Assignment 1 - Becoming familiar with EEG analysis

Marieke van Vugt

April 15, 2022

In this assignment, you will become familiar with EEG and EEG analysis. We will use Fieldtrip software, a Matlab toolbox, which can be downloaded from `www.fieldtriptoolbox.org` (note that it is possible to use fieldtrip-lite for this assignment). If you have trouble downloading Fieldtrip from this location, you can also go to its github page `https://github.com/fieldtrip/fieldtrip/` and click on the "clone or download" button. Once you have downloaded it, you can install it by going to the fieldtrip directory on your computer and typing `ft_defaults`.

Note: On the UWP, you may not have the permissions to do the full installation of Fieldtrip. In that case, use the Matlab command addpath to add the path to fieldtrip to matlab. If you are on a Mac, you may need to tell it that the compiled Fieldtrip files are fine using the instructions in `https://www.fieldtriptoolbox.org/faq/mexmaci64_cannot_be_opened_because_the_developer_cannot_be_verified/`.

The sample dataset we will use is on Nestor under Assignments → Assignment 1. These data come from a participant who does a decision making task, similar to the one discussed in the lecture. They saw field of randomly moving dots of which a subset moved coherently to left or right, and they had to indicate by button press whether dots moved left or right. Every correct response was rewarded with a penny. They were asked to make as much money as they could in blocks of 4 minutes long. Difficulty was varied such that in some blocks, they would perform approximately 70% correct ("low coherence"–in this case 3%), and in other blocks they would perform approximately 90% correct ("high coherence"–in this case 8%). This difficulty could easily be manipulated by the fraction of dots that moved coherently in the relevant direction.

Our main interest in that experiment was the neural dynamics of evidence accumulation leading up to the decision, but in this assignment we will be focusing on the neural correlates of difficulty, which is easier to assess with standard methods for EEG analysis.

To prepare the EEG data for analysis, you first have to remove artifacts. The *Fieldtrip for Dummies* guide (Introduction to Fieldtrip) that you can find on Nestor under Assignment 1 has more explanation on this process, and it will also be discussed in some of the prerecorded video lectures which you can find

under Week 1 on Nestor.

1 Artifact Removal

- Load the data file and save the config variable for backup (`load('subsetData.mat'); oldcfg = cfg; lay = cfg.layout;`)
- Run the databrowser (`cfg = ft_databrowser(cfg,data);`). Be sure to have cfg before the "=" sign, because if you don't, you will lose all the data from artifacts you have collected (see next point).
- Now walk through the trials and mark artifacts that distort almost all channels (i.e., not the trials in which only a few channels are distorted). These trials typically look like:
  - High-amplitude excursions
  - High variance

  You mark the artifacts by selecting an arbitrary time interval in that window and then clicking on it. Do not worry about getting the timing right because the algorithm will remove the whole trial. You can inspect artifactual data segments in more detail by clicking your right mouse button, and choosing an option such as **topoplotER** from the menu, which allows you to see its topography). Note that if you reject the majority of your data, you're doing something wrong. You want to just mark bits of EEG that are bad for a substantial number of channels, because it will mess up the ICA decomposition that you'll do in a later step. Individual bad channels will be removed then. Aim for at most 30% of data removed. When you are done with the data, you should press "q" to quit the databrowser (see *Fieldtrip for Dummies* tutorial).
  (Q1) What fraction of the trials have artifacts? give a global overview of artifacts you detected together with some pictures to illustrate (use screenshots or matlab's saveas)
- Actually reject the artifacts you spotted:

```
cfg.artfctdef.reject = 'complete';
cleandata = ft_rejectartifact(cfg,data);
```

- You can then use ICA on the cleaned-up dataset in the next step of artifact removal:

```
cfg = [];
cfg.channel = [2:116]; % remove the first channel (EOG) to ensure ICA
                       % goes well
ic_data =ft_componentanalysis(cfg,cleandata);
```

(Q2) Describe what ICA does in theory and why that makes it suitable to remove artifacts. Inspect these components with the following code (where we chose to look at 30 seconds of data at a time, for 10 components).

```
cfg = [];
cfg.viewmode = 'component';
cfg.continuous = 'yes';
cfg.blocksize = 30;
cfg.channel = [1:10];
cfg.layout = oldcfg.layout;
```

