Brain Topography manuscript No.
(will be inserted by the editor)

A simulation framework for benchmarking EEG-based brain connectivity estimation methodologies

Supplementary Material

Stefan Haufe · Arne Ewald

Received: date / Accepted: date

S. Haufe
Laboratory for Intelligent Imaging and Neural Computing, Columbia University, New York, NY, 10027, USA
Machine Learning Department, Technische Universität Berlin, 10587 Berlin, Germany
E-mail: stefan.haufe@tu-berlin.de

A. Ewald
Department of Neurophysiology and Pathophysiology, University Medical Center Hamburg-Eppendorf, 20246 Hamburg, Germany
E-mail: mail@aewald.net

A Data provided

Data will be provided under http://bbci.de/supplementary/EEGconnectivity/. The struct sa in data/sa.mat contains the New York Head (see Huang et al., 2016, Neuroimage, In Press.) model including all data required to perform forward calculations and plotting. It contains the following fields.

sa		
.cortex75K		Structure containing a high-resolution (75K nodes) triangular mesh of the cortical surface (gray matter/CSF interface).
	.vc	MNI coordinates of mesh vertices.
	.vc_smooth	Vertex coordinates of a spatially smoothed mesh with identical structures.
	.tri	Indices of nodes forming the triangles (faces) of the mesh.
	.tri_left	(.tri_right) Face indices for left and right hemispheres separately.
	.normals	Normal vector for each vertex.
	.curvature	Curvature for each vertex.
	.sulcimap	Binary mask indicating for each vertex whether it is located in a sulcus.
	.EEG_V_fem	EEG lead fields for 108 electrodes computing using a highly accurate finite element model. Source dipoles are located at the 75K cortical mesh nodes along the three spatial dimension.
	.EEG_V_fem_normal	EEG lead fields assuming dipole orientations perpendicular to the cortical surface (the product of .V_cortex75K.EEG_V_fem and .cortex75K.normals).
	.roi_mask	A numeric mask that indicates the corresponding octant (ROI) for each voxel.
	.roi_mask_str	A text mask that indicates the corresponding octant (ROI) for each voxel.
.cortex2K		Low-resolution (2K nodes) triangular mesh of the cortical surface. Useful for efficient forward and inverse modeling.
	.in_from_cortex75K	Vertex indices relative to the high-resolution mesh cortex75K, of which this mesh is a strict subset. MNI coordinates of mesh vertices of cortex2K are given by cortex75K.vc(cortex2K.in_from_cortex75K,:) and cortex75K.vc_smooth(cortex2K.in_from_cortex75K,:).
	.tri	(.in_L , ... , .in_L_inner , ... , .in_LPS , ... , .in_LPS_inner) Analogous to cortex75K, but indexing cortex2K.in_from_cortex75K. For example, MNI coordinates for mesh nodes located in the left hemisphere are obtained as cortex75K.vc(cortex2K.in_from_cortex75K(cortex2K.in_L),:).
	.in_to_cortex75K_eucl	Index vector used to interpolate functions defined on the 2K cortical mesh (such as inverse solutions) on the 75K mesh for plotting purposes. If S is a function defined on the 2K mesh, then S(cortex2K.in_to_cortex75K_eucl) is the function projected onto the 75K mesh. Here, interpolation is based on the Euclidean distance.
	.in_to_cortex75K_geod	Interpolation indices based on the geodesic distance along the 75K mesh, using that the 2K mesh is a subset of the 75K mesh. To be used in the same way as .in_to_cortex75K_eucl.
	.roi_mask	A numeric mask that indicates the corresponding octant (ROI) for each voxel.
	.roi_mask_str	A text mask that indicates the corresponding octant (ROI) for each voxel.

sa (cntd.)		
.head		Triangulated head surface.
	.vc	(.tri, .normals, .curvature) Analogous to cortex75K.
.mri		Structure containing the ICBM152 v2009b anatomical MRI data.
	.data	3D MR image with enhanced brain to background contrast.
	.brainmask	Binary brain mask.
.EEG_clab_electrodes		Cell array of EEG electrode labels.
.EEG_locs_2D		2D projection of the EEG electrodes to the x, y-plane. Useful for plotting 2D scalp maps.
.EEG_locs_3D		MNI coordinates of EEG electrodes, and normal vectors relative to the scalp surface.
.EEG_elec2head		Matrix to project functions defined per EEG electrode to the head surface. Useful to create 3D scalp potential maps.
.mni2mri		Affine transformation matrix to convert MNI coordinates into index vectors for the MR image mri.data. Consists of a scaling and a translation.
.mri2mni		Affine transformation to convert MRI coordinates into MNI space.
.naspalparori		MNI coordinates of Nasion (NAS), left and right pre-auricular points (PAL/PAR), and the origin (ORI) of the coordinate system defined by NAS, PAL and PAR.
.acpcihori		MNI coordinates of Anterior and Posterior Commissure (AC/PC), inter-hemispheric point (IH), and the origin of the coordinate system defined by AC, PC, and IH.

