The Journal of Open Source Software

UnfoldMakie.jl: EEG/ERP visualization package

Vladimir Mikheev [1*] and **Benedikt Ehinger** [1,2*]

1 University of Stuttgart, Institute for Visualization and Interactive Systems, Germany **2** University of Stuttgart, Stuttgart Center for Simulation Science, Germany * These authors contributed equally.

DOI: 10.21105/joss.07560

Software

- Review
- Repository
- Archive

Editor: Fabian Scheipl

Reviewers:

- @ferchaure
- @AdamWysokinski

Submitted: 13 November 2024
Published: 09 January 2025

Statement of need

`UnfoldMakie.jl` is a Julia package for visualization of electroencephalography (EEG) data, with a focus on event-related potentials (ERPs) and regression-ERPs (rERPs). `UnfoldMakie.jl` fills a gap as one of the only dedicated EEG visualization libraries and offers ease of use, customization, speed, and detailed documentation. It allows for both explorative analysis (zooming/panning) and export to publication-ready vector graphics. This is achieved via multiple backends of `Makie.jl` (Danisch & Krumbiegel, 2021): vector graphics with `CairoMakie.jl`, GPU-accelerated interactive graphics with `GLMakie.jl` and the browser-based `WGLMakie,jl`.

In an earlier study (Mikheev et al., 2024), we asked novice and expert practitioners for insights into their ERP visualization practices. The results of this survey were used to develop and improve `UnfoldMakie.jl`. Further, `UnfoldMakie.jl` is agnostic (independent) of any specific analysis framework, but it nicely accompanies the rERP analysis package `Unfold.jl` (Ehinger & Dimigen, 2019).

The target audience of this package is anyone working with EEG, MEG, ERP, iEEG or other time-series data.

Functionality

`UnfoldMakie.jl` excels in various fields:

- Focused. `UnfoldMakie.jl` focuses exclusively on visualizing (r)ERPs, unlike other toolboxes such as `EEGLAB`, `FieldTrip`, `Brainstorm`, or `MNE`. This makes it easier to understand, use, and maintain the package in the future.
- Customizable. The majority of EEG researchers perceive the flexibility of coding as the most important feature of the EEG toolbox (Mikheev et al., 2024). Consequently, users of `UnfoldMakie.jl` have great flexibility in customizing most aspects of the plots, such as colors, line styles, and axis decorations.
- Combining plots. The layout system for subplots in `Makie.jl` makes it easy to combine and align various plot types.
- Flexible mapping. With `AlgebraOfGraphics.jl`, it is easy to map conditions, groups, channels, or other features, to a large variety of aesthetics like linestyle, color, marker and others. It works similarly to the popular R package `ggplot2`.
- Flexible data input. All functions support basic `Arrays` and tidy `DataFrames`.
- Fast. Julia and `Makie.jl` allows fast plotting of figures with very complex layouts. As an example, it is able to plot one figure with 50 topoplots in 1.9 seconds (1.6 sec with

The Journal of Open Source Software

DelaunayMesh interpolation), which is ~20 times faster than EEGLAB. Although, the Python-based MNE is faster by one second (Table 1). [1]

- Faster updating. Makie.jl is incredibly fast at updating figures, which is beneficial for developing interactive tools and creating animations. UnfoldMakie.jl can create and save a topoplot gif file with 50 frames in 1.7 times faster than MNE (Table 2).
- Interactive. Several plots in our package have interactive features. They are supported by Observables.jl, which allows for fast data exchange and quick plot updating.
- Scientific color maps. According to our previous study (Mikheev et al., 2024), 40% of EEG researchers are not aware of the issue of perceptually non-uniform color maps. UnfoldMakie.jl uses scientific color maps throughout (Crameri et al., 2020) (Moreland, 2015).
- Documented. There is extensive documentation with many usage examples and docstrings (Hatherly et al., 2024).

Table 1: Table 1. Benchmark for a topoplot series with 50 topolots

Languages	Package	Median speed
MATLAB	EEGLAB	36 sec
Julia	UnfoldMakie	1.998 sec
Python	MNE	0.826 sec

Table 2: Table 2. Benchmark for generating (and saving) a topoplot animation with 50 timepoints. No similar functionality exists in EEGLAB.

Languages	Package	Median speed
Julia	UnfoldMakie (.gif)	5.801 sec
Python	MNE (.gif)	9.494 sec

We currently support nine general EEG plot types (Figure 1) and two Unfold-specific plots: Design matrices and Spline plots.

