Table Of Contents

Published on *elan* (http://elan.lyon.inserm.fr)

EEG Processing Tools

eeg2asc

- ***Description***

  Extracts data from an **.eeg** files and writes to a text file. The data is epoched around events defined in the parameter file.

- ***Usage***

  eeg2asc myeegfilein.eeg myposfile.pos myparfile.par myasciifileout
  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - myposfile.pos : input event file (with extension).
  - myparfile.par : input parameter file (with extension).
  - myasciifileout : output text file (with extension) containing concatenated epochs of data.
  - options:
    +v : verbose mode on. If omitted, verbose mode is off.

- ***Fields of parameter file and examples***

nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself.
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to average: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be averaged and stored in the output .p files.
ep_channel_ref 0 5 0 6 0 0 0	List of the new reference for each channel before averaging (bipolar montage for instance): 0: no change of the reference, ≠0: electrode number (rank) to which the current channel should be re-referenced. The total number of values is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. This field is optional. If omitted, the channels are not modified. In this example, N=5, and channel 1 is unchanged, channel 2 is referenced to channel 5, and channel 4 re-referenced to channel 6.

- ***Example***

- ***Comments***

- ***Current version***

  1.04 17-01-2012

- ***History***
  - 1.03 29-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.
  - 1.04 17-01-2012 (PEA) : fixes an error on event counters.

- ***Files***

  $ELANPATH/bin/eeg2asc

- ***See also***

eegavg

- ***Description***

  Computes average evoked potentials from an ***.eeg*** file, with artifact rejection. The output is stored in ***.p*** file format (one ***.p*** file by event code).

- ***Usage***

  eegavg *myeegfile**.eeg*** *myposfilein**.pos*** *myparfile**.par*** *myposfileout**.pos*** [+v] [+s] [+z] [+z2] [+t] [+allchannels] [+noreref] [+norejection]

  with :

  - *myeegfile**.eeg***: input ***.eeg*** file to process (with extension).
  - *myposfilein**.pos***: input event file (with extension).
  - *myparfile**.par***: text file containing average and rejection parameters (with extension).
  - *myposfileout**.pos***: output event file (with extension). It contains the events marked with a rejection code or not.
  - options:
    +v : verbose mode on. If omitted, verbose mode is off.
    +s : creates an ***.sem.p*** file with the standard error of the mean (sem) across trials, and an ***.T.p*** file with t-test T values for comparison to baseline level (mean/sem). If omitted, files are not created.
    +z : creates a ***.Z.p*** file with a Z-transform of the data computed across trials with respect to the baseline : (data – mean baseline)/standard deviation of baseline. If omitted, the file is not created.
    +z2 : creates an ***.Z2.p*** file with a Z-transform of the squared data computed across trials with respect to the baseline : (squared data – mean squared baseline)/standard deviation of squared baseline. If omitted, the file is not created.
    +allchannels : computes on all input data file's channels. The "ep_channel_flag" field of parameter file isn't read.
    +noreref : don't change reference of input data channels. The "ep_channel_ref" field of parameter file isn't read.
    +norejection : accept all trials without processing any rejection. The "ep_channel_rejtype" field of parameter file isn't read.

- ***Fields of parameter file and examples***

fileprefix myfilename	Prefix of the output .p files.
nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
flag_eventcode 1 0	List of flags on the event codes: 1: event code is processed (rejection and averaging) 0: event code is written in myposfileout.pos, but not processed. This field is optional.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the

	extra sample corresponds to the event itself.
baseline_eventcode 2 2	List of the event codes used to compute the baseline. 0: no baseline correction (default), ≠0: event code on which baseline will be computed. In this example, the baseline for event code 2 will be computed from event code 2, and the baseline of event code 5 from event code 2. This field is optional.
baseline_msec_start -200 -200	List of the baseline start latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
baseline_msec_stop -50 -50	List of the baseline stop latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to average: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be averaged and stored in the output .p files.
ep_channel_ref 0 5 0 6 0 0 0	List of the new reference for each channel before averaging (bipolar montage for instance): 0: no change of the reference, ≠0: electrode number (rank) to which the current channel should be re-referenced. The total number of values is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. This field is optional. If omitted, the channels are not modified. In this example, N=5, and channel 1 is unchanged, channel 2 is referenced to channel 5, and channel 4 re-referenced to channel 6.
ep_channel_rejtype 0 0 0 1 0 0 0	List of the rejection types per channel : 0, 1 or 2 0: no artifact rejection (default) 1: trial rejection when the amplitude of one of the selected channel exceeds a specified threshold (value given by rej_max_val and rej_min_val; the trial is rejected if any sample is outside the interval [rej_min_val rej_max_val]); 2: trial artifact rejection when, for one of the selected channel, the amplitude range in a time window exceeds a specified threshold (amplitude specified by rej_range_val and time-window by rej_time_window); event rejected if the signal has an amplitude range larger than rej_range_val in any time-window of rej_time_window duration. The total number of flags should be N+2 as in ep_channel_flag (last 2 channels set to 0); artifact rejection will be active only on channels selected in the ep_channel_flag_list. Mixing artifact rejection types 0, 1 and 2 is possible for different channels.
rej_max_val 0 0 0 15 0 0 0	List of maximum threshold values per channel (physical units (microV, fT, ...)) for amplitude artifact rejection (ep_channel_rejtype = 1). The total number of values should be N+2 (last 2 channels set to 0).
rej_min_val 0 0 0 -20 0 0 0	List of minimum threshold values per channel (in physical units (microV, fT, ...)) for amplitude artifact rejection (ep_channel_rejtype = 1). The total number of values should be N+2 (last 2 channels set to 0). If omitted, the minimum amplitude threshold values are set to - rej_max_val.
rej_range_val 0 0 0 30 0 0 0	List of threshold values per channel (in physical units (microV, fT, ...)) for amplitude range artifact rejection (ep_channel_rejtype = 2). This threshold is applied either on the whole trial duration (if rej_time_window=0) or on a moving window (if rej_time_window≠0). (The total number of values should be N+2 (last 2 channels set to 0).
rej_range_val_min 2 2 2 2 2 2 2	List of upper threshold values per channel (in physical units (microV, fT, ...)) for minimum signal variation required in the rejection window (ep_channel_rejtype = 2). This allows to reject trials with “flat” signals (amplitude variation below 2 µV in this example) corresponding to saturating amplifiers. The total number of values should be N+2 (last 2 channels set to 0). If omitted, the rejection on this upper threshold is not applied.
rej_time_window 0 0 0 50 0 0 0	List of time-window length per channel (in ms) for amplitude range artifact rejection (ep_channel_rejtype = 2). The total number of flags is N+2 (last 2 channels set to 0). If omitted, rejection on amplitude range will be performed on the whole epoch.
rej_beg_lat_msec 100 50	List of starting latencies (in ms) to perform rejection for each event. If omitted, rejection will be performed from the beginning of the epoch.
rej_end_lat_msec 500 250	List of ending latencies (in ms) to perform rejection for each event. If omitted, rejection will be performed until the end of the epoch.