The file data/miscdata.mat contains the following additional variables.

me	MNI coordinates of the origin used to define hemispheres and octants based on median splits.
directions	List of the names of the six hemispheres: ‘left’, ‘right’, ‘posterior’, ‘anterior’, ‘superior’, ‘inferior’.
inds_dir_outer_75K	Cell array containing the indices cortex75K.in_L , . . . , cortex75K.in_I of the mesh vertices belonging to each of the six hemispheres in the order given by directions.
inds_dir_inner_75K	Analogous list for the indices cortex75K.in_L_inner , . . . , cortex75K.in_I_inner.
rois	List of the names of the eight brain octants: ‘LPS’, ‘LPI’, ‘LAS’, ‘LAI’, ‘RPS’, ‘RPI’, ‘RAS’, ‘RAI’.
inds_roi_outer_75K	Cell array containing the indices cortex75K.in_LPS , . . . , cortex75K.in_RAI of the mesh vertices belonging to each of the eight octants in the order given by rois.
inds_roi_inner_75K	Analogous list for the indices cortex75K.in_LPS_inner , . . . , cortex75K.in_RAI_inner.
inds_dir_outer_2K	(inds_dir_inner_2K , . . . , inds_roi_outer_2K , . . . , inds_roi_inner_2K) Analogous lists of the index vectors of the low-resolution mesh cortex2K.

In addition, 100 pre-computed instances of the proposed benchmark are provided in the directory data. Each instance is contained in a subfolder EEG/dataset_i, $i = 1, \ldots, 100$. Each folder contains a file data.mat that was generated using generate_datasets_ar.m (see below). The corresponding ground truth truth.mat, as well as the random number generator seed are not disclosed.

B Code provided

The main folder contains the following core functions.

`[sources_int, sources_nonint] = generate_sources_ar(fs, len, bandpass)`	
	Generates interacting and non-interacting bivariate time-series according to a 5th order linear autoregressive model.
`fs`	Sampling rate in Hz.
`len`	Length of the simulated recording in seconds.
`bandpass`	Limits of the bandpass to be applied, in Hz. If empty broadband signals are returned.
`sources_int`	$2 \times$ `fs` $\cdot$ `len` bivariate times series with uni-directional information flow from first to second variable.
`sources_nonint`	$2 \times$ `fs` $\cdot$ `len` bivariate time series with the first and second variable statistically independent.

`generate_datasets_ar(ndatasets, dataset_string)`	
	Generates `ndatasets` datasets composed of baseline and data pseudo-measurements, and saves them to a directory structure in `data/dataset_string`. For each instance $i = 1, \ldots,$ `ndatasets` of the experiment, a subfolder `EEG/dataset_i` is created. Each subfolder contains data of that instance as `data.mat`. In a separate folder `truth/dataset_i`, the ground truth is stored as `truth.mat`. `data.mat` contains the variables `EEG_data`, `EEG_baseline_data` and `fs`, where `EEG_data` is contains the $108 \times$ `fs` $\cdot$ `len` pseudo-EEG data and `EEG_baseline_data` contains the baseline pseudo-EEG data of the same dimensionality, and `fs = 100` is the sampling rate. `truth.mat` contains a structure `truth` aggregating information about the true underlying source locations and interaction structure of the generated data. The seed `sd` of the Matlab random number generator before generating the data is stored in `data/dataset_string/sd.mat`.
`ndatasets`	Number of datasets to be generated.
`dataset_string`	Name of the directory to store the generated data in.

`res = evaluate_performance(truth, est)`	
	Compares the ground truth w. r. t. source locations, connectivity presence and interaction direction with an estimate, and calculates the performance measures LOC, CONN and DIR.
`truth`	Data structure containing the ground truth, as saved by `generate_datasets_ar`.
`est`	Data structure containing the estimates regarding source octants and their interaction. `est.rois` is a cell array containing the estimated source octants in arbitrary order, e.ġ., `est.rois =` {`’LAI’ ’LPI’`}. If no guess is made, the field is supposed to be empty, i.e. `est.rois = []`. `est.interaction` is the estimation, if `eeg_data` is believed to contain interacting sources (`est.interaction = 1`), or not (`est.interaction = 0`). `est.interaction = []` indicates that no guess is made. The third variable, `est.sender` indicates the octant containing the sending source, e. g. `est.sender = 2` if `est.rois`{`2`} is believed to contain the sending source, or `est.sender = []` if no guess is made. `est.sender` is only evaluated if `est.interaction_dataset` is provided.
`res`	Results structure containing the fields `.loc`, `.conn` and `.dir`.

`EEG_estimate_lcmv_imcoh_psi(ndatasets, dataset_string)`	
	Analyzes given EEG datasets using LCMV beamforming, the ImCoh and PSI, and calculates the performance measures LOC, CONN and DIR using the ground truth and the function `evaluate_performance`.
`ndatasets`	Number of datasets to be analyzed.
`dataset_string`	String specifying the folder from which the data are loaded, the same as in `generate_datasets_ar`.

`run_benchmark`	
	Example script generating 100 datasets with `generate_datasets_ar` and analyzing it with `EEG_estimate_lcmv_imcoh_psi`.

The `tools` folder contains utilities for data generation, analysis using LCMV beamforming, the ImCoh and PSI, and plotting. To add them to the path, call `set_path`.