State of the field

There are dozens of libraries in Python and MATLAB for ERP analysis and visualization. According to a recent survey (Mikheev et al., 2024), most EEG practitioners (82%) have experience with MATLAB-based tools like EEGLAB (Delorme & Makeig, 2004), FieldTrip (Oostenveld et al., 2011), ERPLAB (Lopez-Calderon & Luck, 2014) and Brainstorm (Tadel et al., 2019). The Python-based MNE (Gramfort et al., 2013) (41%) and the commercial software Brain Vision Analyzer (22%) further showed strong popularity. None of these toolboxes focuses particularly on visualizations. Indeed, in terms of specialized EEG visualization toolboxes, we are aware of only two such libraries, both MATLAB-based and both named eegvis (Robbins, 2012) and (Ehinger, 2018).

Few EEG/ERP analysis and/or visualization libraries have been written in Julia. We are aware of NeuroAnalyzer.jl (Wysokiński, 2024), EEGToolkit.jl (Pereyra, 2024), Neuroimaging.jl (Luke, 2021). There are also traces of several abandoned projects. Worth highlighting is PyMNE.jl, a wrapper for the Python-MNE toolbox. (Alday & Arslan, 2024)

[1] Be aware that results of benchmarking can vary each run and depends on a OS, package environment, other processes running on computer etc. Current measurements were done by using BenchmarkTools.jl. (Chen & Revels, 2016)

The Journal of Open Source Software

However, all these packages are focused on the analysis of EEG data, while our package is specialized on the visualization of ERPs and rERPs. This is the gap that `UnfoldMakie.jl` fills.

