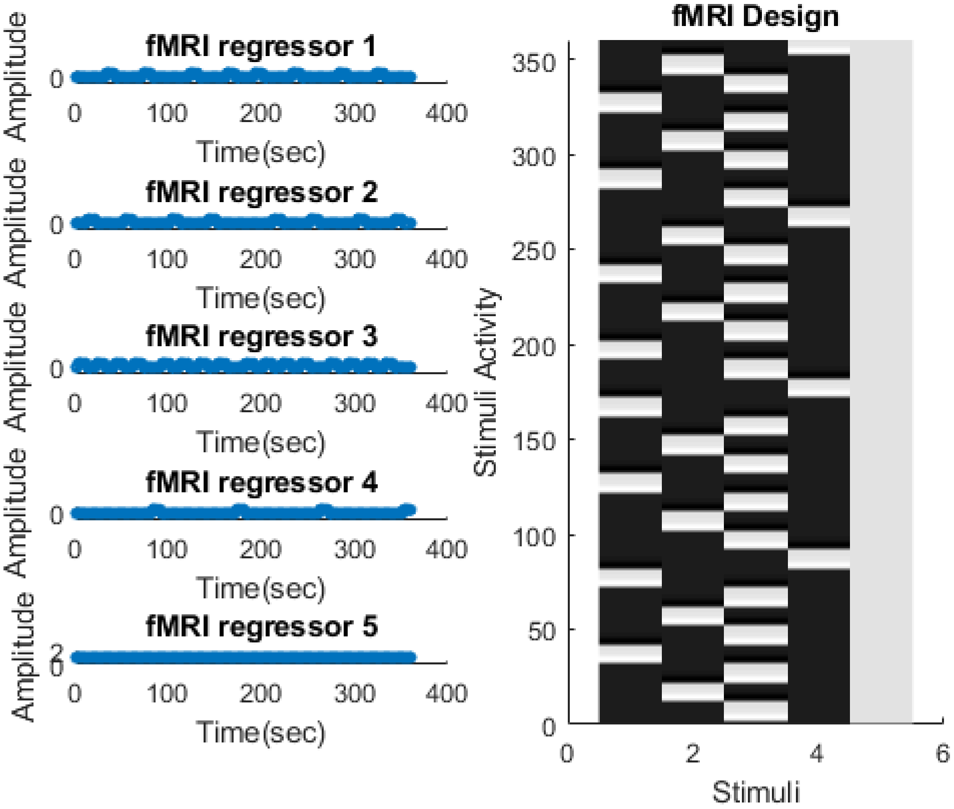
analysefMRIData.m

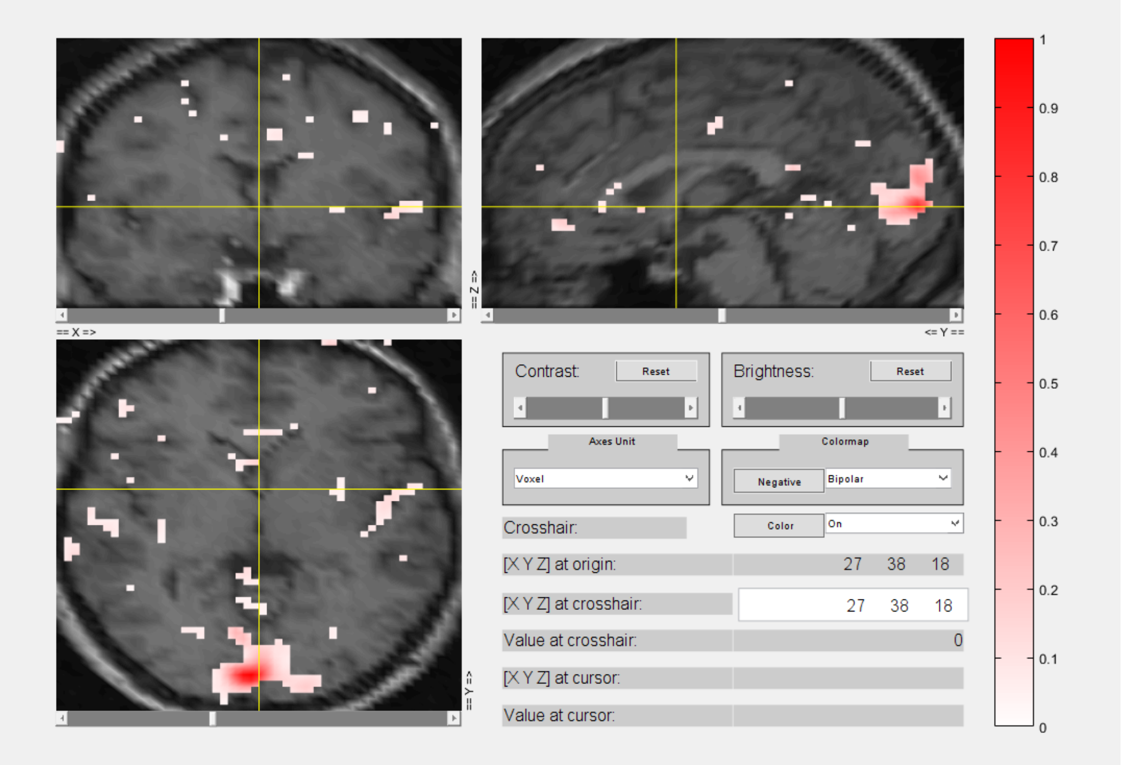
```
function [C, CTags, CAct] = analysefMRIData(file_name)
% Does analysis on fMRI data in a given file name.
% Arguments:
% File_name of fMRI data file.
% Output:
% C a matrix with the contrast of each stimuli against each other
 stimuli
% per voxel. format: (x,y,z,stim1,stim2).
% CTags a matrix with tags of under what x,y pair what stimuli are
% compared.
% CAct a matrix with x,y,z offsets in mm from the center at each
 contrast
% of high activity.
% The fMRI data file !!MUST!! contain:
%   ana (clear structural scan).
%   hrf (Haemodynamic Response Function).
%   Y (bolt response of the actual scan) with x,y,z,nr scans.
%   At least two (can be as many as you like) stimuli activation
 vectors
%   containing the scan nrs during wich the given stimuli was
%   active. The vector names will be used for output (CTags).
% The function will call "generateDesignMatrix2" to generate a design
% matrix. It will then do liniear regression with
% "computeBetaYhatResiduals" followed by computing the contrast
 between all
% given stimuli vectors. (in this demo function it will show a select
 3
% contrasts).
    % mm ofsset zeros for x, y and z voxels (this should be an
 argument or
    % in the data but we hard code it for convience.
    zero = [-26, -37, -17];

    % Loading vars
    data = load(file_name);

    % Init vector for stimuli tags so we can refer to them later.
    vars = fieldnames(data);

    % We use -3 here because we do not need the tags for Y, hrf and
 ana
    % sise they are supposed to be in every dataset.
    nrstim = size(vars,1)-3;
    tags = cell(1,nrstim);

    % Determine the number of scans, x, y and z resolutions for later
 use.
    nrscans = size(data.Y,4);
    resx = size(data.ana, 1);
    resy = size(data.ana, 2);
```

```
   resz = size(data.ana, 3);

