Assignment 2 - Looking at oscillatory power

Marieke van Vugt

April 27, 2022

In this assignment, you will learn how you can compute oscillatory power and synchronization measures for your dataset. You will use the same data that we used in the previous assignment. If you do not have those data, you can download mine from `https://www.dropbox.com/s/v1rhs2grvx3negz/datForAnal.mat?dl=0` as before. This dataset is called `hpdat`, and it is the data from the first assignment after all the preprocessing steps (i.e., after high-pass filtering). The vectors `highCohTrials` and `lowCohTrials` contain indicators for whether a particular trial a high- or low-coherence trial, respectively (recall that high-coherence here refers to trials in which many dots move in the same direction, hence creating an easy condition, whereas low-coherence trials has dots move more randomly, creating a difficult condition).

1 Oscillatory power

First you will examine whether there is a difference in oscillatory power between the easy and difficult (low- and high-coherence dot motion) trials. This is the oscillatory analogue of the ERPs you looked at in assignment 1.

(a) To perform this oscillatory analysis, you will need the Fieldtrip function `ft_freqanalysis`. Look at the frequencies between 4 and 50 Hz, with the following code:

```
cfg = [];
cfg.method='mtmconvol';
cfg.taper = 'dpss';
cfg.tapsmofrq = 2;
cfg.output = 'pow';
cfg.trials = highCohTrials';
cfg.keeptrials = 'yes';
cfg.foi = [4:50];
cfg.t_ftimwin = ones(length(cfg.foi),1).*0.5;
cfg.toi = -0.5:0.05:1;
TFRhigh = ft_freqanalysis(cfg,hpdat);
```

Explain what analysis you are doing with the above code and what the different parameters mean. The **Fieldtrip for Dummies** guide gives a bit of background on this. See also `http://www.fieldtriptoolbox.org/tutorial/timefrequencyanalysis` for more information.

(b) Now run the code. You can ignore all the warnings. Note that I have taken here the subset of trials that correspond to high coherence dot motion, and given that the name highCohTrials. Be sure to repeat the analysis for low-coherence dot motion trials. Also note that there are many different algorithms for computing oscillatory power, and we just use one.

(c) There are various ways to inspect the results of this analysis. The main question to consider is whether there are frequencies and time periods in which oscillatory power is different between low- and high coherence (difficult and easy) trials. To help you with this, you can look at all channels at the same time with `ft_multiplotTFR(cfg,TFRhigh)` (here `TFRhigh` is the output of `ft_freqanalysis` for the high-coherence trials. Note that `ft_multiplotTFR` does not allow you to automatically subtract low- and high-coherence spectrograms (as you can in `ft_multiplot`). Use code like this:

```
cfg = [];
TFRlowMean = ft_freqdescriptives(cfg,TFRlow);
TFRhighMean = ft_freqdescriptives(cfg,TFRhigh);
TFRdiff  = TFRlowMean;
TFRdiff.powspctrm = TFRlowMean.powspctrm-TFRhighMean.powspctrm;
cfg = [];
cfg.baseline     = [-0.5 -0.1];
cfg.elec = TFRhigh.elec;
cfg.baselinetype = 'absolute';
cfg.showlabels   = 'yes';
cfg.showoutline = 'yes';
ft_multiplotTFR(cfg, TFRdiff);
```

Explain first what this code does, and then what you see. In other words: report at what parts of the brain and during what time periods that are possible differences between easy and more difficult trials (to be confirmed by statistics). What do we learn about how the brain makes decisions from this?

(d) Just like for the ERPs, you can zoom in a bit on an individual channel to see the patterns a bit better. It is also possible to plot only one channel at a time. Let's try this for Cz (because that is everyone's favourite channel). You can use code like this:

```
cfg = [];
```

```
cfg.baseline     = [-0.5 -0.1];
cfg.baselinetype = 'absolute';
cfg.channel      = 'CZ';
figure
ft_singleplotTFR(cfg, TFRdiff);
```

What do you observe? Based on your multiplot in the previous question, is Cz the best channel to look at? Why, or why not?

