ADJUST: An Automatic EEG artifact Detector based on the Joint Use of Spatial and Temporal features

A Tutorial.

Marco Buiatti[1] and Andrea Mognon[2]

[1] INSERM U992 Cognitive Neuroimaging Unit, Gif sur Yvette, France
Contact : marco.buiatti@gmail.com
[2] Center for Mind/Brain Sciences, University of Trento, Italy

Summary

Introduction 2
ADJUST requirements and installation 2
EEGLAB basics 3
  How to run EEGLAB 3
  Loading data 3
Running ADJUST for artifact detection 4
  Pre-processing 4
  ICA decomposition 5
  ADJUST Artifact Detection 5
  Running ADJUST 6
  Visualization of the results and artifact removal 7
  Additional notes. 12
    Channel number and position. 12
    Running ADJUST from the command line 12
    Benchmark version 13
  References 13

Introduction

A major problem in the analysis of electroencephalographic (EEG) recordings is that the activity due to artifacts has typically much higher amplitude than the one generated by neural sources. A successful method for removing artifacts from EEG recordings is Independent Component Analysis (ICA), but its implementation remains largely user-dependent. To overcome this limitation, we have developed ADJUST, a completely automatic algorithm that identifies artifacted Independent Components (IC) by combining stereotyped artifact-specific spatial and temporal features. Features are optimised to capture blinks, eye movements and generic discontinuities. Once artifacted IC are identified, they can be simply removed from the data while leaving the activity due to neural sources almost unaffected.

ADJUST has been implemented as a plugin of the EEGLAB toolbox (Delorme & Makeig, 2004), a matlab-based software for analysis of electrophysiological data. The ADJUST plugin uses EEGLAB's excellent visualization tools to display the properties of the artifacted IC in multiple dimensions (topography, time course, power spectrum). The present tutorial is a guide to the use of the ADJUST plugin within the EEGLAB toolbox.

For the details of the algorithm and its validation on real data see the **reference paper**:

Mognon, A., Jovicich, J., Bruzzone, L., Buiatti, M.
ADJUST: An Automatic EEG artifact Detector based on the Joint Use of Spatial and Temporal features. *Psychophysiology* 48 (2), 229-240 (2011).

Please cite this paper to reference ADJUST in publications.

ADJUST requirements and installation.

ADJUST runs in Matlab (The Mathworks, Inc.), within the EEGLAB software (Delorme and Makeig, 2004).

No Matlab toolboxes are necessary[1]*.*

[1]The plot of the power spectrum within the visualization of IC properties will not be displayed if Matlab's signal processing toolbox is missing, but ADJUST will run regularly.

The ADJUST plugin, the present tutorial and a sample dataset (*adjust_sampledata.set*) can be downloaded here.

To install ADJUST, you have to:

- Download EEGLAB;
- Download ADJUST;
- Copy ADJUST plugin folder into EEGLAB's plugins directory /eeglab13_1_1b/plugins/.

EEGLAB basics

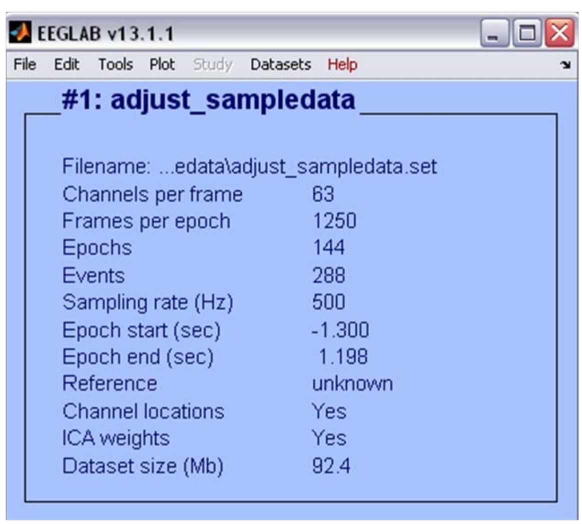
EEGLAB is a well-designed, well-documented software for analysis of electrophysiological signals including a wide range of data processing and visualization tools. Here we introduce the basic steps necessary to use the ADJUST plugin within EEGLAB. For a complete guide on how to use EEGLAB please refer to EEGLAB's wiki.

How to run EEGLAB

Open Matlab and set as Current Directory the folder where EEGLAB is installed.
Type 'eeglab' in the Command Window and the EEGLAB window will open.