erpa_positivity_up 1	Value stored in all output .p file to set the default upward/downward positivity orientation for curve display (by erpa). -1 (positivity down) or 1 (positivity up) This field is optional. If omitted, erpa will display curve with positivity down.
erpa_view 4	Value stored in all output .p file. It is used by erpa to set the default mapping view. 1: right view 2: left view 3: top view 4: back view 5: large top view 6: back-top view This field is optional. If omitted, it is set to 0, and erpa default will be top view.

Output:

- averaged evoked responses (ep files ***.p***) (event codes 002 and 005 in this example):
  *myfilename.002****.p***
  *myfilename.005****.p***
- text file with all the output displayed on the screen by the program in verbose mode :
  myparfile.par.res
- if option +s: standard error of the mean across trials (.p files) (event codes 002 and 005 in this example) :
  myfilename.002.sem.p
  myfilename.005.sem.p
- if option +s: t-test T values for comparison to baseline level (mean/sem) (.p files) (event codes 002 and 005 in this example) :
  myfilename.002.T.p
  myfilename.005.T.p
- if option +z: Z-transform of the data with respect to the baseline (.p files) (event codes 002 and 005 in this example) :
  myfilename.002.Z.p
  myfilename.005.Z.p
- if option +z2: Z-transform of the squared data with respect to the squared baseline (.p files) (event codes 002 and 005 in this example) :
  myfilename.002.Z2.p
  myfilename.005.Z2.p

- ***Examples***
  - baseline correction :
    fileprefix example1
    nb_event_code 3
    list_event_code 80 2 44
    prestim_nbsample 400 100 200
    poststim_nbsample 400 400 400

    baseline_eventcode 80 44 0
    baseline_ms_start -200 -100 0
    baseline_ms_stop -100 0 0

    ep_channel_flag 1 1 1 1 1 1 1 1 0 0
    ep_channel_rejtype 0 0 0 0 0 0 0 0 0 0

    In this example:
    - response to the event code 80 will be baseline corrected using the mean value between -200 and -100 ms prior to event 80,
    - response to the event code 2 will be baseline corrected using the mean value between -100 and 0 ms prior to event 2,
    - response to the event code 44 will not be baseline corrected.

  - artifact rejection :
    fileprefix example2
    nb_event_code 1
    list_event_code 80
    prestim_nbsample 400

poststim_nbsample 400

baseline_eventcode 80
baseline_ms_start -100
baseline_ms_stop 0

ep_channel_flag 1 1 1 1 1 1 1 1 0 0
ep_channel_rejtype 0 0 0 0 0 0 0 1 0 0
rej_max_val 0 0 0 0 0 0 0 30 0 0

In this example, all events with the code 80 will be averaged on the 7 acquired channels, except those for which the signal in channel # 7 is more than 30 uV or less than -30 uV, from 400 samples prior to 400 samples after event 80.

- artifact rejection :
fileprefix example3
nb_event_code 1
list_event_code 80
prestim_nbsample 200
poststim_nbsample 400

baseline_eventcode 80
baseline_ms_start -100
baseline_ms_stop 0

ep_channel_flag 1 1 1 1 1 1 1 1 0 0
ep_channel_rejtype 2 2 2 2 2 2 2 2 0 0
rej_range_val 30 30 30 30 30 30 30 30 0 0
rej_time_window 10 10 10 10 10 10 10 10 0 0

In this example, all events with the code 80 are averaged, except those for which the signal in any channel has an amplitude range larger than 30 uV on a 10 ms moving time-window. The time-window is shifted by 1 sample step from 200 samples prior to 400 samples after event 80.

