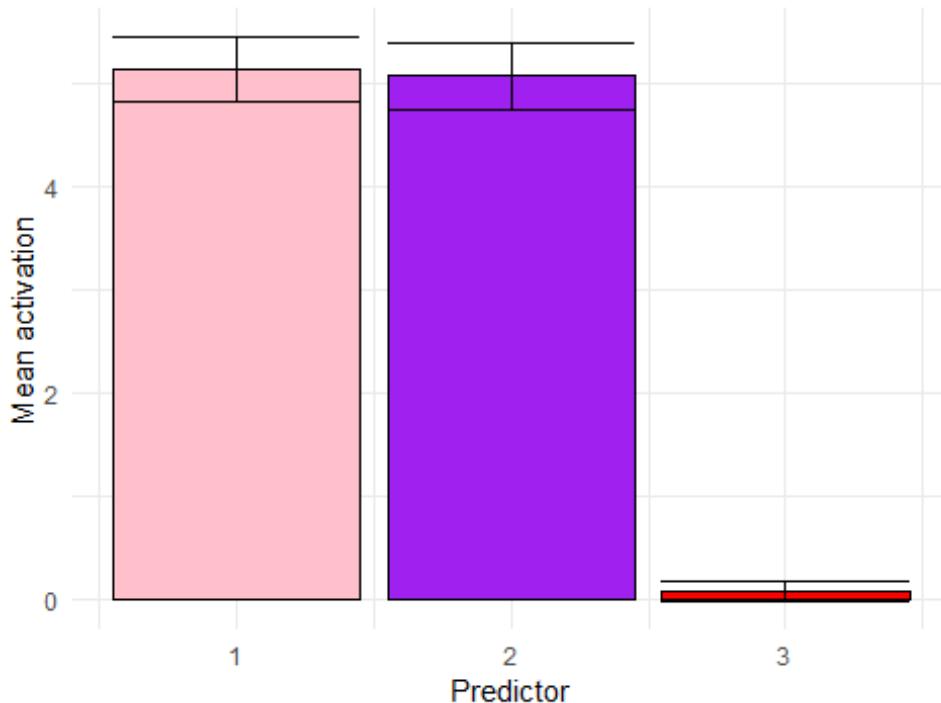
```

geom_errorbar(data= fiction, aes(x=1, y=V1), stat="summary", fun.data=mean_se) +
  geom_bar(data = factual, aes(x=2, y = V2), stat="summary",fun.y=mean,
  fill="purple", color="black") +
  geom_errorbar(data= factual, aes(x=2, y=V2), stat="summary", fun.data=mean_se) +
  geom_bar(data = contrast_df, aes(x = 3, y = contrast), stat="summary", fun.y =
  mean, fill = "red", color = "black")+
  geom_errorbar(data= contrast_df, aes(x=3, y=contrast), stat="summary",
  fun.data=mean_se)+  

  labs(title = "Mean effect of V1, V2 and contrast", x = "Predictor", y = "Mean
activation") +
  theme_minimal()

```

Mean effect of V1, V2 and contrast



## 7: Adding a covariate

7.a - For each participant, add a covariate that models the effect of time (hint: 1:400).

```

# We create a time vector from 1 to 400
time <- rep(1:400)
# And add it to the full_model from before
full_model2 <- lm(fmri2 ~ time + design$V1 + design$V2)

```

7.b - Does that improve the group results in term of higher t-values?

```

# Getting the coefficients for all the participants and transposing it into a
dataframe.

```

```

estimates2 <- t(full_model2$coefficients)

# Converting the matrix into a dataframe
estimates2 <- as.data.frame(estimates2)
# Renaming the columns for the covariates
colnames(estimates2)[3] <- "V1"
colnames(estimates2)[4] <- "V2"

# Adding the contrast between the covariates to the dataframe
estimates2$contrast <- estimates2$V1-estimates2$V2

# T-test of the coefficients from covariate V1 for all the participants being
# different than 0.
t.test(estimates2$V1, mu=0)

##
## One Sample t-test
##
## data: estimates2$V1
## 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

# T-test of the coefficients from covariate V2 for all the participants being
# different than 0.
t.test(estimates2$V2, mu = 0)

##
## One Sample t-test
##
## data: estimates2$V2
## 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

```

For the t-test in question 6, we got  $t(36) = 16.6$ ,  $p < .00001$  for the slopes of V1. In the t-test adding the time as a covariate we got  $t(36) = 18.3$ ,  $p < .00001$ . Likewise, for the first t-test of covariate V2  $t(36) = 15.6$ ,  $p < .00001$ , whereas the t-value again increased in the second t-test,  $t(36) = 17.6$ ,  $p < .00001$ . Evidently, in both cases the t-value increases when adding the time as a covariate to the model, making it less likely for the the mean to be 0.

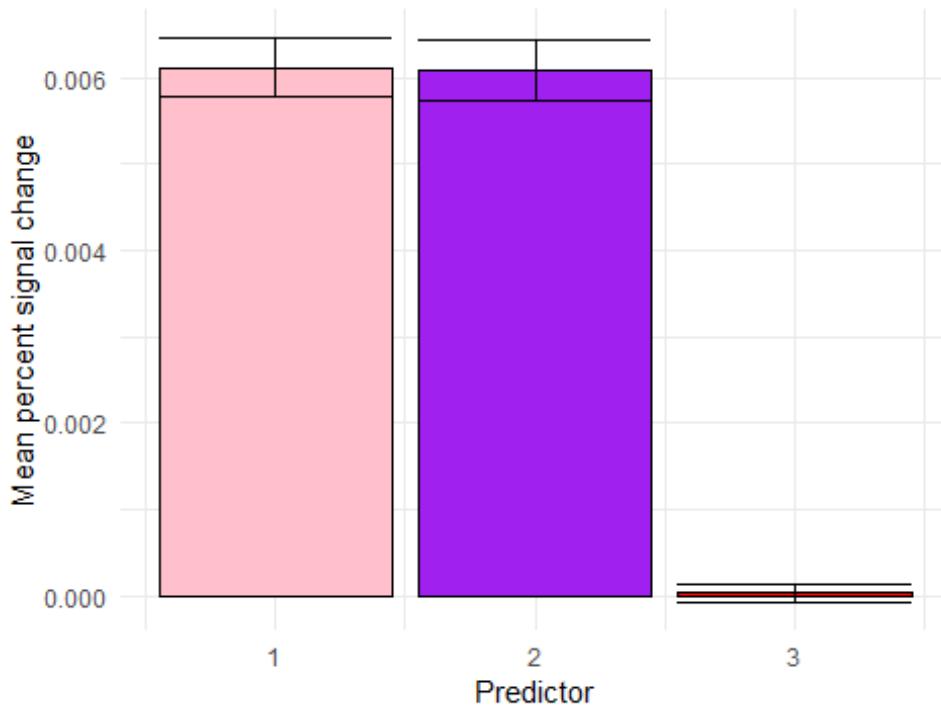
## 8. Make a bar diagram like the above, but display effects as percent signal change (hint: percent signal change is slope divided by intercept)

```
# Making 2 new columns in the estimates2 dataframe corresponding to the percent signal change of the two covariates
estimates2$percentageV1 <- estimates2$V1 / estimates2$`(Intercept)`` 
estimates2$percentageV2 <- estimates2$V2 / estimates2$`(Intercept)`` 
estimates2$percentagecontrast <- estimates2$contrast / estimates2$`(Intercept)`` 

# Making the two new columns into two different dataframes
percentageV1 <- select(estimates2, percentageV1)
percentageV2 <- select(estimates2, percentageV2)
percentageContrast <- select(estimates2, percentagecontrast)

# Barplot of the mean percent signal change for the two covariate with errorbars.
ggplot() +
  geom_bar(data = percentageV1, aes(x=1, y = percentageV1),
stat="summary", fun.y=mean, fill="pink", color="black") +
  geom_errorbar(data= percentageV1, aes(x=1, y=percentageV1), stat="summary",
fun.data=mean_se) +
  geom_bar(data = percentageV2, aes(x=2, y = percentageV2),
stat="summary", fun.y=mean, fill="purple", color="black") +
  geom_errorbar(data= percentageV2, aes(x=2, y=percentageV2), stat="summary",
fun.data=mean_se) +
  geom_bar(data = percentageContrast, aes(x = 3, y = percentagecontrast),
stat="summary", fun.y = mean, fill = "red", color = "black") +
  geom_errorbar(data = percentageContrast, aes(x=3, y=percentagecontrast),
stat="summary", fun.data=mean_se) +
  labs(title = "Percent signal change of the covariates V1 and V2", x =
"Predictor", y = "Mean percent signal change") +
  theme_minimal()
```

**Percent signal change of the covariates V1 and V2**



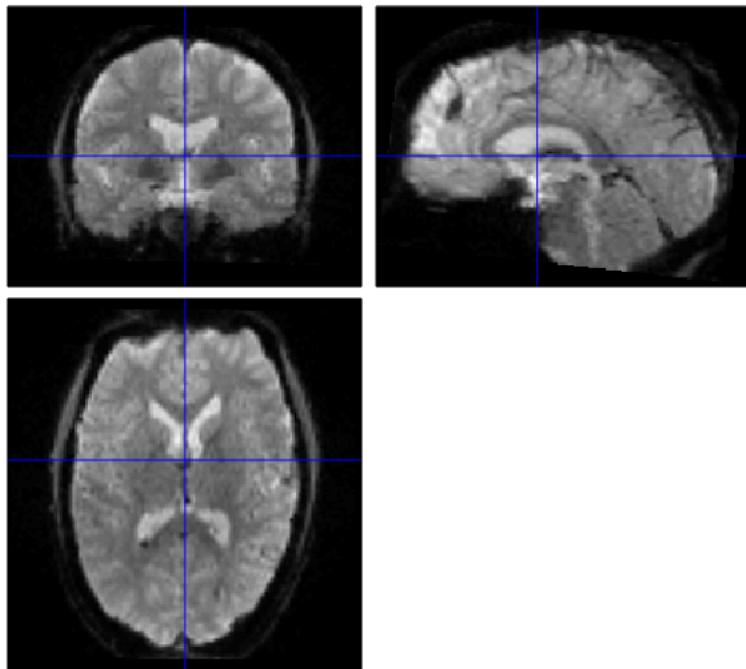
## Portfolio 4: fMRI preprocessing

Group 3: Mie, Tobias, Riikka, Ruta and Jana

March 6, 2019

### 1: Initial alignment of data to standard stereotaxic space (MNI-space)

#### 1.a How much does it have to be moved (indicate 3 translations and 3 rotations)?

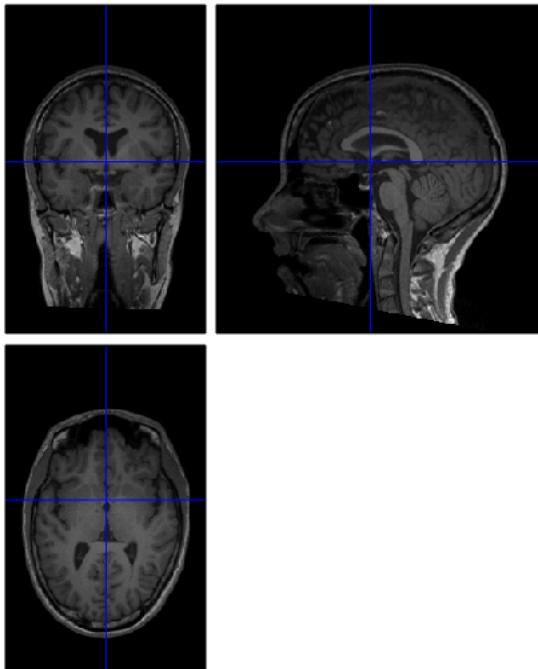


Crosshair Position	Origin
mm:	0.0 0.0 0.0
vx:	48.3 55.1 29.7
Intensity:	756.501
right {mm}	-9.3
forward {mm}	-18
up {mm}	-15.5
pitch {rad}	0.05
roll {rad}	-0.06
yaw {rad}	-0.03
resize {x}	1
resize {y}	1
resize {z}	1
<input type="button" value="Set Origin"/>	<input type="button" value="Reorient..."/>

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: 48.3 55.1 29.7  
Dir Cos: 1.000 0.007 0.021  
-0.004 0.994 -0.111  
-0.022 0.111 0.994

Full Volume	Hide Crosshair
World Space	Trilinear interp.
Auto Window	Add Overlay...

**1.b. Align the anterior commisure of the structural image to [0,0,0]. How much does that have to be moved?**



Crosshair Position

mm:	0.0 0.0 0.0
vx:	119.9 133.7 89.0
Intensity:	467.193

right {mm}	-7.2096
forward {mm}	-45.458
up {mm}	10.5943
pitch {rad}	0
roll {rad}	0
yaw {rad}	0
resize {x}	1
resize {y}	1
resize {z}	1

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

Vox size: -0.977 x 0.977 x 1  
Origin: 120 134 89  
Dir Cos: -0.002 0.017 1.000  
0.982 -0.189 0.006  
0.189 0.982 -0.016

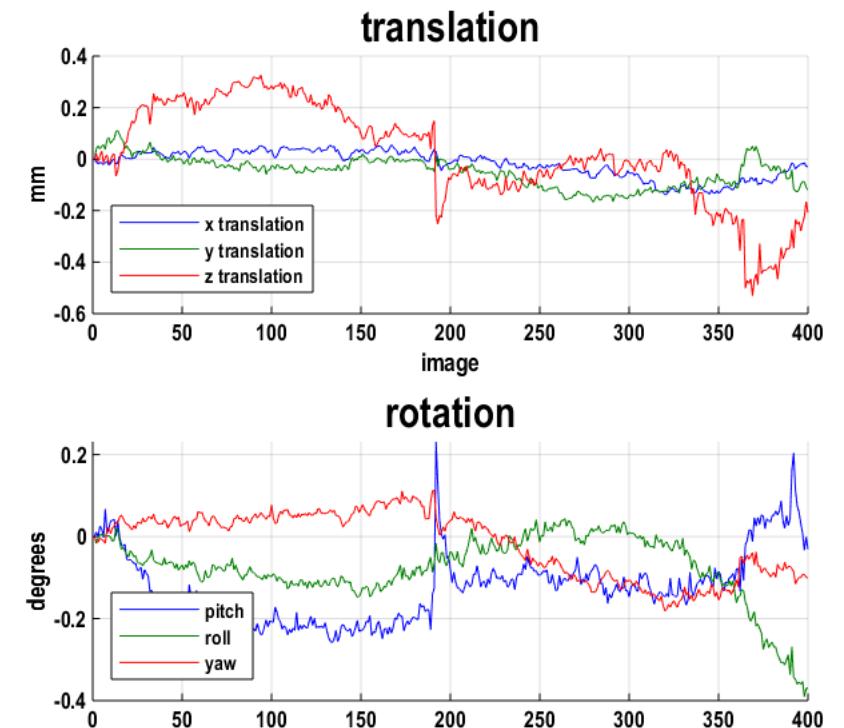
Full Volume   
World Space   
Auto Window

## 2: Preprocessing of fMRI data

### 2.a. Realignment

#### Image realignment

```
1 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
2 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
3 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
4 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
5 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
6 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
7 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
8 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
9 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
10 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
11 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
12 C:\Users\Me\Documents\MATLAB\Portfolio 4\Functional\fSubj
..... etc
```



## 2.b. Coregistration

### Normalised Mutual Information Coregistration

$$X_1 = 0.000^*X + 0.000^*Y - 0.500^*Z + 92.900$$

$$Y_1 = -0.488^*X - 0.004^*Y - 0.000^*Z + 117.239$$

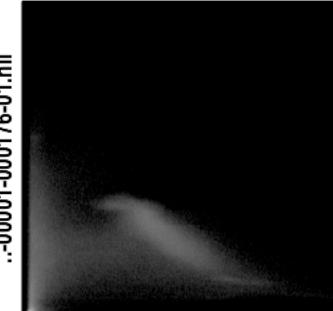
$$Z_1 = -0.004^*X + 0.488^*Y + 0.000^*Z - 40.237$$

Original Joint Histogram

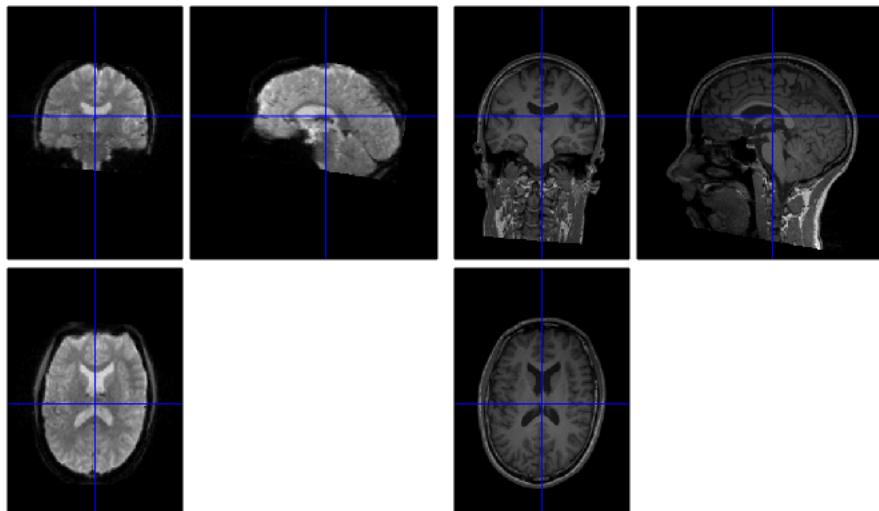


..-00001-00001-01.nii

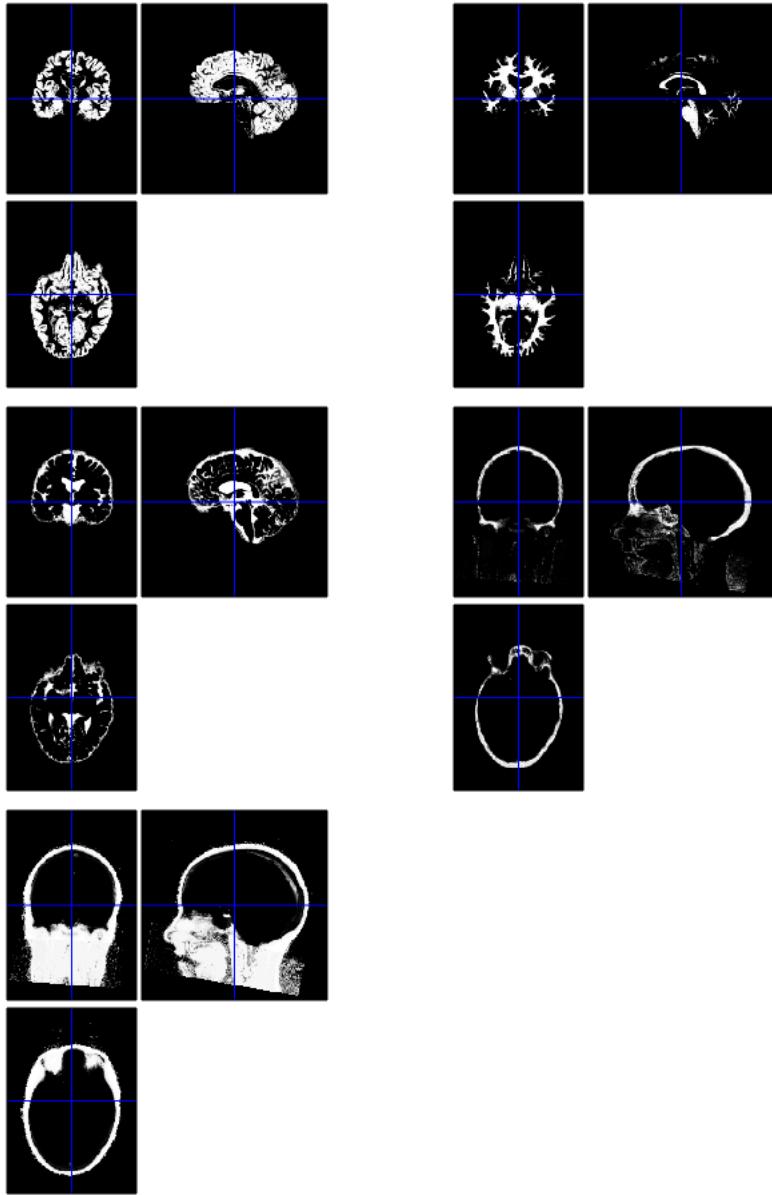
Final Joint Histogram



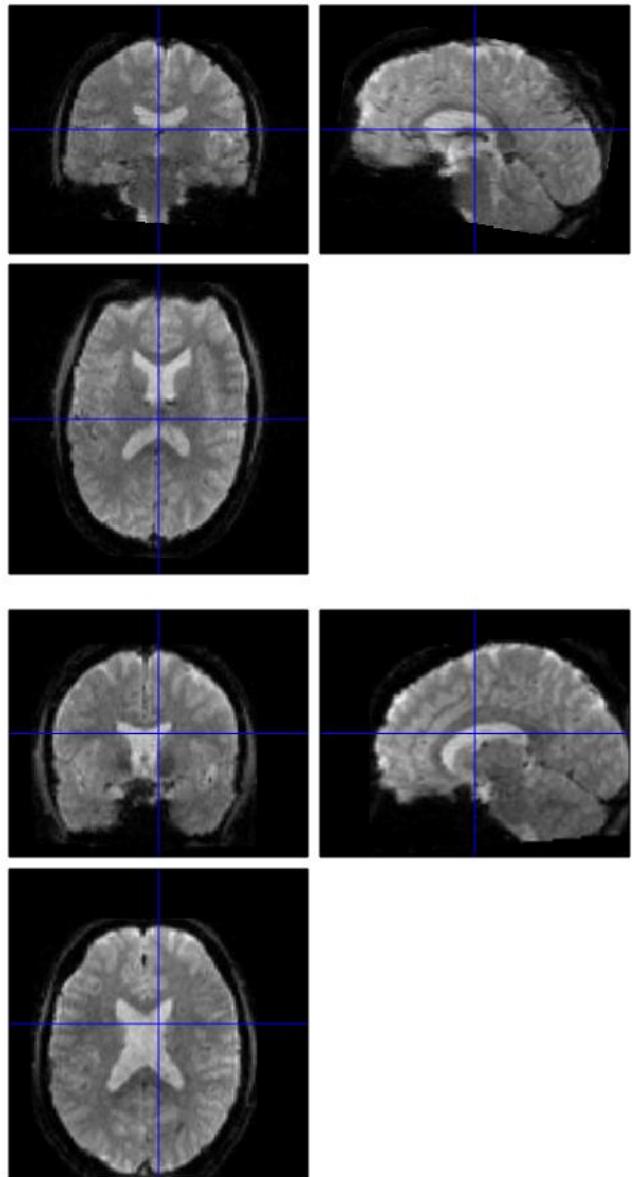
..-00001-00001-01.nii



### 2.c. Segmentation

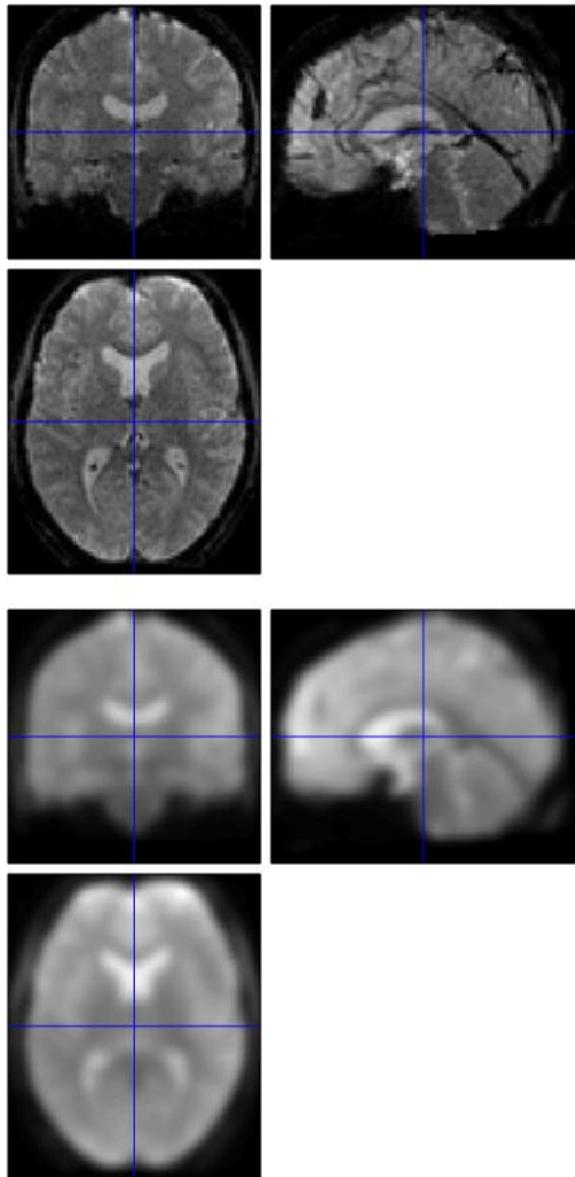


## 2.d. Normalization



Here we chose a functional image (top) and match it with the normalized version of the same image (bottom).

## 2.e. Smoothing



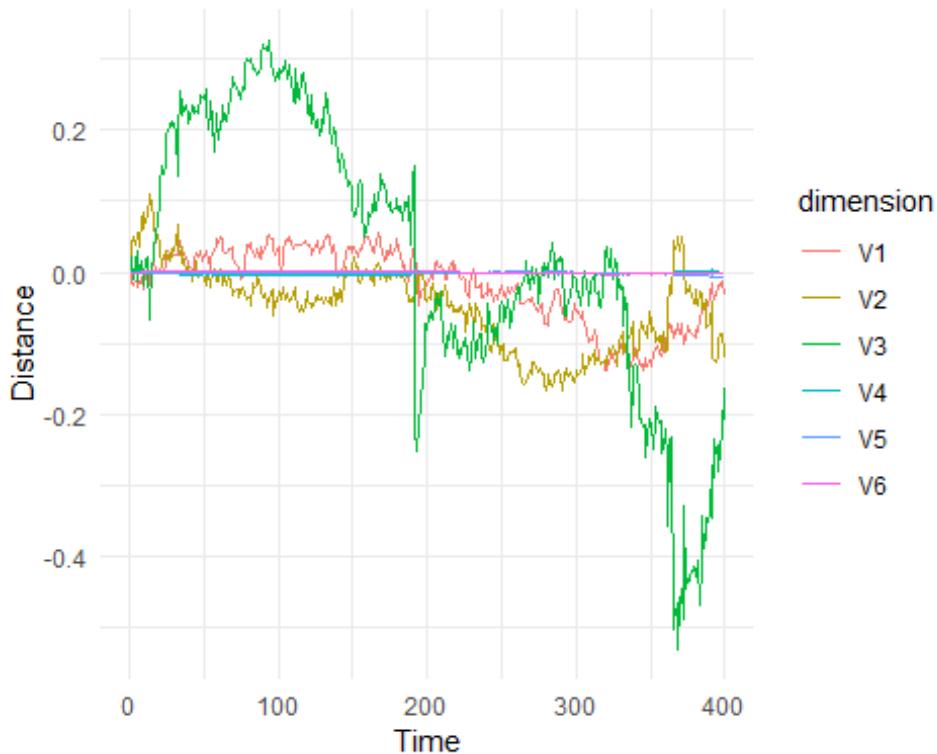
Here we chose a normalized image and compared it to the smoothed version of the same image.

### 3: Realignment parameters

#### 3.a. Make a lineplot of the realignment parameters in R.

```
# Changing the format of the table to long format so plotting is easier
long_format <- stack(df, select = c(V1, V2, V3, V4, V5, V6))
names(long_format)[2] <- "dimension" # renaming of a column
long_format$time <- rep(1:400) # creating a column for time

lineplot <- ggplot(long_format, aes(x = time, y = values, colour = dimension)) #
# Line plot
lineplot + geom_line() + theme_minimal() +
  labs(x = "Time", y = "Distance")
```



#### 3.b. How far has the participant moved for each dimension during the experiment (Hint: use "apply()" to run a function across columns)?

```
# Apply sum function across columns (therefore choosing MARGIN = 2)
apply(X = df, MARGIN = 2, FUN = sum)
```

```
##           V1           V2           V3           V4           V5           V6
## -7.4305348 -19.4862866  7.4245363 -0.8709879 -0.5339383 -0.1197184
```

### 3.c. Are any of the realignment parameters significantly correlated with the fMRI model (same model as used in exercise 3)?

```
# Loading the design model from portfolio 3
fmrides <- as.data.frame(read.csv("aud_fmri_design.csv", header=FALSE))

# Correlation tests for all the columns with each of the covariate from the model
cor.test(fmrides$V1, df$V1)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and df$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

cor.test(fmrides$V2, df$V1)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and df$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

cor.test(fmrides$V1, df$V2)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and df$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
```

```

cor.test(fmrides$V2, df$V2)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and df$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

cor.test(fmrides$V1, df$V3)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and df$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

cor.test(fmrides$V2, df$V3)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and df$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

cor.test(fmrides$V1, df$V4)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and df$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

cor.test(fmrides$V2, df$V4)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and df$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

cor.test(fmrides$V1, df$V5)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and df$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

cor.test(fmrides$V2, df$V5)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and df$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

```

```

cor.test(fmrides$V1, df$V6)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and df$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

cor.test(fmrides$V2, df$V6)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and df$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

```

Only V5 is significantly correlated with the covariate V1,  $r(398) = 0.17$ ,  $p = .0005$ .

**Remove linear effects of time from the realignment parameters (hint: 1:400, fit a line and use residuals).**

```

df$time <- rep(1:400)

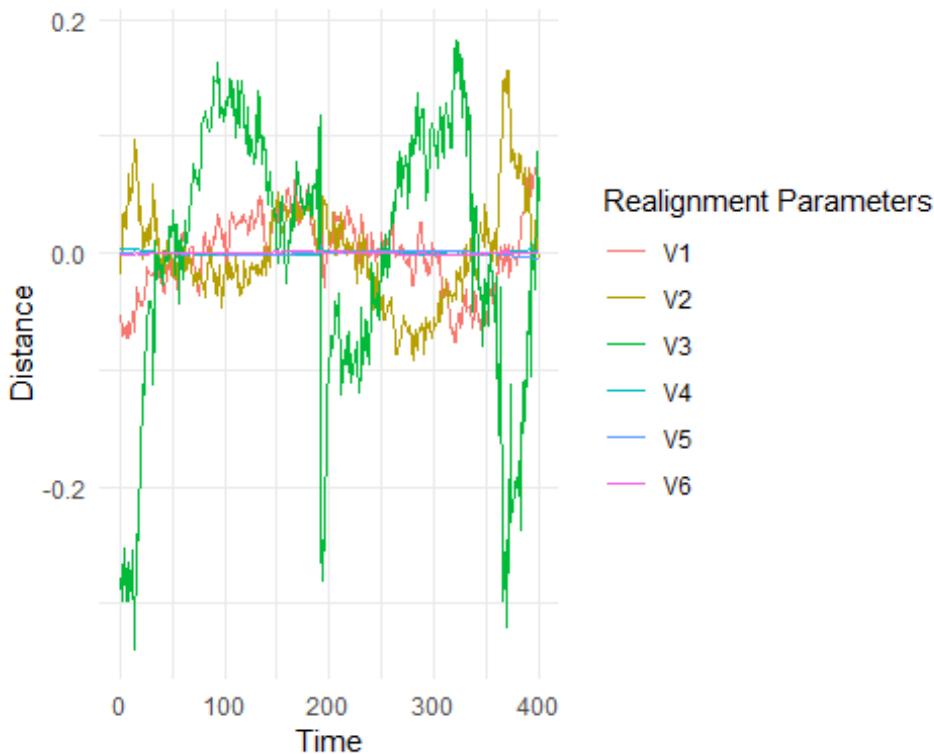
#make new dataframe for residuals
residuals <- as.data.frame(rep(1:400))
colnames(residuals)[1] <- "time"

# Running lm for each column for the variance predicted by time and adding the
# variance not explained by the line to a new dataframe called residuals
for (i in colnames(df)){
  if (i == "time"){ # Excluding the time column from the model
    break
  }
  else{
    residuals[,i] <- lm(data = df, df[,i] ~ time)$residuals
  }
}

```

### 3.d. Make a lineplot of the realignment parameters with time removed.

```
# Using the melt function to make the residual table into a data frame
m_residuals <- melt(residuals, id.vars="time")
# Plotting the data
ggplot(m_residuals, aes(x=time, y=value, group=variable, colour=variable)) +
  geom_line() +
  theme_minimal() +
  labs(x = "Time", y = "Distance", color = "Realignment Parameters")
```



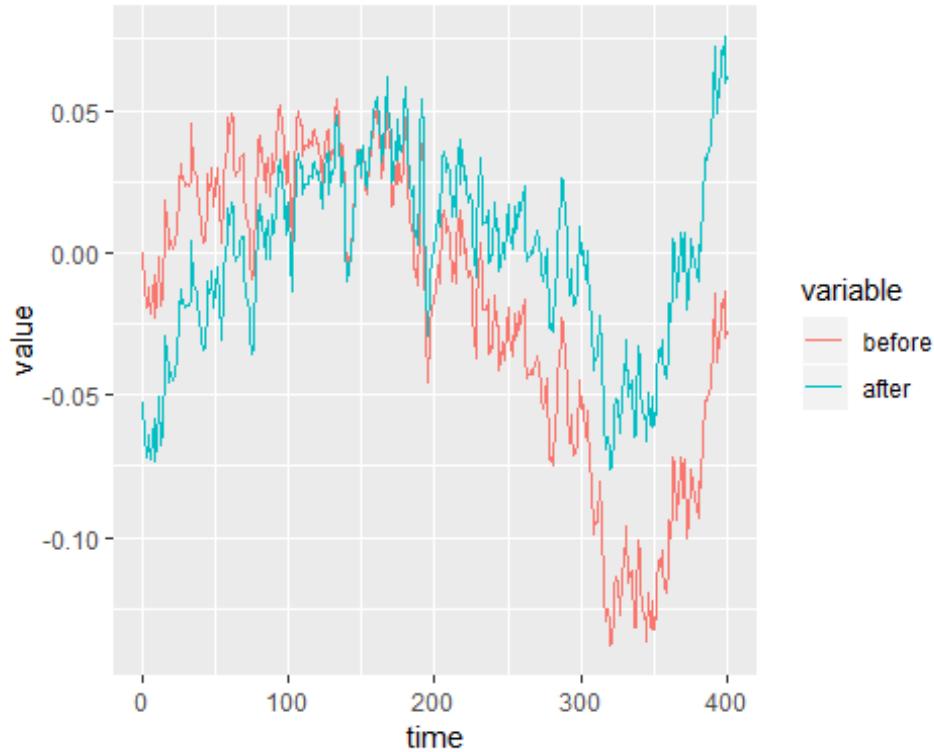
### 3.e. Make a lineplot including only the first realignment parameter before and after removal.

```
# Making a new dataframe taking only the first realignment parameter (V1) from the
# df and after time was removed.
V1 <- as.data.frame(df$V1)
colnames(V1)[1] <- "before"
V1$time <- rep(1:400)
V1$after <- residuals$V1

# Melt for plotting
m_V1 <- melt(V1, id.vars="time")

# Plotting the first realignment parameter before and after removal of variance
# explained by time
```

```
ggplot(m_V1, aes(x = time, y=value, group=variable, colour = variable)) +
  geom_line()
```



### 3.f. Are the realignment parameters (corrected for effects of time) now correlated with the fMRI model?

```
cor.test(fmrides$V1, residuals$V1)

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

cor.test(fmrides$V2, residuals$V1)

##
## Pearson's product-moment correlation
##
```

```

## data: fmrides$V2 and residuals$V1
## 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

cor.test(fmrides$V1, residuals$V2)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and residuals$V2
## 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

cor.test(fmrides$V2, residuals$V2)

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

cor.test(fmrides$V1, residuals$V3)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and residuals$V3
## 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

cor.test(fmrides$V2, residuals$V3)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and residuals$V3
## 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

cor.test(fmrides$V1, residuals$V4)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and residuals$V4
## 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

cor.test(fmrides$V2, residuals$V4)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and residuals$V4
## 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

cor.test(fmrides$V1, residuals$V5)

##
## Pearson's product-moment correlation
##

```

```

## data: fmrides$V1 and residuals$V5
## 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

cor.test(fmrides$V2, residuals$V5)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and residuals$V5
## 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

cor.test(fmrides$V1, residuals$V6)

##
## Pearson's product-moment correlation
##
## data: fmrides$V1 and residuals$V6
## 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

cor.test(fmrides$V2, residuals$V6)

##
## Pearson's product-moment correlation
##
## data: fmrides$V2 and residuals$V6
## 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
```

Now two of the parameters are significantly correlated with the covariates, where the residuals of parameter V1 is significantly correlated to the first covariate V1,  $r(398) = -0.16$ ,  $p = .0012$ , and the residuals of parameter V5 is significantly correlated again to the first covariate V1,  $r(398) = 0.17$ ,  $p = .0006$ .

# Portfolio 5: Model specification

Group 3: Mie, Tobias, Riikka, Ruta and Jana

March 6, 2019

## 1: Checking input using R

Test the following hypotheses:

**1.a. There was a significant difference between the durations of the two story types.**

```
# Making to vectors for the durations of the two different story types
duration1 <- c(35, 27, 27, 36, 26, 16, 29, 42, 33, 54, 22, 38, 43, 43, 21)
duration2 <- c(55, 33, 23, 37, 16, 30, 48, 40, 26, 24, 46, 27, 25, 27, 30)

# Testing whether they differ significantly
t.test(duration1, duration2)

##
## Welch Two Sample t-test
##
## data: duration1 and duration2
## 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
```

The two story type durations did not significantly differ,  $t(27.9) = 0.09$ ,  $p > .9$ .

**1.b. There was a significant difference between the ratings of the two story types.**

```
# Making to vectors for the ratings of the two different story types
Rating1 <- c(7, 4, 4, 5, 3, 1, 6, 2, 1, 4, 2, 3, 2, 4, 1)
Rating2 <- c(8, 2, 3, 2, 5, 7, 7, 3, 3, 2, 3, 3, 2, 4, 3)

# Testing whether they differ significantly
t.test(Rating1, Rating2)

##
## Welch Two Sample t-test
##
## data: Rating1 and Rating2
## 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
```

```

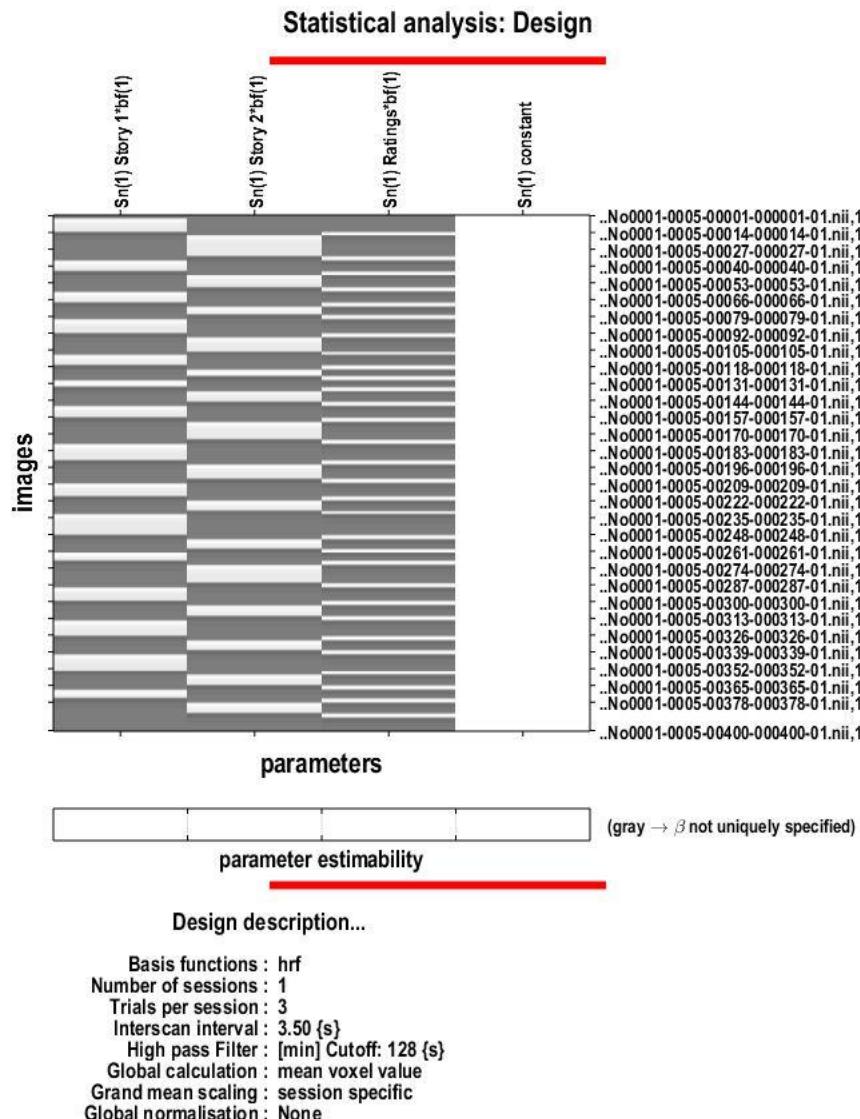
## sample estimates:
## mean of x mean of y
## 3.266667 3.800000

```

The two story type ratings did not significantly differ,  $t(27.8) = -0.76$ ,  $p > .45$ .