   %Loop over loaded vars, ignore Y, hrf and ana.
   index = 0;
   for i = 1:length(vars)
       var = string(vars(i));
       if(var == "Y" || var == "hrf" || var == "ana")
           continue;
       end
       index = index + 1;
       tags(index) = vars(i);
   end

   %Make a design matrix
   [X] = generateDesignMatrix2(data, tags, nrstim, data.hrf,
nrscans);

   %Plot design matrix
   design(X,1:1:nrscans,"fMRI");

   % Calculate the betas, Yhat and residuals
   [betas, Yhat, residuals] = computeBetaYhatResiduals(X, data.Y);

   % init contrast vector C and CTags where we will put the contrast
   % between possible combinations of stimuli and their tags.
   C = zeros(resx, resy, resz, nrstim, nrstim, 'single');
   CTags = string(zeros(nrstim,nrstim));
   topnr = 10;
   CAct = zeros(nrstim,nrstim,topnr,3, 'single');

   % Permutate over all possible combinations, do not compare a
stimuli
   % against itself. Here b1 - b2 = b2 - b1 so it is going to be a
   % symmetric matrix over the diagonals with diagonals 0;
   % Also get the offsets in mm from center of the top 10 most active
   % significant voxels.
   for i = 1:1:nrstim
       for j = 1:1:nrstim
           if string(tags(i)) == string(tags(j))
               CTags(i,j) = "x";
               continue;
           end
           CTags(i,j) = string(tags(i)) + " - " + string(tags(j));
           C(:,:,:,i,j) =
applyContrast(betas(:,:,:,i),betas(:,:,:,j),residuals);
           CAct(i,j,:,:) = findActivityCoord(C(:,:,:,i,j), zero,
topnr);
       end
   end

   %Plotting contrast for,
   % STAT-FIX:
   disp("Contrast for " + CTags(4,3));
   view_scan(data.ana, C(:,:,:,4,3));
```

```
    % NATT-STAT:
    disp("Contrast for " + CTags(1,4));
    view_scan(data.ana, C(:,:,:,1,4));

    % ATT-NATT:
    disp("Contrast for " + CTags(2,1));
    view_scan(data.ana, C(:,:,:,2,1));

end
```

```
Contrast for stat - fix
Contrast for natt - stat
Contrast for att - natt
```