- ***Comments***
  1. Note to CTF 275 MEG users : an example of parameter file with 275 channels is available to download (ctf275_meg.par [1])
  2. See eegchref [2] to create a re-referenced .eeg data file (several referencing options available).

- ***Current version***

1.51 19-02-2014

- ***History***
  - 1.31 13-01-2006 (OB) : bug fix for baseline correction base on another event.
  - 1.32 13-01-2006 (PEA) : bug fix for EEG file size > 2GB.
  - 1.33 29-03-2006 (PEA) : modification for 64 bits compilation.
  - 1.35 31-07-2006 (PEA) : change in rejection. Output event file contains averaged and rejected events (with rejection code).
  - 1.36 17-07-2006 (PEA) : bug fix for rejection code (power of 2). When multiple rejection case for one event, operation between codes is OR.
  - 1.37 14-03-2007 (PEA) : add "+t" command line option. If present, rejection results are written to a text file (fileprefix.rej.out.txt).
  - 1.38 01-08-2007 (PEA) : bug fix : sum in an array of double instead of int (causes overflow).
  - 1.39 13-08-2007 (PEA) : minor modification.
  - 1.40 10-11-2009 (PEA) : bug fix for rejection for dynamic saturation (overflow int values for 32 bits EEG V3).
  - 1.41 05-02-2010 (PEA) : minor modification (use fprintf instead of printf, and make difference between stderr and stdout).
  - 1.42 22-02-2010 (PEA) : correction for reading all events (not just good events) to have all events in output event file.
  - 1.43 23-09-2010 (PEA) : update to use cmake and free release of Elan.
  - 1.44 07-01-2011 (PEA) : fixes buffer overflow and memory problems.

- 1.45 23-10-2012 (PEA) : remove all static allocations. No maximum nuber of channels.
- 1.46 24-10-2012 (PEA) : add +allchannels, +noreref and +norejection options to force compute on all channels, no reference change and no rejection without reading corresponding fields of parameter file.
- 1.47 11-02-2013 (PEA) : fixes a memory error when storing baseline values when no baseline correction is performed.
- 1.48 19-07-2013 (PEA) : fixes an error when rereferencing a channel with a channel with higher amplitude data may cause data overflow.
- 1.49 11-09-2013 (PEA) : fixes an error in rejection due to last modification. Fixes a memory deallocation (output data array).
- 1.50 16-09-2013 (PEA) : fixes an error in baseline correction between events (baseline_eventcode field of parameter file).
- 1.51 19-02-2014 (PEA) : fixes an error for dynamic rejection (type 2) and specific rejection window (rej_beg_lat_msec and rej_end_lat_msec). Bad rejection window was used.

- ***Files***

  $ELANPATH/bin/eegavg

- ***See also***

  eegstat [3], eegchref [2]

Attachment	Size
ctf275_meg.par [4]	7.67 KB

eegcalib

- ***Description***

  Calibrates an EEG file by using a sinusoidal signal of known amplitude on some channels.

- ***Usage***

  eegcalib myeegfilein.eeg physical_value calib_signal_freq rank1 rank2 ... rankn
  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - physical_value : physical peak-to-peak value (microV or any unit) of calibration signal.
  - calib_signal_freq : calibration signal frequency.
  - rank1 rank2 ... rankn : list of channel indices containing calibration signal (first is number 1).

- ***Fields of parameter file and examples***

- ***Example***

- ***Comments***

- ***Current version***

  1.02 29-09-2010

- ***History***
  - 1.02 29-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.

- ***Files***

  $ELANPATH/bin/eegcalib

- ***See also***

eegchref

- ***Description***

  Tool for changing reference of continuous data. The new reference can be a channel or average reference.

- ***Usage***

  eegchref myeegfile_in.eeg myparfile myeegfile_out.eeg

  with :

  - myeegfile_in.eeg : input file to change reference (with extension).
  - myparfile : text file containing reference parameters.
  - myeegfile_out.eeg : output file (with extension).

- ***Fields of parameter file and example***

ref_type 0 0 0 3 0 0 -1 0 0	Type of reference to apply. Values can be : 0: no change. -1: average reference. In this case, the field ref_avg_def is required. any other value: indice of channel used as reference. The total number of values is N+2, N being the number of recorded channels in myeegfile_in.eeg file; the last 2 flags should be set to 0.
ref_avg_def 0 0 1 1 1 0 1 0 0	List of channel to use for average reference: 1/0 for selected/unselected channels. The total number of flags is N+2, N being the number of recorded channels in myeegfile_in.eeg file; the last 2 flags should be set to 0.

- ***Example***

- ***Comments***

- ***Current version***

  2.25 02-06-2017

- ***History***
  - 2.22 08-01-2004 (PEA) : 1st documented version.
  - 2.23 08-12-2010 (PEA) : correction for overflow of digital value after reference change.
  - 2.24 14-02-2014 (PEA) : correction for self reference channel (null value for physical min/max).
  - 2.25 02-06-2017 (PEA) : minor modification: test for parameter file existence added.

- ***Files***

  $ELANPATH/bin/eegchref

- ***See also***

eegcomponent

- ***Description***

  Computes component analysis from continuous data and and outputs inverse of weight matrix.
  The analysis may be processed on epochs (around events) or on a part of the file by defining beginning and ending latencies.

