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

Andrea Mognon[1] and Marco Buiatti[2]
1: Center for Mind/Brain Sciences, University of Trento, Italy
2: INSERM-CEA Cognitive Neuroimaging Unit, Gif sur Yvette, France

Last revised: May 2010

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, in press (2010).

Contents

Chapter 1

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. 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, in press (2010). Please cite this paper to reference ADJUST in publications.

ADJUST has been implemented as a plugin of the EEGLAB toolbox, a matlab-based software for analysis of electrophysiological data (`http://sccn.ucsd.edu/eeglab/`). The ADJUST plugin uses EEGLAB's functions to preprocess the data and compute the ICA, and its excellent tools of visualization 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.

Chapter 2

ADJUST installation

2.1 Plugin Content

The ADJUST plugin contains:

- a folder called *ADJUST_PLUGIN* including the m-files:
  - LICENSE.txt
  - eegplugin_adjust.m
  - ADJUST.m
  - pop_ADJUST_interface.m
  - interface_GA.m
  - interface_ADJ.m
  - computeSED_NOnorm.m
  - computeSAD.m
  - compute_GD_feat.m
  - trim_and_mean.m
  - trim_and_max.m
  - EM.m
  - GrossArtifactADJ.m
  - mat2vec.m
  - pop_selectcomps_ADJ.m
  - pop_prop_ADJ.m
- A tutorial dataset called *adjust_sampledata.set/.dat* (belonging to the dataset used for validation in the reference paper);
- ...and of course the present tutorial.

2.2 Installation

The software runs in Matlab (The Mathworks, Inc.), within the EEGLAB software (see `http://sccn.ucsd.edu/eeglab/` and Delorme and Makeig (2004)). To install it, you have to:

- download EEGLAB from `http://sccn.ucsd.edu/eeglab/install.html`;
- install EEGLAB;
- copy the *ADJUST_PLUGIN* folder and paste it into the EEGLAB *plugins* directory, as in Figure 2.1.

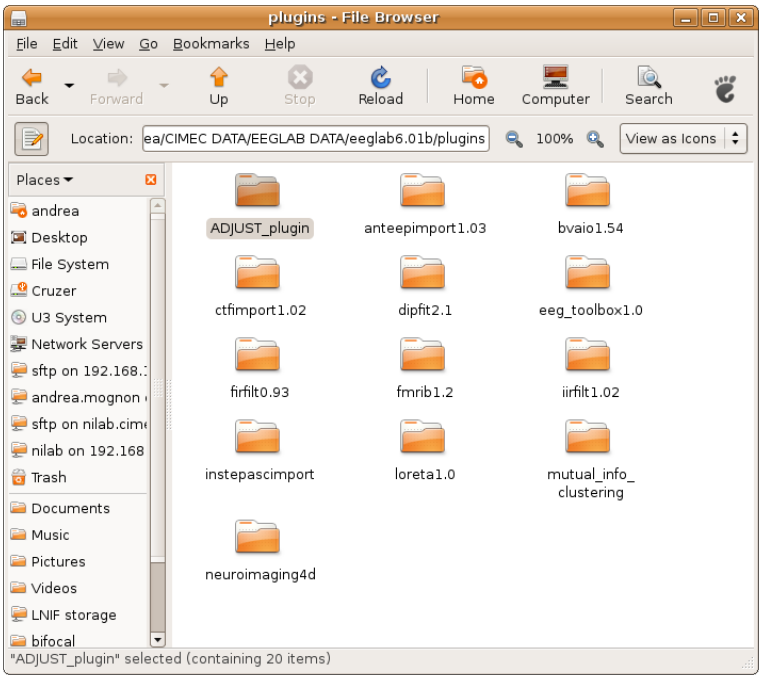


Figure 2.1: The EEGLAB *plugins* folder.

Chapter 3

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 `http://sccn.ucsd.edu/wiki/Eeglab`.