Loading data

A dataset can be loaded by clicking on *File > Load existing dataset* and browsing an existing *.set* file. For example, browse to the folder where you saved the tutorial dataset and select *adjust_sampledata.set*. The EEGLAB window will now display the most relevant information on the dataset (number of channels, number of epochs etc.):



Running ADJUST for artifact detection

Pre-processing

EEG artifacts can be broadly divided in two classes: *non-stereotyped* artifacts due to multiple factors like the subject's movements or external sources of interference, and *stereotyped* artifacts due, for example, to ocular eye movements, blinks, heart beats (Onton *et al.*, 2006). Artifacts from the second class are likely to be captured by some ICA components because they have a highly reproducible spatial distribution and temporal profiles. The aim of ADJUST is to automatically identify those components in the ICA decomposition and remove them from the data. Artifacts from the first class are problematic for ICA because since their spatial distribution is extremely variable, they introduce a large number of unique scalp maps, leaving few ICs available for capturing brain sources. Accordingly, ADJUST does not attempt to remove these artifacts, and it relies on a suitable pre-processing for removing them before the ICA decomposition. For this reason, an accurate pre-processing of the data prior to ICA decomposition is *crucial* for an efficient performance of ADJUST.

Pre-processing strategies vary depending on the experimental protocol, the kind of analysis that will follow etc. Any procedure that removes most of the non-stereotypical artifacts without severely distorting the data is acceptable for ADJUST.

As a general guideline, here we propose a basic pre-processing pipeline suitable for a classical repeated measures protocol and an Event-Related Potential analysis:

- High-pass filter the continuous data (to remove slow linear trends). Typical values: 0.05-0.3 Hz (higher cutoffs might distort 'late' ERPs like P300, N400, see (Acunzo *et al.*, 2012)).
- Remove from your continuous data all segments containing paroxysmal artifacts (but NOT pure eye movements, as you will remove them with ADJUST without rejecting the data). EEGLAB provides efficient visualization tools to do this by visual inspection.
- Remove bad channels.
- Low-pass filter and epoch (better if WITHOUT removing the baseline, see (Groppe *et al.*, 2009));
- Run ICA;
- Run ADJUST and remove artifacted components;
- Re-check your epochs for residual artifacts, e.g. by using EEGLAB's epoch rejection tools.
- Re-interpolate bad channels from neighbour channels.
- Re-reference and baseline-correct.

See here for EEGLAB’s strategy and rejection tools.

ICA decomposition

ADJUST is based on EEGLAB’s default ICA implementation. Once the non-stereotyped artifacts have been removed from the data, the necessary condition for having a high-quality ICA decomposition is to have *enough data*: number of data points ≥ k*N^2 where N=number of channels, and k≥30 (see EEGLAB’s ICA page for details).

ICA must be computed *before* running ADJUST.

ADJUST Artifact Detection

To identify artifacted ICs, ADJUST automatically computes a set of spatial and temporal features for each IC (see (Mognon *et al.*, 2011) for details). Temporal features are computed on the variability of the data across epochs. If data are continuous, they are automatically epoched to consecutive 5 s epochs for the only purpose of computing the temporal features.

Each artifact class is associated with one spatial and one temporal feature (plus a few supplementary constraints):

Eye blinks:

- Spatial Average Difference (SAD) - sensitive to higher amplitude in frontal areas compared to posterior areas;
- Temporal Kurtosis (TK) - sensitive to outliers in the amplitude distribution, a typical signature of eye blinks.
- Supplementary controls: a) Variance of amplitude larger in frontal than in posterior areas; b) Symmetry of frontal topography.

Vertical eye movements:

- Spatial Average Difference (SAD) - same as above;
- Maximum Epoch Variance (MEV) - computes the maximum value over the epochs of temporal variance, a measure that is more sensitive than kurtosis to the fluctuations typical of vertical eye movements (slower than the ones caused by blinks).
- Supplementary controls: a) Variance of amplitude larger in frontal than in posterior areas; b) Symmetry of frontal topography.