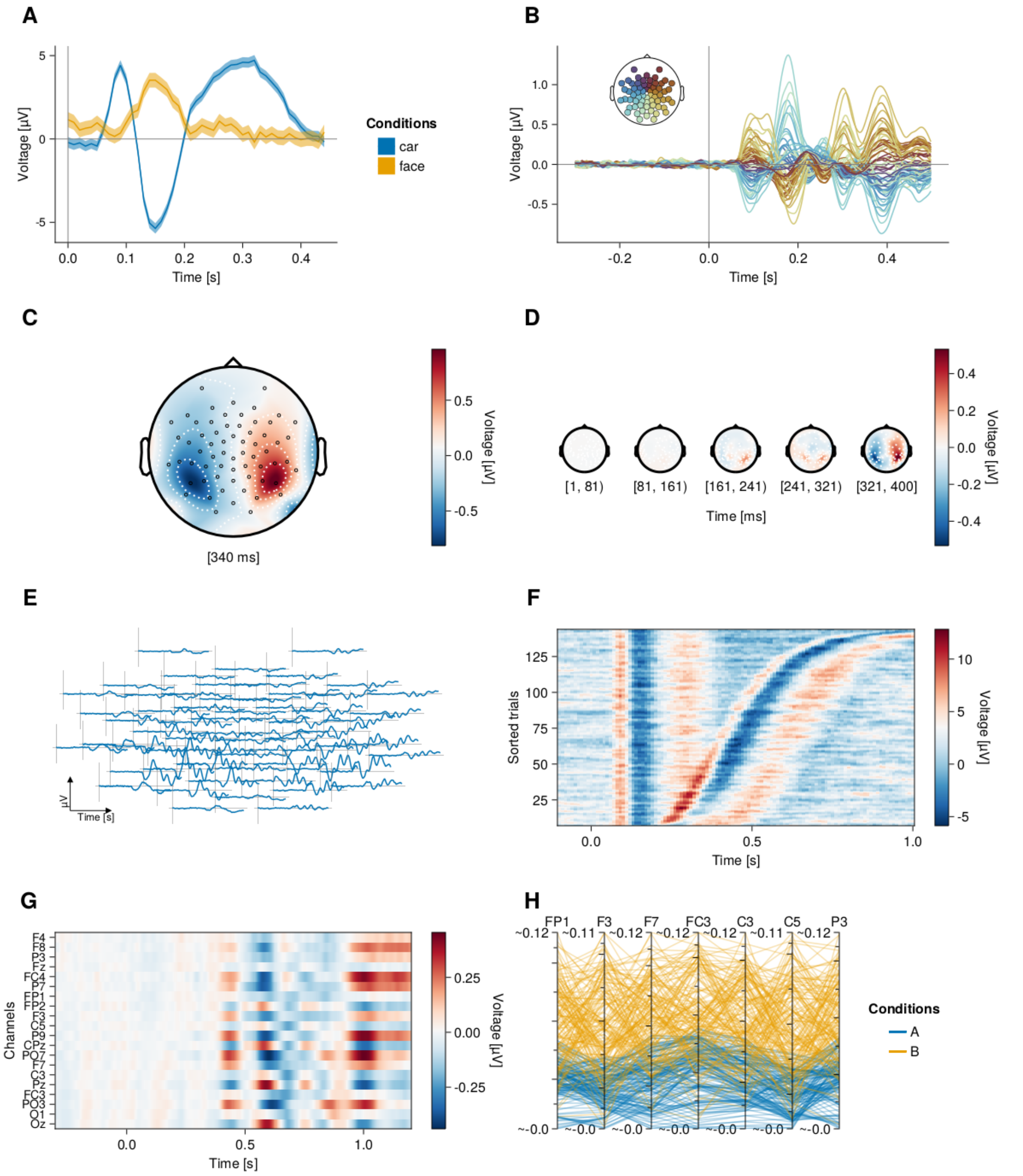

Figure 1: 8 plots generated by UnfoldMakie.jl. A) ERP plot, B) Butterfly plot, C) Topoplot, D) Topoplot timeseries, E) ERP image (Jung et al., 1998), G) Channel image, H) Parallel coordinate plot (Ten Caat et al., 2006).

Funding

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 251654672 – TRR 161 and in the Emmy Noether Programme - Project-ID 538578433.

Acknowledgements

We acknowledge contributions from Daniel Baumgartner, Niklas Gärtner, Soren Doring, Fadil Furkan Lokman, Judith Schepers, and René Skukies.

Toolbox dependencies

Only the main dependencies are listed here. The dependencies of the dependencies can be found in the respective `Manifest.toml` files. Furthermore, please note that we only list rather than cite the packages for which we could not find any citation instructions.

`AlgebraOfGraphics.jl` (Krumbiegel & Vertechi, 2024), `BenchmarkTools.jl`, `BSplineKit.jl`, `CSV.jl`, `CairoMakie.jl`, `CategoricalArrays.jl`, `ColorSchemes.jl`, `ColorTypes.jl`, `Colors.jl`, `CoordinateTransformations.jl`, `DataFrames.jl` (Bouchet-Valat & Kamiński, 2023), `DataFramesMeta.jl`, `DataStructures.jl`, `DocStringExtensions.jl`, `Documenter.jl` (Hatherly et al., 2024), `DSP.jl` (Kornblith et al., 2023), `FFMPEG_jll.jl`, `GeometryBasics.jl`, `Glob.jl`, `GridLayoutBase.jl`, `ImageFiltering.jl`, `Interpolations.jl`, `Literate.jl`, `LiveServer.jl`, `LinearAlgebra.jl`, `Makie.jl` (Danisch & Krumbiegel, 2021), `MakieThemes.jl`, `Observables.jl`, `PyMNE.jl` (Alday & Arslan, 2024), `PythonCall.jl`, `PythonPlot.jl`, `Random.jl`, `Revise.jl`, `SparseArrays.jl`, `StaticArrays.jl`, `Statistics.jl`, `StatsModels.jl`, `Test.jl`, `TopoPlots.jl` (Ehinger et al., 2024), `Unfold.jl` (Ehinger & Dimigen, 2019), `UnfoldSim.jl` (Schepers et al., 2024), `XML2_jll.jl`

References

Alday, P., & Arslan, A. (2024). *PyMNE.jl* (Version 0.2.2). https://doi.org/10.5281/zenodo.14172218

Bouchet-Valat, M., & Kamiński, B. (2023). DataFrames.jl: Flexible and fast tabular data in julia. *Journal of Statistical Software*, *107*(4), 1–32. https://doi.org/10.18637/jss.v107.i04

Chen, J., & Revels, J. (2016). Robust benchmarking in noisy environments. *arXiv e-Prints*. https://doi.org/10.48550/arXiv.1608.04295

Crameri, F., Shephard, G. E., & Heron, P. J. (2020). The misuse of colour in science communication. *Nature Communications*, *11*(1), 5444. https://doi.org/10.1038/s41467-020-19160-7

Danisch, S., & Krumbiegel, J. (2021). Makie.jl: Flexible high-performance data visualization for julia. *Journal of Open Source Software*, *6*(65), 3349. https://doi.org/10.21105/joss.03349

Delorme, A., & Makeig, S. (2004). EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, *134*(1), 9–21. https://doi.org/10.1016/j.jneumeth.2003.10.009