- ***Usage***

  eegcomponent myeegfilein.eeg myparameterfile.par mymatrixout.xml [myeventfile.pos]

  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - myparameterfile.par : parameter file (with extension) containing computing parameters.
  - mymatrixout.xml : output XML matrix file (with extension) containing inverse matrix of weights.
  - options:
    myeventfile.pos : input event file (with extension) used for epoching data. If omitted, whole file is used to compute component analysis.

- ***Fields of parameter file and example***

component_analysis_channel 1 1 0 1 0 0 0	List of the channels to use for component analysis: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be used.
component_analysis_method 0	Used method to compute component analysis. Valid values are : 0 : PCA 1 : SVD 2 : ICA using FASTICA algorithm (see http://research.ics.tkk.fi/ica/fastica/ [5])
nb_eventcode 2	Number of event code to use for component analysis processing.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code.
baseline_msec_start -200 -200	List of the baseline start latencies (in ms); one value for each event code. If omitted, the baseline value is not computed.
baseline_msec_stop -50 -50	List of the baseline stop latencies (in ms); one value for each event code. Required if baseline_msec_start is specified. If omitted, the baseline value is not computed.
first_sample 100	Sample number (from beginning of file) of data beginning to process component analysis.
sample_nb 5000	Number of time samples to use for component analysis.

  Use parameters when processing component analysis from an epoch list.
  Use parameters when processing component analysis from a part of file only.

- ***Examples***

- ***Comments***

- ***Current version***

  1.01 03-04-2012

- ***History***

- 1.00 05-03-2012 (PEA) : 1st version.
- 1.01 03-04-2012 (PEA) : adds FASTICA method.

- ***Files***

  $ELANPATH/bin/eegcomponent

- ***See also***

  eegproject [6], eegfiltica [7], matrix2p [8]

eegconcat

- ***Description***

  Concatenates .eeg files having the same number of electrodes and the same sampling frequency, but not necessarily the same gain. In case of .eeg files having different gain factors, all numerical data are scaled to correspond to the gain factors of the first .eeg file.

- ***Usage***

  eegconcat [+pos]
  with :

  - option :
    +pos : concatenates event files associated to EEG files. If present, for each EEG file, the program asks for a .pos event file (with extension). If an EEG file has n o event file associated, press the Enter key when the program asks for the event file. If omitted, no event file is asked for.

- ***Interactive input***

  EEG file to concatenate (name without extension, hit Return to stop concatenation):
  myfile1
  EEG file to concatenate (name without extension, hit Return to stop concatenation):
  myfile2
  EEG file to concatenate (name without extension, hit Return to stop concatenation):
  Output EEG file (name without extension):
  mybigfile

  In this example mybigfile.eeg contains the data of myfile1.eeg followed by the data of myfile2.eeg.

- ***Example***

- ***Comments***
  1. WARNING : this program doesn't check for the electrode names and numbers (in elec.dat).

- ***Current version***

  1.10 23-09-2010

- ***History***
  - 1.06 17-09-2008 (PEA) : add +pos option to concatenate event files. Suppress maximum number of files to concatenate. Read/write EEG files by blocks of samples instead of one sample by one sample.
  - 1.07 05-12-2008 (PEA) : bug fix : looking for maximum/minimum values.
  - 1.08 le 10-09-2009 (PEA) : bug fix : storing maximum/minimum values.
  - 1.09 15-09-2009 (PEA) : bug fix : memory allocation.
  - 1.10 23-09-2010 (PEA) : update to use cmake and free release of Elan

- ***Files***

  $ELANPATH/bin/eegconcat

- ***See also***

eegcut

- ***Description***

  Creates a truncated **.eeg** file.

- ***Usage***

  eegcut myeegfilein.eeg myeegfileout.eeg samples_nb
  or
  eegcut myeegfilein.eeg myeegfileout.eeg 1st_sample last_sample
  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - myeegfileout.eeg : output EEG file (with extension).
  - samples_nb : number of samples to keep (from the beginning of file).
  - 1st_sample : indice of 1st sample to keep.
  - last_sample : indice of last sample to keep.

- ***Fields of parameter file and examples***

- ***Examples***
  1. Get the 1st 15000 samples of myeeg.eeg file and saves to myeeg.15000.eeg :

     eegcut myeeg.eeg myeeg.1-15000.eeg 15000

  2. Get 20000 samples starting at sample #10000 of myeeg.eeg file and saves to myeeg.10000-30000.eeg :

     eegcut myeeg.eeg myeeg.10000-30000.eeg 10000 30000

- ***Comments***

- ***Current version***

  1.04 10-03-2011

- ***History***
  - 1.02 05-05-2005 (OB) : 1st documented version.
  - 1.03 29-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.
  - 1.04 10-03-2011 (PEA) : adds 1st and last sample usage (gives beginning and ending of data in samples to keep).

- ***Files***

  $ELANPATH/bin/eegcut

- ***See also***

eegdelchan

- ***Description***

  Deletes channels from an EEG file and creates a new EEG file.

- ***Usage***

  eegdelchan myeegfilein.eeg myeegfileout.eeg del_chan_nb rank1 rank2 ... rankn
  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - myeegfileout.eeg : output EEG file to process (with extension).
  - del_chan_nb : number of channels to remove from myeegfilein.eeg.
  - rank1 rank2 ... rankn : list of channel indices to delete (first is number 1).