3.1 How to run EEGLAB

Open Matlab and set as Current Directory the folder where EEGLAB is installed; in my computer, it is
/home/andrea/CIMEC_DATA/EEGLAB_DATA/eeglab6.01b:

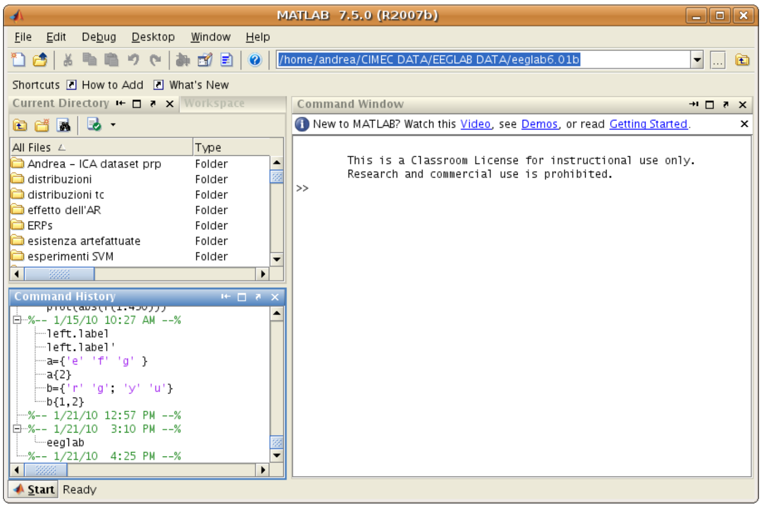


Figure 3.1: Set the Current Directory in Matlab.

Type 'eeglab' in the Command Window and the EEGLAB window will open as you can see in Figure 3.2.

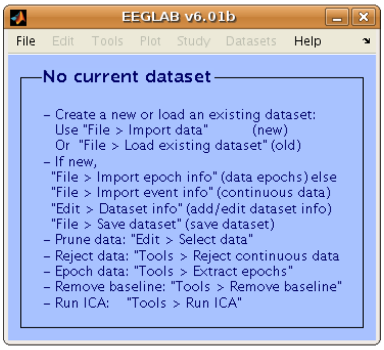


Figure 3.2: EEGLAB is running now!

3.2 Loading data

A dataset can be loaded by clicking in *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.) as in Figure 3.4.

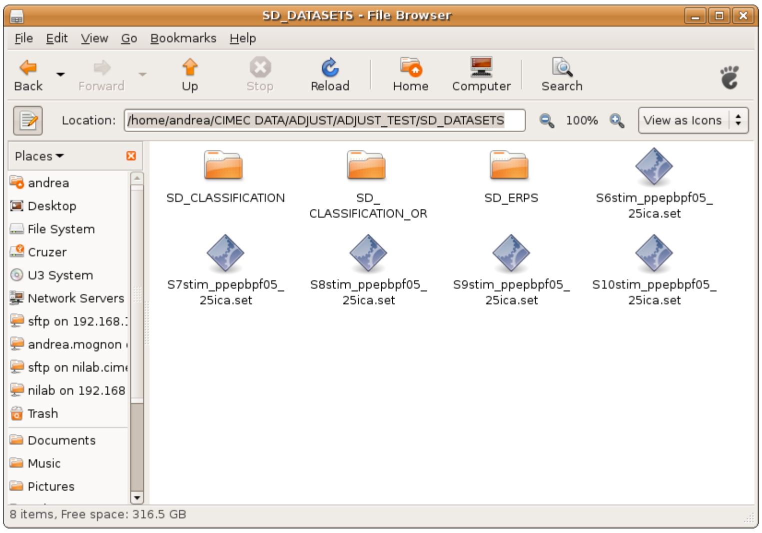


Figure 3.3: The dataset folder.

To load the dataset from the command line, type
\>> EEG = pop_loadset('dataset_name.set',directory);
\>> EEG = eeg_checkset(EEG);
Fot other loading modalities, see `http://sccn.ucsd.edu/eeglab/readtut/eeglabreadata.html`.

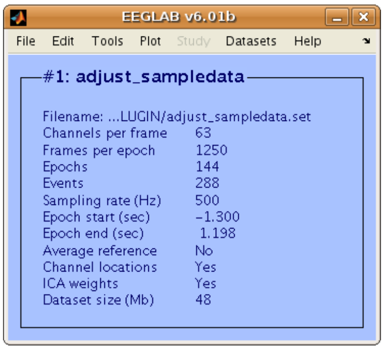


Figure 3.4: Our dataset is loaded.