## 2: Create the model in SPM.

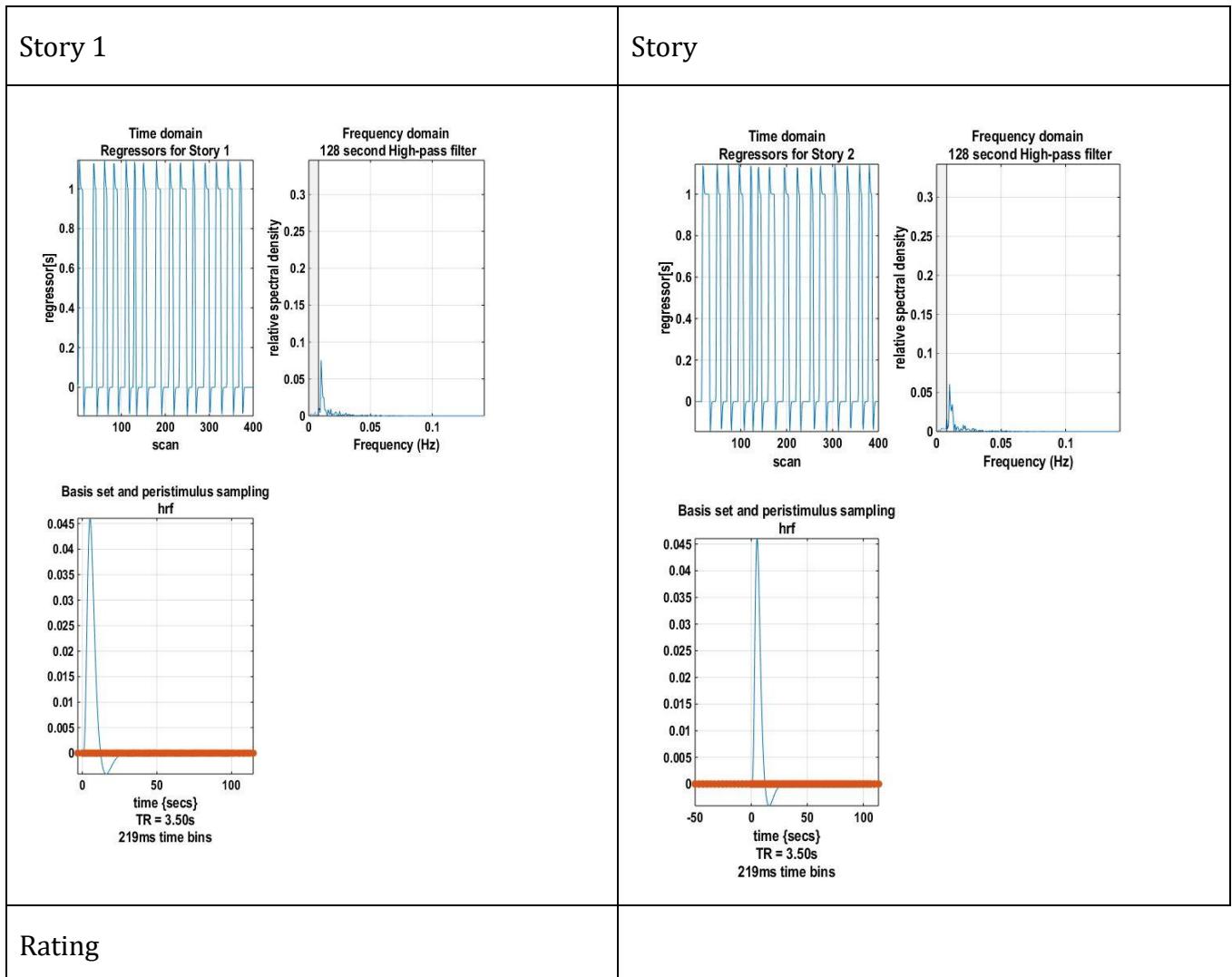
**2.a. Make a screenshot and report the design matrix figure generated by SPM. How many columns does it have? What do the different columns represent?**

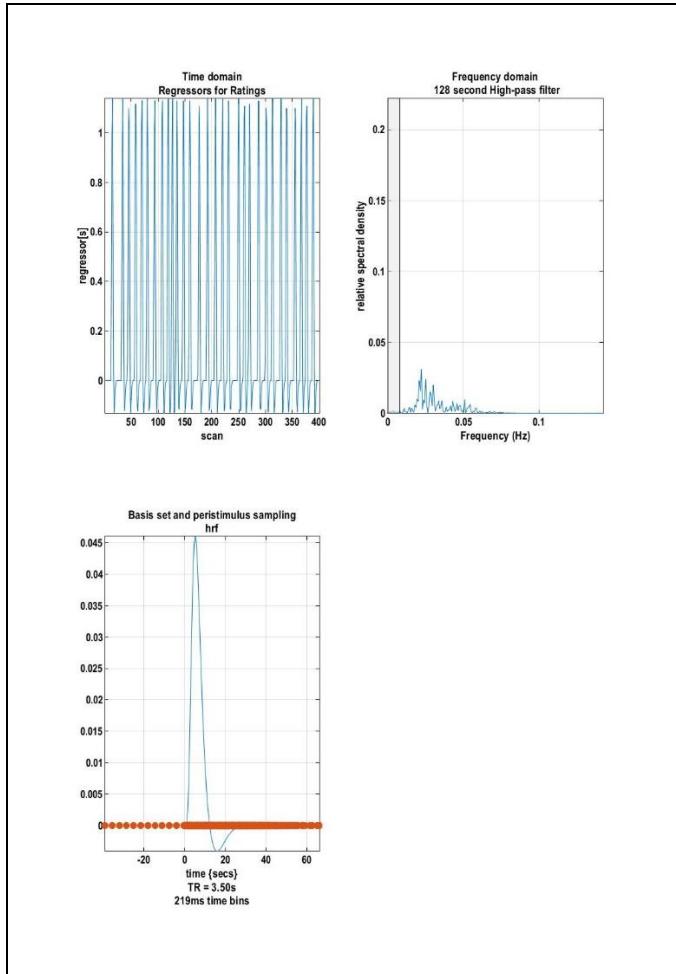


The design matrix contains 4 different columns. The first column represents story 1, the second column represents the story 2, the third column represents the rating and the last column is the intercept.

### 3: Checking the model

#### 3.a - Report periodogram plots of the Frequency domain for the three conditions.



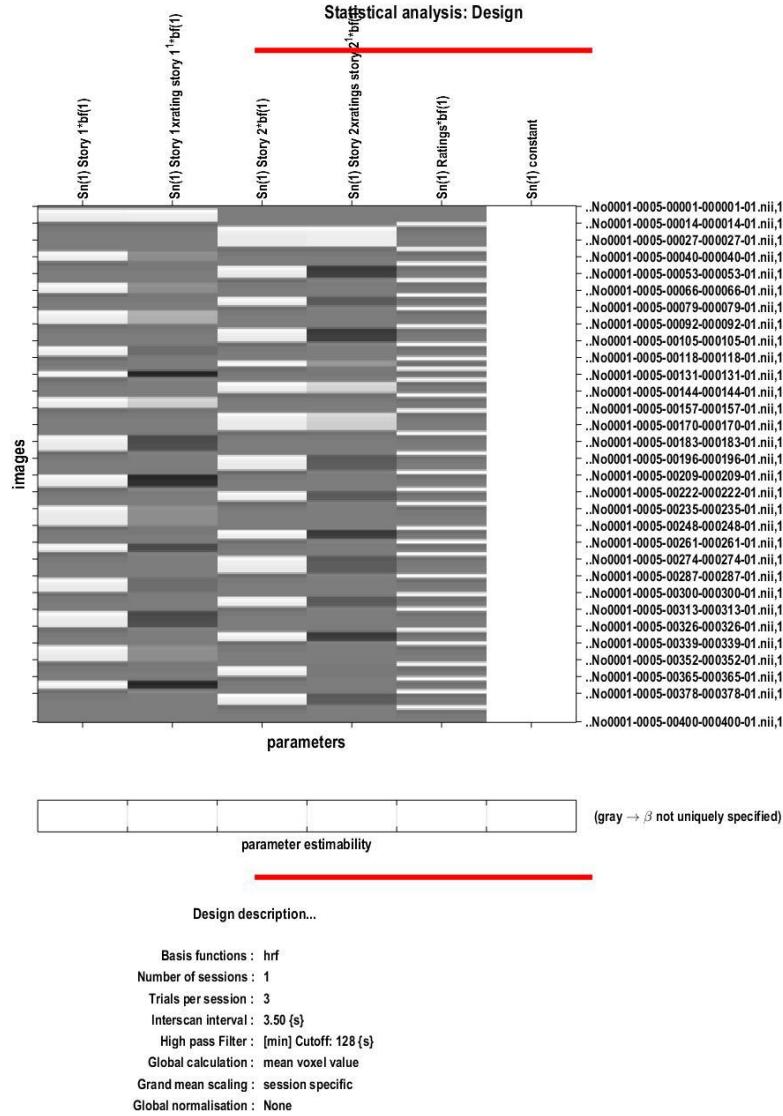


**3.b - Eye-balling task: What are the most predominant frequencies for the three condition, as seen from these plots?**

For story 1 the frequency peak is close to 0.005 and for story 2 it appears to be the same. For ratings the frequency peak appears to be close to 0.02.

## 4: Adding covariates

4.a - Make a screenshot and report the new design matrix figure. How many columns does it now have?

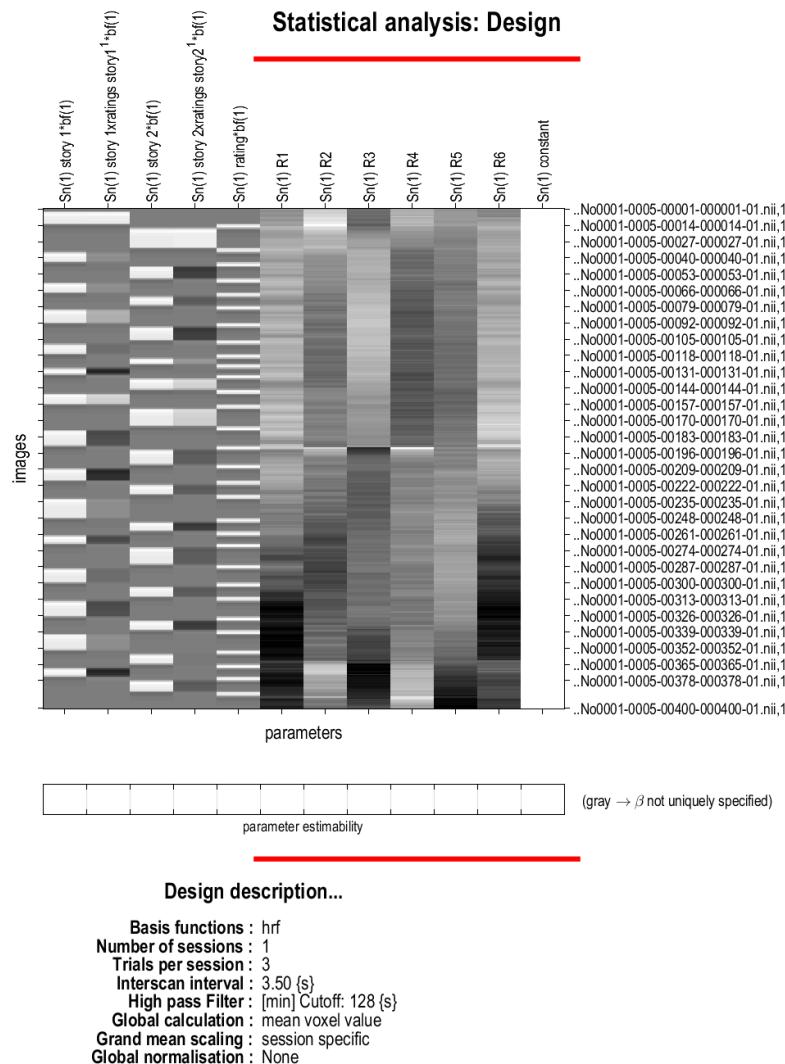


Which columns model the rating effects? There are 6 columns whereas the second and fourth column model the rating effects.

#### 4.b - Why is it important to subtract the mean?

By subtracting the mean from the data, the intercept can then be interpreted as the mean and therefore it becomes a meaningful value that can be compared to the other variables/predictors.

#### 4.c - Make a screenshot and report the new design matrix figure. How many columns does it now have? Which columns are modeling the motion?

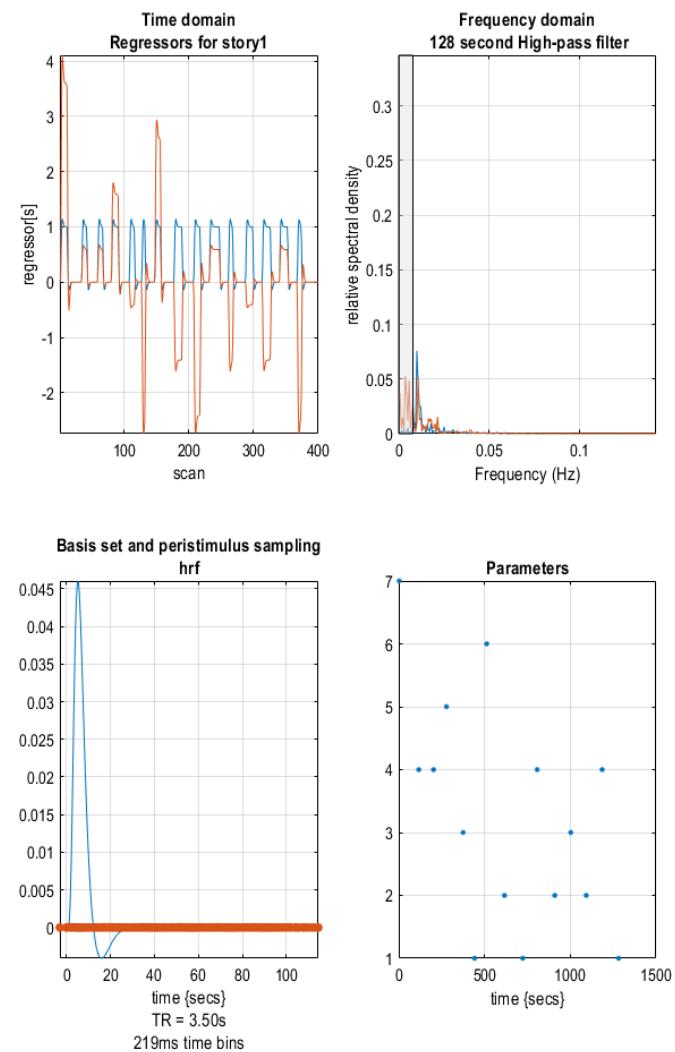


The design matrix now has 12 columns, where column 1-5 are indifferent to the ones in 4.a, column 6-11 are the motion parameters for the 6 different dimensions and column 12 is the intercept.

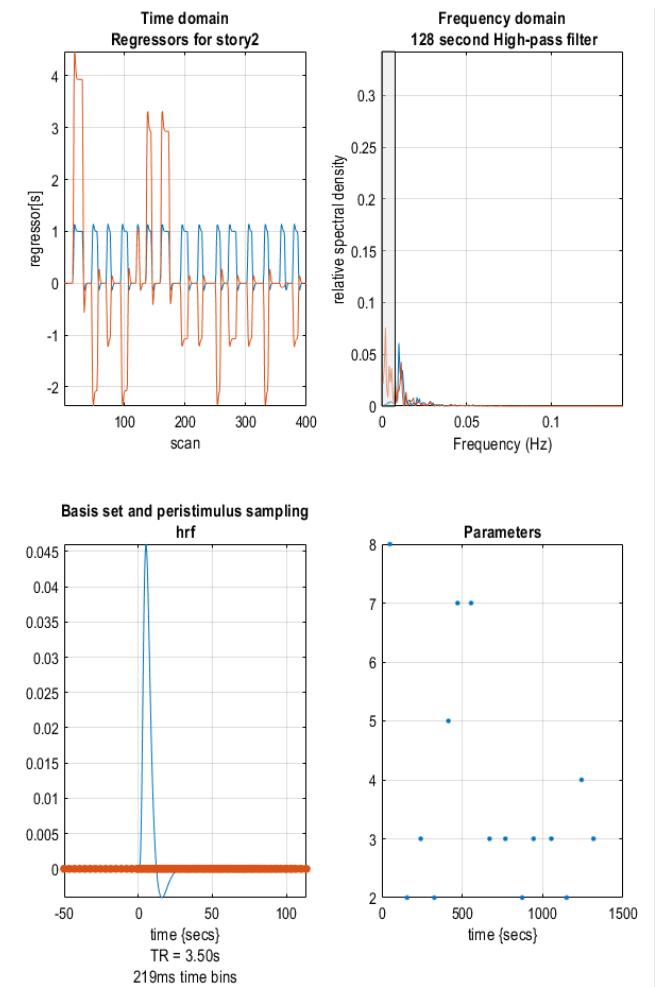
## 5: Checking the new model

### 5.a - Report plots of the Frequency domain for the three conditions.

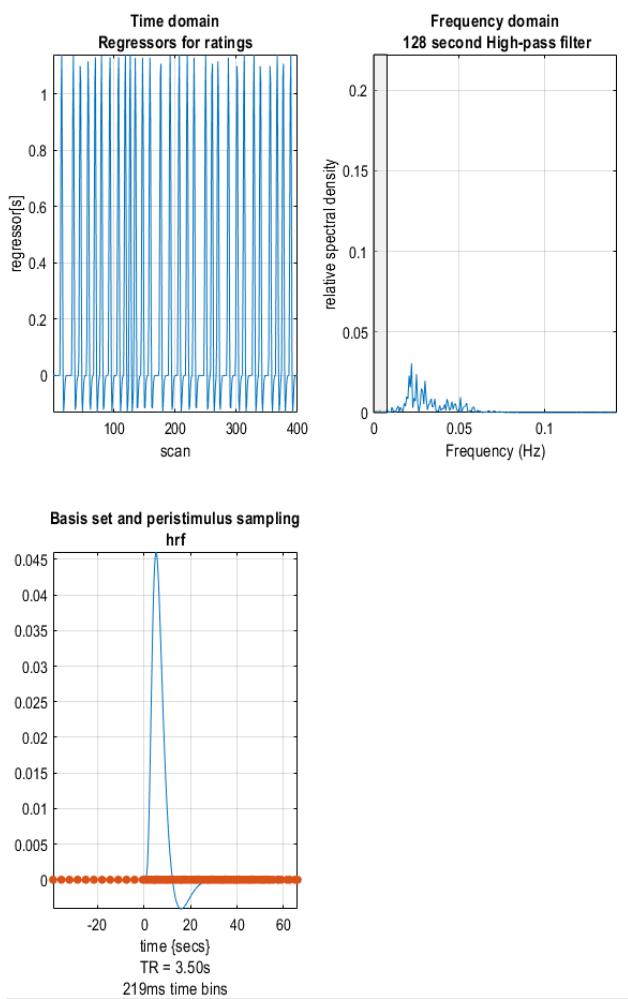
#### Story 1



## Story 2



## Ratings



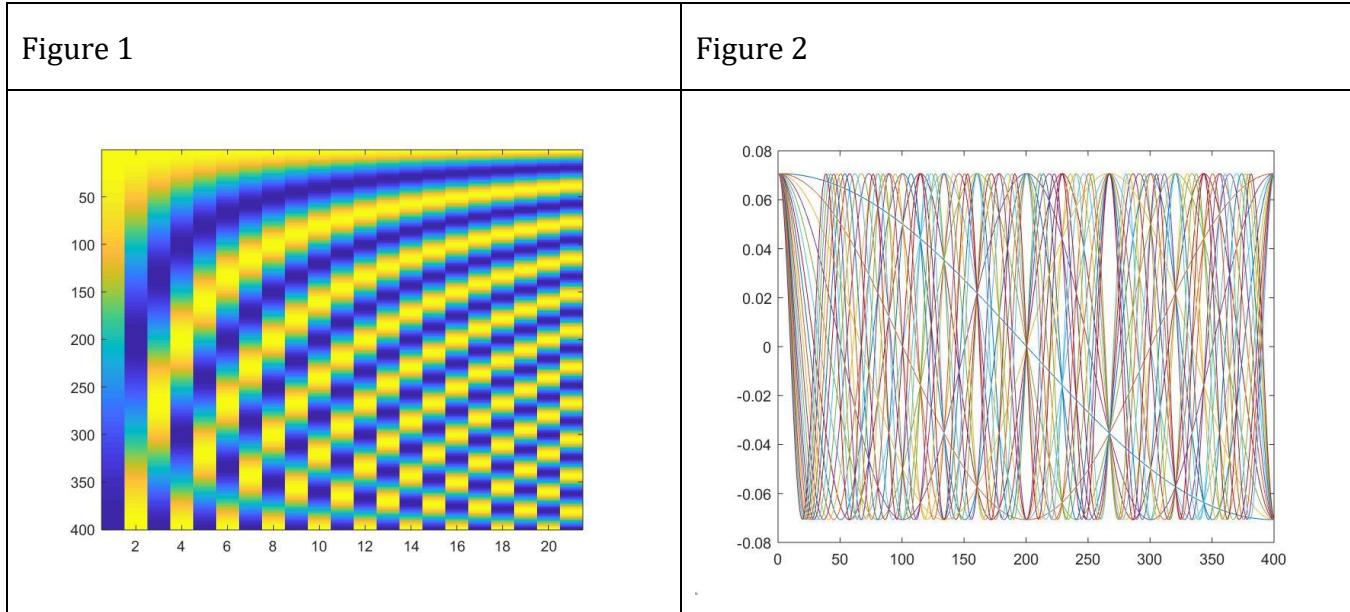
**5.b - What are the most predominant frequencies for the covariates, as seen from these plots?**

For both story 1 and story 2 the most predominant frequencies are closer to zero than the story 1 and story 2 conditions from the old model.

**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?**

Yes, it has a big effect on the covariates since most of the predominant frequencies lie below the high pass filter.

**5.c - The high-pass filter consists of low-frequency cosine-waves, which together can model any fluctuation below the specified frequency. Plot and report figures of the high-pass filter.**



**5.d - How many cosine waves are in this specific high-pass filter?**

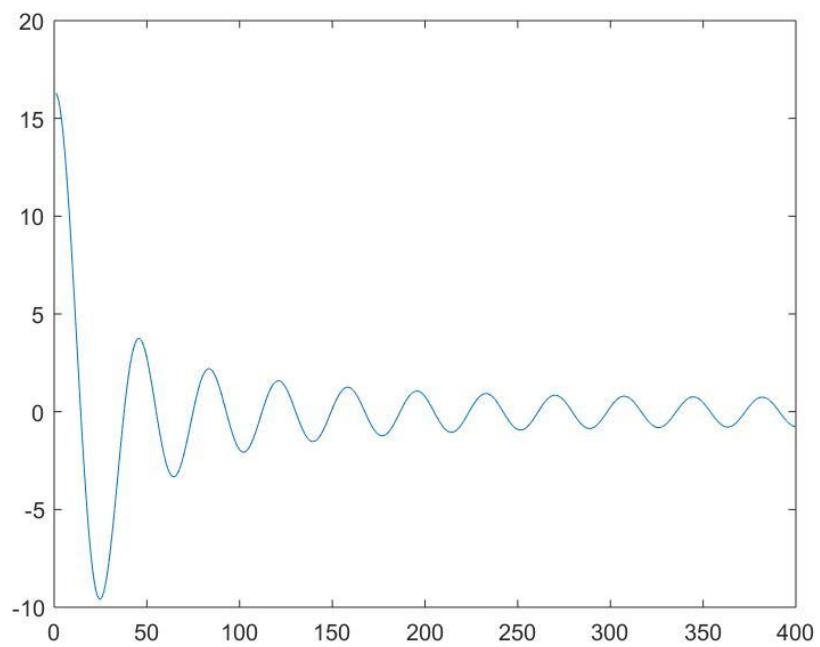
As can be counted on figure 1 the high-pass filter has 21 cosine waves.

**5.e - Make a hypothetical slow wave signal by creating a vector in Matlab with the same length as the number of waves as in the high-pass filter. Multiply the vector with the filter (using “\*”) and plot the result (figure, plot(my\_result\_vector)). Report the plot.**

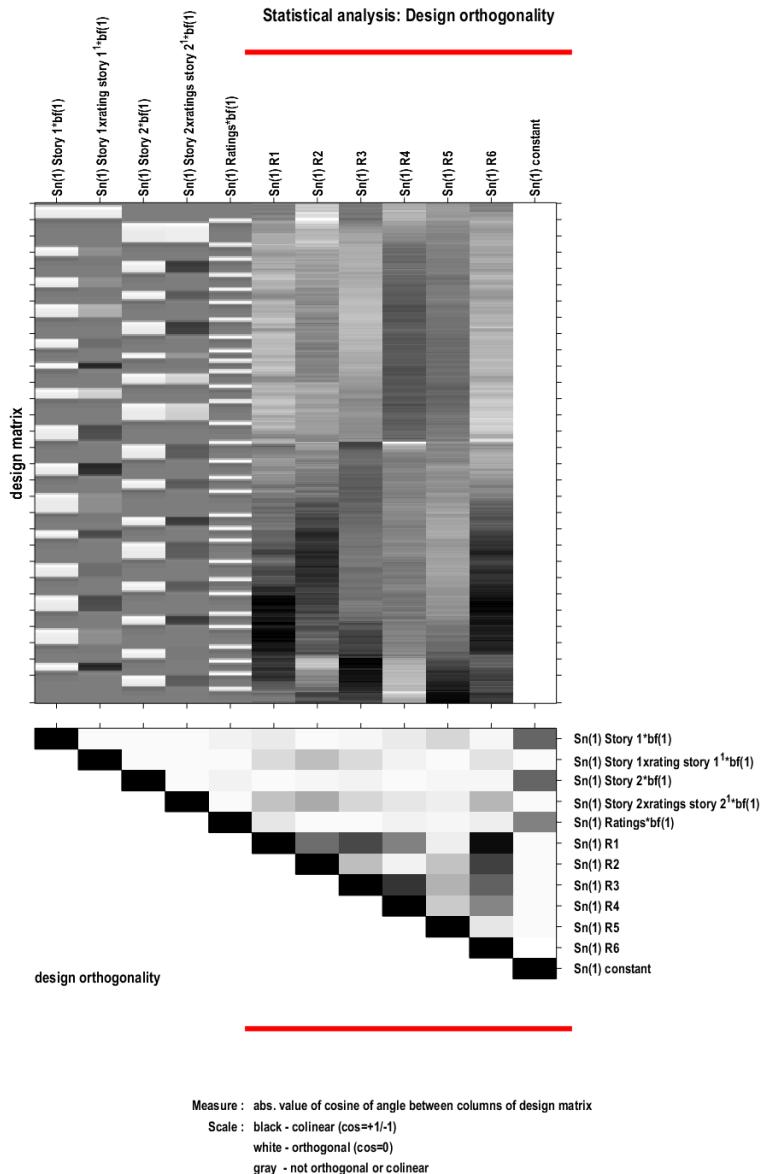
We used a vector with 21 entities from 1 to 21.

```
a <- matrix(1:21, nrow = 21, ncol = 1)
```

The high-pass filter was then multiplied with vector a creating a new plot:



**5.f. Eyeballing the bottomless pit of despair: Explore “design orthogonality” (in the “review” function). Dark colors in the design “orthonogality matrix” (include it in report) indicate that different covariates are correlated. Which covariates are most correlated in the current design?**

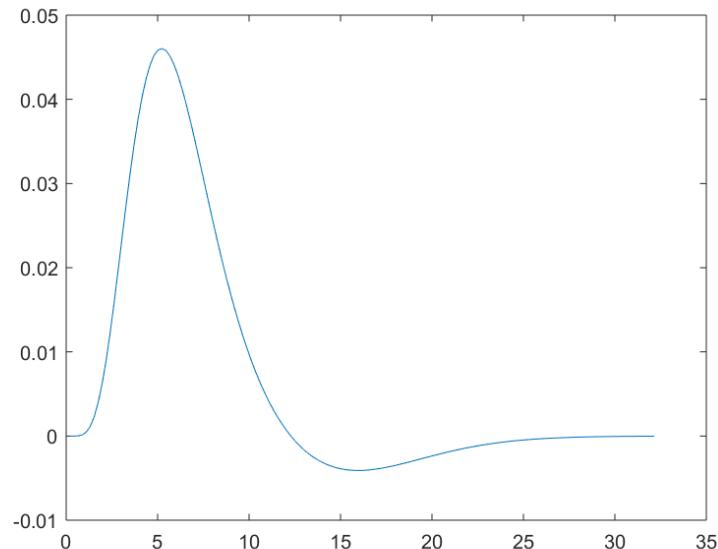


The rating covariate is slightly correlated with both story 1 and story 2, which makes sense since the rating depends on which story is told. The correlation is very small though ( $r = -0.03$  and  $r = -0.04$  respectively), and so we are not that deep into the bottomless pit of despair.

From the design orthogonality one can also observe that the movements in the different dimensions are highly correlated with each other.

It is also apparent that there is a small correlation between story 1 and rotation in direction R5 ( $r = 0.14$ ).

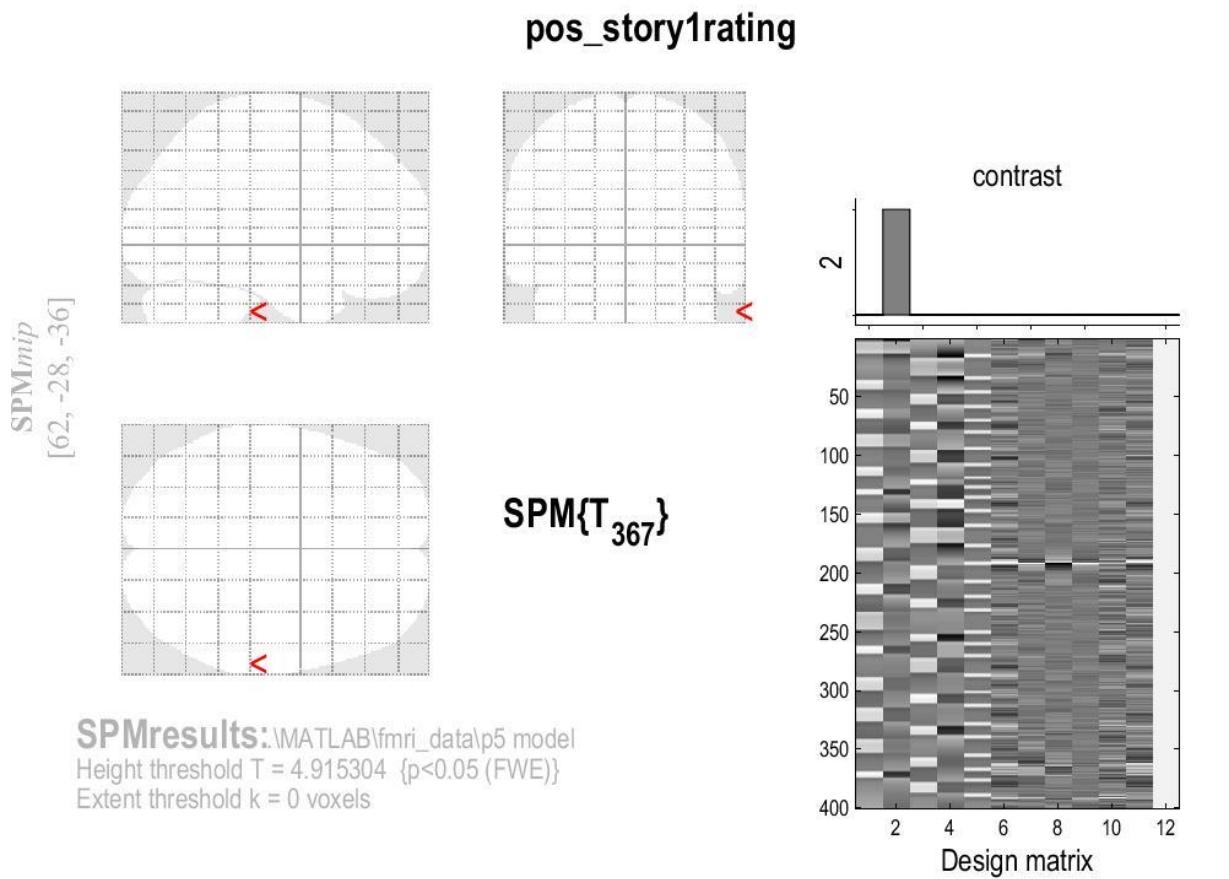
**5.g. Plot and report the hemodynamic response function (HRF) using this call in Matlab  
(you need to have loaded the SPM.mat file): figure,  
plot(SPM.xBF.dt:SPM.xBF.dt:SPM.xBF.length,SPM.xBF.bf)**



## **Portfolio 6: fMRI model estimation**

Group 3: Mie, Tobias, Riikka, Ruta and Jana

**Report the output coordinate table for each of the 14 contrasts, both significant and non-significant (e.g. using the “save figure” function in the “SPM Figure” menu).**



#### Statistics: p-values adjusted for search volume

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

*no suprathreshold clusters*

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,  $\langle c \rangle = 0.05$

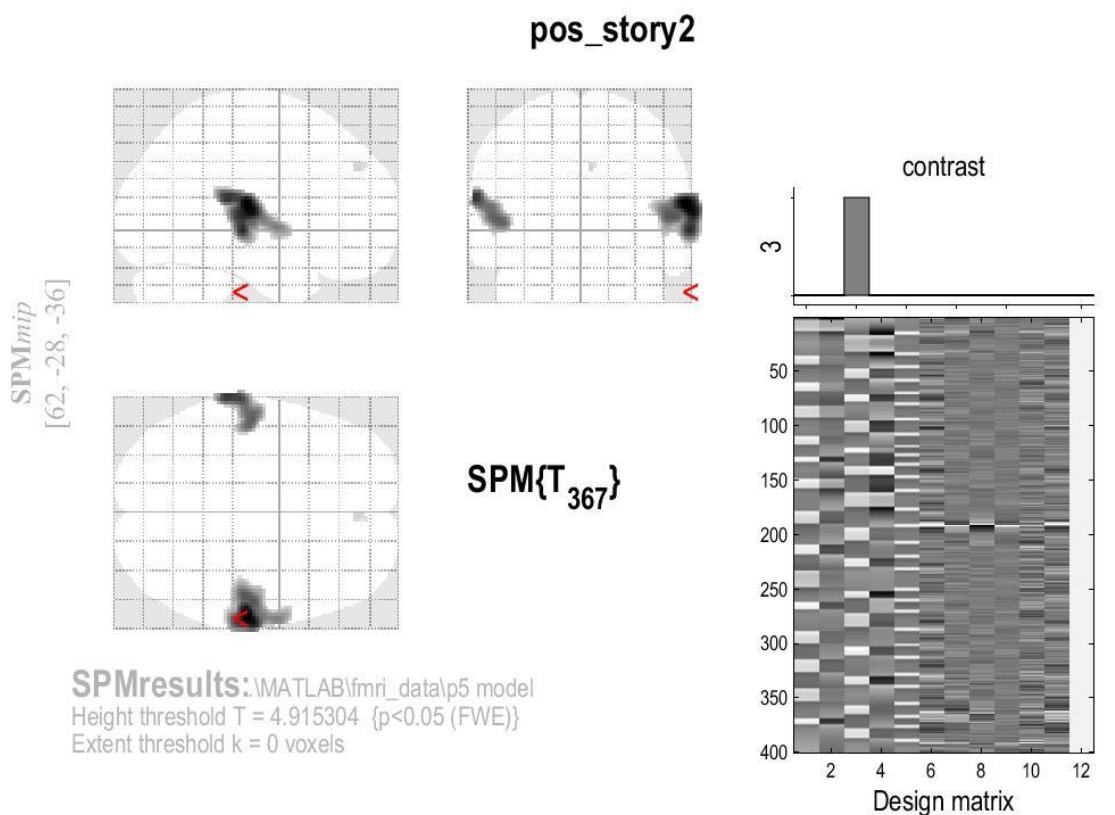
FWEp: 4.915, FDRp: Inf, FWEc: Inf, FDRc: Inf

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: 1846704 = 230838 voxels = 1982.5 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)



#### Statistics: *p*-values adjusted for search volume

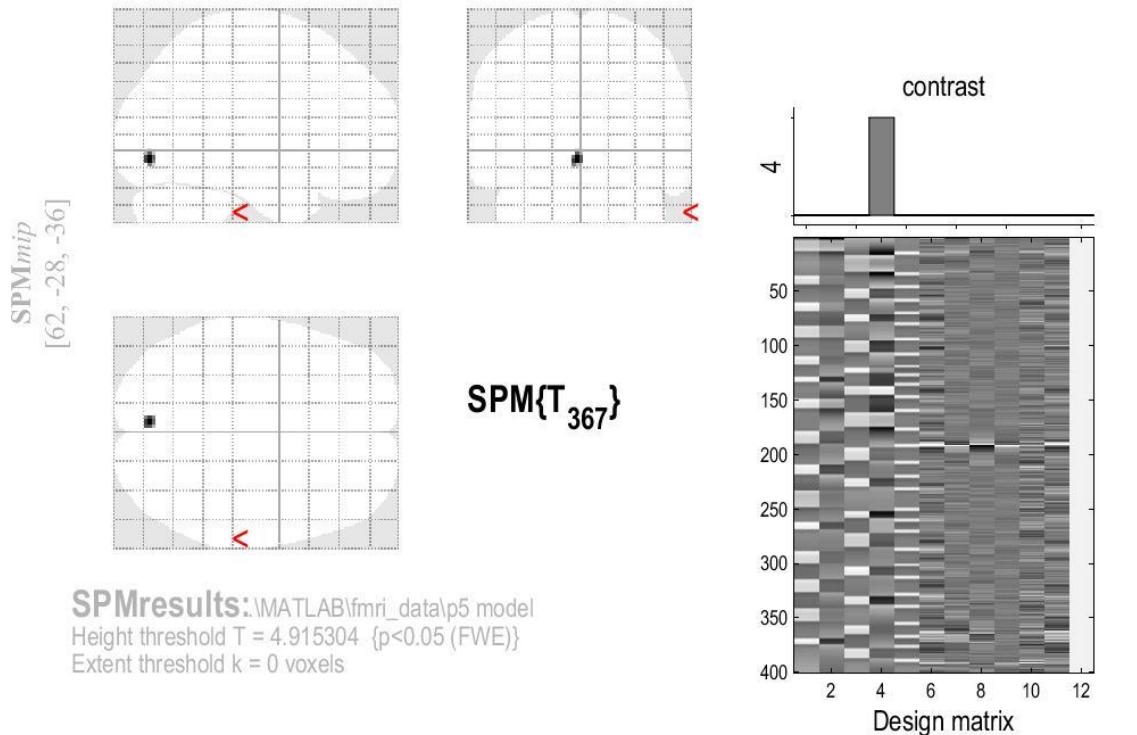
set-level	cluster-level				peak-level					mm mm mm		
	$p$	$c$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$T$	$(Z_{\equiv})$	$p_{\text{uncorr}}$	
0.000 3		0.000	0.000	1245	0.000		0.000	0.000	10.80	<b>Inf</b>	0.000	62 -22 12
							0.000	0.000	9.52	<b>Inf</b>	0.000	64 -24 -2
							0.000	0.000	8.45	<b>Inf</b>	0.000	54 -18 6
		0.000	0.000	489	0.000		<b>0.000</b>	<b>0.000</b>	<b>9.25</b>	<b>Inf</b>	<b>0.000</b>	<b>-66 -32 18</b>
							0.000	0.000	8.12	7.77	0.000	-52 -18 4
							0.000	0.000	7.88	7.57	0.000	-56 -22 10
		0.004	0.087	9	0.087		<b>0.013</b>	<b>0.263</b>	<b>5.22</b>	<b>5.13</b>	<b>0.000</b>	<b>4 46 34</b>

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: 7.338, FWEc: 9, FDRC: 489

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: 1846704 = 230838 voxels = 1982.5 resels  
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)

## pos\_story2rating



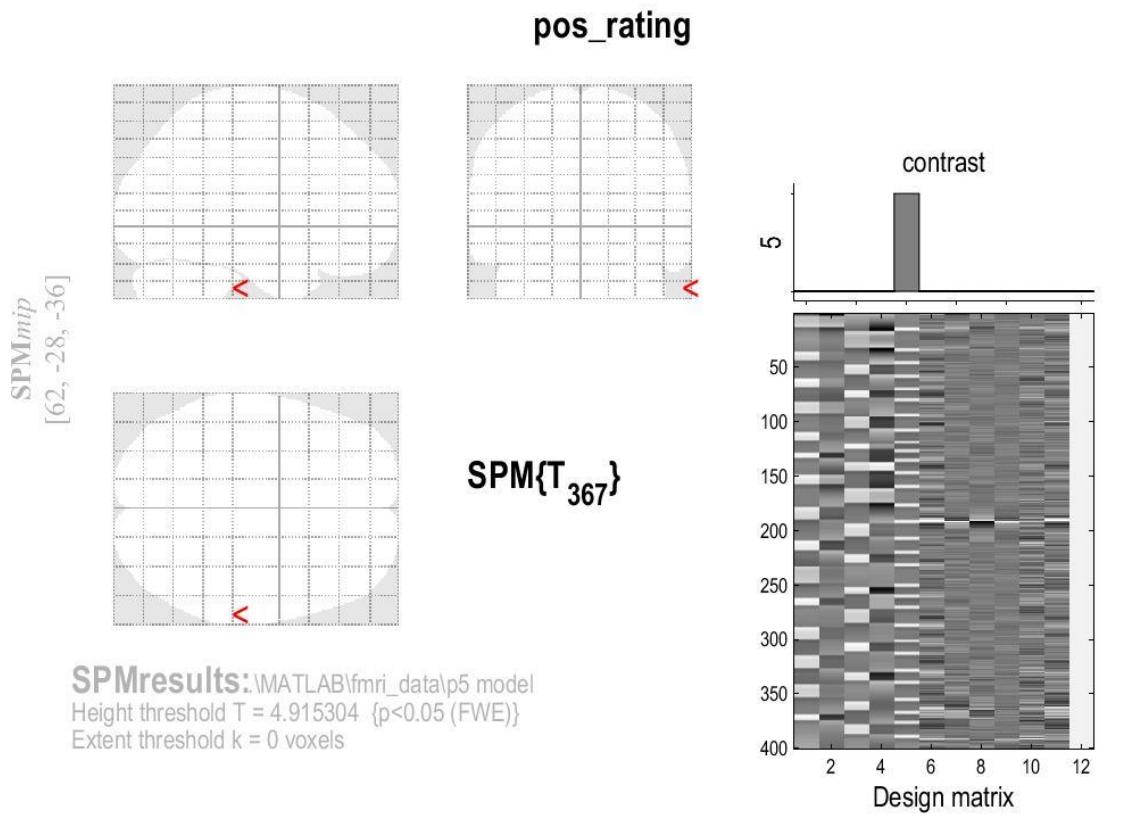
### Statistics: p-values adjusted for search volume

cluster-level				peak-level						mm mm mm		
p <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	k <sub>E</sub>	p <sub>uncorr</sub>	p <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	T	(Z <sub>≡</sub> )	p <sub>uncorr</sub>	-4	-80	-8	
0.000	0.007	26	0.007	0.002	0.043	5.61	5.49	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, <> = 0.05  
FWEp: 4.915, FDRp: 5.613, FWEc: 26, FDRc: 26

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: 1846704 = 230838 voxels = 1982.5 resels  
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)



#### Statistics: p-values adjusted for search volume

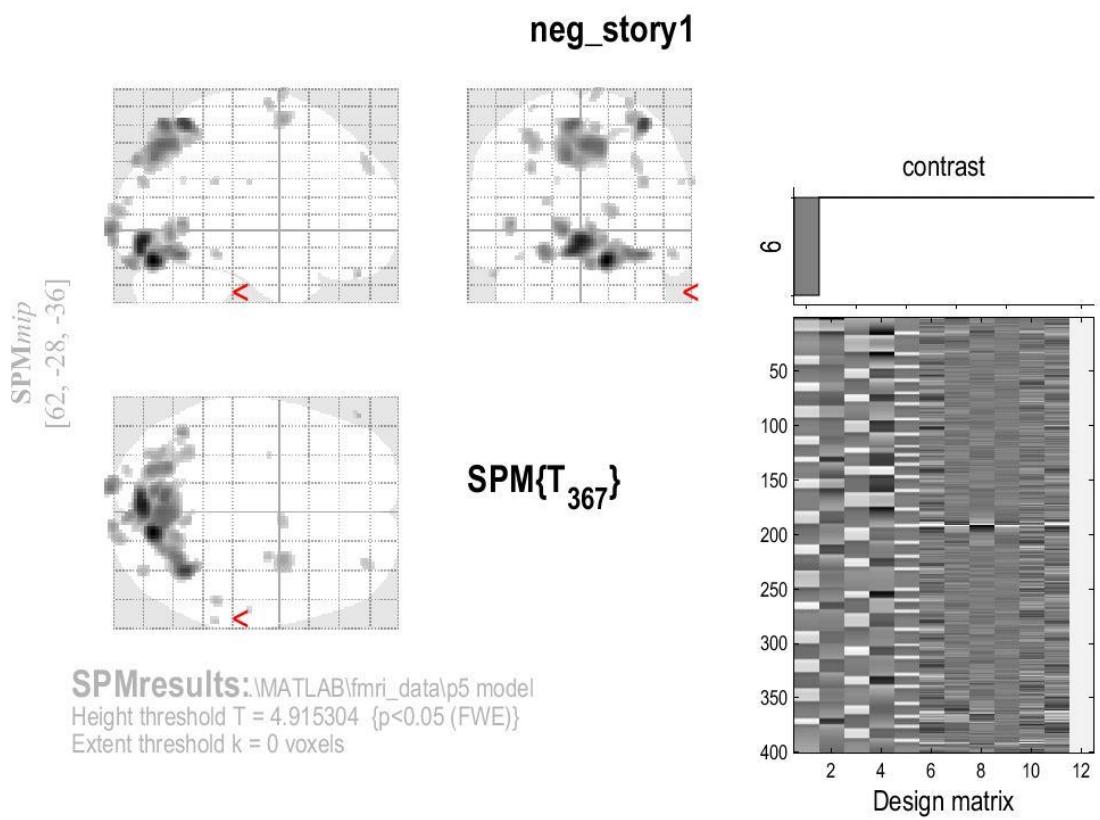
set-level		cluster-level				peak-level				mm	mm	mm	
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$			