- ***Fields of parameter file and examples***

- ***Example***

  eegdelchan myeeg_in.eeg myeeg_out.eeg 2 3 5
  myeeg_out.egg contains all data from myeeg_in.eeg except channel 3 and 5.

- ***Comments***

- ***Current version***

  1.03 08-04-2011

- ***History***

  - 1.01 13-01-2004 (PEA) : 1st documented version.
  - 1.02 29-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.
  - 1.03 08-04-2011 (PEA) : minor modification (progression text).

- ***Files***

  $ELANPATH/bin/eegdelchan

- ***See also***

eegdetectstep

- ***Description***

  Detects on single-trials (epochs) an amplitude variation greater than a threshold in a time window. One results text file is created for each event type. It contains the event code, the event code indice, the channel number, the threshold number, the latency (starting from the event), the amplitude and duration, for each threshold detection.

- ***Usage***

  eegdetectstep myeegfilein.eeg myeventfile.pos myparameterfile.par [-e]
  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - myeventfile.pos : input event file (with extension) used to define epochs.

- myparameterfile.par : text file containing detection parameters (with extension).
- options :

  -e : no header is written in output file.

- ***Fields of parameter file and examples***

fileprefix myoutfilename	Prefix of the output text files.
nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself.
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to average: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be used to detect thresholds.
step_size_ms 100 200 150	Window size for amplitude variation detection for each channel used. In this example, the number of channels for the detection is 3 (indices 1, 2 and 4), and the window durations are 100 ms for channel 1, 200 ms for channel 2, and 150 ms for channel 4.
step_size_microV 1000.0 500.0 750.0	Physical value (in physical units, not only microV) of the amplitude variation for each channel used. In this example, the number of channels for the detection is 3 (indices 1, 2 and 4), and the amplitude variations are 1000 for channel 1, 500 channel 2, and 750 for channel 4. These amplitude variations are in physical units.

- ***Example***

- ***Comments***

- ***Current version***

  1.04 29-09-2010

- ***History***
  - 1.03 12-01-2004 (PEA) : 1st documented version.
  - 1.04 29-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.

- ***Files***

  $ELANPATH/bin/eegdetectstep

- ***See also***

eegdetecttrig

- ***Description***

  Detects pulses (triggers) on analog continuous channels of an EEG file, and creates a new EEG file with event codes stored in the digital channel. It requires a parameter file to define the pulse detection thresholds and event codes.

- ***Usage***

  eegdetecttrig [-?] eegfilein.eeg parameterfile.trig.par eegfileout.eeg

  with :

  - eegfilein.eeg : input EEG file (with extension) with pulses on some channels.
  - parameterfile.trig.par : parameter file (with extension) for pulse detection.
  - eegfileout.eeg : output EEG file (with extension) after pulse detection with event codes stored in the digital channel.
  - option :
    -? : to display details on the parameter file.

- ***Fields of parameter file and example***

trigger_nb 2	Number of analog channels with pulses (triggers).
trigger_channel 38 40	Channel numbers with trigger signals. In this example, channels # 38 and 40 contain pulses.
trigger_threshold 10 10	Threshold values (in microVolts or physical units) above (or below) which the trigger signals will be detected. In this example, threshold will be 10 microVolts for both channels.
trigger_rise_fall 1 1	Detection on the rising (1) or falling (-1) edge of the pulses. In this example, detection will be performed on the rising edge for both channels.
trigger_code 10 12	Digital codes (<=65535) that is generated for all pulses on each channel. In this example, trigger codes 10 and 12 are generated for pulses found on channels # 38 and 40 respectively.

- ***Example***

- ***Comments***

- ***Current version***

  1.18 23-09-2010

- ***History***
  - 1.15 01-12-2003 (PEA) : 1st version with documentation.
  - 1.16 05-06-2007 (PEA) : read/write files by block instead of sample.
  - 1.17 13-08-2007 (PEA) : minor modification.
  - 1.18 23-09-2010 (PEA) : update to use cmake and free release of Elan.

- ***Files***

  $ELANPATH/bin/eegdetecttrig

- ***See also***

eegdetecttrigl

- ***Description***

  Detects square pulses of a given duration on a channel of an **.eeg** file and creates events. Events are stored in a **.eeg** EEG file containing all channels. They may be extracted later with eegpos [9] to create a **.pos** text event file.

- ***Usage***

  eegdetecttrigl myeegfilein.eeg myparameterfile.par myeegfileout.eeg
  with :