Chapter 4

Running ADJUST for artifact detection

4.1 Pre-processing

EEG artifacts can be broadly divided in two classes: non-stereotyped artifacts due to movements of the electrodes on the scalp arising from large muscle movements or external sources, and stereotyped artifacts, mainly due to ocular eye movements and blinks (Onton, Westerfield, Townsend, & Makeig, 2006). 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, the ADJUST plugin also contains two simple pre-processing steps to obtain a clean ICA decomposition prior to artifact detection.

To start ADJUST, go to the EEGLAB menu and click *Tools* → *ADJUST*. The following list menu will open:

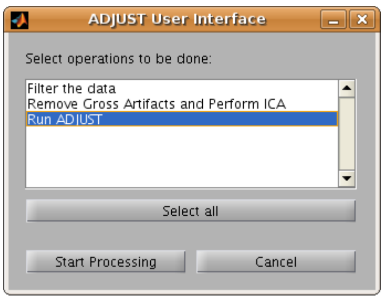


Figure 4.1: ADJUST main menu.

From the list menu you can choose three possible actions:

- filter the data;
- remove gross artifacts and perform ICA;
- run ADJUST.

The first two actions concern the pre-processing and computation of the ICA decomposition.

1. If you are not interested in very low-frequency or high-frequency components, spectral band-pass filtering might enhance the quality of the ICA decomposition. The filtering option proposed by the ADJUST menu is equivalent to selecting *Tools → Filter the data* from the EEGLAB menu. In the tutorial dataset, **spectral filtering is already performed** (Band pass filter between 0.5 and 25 Hz).

2. In order to remove data containing non-stereotyped "gross artifacts", epochs in which the EEG signal overcomes the voltage threshold of 150 V in more than 33% of the channels are removed. As stated above, this step is fundamental for a clean ICA decomposition. **Gross artifact removal is already performed on tutorial dataset.**

3. Independent Component Analysis (ICA) is the same performed by EEGLAB from the EEGLAB menu *Tools → Run ICA* from EEGLAB menu or by typing
   >> EEG = pop_runica(EEG);
   on Command Window. Type
   >> help pop_runica
   for the function optional inputs. **ICA is already performed on tutorial dataset.**
   Note: running ICA is a very time-consuming step of the algorithm. Large datasets may require very high computational power.

If you decide to compute these steps separately, you can directly proceed to run ADJUST.

You can choose the steps you wish to perform on the dataset by selecting them in the list and pressing the 'Start Processing' button. Use 'Ctrl' to select more than one entry; press 'Select all' to select all entries. More details on the pre-processing procedure in the Appendix.

4.2 Artifact detection

To identify artifacted ICs, ADJUST automatically computes a set of spatial and temporal features for each IC. Please refer to the reference paper Mognon et al. (2010) for details. Each artefact class is associated with one spatial and one temporal feature:

1. Eye blinks:

    - Spatial Average Difference (SAD) - a measure specifically sensitive to higher amplitude in frontal areas compared to posterior areas;
    - Temporal Kurtosis (TK) - computes the kurtosis over the IC time course, a measure that is very sensitive to outliers in the amplitude distribution typical of blinks.

2. 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 slower fluctuations typical of vertical eye movements.

3. Horizontal eye movements:

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

4. Generic discontinuities:

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

You can run ADJUST on the loaded dataset from the *Tools* → *ADJUST* list menu: select 'Run ADJUST' and press 'Start Processing'. Please note again that ICA weights must be computed before running the algorithm; if they are not present, the program cannot work. The program will ask you for a **report file name**: report file is a text file where analysis parameters and list of artifacted ICs will be written. It will be saved in the current directory. Specify any name you prefer (default is 'report.txt') and press OK:

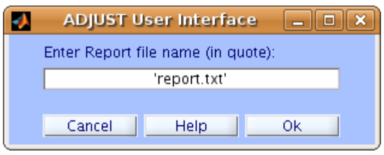


Figure 4.2: Enter report name.