Ehinger, B. V. (2018). *EEGVIS toolbox* (Version 0.1). https://doi.org/10.5281/zenodo.1312813

Ehinger, B. V., Alday, P., Singhvi, A., & Simon, D. (2024). *TopoPlots.jl: A julia package for topoplot generation* (Version 0.2.0). https://doi.org/10.5281/zenodo.14179492

Ehinger, B. V., & Dimigen, O. (2019). Unfold: An integrated toolbox for overlap correction, non-linear modeling, and regression-based EEG analysis. *PeerJ*, *7*, e7838. https://doi.org/10.5281/zenodo.5759066

Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., Goj, R., Jas, M., Brooks, T., Parkkonen, L., & Hämäläinen, M. S. (2013). MEG and EEG Data Analysis with MNE-Python. *Frontiers in Neuroscience*, *7*(267), 1–13. https://doi.org/10.3389/fnins.2013.00267

Hatherly, M., Piibeleht, M., & Ekre, F. (2024). *Documenter: A documentation generator for julia* (Version 1.7). https://github.com/JuliaDocs/Documenter.jl

Jung, T.-P., Makeig, S., Westerfield, M., Townsend, J., Courchesne, E., & Sejnowski, T. J. (1998). Analyzing and visualizing single-trial event-related potentials. *Advances in Neural Information Processing Systems*, *11*. https://doi.org/10.1002/hbm.1050

Kornblith, S., Lynch, G., Holters, M., Santos, J. F., Russell, S., Kickliter, J., Bezanson, J., Adalsteinsson, G., Arslan, A., Yamamoto, R., jordancluts, Pastell, M., Kelman, T., Arthur, B., Krauss, T., HDictus, El-Saawy, H., Kofron, J., Hanson, E., … Smith, J. (2023). *JuliaDSP/DSP.jl: v0.7.9* (Version v0.7.9). Zenodo. https://doi.org/10.5281/zenodo.8344531

Krumbiegel, J., & Vertechi, P. (2024). *AlgebraOfGraphics.jl* (Version 0.8.13). https://github.com/MakieOrg/AlgebraOfGraphics.jl

Lopez-Calderon, J., & Luck, S. J. (2014). ERPLAB: An open-source toolbox for the analysis of event-related potentials. *Frontiers in Human Neuroscience*, *8*, 213. https://doi.org/10.3389/fnhum.2014.00213

Luke, R. (2021). *Neuroimaging.jl* (Version 1.1.2). https://github.com/rob-luke/Neuroimaging.jl

Mikheev, V., Skukies, R., & Ehinger, B. V. (2024). The art of brainwaves: A survey on event-related potential visualization practices. *Aperture Neuro*, *4*. https://doi.org/10.52294/001c.116386

Moreland, K. D. (2015). *Why we use bad color maps and what you can do about it.* Sandia National Lab.(SNL-NM), Albuquerque, NM (United States). https://doi.org/10.2352/issn.2470-1173.2016.16.hvei-133

Oostenveld, R., Fries, P., Maris, E., & Schoffelen, J.-M. (2011). FieldTrip: Open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. *Computational Intelligence and Neuroscience*, *2011*(1), 156869. https://doi.org/10.1155/2011/156869

Pereyra, S. L. (2024). *EEGToolkit.jl* (Version 0.2). https://github.com/slopezpereyra/EEGToolkit.jl

Robbins, K. A. (2012). EEGVIS: A matlab toolbox for browsing, exploring, and viewing large datasets. *Frontiers in Neuroinformatics*, *6*, 17. https://doi.org/10.3389/fninf.2012.00017

Schepers, J., Lips, L., Marathe, M., & Ehinger, B. (2024). *UnfoldSim.jl*. https://doi.org/10.5281/zenodo.13933412

Tadel, F., Bock, E., Niso, G., Mosher, J. C., Cousineau, M., Pantazis, D., Leahy, R. M., & Baillet, S. (2019). MEG/EEG group analysis with brainstorm. *Frontiers in Neuroscience*, *13*, 76. https://doi.org/10.3389/fnins.2019.00076

Ten Caat, M., Maurits, N. M., & Roerdink, J. B. (2006). Design and evaluation of tiled parallel coordinate visualization of multichannel EEG data. *IEEE Transactions on Visualization and Computer Graphics*, *13*(1), 70–79. https://doi.org/10.1109/tvcg.2007.9

Wysokiński, A. (2024). *NeuroAnalyzer* (Version 0.24.11). https://doi.org/10.5281/zenodo.14010334