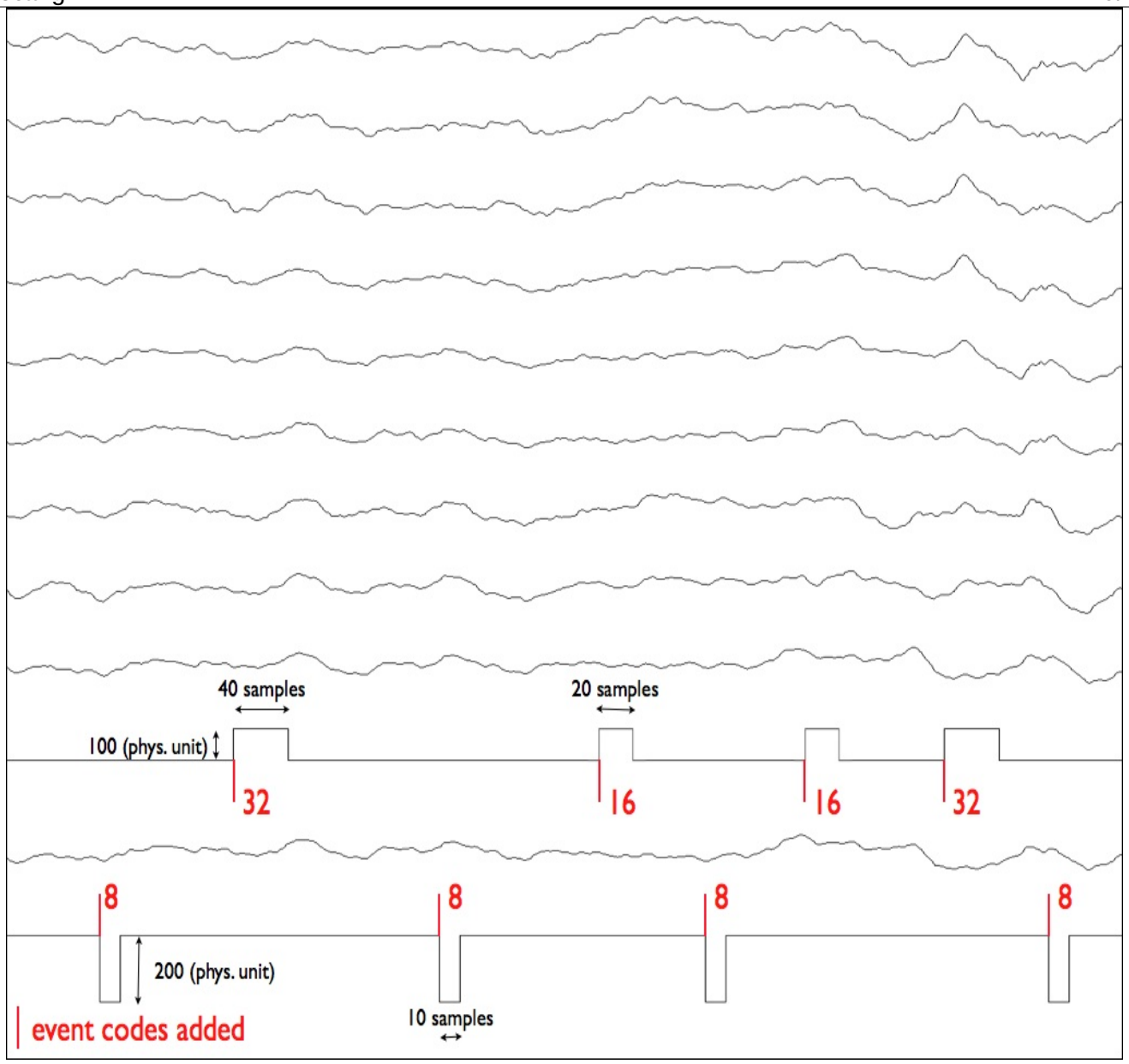
  - myeegfilein.eeg : input EEG file to process (with extension).
  - myparameterfile.par : input event file (with extension) used to define epochs.
  - myeegfileout.eeg : output EEG file (with extension).

- ***Fields of parameter file and examples***

trigger_chan_nb 2	Number of channel containing square pulses to detect.
trigger_chan_list 10 12	List of channel ranks containing the pulses to detect. The 1st channel has indice 1.
trigger_threshold 100 200	List of the amplitude threshold to detect square pulses, in physical units, for each channel.
trigger_edge 1 -1	List of the edge parameters to detect triggers. Possible values are : 1 : rising edge. -1 : falling edge.
trigger_nb 3	Number of triggers of different durations.
trigger_duration 10 20 40	Trigger durations (in samples) for each trigger type.
trigger_code 8 16 32	Trigger codes to store in EEG file for each trigger type.

- ***Example***

  The following picture illustrates the above parameter file :



[10]

- ***Comments***

- ***Current version***

  1.03 29-09-2010

- ***History***
  - 1.02 13-01-2004 (PEA) : 1st documented version.
  - 1.03 29-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.

- ***Files***

  $ELANPATH/bin/eegdetecttrigl

- ***See also***

Attachment	Size

fig_eegdetecttrigl.png [10] 101.75 KB

eegepochfiltfilt

- ***Description***

  Computes zero-phase filtering (forward and reverse) on analog continuous channels of an EEG file containing concatenated epochs, and creates a new EEG file. As input parameters, it can use B and A coefficients computed with Matlab (or SciPy, Octave, Scilab), or it can compute Butterworth filter coefficients from cutoff frequencies. It can compute band-pass, band-stop, low-pass or high-pass filtering.
  When computing Butterworth coefficients, a test for filter stability is processed. If the filter is unstable, no filtering is computed and a message for the new filter order to apply is displayed.
  The output EEG file contains only concatenated epochs. The epochs are made from event codes and pre-stimulus and post-stimulus number of samples.

- ***Usage***

  eegepochfiltfilt myeegfilein.eeg myparameterfile.par myevent.pos myeegfileout.eeg

  with :

  - myeegfilein.eeg : input EEG file to filter (with extension).
  - myparameterfile.par : filtering parameter file (with extension).
  - myevent.pos : event file used for epoching (with extension). It contains the events marking zero time of epochs.
  - myeegfileout.eeg : output (filtered) EEG file (with extension).