Horizontal eye movements:

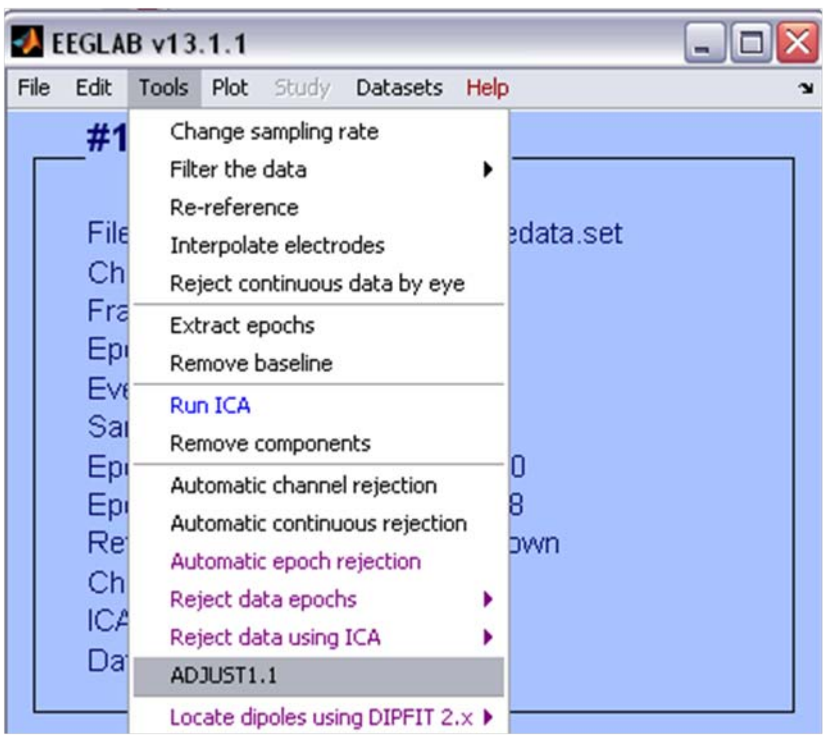
- Spatial Eye Difference (SED) - sensitive to anti-phase (one negative and one positive) large amplitudes in frontal channels near the eyes, typical of horizontal eye movements;
- Maximum Epoch Variance (MEV) - same as above.

Generic discontinuities:

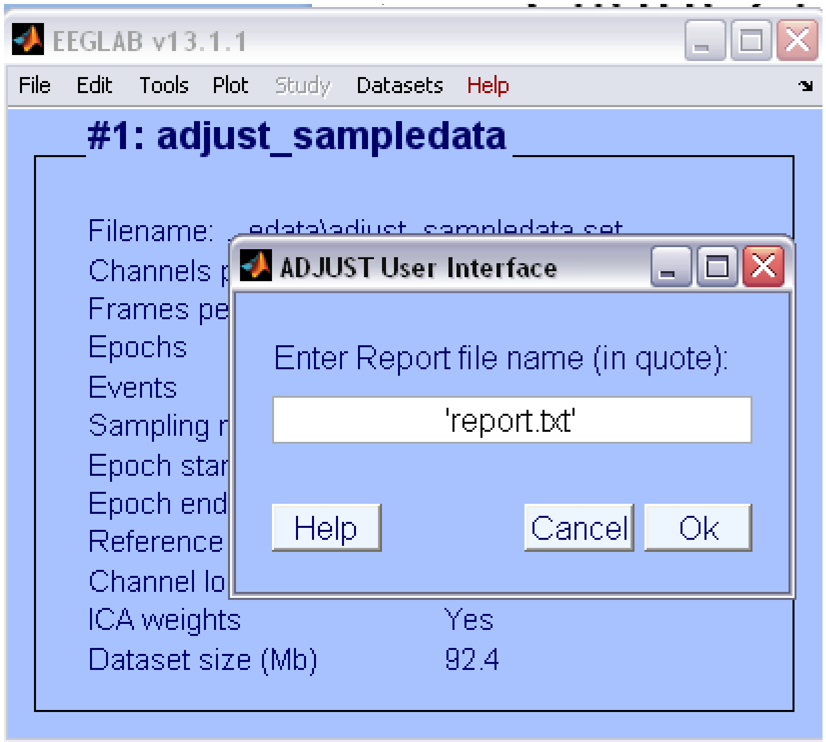
- Generic Discontinuities Spatial Feature (GDSF) - sensitive to local spatial discontinuities;
- Maximum Epoch Variance (MEV) - same as above.

