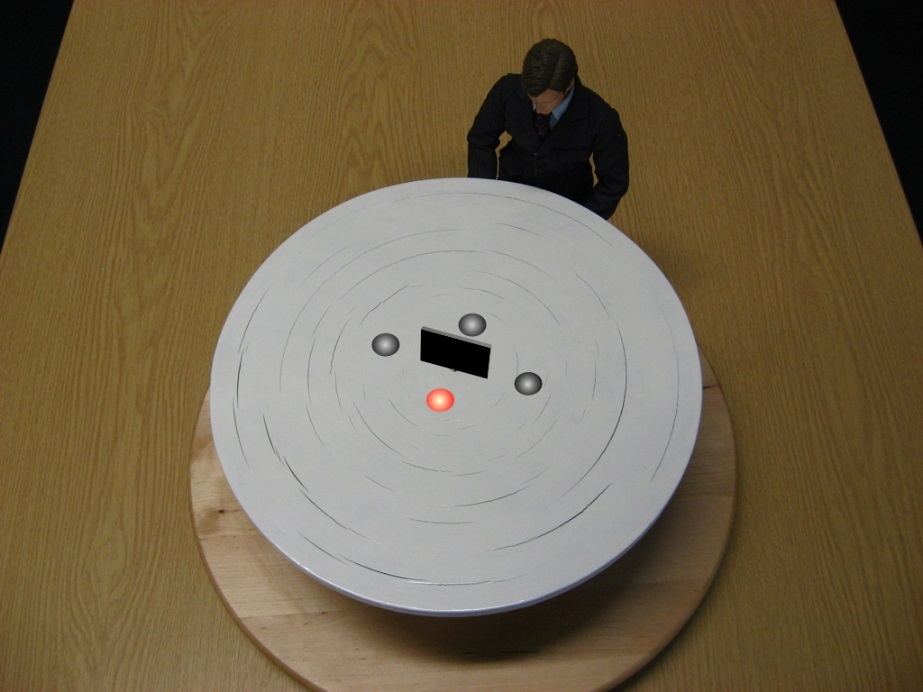
**Pre-processing more complex datasets with Fieldtrip**

For simpler preprocessing (one file, one trigger) please refer to the other preprocessing tutorial.

Often you will need to preprocess data from multiple runs and/or containing multiple triggers. This tutorial will address these steps using the Fieldtrip toolbox.

In this example I will be analysing data from my perspective taking task which has been organised into 3 separate runs. Post-acquisition, the data has been Maxfiltered and transformed into a common headspace using the –trans option of Maxfilter.



Run 1

Break

Run 2

Break

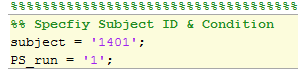
Run 3

So we start with three files:

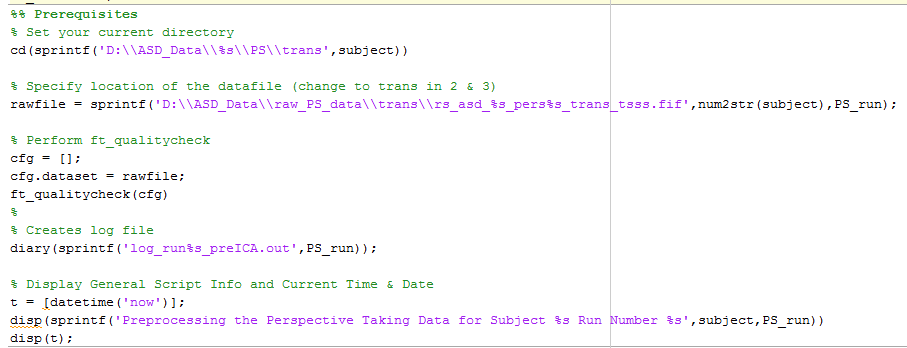
rs\_asd\_1401\_pers1\_trans\_tsss.fif rs\_asd\_1401\_pers2\_trans\_tsss.fif