- ***Fields of parameter file and example***

nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself.
filter_channel 1 1 0 1 0 0 0	List of the channels to filter: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be filtered but all channels are stored in the output files.
filter_type 0	Type of Butterworth filter. Valid values are : 0 for low-pass filter 1 for high-pass filter 2 for band-pass filter 3 for band-stop filter In this example, a low-pass filter is processed.
filter_order 3	Butterworth filter order. In this example, the filter order is 3.
filter_cutoff_freq1 30	Cutoff frequency 1. In case of filter type : 0 (low-pass) filter : attenuates higher frequencies. 1 (high-pass) filter : attenuates lower frequencies. 2 (band-pass) filter : lower frequency of the range of frequencies to pass. 3 (band-stop) filter : lower frequency of the range of frequencies to attenuate.
filter_cutoff_freq2 100	Cutoff frequency 2. Needed only for band-pass and band-stop filter (filter_type 2 or 3). In case of filter type : 2 (band-pass) filter : higher frequency of the range of frequencies to pass. 3 (band-stop) filter : higher frequency of the range of frequencies to attenuate.
filter_b_coeff_nb 4	Number of B coefficients (in the Matlab way). These coefficients can be of a

filter_b_coeff_nb 4	Butterworth filter, or any other filter.
filter_b_coeffs 2.71835675758059e-05 8.15507027274176e-05 8.15507027274176e-05 2.71835675758059e-05	List of B coefficients. **As it is text file, be careful of include a maximum of significant digits.** In this example, coefficients are corresponding to a low-pass Butterworth filter of order 3.
filter_a_coeff_nb 4	Number of A coefficients (in the Matlab way). These coefficients can be of a Butterworth filter, or any other filter.
filter_a_coeffs 1 -2.87730072411486 2.76201379931893 - 0.884495606663461	List of A coefficients. **As it is text file, be careful of include a maximum of significant digits.** In this example, coefficients are corresponding to a low-pass Butterworth filter of order 3.

Use parameters when working with a Butterworth filter and setting filter order, type and cutoff frequency(ies).
Use parameters when working with a coefficients computed by another way.

- ***Examples***

Both examples below are used to filter concatenated (epoched) data. The epoching is done for event codes 2 and 5.
The events with code 2 have 400 samples in the pre-stimulus and 1000 samples in the post-stimulus (the epochs contains 400+1+1000 samples).
The events with code 5 have 800 samples in the pre-stimulus and 1200 samples in the post-stimulus (the epochs contains 800+1+1200 samples).

1. This parameter file (lowpass_butter_10hz_epoch.par [11]) computes Butterworth coefficients from the filter definition, and filters all channels of an EEG file containing 10 channels:

```
nb_eventcode 2
list_eventcode 2 5

prestim_nbsample 400 800
poststim_nb_sample 1000 1200


#low_pass Butterworth filter
filter_type 0

filter_order 3

filter_cutoff_freq1 10

filter_channel 1 1 1 1 1 1 1 1 0 0
```

2. This parameter file (lowpass_butter_10hz_coefs_epoch.par [12]) filters all channels of an EEG file containing 10 channels, with a Butterworth filter of same characteristics designed in Matlab (with butter function):

```
nb_eventcode 2
list_eventcode 2 5

prestim_nbsample 400 800
poststim_nb_sample 1000 1200

filter_channel 1 1 1 1 1 1 1 1 0 0

#low_pass Butterworth filter designed with Matlab butter function
filter_b_coeff_nb 4

filter_b_coeffs
2.71835675758059e-05    8.15507027274176e-05    8.15507027274176e-05
2.71835675758059e-05

filter_a_coeff_nb 4

filter_a_coeffs
1       -2.87730072411486       2.76201379931893 -0.884495606663461
```

3. This parameter file (bp_2-30Hz_epoch.par [13]) computes a band-pass (2-30Hz) Butterworth filter, and filters all channels of an EEG file containing 10 channels:

```
nb_eventcode 2
list_eventcode 2 5

prestim_nbsample 400 800
poststim_nb_sample 1000 1200
```

```
#band_pass
filter_type 2

filter_order 3

filter_cutoff_freq1 2
filter_cutoff_freq2 30

filter_channel 1 1 1 1 1 1 1 1 0 0
```

- ***Comments***

- ***Current version***

  1.02 24-01-2017

- ***History***
  - 1.00 29-11-2010 (PEA) : 1st version.
  - 1.01 05-09-2011 (PEA) : changes initial conditions computation (fits to Matlab).
  - 1.02 24-01-2017 (PEA) : fix an error in data offset computing on large files (>2GB).

- ***Files***

  $ELANPATH/bin/eegepochfiltfilt

- ***See also***

  epfiltfilt [14], eegfiltfilt [15]

Attachment	Size
bp_2-30Hz_epoch.par [16]	215 bytes
lowpass_butter_10hz_coefs_epoch.par [17]	426 bytes
lowpass_butter_10hz_epoch.par [18]	426 bytes

eegextract

- ***Description***

  Extract data from a **.eeg** file and save data to another **.eeg** file.

- ***Usage***

  eegextract myeegfilein myeegfileout 1st_sample [last_sample]
  with :

  - myeegfilein : input EEG file to process (without extension).
  - myeegfileout : output EEG file to process (without extension).
  - 1st_sample : index of first sample (starting from 0).
  - options :

    last_sample : index of last sample (starting from 0). If omitted, the last sample of myeegfilein is used (i.e. the end of the file).