Determine which components are artifactual (report the topography and sample time course for each artifactual component). Repeat this of course for all 115 components (you can switch components easily by clicking on "channel" and changing the selection of channels you're looking at). Then look at the resulting components in the databrowser: `ft_databrowser(cfg,ic_data)`. You can recognize bad ICA components based on a combination of time course (blinks, muscle activity or other non-EEG-like activity) and topography (look for small, isolated topographies). Look for bad ICA components that are consistently bad over the course of the whole session. Note that there is not always a consensus about what components are 'bad', so for grading we will be more interested in your argumentation about why the components are bad than your exact choice of components.
(Q3) What fraction of components do you remove? (again, be sure to not remove the majority of your data; aim for at most 20% of the components). Add screenshots of the components you remove.

- Once identified, actually remove the suspicious components with the following code:

```
cfg = [];
cfg.component = [1 2 4];
data_iccleaned = ft_rejectcomponent(cfg,ic_data);
```

(of course you should replace the numbers with the components you think should be removed)

- **(Q4) Then we look for artifacts again in the same way as before. What differences do you notice? Illustrate this with a screen shot from the databrowser.**

- **(Q5) Study the other methods for artifact rejection in the Fieldtrip toolbox using the Fieldtrip website. Write down what methods are contained in Fieldtrip, including a brief description of**

what each one does and try one of these (not one that you have already used earlier in this assignment). Describe what you observe.

2 Filtering

In addition to removing artifacts from individual trials, EEG data is typically filtered in various ways to remove contamination by the electrical system.

1. Use a notch-filter: `ft_preprocessing` where

   ```
   cfg.bsfilter = 'yes';
   cfg.bsfreq = [59 61];
   ```

   (Q6) Can you explain what you have done there? What do you observe in the data (illustrate your answer with graphics).

2. Then add a high-pass filter: `cfg.hpfilter = 'yes'; cfg.hpfreq = [0.2];`. **Why would one use a high-pass filter? Illustrate the effect of this filter on the data with a plot.**

3 ERP analysis

Now we are finally in a position to do actual data analysis on this dataset. We will use a larger dataset for that, which I have already cleaned for you: `datForAnal.mat` (which contains a variable called `hpdat`). You can find the link to this dataset on Nestor. We will use `ft_timelockanalysis.m` to compute event-related averages. You are asked to plot an ERP first for all trials together, then separately for low- and high-coherence trials, so that you can inspect the effect of task difficulty on the ERP. To separate the data into low- and high-coherence trials, you should make use of the information about coherence that can be found in the second column of the trialinfo field of your data, which gives for every trial the percentage of dots coherence. You will want to get a vector of ones and zeros for whether a trial is high-coherence or not (which in the code below I call 'highCohTrials'), and a similar vector for whether a trial is low-coherence or not. Note that in some versions of Fieldtrip you have to replace cfg.trials by cfg.trial.

Use code like this (adapt this yourself for the low-coherence trials, and note the transpose of the vector: the vector with trials should have dimensions 1xN, where N is the number of trials in your cleaned-up data–called hpdat in the example below)):

```
cfg = [];
cfg.channel = 'all';
cfg.trials = highCohTrials'; % note that the ' is a transpose
% (not a typo)
```

```
cfg.keeptrials = 'yes'; % necesaary for the statistics to work
timelockHigh = ft_timelockanalysis(cfg,hpdat);
```

You can plot the ERP for a single channel with `ft_singleplotER`, for all channels simultaneously with `ft_multiplotER`, and create a topography with `ft_topoplotER`.

1. Draw the plots for channel Cz:

   ```
   cfg = [];
   cfg.xlim = [-0.2 1.0];
   cfg.ylim = [-3.5 3];
   cfg.channel = 'CZ';
   cfg.fontsize = 18;
   cfg.linewidth = 2;
   ft_singleplotER(cfg,timelockHigh,timelockLow);
   ```

   (Q7) What do you see here? A narrative of such a plot would involve something like: "in channel Cz, there is a difference between the easy and more difficult decisions from 100-300 ms. For this period, there is higher EEG activity for more difficult decisions."

2. then all channels (`ft_multiplotER`)

   ```
   cfg.channel = 'all';
   cfg.layout = lay;
   ft_multiplotER(cfg,timelockHigh,timelockLow);
   ```

   (Q8) Show the multiplot and describe the results. A narrative would include a description of the patterns of the waveform and especially differences between conditions across different parts of the brain.