*no suprathreshold clusters*

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: 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: 1846704 = 230838 voxels = 1982.5 resels  
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)



### Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	(Z <sub>≡</sub> )	$p_{\text{uncorr}}$	mm	mm	mm
0.000	23	0.000	0.000	1265	0.000	0.000	0.000	8.81	Inf	0.000	14	-78	-20
						0.000	0.000	8.20	7.84	0.000	-2	-84	-8
						0.000	0.000	6.91	6.70	0.000	34	-62	-16
		0.000	0.000	156	0.000	0.000	0.000	7.63	7.35	0.000	36	-60	60
		0.000	0.000	1226	0.000	0.000	0.000	7.04	6.81	0.000	-6	-72	60
						0.000	0.000	6.99	6.77	0.000	2	-72	48
						0.000	0.001	6.78	6.57	0.000	-10	-76	42
		0.000	0.000	96	0.000	0.000	0.001	6.70	6.50	0.000	-12	-104	-2
		0.000	0.002	45	0.001	0.000	0.008	6.15	6.00	0.000	-44	-60	2
		0.000	0.003	40	0.001	0.000	0.013	6.01	5.86	0.000	10	-102	-8
		0.000	0.000	109	0.000	0.000	0.013	6.00	5.85	0.000	30	0	62
						0.005	0.120	5.46	5.35	0.000	28	-8	72
		0.000	0.000	152	0.000	0.001	0.030	5.82	5.69	0.000	-30	-76	48
						0.001	0.046	5.71	5.58	0.000	-32	-56	50
						0.007	0.162	5.38	5.28	0.000	-28	-64	52
		0.000	0.002	49	0.001	0.001	0.037	5.77	5.65	0.000	18	-98	8
		0.001	0.022	21	0.014	0.001	0.045	5.72	5.60	0.000	-32	-80	34
		0.001	0.022	22	0.012	0.002	0.064	5.63	5.51	0.000	36	-84	26
		0.000	0.008	32	0.003	0.002	0.064	5.62	5.50	0.000	8	-60	58
		0.008	0.198	6	0.155	0.002	0.065	5.61	5.49	0.000	-54	46	-28
		0.002	0.058	14	0.038	0.003	0.070	5.59	5.47	0.000	54	-40	-16

table shows 3 local maxima more than 8.0mm apart

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

Degrees of freedom = [1.0, 367.0]

Extent threshold: k = 0 voxels

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

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

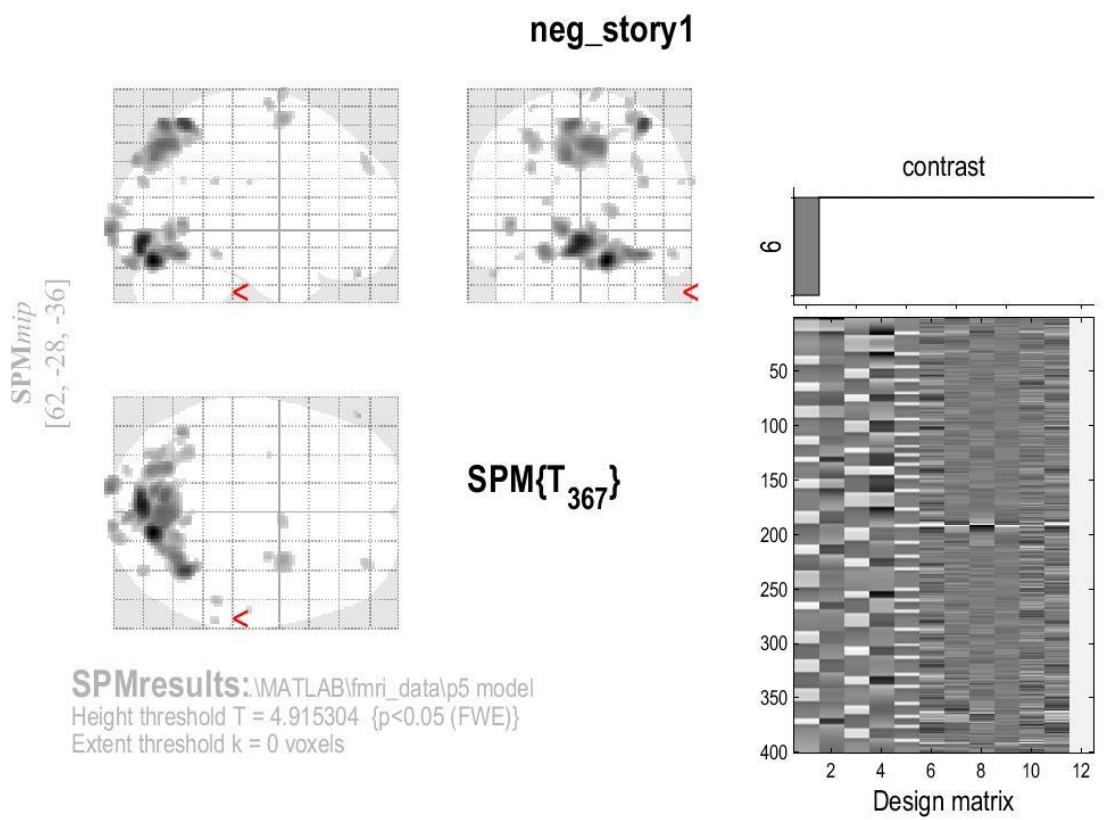
Volume: 1846704 = 230838 voxels = 1982.5 resels

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

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)

FWEp: 4.915, FDRp: 5.708, FWEc: 1, FDRc: 21

Page 1



#### Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$	mm	mm	mm
		0.001	0.022	21	0.014	0.004	0.096	5.51	5.40	0.000	8	0	80
		0.003	0.088	11	0.061	0.007	0.164	5.37	5.27	0.000	-22	-94	14
		0.001	0.022	21	0.014	0.008	0.176	5.35	5.24	0.000	32	50	38
		0.010	0.232	5	0.192	0.008	0.176	5.35	5.24	0.000	66	-40	26
		0.006	0.171	7	0.126	0.012	0.273	5.25	5.15	0.000	-40	-62	-18
		0.021	0.426	2	0.408	0.017	0.376	5.16	5.07	0.000	60	-20	26
		0.021	0.426	2	0.408	0.031	0.637	5.03	4.94	0.000	-30	-8	70
		0.021	0.426	2	0.408	0.046	0.939	4.94	4.85	0.000	-16	-60	28
		0.029	0.568	1	0.568	0.049	0.973	4.92	4.84	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)

Degrees of freedom = [1.0, 367.0]

Extent threshold: k = 0 voxels

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

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

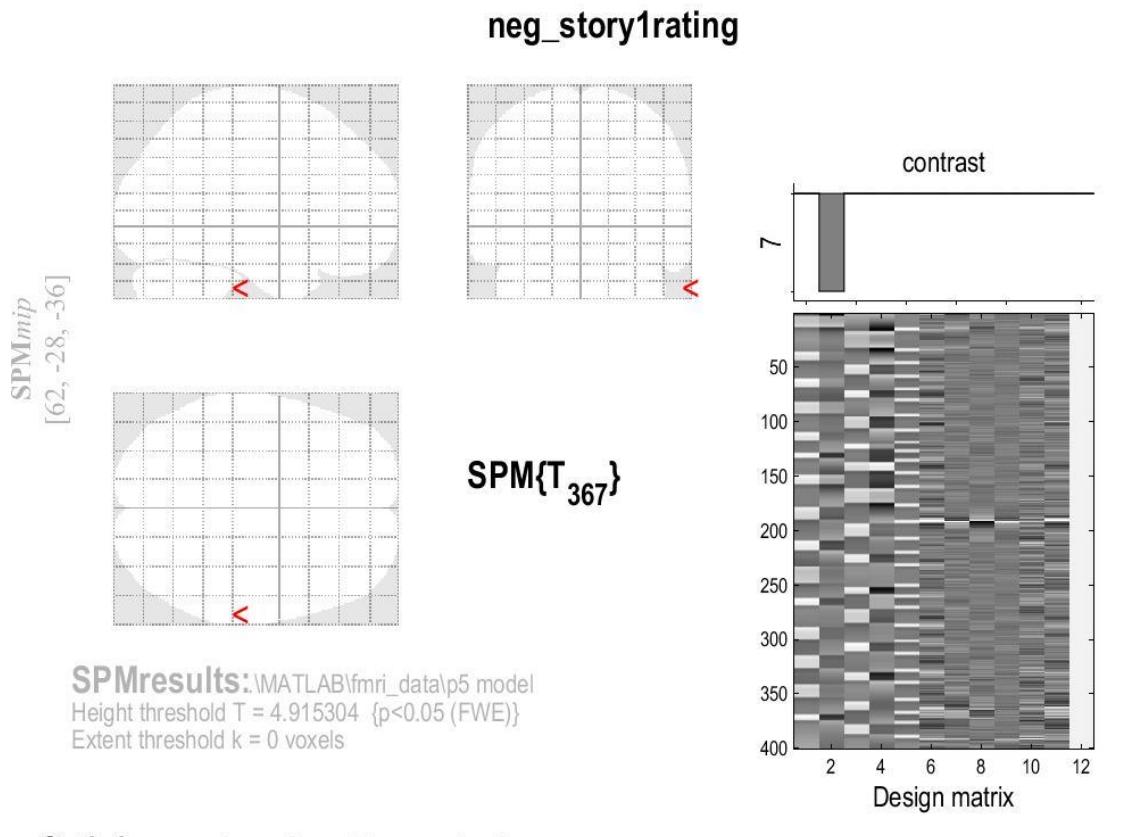
Volume: 1846704 = 230838 voxels = 1982.5 resels

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

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)

FWEp: 4.915, FDRp: 5.708, FWEc: 1, FDRC: 21

Page 2/2



**Statistics: p-values adjusted for search volume**

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

*no suprathreshold clusters*

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,  $\langle c \rangle = 0.05$

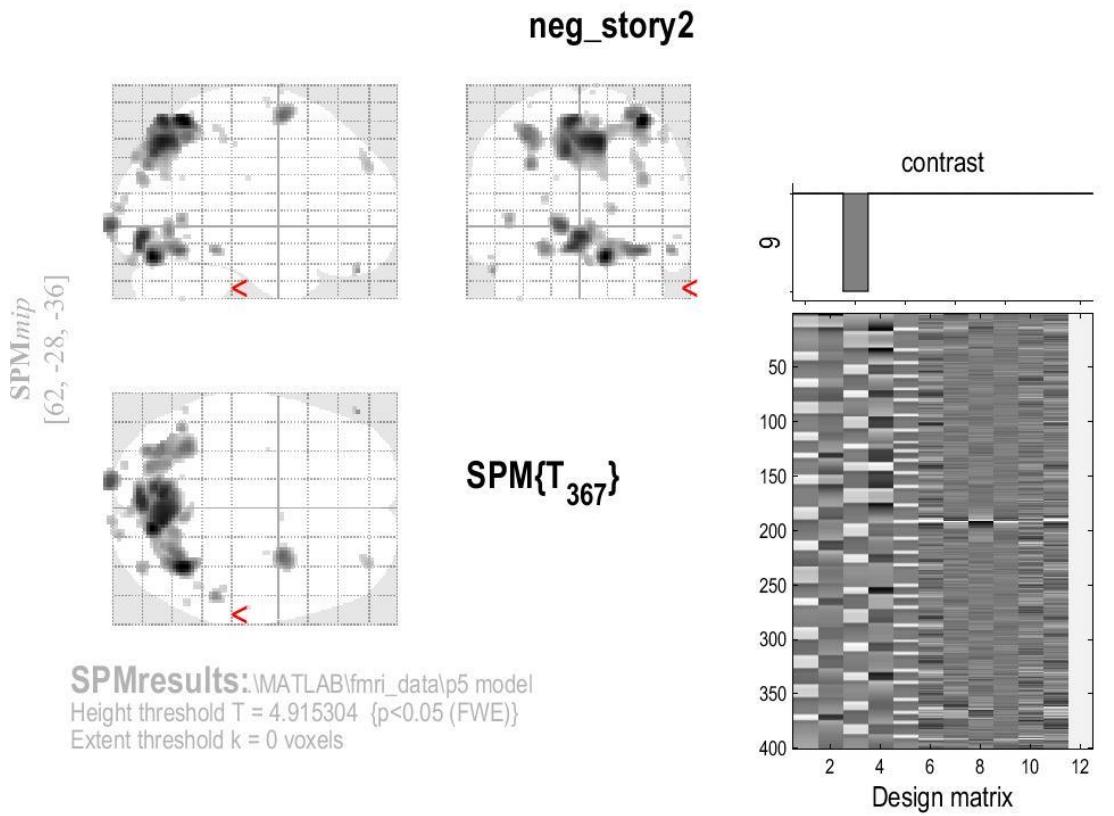
FWEp: 4.915, FDRp: Inf, FWEc: Inf, FDRc: Inf

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: 1846704 = 230838 voxels = 1982.5 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)



#### Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$	36	-60	60
0.000	27	0.000	0.000	1800	0.000	0.000	0.000	8.04	7.70	0.000	36	-60	60
						0.000	0.000	7.63	7.35	0.000	10	-72	48
						0.000	0.000	7.46	7.20	0.000	-6	-72	60
		0.000	0.000	802	0.000	0.000	0.000	7.99	7.66	0.000	14	-78	-20
						0.000	0.000	7.38	7.12	0.000	-2	-84	-8
						0.000	0.000	6.76	6.56	0.000	34	-62	-16
		0.000	0.000	101	0.000	0.000	0.000	6.86	6.65	0.000	-14	-102	-2
		0.000	0.000	156	0.000	0.000	0.001	6.58	6.40	0.000	30	0	62
		0.000	0.000	423	0.000	0.000	0.002	6.47	6.29	0.000	-34	-56	50
						0.000	0.003	6.37	6.20	0.000	-32	-72	54
						0.000	0.005	6.25	6.09	0.000	-32	-80	34
		0.000	0.000	106	0.000	0.000	0.010	6.12	5.97	0.000	-22	-80	-16
						0.000	0.016	6.00	5.86	0.000	-24	-70	-14
		0.000	0.020	25	0.008	0.000	0.015	6.03	5.88	0.000	54	-40	-16
		0.006	0.201	7	0.126	0.000	0.021	5.92	5.78	0.000	-54	46	-28
		0.000	0.001	56	0.000	0.001	0.024	5.88	5.75	0.000	8	-60	58
						0.014	0.368	5.22	5.12	0.000	2	-54	62
		0.000	0.017	27	0.006	0.001	0.025	5.86	5.73	0.000	-44	-60	2
		0.000	0.008	35	0.002	0.001	0.037	5.77	5.65	0.000	36	-84	26
		0.001	0.037	19	0.018	0.002	0.075	5.60	5.48	0.000	18	-98	8
		0.001	0.035	20	0.016	0.004	0.119	5.49	5.38	0.000	10	-102	-8

table shows 3 local maxima more than 8.0mm apart

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

Degrees of freedom = [1.0, 367.0]

Extent threshold: k = 0 voxels

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

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

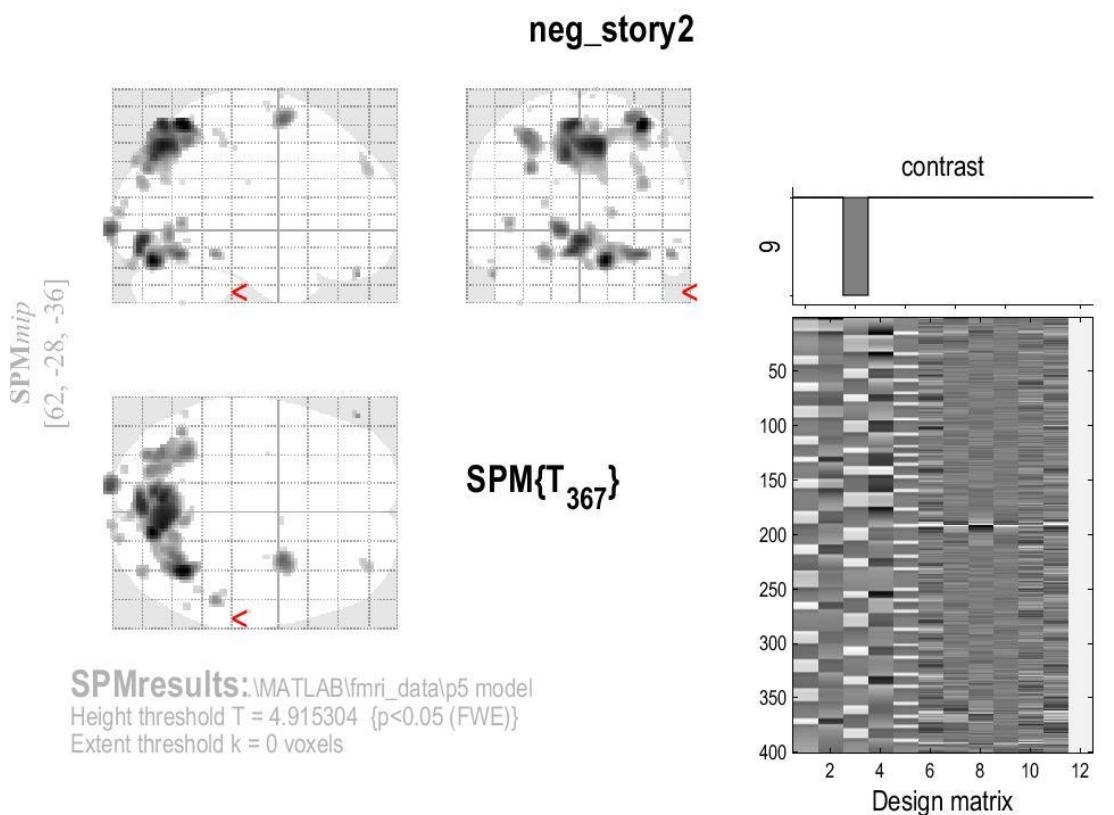
Volume: 1846704 = 230838 voxels = 1982.5 resels

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

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)

FWEp: 4.915, FDRp: 5.716, FWEc: 1, FDRc: 19

Page 1



#### Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level				mm mm mm			
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$	mm	mm	mm
		0.000	0.017	27	0.006	0.006	0.171	5.40	5.30	0.000	34	52	34
		0.005	0.176	8	0.104	0.013	0.363	5.23	5.13	0.000	-24	-94	14
		0.002	0.062	15	0.032	0.017	0.447	5.17	5.07	0.000	50	-40	56
						0.018	0.448	5.15	5.06	0.000	48	-34	48
		0.016	0.439	3	0.309	0.018	0.448	5.16	5.06	0.000	-56	-58	-20
		0.005	0.176	8	0.104	0.020	0.494	5.13	5.03	0.000	48	-50	52
		0.012	0.361	4	0.241	0.023	0.535	5.10	5.01	0.000	58	-56	-8
		0.029	0.568	1	0.568	0.031	0.723	5.03	4.94	0.000	64	-28	46
		0.029	0.568	1	0.568	0.036	0.818	4.99	4.91	0.000	8	0	80
		0.029	0.568	1	0.568	0.038	0.830	4.98	4.90	0.000	6	0	76
		0.021	0.550	2	0.408	0.041	0.889	4.96	4.88	0.000	28	-10	72
		0.029	0.568	1	0.568	0.045	0.952	4.94	4.86	0.000	-32	-8	68
		0.029	0.568	1	0.568	0.046	0.952	4.94	4.85	0.000	60	-20	26
		0.029	0.568	1	0.568	0.048	0.980	4.92	4.84	0.000	-16	-60	26
		0.029	0.568	1	0.568	0.050	0.994	4.92	4.84	0.000	-38	-60	-44

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: 5.716, FWEc: 1, FDRc: 19

Degrees of freedom = [1.0, 367.0]

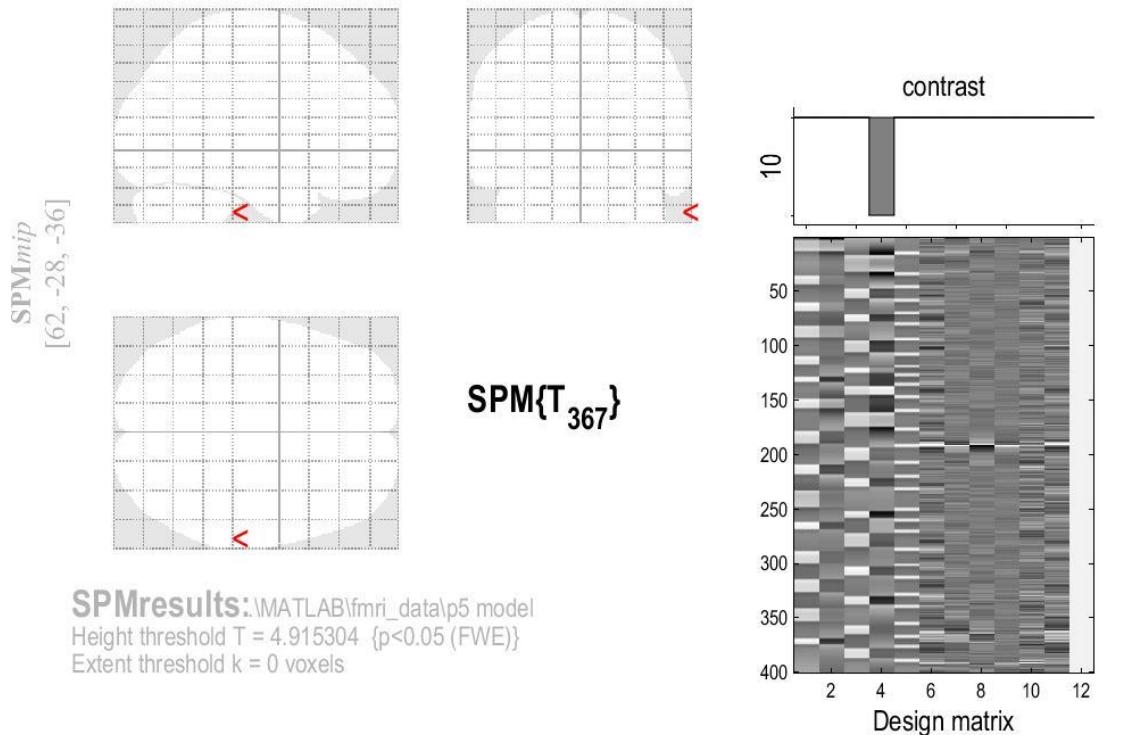
FWHM = 9.6 9.6 9.4 mm mm mm; 4.8 4.8 4.7 {voxels}

Volume: 1846704 = 230838 voxels = 1982.5 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)

Page 2/2

## neg\_story2rating



### Statistics: p-values adjusted for search volume

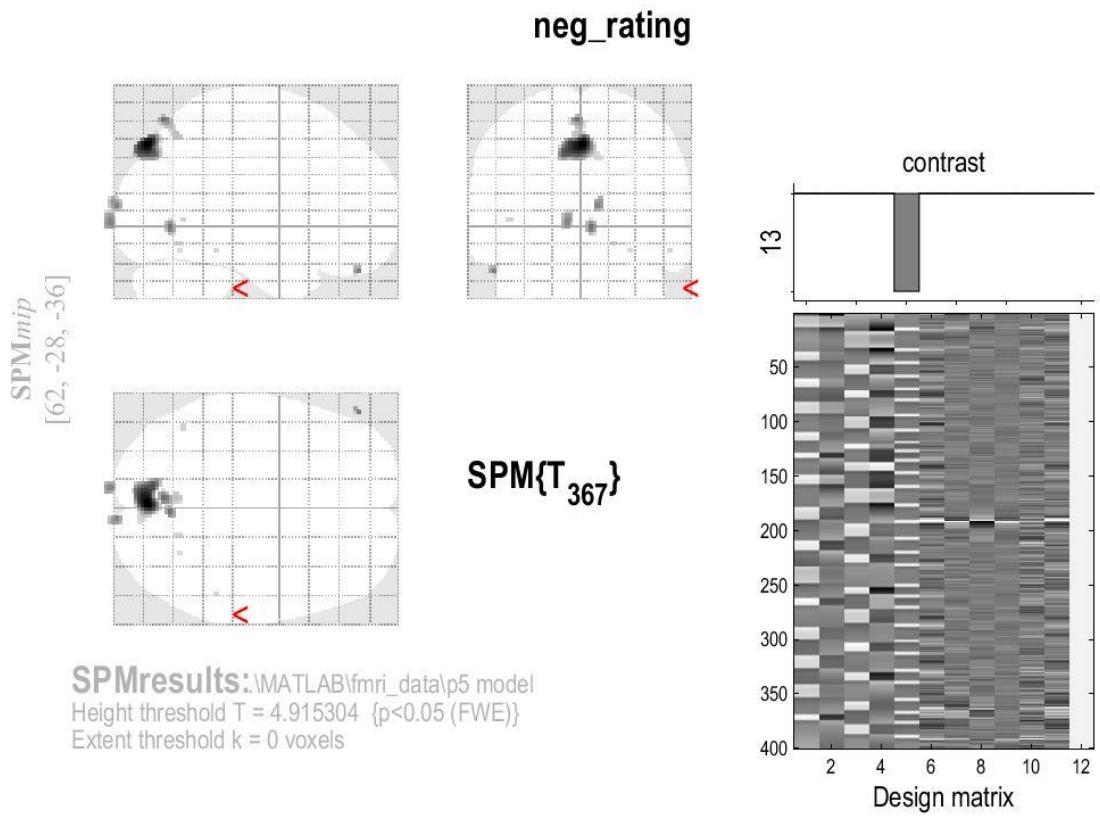
set-level		cluster-level				peak-level				mm mm mm		
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$		

*no suprathreshold clusters*

*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: 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: 1846704 = 230838 voxels = 1982.5 resels  
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)



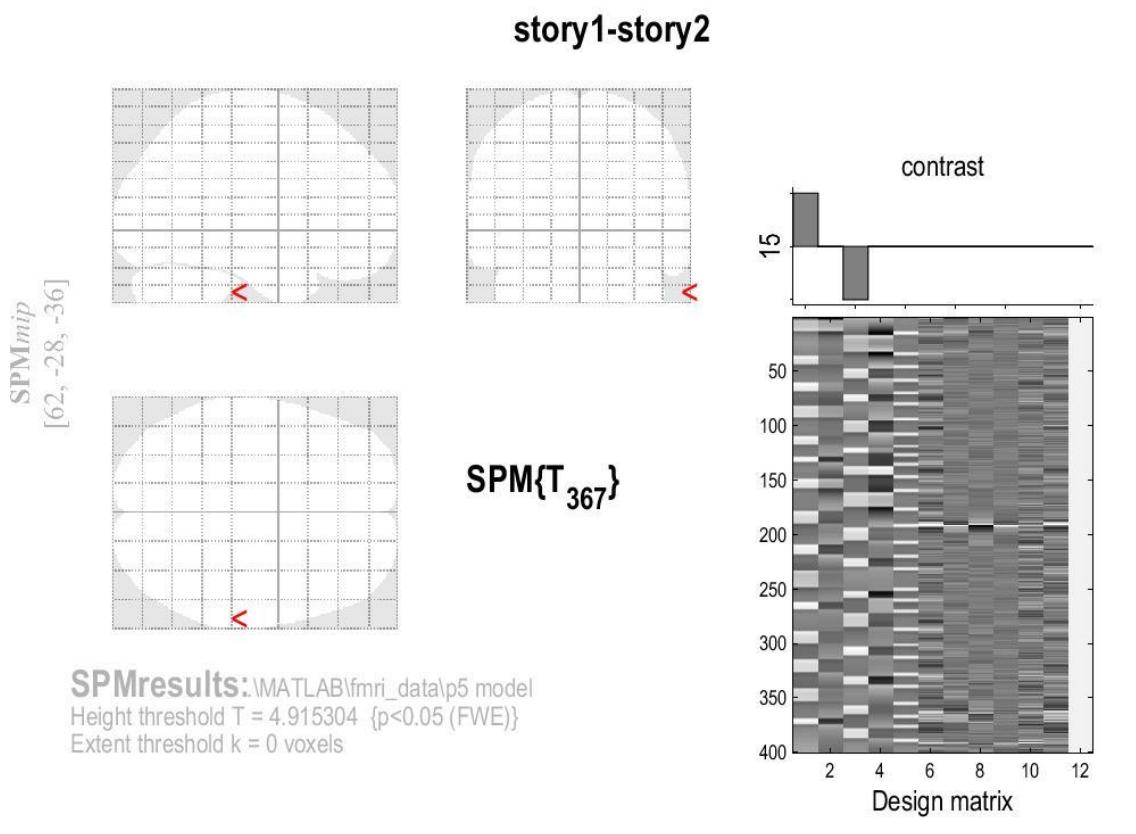
#### Statistics: p-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$	-2	-82	46
0.000	10	0.000	0.000	276	0.000	0.000	0.009	6.41	6.24	0.000	-2	-82	46
						0.036	0.839	4.99	4.91	0.000	8	-86	38
		0.000	0.016	26	0.007	0.002	0.158	5.64	5.52	0.000	4	-68	-2
		0.000	0.004	45	0.001	0.002	0.158	5.63	5.51	0.000	-4	-72	60
						0.020	0.652	5.13	5.04	0.000	-4	-64	50
		0.008	0.258	6	0.155	0.003	0.158	5.57	5.45	0.000	-54	46	-28
		0.000	0.016	25	0.008	0.004	0.158	5.51	5.39	0.000	8	-100	10
		0.000	0.013	31	0.004	0.004	0.158	5.50	5.39	0.000	-10	-104	0
		0.021	0.510	2	0.408	0.032	0.839	5.02	4.94	0.000	-44	-60	2
		0.021	0.510	2	0.408	0.035	0.839	5.00	4.91	0.000	18	-62	-12
		0.029	0.568	1	0.568	0.039	0.839	4.98	4.89	0.000	52	-40	-16
		0.029	0.568	1	0.568	0.042	0.845	4.96	4.87	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,  $\langle k \rangle = 3.128$   
Expected number of clusters,  $\langle c \rangle = 0.05$   
FWEp: 4.915, FDRp: 6.410, 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: 1846704 = 230838 voxels = 1982.5 resels  
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)



**Statistics: p-values adjusted for search volume**

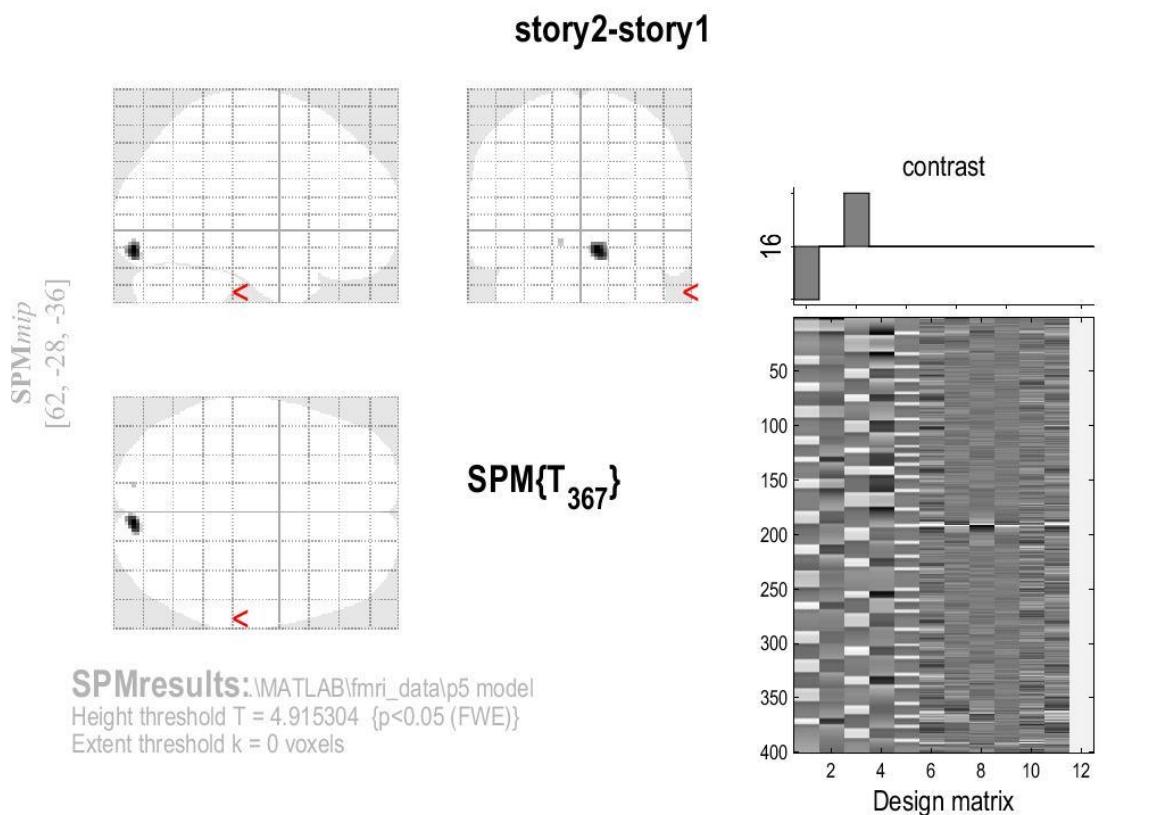
set-level		cluster-level				peak-level				mm mm mm		
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$		

*no suprathreshold clusters*

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: 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: 1846704 = 230838 voxels = 1982.5 resels  
Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)



**SPMresults:** \MATLAB\fmri\_data\p5 model  
 Height threshold T = 4.915304 {p<0.05 (FWE)}  
 Extent threshold k = 0 voxels

#### Statistics: p-values adjusted for search volume

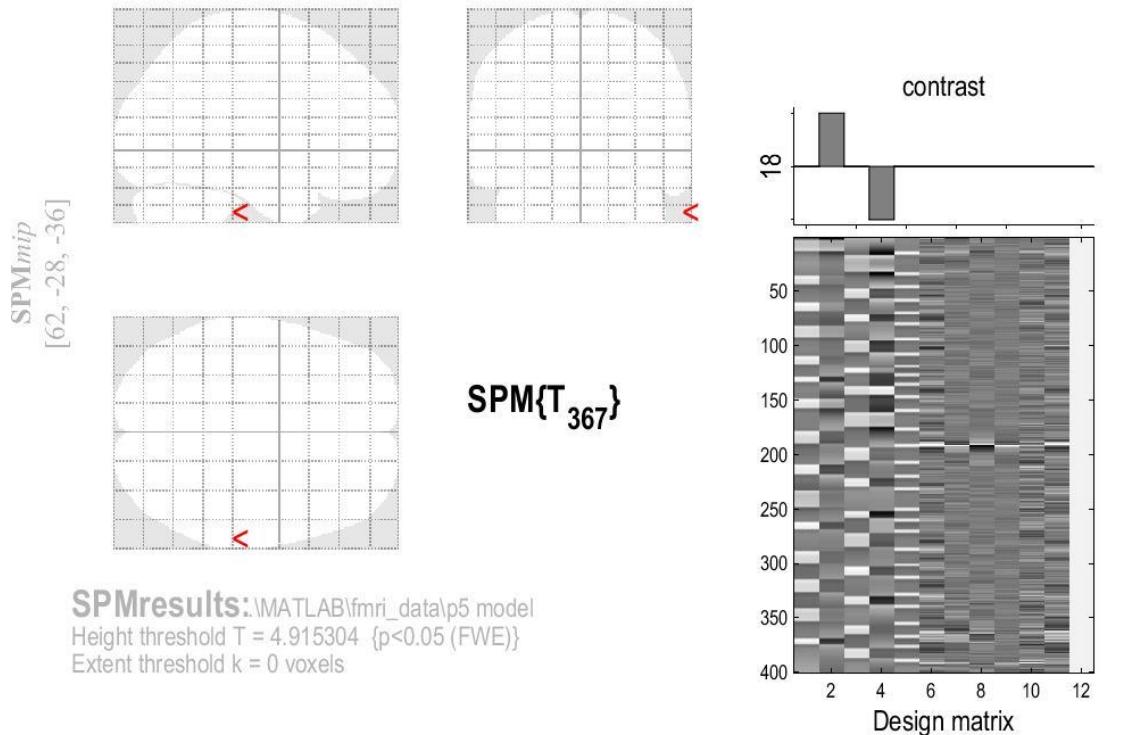
set-level		cluster-level				peak-level					mm mm mm		
p	c	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$	8	-90	-14
0.001	2	0.000	0.000	64	0.000	0.000	0.001	6.58	6.39	0.000	8	-90	-14
		0.021	0.408	2	0.408	0.018	0.350	5.16	5.07	0.000	-14	-90	-8

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: 6.581, FWEc: 2, FDRC: 64

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: 1846704 = 230838 voxels = 1982.5 resels  
 Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)

## rating\_story1-story2



### Statistics: p-values adjusted for search volume

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

*no suprathreshold clusters*

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: Inf, FWEc: Inf, FDRc: Inf

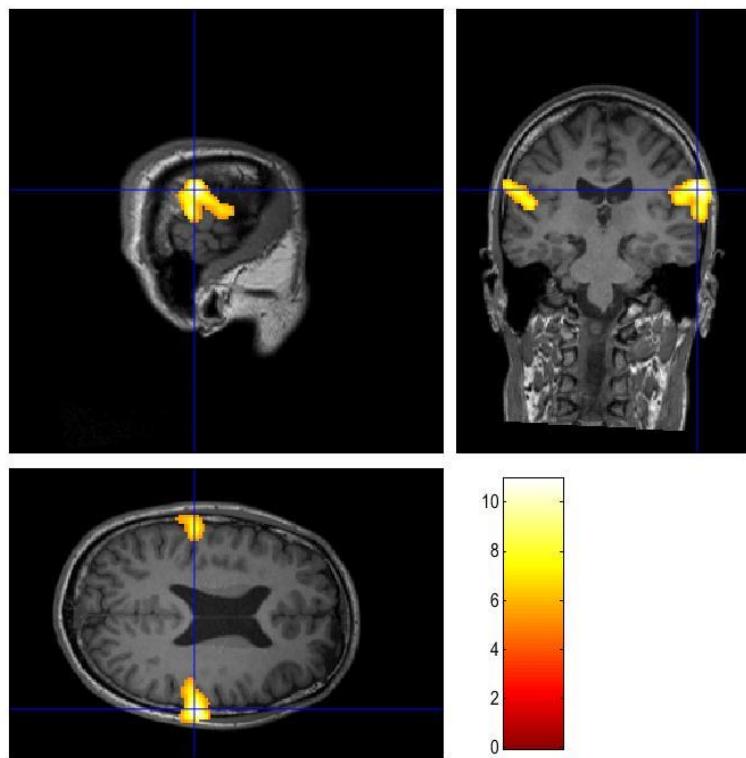
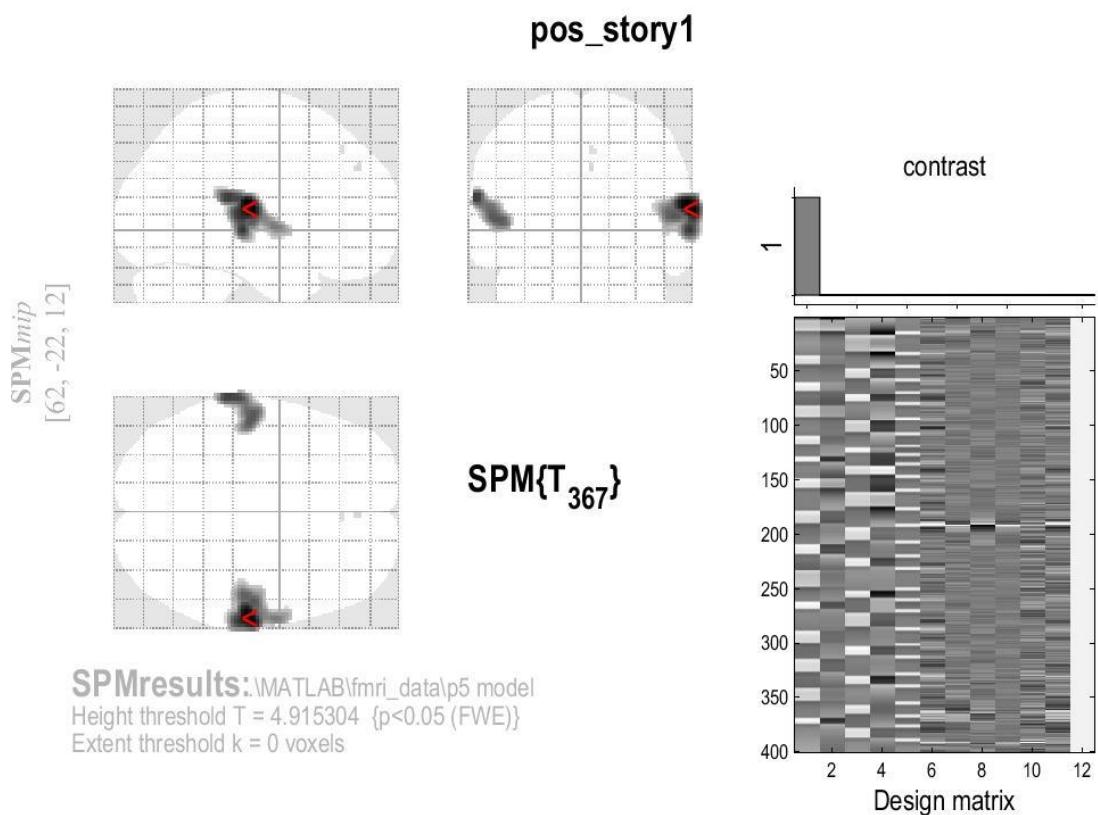
Degrees of freedom = [1.0, 367.0]

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

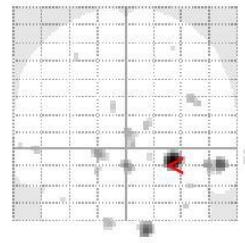
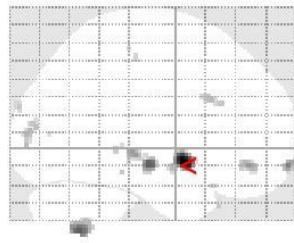
Volume: 1846704 = 230838 voxels = 1982.5 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 108.16 voxels)

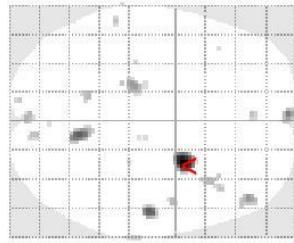
**For significant contrast, include a nice overlayed image, displaying the most significant effect.**