Running ADJUST

To start ADJUST, go to the EEGLAB menu and click *Tools > ADJUST*:



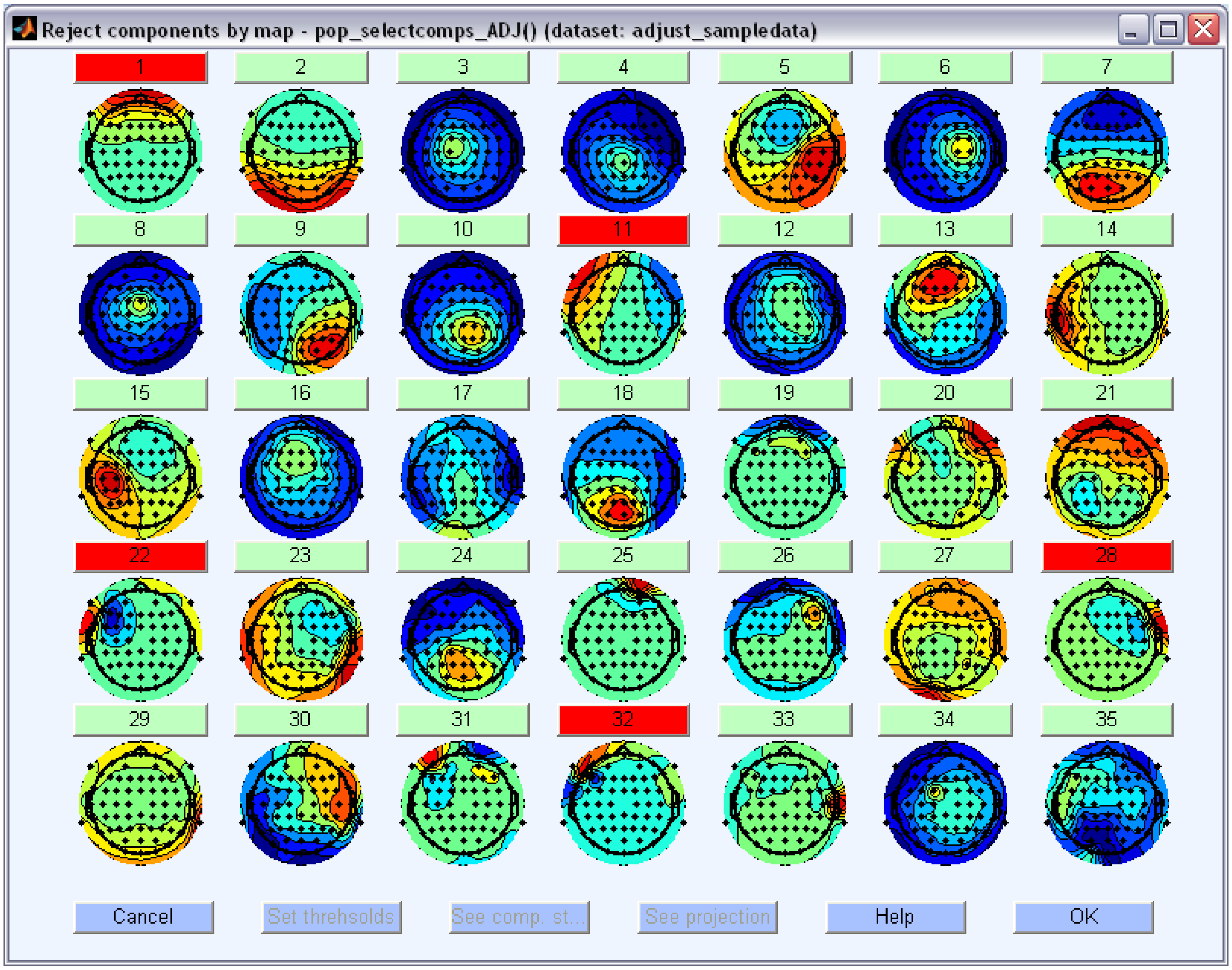
The program will ask you for the name of a report (text) file where ADJUST feature values and list of artifacted ICs will be written. The file will be saved in the current directory. Specify any name you prefer (default is 'report.txt') and press OK:



While running, ADJUST displays the key informations about the processing on your command window: the dataset name, the computation of all the features and the respective thresholds, the artifacted IC identification, the name of the report file and the name of a Matlab file (also saved in the current directory) where the index of the artifacted ICs will be saved.