   Based on the multiplot, you can select a time period of interest, which typically shows a difference in activity between conditions, and assess that topography more precisely using a topography plot:

3. Make a topography (`ft_topoplotER`). You can choose a time interval by setting **xlim** variable in the **cfg**.

   ```
   cfg = [];
   cfg.xlim = [0.1 0.2];
   cfg.layout = lay;
   figure; ft_topoplotER(cfg,timelockHigh,timelockLow)
   ```

   (Q9) Describe the topoplot results and argue why you chose the time interval you used. (hint: look at the graphs obtained from the `multiplotER` for times where the two conditions differ from each other).

(see `http://www.fieldtriptoolbox.org/tutorial/plotting` for more information about plotting). In general, the help for every one of these fieldtrip functions can also explain a lot about what the different options are.

In EEG data, the signal tends to sometimes drift over time, which is a property of the equipment and has nothing to do with brain function. To correct for this artifactual signal, researchers will take the average activity of a neutral period of time, e.g., just before the start of the trial, and subtract the average signal in this period from the EEG. This ensures that an ERP will be centered around a value of 0 in the period before zero, and more generally this will remove noise from your data. You can subtract a baseline from the signal by setting `cfg.baseline = [-.1 0]` in `ft_singleplotER` (in which we subtract the average signal in the interval -0.1-0 seconds from the EEG).

4. **(Q10) What baseline would you choose? A baseline in general is around 100-200 ms duration. Demonstrate the effect of a baseline correction by using the ft_singleplotER.m function.**

4 Statistics

The last part of this assignment concerns statistics: when are the patterns for low and high coherence different from each other? To assess this, you'll need to use the function `ft_timelockstatistics`, followed by `ft_clusterplot` to correct for multiple comparisons. You should set the method to be 'montecarlo'. What is crucial (and not well-documented in the Fieldtrip documentation) is setting the design correctly. The design in this case is basically within-subject, where you want to specify that the first set of trials comes from one condition and the second set of trials from the other:

```
design = [ones(1,size(timelockHigh.trial,1)) 2*ones(1,size(timelockLow.trial,1))];
cfg.design = design;
cfg.ivar = 1;
```

A very cool method for correcting for multiple comparisons is the cluster-method. You can use this method with `cfg.correctm='cluster'`. Yet to be able to use this method, you will also need to define the neighbouring channels with the function `ft_prepare_neighbours`.

```
cfg = [];
cfg.neighbourdist = 3.5;
cfg.method = 'distance';
cfg.elec = oldcfg.elec;
neighbours = ft_prepare_neighbours(cfg,timelockHigh);
```

Now to run cluster statistics, use the following `cfg`:

```
cfg = [];
cfg.method = 'montecarlo';
cfg.parameter = 'trial';
cfg.numrandomization = 200;
cfg.correctm = 'cluster';
cfg.alpha = 0.05;
cfg.tail = 0;
cfg.statistic = 'indepsamplesT';
cfg.clusterstatistic = 'maxsum';
cfg.clusterthreshold = 'parametric';
cfg.clusteralpha = 0.05;
cfg.neighbours = neighbours;
cfg.design = design;
cfg.ivar = 1;
stat = ft_timelockstatistics(cfg,timelockHigh,timelockLow);
```

1. **(Q11) Explain how the cluster statistics work and how the above code generates these statistics.**

   After having run all of that, you can visualize your results with `ft_clusterplot`. (see also `http://www.fieldtriptoolbox.org/tutorial/cluster_permutation_timelock`). To use the cluster plot, use the following configuration:

   ```
   cfg = [];
   cfg.alpha = 0.05;
   cfg.parameter = 'prob';
   cfg.layout = lay;
   ft_clusterplot(cfg,stat);
   ```

2. **(Q12) Take the reader through the results: what do you see and what does/could this mean? Note that it is possible you will get an error when there are no significant clusters. In that case, report the non-significance, including the magnitude of the lowest p-value.**

5 How could one incorporate modeling?

(Q13) You have learned a lot about event-related averaging of EEG data in this assignment. Choose one of the models we have talked about in the course so far, and come up with a model prediction that you could test using ERPs. You do not actually need to simulate the model, but just think about the type of model output that could be used to examine predictions for ERPs, and the results that you would expect to obtain if your hypothesis is correct. Be as specific as possible about how you would go about testing the model predicts and what would be the expected results (in terms of ERP, topography etc). This is your chance to be creative!