**SPM<sub>mip</sub>**  
[26, 4, -10]

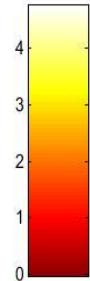
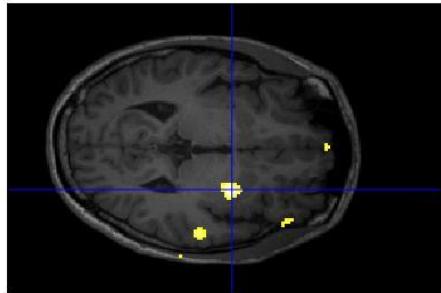
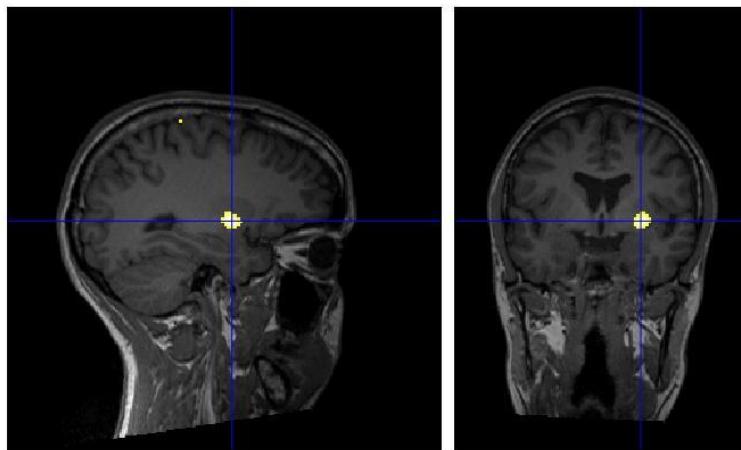
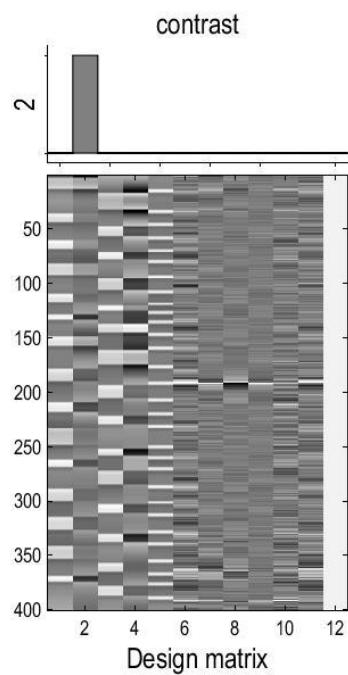


**SPM{T<sub>367</sub>}**

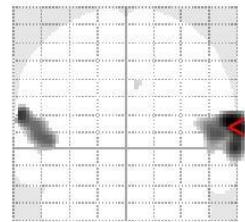
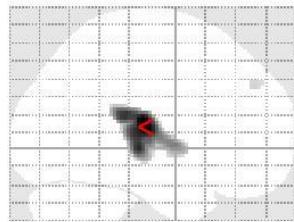


**SPM results:** \MATLAB\fmri\_data\p5 model  
Height threshold T = 3.112586 {p<0.001 (unc.)}  
Extent threshold k = 0 voxels

**pos\_story1rating**

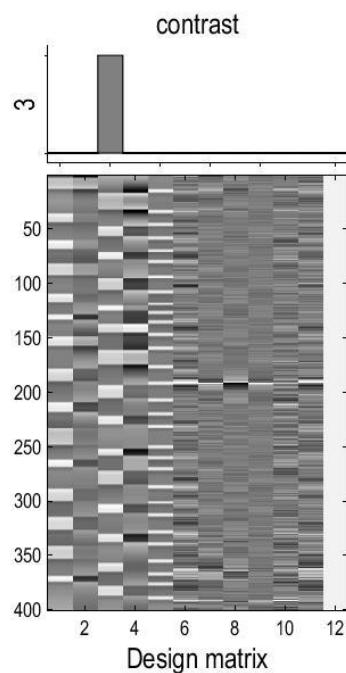
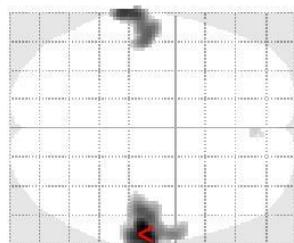


SPM<sub>mip</sub>  
[62, -22, 12]

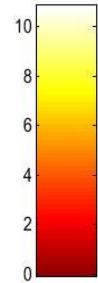
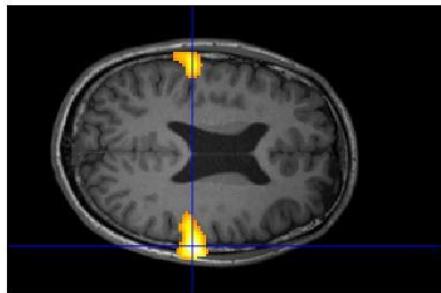
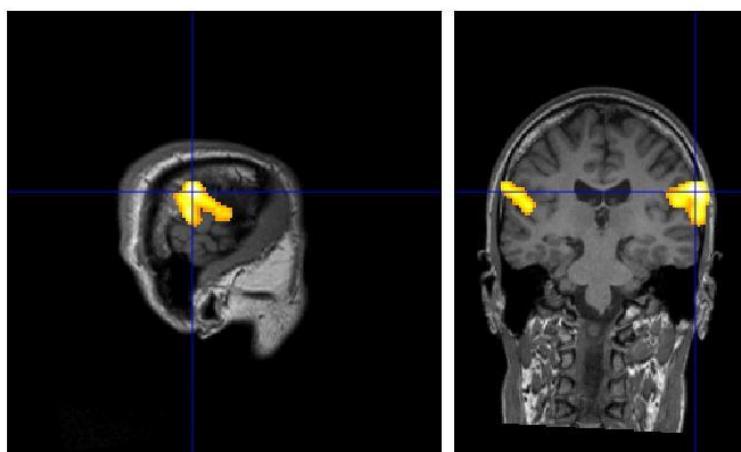


pos\_story2

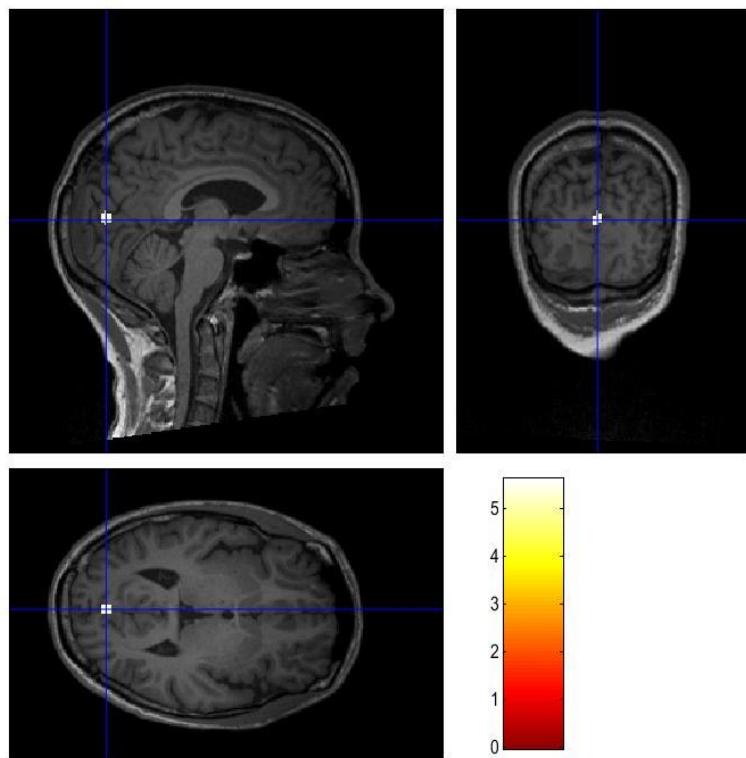
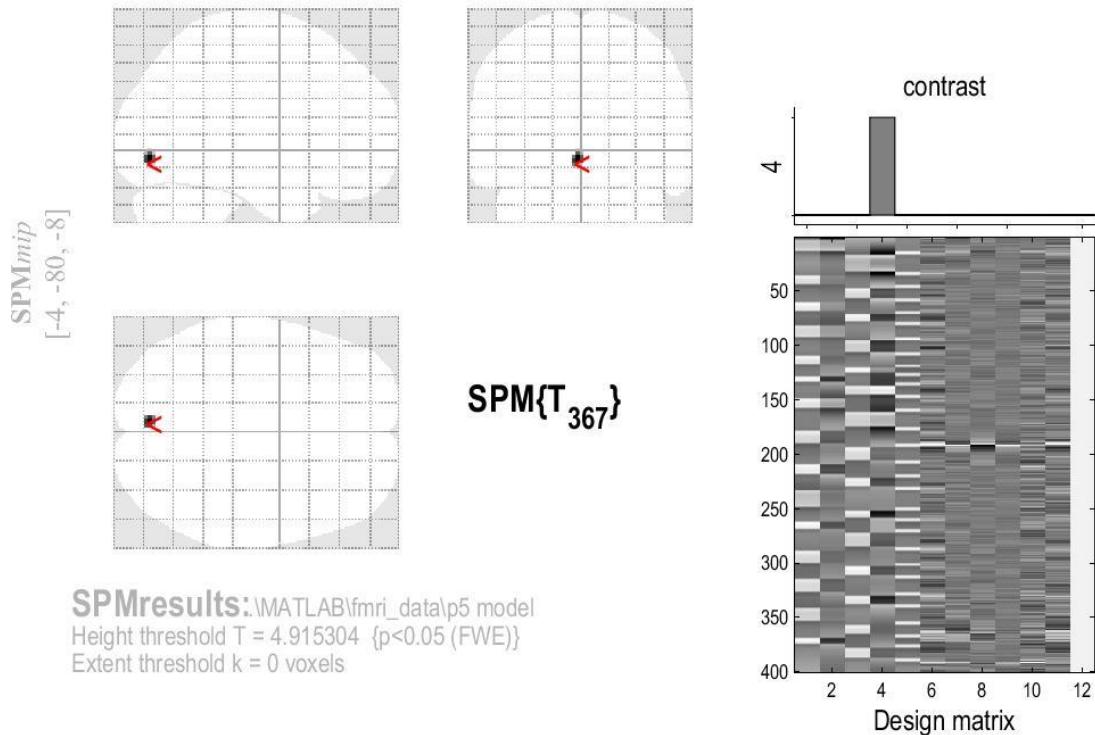
SPM{T<sub>367</sub>}

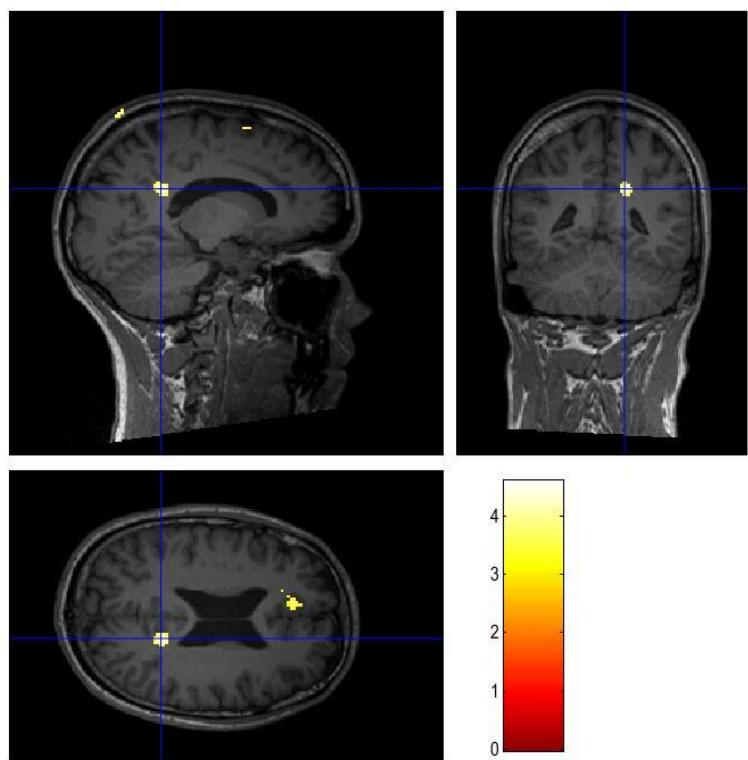
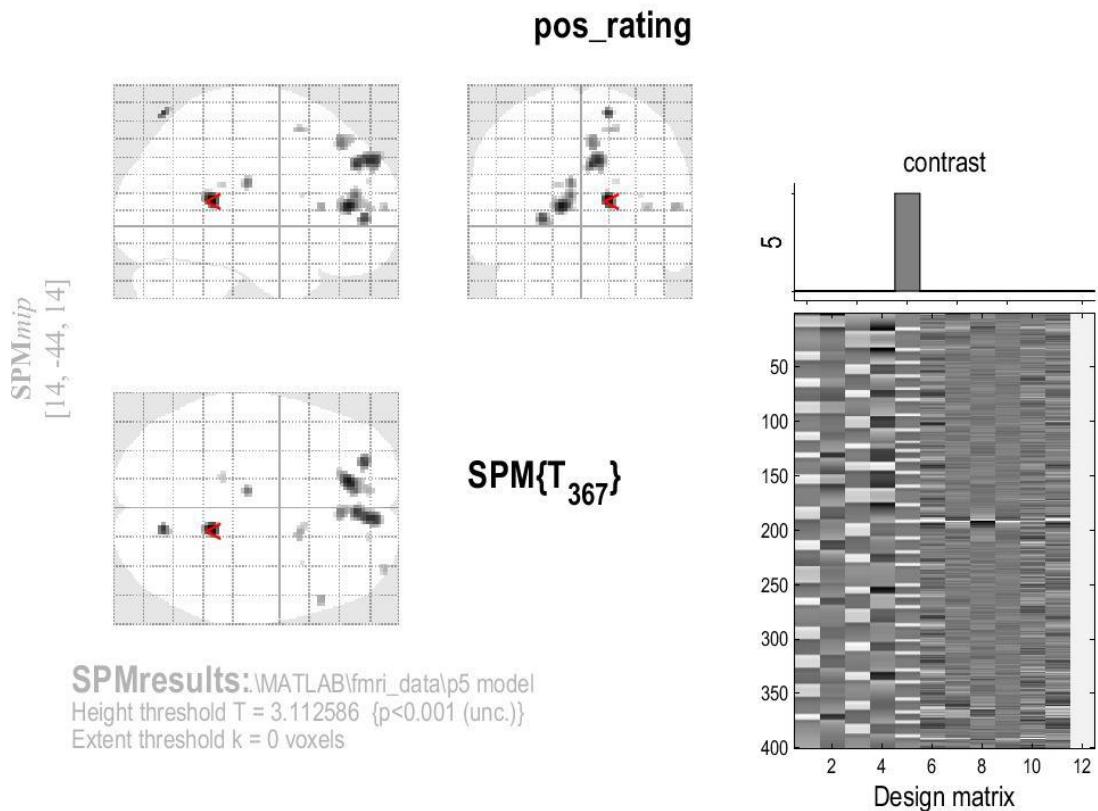


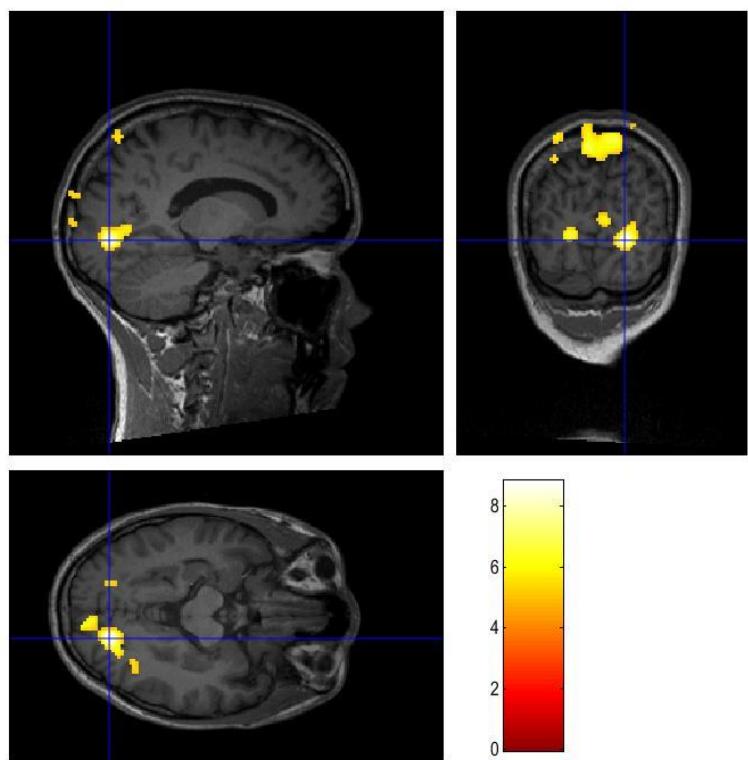
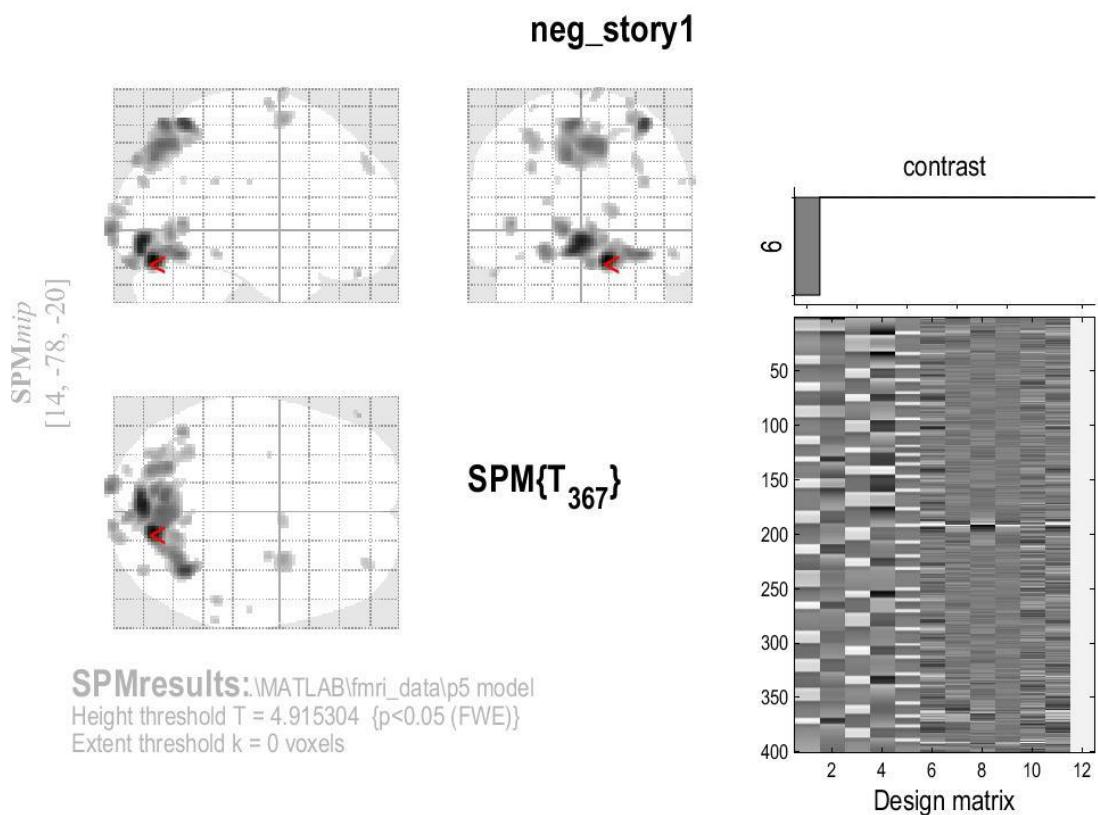
**SPMresults:** \MATLAB\fmri\_data\p5 model  
Height threshold T = 4.915304 {p<0.05 (FWE)}  
Extent threshold k = 0 voxels



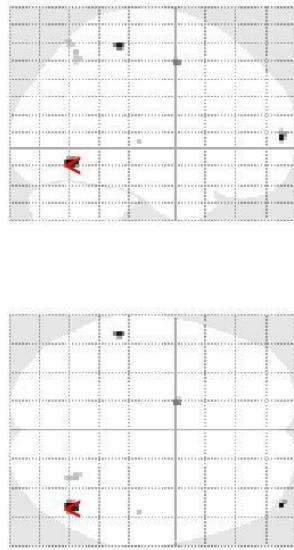
## pos\_story2rating



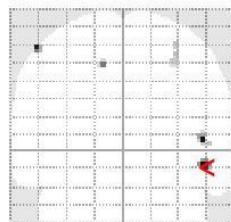




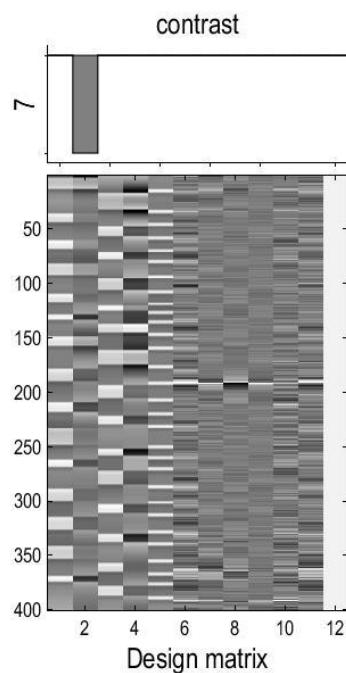
SPM<sub>mip</sub>  
[46, -66, -10]



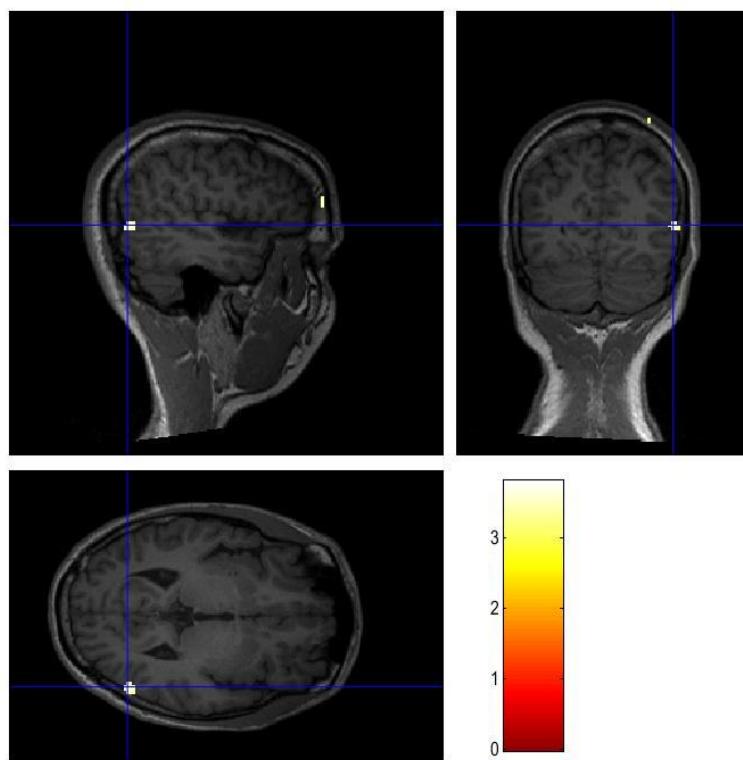
### neg\_story1rating



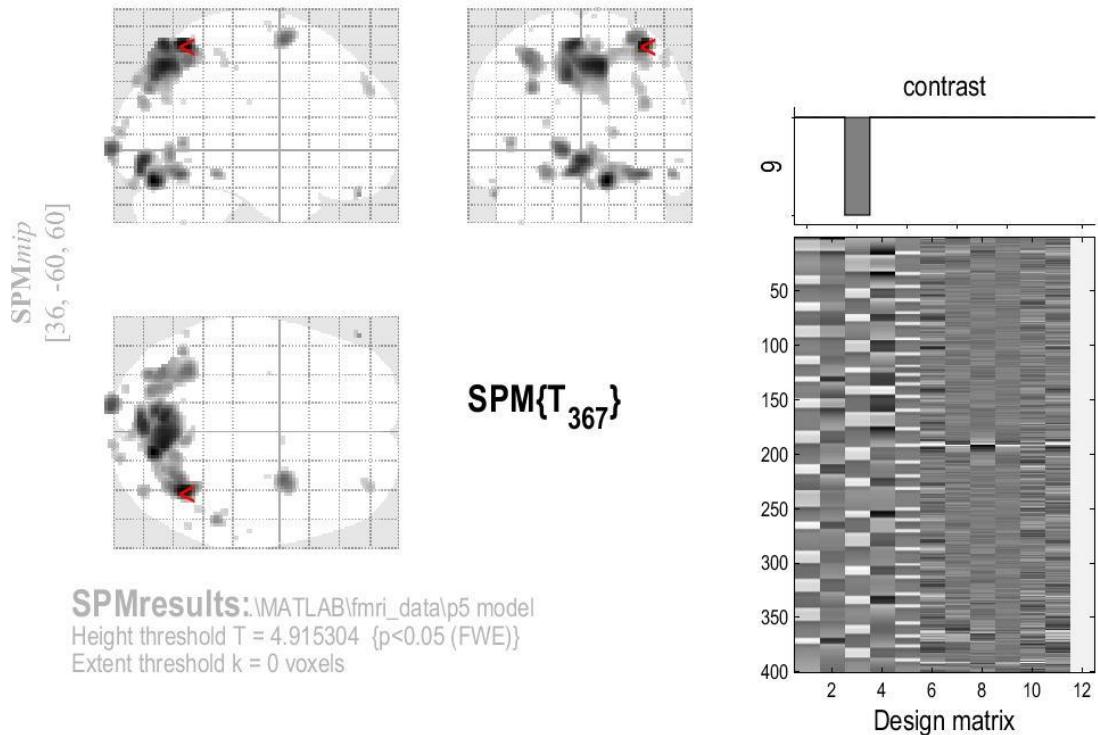
SPM{T<sub>367</sub>}



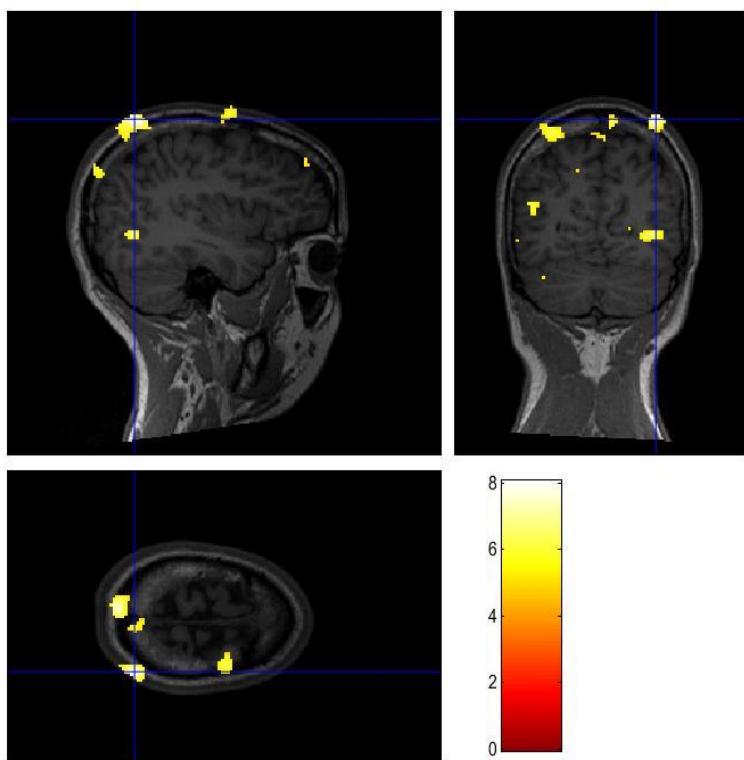
**SPM results:** \MATLAB\fmri\_data\p5 model  
Height threshold T = 3.112586 {p<0.001 (unc.)}  
Extent threshold k = 0 voxels



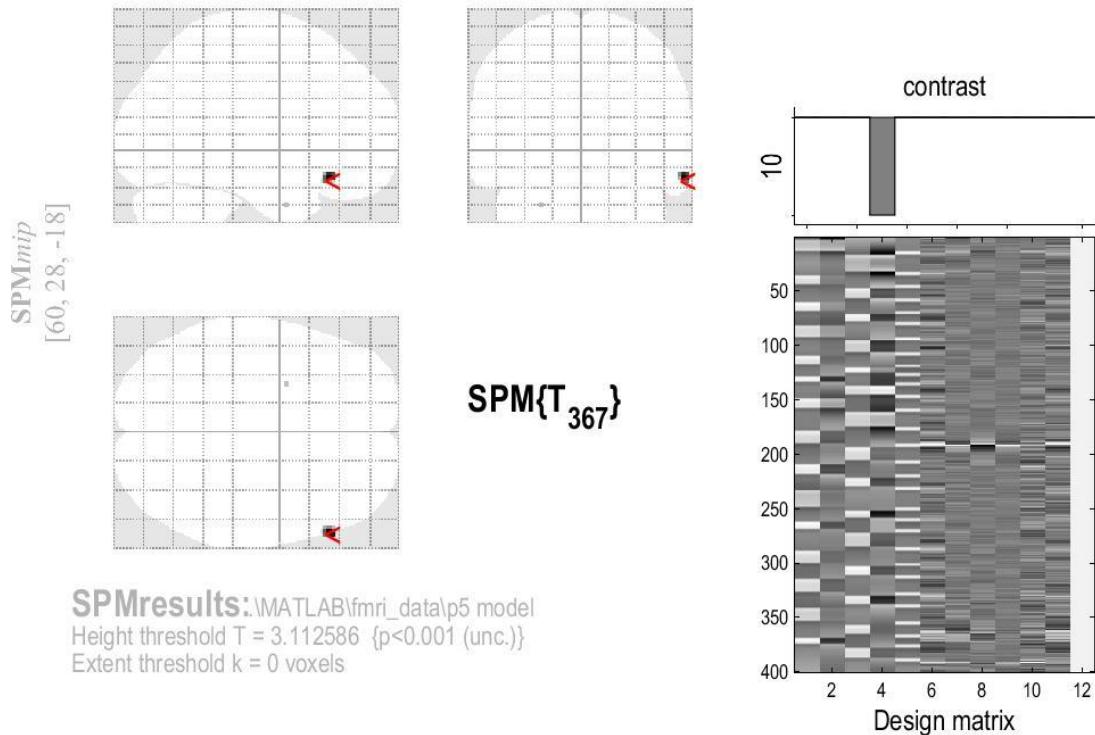
## **neg\_story2**



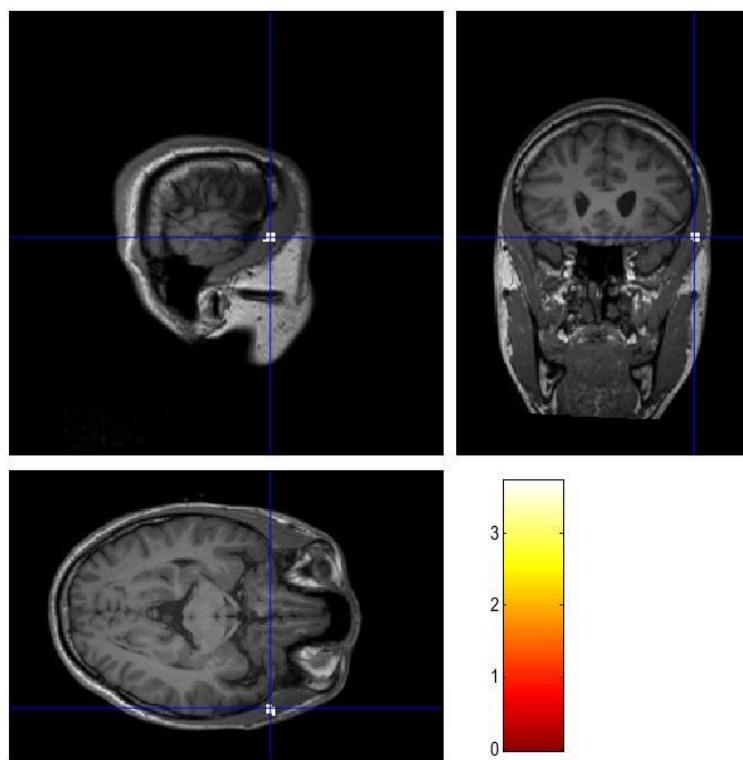
**SPMresults:** \MATLAB\fmri\_data\p5 model  
Height threshold  $T = 4.915304$  { $p < 0.05$  (FWE)}  
Extent threshold  $k = 0$  voxels

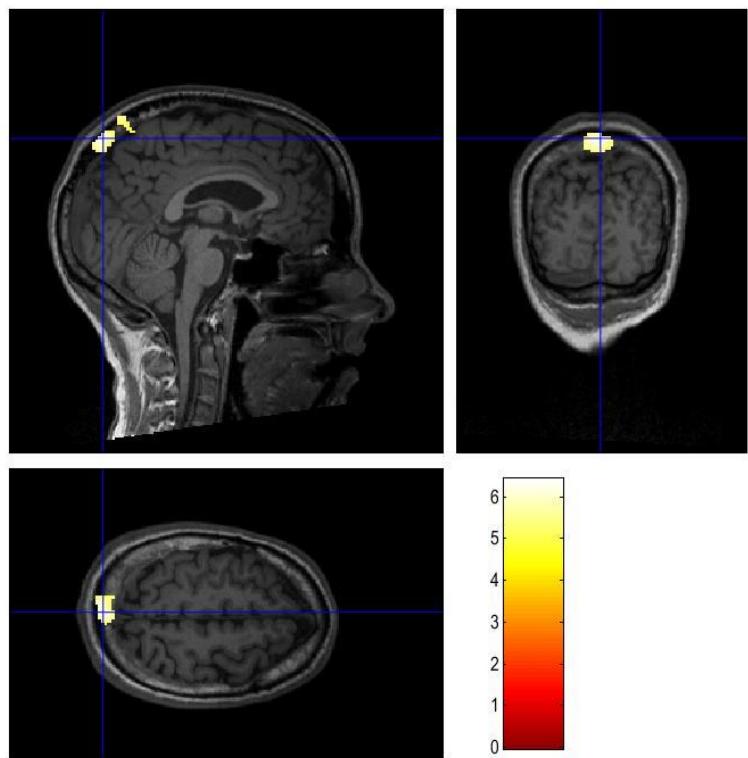
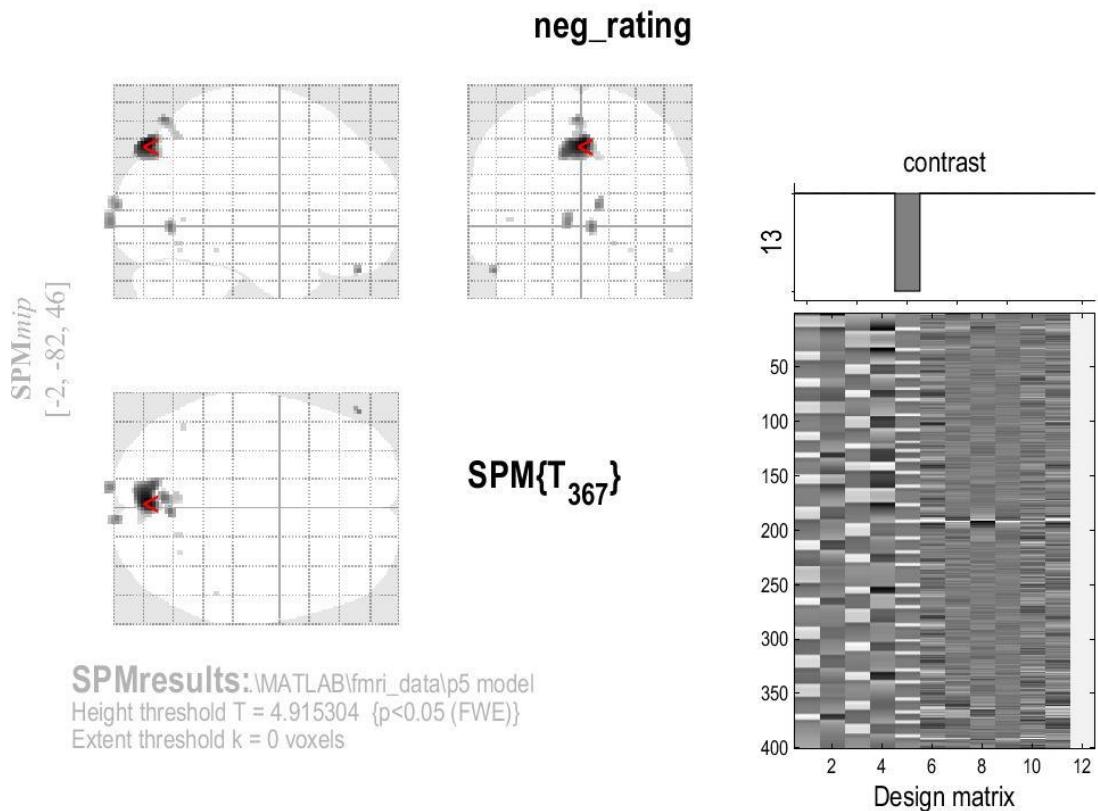


## **neg\_story2rating**

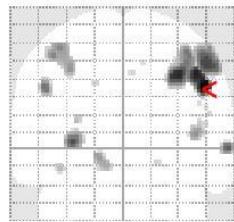
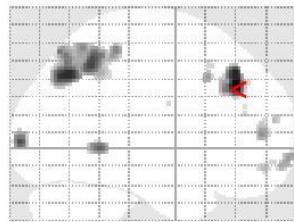


**SPMresults:** \MATLAB\fmri\_data\p5 model  
Height threshold T = 3.112586 {p<0.001 (unc.)}  
Extent threshold k = 0 voxels

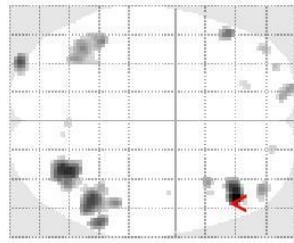




**SPM<sub>mip</sub>**  
[48, 34, 34]

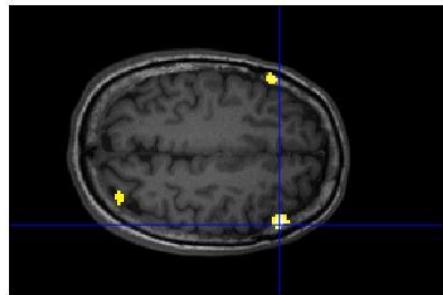
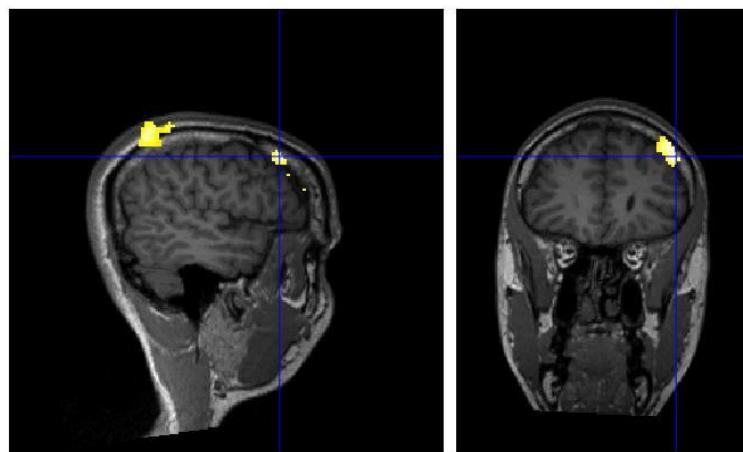
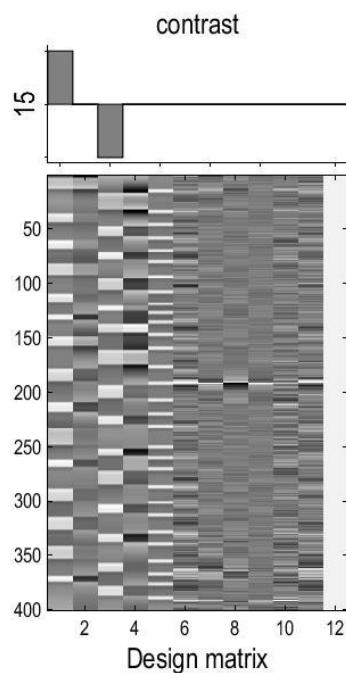


**SPM{T<sub>367</sub>}**

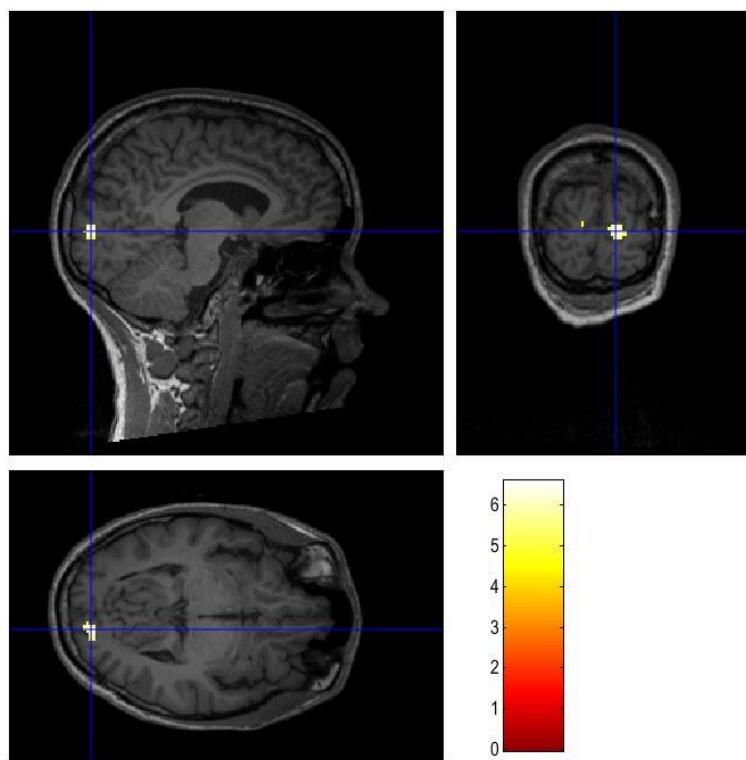
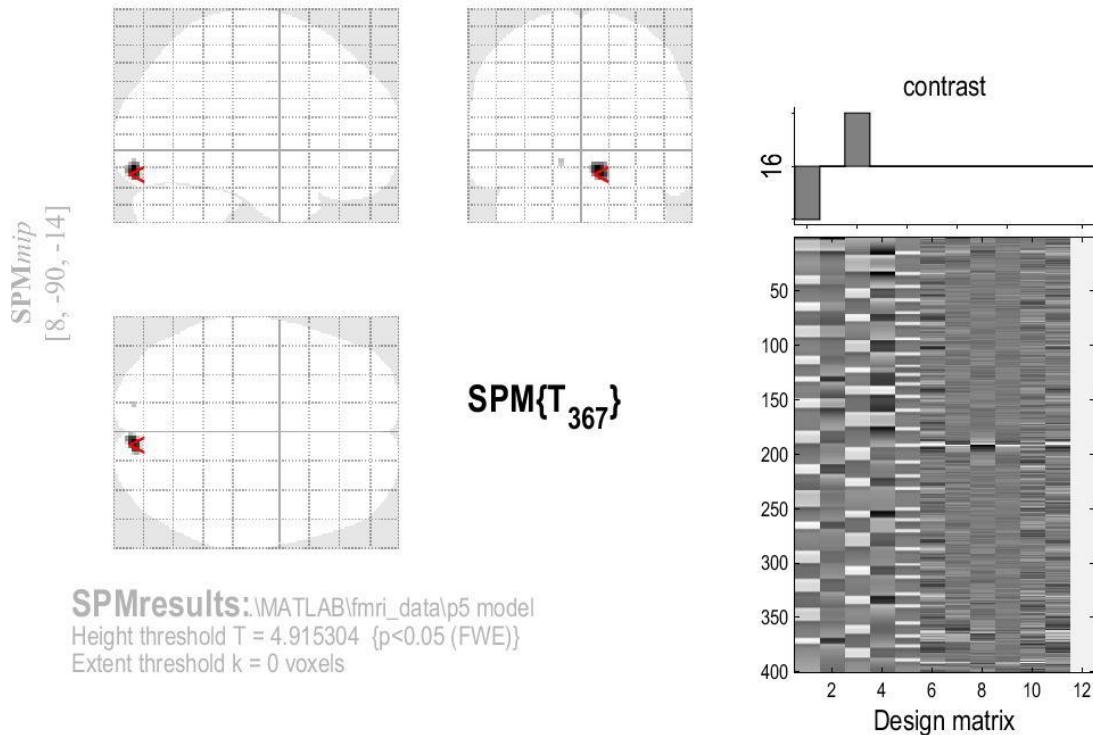