Visualization of the results and artifact removal

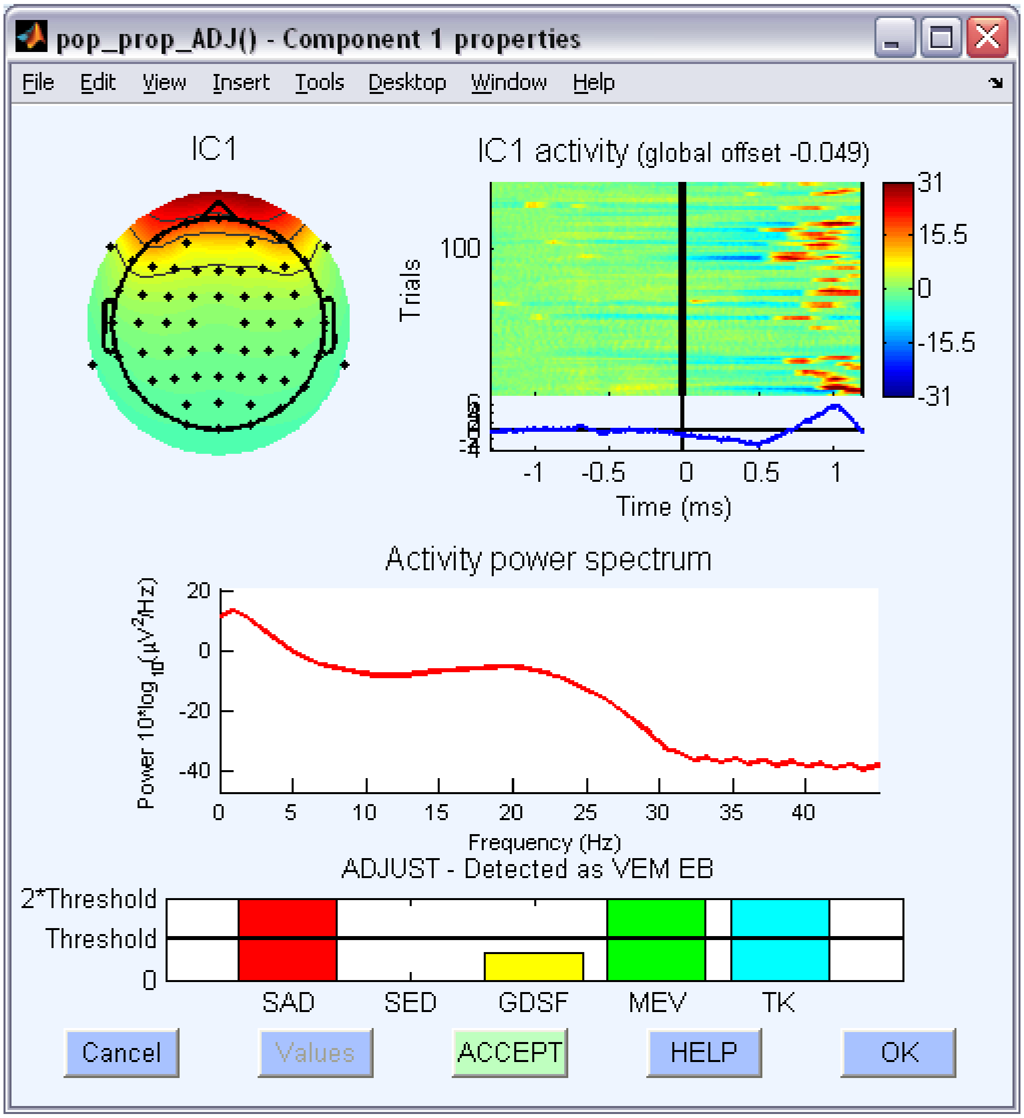
At the end of ADJUST computation, a number of figures are displayed to illustrate the results. Figures show the topographies of all ICs ordered by percentage of explained variance of the data (one figure shows a maximum of 35 ICs, a rectangular panel on top of each topography indicates its index number); ICs detected as artifacted by ADJUST are marked by a red index number panel. The figure below illustrates the first 35 ICs of the tutorial dataset. ICs 1, 11, 22, 28 and 32 have been classified as artifacted by ADJUST.



Figures are generated in a useful interactive mode that allows an easy and fast check of the spatial, temporal and spectral properties of each IC to confirm the classification proposed by ADJUST. To display these properties, click on the IC index number panel on top of its topography. A new figure will open displaying the IC topography (top left), the 'ERP image' (a useful graphic representation showing the time course of the color-coded IC amplitude through all epochs (Jung *et al.*, 2000), top right), the IC power spectrum (center), and the normalized values[2] of the IC features computed by ADJUST.