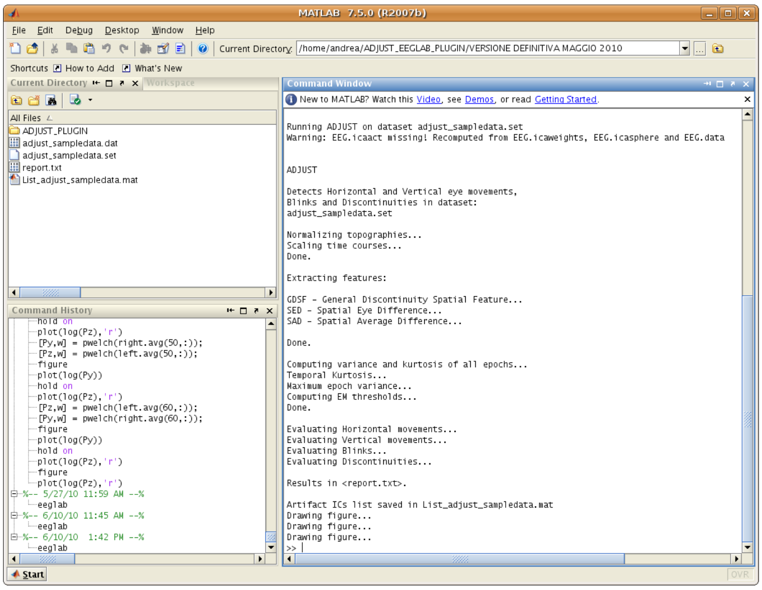


Figure 4.3: ADJUST comments on command window.

While running, ADJUST displays the key informations about the processing on your command window (Figure 4.3).

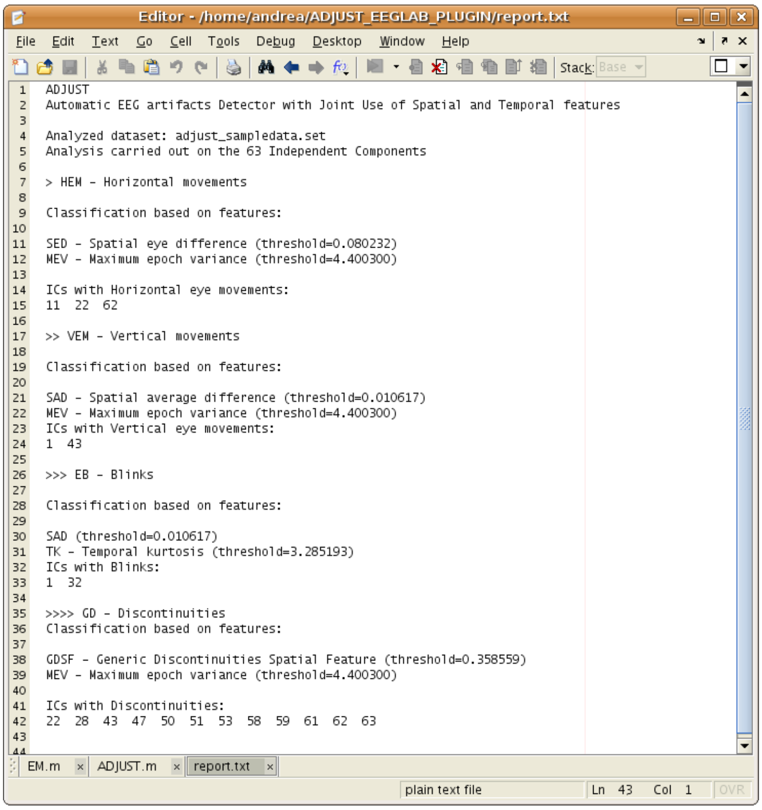
First of all, ADJUST displays the dataset name; then it tracks the computation of all the features and the respective thresholds; finally, it identifies the artifacted ICs, displays the names of the report file and the Matlab binary file and draws some figures illustrating the results.

- ***.mat file** is a Matlab structure including four variables:

  1. 'blink' contains the ICs indexes associated to blinks;
  2. 'vert' contains the ICs indexes associated to vertical eye movements;
  3. 'horiz' contains the ICs indexes associated to horizontal eye movements;
  4. 'disc' contains the ICs indexes associated to local discontinuities.

  You may want to read these data to play with these indexes in further analyses. To read them, type:
  \>> load 'MAT_file_name.mat'
  on your command window.