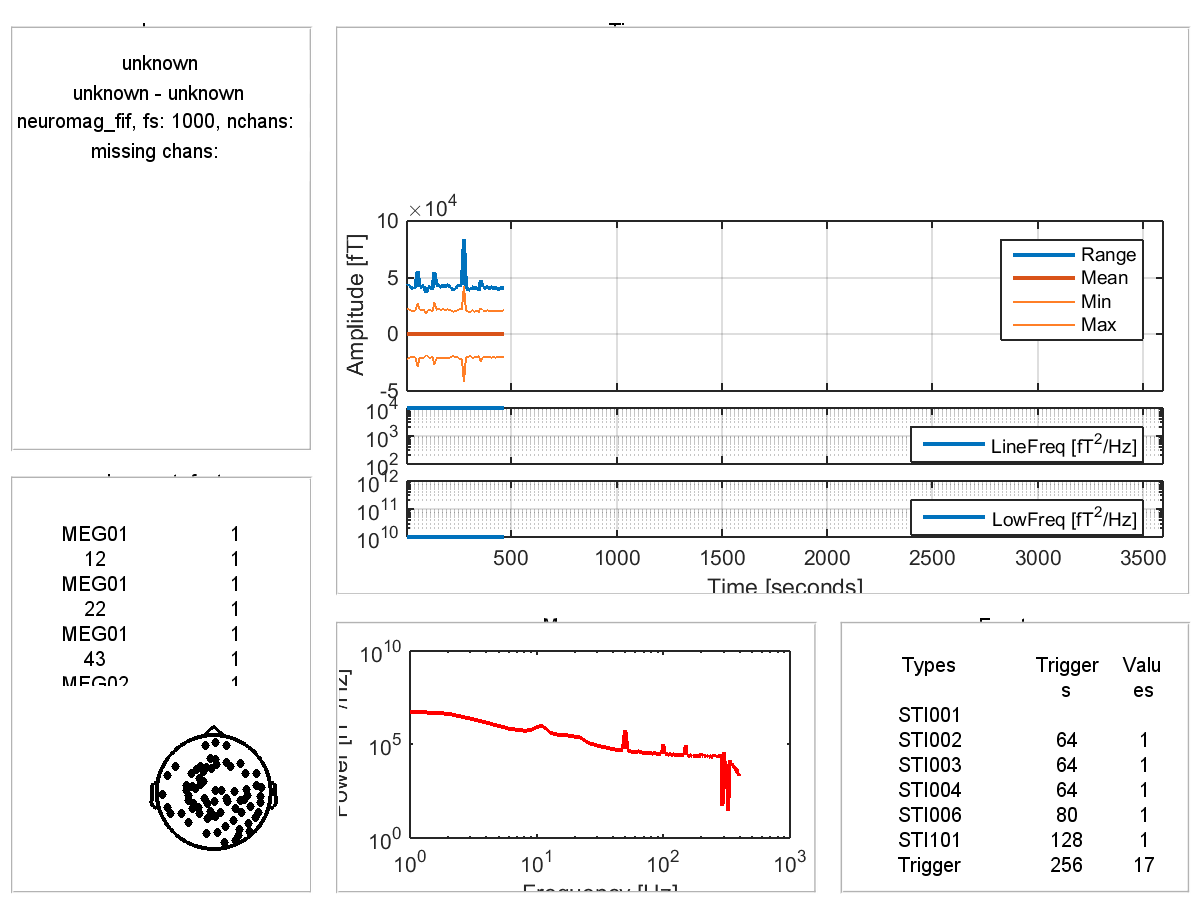
rs\_asd\_1401\_pers3\_trans\_tsss.fif

First we specify the subject ID & condition which will be used throughout the script



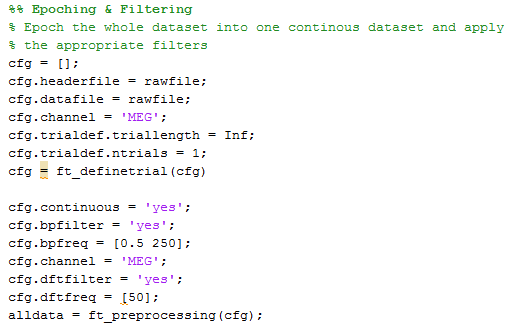
Next we specify the output directory and load in the Maxfiltered MEG data (in this case rs\_asd\_1401\_pers1\_trans\_tsss.fif). This is followed by ft\_qualitycheck. A log file is also created to keep track of the progress of preprocessing for later reference.





