Introducing UnfoldSim.jl: A simulation toolbox for event-related time series

Luis Lips, Judith Schepers[1], Benedikt V. Ehinger[12]

[1] Institute for Visualization and Interactive Systems, University of Stuttgart, Germany
[2] Center for Simulation Science, University of Stuttgart, Germany

Motivation

EEG data are among the most popular brain-related timeseries in neuroscience. Experimental designs as well as the required analysis methods are becoming more complex, but evaluation and validity checks of these methods are lacking, most likely because few simulation tools for such data exist.

Surprisingly, even the most advanced simulation toolboxes do not support core requirements:
- multi-subject simulation
- continuous as well as segmented data simulation
- complex event-related signals, with linear and non-linear covariates

We present UnfoldSim.jl, a JuliaLang based toolbox to simulate continuous-time model-based event-related data, with arbitrary event-responses. The toolbox is fast, modular, and allows the users to easily replace any of the pre-specified modules.

15s-Summary

Motivation: Advanced analysis methods in EEG are becoming more prevalent, but often they are only tested on real data, simulations are rare due to the complexity of the underlying signal generation.

UnfoldSim.jl: A modular JuliaLang toolbox to simulate continuous-time, model-based event-releated timeseries. Support for overlap between events and various noise functions, with a general focus on EEG signals.

Application: Simulations of multi-subject experiments to estimate false-positive rate and power of two popular analysis methods

Simulating event-related potentials

	subj	cond	item
1	S1	A	I1
2	S1	B	I2
3	S2	A	I1
4	S2	B	I2

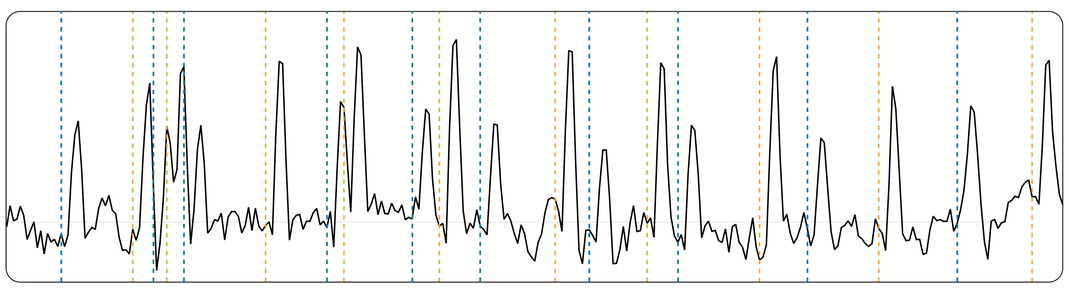
Basis

Formula & Parameters

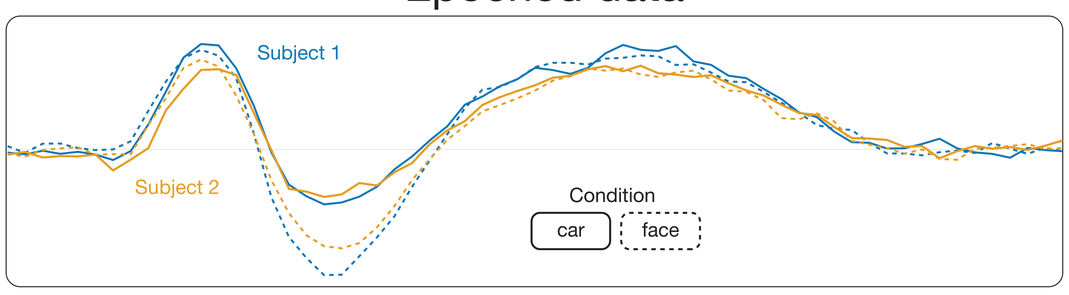
```
0~1+cond+(1+cond|subj)
β = [1.0,0.5]
σs = [0.1,0.2]
```

Noise

Continuous timeseries



Epoched data



Key-Features

Signals

Multiple components can be combined to create more complex arbitrary signals!

Experiment Designs & Parameters

- Single-Subject
- Repeated
- Multi-Subject / Multi-Item Hierarchical
- Custom / User-specified

Overlap

- Uniform
- Log-Normal
- User-specified

Noise

- White
- Pink
- Brown
- AR(1)
- Exponential-AR
- User-specified

Upcoming...