- **Report file** is a text file. You can open it into Matlab editor by double-clicking from the 'Current directory' browser. Report includes the list of artifacted ICs detected by the algorithm and the parameters used for their computation: features used by each detector and threshold values computed for the features.



```
ADJUST
Automatic EEG artifacts Detector with Joint Use of Spatial and Temporal features

Analyzed dataset: adjust_sampledata.set
Analysis carried out on the 63 Independent Components

> HEM - Horizontal movements

Classification based on features:

SED - Spatial eye difference (threshold=0.080232)
MEV - Maximum epoch variance (threshold=4.400300)

ICs with Horizontal eye movements:
11  22  62

>> VEM - Vertical movements

Classification based on features:

SAD - Spatial average difference (threshold=0.010617)
MEV - Maximum epoch variance (threshold=4.400300)
ICs with Vertical eye movements:
1  43

>>> EB - Blinks

Classification based on features:

SAD (threshold=0.010617)
TK - Temporal kurtosis (threshold=3.285193)
ICs with Blinks:
1  32

>>>> GD - Discontinuities
Classification based on features:

GDSF - Generic Discontinuities Spatial Feature (threshold=0.358559)
MEV - Maximum epoch variance (threshold=4.400300)

ICs with Discontinuities:
22  28  43  47  50  51  53  58  59  61  62  63
```

Figure 4.4: The report file.

4.3 Visualization of the results and artifact removal

At the end of ADJUST computation, a number of **figures** will open up to illustrate the results. Figures show the topographies of all ICs (one figure shows a maximum of 35 ICs); the ICs detected as artifacted by ADJUST are marked by highlighting in red the panel displaying the IC number at the top of the IC topography. Figure 4.5 illustrates the first 35 ICs computed on the tutorial dataset. In this example, ADJUST detected 5 artifacted ICs: 1, 11, 22, 28 and 32.

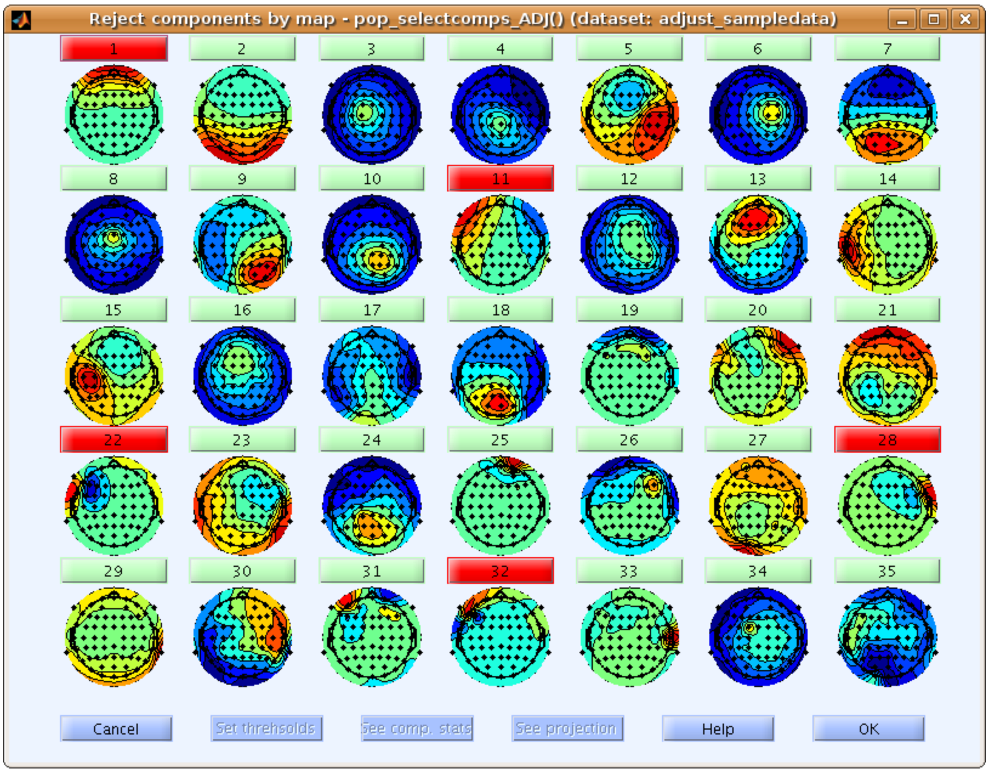


Figure 4.5: IC topographies; the ones detected as artifacted by ADJUST have their numbers marked in red.