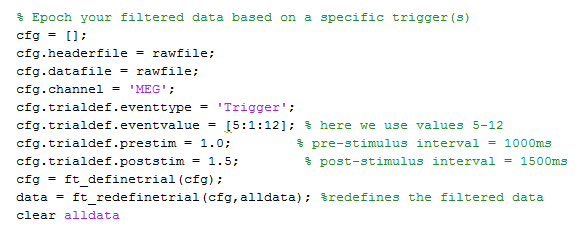
*Output of ft\_qualitycheck*

Here we load in the continuous MEG data and apply filters at this stage (to avoid edge artefacts)

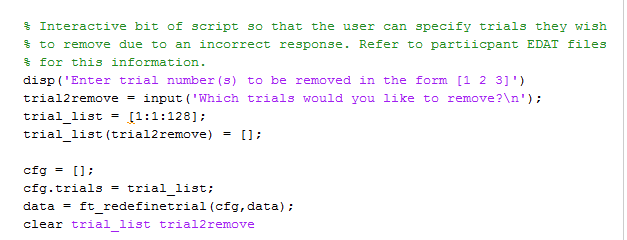


This is followed by a bandstop filter to remove residual 50/100Hz line noise within the data.

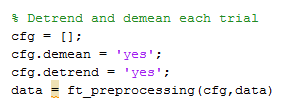
Next, we epoch the continuous data based on specific triger values. Here we are using trigger values 5,6,7,8,9,10,11,12 and extracting 1000ms of data pre-trigger and 1500ms post-trigger. These settings can be easily changed for your data.



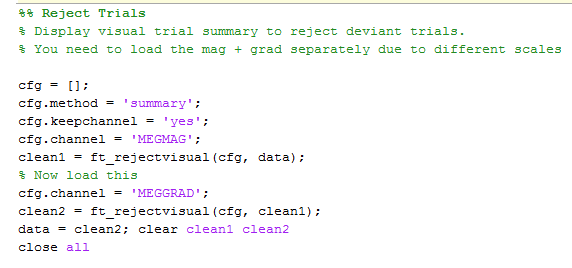
For this experiment we only want to analyse the correct trials and therefore this bit of script allows the user to specify the trial number(s) to remove. I get this information from an EDAT file created after the participant has completed each run.



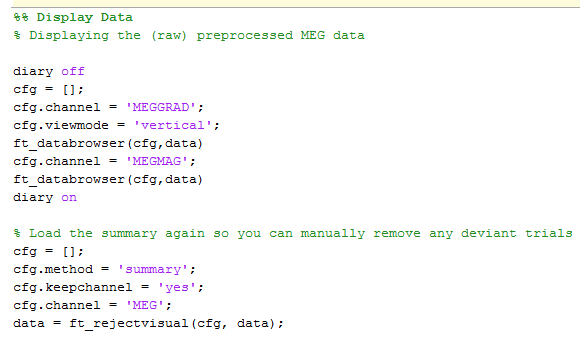
Next we demean and detrend each trial.

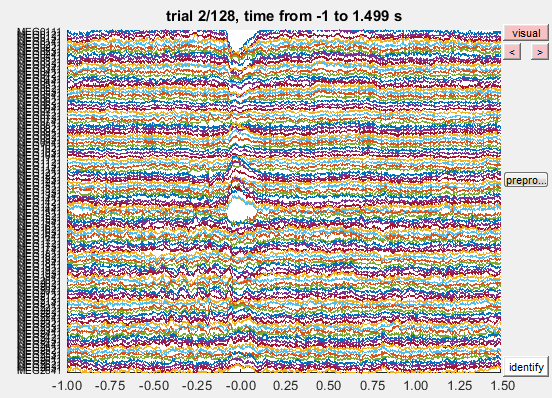
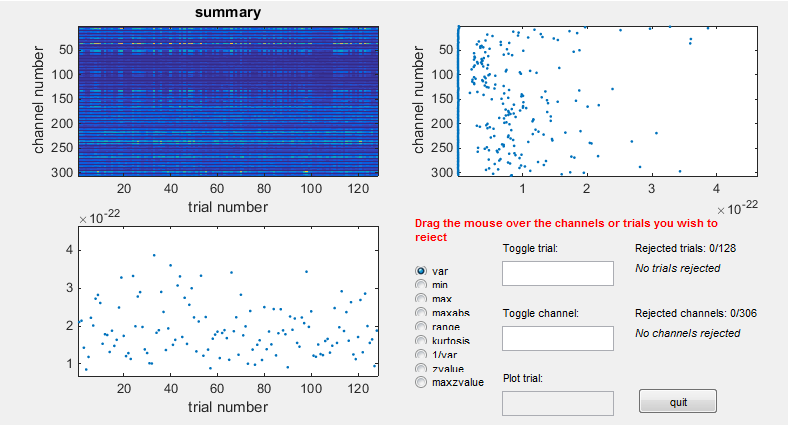


Next, we reject trials which show large deviations in variance (I don’t tend to reject too many trials at this stage as you don’t know when in the trial the deviations come from).