Axes Unit: Voxel

Colormap: Negative Bipolar

Crosshair: Color On

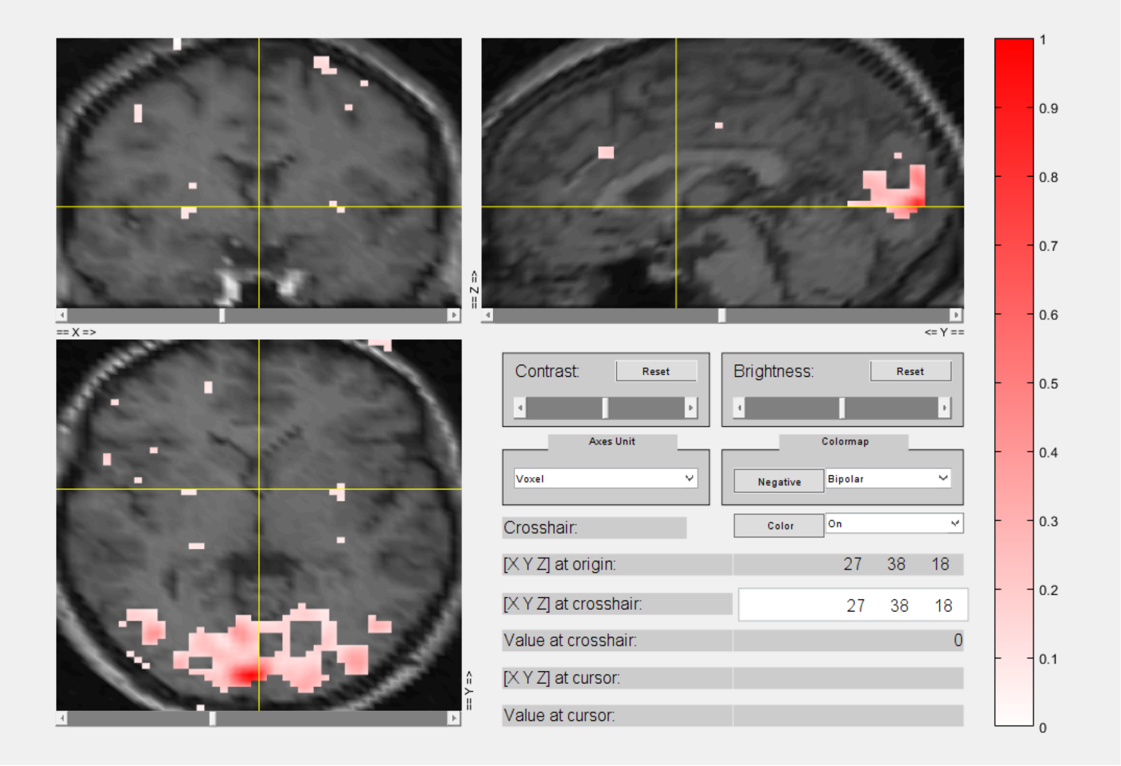
[X Y Z] at origin: 27 38 18

[X Y Z] at crosshair: 27 38 18

Value at crosshair: 0

[X Y Z] at cursor:

Value at cursor:



Axes Unit: Voxel

Colormap: Negative Bipolar

Crosshair: Color On

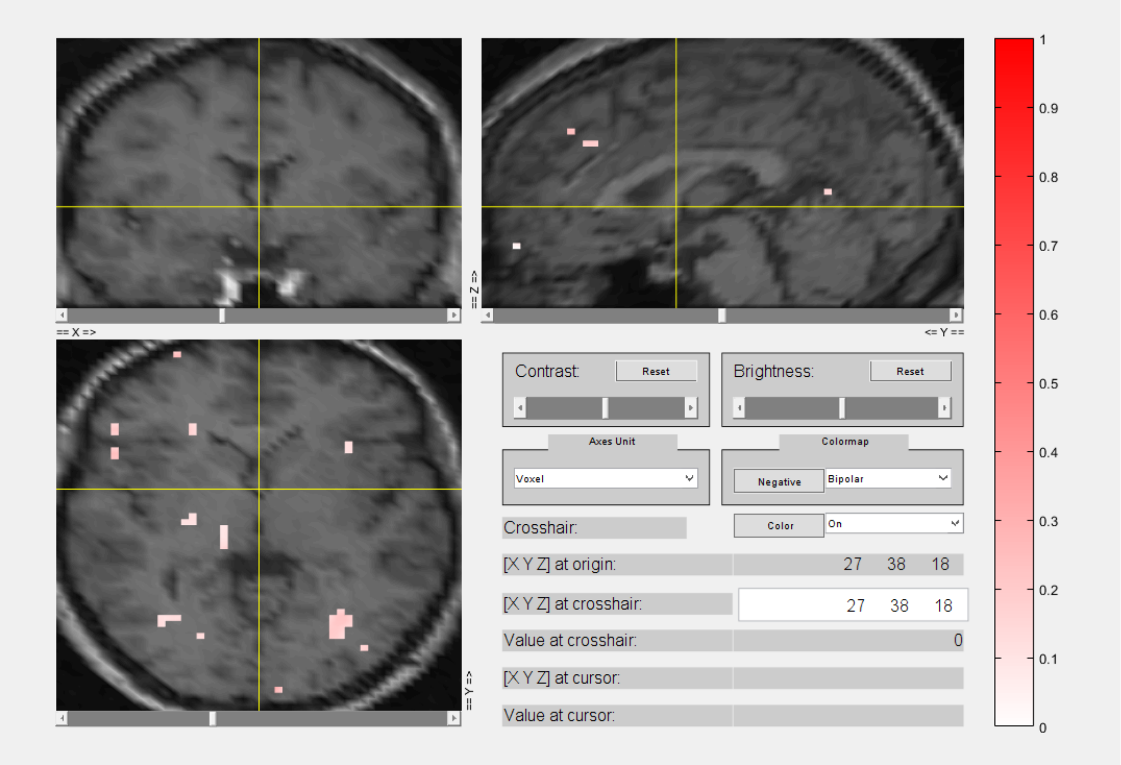
[X Y Z] at origin: 27 38 18

[X Y Z] at crosshair: 27 38 18

Value at crosshair: 0

[X Y Z] at cursor:

Value at cursor:



Axes Unit: Voxel

Colormap: Bipolar

Crosshair: On

[X Y Z] at origin: 27 38 18

[X Y Z] at crosshair: 27 38 18

Value at crosshair: 0

[X Y Z] at cursor:

Value at cursor:

Published with MATLAB® R2017b