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 number on top of its topography. A new figure will open (see Figure 4.6 as an example) displaying at the top the IC topography and the 'ERP image' (Jung et al., 2000), a useful graphic representation showing the IC time course of all epochs within the same figure by coding amplitudes in colour scale; the power spectra of the IC is shown in the center; the values of the features computed by ADJUST are reported at the bottom with respect to their own threshold value, (when the value crosses the threshold, that feature indicates an artifactual behavior).

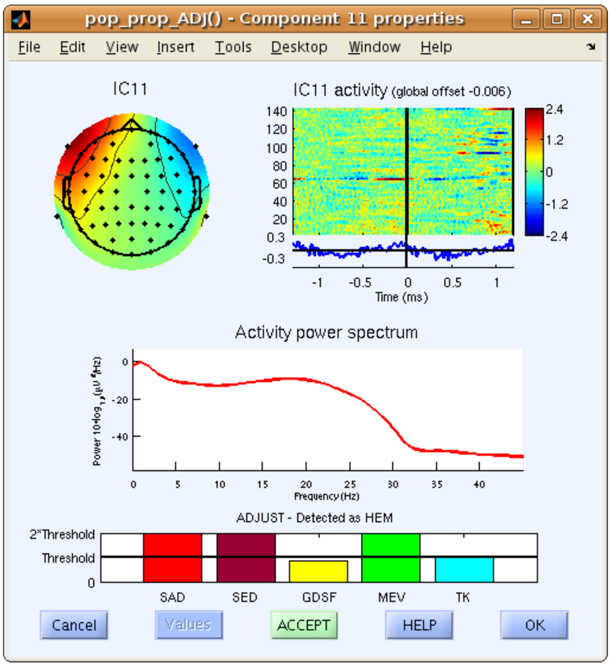


Figure 4.6: Properties of IC9, classified by ADJUST as a Horizontal Eye Movement

The IC shown in Figure 4.6 is classified by ADJUST as an horizontal eye movement (HEM) because both features associated with HEM, SED (Spatial Eye Difference) and MEV (Maximum Epoch Variance), clearly cross the threshold.

As another example, Figure 4.7 displays the properties of an IC of clearly neural origin, as it is evident from the strictly time-locked evoked potential in the ERP image, the posterior topography and the smoothly decaying power spectrum. In this case, the value of one temporal feature (TK) slightly 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.

Now, let's assume that you are checking ADJUST's results and you want to reject the indicated ICs; what you have to do is to click on the 'ACCEPT' button you can see at the bottom of the individual figure, and it will change into a pink-colored button 'REJECT' meaning that you marked the IC for rejection. See for example Figure 4.8.

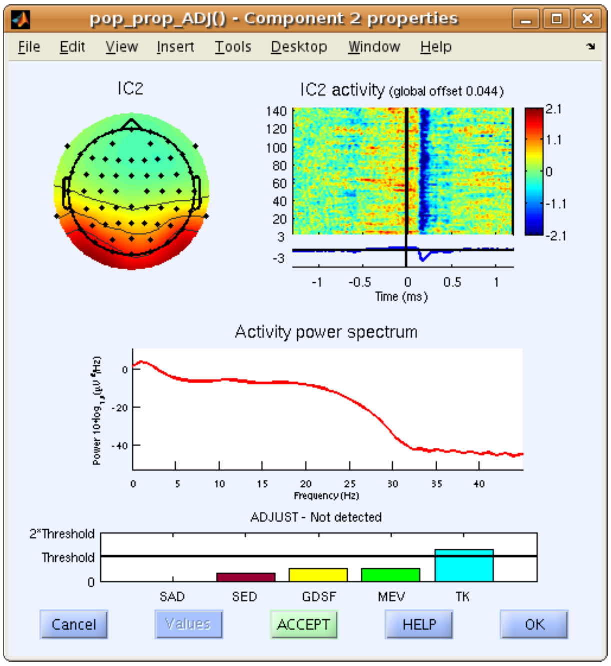


Figure 4.7: IC of neural origin.

