FaPra: Methods in Computational EEG

Overview:

I am interested in developing computational methods for EEG analysis. These methods have very broad applications, enabling accurate EEG analyses from simple laboratory experiments, to complex real-world or VR EEG studies.

Recently, the Julia programming language gained lots of traction due to its well-designed syntax, integrated reproducible packet-management and because it appears to solve the 2-language problem (Julia is as easy as python, but (nearly) as fast as C). Due to the computational complexity, but also the necessary scientific flexibility, many researchers switch to Julia. Therefore, all implementations in this FaPra will be done in Julia.

I propose four projects for 2-3 students each. Students will work on the problems under my supervision.

I will meet with each group for 30min / week in a structured meeting if necessary, to discuss issues and provide help. In addition, we can discuss issues via email/github issues.

In addition, we will all meet every two weeks for a group-discussion, updating each other on the progress being made. This will involve short presentations of each group of 10min each.

The grade will consist of:

- 20% Presentation Skills(rhetorics, slide quality, Q&A)
- 50% Implementation (Code & Documentation)
- 30% Report (quality, figures, structure, writing; up to 8 pages)

The report should be structured as a typical conference paper.

Project 1) Comparing statistical models for EEG data analysis

You will work on a comparison of statistical power between Linear Mixed Models, the classical two-stage approach and a meta-study approach.

Mandatory goals:

- Extract simulation-parameters from EEG meta-reviews (i.e. within/between subject variability, noise, n-trials etc.).
- Simulate aggregated EEG data for statistical testing
- Implement and improve upon the statistical approaches in Julia
- Compare statistical power between the three approaches

Stretch-Goals:

- Run the same analysis without aggregating time-windows using FDR multiple-comparison correction
- Compare statistical approaches with cluster-permutation multiple comparison correction
- Create an interactive tool to perform power analyses

This research project is important, as there is no consensus on what the best strategy is to analyze EEG data, and often computational issues arise with the more complex methods. This is actual research, so it is not without risk. Therefore, the process will be evaluated and not the results.

The project is sized for a group of 2-3 students.

20% domain-Knowledge, 5% literature-research, 20% thinking-hard, 45% implementation in Julia, 20% documentation

2) Visualizations for complex encoding models

You will work on an implementing and developing visualizations for mass-univariate and deconvolution models in the Julia programming language.

Mandatory goals:

- Implement the model-visualizations from www.unfoldtoolbox.org in unfold.jl using Makie.jl, AlgebraOfGraphics.jl package and pyMNE.jl. This includes the plots to investigate designmatrices, result-plots as timeseries & topographies

Potential Stretch-Goals:

- Develop plots to help with non-linear spline-visualizations
- Develop an estimated marginal means – interface for unfold
- Reimplement some plotting tools (e.g. topoplots) natively in Makie
- Generate an interactive tool for model-interrogation (potentially relying on other stretch goals)

Visualization is one of the most important tools to analyze data. Complex models, as we are deploying for EEG analysis, need exponentially more visualization than other analyses. Unfortunately, if visualizations are difficult to perform, less will be performed in the first place. This will hurt analysis and understanding.

The project is sized for a group of 2-3 students.

20% Domain-Understanding, 20% thinking hard, 40% Implementation in Julia, 20% documentation/tutorials

3) Marrying Encoding and Decoding Models

There are broadly two categories of EEG analyses: Decoding, (f(brain) = stimulus) and encoding (f(stimulus) = brain). In this project we will try to combine the benefits of both methods based on the analysis-approach of back2back regression, which in some sense encompasses both.

Mandatory goals:

- Simulate a dataset suitable for decoding and encoding analyses
- Apply the back2back regression implementation in unfold.jl
- Apply back2back regression to a real EEG/ET dataset

Stretch-Goals:

- Substitute the back2back regression algorithms from ridge regression to potentially more powerful algorithms e.g. SVMs
- Systematically vary different parameters of the simulation, e.g. correlatedness / “confounding”, amount of data, correlated-error,
- Compare linear and non-linear effects in simulation

This methodology is important, as it potentially allows to combine complex encoding models with powerful multivariate decoding methods.

The project is sized for a group of 2-3 students.

20% Domain-Understanding, 20% thinking hard, 40% Implementation in Julia, 20% documentation

4) An adventure into temporal-basis function land

Typical EEG deconvolution analysis makes use of “structure-free” least-squares deconvolution. But it seems plausible and possible to introduce (empirical) prior information into the model fit. In this project we will introduce prior information on the temporal ERP-response-shape.

Mandatory goals:

- Implement the spline temporal-basis according to www.unfoldtoolbox.org / Ehinger & Dimigen 2019 in unfold.jl (Julia)
- Benchmark fits with and without temporal-basis in terms of computation time & R2-accuracy

Stretch-Goals:

- Develop an PCA-based empirical basis approach (akin to https://doi.org/10.1152/jn.00220.2011)
- This methodology is important, as it potentially allows to combine complex encoding models with powerful multivariate decoding methods.

The project is sized for a group of 2-3 students.

20% Domain-Understanding, 20% thinking hard, 40% Implementation in Julia, 20% documentation