**SPM results:** \MATLAB\fmri\_data\p5 model  
Height threshold T = 3.112586 {p<0.001 (unc.)}  
Extent threshold k = 0 voxels

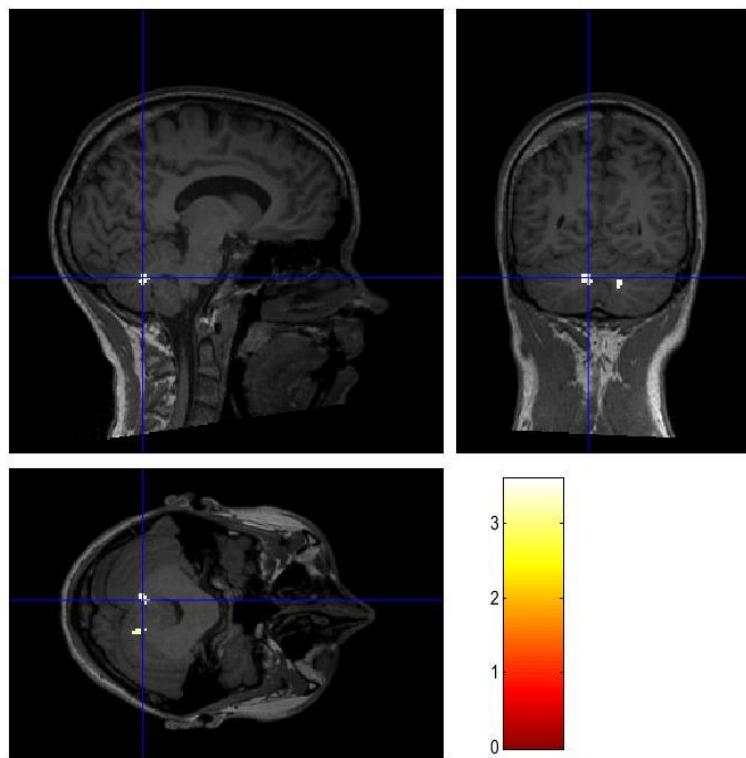
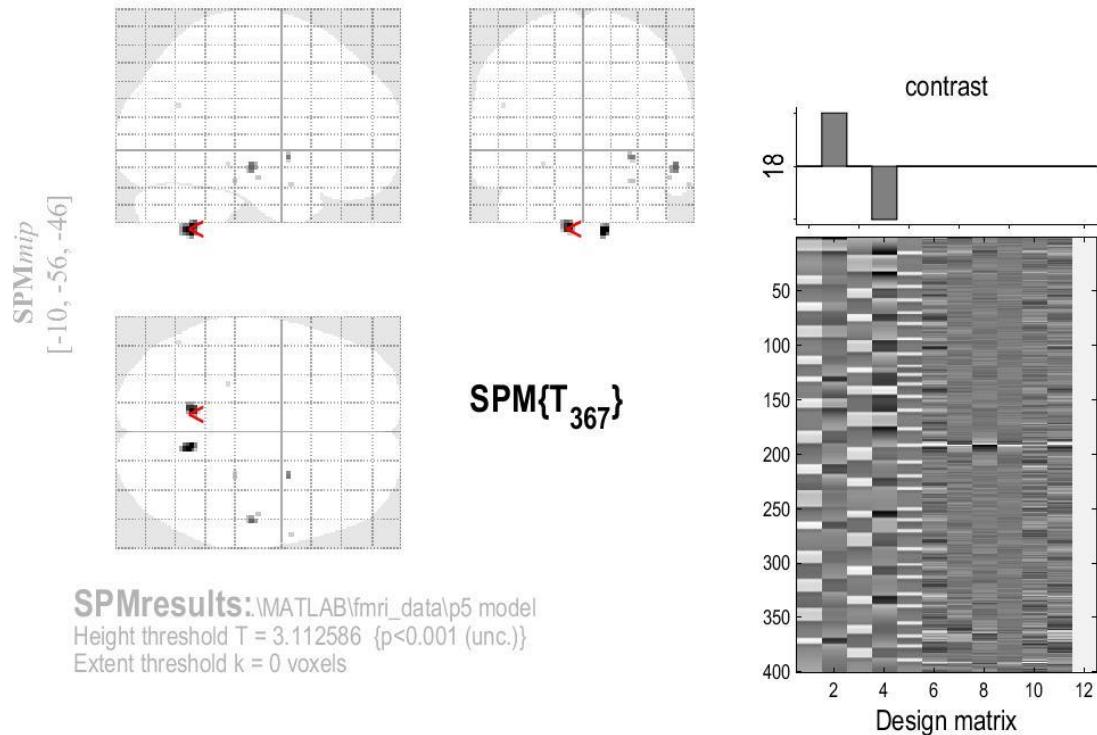
**story1-story2**



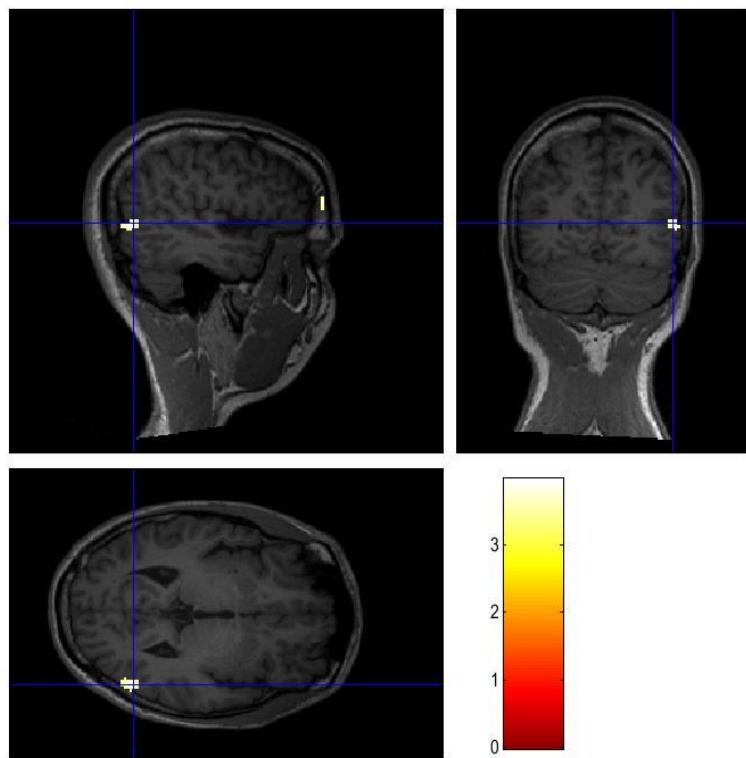
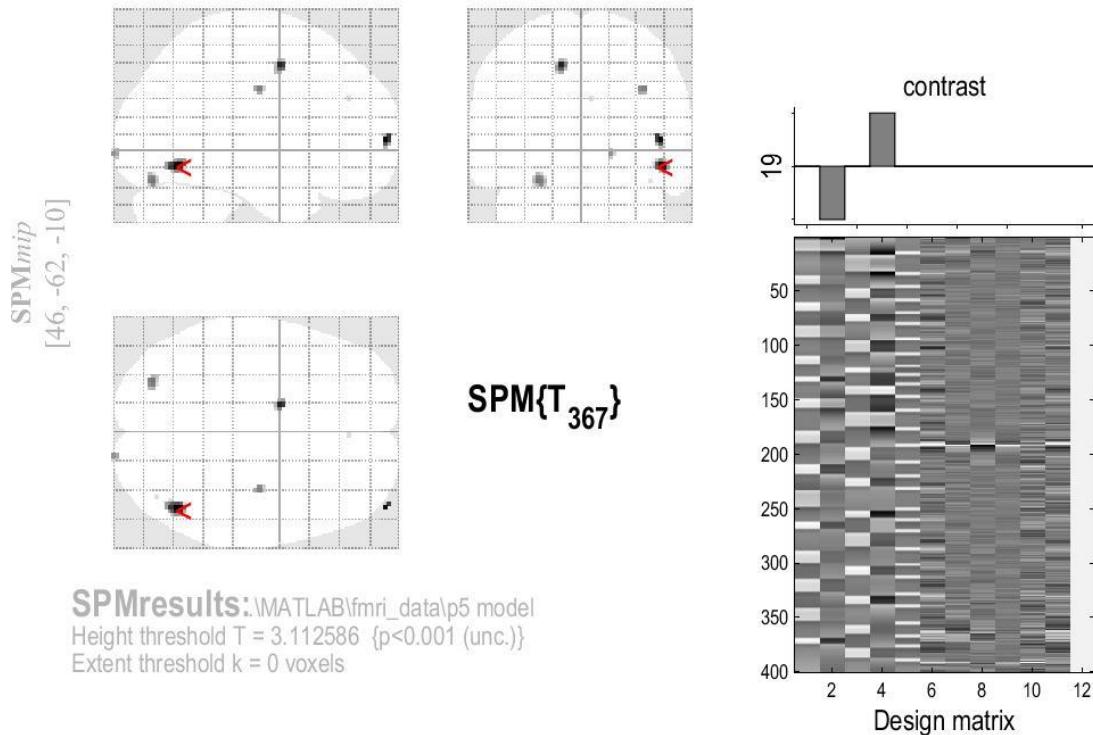
## story2-story1



## rating\_story1-story2



### rating\_story2-story1



**How many voxels are included in your analysis? Recall that a p-value reflects the probability of finding a specific effect, given that the null hypothesis is true. If all voxels were independent, how many voxels would then on average appear to be activated by chance in this analysis if using an uncorrected threshold of  $p<0.001$ ?**

From the output of the coordinate table for each of the contrasts it is possible to see how many voxels are included in the analysis. Similar for all of them 230838 voxels were included.

When using a threshold of  $p<0.001$  there would on average be 0.1% false positives (an effect would show, even though the null hypothesis is true). In our case that would mean that 231 voxels would on average show an effect, even though no true effect was present.

## Portfolio 7: Mixed effects

Group 3: Tobias, Riikka, Ruta Jana and Mie

March 27, 2019

```
# Setting the working directory
setwd("C:/Users/Mie/Desktop/Cognitive science/Experimental methods
2/Assignments/Portfolio 7")

# Loading libraries
library(pacman)
p_load(ggplot2, pastecs, nlme, tidyverse)

# Loading the data
face_exp_2016<- read.csv("face_exp_data_all_160310.csv", sep=";")
face_exp_2017<- read.csv("face_exp_all_logs_2017.csv", sep=";")

# Binding the two datasets together
face_exp1<-rbind(face_exp_2016,face_exp_2017)
# Conditions are coded in the "cond_blue", "cond_emo" and "freq" variables

# Checking if variables are factors
str(face_exp1)

## 'data.frame': 2400 obs. of 21 variables:
## $ rt : num 1.255 0.718 1 1.018 0.801 ...
## $ duration_measured: num 0.705 0.701 0.699 0.702 0.701 ...
## $ scan.day : Factor w/ 2 levels "Tuesday","Wednesday": 2 2 2 2 2 2 2 2
2 2 ...
## $ img : Factor w/ 4 levels "stimuli1b.png",...: 1 3 1 1 2 2 2 2 4
2 ...
## $ cond_emo : int 0 1 0 0 0 0 0 0 1 0 ...
## $ cond_blue : int 1 1 1 1 0 0 0 0 0 0 ...
## $ no : int 1 2 3 4 5 6 7 8 9 10 ...
## $ gender : Factor w/ 2 levels "female","male": 2 2 2 2 2 2 2 2 2 2
...
## $ age : int 21 21 21 21 21 21 21 21 21 21 ...
## $ duration_frames : int 42 42 42 42 42 42 42 42 42 42 ...
## $ response : Factor w/ 4 levels "", "b", "t", "y": 2 2 2 2 4 4 4 4 4 4
...
## $ onset : num 0.0192 5.5033 9.6533 12.4665 19.2804 ...
## $ key_t : num 1.27 6.22 10.65 13.48 20.08 ...
## $ delay_frames : int 286 206 126 366 286 206 366 126 206 206 ...
## $ offset : num 0.724 6.205 10.353 13.168 19.981 ...
## $ freq : Factor w/ 2 levels "b", "y": 2 2 2 2 2 2 2 2 2 2 ...
## $ correct_resp : int 1 1 1 1 1 1 1 1 1 1 ...
```

```

## $ ID                  : Factor w/ 25 levels "anders","blanka",...: 1 1 1 1 1 1 1 1
1 1 ...
## $ condition          : Factor w/ 1 level "face_exp": 1 1 1 1 1 1 1 1 1 ...
## $ scanner             : Factor w/ 2 levels "Scyra","Trio": 1 1 1 1 1 1 1 1 1
...
## $ year                : int  2016 2016 2016 2016 2016 2016 2016 2016 2016
...
#Changing the variables that were not factors
face_exp1$cond_emo <- as.factor(face_exp1$cond_emo)
face_exp1$cond_blue <- as.factor(face_exp1$cond_blue)
face_exp1$correct_resp <- as.factor(face_exp1$correct_resp)

```

## Part A

**1)**

**1.a. Comprehension question. Please explain which factor was between-participants and which were within-participants and why.**

Between-participants: Frequency, because each participant had only one frequency for the colors. Within-participants: Color and emotion, because each participant saw many colors and emotions.

**1.b. What was the age range of the participants?**

```

min(face_exp1$age) # 19
## [1] 19
max(face_exp1$age) # 27
## [1] 27

```

**2)**

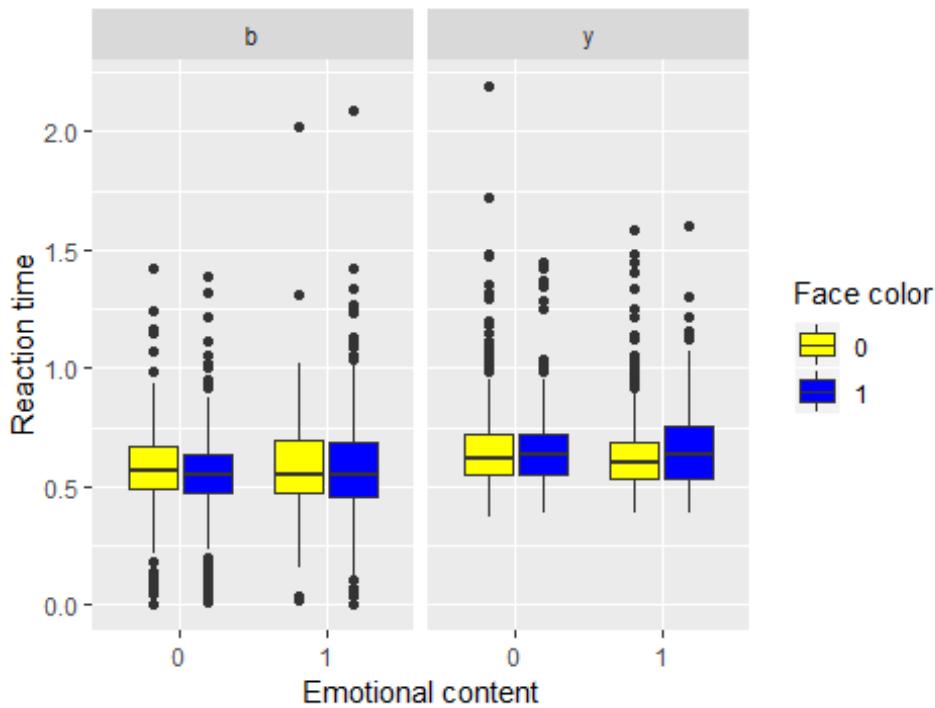
**2.a: Make a box-plot of the data with RT on the y-axis and emotional condition on the x-axis. Make a box-plot for each of the color conditions by using “fill”. Use facet\_wrap() to make two separate graphs for each frequency group. Give the boxes colors that matches the stimuli.**

```

boxplot <- ggplot(face_exp1, aes(cond_emo, rt, fill = cond_blue))
boxplot + geom_boxplot() +
  labs(title = "Boxplot of RT separated by emotional condition", x = "Emotional
content", y = "Reaction time", fill = "Face color") +
  facet_wrap(vars(freq)) + scale_fill_manual(values = c("yellow", "blue",
"yellow", "blue", "yellow", "blue", "yellow", "blue"))
## Warning: Removed 3 rows containing non-finite values (stat_boxplot).

```

Boxplot of RT separated by emotional condition



**2.b: Comprehension question. Explain why this plot shows that there is something wrong with the data.**

There seems to be many outliers in the data. There are some that took very little time to answer, and some of them even took 0 seconds which is impossible.

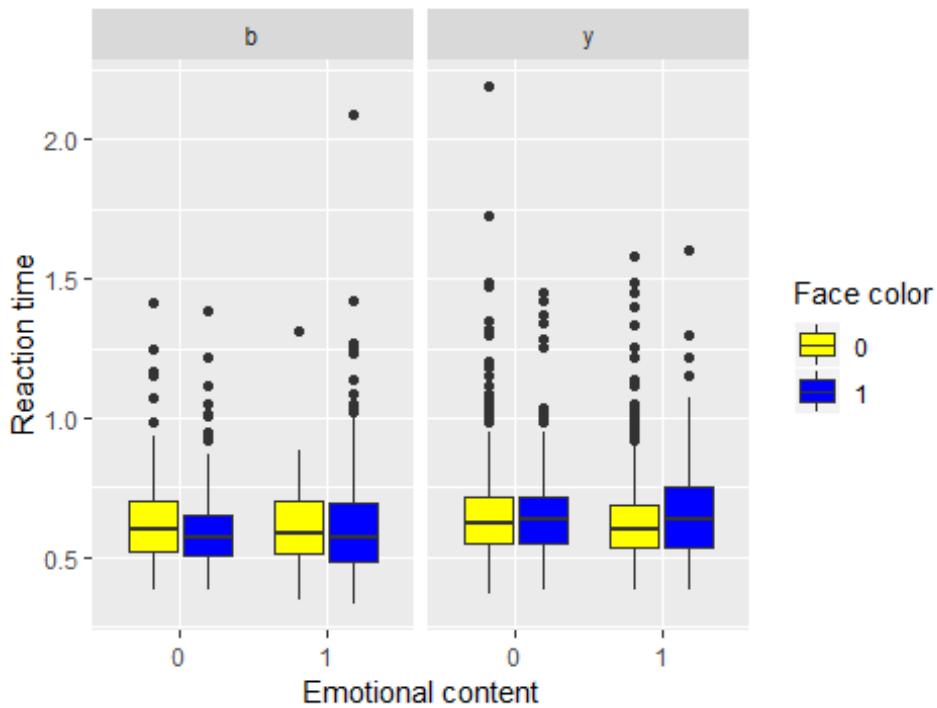
**2.c: Make a subset of the data, including only correct responses.**

```
# Filtering the data
face_exp <- filter(face_exp1, face_exp1$correct_resp == "1")
# keeps only the rows where the participant answered correctly
```

**2.d: Make another boxplot similar to that in 2.a. Did it solve the observed problem?**

```
boxplot2 <- ggplot(face_exp, aes(cond_emo, rt, fill = cond_blue))
boxplot2 + geom_boxplot() +
  labs(title = "Boxplot2 of RT separated by emotional condition", x = "Emotional content", y = "Reaction time", fill = "Face color") +
  facet_wrap(vars(freq)) + scale_fill_manual(values = c("yellow", "blue",
"yellow", "blue", "yellow", "blue", "yellow", "blue"))
```

Boxplot2 of RT separated by emotional condition



Yes, the problem seems

to have been solved.

**2.e.: Use the `by()` function and `stat.desc` (in `library(pastecs)`) to get descriptive measures for the different conditions (e.g. see Field's book chapter 5.5.3.2.). Try to 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?**

```
# Descriptive statistics for the reaction time between the blue/yellow face
# condition
by(data = face_exp$rt, INDICES = face_exp$cond_blue, stat.desc, basic = F)

## face_exp$cond_blue: 0
##      median      mean      SE.mean CI.mean.0.95      var
## 0.602198912 0.638994597 0.005035643 0.009879979 0.029465646
##      std.dev    coef.var
## 0.171655604 0.268633889
## -----
## face_exp$cond_blue: 1
##      median      mean      SE.mean CI.mean.0.95      var
## 0.600979187 0.624618041 0.005201635 0.010206873 0.028220459
##      std.dev    coef.var
## 0.167989462 0.268947502

# Descriptive statistics for the reaction time between the fearful/neutral
# condition
by(data = face_exp$rt, INDICES = face_exp$cond_emo, stat.desc, basic = F)
```

```

## face_exp$cond_emo: 0
##      median      mean      SE.mean CI.mean.0.95      var
## 0.602302368 0.635330286 0.005109437 0.010025273 0.028925827
##      std.dev      coef.var
## 0.170075945 0.267696895
## -----
## face_exp$cond_emo: 1
##      median      mean      SE.mean CI.mean.0.95      var
## 0.600937149 0.629026784 0.005133648 0.010072889 0.028910715
##      std.dev      coef.var
## 0.170031513 0.270308859

# We do the same for the reaction time of interaction between frequency and
# whether the participant was looking at a blue or a yellow face.
by(data = face_exp$rt, INDICES = list(face_exp$freq, face_exp$cond_blue),
stat.desc, basic = F)

## : b
## : 0
##      median      mean      SE.mean CI.mean.0.95      var
## 0.586712092 0.621161189 0.009198697 0.018107382 0.023777104
##      std.dev      coef.var
## 0.154198262 0.248241946
## -----
## : y
## : 0
##      median      mean      SE.mean CI.mean.0.95      var
## 0.602794179 0.644682665 0.005948624 0.011675146 0.031175177
##      std.dev      coef.var
## 0.176564936 0.273878834
## -----
## : b
## : 1
##      median      mean      SE.mean CI.mean.0.95      var
## 0.568222113 0.601009045 0.006671442 0.013102040 0.026927423
##      std.dev      coef.var
## 0.164095773 0.273033782
## -----
## : y
## : 1
##      median      mean      SE.mean CI.mean.0.95      var
## 0.634669934 0.657228642 0.008028866 0.015779992 0.028234659
##      std.dev      coef.var
## 0.168031719 0.255667068

```

**H1: The index finger (blue) trials will lead to a shorter response time than middle finger (yellow) trials.**

The mean reaction time measured in seconds for blue trials is  $M = 0.625$ ,  $SE = 0.005$ . For the yellow trials the mean is  $M = 0.639$ ,  $SE = 0.005$ . The hypothesis seems to be confirmed, since blue takes less time.

## H2: Fearful faces will yield a shorter response time than neutral.

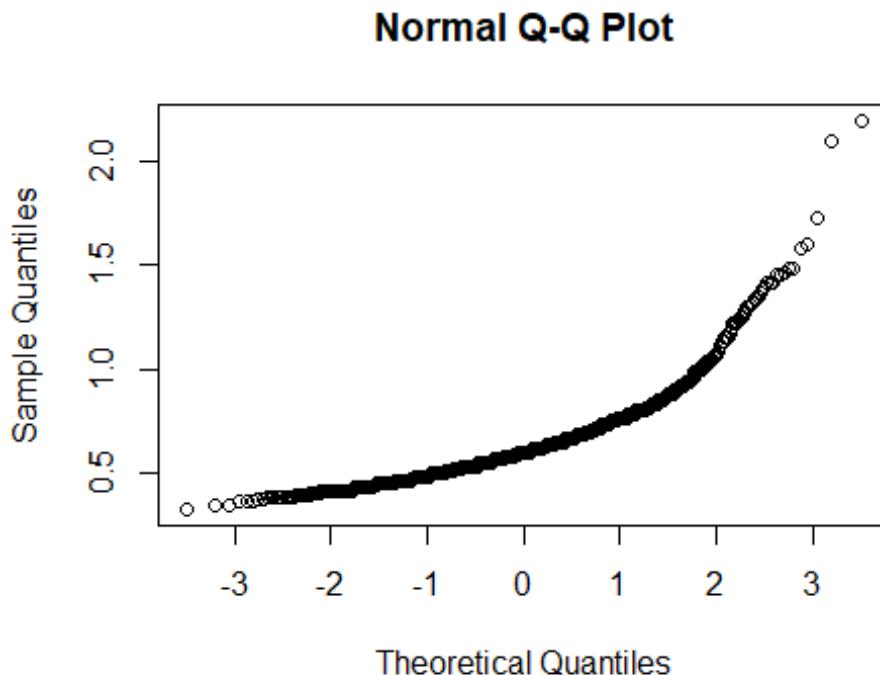
The mean reaction time for fearful faces is  $M = 0.629$ ,  $SE = 0.005$ . For the neutral faces the mean is  $M = 0.635$ ,  $SE = 0.005$ . The hypothesis seems to be confirmed, since fearful faces has a shorter response time.

## H3: Infrequent stimuli will yield longer response time than frequent. This should surface as an interaction between color and frequency group.

Frequent blue, color blue: mean = 0.601, SE = 0.007 Frequent blue, color yellow: mean = 0.621, SE = 0.009 Frequent yellow, color yellow: mean = 0.645, SE = 0.006 Frequent yellow, color blue: mean = 0.657, SE = 0.008 The infrequent stimuli seems to take longer time to answer, if we only look at individual frequent colors. But when blue is frequent and the color is blue, the reaction time is lower than when yellow is frequent and the color is yellow.

### 2.f: Explore if the RT data is normally distributed using a qq-plot (e.g. `qqnorm()`).

```
qqnorm(face_exp$rt)
```



```
stat.desc(face_exp$rt, basic = F, norm = T)
```

```

##      median      mean     SE.mean CI.mean.0.95      var
## 6.017652e-01 6.321943e-01 3.621248e-03 7.101415e-03 2.891513e-02
##      std.dev    coef.var   skewness   skew.2SE   kurtosis
## 1.700445e-01 2.689750e-01 2.251314e+00 2.159384e+01 1.023275e+01
##      kurt.2SE  normtest.W  normtest.p
## 4.909671e+01 8.470941e-01 7.608523e-42

```

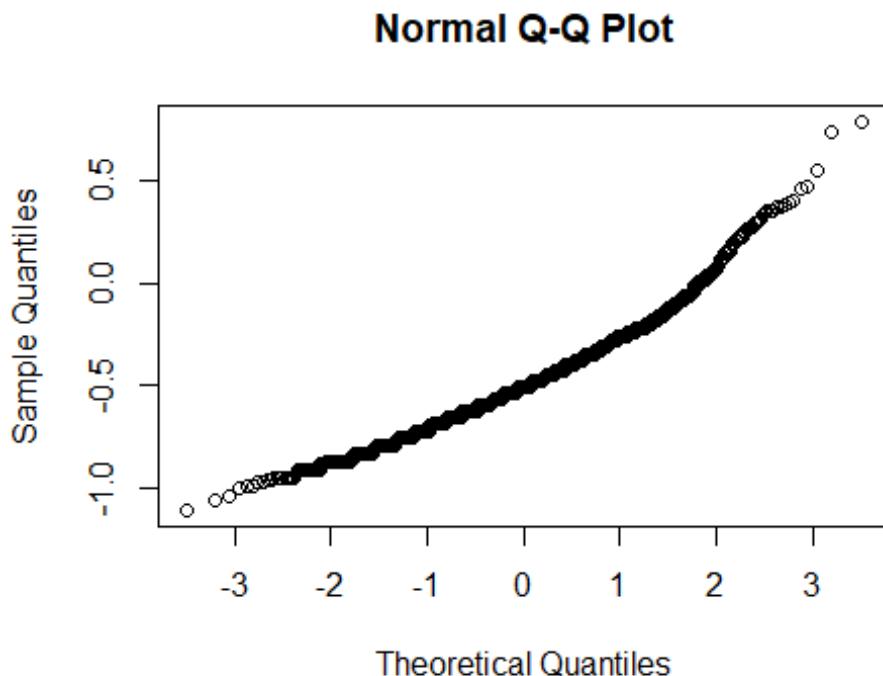
The reaction time does not seem to be normally distributed based on the QQ-plot. The normality test is highly significant,  $p < .0001$ , so the data is not normal.

## 2.g: log-transform the RT data.

```
face_exp$rt_transformed <- log(face_exp$rt)
```

## 2.h: Use a qq-plot to explore if the transformed data appear more normal than the untransformed.

```
qqnorm(face_exp$rt_transformed)
```



```

stat.desc(face_exp$rt_transformed, norm = T)
##      nbr.val      nbr.null      nbr.na      min      max
## 2.205000e+03  0.000000e+00  0.000000e+00 -1.110294e+00  7.863963e-01
##      range        sum      median       mean      SE.mean
## 1.896691e+00 -1.076650e+03 -5.078879e-01 -4.882768e-01  5.013825e-03
##      CI.mean.0.95      var      std.dev    coef.var      skewness

```

```

## 9.832317e-03 5.543027e-02 2.354363e-01 -4.821780e-01 8.186784e-01
## skew.2SE kurtosis kurt.2SE normtest.W normtest.p
## 7.852486e+00 1.636163e+00 7.850306e+00 9.665418e-01 3.198046e-22

```

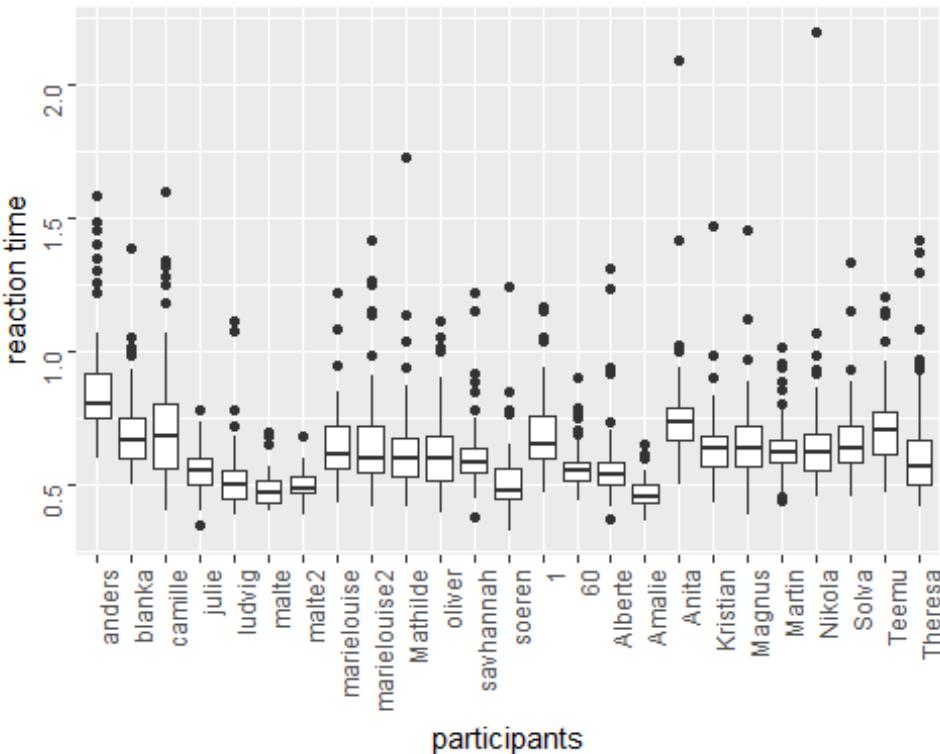
The transformation somewhat helped normalizing the data. However, the p-value is still showing the distribution to be significantly different from a normal distribution.

## 2.i: Make a plot that explores the response times for participants, individually, using a boxplot. Does anybody stick out as unusual?

```

boxplot3 <- ggplot(face_exp, aes(ID, rt))
boxplot3 + geom_boxplot() + labs(x = "participants", y = "reaction time") +
  theme(axis.text = element_text(angle = 90, hjust = 1)) #This Line rotates the
  participant names so we can get an overview of who the outliers are

```



There are some participant with one reaction time that is above 2 seconds which seems very long. Maybe this is due to a loss of concentration or motivation to do the task. These are Anita and Nikola.

### 3) Data analysis

3.a Make mixed effects model where you predict reaction time using the three factors as fixed effects, and include random intercepts for each participant (use “ID” from the log). Include 2-way and 3-way interactions as well. To do this use `lme()` from the “nlme” package, and use maximum-likelihood as estimation method(`method = “ML”`).

```
# Model with the log transformed reaction times
model <- lme(rt_transformed ~ cond_blue * cond_emo * freq, random = ~1|ID, data =
face_exp, method = "ML")
```

3.b.: Report the t-statistics using `summary()`.

```
# Summary of the model
summary(model)

## Linear mixed-effects model fit by maximum likelihood
##  Data: face_exp
##          AIC      BIC    logLik
## -921.7042 -864.7194 470.8521
##
## Random effects:
##   Formula: ~1 | ID
##             (Intercept) Residual
## StdDev:  0.1321929 0.1913286
##
## Fixed effects: rt_transformed ~ cond_blue * cond_emo * freq
##                  Value Std.Error DF   t-value p-value
## (Intercept)      -0.5042037 0.04314318 2174 -11.686754 0.0000
## cond_blue1       -0.0611733 0.01962203 2174 -3.117586 0.0018
## cond_emo1        -0.0355707 0.02288740 2174 -1.554162 0.1203
## freqy            0.0494312 0.05654396  23  0.874209 0.3910
## cond_blue1:cond_emo1 0.0546449 0.02769730 2174  1.972931 0.0486
## cond_blue1:freqy  0.0668937 0.02518765 2174  2.655813 0.0080
## cond_emo1:freqy  0.0047882 0.02628060 2174  0.182193 0.8554
## cond_blue1:cond_emo1:freqy -0.0224106 0.03563167 2174 -0.628952 0.5294
## Correlation:
##                  (Intr) cnd_b1 cnd_m1 freqy cn_1:_1 cnd_b1:
## cond_blue1       -0.313
## cond_emo1        -0.268  0.589
## freqy            -0.763  0.239  0.205
## cond_blue1:cond_emo1 0.221 -0.707 -0.826 -0.169
## cond_blue1:freqy  0.244 -0.779 -0.459 -0.245  0.551
## cond_emo1:freqy  0.234 -0.513 -0.871 -0.234  0.719   0.525
## cond_blue1:cond_emo1:freqy -0.172  0.550  0.642  0.172 -0.777 -0.706
##                  cnd_m1:
## cond_blue1
## cond_emo1
```

```

## freq
## cond_blue1:cond_emo1
## cond_blue1:freq
## cond_emo1:freq
## cond_blue1:cond_emo1:freq -0.737
##
## Standardized Within-Group Residuals:
##      Min       Q1       Med       Q3       Max
## -2.79611058 -0.62288065 -0.09371291  0.45667751  6.44012772
##
## Number of Observations: 2205
## Number of Groups: 25

```

Main effects: There is a significant main effect of color,  $t = -3.12$ ,  $p < .01$ . However, the main effect of emotion seems non-significant,  $t = -1.55$ ,  $p = .12$ . Likewise, the main effect of frequency is also non-significant,  $t = 0.87$ ,  $p = .39$ .

Interaction effects: The interaction effects between color and emotion is significant,  $t = 1.97$ ,  $p < .05$  and so is the interaction effect between color and frequency,  $t = 2.66$ ,  $p < .01$ . However, interaction effects for emotion and frequency does not appear significant,  $t = 0.18$ ,  $p = .86$ . Neither does the interaction of color, emotion and frequency,  $t = -0.63$ ,  $p = .53$ .

### 3.c: Report the F-statistics using anova() and type='sequential', which gives you type='I' analysis.

```

# F-statistics using type I sum of squares
anova(model, type = "sequential")

##                               numDF denDF   F-value p-value
## (Intercept)                  1    2174 346.6300 <.0001
## cond_blue                   1    2174   0.0115  0.9147
## cond_emo                     1    2174   1.8874  0.1696
## freq                         1     23   2.3160  0.1417
## cond_blue:cond_emo           1    2174   7.0788  0.0079
## cond_blue:freq                1    2174   9.7815  0.0018
## cond_emo:freq                1    2174   0.1736  0.6770
## cond_blue:cond_emo:freq      1    2174   0.3956  0.5294

```

There seems to be two significant effects, one is the interaction between emotion and color,  $F(2174) = 7.08$ ,  $p < .01$ . The other is the interaction between color and frequency,  $F(2174) = 9.78$ ,  $p < .01$ . All the other effects appear non-significant.

### 3.d.: Report the F-statistics using anova() and type='marginal'. Why might there be differences between results from 3.c and 3.d?

```

# F-statistics using type II sum of squares
anova(model, type = "marginal")

```

```

##                               numDF denDF   F-value p-value
## (Intercept)                  1    2174 136.58022 <.0001
## cond_blue                    1    2174   9.71934  0.0018
## cond_emo                     1    2174   2.41542  0.1203
## freq                          1     23   0.76424  0.3910
## cond_blue:cond_emo          1    2174   3.89246  0.0486
## cond_blue:freq                1    2174   7.05335  0.0080
## cond_emo:freq                1    2174   0.03319  0.8554
## cond_blue:cond_emo:freq      1    2174   0.39558  0.5294

```

When using type II sum of squares for our analysis of variance, we get the same significant interaction effects of color and emotion,  $F(2174) = 3.89$ ,  $p < .05$ , and for color and frequency,  $F(2174) = 7.05$ ,  $p < .01$ . In addition, the type II sum of squares show a significant effect of color,  $F(2174) = 9.72$ ,  $p < .01$ .

Type sequential in 3c used type I sum of squares, where factors are tested in the order they are fed in the model. Type marginal uses type II sums of squares, which tests for each main effect after the other main effect. It does not take any interaction terms into account.

**3.e.: Make a new model including a random slope from trial number ('no' in the log-file). Repeat 3.b. What does the inclusion of such a random slope model? Did it change the results?**

```

# Making a second model including a random slope
model2 <- lme(rt_transformed ~ cond_blue * cond_emo * freq, random = ~1+no|ID,
data = face_exp, method = "ML")
# Summary of the second model
summary(model2)

## Linear mixed-effects model fit by maximum likelihood
## Data: face_exp
##           AIC       BIC       logLik
## -951.6366 -883.2548  487.8183
##
## Random effects:
## Formula: ~1 + no | ID
## Structure: General positive-definite, Log-Cholesky parametrization
##             StdDev     Corr
## (Intercept) 0.133853748 (Intr)
## no          0.001206009 -0.24
## Residual    0.188433336
##
## Fixed effects: rt_transformed ~ cond_blue * cond_emo * freq
##                   Value Std.Error DF   t-value p-value
## (Intercept)      -0.5161524 0.04265825 2174 -12.099711 0.0000
## cond_blue1       -0.0590149 0.01938506 2174  -3.044349 0.0024
## cond_emo1        -0.0354338 0.02260929 2174  -1.567224 0.1172
## freq              0.0611364 0.05588388    23   1.093990 0.2853

```

```

## cond_blue1:cond_emo1      0.0567175 0.02735186 2174   2.073625  0.0382
## cond_blue1:freqy         0.0640529 0.02489404 2174   2.573023  0.0101
## cond_emo1:freqy         0.0041543 0.02597105 2174   0.159960  0.8729
## cond_blue1:cond_emo1:freqy -0.0272428 0.03521495 2174   -0.773613  0.4392
## Correlation:
##                               (Intr) cnd_b1 cnd_m1 freqy cn_1:_1 cnd_b1:
## cond_blue1                 -0.316
## cond_emo1                  -0.267  0.587
## freqy                      -0.763  0.241  0.204
## cond_blue1:cond_emo1       0.222 -0.706 -0.825 -0.170
## cond_blue1:freqy          0.246 -0.779 -0.457 -0.246  0.549
## cond_emo1:freqy          0.232 -0.511 -0.871 -0.233  0.718   0.524
## cond_blue1:cond_emo1:freqy -0.173  0.548  0.641  0.174 -0.777 -0.705
##                               cnd_m1:
## cond_blue1
## cond_emo1
## freqy
## cond_blue1:cond_emo1
## cond_blue1:freqy
## cond_emo1:freqy
## cond_blue1:cond_emo1:freqy -0.736
##
## Standardized Within-Group Residuals:
##      Min        Q1        Med        Q3        Max
## -2.8498046 -0.6183224 -0.1120300  0.4594741  6.3231814
##
## Number of Observations: 2205
## Number of Groups: 25

```

The inclusion of the random slope takes into account that participants might have different reactions to different trials. With this, we can observe that the effect of color appears significant,  $t(2174) = -3.04$ ,  $p < .01$ . Likewise, the interaction between color and emotion seems significant,  $t(2174) = 2.07$ ,  $p < .05$  as well as the interaction between color and frequency,  $t(2175) = 2.57$ ,  $p = .01$ .

If we compare with the results from the model from 3.b, it is the same effects that seem significant.

### 3.f: Make a model comparison of model 3.a and 3.e using `anova()`. Did the inclusion of a random slope significantly improve the model?

```

# Comparing model and model2
anova(model, model2)

##      Model df     AIC     BIC   logLik   Test L.Ratio p-value
## model     1 10 -921.7042 -864.7194 470.8521
## model2    2 12 -951.6366 -883.2548 487.8183 1 vs 2 33.93231 <.0001

```

Random slope did significantly improve the model,  $X^2 = 33.93$ ,  $p < .0001$ .