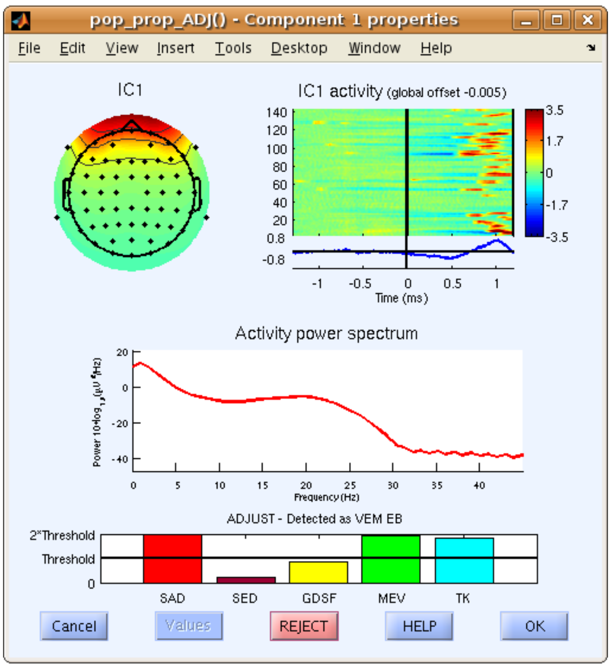


Figure 4.8: This IC has been detected as a blink, and is marked for rejection: note the REJECT button.

Mark all the ICs you wish to reject from data and press 'OK'. If you are using the tutorial dataset and mark for rejection all the ICs detected as artifacted by ADJUST, you will obtain the following Figure 4.9.

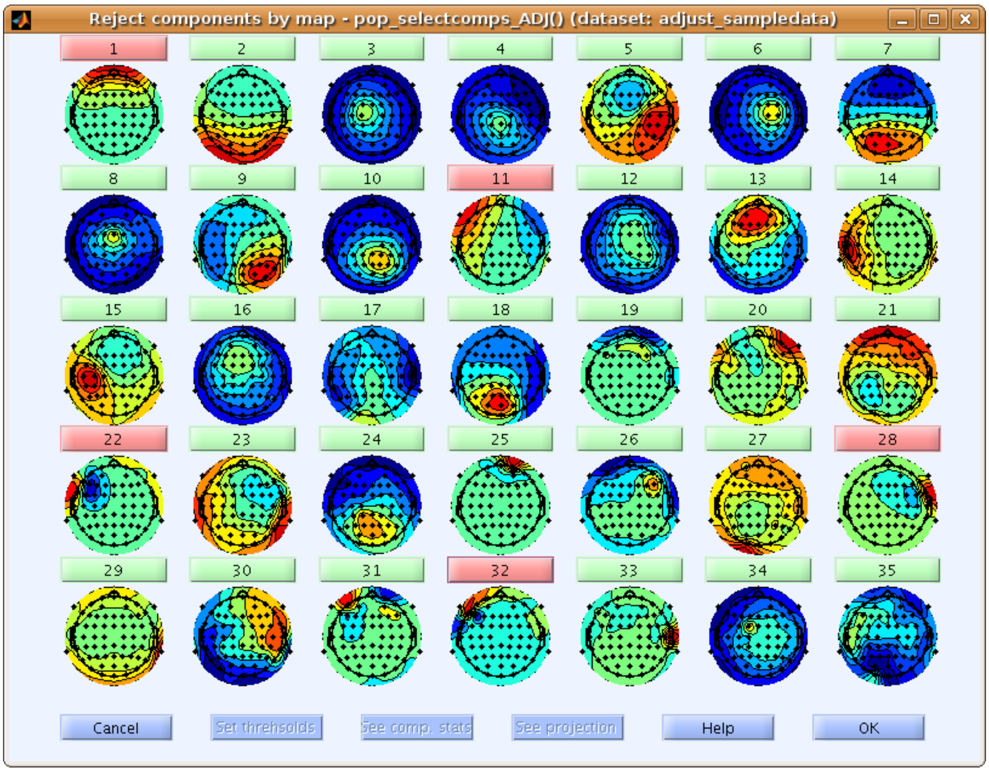


Figure 4.9: Artifact ICs are marked.