As a first example, click on IC component 1 (IC1):

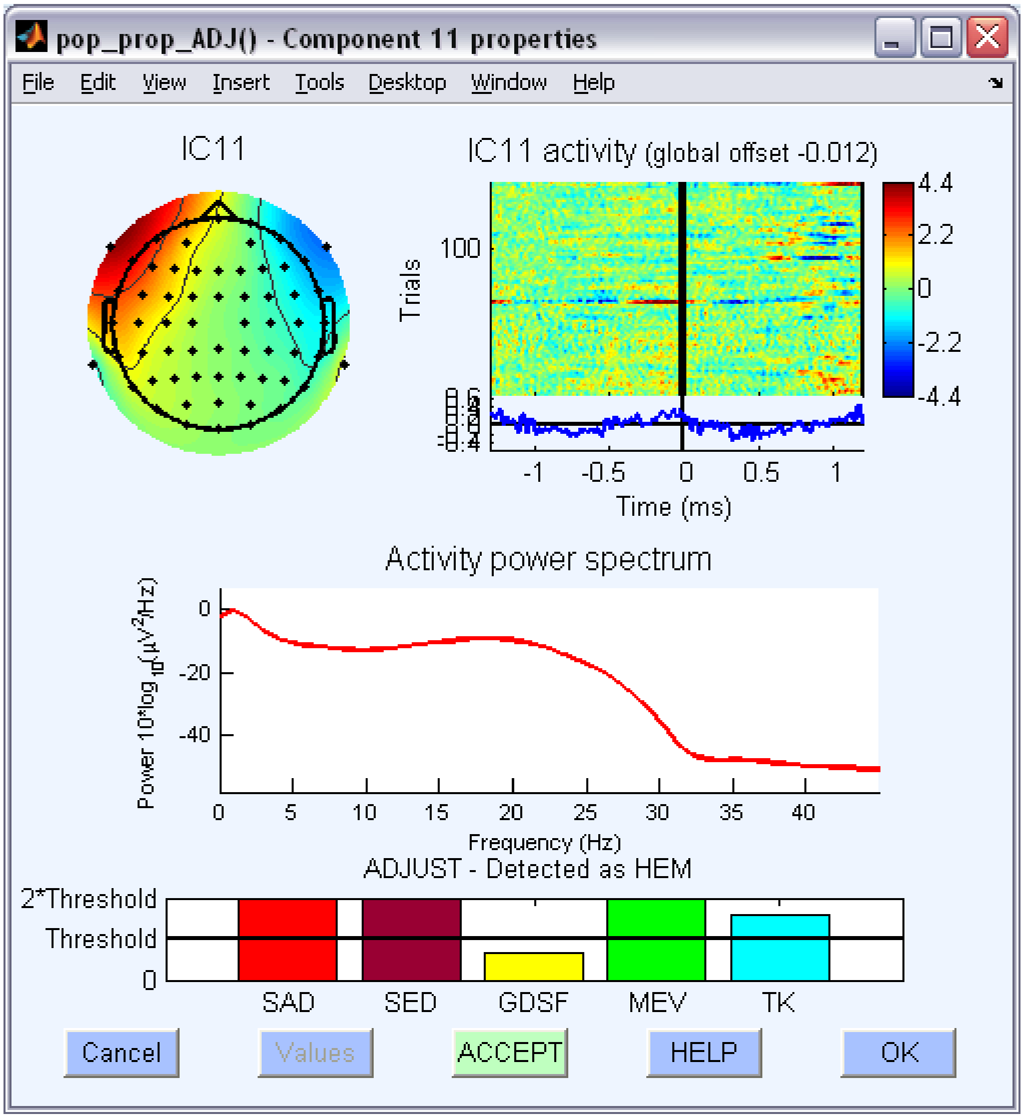
[2] For each feature, the value of the bar in the figure is = (feature value-median of first peak of feature's bimodal distribution) / (threshold-median of first peak of feature's bimodal distribution). For details about the estimation of the threshold and feature's bimodal distribution, see ADJUST reference paper (Mognon *et al.,* 2011).



IC1 nicely captures blink artifacts, as it is evident from the frontal topography and the high-amplitude peaks in the ERP image. ADJUST correctly identifies the IC as a blink artifact - both SAD and TK features abundantly cross the threshold. Since MEV also crosses the threshold, this component is also classified as a vertical eye movement (since the aim of ADJUST is to identify artifacted ICs independently from the cause of the artifact, the features' 'space' of each artifact class does not need to be independent).