(e) Then make a topographical plot. You can decide on your time and frequency window of interest, but one option would be to look at the end-of-trial theta frequency, e.g., `cfg.xlim=[0.4 0.6]; cfg.ylim=[5 10]`. Note that you have to specify the electrode topography in your cfg variable: `cfg.elec = TFRhigh.elec` (where `TFRhigh` is the output I obtained from `ft_freqanalysis`. Sample code is:

```
cfg = [];
cfg.baseline     = [-0.5 -0.1];
cfg.baselinetype = 'absolute';
cfg.xlim         = [0.4 0.6];
cfg.elec = TFRhigh.elec;
cfg.ylim         = [5 10];
cfg.showlabels   = 'markers';
ft_topoplotTFR(cfg, TFRdiff);
```

Interpret the results. Also mention why you chose the time window you used.

(f) Next examine whether there are any significant clusters of oscillatory activity at 45.3333 Hz. Note that just as in real data, there may or may not be a significant difference between the conditions. Use `ft_freqstatistics` to answer this question, where we'll use the cluster statistics we used before with the event-related potentials. We use the latency [0 0.7], because this is the period between the stimulus and the average response time. This means you'll use a configuration like this:

```
cfg = [];
cfg.channel = 'all';
cfg.latency = [0 0.7];
cfg.frequency = [45.3333 45.3333];
cfg.method = 'montecarlo';
cfg.statistic = 'indepsamplesT';
cfg.correctm = 'cluster';
cfg.clusteralpha = 0.05;
cfg.clusterstatistic = 'maxsum';
cfg.minnbchan = 1;
```

```
cfg.alpha = 0.05;
cfg.numrandomization = 100;
cfg.neighbours = neighbours;
design = zeros(1,size(TFRhigh.powspctrm,1) + size(TFRlow.powspctrm,1));
design(1,1:size(TFRhigh.powspctrm,1)) = 1;
design(1,(size(TFRhigh.powspctrm,1)+1):(size(TFRlow.powspctrm,1)+
size(TFRhigh.powspctrm,1))) = 2;
cfg.design           = design;
cfg.ivar             = 1;
oneFstat = ft_freqstatistics(cfg,TFRhigh,TFRlow);
```

This is quite a lot of code. **Explain what is happening here.** I.e., what analysis are we doing? How do the statistics work? You can find out more about doing statistics on oscillatory power: `http://www.fieldtriptoolbox.org/tutorial/cluster_permutation_freq` and also in the video mini-lectures.

(g) Then we come to the most rewarding part: plotting the data so we can actually look at the clusters that show a significant difference between low and high coherence (if there are any). You can plot the data with `ft_clusterplot`. Use the following cfg settings:

```
cfg = [];
cfg.alpha = .1;
cfg.zparam = 'prob';
cfg.elec = TFRhigh.elec;
ft_clusterplot(cfg,oneFstat);
```

First explain what the settings of the plotting function mean. Then describe what you can conclude from these plots. Remember that it is possible that you do not get any plots, which depends on the results of your statistical test. In that case you can explain why you do not see any plots.

2 Functional Connectivity

Probably theoretically a lot more interesting is the question of whether there is any functional connectivity between regions. There exist many methods for computing functional connectivity (also explained in the mini lectures). Here, we will look at a few of them.

(a) First, we look at coherence between channels, comparing low and high coherence. **Describe what coherence means. In other words: how is it defined and what can you learn about the brain by looking at coherence?**

(b) To look at any measure of connectivity in Fieldtrip, you need to use `ft_connectivityanalysis`, which is followed by `ft_connectivityplot` to visualize the results. As an input to this analysis create new frequency-transformed data because coherence plots only work with Fourier-transformed data. We only look at a subset of the channels so we can actually make some sense of it. **Why did we choose the set of channels in the `cfg.channel` variable below?** (refer to the "layout of channels" on Nestor if necessary)

```
cfg            = [];
cfg.output     = 'fourier';
cfg.method     = 'mtmfft';
cfg.foilim     = [4 48];
cfg.tapsmofrq  = 4;
cfg.keeptrials = 'yes';
cfg.channel    = {'F1' 'F2' 'CZ' 'C5' 'C6' 'PZ' 'O1' 'O2' };
cfg.trials = highCohTrials';
fftHigh           = ft_freqanalysis(cfg,hpdat);
cfg.trials = lowCohTrials';
fftLow           = ft_freqanalysis(cfg,hpdat);
```