**3.g.: Response times are correlated in time which goes against the assumption of independence. It might therefore be an idea to model this by including a so-called auto-regressive component in the model (e.g. this is default in SPM analyses of fMRI-data). In lme(), this is done by adding the following to the model specification: "cor=corAR1(form=~1|ID)". Make a new model comparison. Does that have an effect?**

```
# Making a new model including an auto-regressive component
model3 <- lme(rt_transformed ~ cond_blue * cond_emo * freq, random = ~1+no|ID,
data = face_exp, method = "ML", cor = corAR1(form = ~1|ID))
# Summary of the new model
summary(model3)

## Linear mixed-effects model fit by maximum likelihood
## Data: face_exp
##          AIC      BIC    logLik
## -979.5929 -905.5126 502.7964
##
## Random effects:
## Formula: ~1 + no | ID
## Structure: General positive-definite, Log-Cholesky parametrization
##             StdDev     Corr
## (Intercept) 0.132601179 (Intr)
## no          0.001138109 -0.216
## Residual    0.189247286
##
## Correlation Structure: AR(1)
## Formula: ~1 | ID
## Parameter estimate(s):
##           Phi
## 0.1280107
## Fixed effects: rt_transformed ~ cond_blue * cond_emo * freq
##                  Value Std.Error DF t-value p-value
## (Intercept) -0.5148625 0.04262693 2174 -12.078338 0.0000
## cond_blue1 -0.0567939 0.01900354 2174 -2.988596 0.0028
## cond_emo1 -0.0380293 0.02219935 2174 -1.713083 0.0868
## freqy 0.0648832 0.05587675 23 1.161184 0.2575
## cond_blue1:cond_emo1 0.0580627 0.02683977 2174 2.163310 0.0306
## cond_blue1:freqy 0.0558958 0.02447713 2174 2.283593 0.0225
## cond_emo1:freqy 0.0029192 0.02552226 2174 0.114377 0.9089
## cond_blue1:cond_emo1:freqy -0.0247166 0.03456095 2174 -0.715159 0.4746
## Correlation:
##                  (Intr) cnd_b1 cnd_m1 freqy cn_1:_1 cnd_b1:
## cond_blue1 -0.309
## cond_emo1 -0.262 0.583
## freqy -0.763 0.236 0.200
## cond_blue1:cond_emo1 0.217 -0.705 -0.822 -0.166
```

```

## cond_blue1:freqy          0.240 -0.776 -0.452 -0.241  0.548
## cond_emo1:freqy          0.228 -0.507 -0.870 -0.229  0.715   0.521
## cond_blue1:cond_emo1:freqy -0.169  0.548  0.638  0.171 -0.777 -0.706
##                                         cnd_m1:
## cond_blue1
## cond_emo1
## freqy
## cond_blue1:cond_emo1
## cond_blue1:freqy
## cond_emo1:freqy
## cond_blue1:cond_emo1:freqy -0.734
##
## Standardized Within-Group Residuals:
##      Min       Q1       Med       Q3       Max
## -2.8589690 -0.6186939 -0.1115944  0.4530465  6.2730827
##
## Number of Observations: 2205
## Number of Groups: 25

# Comparing the new model to the two previous models
anova(model, model2, model3)

##      Model df     AIC     BIC   logLik   Test L.Ratio p-value
## model     1 10 -921.7042 -864.7194 470.8521
## model2    2 12 -951.6366 -883.2548 487.8183 1 vs 2 33.93231 <.0001
## model3    3 13 -979.5929 -905.5126 502.7964 2 vs 3 29.95630 <.0001

```

Adding the auto-regressive component does significantly improve the model,  $X^2 = 29.96$ ,  $p < .0001$ . However, the results show the same effects to be significant, though with altered p-values.

## 4. Results and interpretation.

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

**Below are some ideas that you may want to consider:**

**Rule number 1: Report the first model you did.**

**Rule number 2: Report the most sensible model.**

**Rule number 3: Report the simplest model.**

**Rule number 4: Report the most extensive and complete model.**

Since we more or less get the same results either way, we choose the first model which is also the simplest.

**4.b: Throughout part 3 of this exercise we made several models to choose from. What is the problem of this strategy? (This is analogous to the motivation for using family-wise-error corrected p-values in the SPM analysis)**

By making several models, we would be more inclined to choose the most complicated model that can explain more of our data therefore putting us in greater risk of getting false positives.

**4.c. 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.**

H1: Reaction time for the index finger (blue) is quicker than for the middle finger (yellow) which also turned out to be significant,  $t(2174) = -3.12$ ,  $p < .01$ . Therefore, we cannot confirm the null hypothesis which provides evidence for H1.

H2: The reaction times for emotional and normal faces show no significant difference between them,  $t(2174) = -1.55$ ,  $p = .12$ . Therefore, the null hypothesis can not be rejected.

H3: There is a significant interaction between color and frequency,  $t(2174) = 2.66$ ,  $p < .01$ . Therefore, we cannot confirm the null hypothesis which supports H3.

## Part B

**5)**

**5.a. Find the data on Blackboard, load it and report figure and analysis using the code below.**

```
#Load data
trypt_long<-read.csv(file='trypt_long.csv',header=TRUE,sep=",")
trypt_long$ID<-as.factor(trypt_long$ID)
trypt_long$time<-as.factor(trypt_long$time)

#use ggline to make nice Line plot. Install ggpibr, if you haven't got it
p_load(ggpibr)

## Installing package into 'C:/Users/Mie/Documents/R/win-library/3.5'
## (as 'lib' is unspecified)

## also installing the dependencies 'ggrepel', 'ggsci', 'cowplot', 'ggsignif',
'polynom'

## Warning: unable to access index for repository
http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.5:
##   cannot open URL
'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.5/PACKAGES'
```

```

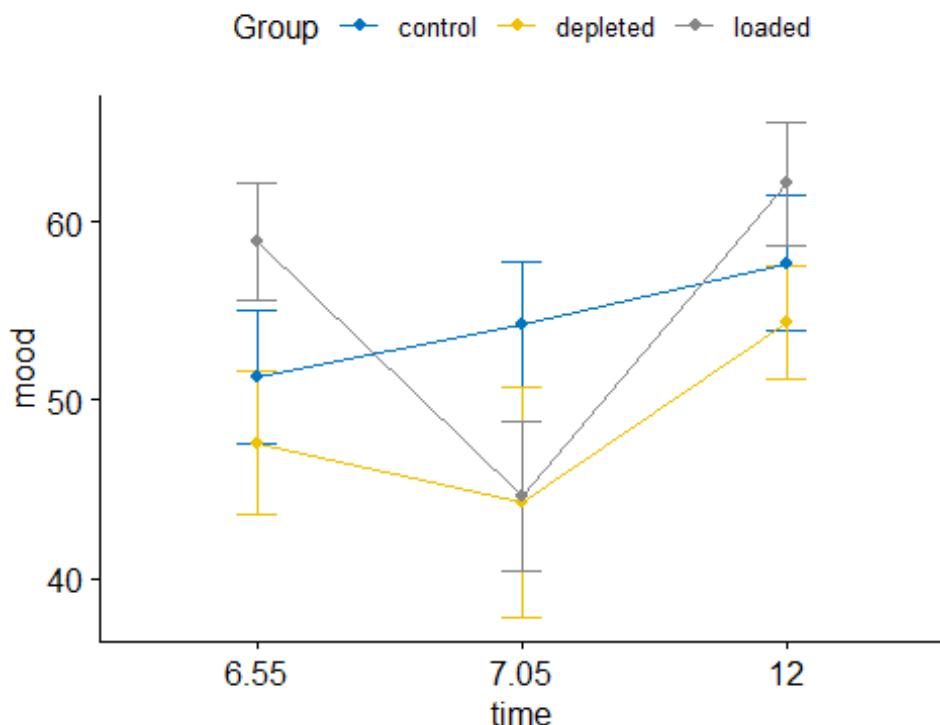
## package 'ggrepel' successfully unpacked and MD5 sums checked
## package 'ggsci' successfully unpacked and MD5 sums checked
## package 'cowplot' successfully unpacked and MD5 sums checked
## package 'ggsignif' successfully unpacked and MD5 sums checked
## package 'polynom' successfully unpacked and MD5 sums checked
## package 'ggpubr' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##   C:\Users\mie\AppData\Local\Temp\RtmpMNWdwg\downloaded_packages

##
## ggpubr installed

## Warning: package 'ggpubr' was built under R version 3.5.3

ggline(tryp_t_long, x = "time", y = "mood", col='Group',
       add = c("mean_se", "dodge"), palette = "jco")

```



```

library(lmerTest)

## Loading required package: lme4

## Warning: package 'lme4' was built under R version 3.5.2

## Loading required package: Matrix

```

```

## 
## Attaching package: 'Matrix'
## The following object is masked from 'package:tidyverse':
##   expand
##
## Attaching package: 'lme4'
## The following object is masked from 'package:nlme':
##   lmList
##
## Attaching package: 'lmerTest'
## The following object is masked from 'package:lme4':
##   lmer
## The following object is masked from 'package:stats':
##   step

#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))

```

```

#Show us what you've got
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

#Use Library(emmeans) to get more comprehensible pairwise interactions
#(uncorrected for multiple comparisons)
library(emmeans)
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  -17.239608 5.302074 80  -3.251  0.0017
## loaded - depleted    7.05 - 6.55  -10.959394 5.941356 80  -1.845  0.0688
## control - depleted    7.05 - 6.55   6.280214 5.791607 80   1.084  0.2815
## loaded - control    7.05 - 12     -14.127059 5.302074 80  -2.664  0.0093
## loaded - depleted    7.05 - 12     -7.470909 5.941356 80  -1.257  0.2123
## control - depleted    7.05 - 12     6.656150 5.791607 80   1.149  0.2539
## loaded - control    6.55 - 12      3.112549 5.302074 80   0.587  0.5588
## loaded - depleted    6.55 - 12      3.488485 5.941356 80   0.587  0.5588
## control - depleted    6.55 - 12      0.375936 5.791607 80   0.065  0.9484

```

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

**H1: Being depleted of tryptophan is hypothesised to lead to alterations of mood.**

**H2: mood is altered by forcing yourself to eat a nasty powder at 7.00**

**H3: mood is altered by becoming hungry at 12.00**

From the plot it can be seen that the mood improved in every group at 12. The mood was lowest in the depleted group and best in the loaded group.

There is a significant effect of time at 6.55 and 12 to mood. There are also significant interactions between control group and time 6.55, and control group and time 12.

From the pairwise comparisons, there is significant difference between tryptophan loaded and control group both between 6.55-7.05 and 7.05-12. All other pairwise comparisons are non-significant.

For the first hypothesis, there is no significant difference between mood in the tryptophan depleted and loaded group,  $t(80) = -1.28$ ,  $p = .21$ . Therefore, we cannot confirm the hypothesis.

For the second hypothesis, there was a significant difference between the control group and the tryptophan loaded group,  $t(80) = -3.25$ ,  $p = .0017$ . The difference between the control group and the depleted group did not prove significant,  $t(80) = 1.84$ ,  $p = .28$ . Whether or not tryptophan is ingested seem to affect the hypothesis.

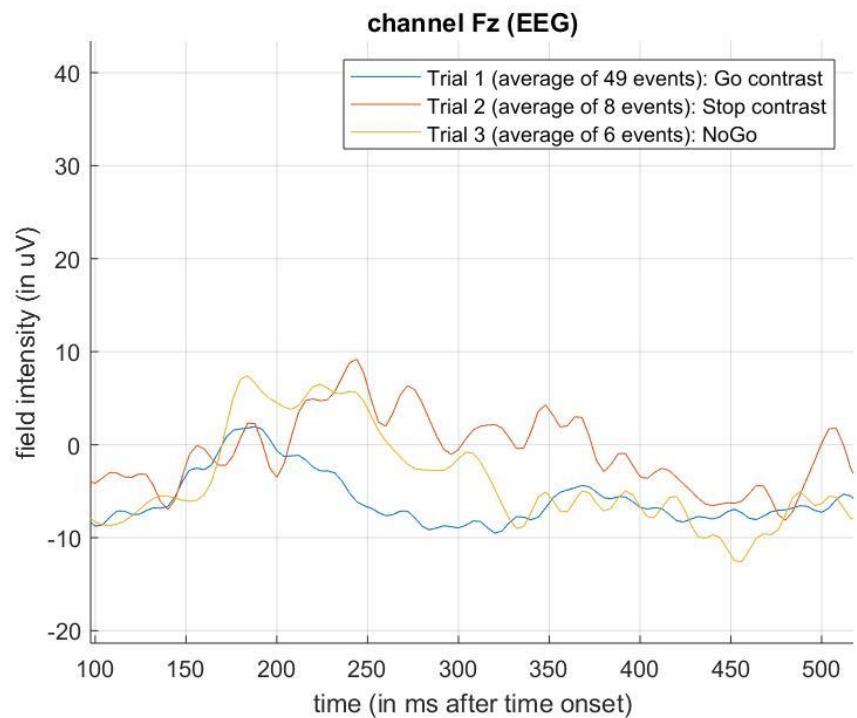
The last hypothesis can be rejected as no significant difference were found for the control group versus the loaded group,  $t(80) = 0.59$ ,  $p = .56$ .

# Portfolio 8: EEG Stop-Signal-NoGo

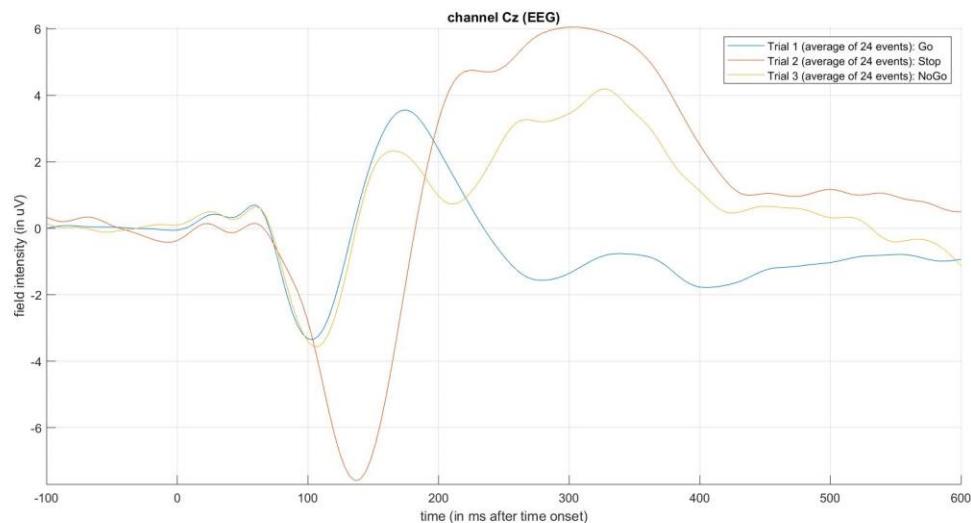
Group 3: Mie, Tobias, Riikka, Ruta and Jana

April 24, 2019

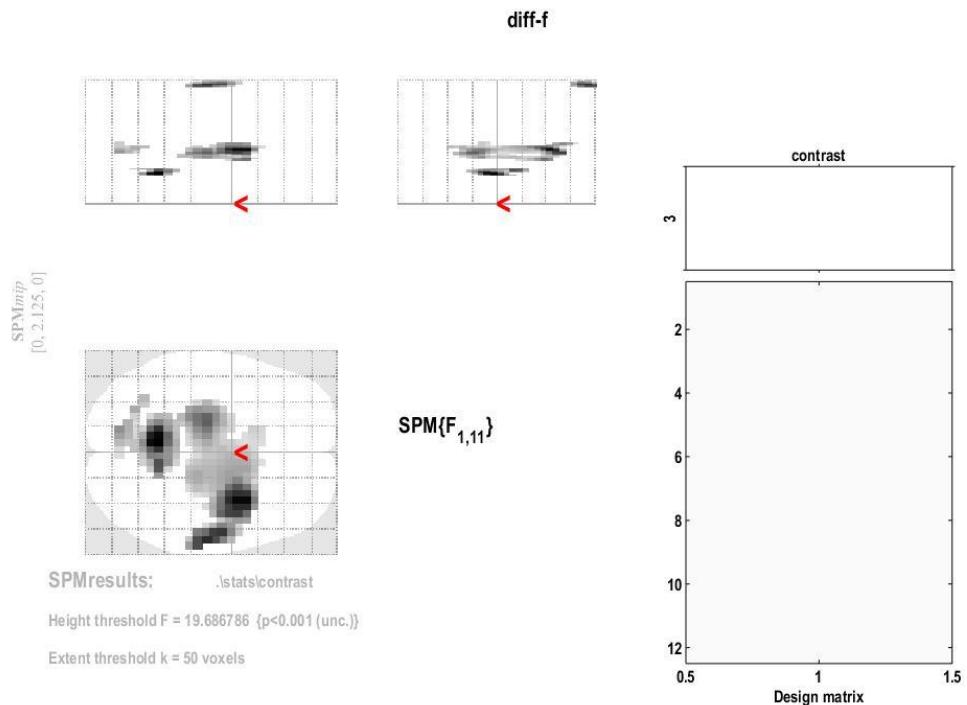
## 1. Plot of group 3 participant ERPs to the Go-, Stop- and NoGo-sounds from channel Fz.



## 2. Plot of the lowpass-filtered group-average ERP to the Go-, Stop- and NoGo-sounds from channel Cz.



### 3. Stats output coordinate table of the diff-f contrast



Statistics: *p*-values adjusted for search volume

set-level		cluster-level				peak-level								
<i>p</i>	<i>c</i>	<i>p</i> <sub>FWE-corr</sub>	<i>q</i> <sub>FDR-corr</sub>	<i>k</i> <sub>E</sub>	<i>p</i> <sub>uncorr</sub>	<i>p</i> <sub>FWE-corr</sub>	<i>q</i> <sub>FDR-corr</sub>	<i>F</i>	( <i>Z</i> )	<i>p</i> <sub>uncorr</sub>	mm	mm	ms	
0.001	4	0.039	0.078	156	0.011	0.221	0.855	51.17	4.12	0.000	-4	-52	88	
						0.437	0.855	39.07	3.84	0.000	9	-52	100	
						0.700	0.855	30.19	3.56	0.000	-4	-62	104	
		0.000	0.000	778	0.000	0.244	0.855	49.37	4.09	0.000	34	2	168	
						0.385	0.855	41.33	3.90	0.000	43	2	156	
						0.490	0.855	37.04	3.78	0.000	-17	-19	156	
		0.174	0.250	82	0.054	0.324	0.855	44.37	3.97	0.000	60	-19	384	
						0.355	0.855	42.76	3.93	0.000	55	-9	384	
		0.236	0.265	68	0.076	0.686	0.855	30.63	3.57	0.000	-13	-79	164	
						0.803	0.855	27.02	3.44	0.000	-26	-68	172	
						0.866	0.855	24.92	3.35	0.000	-34	-62	176	

table shows 3 local maxima more than 8.0mm apart

Height threshold: *F* = 19.69, *p* = 0.001 (0.972)

Degrees of freedom = [1.0, 11.0]

Extent threshold: *k* = 50 voxels, *p* = 0.122 (0.353)

FWHM = 37.0 43.7 28.8 mm mm ms; 8.7 8.1 7.2 {voxels}

Expected voxels per cluster, <*k*> = 21.790

Volume: 5777276 = 63226 voxels = 112.0 resels

Expected number of clusters, <*c*> = 0.43

Voxel size: 4.3 5.4 4.0 mm mm ms; (resel = 509.64 voxels)

FWEp: 82.261, FDRp: Inf, FWEc: 156, FDRc: 778

## Portfolio 9: fMRI group analysis

Group 3: Ruta, Riikka, Jana, Tobias and Mie

### 1.a) Order of conditions

The order of the conditions is blue\_neutral, blue\_fear, yellow\_neutral, yellow\_fear and incorrect.

### 1.b) What will the seven contrasts provided in the contrast specification test for?

C1: The first contrast is [1, 1, 1, 1] which match the first hypothesis (H1) that there will be activation in the occipital face area (OFA) and in the fusiform face area (FFA) across all conditions. This is very intuitive since all stimuli presents faces.

C2: The second contrast is [1, 1, -1, -1] which compares the blue face trials with the yellow face trials by removing the effects of the yellow face stimuli. This match the third hypothesis (H3) which state that we will have a greater BOLD response for the middle finger (yellow trials) than for the index finger (blue trials) in the motor cortex. This difference is thought to be due to the hand being more adapted to use the index finger.

C3: The third contrast is [-1, 1, -1, 1]. This compares the neutral faces to the fearful faces and according to the second hypothesis (H2) the fearful faces will elicit a greater response in the emotional regions (like the amygdala) than the neutral faces.

C4: The fourth contrast is [-1, 1, 1, -1]. This looks for the interaction effect between the blue and yellow faces and the faces being either neutral or fearful. It does this by combining the effects of the second and third contrast.

$$C4 = C2 \times C3 = [1, 1, -1, -1] \times [-1, 1, -1, 1] = [1 * -1, 1 * 1, -1 * -1, -1 * 1] = [-1, 1, 1, -1]$$

This new contrast then compares the blue\_fear and the yellow\_neutral conditions to the blue\_neutral and yellow\_fear conditions.

C5: The fifth contrast is [-1, -1, 1, 1] which compares the yellow face trials with the blue face trials in the opposite order of the second contrast. Here the activation elicited by the blue faces are removed leaving only the effects of the yellow faces. Like the second contrast, this match the third hypothesis.

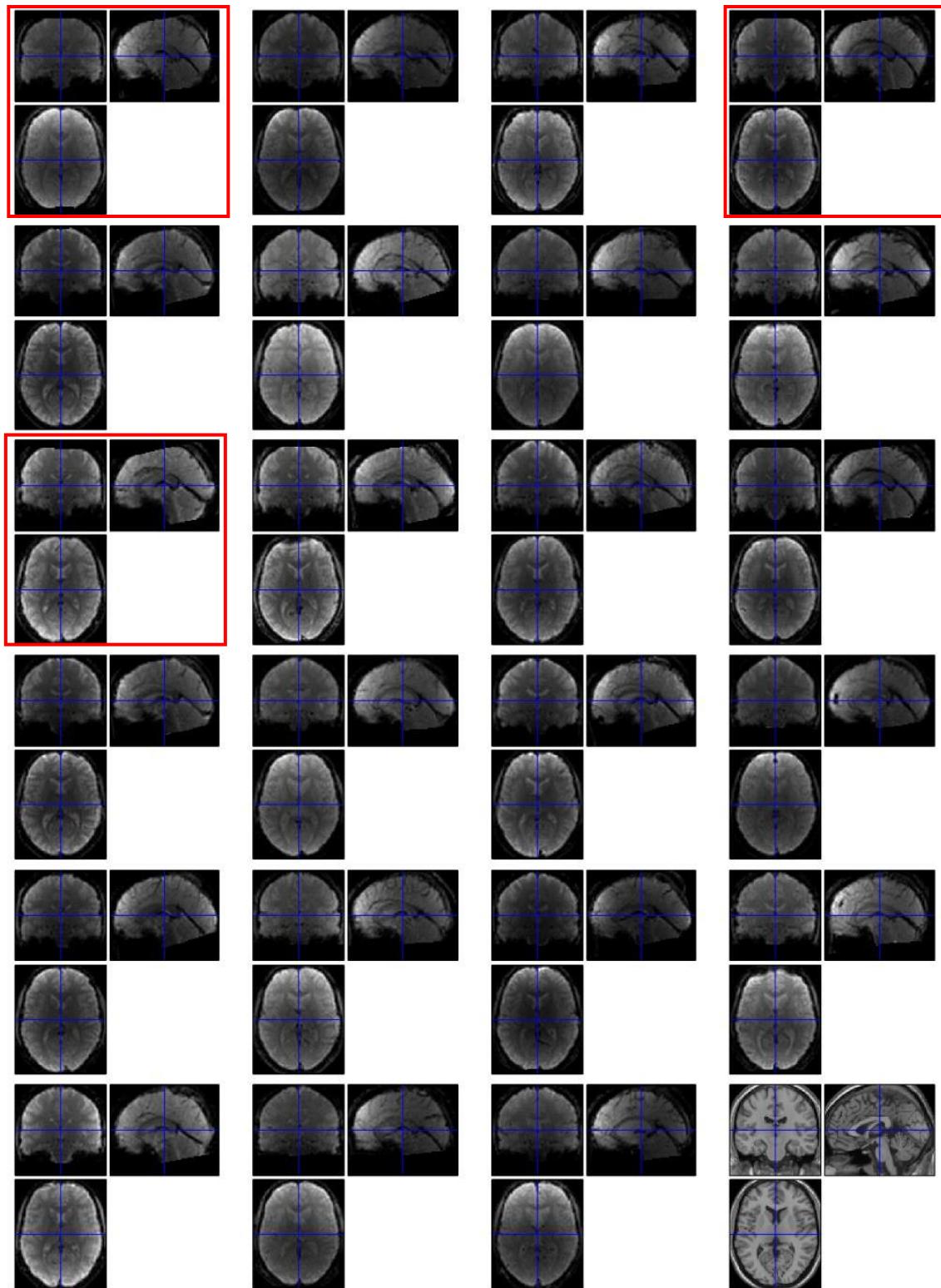
C6: The sixth contrast is [1, -1, -1, 1] and like the fourth contrast this looks at the interaction effect. However, this contrast compares the blue\_neutral and the yellow\_fear conditions with the blue\_fear and the yellow\_neutral conditions.

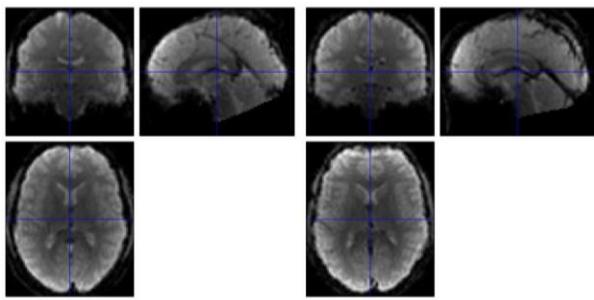
C7: The seventh contrast is [1, -1, 1, -1] which is the opposite of the third contrast. This is testing that the activation for the fearful faces is smaller than the activation for the neutral faces.

**1.c) Report the collected time it took to analyze all participants.**

It took approximately 8 hours (20 minutes per participant) to process all the data.

**1.d. Checking preprocessing: What are the pros and cons of keeping the participant in the dataset where some of the brain is not in the scans, given the hypotheses? What would you choose?**





The images that are highlighted by a red square have cut off the top of the brain. The participants in question are number 2, 5 and 10. Given the hypotheses, we are looking for activation in the occipital face area, the fusiform face area and the regions associated with emotions (like the amygdala). All of these will still be present in all the scans since they are located in the temporal lobes. We are also interested in the areas in the motor cortex that represent the middle and index fingers, and these are located in the frontal cortex, which might be cut off in the images.

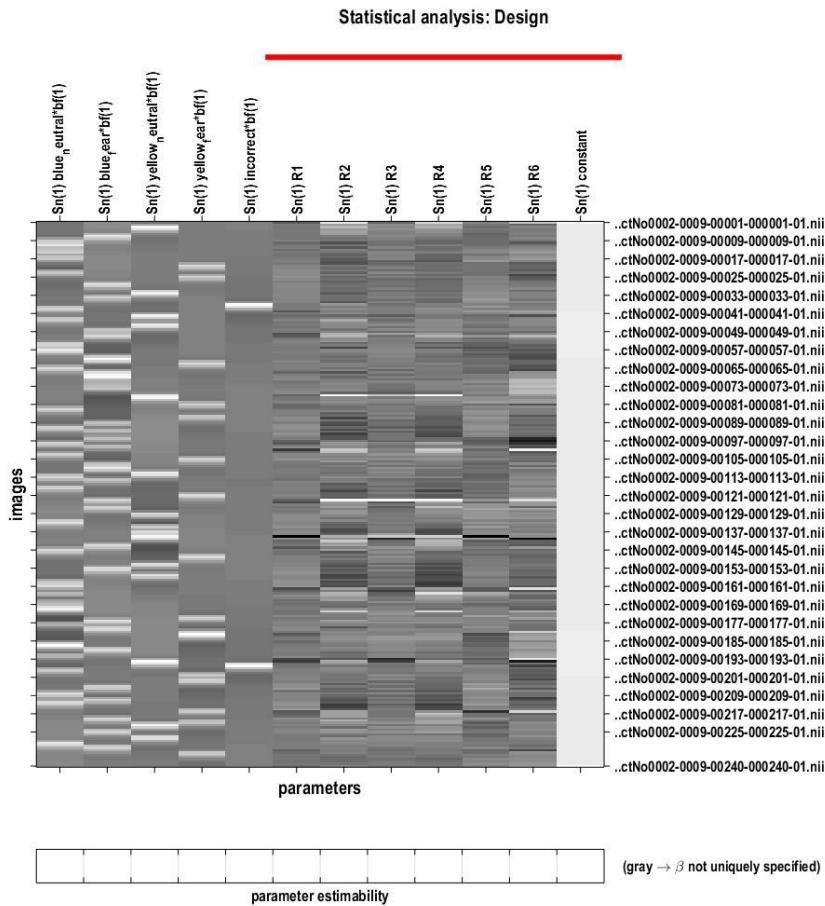
The pros of keeping the images is that we have more data so our results would have more statistical power.

The cons of keeping the participants would be that the comparisons of the activation in the motor cortex (which is relevant for H3 and H4) could be compromised.

Since we cannot be sure that the motor cortex is cut off, we decide to prioritize statistical power over precision by not removing any data.

## 2.a Investigate the design matrix

We chose the SPM.mat file of participant 2.



**Design description...**

```

Basis functions : hrf
Number of sessions : 1
Trials per session : 5
Interscan interval : 2.00 (s)
High pass Filter : [min] Cutoff: 128 (s)
Global calculation : mean voxel value
Grand mean scaling : session specific
Global normalisation : None

```

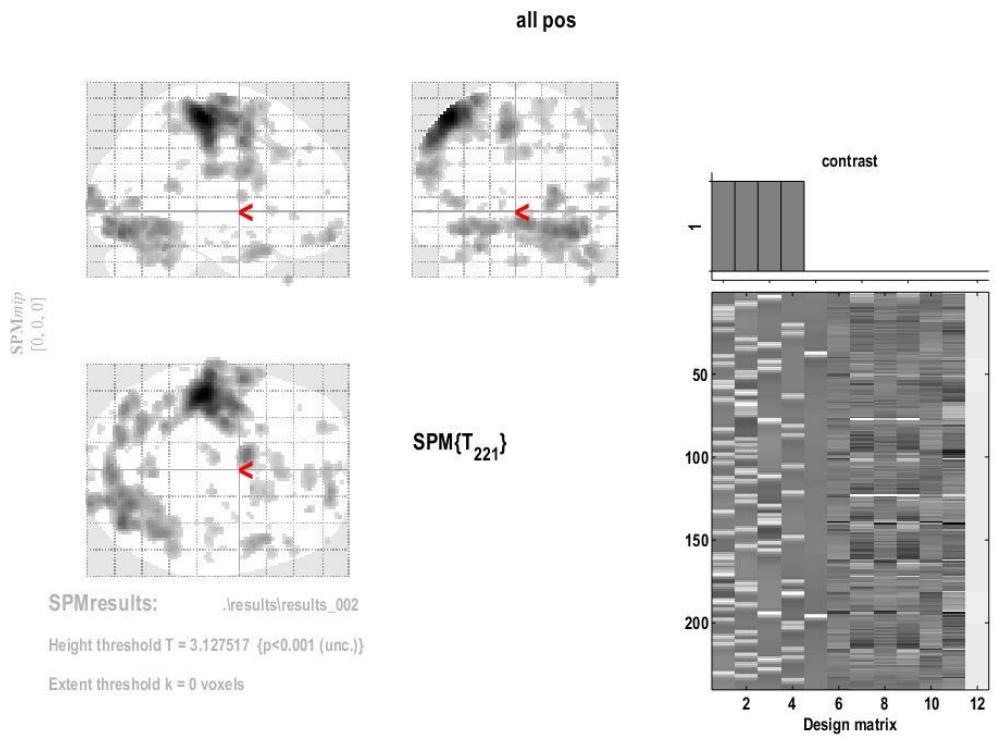
From the design matrix, we can see that participant 2 was shown blue faces more frequent than yellow faces.

Looking at the fifth column in the matrix, we can tell that the participant had two incorrect answers.

It is a good idea to discard or separately model incorrect responses, since it could be an indication that the participant lost interest or stopped paying attention. We can therefore not be sure that the activation is due to the experimental manipulation.

## **2.b Investigate the results**

Again, we choose participant 2 (though we know we argued in 1.d that we would remove the data).



Statistics: *p*-values adjusted for search volume

set-level		cluster-level				peak-level					mm mm mm		
$p$	$c$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$T$	$(Z_{\text{c}})$	$p_{\text{uncorr}}$	$-46$	$-24$	$56$
0.000	59	0.000	0.000	3031	0.000	0.000	0.000	7.44	7.02	0.000	-36	-20	60
						0.000	0.001	6.14	5.90	0.000	-56	-22	44
		0.000	0.000	2526	0.000	0.005	0.006	5.46	5.28	0.000	36	-80	-16
						0.016	0.013	5.20	5.05	0.000	24	-74	-14
		0.001	0.001	382	0.000	0.045	0.030	4.95	4.81	0.000	6	-86	-10
						0.025	0.019	5.09	4.95	0.000	-8	4	50
						0.387	0.110	4.32	4.23	0.000	-8	6	60
						1.000	0.790	3.33	3.28	0.001	-8	14	42
		0.049	0.030	159	0.003	0.097	0.049	4.75	4.63	0.000	24	-100	12
						0.273	0.091	4.44	4.34	0.000	22	-94	2
		0.630	0.250	54	0.059	0.240	0.083	4.49	4.38	0.000	-26	-50	-24
		0.005	0.004	266	0.000	0.320	0.100	4.39	4.29	0.000	16	58	-18
						0.792	0.242	3.98	3.90	0.000	2	56	-20
						0.954	0.375	3.76	3.70	0.000	8	46	-20
		0.032	0.023	177	0.002	0.339	0.101	4.37	4.28	0.000	46	14	-26
						0.489	0.138	4.23	4.15	0.000	48	6	-30
						1.000	0.943	3.19	3.15	0.001	58	0	-26

table shows 3 local maxima more than 8.0mm apart

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

Degrees of freedom = [1.0, 221.0]

Extent threshold:  $k = 0$  voxels

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

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

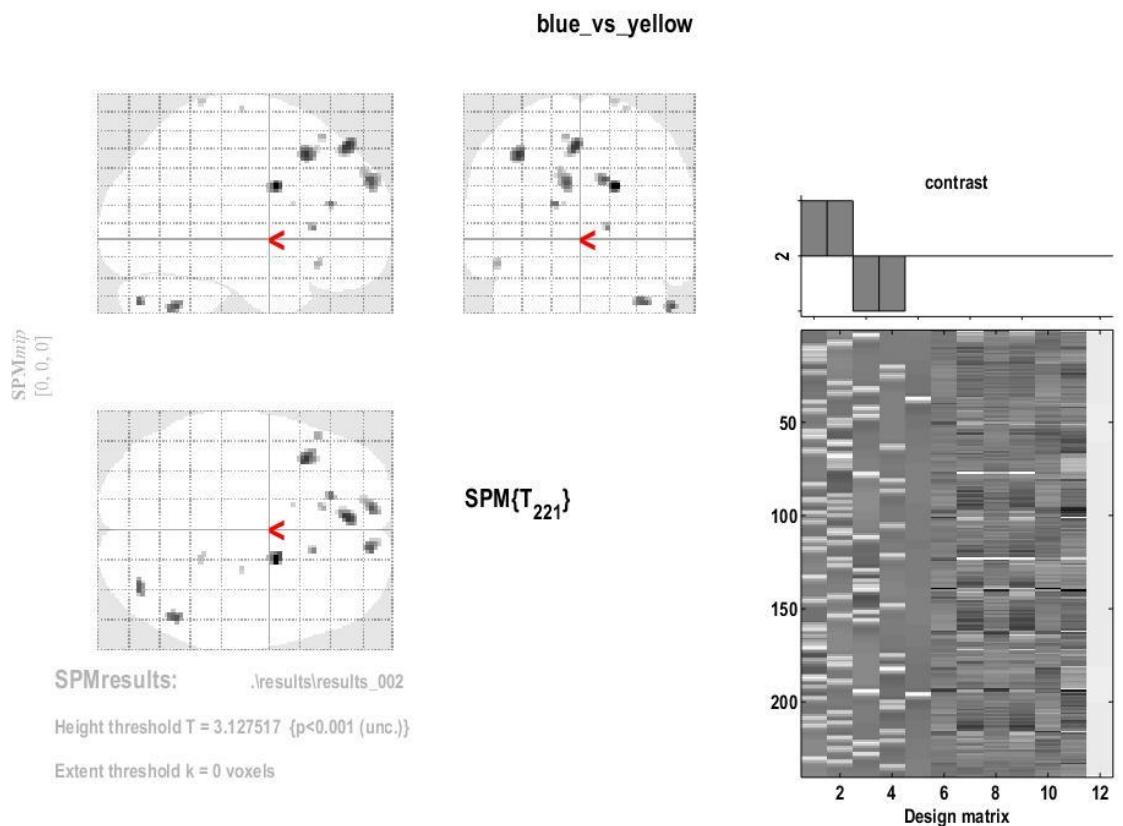
Volume: 1781184 = 222648 voxels = 1560.6 resels

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

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

FWEp: 4.922, FDRp: 4.731, FWEc: 159, FDRc: 144

Page 1



#### Statistics: p-values adjusted for search volume

set-level	cluster-level					peak-level					mm mm mm		
	p	c	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	$(Z_{\geq})$	$p_{\text{uncorr}}$		
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,  $\langle c \rangle = 16.73$

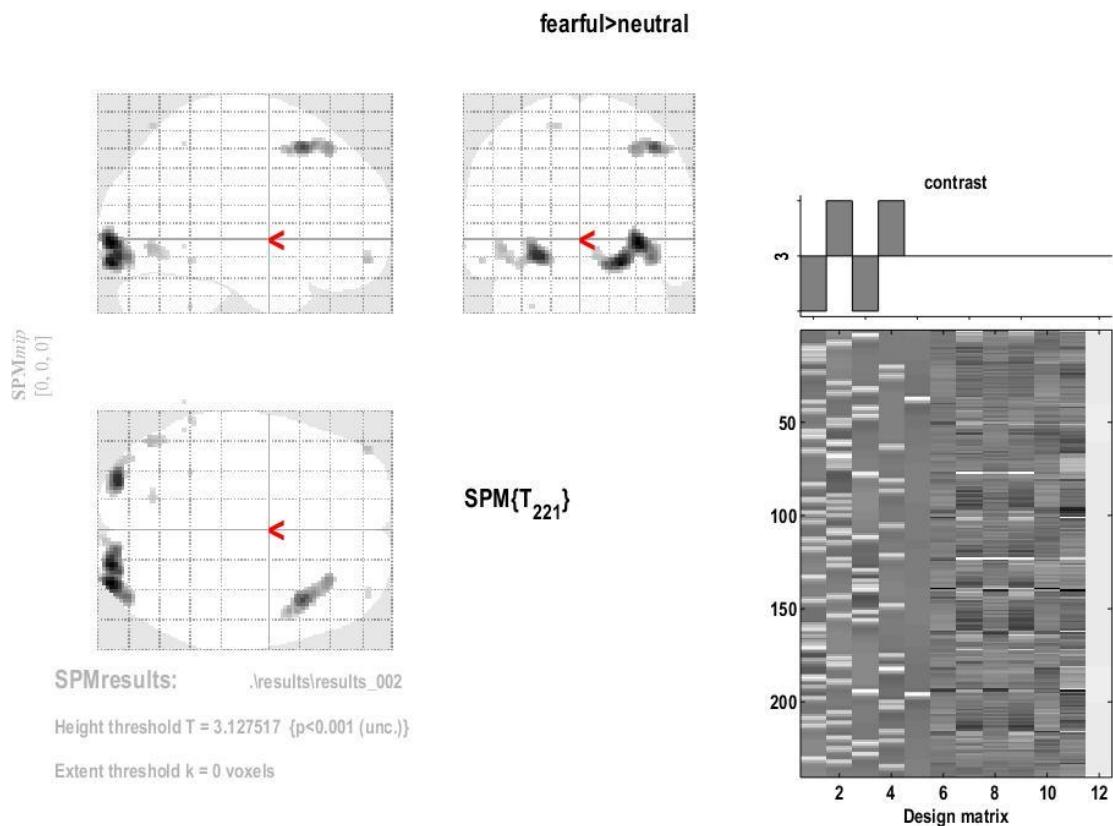
FWEp: 4.922, FDRp: Inf, FWEc: Inf, FDRc: Inf

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)



#### 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	$(Z_{\geq})$	$p_{uncorr}$		
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,  $\langle c \rangle = 16.73$

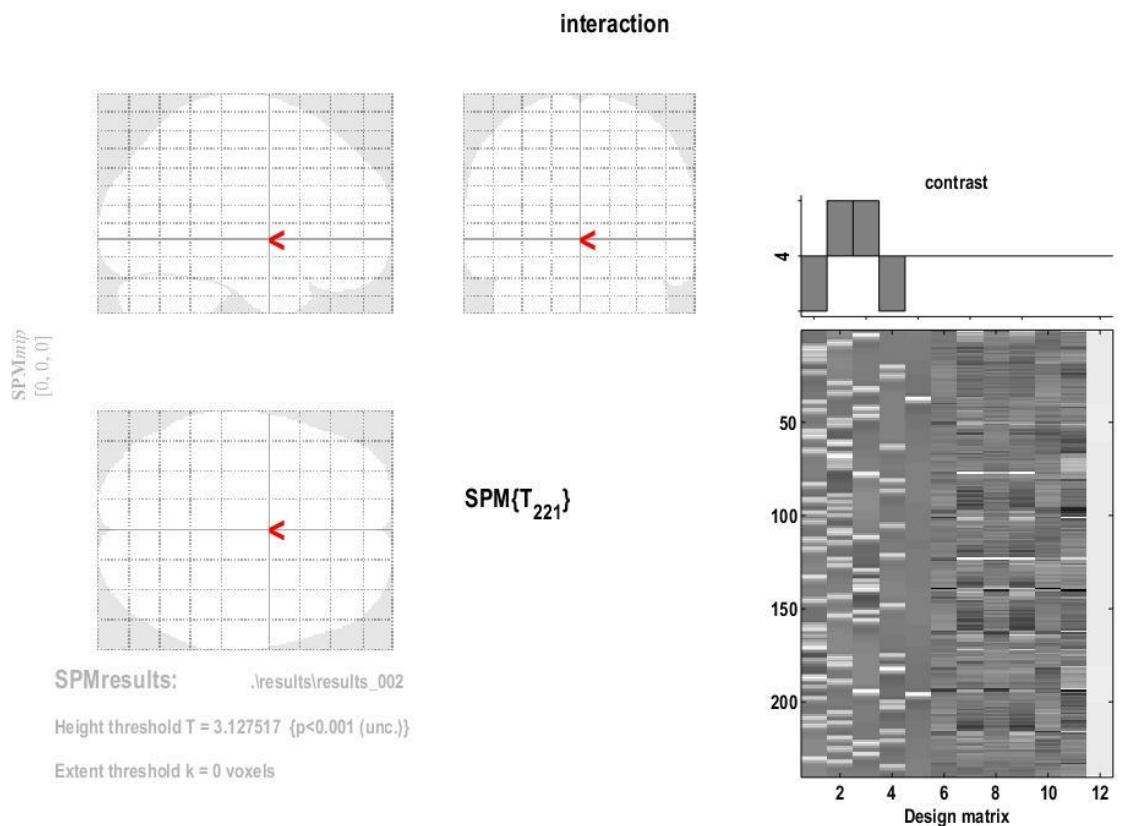
FWEp: 4.922, FDRp: 4.939, FWEc: 180, FDRc: 180

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			
<i>p</i>	<i>c</i>	<i>p<sub>FWE-corr</sub></i>	<i>q<sub>FDR-corr</sub></i>	<i>k<sub>E</sub></i>	<i>p<sub>uncorr</sub></i>	<i>p<sub>FWE-corr</sub></i>	<i>q<sub>FDR-corr</sub></i>	<i>T</i>	( <i>Z<sub>=</sub></i> )	<i>p<sub>uncorr</sub></i>			