- Multi-channel support via forward modelling
- Diffusion-noise

Code-Snippet

```
design = SingleSubjectDesign(;
  conditions = Dict(:cond=>["A","B"])
) |> d->RepeatDesign(d,5);
```

Define the design / event-table

```
signal = LinearModelComponent(;
  basis = hanning(10),
  formula = @formula(0~1+cond), β = [1,0.5]
);
```

Specify a ground-truth signal & relation to events

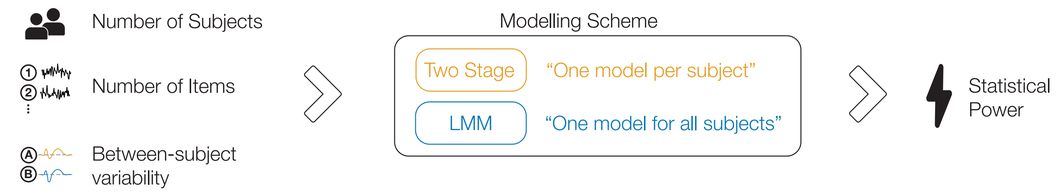
```
data, events = simulate(StableRNG(1),
  design, signal, UniformOnset(width=4),
  PinkNoise(noiselevel=0.0)
);
```

Define onset distribution and noise, and simulate!

Application: Power-Analysis

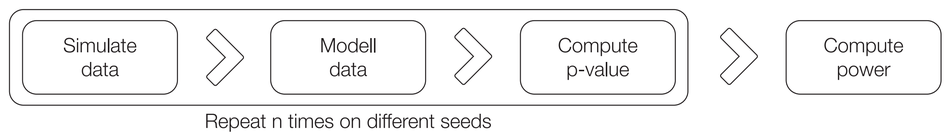
Goal

- Compare the statistical power of selected models with regard to different parameters



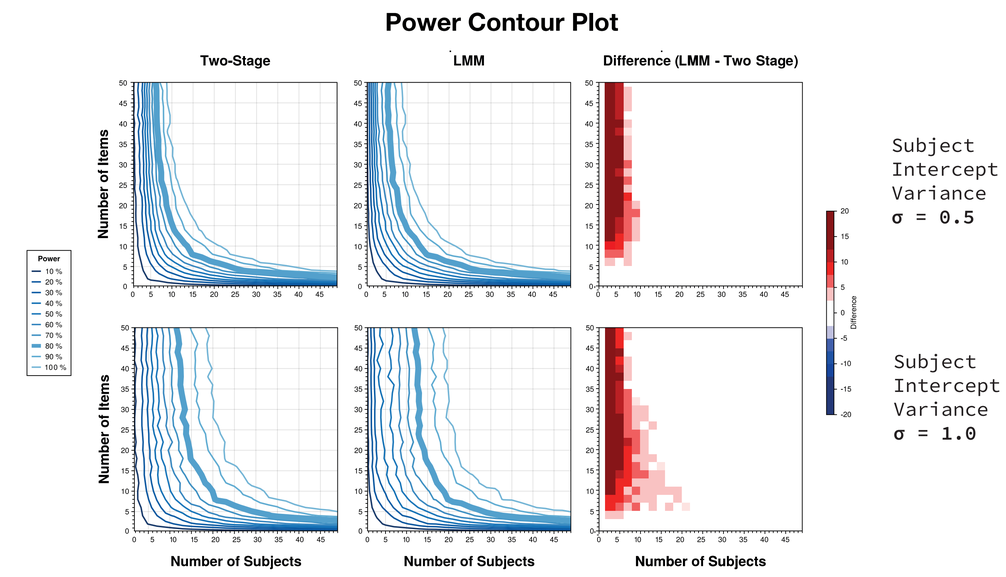
Design

- For the power analysis, we repeatedly simulate data, and check whether we can find a significant effect. Subsequently the statistical power is computed.
- Significant testing via t-test (two-stage) and permutation test (LMM)



Two-Stage vs LMM

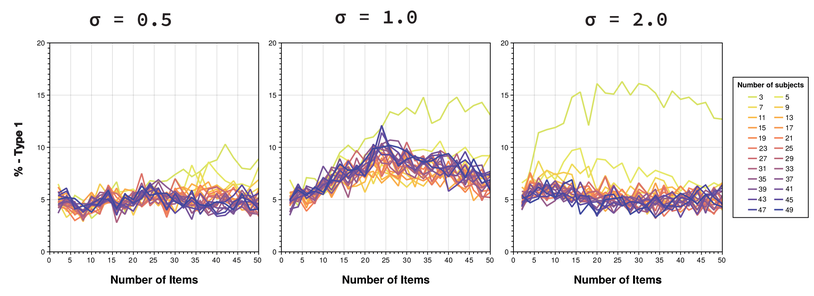
Power Contour Plot



- We observed differences for experiment designs with a small number of subjects

Type 1 Error (LMM Permutation Test)

Subject Intercept Variance



- Increased type-1 error for LMM + permutation test approach
- Observed differences for experiment designs with a small number of subjects likely caused by increased type-1 error
- Preliminary results did not show an significant advantage of the LMMs over the two-stage approach for models without item effects and in balanced data

- Broader parameter space needs to be investigated for a more founded conclusion
- LMMs could outperform the two-stage approach in unbalanced designs and circumstances with high within-item variance