When done, press OK and the window will close. At this point, go to *Tools* → *Remove components* to check your selection of ICs to reject (Figure 4.10).

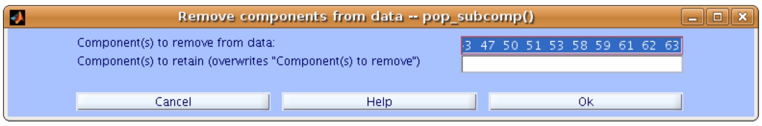


Figure 4.10: Selection of ICs to reject.

Press 'OK' to remove the ICs from the data. EEGLAB will ask for confirmation; press Accept.

Then save the new dataset.

Now the *bad* components have been removed from the dataset; check out the new time courses: the effect of artifacts should have disappeared!

Figure 4.11: If you are sure about the selection, press 'Accept'.

Chapter 5

References

1. 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, in press (2010).

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

3. Delorme, A., Makeig, S., *EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis.* Journal of Neuroscience Methods 134, 9-21 (2004).

4. Onton, J., Westerfield, M., Townsend, J., Makeig, S., *Imaging human EEG dynamics using independent component analysis.* Neuroscience and Biobehavioral Reviews, 30(6), 808-822 (2006).

Appendix A

Other features

A.1 Data preprocessing

Let's assume you have a raw dataset you wish to process with ADJUST - nor spectral filtering neither gross artifact removal done yet. You can use ADJUST plugin to process it. From the *Tools* → *ADJUST* list menu select the first two entries: 'Filter the data' and 'Remove Gross Artifacts and Perform ICA' (keep Ctrl pressed to select both of them). Then start processing.
The first interactive window appears and shows you cutoff frequencies for spectral filtering. You can specify either a lower edge or a higher edge or both of them. In this example, I choose both of them: 0.5 and 40 Hz, as you can see in Figure A.1.

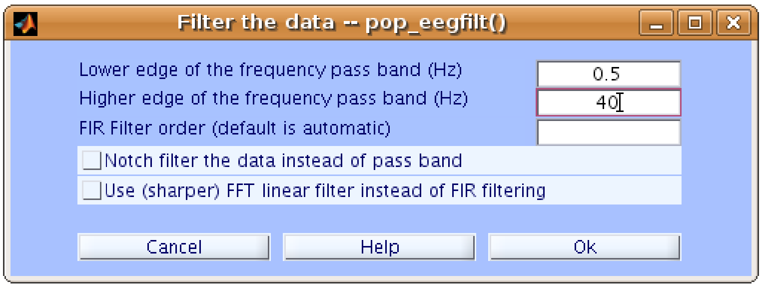


Figure A.1: Spectral filtering menu.

Press 'OK' and let it work. When filtering is over, ADJUST wants to know whether to save the new dataset. I choose to save it into memory since it's an intermediate step. Just print the name and press 'OK' (Figure A.2).

Now ADJUST removes Gross artifact portions from data. It should be quick; when it's done, the message in Figure A.3 is shown.

Again we are asked what to do with data. I decide to save it in memory again just like I did before. In the end ICA is performed, and this will usually require a lot of time. When done, a notification appears telling you ICA has

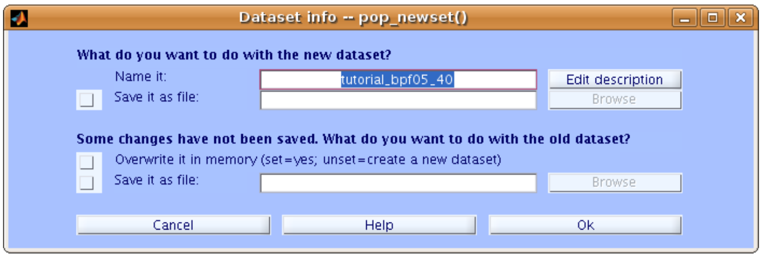


Figure A.2: ...Save the data...

Figure A.3: Gross artifacts are gone.

finished and the usual saving window is shown. I suggest to save the dataset on disk this time since preprocessing is over.

A.2 Benchmark version

EEGLAB version 6.01b on Matlab 7.5.0 (R2007b).

OS: Ubuntu Linux 8.04 (hardy)... but it works on Windows, too.

This pdf has been realized using LaTeX.