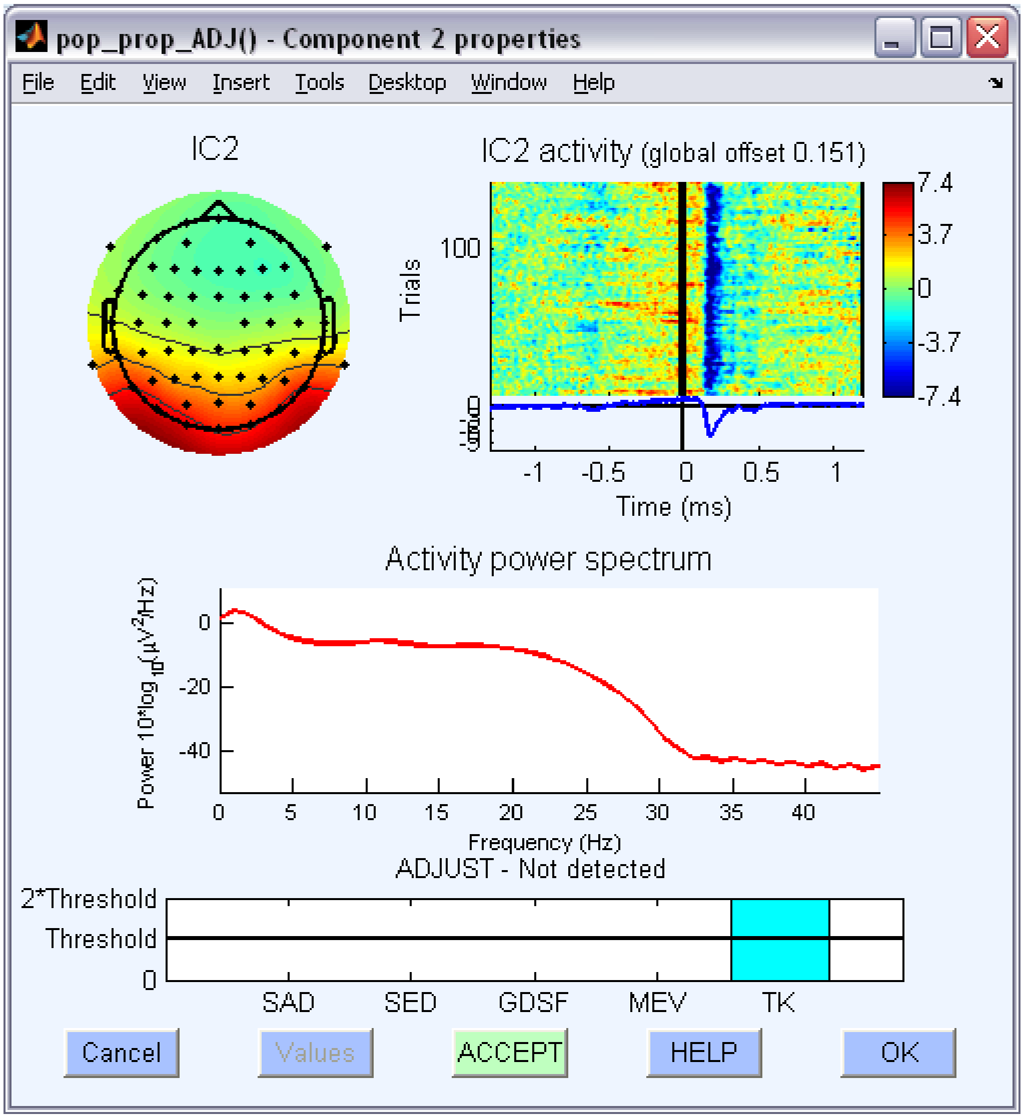
If you agree with ADJUST and you want to reject this IC, you have to click on the 'ACCEPT' button at the bottom of the figure; the button will change into a pink-colored 'REJECT' button, meaning that you marked the IC for rejection. Press the 'OK' button: the figure will close and EEGLAB will remember that this component has been marked for removal.