*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, <k> = 15.128

Expected number of clusters, <c> = 16.73

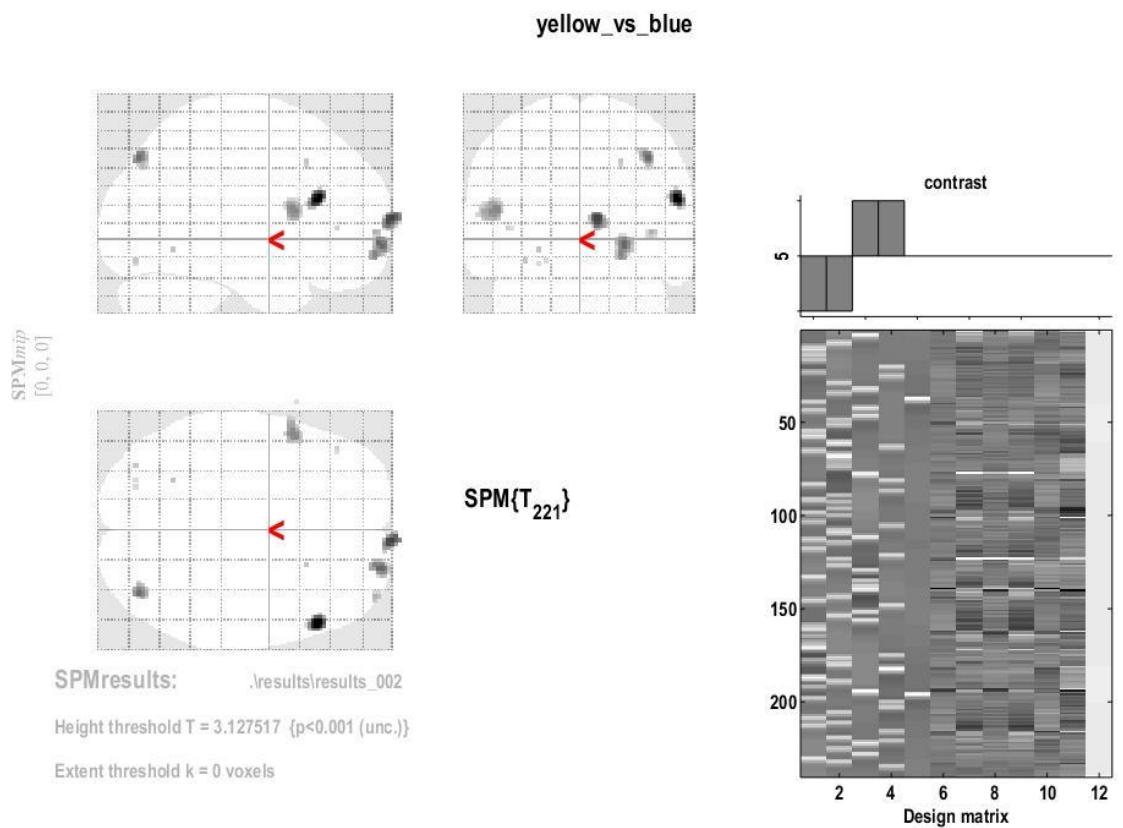
FWEp: 4.922, FDRp: Inf, FWEc: Inf, FDRC: Inf

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 <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	k <sub>E</sub>	p <sub>uncorr</sub>	p <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	T	(Z <sub>+</sub> )	p <sub>uncorr</sub>		
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, <k> = 15.128

Expected number of clusters, <c> = 16.73

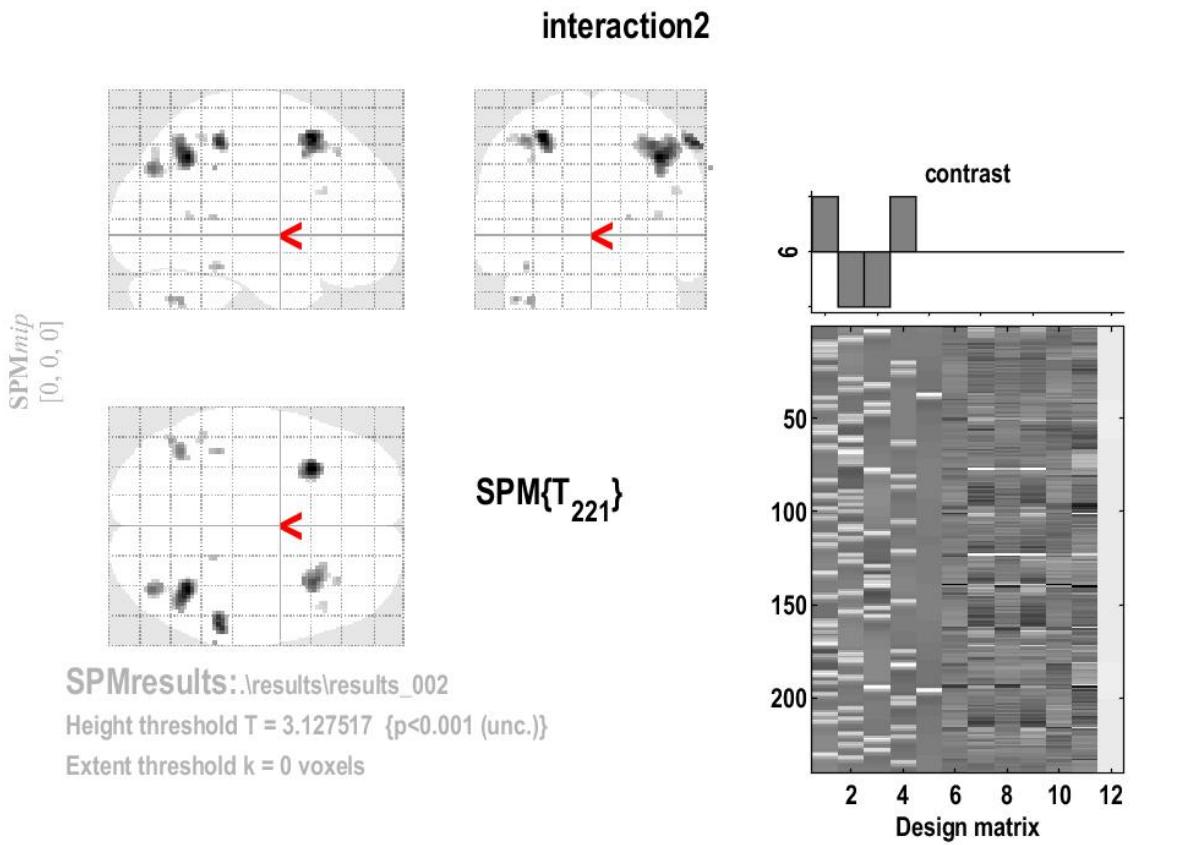
FWEp: 4.922, FDRp: Inf, FWEc: Inf, FDRc: Inf

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	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{FWE\text{-corr}}$	$q_{FDR\text{-corr}}$	T	$(Z_{\equiv})$	$p_{\text{uncorr}}$	mm	mm	mm
0.506	17	0.014	0.014	216	0.001	0.433	0.432	4.28	4.19	0.000	38	-56	42
						1.000	0.825	3.43	3.38	0.000	50	-60	46
		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,  $\langle c \rangle = 16.73$

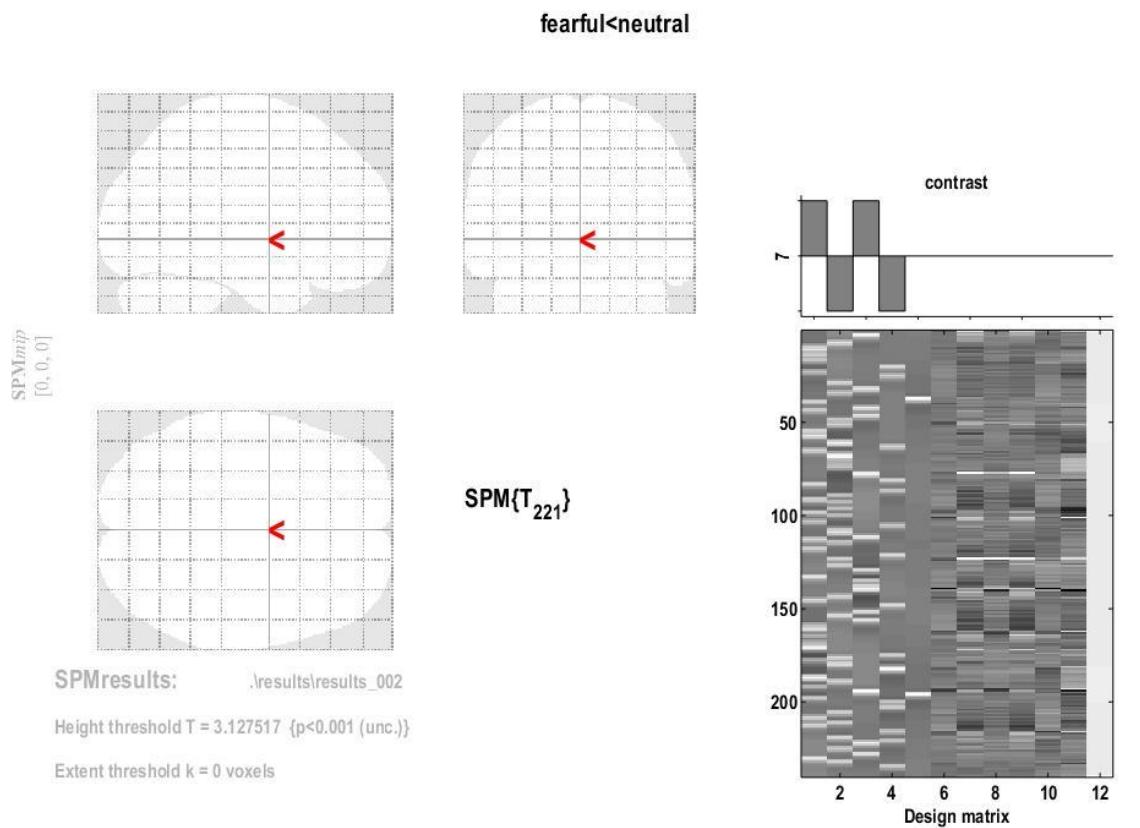
FWEp: 4.922, FDRp: Inf, FWEc: 216, FDRc: 129

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)



#### Statistics: *p*-values adjusted for search volume

set-level		cluster-level			peak-level					mm mm mm			
<i>p</i>	<i>c</i>	<i>p<sub>FWE-corr</sub></i>	<i>q<sub>FDR-corr</sub></i>	<i>k<sub>E</sub></i>	<i>p<sub>uncorr</sub></i>	<i>p<sub>FWE-corr</sub></i>	<i>q<sub>FDR-corr</sub></i>	<i>T</i>	( <i>Z<sub>≥</sub></i> )	<i>p<sub>uncorr</sub></i>			

*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, <k> = 15.128

Expected number of clusters, <c> = 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)

Briefly explain if you find signs to support the hypotheses.

H1: For first contrast, the findings seem to support the hypothesis.

The coordinates corresponding to the right fusiform face area (found in the article, though slightly changed to not point to the cerebellum: 40, -52, -20) seems to be part of a significantly activated cluster. For the left fusiform face area (-40, -48, -20) seems to be just outside of a cluster of activation. This also corresponds with the literature where the right fusiform face area is related to seeing faces of strangers. On the contrary, the left fusiform face area is related to seeing your own face. Since the participant is only observing faces of others it seems reasonable that the activation is only evident in the right side.

For the right occipital face area, the coordinates given in the article point to there not being activation in the first contrast. However, other articles point to the x-coordinate being  $35 \pm 5$  and substituting with this suggest that there is activation in the right occipital face area (Zhang, J. et al, 2012<sup>1</sup>).

For the left occipital face area, the same can be observed having the coordinates from the given article point to an area without activation, whereas the second article gives coordinates that point to a cluster of activation.

H2: This time looking only at the coordinates from the second article, fearful faces show a greater activation in the occipital face area than neutral faces, where the right side showed greater activation than the left side.

Since the contrast of neutral faces being greater than fearful faces show no significant activation throughout the brain, it would seem that the fearful faces elicit a higher BOLD response than neutral faces.

Activation in the amygdala as well as in the fusiform face area was not detected in either contrasts.

H3: The activation in the motor cortex seems to be the same for both responses. This is evident in that we couldn't detect activation in the motor cortex in either contrast.

H4: Participant 2 had blue faces shown more frequently than yellow faces and so the contrast yellow>blue should show more activation in the visual areas and in the motor areas than the

---

<sup>1</sup> <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0040390>

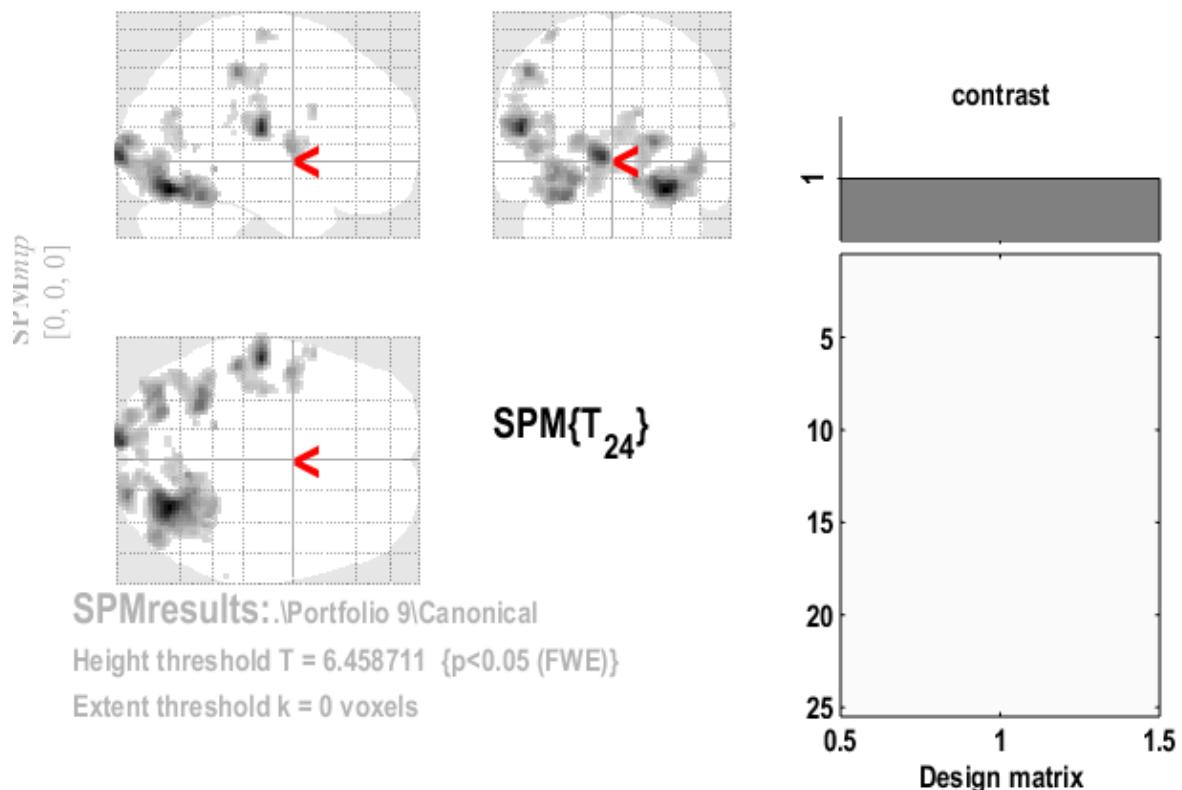
contrast blue>yellow. However, this is not the case therefore suggesting that the frequency does not affect the activation.

### **3.a All positive condition**

We used a T-contrast displaying all positive effects by contrasting all activation with no activation.

FWE-corrected image:

## All positive contrast



### Statistics: $p$ -values adjusted for search volume

set-level	cluster-level				peak-level				mm mm mm					
	$p$	$c$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$T$	$(Z_{\equiv})$	$p_{\text{uncorr}}$	$\text{mm}$	$\text{mm}$	$\text{mm}$
0.000 14	0.000	0.000	1645	0.000	0.000	0.000	15.47	7.52	0.000	28	-72	-18		
					0.000	0.006	10.70	6.43	0.000	22	-62	-16		
					0.000	0.006	10.58	6.39	0.000	22	-96	8		
	0.000	0.000	579	0.000	0.000	0.000	13.88	7.20	0.000	-54	-20	16		
					0.000	0.007	10.36	6.33	0.000	-50	-34	48		
					0.004	0.128	7.86	5.48	0.000	-38	-28	22		
	0.000	0.000	1211	0.000	0.000	0.000	12.96	7.00	0.000	-8	-100	0		
					0.000	0.005	10.94	6.50	0.000	-30	-54	-22		
					0.000	0.010	10.02	6.23	0.000	-26	-76	-12		
	0.000	0.000	139	0.000	0.000	0.028	9.10	5.93	0.000	-46	-4	6		
					0.010	0.257	7.32	5.26	0.000	-58	6	8		
	0.007	0.171	4	0.134	0.001	0.062	8.43	5.69	0.000	-16	-46	-6		
	0.000	0.002	28	0.001	0.002	0.096	8.16	5.59	0.000	-36	-18	68		
	0.000	0.000	162	0.000	0.002	0.111	8.05	5.55	0.000	12	-70	24		
					0.006	0.165	7.60	5.38	0.000	2	-66	14		
					0.017	0.388	7.03	5.13	0.000	12	-70	6		
	0.000	0.007	19	0.003	0.005	0.145	7.71	5.42	0.000	-56	10	26		
0.001	0.030	11	0.019	0.014	0.346	7.13	5.18	0.000	-52	-36	28			

table shows 3 local maxima more than 8.0mm apart

Height threshold:  $T = 6.46$ ,  $p = 0.000$  (0.050)

Degrees of freedom = [1.0, 24.0]

Extent threshold:  $k = 0$  voxels

FWHM = 10.5 10.5 10.3 mm mm mm; 5.3 5.2 5.1 {voxels}

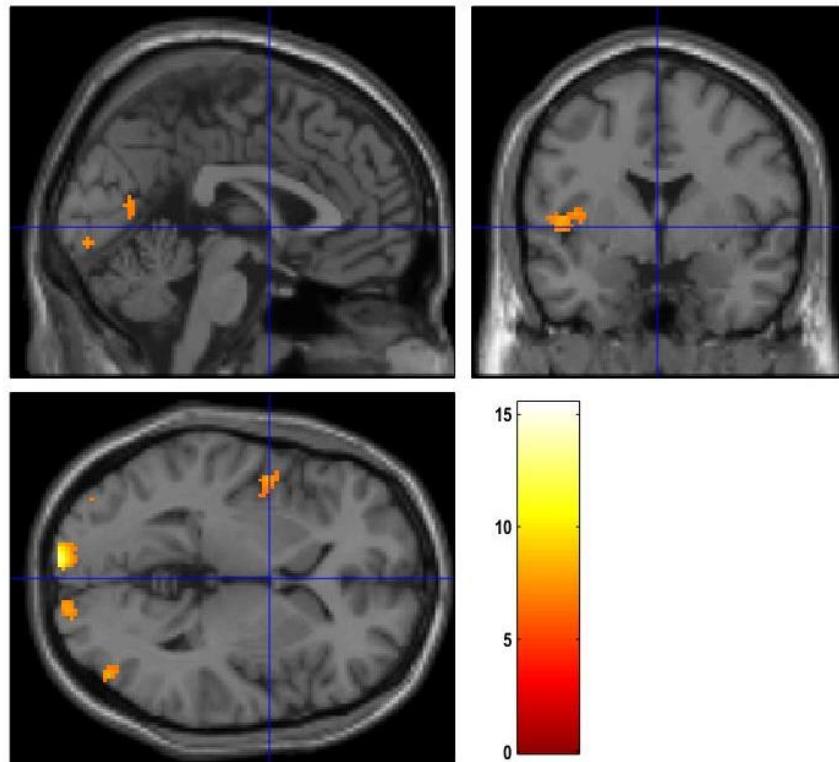
Expected voxels per cluster,  $\langle k \rangle = 1.871$

Volume: 1341864 = 167733 voxels = 1074.1 resels

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

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 142.45 voxels)

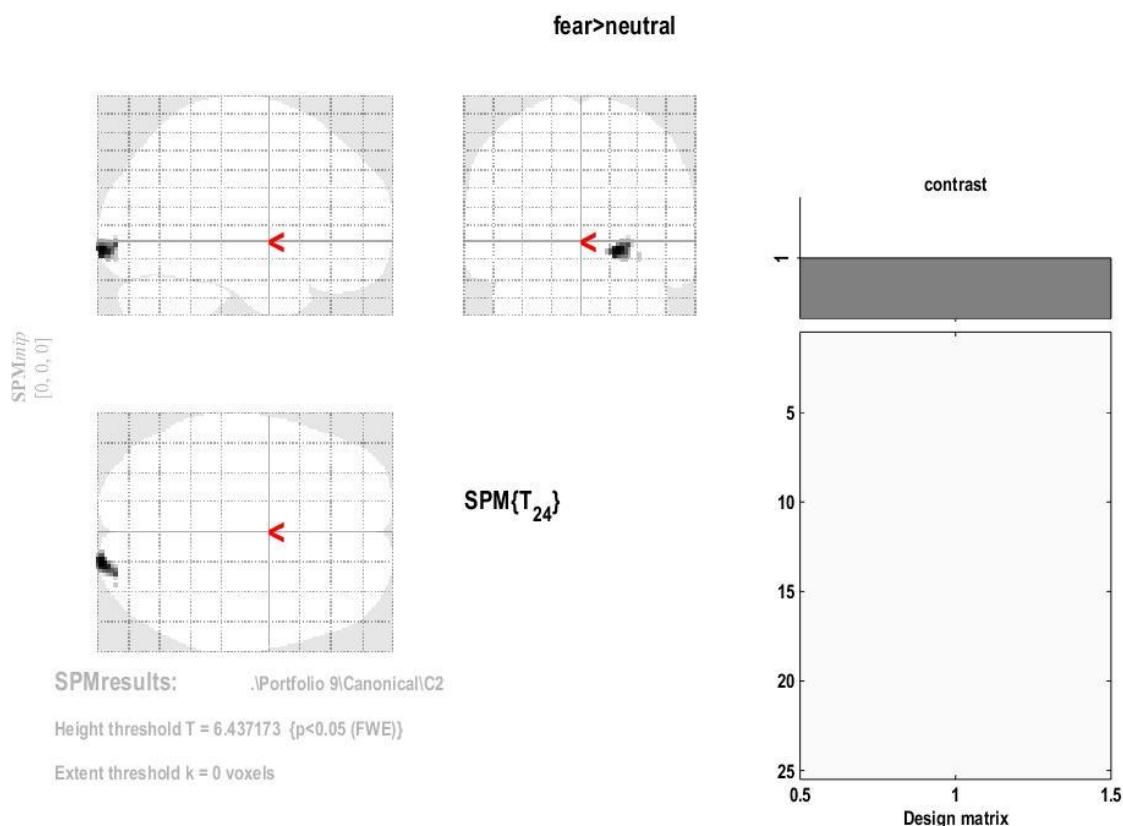
Overlaid image:



The images show that there seem to be significant activation in the occipital face area and the fusiform face area as well as in the left motor cortex across all conditions. The activation in the left motor cortex suggest that the right hand was used for responding and was also used across conditions.

### 3.b Effect of emotion

Results of emotional effect using p-value < 0.05 FWE-correction:



#### Statistics: p-values adjusted for search volume

set-level	c	cluster-level				peak-level				mm mm mm			
		p <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	k <sub>E</sub>	p <sub>uncorr</sub>	p <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	T	(Z <sub>≥</sub> )	p <sub>uncorr</sub>	20	-100	-6
0.001	2	0.000	0.000	84	0.000	0.002	0.105	8.20	5.61	0.000	26	-92	-4
		0.015	0.295	2	0.295	0.006	0.171	7.56	5.36	0.000	32	-92	-10
						0.035	0.702	6.62	4.95	0.000			

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, <k> = 1.974

Expected number of clusters, <c> = 0.05

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

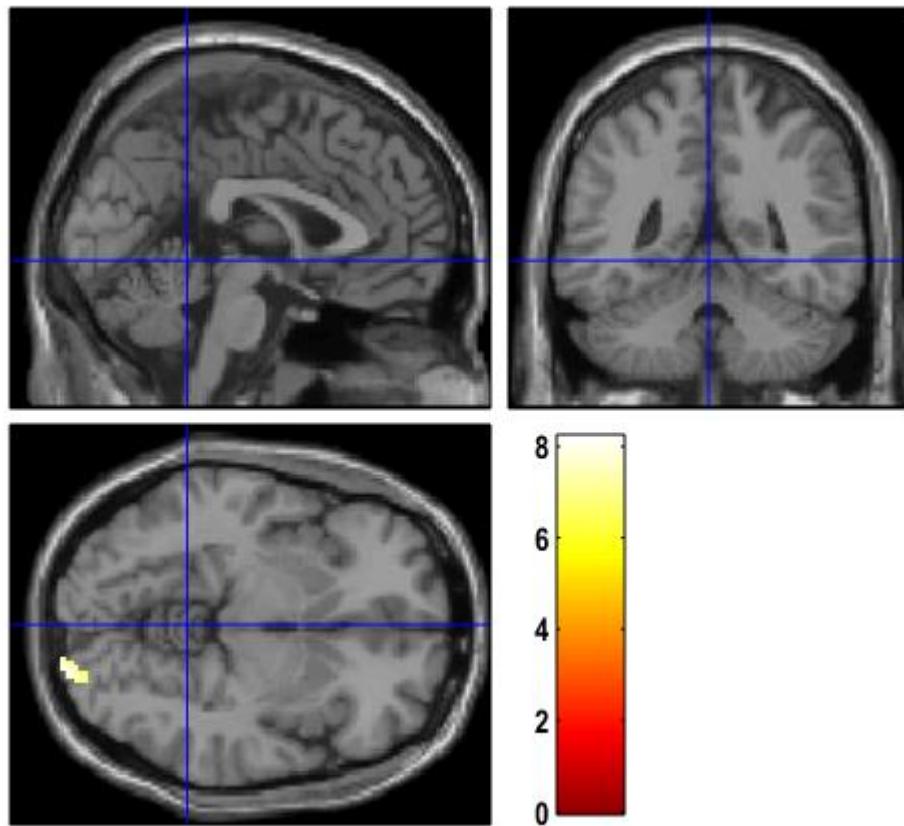
Degrees of freedom = [1.0, 24.0]

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)

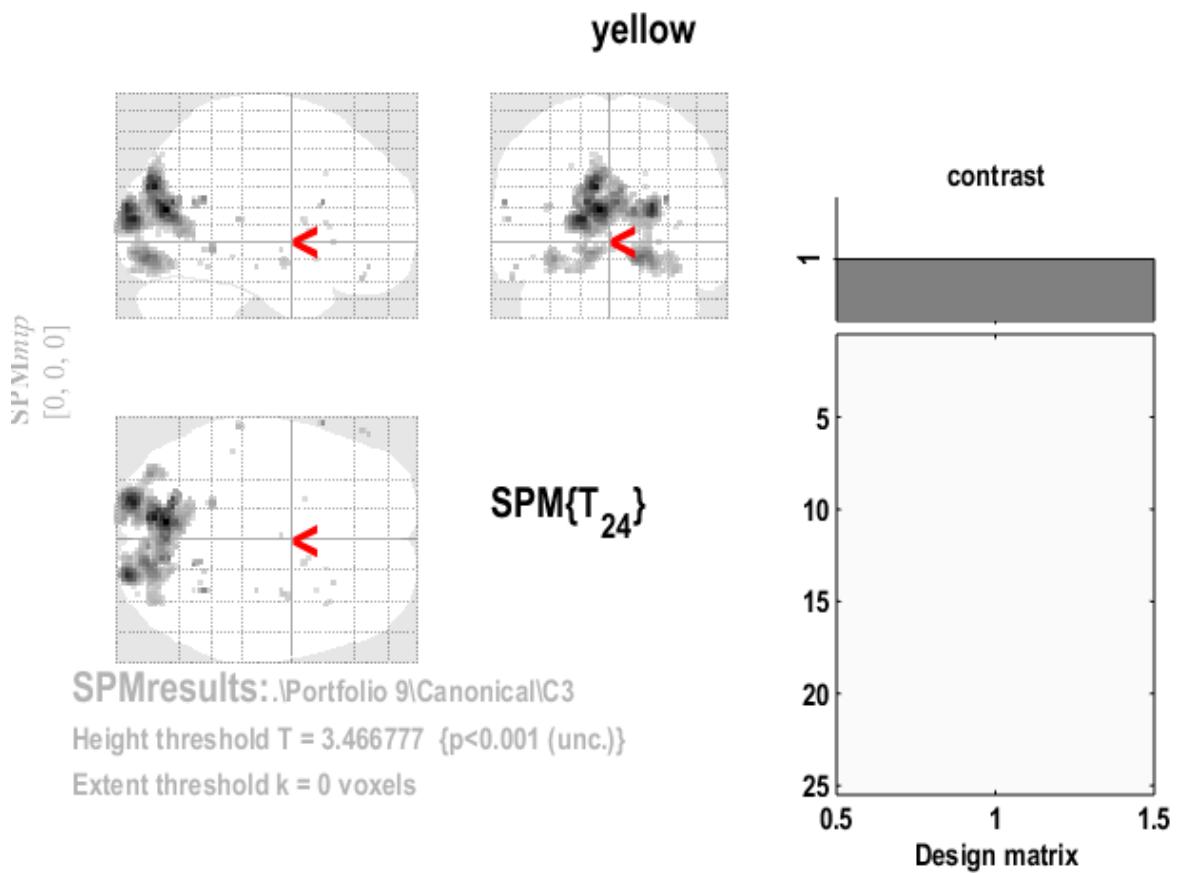
Overlaid image:



The fearful faces did elicit a greater response than the neutral faces but only in what seems to be the right occipital face area.

### 3.c effect of color

Using an uncorrected p-value with  $P<0.001$ :



#### Statistics: *p*-values adjusted for search volume

set-level		cluster-level				peak-level				mm mm mm			
<i>p</i>	<i>c</i>	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$k_E$	$p_{\text{uncorr}}$	$p_{\text{FWE-corr}}$	$q_{\text{FDR-corr}}$	$T$	$(Z_{\equiv})$	$p_{\text{uncorr}}$			
0.000	34	0.000	0.000	2158	0.000	0.110	0.719	5.83	4.56	0.000	-8	-74	16
						0.200	0.719	5.50	4.38	0.000	-10	-80	28
						0.232	0.719	5.42	4.34	0.000	-18	-92	10
		0.999	0.844	3	0.703	0.873	0.883	4.37	3.71	0.000	30	-52	22
		0.021	0.035	221	0.002	0.876	0.883	4.36	3.71	0.000	-12	-90	-8
						0.906	0.883	4.29	3.66	0.000	-34	-82	-12
						0.997	0.929	3.81	3.34	0.000	-20	-88	-18
		1.000	0.844	2	0.764	0.918	0.883	4.26	3.64	0.000	-62	16	20
		0.995	0.844	8	0.508	0.947	0.929	4.18	3.59	0.000	16	-58	20
		0.989	0.844	11	0.433	0.974	0.929	4.06	3.51	0.000	-18	-48	-6
		0.999	0.844	3	0.703	0.995	0.929	3.86	3.37	0.000	-60	-32	10
		1.000	0.844	2	0.764	0.996	0.929	3.84	3.36	0.000	-2	-64	2
		1.000	0.844	2	0.764	0.997	0.929	3.79	3.32	0.000	-58	38	-22
		0.998	0.844	5	0.610	0.998	0.929	3.76	3.30	0.000	-36	-82	18
		1.000	0.844	1	0.844	0.998	0.929	3.76	3.30	0.000	-56	42	-16
		0.996	0.844	7	0.538	0.998	0.929	3.75	3.29	0.000	32	-6	-14
		1.000	0.844	1	0.844	0.999	0.941	3.67	3.23	0.001	20	-52	-6
		1.000	0.844	1	0.844	0.999	0.941	3.66	3.23	0.001	30	-22	-16

table shows 3 local maxima more than 8.0mm apart

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

Degrees of freedom = [1.0, 24.0]

Extent threshold:  $k = 0$  voxels

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

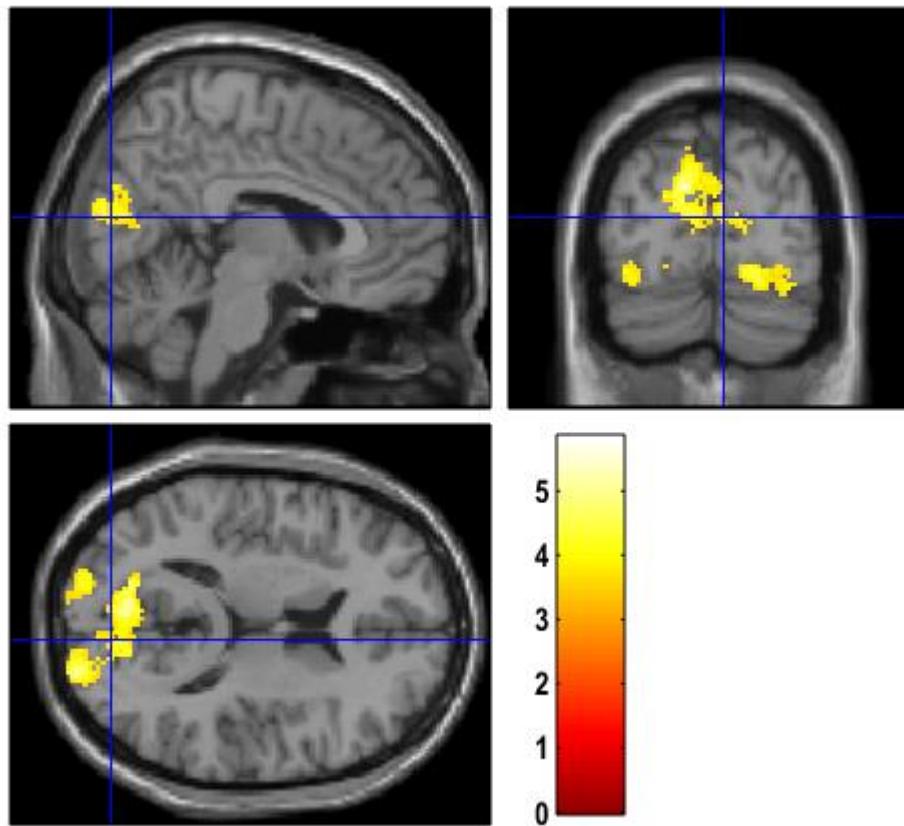
Expected voxels per cluster,  $\langle k \rangle = 19.086$

Volume:  $1341864 = 167733$  voxels = 694.6 resels

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

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

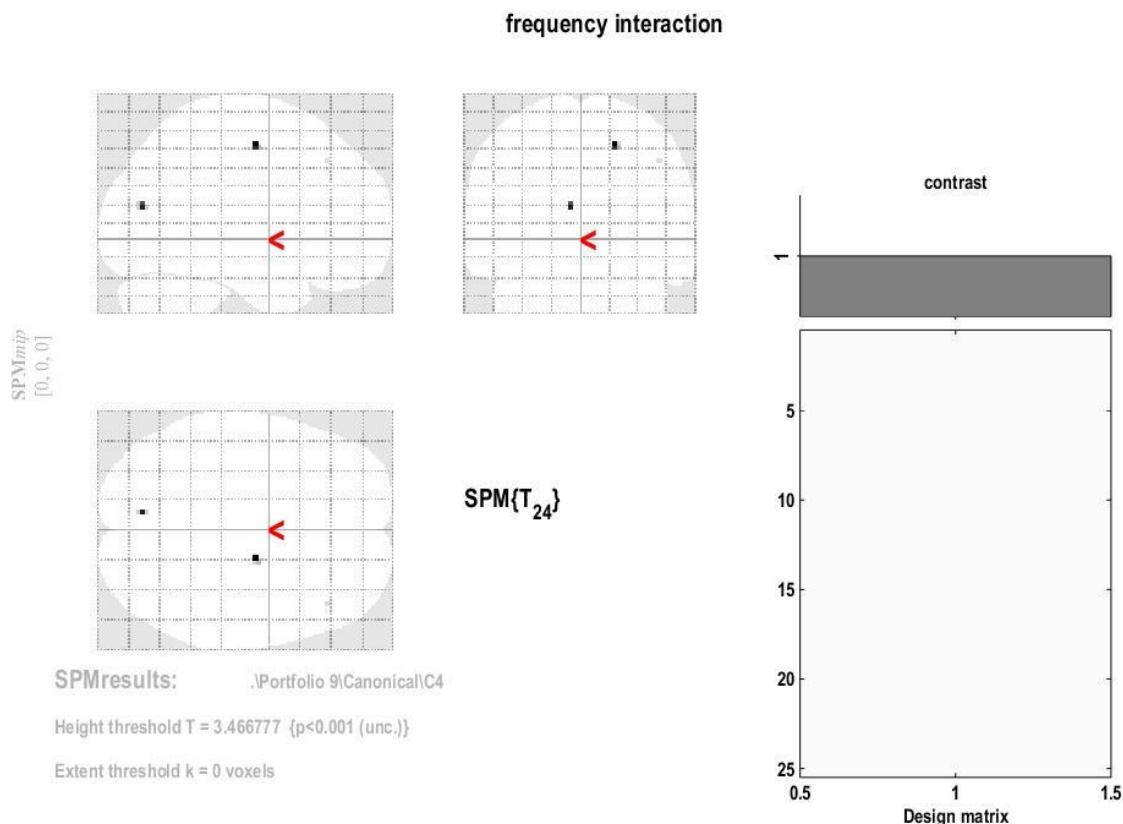
Overlaid image:



The images seem to suggest that the effect of color is only evident in the occipital lobe, where the yellow face trials elicit greater response than the blue face trials. In relation to hypothesis 3, the activation in the motor cortex appear to be the same regardless of condition.

### 3.d Frequency group x colour interaction

Using an uncorrected p-value,  $P < 0.001$ :



#### Statistics: p-values adjusted for search volume

set-level	c	cluster-level					peak-level					mm mm mm		
		p <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	k <sub>E</sub>	p <sub>uncorr</sub>	p <sub>FWE-corr</sub>	q <sub>FDR-corr</sub>	T	(Z <sub>≥</sub> )	p <sub>uncorr</sub>				
0.997	3	0.999	0.849	4	0.662	0.989	0.781	3.92	3.41	0.000	18	-10	52	
		0.998	0.849	5	0.619	0.994	0.781	3.84	3.36	0.000	-8	-76	16	
		1.000	0.849	1	0.849	1.000	0.998	3.47	3.09	0.001	44	32	42	

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, <k> = 20.030

Expected number of clusters, <c> = 9.91

FWEp: 6.223, FDRp: Inf, FWEc: Inf, FDRc: Inf

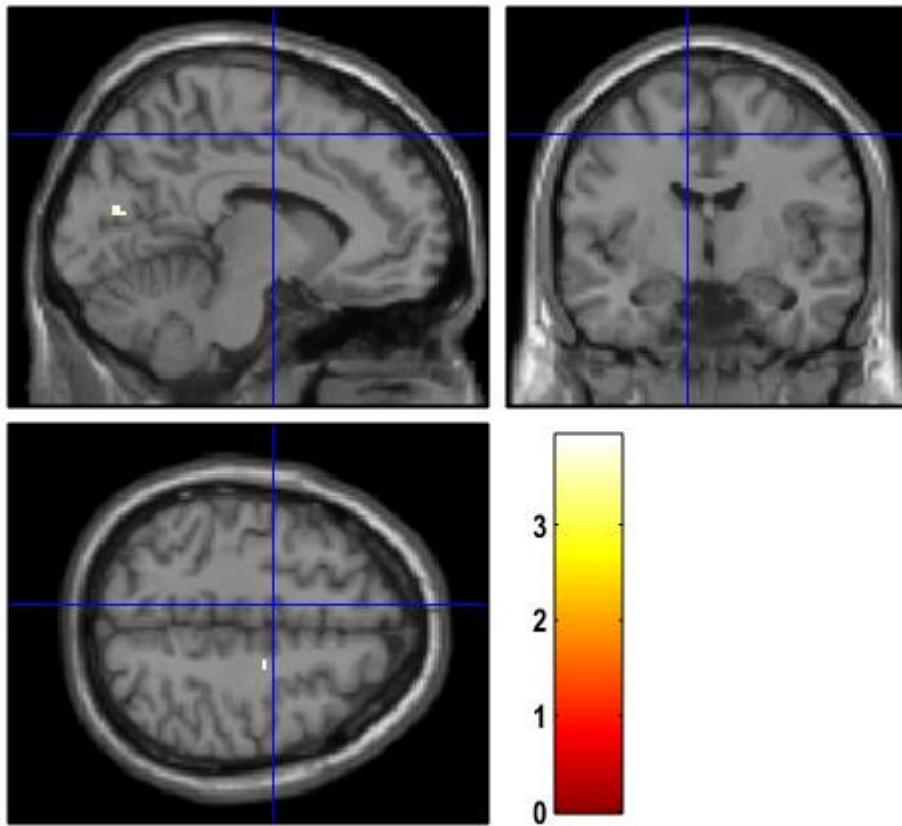
Degrees of freedom = [1.0, 24.0]

FWHM = 12.3 12.3 12.2 mm mm mm; 6.2 6.2 6.1 {voxels}

Volume: 1341864 = 167733 voxels = 661.8 resels

Voxel size: 2.0 2.0 2.0 mm mm mm; (resel = 231.18 voxels)

Overlaid image:



There seems to be a slightly greater activation in what could be the right motor cortex as well as in the occipital lobe for the infrequent stimuli compared to the frequent stimuli. However, the difference in activation is very small and might also be due to other factors.