(c) Then use the following commands to compute coherence:

```
cfg           = [];
cfg.method    = 'coh';
cohHigh   = ft_connectivityanalysis(cfg, fftHigh);
cohLow           = ft_connectivityanalysis(cfg,fftLow);
```

You can plot the resulting data with the following commands (where you may want to change `cfg.foi` based on your examination of the data with `ft_connectivityplot`).

```
cfg           = [];
cfg.parameter = 'cohspctrm';
ft_connectivityplot(cfg,cohLow,cohHigh);

cfg = [];
cfg.elec = hpdat.elec;
lay = ft_prepare_layout(cfg,hpdat);
cfg = [];
cfg.layout = lay;
cfg.foi = 8;
ft_topoplotCC(cfg,cohHigh); colorbar
ft_topoplotCC(cfg,cohLow); colorbar
```

What do you observe? Are there differences in coherence between the low- and high-coherence trials? What could these results tell

us about how decision making works in the brain?

(Unfortunately, we cannot compute statistics on coherence, because we would need multiple participants for that.)

(d) Another way to look at connectivity are autoregressive methods such as Granger causality. The advantage of Granger causality is that it tells you something about the direction of connectivity. Basically, Granger causality indicates how much information you gain for predicting the future of a particular channel by adding information from the past of another channel. If this other channel adds a lot of information, it is said to "granger-cause" the current channel. Various measures of Granger causality are included in Fieldtrip. To be able to do Granger causality, you first have to fit a multivariate-autoregressive model (basically letting each channel predict its own future) with `ft_mvaranalysis`. Follow the commands below:

```
cfg = [];
cfg.trials = find(lowCohTrials);
hpdatLow = ft_selectdata(cfg,hpdat);
cfg.trials = find(highCohTrials);
hpdatHigh = ft_selectdata(cfg,hpdat);
cfg = [];
cfg.order = 5;
cfg.method = 'bsmart';
mdataLow = ft_mvaranalysis(cfg,hpdatLow);
mdataHigh = ft_mvaranalysis(cfg,hpdatHigh);
```

Now we have the multivariate autoregressive coefficients for both datasets. **Explain what multivariate autoregressive coefficients are.** See also `http://www.fieldtriptoolbox.org/tutorial/connectivity` for more information about these analyses and `https://online.stat.psu.edu/stat510/lesson/11/11.2` for more information about multivariate autoregressive models in general.

(e) The next step is to transform these data into the frequency domain:

```
cfg        = [];
cfg.method = 'mvar';
cfg.channel    = {'F1' 'F2' 'CZ' 'C5' 'C6' 'PZ' 'O1' 'O2' };
mfreqLow      = ft_freqanalysis(cfg, mdataLow);
mfreqHigh      = ft_freqanalysis(cfg, mdataHigh);
```

Now we are finally able to compute Granger causality between the channels of the group we have looked at before.

```
cfg           = [];
```

```
cfg.method    = 'granger';
cfg.channel    = {'F1' 'F2' 'CZ' 'C5' 'C6' 'PZ' 'O1' 'O2' };
grangerLow       = ft_connectivityanalysis(cfg, mfreqLow);
grangerHigh       = ft_connectivityanalysis(cfg, mfreqHigh);
```

We can make similar connectivity plots to what we've done for coherence. **What is the theoretical difference between Granger causality and coherence?**

```
cfg           = [];
cfg.parameter = 'grangerspctrm';
ft_connectivityplot(cfg, grangerLow,grangerHigh);
```

(f) **What do you observe in these graphs in comparison to the connectivity measured by coherence?** Are there any things you expect based on what you know the differences are between coherence and Granger causality? What can we learn from this analysis about decision making?

3 Where is the model-based analysis? –revisited

As before, can you say what you would look at in a model-based analysis of these data? What kind of things should your model predict if you want to use these techniques for a model-based data analysis? And what can you do with the methods we studied in this assignment that you could not do with the methods we studied in the previous assignment? The best way to answer this question is to choose a model from the models we have discussed in class, and to think about how you could look at its oscillatory correlates. What questions would you be able to answer by looking at oscillations, which you would not, for example, by looking at ERPs?