As another example, click on IC11:



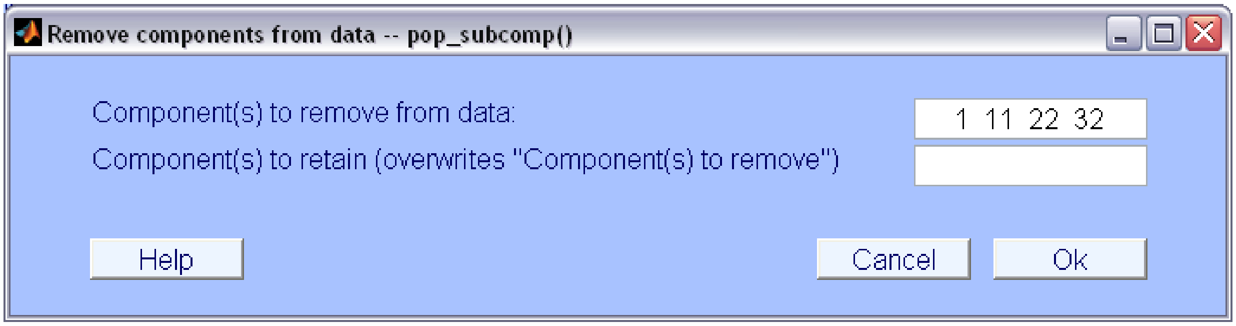
IC11 captures horizontal eye movements, as it is clear from the asymmetric frontal topography and the alternating negative-positive waves in the ERP image. The component is correctly classified by ADJUST as an horizontal eye movement (HEM), as both SED and MEV features cross the threshold. Note that even though TK, MEV and SAD features cross their respective threshold, the component is neither classified as blink nor as vertical eye movement because the frontal topography is not symmetrical (supplementary control).

As a last example, click on IC2:



IC2 clearly captures an activity of neural origin, as it is evident from the time-locked evoked potential in the ERP image and the posterior topography (a visual N170 ERP). In this case, the value of one temporal feature (TK) crosses the threshold, but this is not enough to classify this component as an artifact because the spatial feature associated with TK (SAD) is well below the threshold.

Once you have marked as 'REJECT' all ICs you want to reject, go to *Tools* > *Remove components* to reject them. A window will pop out with the marked components selected both in the 'remove' and the 'retain' space. Remove the once in the 'retain' space and press 'Ok':



Another window will pop out giving you the chance to check the effect of the ICs removal on the ERPs:

Press 'Accept' to definitely remove the ICs from the data and save the clean dataset.

Additional notes.

Channel number and position.

ADJUST assumes that the number of IC components is equal to the number of channels; if this is not the case, the program will stop the computation and an error message will be displayed.

The suggested procedure with bad channel is to remove them *before* running ICA and ADJUST, and to reinterpolate them from neighbouring channels *after* artifact removal.

ADJUST is optimized for running on the first ICA decomposition; it is currently not possible to run it after having removed some components and rerun ICA.

If some channels do not have channel positions (e.g. EOG channels), they will be excluded from the computation of the spatial features, but they will be included in the computation of the temporal features. A warning will be displayed indicating the index of those channels.

Running ADJUST from the command line.

ADJUST can be run directly from Matlab’s command line by typing:

```
>> EEG = interface_ADJ (EEG,'report');
```

where EEG is the EEGLAB dataset, and 'report' is the name of the report file.

Benchmark version

EEGLAB version 13.1.1b on MATLAB 8.1.0.604 (R2013a).
OS: Windows XP, Ubuntu Linux 12.04 LTS.

Last revised: May 2014.

[1] Center for Mind/Brain Sciences, University of Trento, Italy
[2] INSERM U992 Cognitive Neuroimaging Unit, Gif sur Yvette, France

References

Acunzo, D. J., Mackenzie, G. and van Rossum, M. C. (2012) Systematic biases in early ERP and ERF components as a result of high-pass filtering. *J Neurosci Methods* **209,** 212-218.

Delorme, A. and Makeig, S. (2004) EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J Neurosci Meth* **134,** 9-21.

Groppe, D. M., Makeig, S. and Kutas, M. (2009) Identifying reliable independent components via split-half comparisons. *NeuroImage* **45,** 1199-1211.

Jung, T. P., Makeig, S., Humphries, C., Lee, T. W., McKeown, M. J., Iragui, V. and Sejnowski, T. J. (2000) Removing electroencephalographic artifacts by blind source separation. *Psychophysiology* **37,** 163-178.

Mognon, A., Jovicich, J., Bruzzone, L. and Buiatti, M. (2011) ADJUST: An automatic EEG artifact detector based on the joint use of spatial and temporal features. *Psychophysiology* **48,** 229-240.

Onton, J., Westerfield, M., Townsend, J. and Makeig, S. (2006) Imaging human EEG dynamics using independent component analysis. *Neurosci Biobehav Rev* **30,** 808-822.