## Portfolio 10: PCA factor analysis

Group 3: Tobias, Riikka, Ruta, Jana and Mie

May 15, 2019

- 1) People at your agency disagree on how many interesting components/factors are present in the test, so they ask you, the factor analysis expert, to determine this. Please add your argument for the number you end on.

```
# Bartlett test
corcontest.bartlett(empathy)

## R was not square, finding R from data

## $chisq
## [1] 5134.514
##
## $p.value
## [1] 9.710948e-165
##
## $df
## [1] 2628

# This shows that the correlation matrix is significantly different from an identity matrix and therefore there will be some relationships between the variables.

# We then do a Kaiser-Meyer-Olkin-test to measure sampling adequacy, which means the proportion of variance among variables that might be common variance.
kmo_test <- KMO(empathy)
kmo_test # We get an overall KMO-value of 0.69, which is fine except that there are many of the single variables that are below 0.5.

## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = empathy)
## Overall MSA = 0.69
## MSA for each item =
##   X2   X3   X4   X1   X2.1  X2.2  X3.1  X2.3   X0   X3.2   X.1   X3.3
## 0.55  0.30  0.82  0.59  0.71  0.54  0.78  0.59  0.66  0.80  0.60  0.72
##  X2.4  X1.1  X3.4  X4.1  X3.5  X0.1  X2.5  X2.6  X3.6  X1.1  X1.2  X3.7
## 0.79  0.54  0.73  0.68  0.82  0.73  0.73  0.30  0.70  0.61  0.66  0.39
##  X2.7  X2.8  X1.2  X4.2  X2.9  X3.8  X4.3  X3.9  X3.10  X2.10  X2.11  X4.4
## 0.82  0.70  0.76  0.55  0.75  0.71  0.73  0.79  0.70  0.53  0.63  0.50
##  X2.12  X4.5  X3.11  X3.12  X1.3  X2.13  X2.14  X2.15  X2.16  X1.3  X2.17  X2.18
## 0.80  0.77  0.68  0.70  0.77  0.67  0.75  0.84  0.54  0.40  0.76  0.75
##  X1.4  X1.5  X0.2  X3.13  X2.19  X2.20  X1.6  X.2  X1.4  X0.3  X4.6  X4.7
## 0.27  0.51  0.72  0.56  0.42  0.65  0.19  0.34  0.52  0.45  0.73  0.81
##  X4.8  X3.14  X2.21  X1.7  X4.9  X4.10  X3.15  X1.8  X3.16  X4.11  X3.17  X3.18
## 0.83  0.83  0.71  0.80  0.30  0.69  0.81  0.66  0.84  0.77  0.81  0.62
```

```

## X3.19
## 0.85

# Finding the column names of the values below 0.5
kmo_filtered <- kmo_test$MSAi[which(kmo_test$MSAi < 0.5)]
kmo_filtered

##      X3      X2.6      X3.7      X4.4      X.1.3      X1.4      X2.19
## 0.2981336 0.3028929 0.3873374 0.4956539 0.4015545 0.2687124 0.4226022
##      X1.6      X.2      X0.3      X4.9
## 0.1887656 0.3415780 0.4530263 0.2980894

# Removing the values that have a KMO-value below 0.5 from a new dataframe called empathy 2
column.names.remove <- c("X3", "X2.6", "X3.7", "X4.4", "X.1.3", "X1.4", "X2.19",
"X1.6", "X.2", "X0.3", "X4.9")
empathy2 <- empathy[, !names(empathy) %in% column.names.remove]

# Determinant of the correlation matrix to empathy and empathy 2
det(cor(empathy)) # The value is below 0.00001 (2.24e-29)

## [1] 2.240972e-29

det(cor(empathy2)) # The value is still below 0.00001 (2.51e-23) but better after removal of data so we use this data for the rest of the analysis

## [1] 2.541762e-23

# Principal component analysis model using all questions as factors
pc1<-principal(empathy2, nfactors = length(empathy2), rotate = "none")
pc1
Edited output:
## Principal Components Analysis
## Call: principal(r = empathy2, nfactors = length(empathy2), rotate = "none")
## Standardized loadings (pattern matrix) based upon correlation matrix
##
##          PC1   PC2   PC3   PC4   PC5   PC6   PC7   PC8   PC9   PC10
## SS loadings    17.98  4.06  2.96  2.36  2.20  2.01  1.77  1.63  1.50  1.44
## Proportion Var  0.29  0.07  0.05  0.04  0.04  0.03  0.03  0.03  0.02  0.02
## Cumulative Var  0.29  0.36  0.40  0.44  0.48  0.51  0.54  0.56  0.59  0.61
## Proportion Explained  0.29  0.07  0.05  0.04  0.04  0.03  0.03  0.03  0.02  0.02
## Cumulative Proportion  0.29  0.36  0.40  0.44  0.48  0.51  0.54  0.56  0.59  0.61
##          PC11  PC12  PC13  PC14  PC15  PC16  PC17  PC18  PC19  PC20
## SS loadings     1.34  1.27  1.17  1.14  1.06  1.06  1.02  0.95  0.91  0.88
## Proportion Var  0.02  0.02  0.02  0.02  0.02  0.02  0.02  0.02  0.01  0.01
## Cumulative Var  0.63  0.65  0.67  0.69  0.71  0.73  0.74  0.76  0.77  0.79
## Proportion Explained  0.02  0.02  0.02  0.02  0.02  0.02  0.02  0.02  0.01  0.01
## Cumulative Proportion  0.63  0.65  0.67  0.69  0.71  0.73  0.74  0.76  0.77  0.79
##          PC21  PC22  PC23  PC24  PC25  PC26  PC27  PC28  PC29  PC30

```

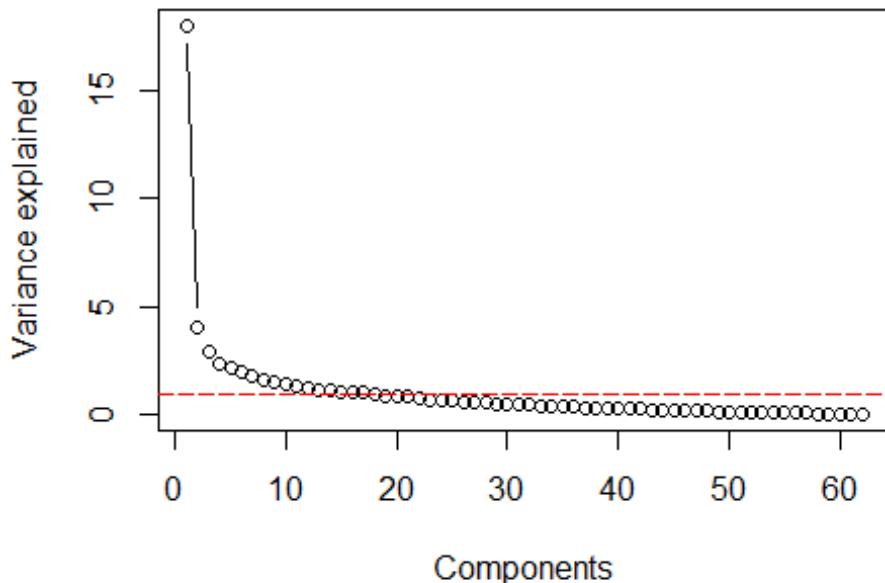
```

## SS loadings      0.86 0.77 0.71 0.68 0.65 0.61 0.59 0.57 0.53 0.50
## Proportion Var  0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01
## Cumulative Var 0.80 0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88 0.89
## Proportion Explained 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01
## Cumulative Proportion 0.80 0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88 0.89
##                                         PC31 PC32 PC33 PC34 PC35 PC36 PC37 PC38 PC39 PC40
## SS loadings      0.47 0.46 0.43 0.42 0.41 0.38 0.34 0.33 0.32 0.31
## Proportion Var   0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01
## Cumulative Var  0.90 0.91 0.91 0.92 0.93 0.93 0.94 0.94 0.95 0.95
## Proportion Explained 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01
## Cumulative Proportion 0.90 0.91 0.91 0.92 0.93 0.93 0.94 0.94 0.95 0.95
##                                         PC41 PC42 PC43 PC44 PC45 PC46 PC47 PC48 PC49 PC50
## SS loadings      0.27 0.26 0.23 0.23 0.22 0.18 0.17 0.16 0.16 0.14
## Proportion Var   0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
## Cumulative Var  0.96 0.96 0.97 0.97 0.97 0.98 0.98 0.98 0.98 0.99
## Proportion Explained 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
## Cumulative Proportion 0.96 0.96 0.97 0.97 0.97 0.98 0.98 0.98 0.98 0.99
##                                         PC51 PC52 PC53 PC54 PC55 PC56 PC57 PC58 PC59 PC60
## SS loadings      0.13 0.11 0.11 0.09 0.09 0.07 0.07 0.06 0.06 0.04
## Proportion Var   0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
## Cumulative Var  0.99 0.99 0.99 0.99 0.99 1.00 1.00 1.00 1.00 1.00
## Proportion Explained 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
## Cumulative Proportion 0.99 0.99 0.99 0.99 0.99 1.00 1.00 1.00 1.00 1.00
##                                         PC61 PC62
## SS loadings      0.04 0.03
## Proportion Var   0.00 0.00
## Cumulative Var  1.00 1.00
## Proportion Explained 0.00 0.00
## Cumulative Proportion 1.00 1.00
##
## Mean item complexity = 8
## Test of the hypothesis that 62 components are sufficient.
##
## The root mean square of the residuals (RMSR) is 0
## with the empirical chi square 0 with prob < NA
##
## Fit based upon off diagonal values = 1
# Using the Kaiser's criterion we would infer that the first 17 variables are of
# interest since these are above or equal to 1, however Kaiser's criterion is
# normally used for smaller sample sizes around 30 and since our data surpasses that
# we decide to not use this to determine number of factors.

# Instead we inspect the scree plot showing point of inlexion around the sixth
# variable.
plot(pc1$values, type = "b", xlab = "Components", ylab = "Variance explained") +
  title("Scree plot")+
  abline(h = 1, col="red", lty=5)

```

## Scree plot



```
## integer(0)

# Second principal component analysis using the point of inflexion as cutoff point
pc2 <- principal(empathy2, nfactors = 5, rotate = "none")
pc2

## Principal Components Analysis
## Call: principal(r = empathy2, nfactors = 5, rotate = "none")
## Standardized loadings (pattern matrix) based upon correlation matrix
##          PC1    PC2    PC3    PC4    PC5     h2     u2 com
## X2       0.37  -0.16   0.38  -0.02   0.14  0.33  0.67  2.7
## X4       0.64   0.27  -0.07  -0.16   0.13  0.53  0.47  1.6
## X1       0.47  -0.34   0.26   0.01   0.14  0.42  0.58  2.7
## X2.1     0.45  -0.01  -0.27   0.15  -0.01  0.30  0.70  1.9
## X2.2     0.29  -0.10   0.31  -0.06   0.06  0.20  0.80  2.4
## X3.1     0.57   0.14  -0.16   0.37   0.13  0.52  0.48  2.2
## X2.3     0.43   0.17   0.02  -0.26  -0.20  0.33  0.67  2.5
## X0       0.42  -0.21   0.26  -0.25  -0.02  0.35  0.65  3.0
## X3.2     0.50  -0.29   0.08  -0.10   0.30  0.44  0.56  2.5
## X.1      0.53   0.23   0.10   0.22  -0.02  0.39  0.61  1.8
## X3.3     0.42   0.13   0.45   0.13  -0.08  0.42  0.58  2.4
## X2.4     0.62   0.09   0.08   0.24  -0.02  0.46  0.54  1.4
## X1.1     0.36   0.18   0.16   0.35   0.14  0.33  0.67  3.2
## X3.4     0.46   0.14   0.24   0.15  -0.02  0.32  0.68  2.0
## X4.1     0.56  -0.36   0.03  -0.13   0.26  0.53  0.47  2.3
```

```

## X3.5  0.66  0.25 -0.19  0.02  0.06  0.54  0.46  1.5
## X0.1   0.60 -0.08  0.37 -0.05 -0.02  0.50  0.50  1.7
## X2.5   0.51  0.01  0.31  0.32  0.12  0.48  0.52  2.6
## X3.6   0.67  0.09 -0.04  0.33  0.08  0.57  0.43  1.5
## X.1.1   0.42  0.07  0.40 -0.05 -0.24  0.40  0.60  2.7
## X1.2   0.42  0.42 -0.25 -0.13 -0.03  0.43  0.57  2.8
## X2.7   0.65  0.40  0.13 -0.06 -0.05  0.60  0.40  1.8
## X2.8   0.59 -0.41  0.25  0.03  0.08  0.60  0.40  2.2
## X.1.2   0.49  0.25  0.18  0.03  0.07  0.34  0.66  1.9
## X4.2   0.36 -0.05  0.22  0.30 -0.20  0.31  0.69  3.3
## X2.9   0.58  0.18 -0.05 -0.34 -0.22  0.53  0.47  2.2
## X3.8   0.43  0.30  0.36 -0.17 -0.16  0.46  0.54  3.5
## X4.3   0.57 -0.23 -0.16 -0.31  0.23  0.55  0.45  2.5
## X3.9   0.67 -0.22 -0.10 -0.24  0.16  0.59  0.41  1.7
## X3.10  0.50 -0.17  0.20 -0.24  0.26  0.44  0.56  2.7
## X2.10  0.39 -0.21  0.02 -0.25  0.32  0.37  0.63  3.3
## X2.11  0.52  0.11 -0.27 -0.15  0.31  0.47  0.53  2.5
## X2.12  0.71 -0.18 -0.20 -0.28  0.21  0.69  0.31  1.8
## X4.5   0.64  0.26 -0.14 -0.05  0.06  0.50  0.50  1.5
## X3.11  0.59  0.20  0.00  0.08 -0.26  0.46  0.54  1.7
## X3.12  0.45  0.26  0.06 -0.22  0.17  0.36  0.64  2.5
## X1.3   0.66  0.10  0.26  0.12  0.11  0.55  0.45  1.5
## X2.13  0.52  0.25  0.26 -0.01 -0.23  0.45  0.55  2.4
## X2.14  0.68  0.28  0.23  0.09  0.14  0.62  0.38  1.7
## X2.15  0.75  0.07  0.25 -0.11  0.14  0.67  0.33  1.4
## X2.16  0.44  0.58 -0.28  0.03  0.03  0.60  0.40  2.4
## X2.17  0.46  0.40 -0.18 -0.22  0.18  0.48  0.52  3.2
## X2.18  0.43  0.46 -0.39 -0.15  0.14  0.59  0.41  3.4
## X1.5   0.31  0.31 -0.18  0.07 -0.30  0.32  0.68  3.7
## X0.2   0.42  0.54 -0.04 -0.13 -0.13  0.51  0.49  2.2
## X3.13  0.37 -0.08 -0.29  0.35  0.28  0.44  0.56  3.9
## X2.20  0.32 -0.07 -0.15  0.58  0.30  0.56  0.44  2.3
## X.1.4   0.29 -0.12 -0.27  0.30  0.15  0.28  0.72  3.8
## X4.6   0.65 -0.27 -0.28 -0.09 -0.10  0.59  0.41  1.9
## X4.7   0.55 -0.41 -0.07  0.04 -0.04  0.48  0.52  1.9
## X4.8   0.63 -0.40 -0.20 -0.14 -0.12  0.63  0.37  2.2
## X3.14  0.65 -0.20 -0.34 -0.06 -0.15  0.60  0.40  1.9
## X2.21  0.65 -0.12  0.05  0.02  0.04  0.44  0.56  1.1
## X1.7   0.67 -0.11 -0.12  0.16  0.07  0.51  0.49  1.3
## X4.10  0.51 -0.23 -0.19  0.14 -0.31  0.47  0.53  2.6
## X3.15  0.70 -0.10 -0.08 -0.03 -0.44  0.69  0.31  1.8
## X1.8   0.55 -0.24 -0.13 -0.09 -0.19  0.43  0.57  1.8
## X3.16  0.68 -0.15 -0.11  0.11 -0.15  0.53  0.47  1.3
## X4.11  0.45 -0.42  0.01 -0.08 -0.40  0.55  0.45  3.0
## X3.17  0.54 -0.13 -0.06 -0.01 -0.28  0.39  0.61  1.7
## X3.18  0.53 -0.29 -0.34  0.22 -0.24  0.59  0.41  3.3
## X3.19  0.67 -0.25 -0.01  0.11 -0.19  0.56  0.44  1.5
##

```

```

##                                     PC1  PC2  PC3  PC4  PC5
## SS loadings                 17.98 4.06 2.96 2.36 2.20
## Proportion Var              0.29 0.07 0.05 0.04 0.04
## Cumulative Var              0.29 0.36 0.40 0.44 0.48
## Proportion Explained        0.61 0.14 0.10 0.08 0.07
## Cumulative Proportion       0.61 0.75 0.85 0.93 1.00
##
## Mean item complexity = 2.3
## Test of the hypothesis that 5 components are sufficient.
##
## The root mean square of the residuals (RMSR) is 0.06
## with the empirical chi square 1525.86 with prob < 0.88
##
## Fit based upon off diagonal values = 0.96

# Correlation matrix of the empathy 2 data
empathyMatrix <- as.data.frame(cor(empathy2))

# Function to calculate the residuals, proportion of residuals above and below
# 0.05 and the root means squared residuals.
# It also gives us a histogram plot of the residuals to see if these are normally
# distributed.
residual.stats<-function(matrix){
  residuals <- as.matrix(matrix[upper.tri(matrix)])
  large.resid <- abs(residuals) > 0.05
  numberLargeResid <- sum(large.resid)
  propLargeResid <- numberLargeResid/nrow(residuals)
  rmsr <- sqrt(mean(residuals^2))

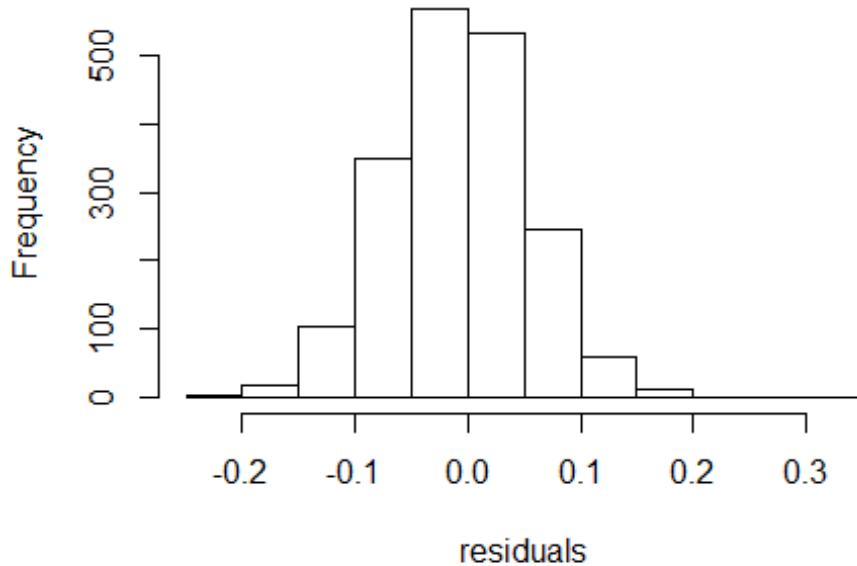
  cat("Root means squared residual = ", rmsr, "\n")
  cat("Number of absolute residuals > 0.05 = ", numberLargeResid, "\n")
  cat("Proportion of absolute residuals > 0.05 = ", propLargeResid, "\n")
  hist(residuals)
}

# Calculating the factor residuals between the correlation matrix and the Loadings
# from the pc2 model.
resids <- factor.residuals(empathyMatrix, pc2$loadings)
# Using the factor residuals as input in the residual.stats function
residual.stats(resids)

## Root means squared residual = 0.06228457
## Number of absolute residuals > 0.05 = 788
## Proportion of absolute residuals > 0.05 = 0.4167107

```

## Histogram of residuals



```
# The root means squared residual is 0.06 and since it is below 0.08 we don't need  
# to extract more factors.  
# The histogram of the residuals seems to be normally distributed and are mostly  
# below 0.05, which is a good thing.
```

```
# We believe that the factors correlate in some way since they are all related to  
# empathy. Therefore, we decide to use the oblique rotation  
# Principal analysis using rotation  
pc3 <- principal(empathy2, nfactors = 5, rotate = "oblimin")  
pc3  
  
## Principal Components Analysis  
## Call: principal(r = empathy2, nfactors = 5, rotate = "oblimin")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
##          TC1   TC3   TC2   TC5   TC4   h2   u2 com  
## X2      -0.06  0.43 -0.20  0.34  0.03  0.33  0.67 2.4  
## X4      0.03  0.19  0.51  0.26  0.04  0.53  0.47 1.8  
## X1      0.12  0.30 -0.26  0.42  0.10  0.42  0.58 3.0  
## X2.1    0.32 -0.07  0.21  0.02  0.28  0.30  0.70 2.8  
## X2.2   -0.01  0.35 -0.13  0.25 -0.05  0.20  0.80 2.2  
## X3.1    0.13  0.16  0.25 -0.03  0.52  0.52  0.48 1.8  
## X2.3    0.25  0.18  0.32  0.06 -0.27  0.33  0.67 3.6  
## X0      0.17  0.29 -0.11  0.38 -0.21  0.35  0.65 3.2  
## X3.2    0.04  0.11 -0.07  0.57  0.14  0.44  0.56 1.2
```

```

## X.1    0.10  0.41  0.22 -0.08  0.24  0.39  0.61  2.5
## X3.3   0.00  0.66 -0.04 -0.03  0.02  0.42  0.58  1.0
## X2.4    0.21  0.38  0.13 -0.01  0.28  0.46  0.54  2.7
## X1.1   -0.11  0.40  0.07 -0.06  0.38  0.33  0.67  2.3
## X3.4    0.05  0.47  0.08 -0.01  0.13  0.32  0.68  1.3
## X4.1    0.15  0.07 -0.08  0.61  0.11  0.53  0.47  1.2
## X3.5    0.18  0.12  0.49  0.11  0.19  0.54  0.46  1.8
## X0.1    0.15  0.54 -0.07  0.26 -0.04  0.50  0.50  1.7
## X2.5   -0.01  0.54 -0.09  0.08  0.35  0.48  0.52  1.8
## X3.6    0.20  0.29  0.19  0.02  0.44  0.57  0.43  2.6
## X.1.1   0.18  0.57 -0.03 -0.02 -0.18  0.40  0.60  1.4
## X1.2    0.09  0.00  0.64 -0.03 -0.01  0.43  0.57  1.0
## X2.7    0.06  0.46  0.47  0.03 -0.01  0.60  0.40  2.0
## X2.8    0.26  0.34 -0.30  0.43  0.13  0.60  0.40  3.8
## X.1.2  -0.05  0.42  0.26  0.09  0.09  0.34  0.66  1.9
## X4.2    0.28  0.41 -0.16 -0.18  0.18  0.31  0.69  3.0
## X2.9    0.34  0.16  0.44  0.13 -0.30  0.53  0.47  3.3
## X3.8    0.01  0.57  0.24  0.00 -0.25  0.46  0.54  1.8
## X4.3    0.18 -0.11  0.18  0.63 -0.01  0.55  0.45  1.4
## X3.9    0.26  0.00  0.16  0.56  0.01  0.59  0.41  1.6
## X3.10   -0.05  0.24  0.01  0.59 -0.04  0.44  0.56  1.3
## X2.10   -0.05  0.01  0.04  0.61  0.00  0.37  0.63  1.0
## X2.11   -0.01 -0.12  0.45  0.41  0.17  0.47  0.53  2.4
## X2.12   0.26 -0.08  0.27  0.60  0.03  0.69  0.31  1.8
## X4.5    0.13  0.15  0.49  0.14  0.12  0.50  0.50  1.7
## X3.11   0.38  0.32  0.29 -0.16  0.04  0.46  0.54  3.3
## X3.12   -0.13  0.22  0.41  0.29 -0.05  0.36  0.64  2.7
## X1.3    0.02  0.53  0.12  0.19  0.21  0.55  0.45  1.7
## X2.13   0.18  0.53  0.21 -0.10 -0.10  0.45  0.55  1.7
## X2.14   -0.07  0.55  0.30  0.15  0.20  0.62  0.38  2.1
## X2.15   0.04  0.49  0.20  0.39  0.03  0.67  0.33  2.3
## X2.16   -0.01  0.05  0.74 -0.14  0.16  0.60  0.40  1.2
## X2.17   -0.10  0.03  0.64  0.22  0.00  0.48  0.52  1.3
## X2.18   -0.02 -0.14  0.76  0.09  0.08  0.59  0.41  1.1
## X1.5    0.32  0.07  0.38 -0.34  0.00  0.32  0.68  3.0
## X0.2    0.04  0.25  0.63 -0.14 -0.12  0.51  0.49  1.5
## X3.13   0.08 -0.13  0.10  0.13  0.59  0.44  0.56  1.3
## X2.20   0.00  0.04 -0.05 -0.01  0.75  0.56  0.44  1.0
## X.1.4   0.17 -0.14  0.04  0.05  0.45  0.28  0.72  1.6
## X4.6    0.60 -0.12  0.12  0.26  0.06  0.59  0.41  1.6
## X4.7    0.49  0.03 -0.16  0.28  0.15  0.48  0.52  2.1
## X4.8    0.63 -0.10 -0.02  0.34 -0.01  0.63  0.37  1.6
## X3.14   0.63 -0.14  0.20  0.17  0.08  0.60  0.40  1.5
## X2.21   0.27  0.25  0.06  0.27  0.14  0.44  0.56  3.6
## X1.7    0.33  0.12  0.12  0.19  0.33  0.51  0.49  3.2
## X4.10   0.68  0.01 -0.02 -0.09  0.11  0.47  0.53  1.1
## X3.15   0.75  0.19  0.13 -0.07 -0.09  0.69  0.31  1.2
## X1.8    0.56  0.01  0.04  0.16 -0.03  0.43  0.57  1.2

```

```

## X3.16  0.53  0.13  0.08  0.08  0.18  0.53  0.47 1.5
## X4.11  0.74  0.09 -0.24  0.06 -0.17  0.55  0.45 1.4
## X3.17  0.56  0.13  0.06  0.01 -0.03  0.39  0.61 1.1
## X3.18  0.72 -0.14 -0.01 -0.06  0.27  0.59  0.41 1.4
## X3.19  0.57  0.21 -0.07  0.09  0.13  0.56  0.44 1.5
##
##          TC1  TC3  TC2  TC5  TC4
## SS loadings    7.56 6.66 6.11 5.68 3.56
## Proportion Var 0.12 0.11 0.10 0.09 0.06
## Cumulative Var 0.12 0.23 0.33 0.42 0.48
## Proportion Explained 0.26 0.23 0.21 0.19 0.12
## Cumulative Proportion 0.26 0.48 0.69 0.88 1.00
##
## With component correlations of
##      TC1  TC3  TC2  TC5  TC4
## TC1 1.00 0.33 0.26 0.42 0.25
## TC3 0.33 1.00 0.28 0.27 0.16
## TC2 0.26 0.28 1.00 0.18 0.17
## TC5 0.42 0.27 0.18 1.00 0.20
## TC4 0.25 0.16 0.17 0.20 1.00
##
## Mean item complexity = 1.9
## Test of the hypothesis that 5 components are sufficient.
##
## The root mean square of the residuals (RMSR) is 0.06
## with the empirical chi square 1525.86 with prob < 0.88
##
## Fit based upon off diagonal values = 0.96

# Binding the pc4 model Loadings from each participant to the empathy2 dataframe.
empathy2 <- cbind(empathy2, pc3$scores)

```

After analysis of the data, we decide on having five principal components. The point of inflexion is a very subjective measure, so one could have argued that it was around 3 or 4.

## 2) In order to short-list candidates for the position, your job is to find the highest and lowest scoring candidate on each factor.

```

min_TC1 <- min(empathy2$TC1)
empathy[which(empathy$TC1 == min_TC1), 0] # Participant 33 scores Lowest on the
first factor

## data frame with 0 columns and 0 rows

max_TC1 <- max(empathy2$TC1)
empathy[which(empathy2$TC1 == max_TC1), 0] # Participant 82 scores highest on the
first factor

## data frame with 0 columns and 1 row

```

```

min_TC2 <- min(empathy2$TC2)
empathy[which(empathy2$TC2 == min_TC2), 0] # Participant 33 also scores Lowest on
the second factor

## data frame with 0 columns and 1 row

max_TC2 <- max(empathy2$TC2)
empathy[which(empathy2$TC2 == max_TC2), 0] # Participant 52 scores highest on the
second factor

## data frame with 0 columns and 1 row

min_TC3 <- min(empathy2$TC3)
empathy[which(empathy2$TC3 == min_TC3), 0] # Participant 33 also scores Lowest on
the third factor

## data frame with 0 columns and 1 row

max_TC3 <- max(empathy2$TC3)
empathy[which(empathy2$TC3 == max_TC3), 0] # Participant 59 scores highest on the
third factor

## data frame with 0 columns and 1 row

min_TC4 <- min(empathy2$TC4)
empathy[which(empathy2$TC4 == min_TC4), 0] # Participant 33 also scores Lowest on
the fourth factor

## data frame with 0 columns and 1 row

max_TC4 <- max(empathy2$TC4)
empathy[which(empathy2$TC4 == max_TC4), 0] # Participant 26 scores highest on the
fourth factor

## data frame with 0 columns and 1 row

min_TC5 <- min(empathy2$TC5)
empathy[which(empathy2$TC5 == min_TC5), 0] # Participant 33 also scores Lowest on
the fifth factor

## data frame with 0 columns and 1 row

max_TC5 <- max(empathy2$TC5)
empathy[which(empathy2$TC5 == max_TC5), 0] # Participant 43 scores highest on the
fifth factor

## data frame with 0 columns and 1 row

```

So our recommendation is to eliminate participant 33 from the list of applicants due to lack of empathy on all factors and short-list participant 82, 52, 59, 26 and 43.

**3) Your boss asks you what you think of his new empathy test (The physical empathy test). Does it really measure anything that the old scales cannot capture?**

```
empathyMatrix <- as.matrix(empathy)
# Making a matrix with the empathy data excluding The Physical Empathy Test
restEmpathy <- empathyMatrix[c(1:58),]

# Doing the PCA on the new matrix
restPCA <- prcomp(restEmpathy, scale = TRUE)
summary(restPCA)

## Importance of components:
##          PC1      PC2      PC3      PC4      PC5      PC6
## Standard deviation 4.7482 2.1850 1.9351 1.77322 1.67155 1.61914
## Proportion of Variance 0.3088 0.0654 0.0513 0.04307 0.03827 0.03591
## Cumulative Proportion 0.3088 0.3742 0.4255 0.46861 0.50688 0.54280
##          PC7      PC8      PC9      PC10     PC11     PC12
## Standard deviation 1.58793 1.52055 1.39849 1.33273 1.30658 1.28690
## Proportion of Variance 0.03454 0.03167 0.02679 0.02433 0.02339 0.02269
## Cumulative Proportion 0.57734 0.60901 0.63580 0.66013 0.68352 0.70621
##          PC13     PC14     PC15     PC16     PC17     PC18
## Standard deviation 1.20584 1.16803 1.13904 1.11964 1.08286 1.04970
## Proportion of Variance 0.01992 0.01869 0.01777 0.01717 0.01606 0.01509
## Cumulative Proportion 0.72612 0.74481 0.76259 0.77976 0.79582 0.81091
##          PC19     PC20     PC21     PC22     PC23     PC24
## Standard deviation 1.0110 0.99037 0.95275 0.91511 0.88421 0.85922
## Proportion of Variance 0.0140 0.01344 0.01243 0.01147 0.01071 0.01011
## Cumulative Proportion 0.8249 0.83835 0.85079 0.86226 0.87297 0.88308
##          PC25     PC26     PC27     PC28     PC29     PC30
## Standard deviation 0.84537 0.80780 0.78824 0.76446 0.72288 0.70689
## Proportion of Variance 0.00979 0.00894 0.00851 0.00801 0.00716 0.00685
## Cumulative Proportion 0.89287 0.90181 0.91032 0.91833 0.92549 0.93233
##          PC31     PC32     PC33     PC34     PC35     PC36
## Standard deviation 0.6675 0.64916 0.64121 0.61971 0.60315 0.57474
## Proportion of Variance 0.0061 0.00577 0.00563 0.00526 0.00498 0.00453
## Cumulative Proportion 0.9384 0.94421 0.94984 0.95510 0.96008 0.96461
##          PC37     PC38     PC39     PC40     PC41     PC42
## Standard deviation 0.55713 0.50673 0.48893 0.47225 0.46485 0.42445
## Proportion of Variance 0.00425 0.00352 0.00327 0.00306 0.00296 0.00247
## Cumulative Proportion 0.96886 0.97238 0.97565 0.97871 0.98167 0.98414
##          PC43     PC44     PC45     PC46     PC47     PC48
## Standard deviation 0.4183 0.39514 0.36692 0.34429 0.32617 0.29765
## Proportion of Variance 0.0024 0.00214 0.00184 0.00162 0.00146 0.00121
## Cumulative Proportion 0.9865 0.98867 0.99052 0.99214 0.99360 0.99481
##          PC49     PC50     PC51     PC52     PC53     PC54
## Standard deviation 0.28635 0.27101 0.25092 0.20644 0.18728 0.16792
## Proportion of Variance 0.00112 0.00101 0.00086 0.00058 0.00048 0.00039
## Cumulative Proportion 0.99593 0.99694 0.99780 0.99839 0.99887 0.99925
##          PC55     PC56     PC57     PC58
```

```

## Standard deviation 0.16211 0.12850 0.10824 2.109e-15
## Proportion of Variance 0.00036 0.00023 0.00016 0.000e+00
## Cumulative Proportion 0.99961 0.99984 1.00000 1.000e+00

# This we compare to the PCA on the original matrix
pcAnalysis <- prcomp(empathyMatrix, scale = TRUE)
summary(pcAnalysis)

## Importance of components:
##                 PC1      PC2      PC3      PC4      PC5      PC6
## Standard deviation 4.2622 2.12544 1.91934 1.77256 1.53282 1.47661
## Proportion of Variance 0.2488 0.06188 0.05046 0.04304 0.03219 0.02987
## Cumulative Proportion 0.2488 0.31073 0.36120 0.40424 0.43642 0.46629
##                 PC7      PC8      PC9      PC10     PC11     PC12
## Standard deviation 1.45532 1.37603 1.34793 1.31911 1.23862 1.22599
## Proportion of Variance 0.02901 0.02594 0.02489 0.02384 0.02102 0.02059
## Cumulative Proportion 0.49530 0.52124 0.54613 0.56997 0.59098 0.61157
##                 PC13     PC14     PC15     PC16     PC17     PC18
## Standard deviation 1.2024 1.16301 1.14830 1.12736 1.07291 1.05523
## Proportion of Variance 0.0198 0.01853 0.01806 0.01741 0.01577 0.01525
## Cumulative Proportion 0.6314 0.64991 0.66797 0.68538 0.70115 0.71640
##                 PC19     PC20     PC21     PC22     PC23     PC24
## Standard deviation 1.04085 1.02624 1.00091 0.98986 0.96831 0.91593
## Proportion of Variance 0.01484 0.01443 0.01372 0.01342 0.01284 0.01149
## Cumulative Proportion 0.73124 0.74567 0.75939 0.77282 0.78566 0.79715
##                 PC25     PC26     PC27     PC28     PC29     PC30
## Standard deviation 0.90684 0.88076 0.87863 0.8628 0.82686 0.82231
## Proportion of Variance 0.01127 0.01063 0.01058 0.0102 0.00937 0.00926
## Cumulative Proportion 0.80842 0.81904 0.82962 0.8398 0.84918 0.85845
##                 PC31     PC32     PC33     PC34     PC35     PC36
## Standard deviation 0.80177 0.78452 0.74915 0.73264 0.7249 0.69569
## Proportion of Variance 0.00881 0.00843 0.00769 0.00735 0.0072 0.00663
## Cumulative Proportion 0.86725 0.87568 0.88337 0.89072 0.8979 0.90455
##                 PC37     PC38     PC39     PC40     PC41     PC42
## Standard deviation 0.6836 0.66606 0.65881 0.62564 0.60728 0.60208
## Proportion of Variance 0.0064 0.00608 0.00595 0.00536 0.00505 0.00497
## Cumulative Proportion 0.9110 0.91703 0.92297 0.92834 0.93339 0.93835
##                 PC43     PC44     PC45     PC46     PC47     PC48
## Standard deviation 0.58984 0.57427 0.54857 0.53475 0.50757 0.50098
## Proportion of Variance 0.00477 0.00452 0.00412 0.00392 0.00353 0.00344
## Cumulative Proportion 0.94312 0.94764 0.95176 0.95568 0.95921 0.96264
##                 PC49     PC50     PC51     PC52     PC53     PC54
## Standard deviation 0.48473 0.47249 0.45686 0.45064 0.4271 0.41432
## Proportion of Variance 0.00322 0.00306 0.00286 0.00278 0.0025 0.00235
## Cumulative Proportion 0.96586 0.96892 0.97178 0.97456 0.9771 0.97941
##                 PC55     PC56     PC57     PC58     PC59     PC60
## Standard deviation 0.40673 0.38927 0.35483 0.34862 0.3420 0.32625
## Proportion of Variance 0.00227 0.00208 0.00172 0.00166 0.0016 0.00146

```

```

## Cumulative Proportion  0.98168 0.98375 0.98548 0.98714 0.9888 0.99020
##                               PC61      PC62      PC63      PC64      PC65      PC66
## Standard deviation     0.31629 0.30214 0.29659 0.27128 0.24951 0.24730
## Proportion of Variance 0.00137 0.00125 0.00121 0.00101 0.00085 0.00084
## Cumulative Proportion  0.99157 0.99283 0.99403 0.99504 0.99589 0.99673
##                               PC67      PC68      PC69      PC70      PC71      PC72
## Standard deviation     0.22326 0.21405 0.19281 0.18844 0.17323 0.14640
## Proportion of Variance 0.00068 0.00063 0.00051 0.00049 0.00041 0.00029
## Cumulative Proportion  0.99741 0.99804 0.99855 0.99904 0.99945 0.99974
##                               PC73
## Standard deviation     0.13785
## Proportion of Variance 0.00026
## Cumulative Proportion 1.00000

# If we compare the two analysis using the Kaiser's criterion we can see that the
original test includes the first 21 components to explain 76% of the variance
whereas the test excluding The Physical Empathy Test only needs the first 19
components to explain 82% of the variance.

```

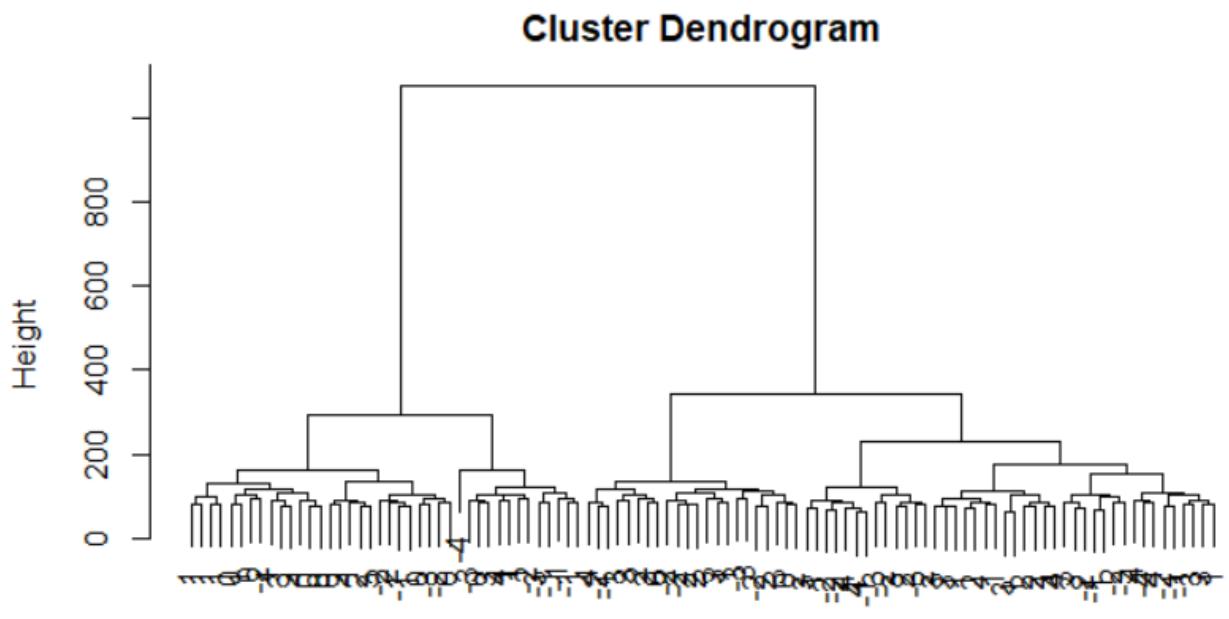
By including the Physical Empathy Test we get 2 more components that helps explain if people are empathetic or not. Therefore, by including it the test measures more than if they are excluded.

#### 4) You also want to impress your boss with a couple of illustrative plots.

```

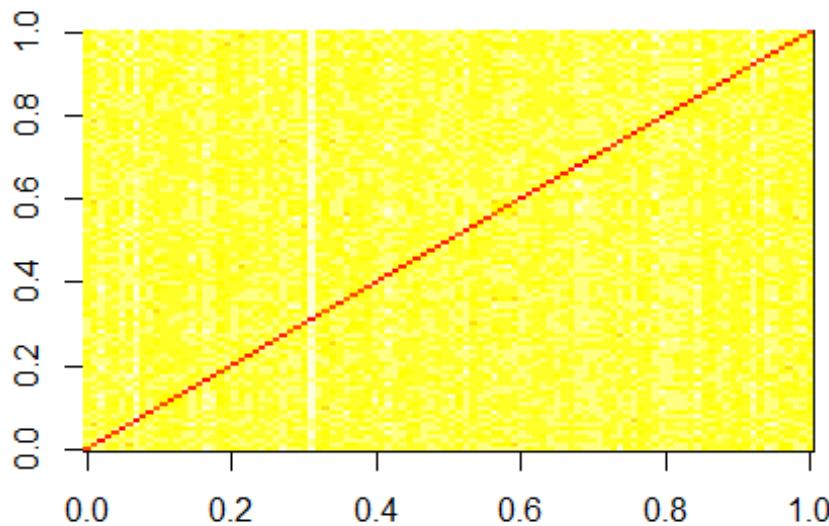
# Making a dendrogram for the original data
empathy_hamming <- hamming.distance(as.matrix(empathyMatrix))
rownames(empathy_hamming) <- empathyMatrix[,1]
#clustering
tree2 <- hclust(dist(empathy_hamming),method="ward.D")
plot(tree2)

```

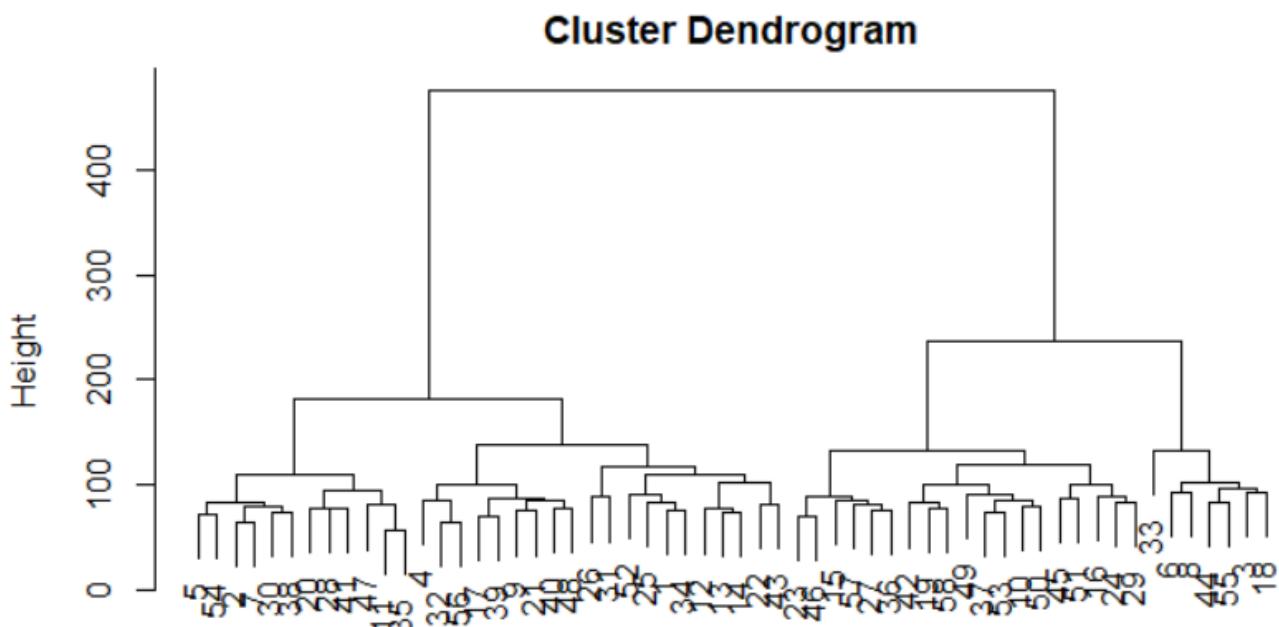


```
dist(empathy_hamming)  
hclust (*, "ward.D")
```

```
# Making a colored correlation matrix  
image(scale(empathy_hamming))
```



```
# Making a dendrogram for the original data excluding the Physical Empathy Test
rest_hamming <- hamming.distance(as.matrix(restEmpathy))
rownames(rest_hamming)[0] <- restEmpathy[,1]
#clustering
tree <- hclust(dist(rest_hamming),method="ward.D")
plot(tree)
```



dist(rest\_hamming)  
hclust (\*, "ward.D")