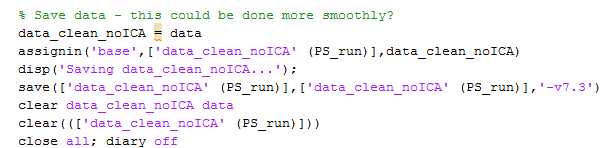
Next we manually scroll through each trial (I tend to focus on the magnetometers) and add deviant trials into the summary window box. Output = variable called data.





Add trials here

We then save this ‘clean’ data as data\_clean\_noICA1 (or 2 or 3 depending on the run number).



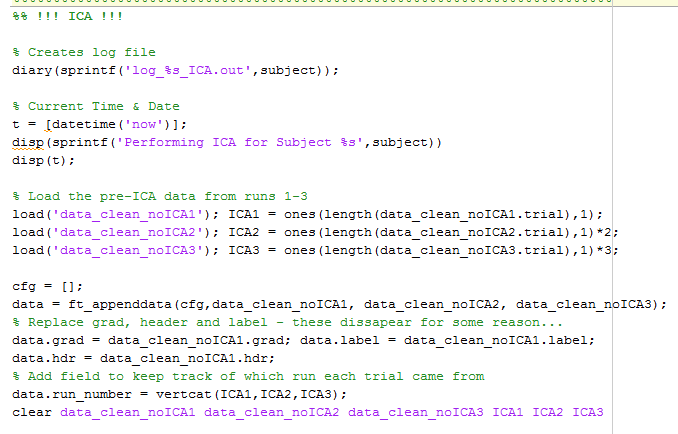
Here I STOP and repeat the above process for runs 2 & 3 (by changing the variable PS\_run)… You could put the script into a loop, but I like to have control to inspect results after each run.

You end up with the saved variables:

data\_clean\_noICA1 data\_clean\_noICA2 data\_clean\_noICA3

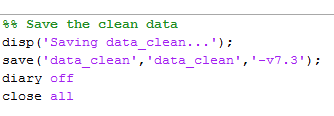
**ICA**

For accuracy we should do ICA on ALL the data. First we create a log file with time & data information…. We then load in all three files and append using ft\_appenddata. There are various extra bits of script to preserve bookkeeping info and keep track of which trials belong to which runs. The variable ‘data’ is the final result of all three variables being concatenated.

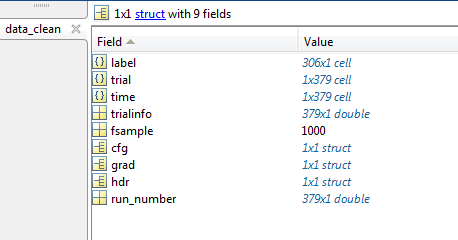


We then run ICA in exactly the same way as for the other tutorial – downsampling, using FASTICA, visualising the components, decomposing the original data as it was prior to downsampling and finally removing the components.

Finally we save the data:



The final final output should look something like this:



**Splitting your data into trials**

The final step is to split your clean data into your conditions of interest. In this experiment there are 4 conditions

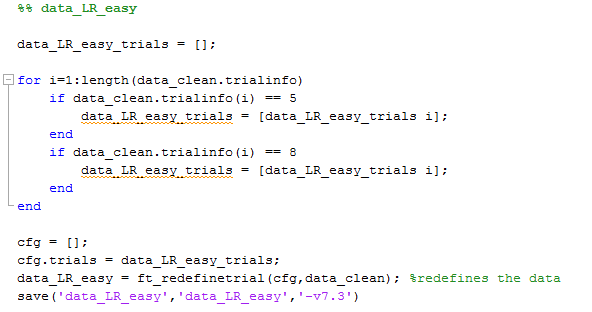
1) Left/Right Easy (60deg/300deg) --> data\_LR\_easy (trigger values: 5,8)

2) Left/Right Hard (160deg/200deg) --> data\_LR\_hard (trigger values: 6,7)

3) Visual/Occluded Easy (60deg/300deg) --> data\_VO\_easy (trigger values: 9,12)

4) Visual/Occluded Hard (160deg/200deg) data\_VO\_hard (trigger values: 10,11)

To we first load in data\_clean.mat. The trigger value information is stored in data\_clean.trialinfo. We can therefore apply the following code to extract the trial belonging to each condition (e.g. see below for data\_LR\_easy).



Repeat for other conditions…

Any questions please email me at: [seymourr@aston.ac.uk](mailto:seymourr@aston.ac.uk)