- ***Fields of parameter file and examples***

- ***Example***

- ***Comments***

- ***Current version***

  1.01 29-09-2010

- ***History***
  - 1.00 27-07-2009 (PEA) : 1st version.
  - 1.01 29-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.

- ***Files***

  $ELANPATH/bin/eegextract

- ***See also***

eegfiltfilt

- ***Description***

  Computes zero-phase filtering (forward and reverse) on analog continuous channels of an EEG file, and creates a new EEG file. As input parameters, it can use B and A coefficients computed with Matlab (or SciPy, Octave, Scilab), or it can compute Butterworth filter coefficients from cutoff frequencies. It can compute band-pass, band-stop, low-pass or high-pass filtering.
  When computing Butterworth coefficients, a test for filter stability is processed. If the filter is unstable, no filtering is computed and a message for the new filter order to apply is displayed.

- ***Usage***

  eegfiltfilt myeegfilein.eeg myparameterfile.par myeegfileout.eeg [+b32]

  with :

  - myeegfilein.eeg : input EEG file to filter (with extension).
  - myparameterfile.par : filtering parameter file (with extension).
  - myeegfileout.eeg : output (filtered) EEG file (with extension).
  - options:
    +b32 : forces output EEG file to be in 32 bits format. If omitted, output data is in the same format as the input file.

- ***Fields of parameter file and example***

filter_channel 1 1 0 1 0 0 0	List of the channels to filter: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be filtered but all channels are stored in the output files.
filter_type 0	Type of Butterworth filter. Valid values are : 0 for low-pass filter 1 for high-pass filter 2 for band-pass filter 3 for band-stop filter In this example, a low-pass filter is processed.
filter_order 3	Butterworth filter order. In this example, the filter order is 3.

filter_cutoff_freq1 30	Cutoff frequency 1. In case of filter type : 0 (low-pass) filter : attenuates higher frequencies. 1 (high-pass) filter : attenuates lower frequencies. 2 (band-pass) filter : lower frequency of the range of frequencies to pass. 3 (band-stop) filter : lower frequency of the range of frequencies to attenuate.
filter_cutoff_freq2 100	Cutoff frequency 2. Needed only for band-pass and band-stop filter (filter_type 2 or 3). In case of filter type : 2 (band-pass) filter : higher frequency of the range of frequencies to pass. 3 (band-stop) filter : higher frequency of the range of frequencies to attenuate.
filter_b_coeff_nb 4	Number of B coefficients (in the Matlab way). These coefficients can be of a Butterworth filter, or any other filter.
filter_b_coeffs 2.71835675758059e-05 8.15507027274176e-05 8.15507027274176e-05 2.71835675758059e-05	List of B coefficients. **As it is text file, be careful of include a maximum of significant digits.** In this example, coefficients are corresponding to a low-pass Butterworth filter of order 3.
filter_a_coeff_nb 4	Number of A coefficients (in the Matlab way). These coefficients can be of a Butterworth filter, or any other filter.
filter_a_coeffs 1 -2.87730072411486 2.76201379931893 - 0.884495606663461	List of A coefficients. **As it is text file, be careful of include a maximum of significant digits.** In this example, coefficients are corresponding to a low-pass Butterworth filter of order 3.

Use parameters when working with a Butterworth filter and setting filter order, type and cutoff frequency(ies).
Use parameters when working with a coefficients computed by another way.

- ***Examples***
  1. This parameter file (lowpass_butter_10hz.par [19]) computes Butterworth coefficients from the filter definition, and filters all channels of an EEG file containing 10 channels (8 EEG channels and 2 ELAN technical channels) :

```
#low_pass Butterworth filter
filter_type 0

filter_order 3

filter_cutoff_freq1 10

filter_channel 1 1 1 1 1 1 1 0 0
```

  2. This parameter file (lowpass_butter_10hz_coefs.par [20]) filters all channels of an EEG file containing 10 channels (8 EEG channels and 2 ELAN technical channels), with a Butterworth filter of same characteristics designed in Matlab (with butter function):

```
filter_channel 1 1 1 1 1 1 1 0 0

#low_pass Butterworth filter designed with Matlab butter function
filter_b_coeff_nb 4

filter_b_coeffs
2.71835675758059e-05    8.15507027274176e-05    8.15507027274176e-05
2.71835675758059e-05

filter_a_coeff_nb 4

filter_a_coeffs
1       -2.87730072411486       2.76201379931893 -0.884495606663461
```

  3. This parameter file (bp_2-30Hz.par [21]) computes a band-pass (2-30Hz) Butterworth filter, and filters all channels of an EEG file containing 10 channels (8 EEG channels and 2 ELAN technical channels):

```
#band_pass
filter_type 2

filter_order 3

filter_cutoff_freq1 2
filter_cutoff_freq2 30
```

```
filter_channel 1 1 1 1 1 1 1 1 0 0
```

- ***Comments***
  - WARNING : this tool is not intended to use with concatenated data. For this kind of data use eegepochfiltfilt [22].

- ***Current version***

  1.05 05-09-2011

- ***History***
  - 1.00 09-11-2010 (PEA) : 1st version.
  - 1.01 17-11-2010 (PEA) : adds test for filter stability when computing Butterworth coefficients.
  - 1.02 08-12-2010 (PEA) : minor modification.
  - 1.03 03-03-2011 (PEA) : adds +b32 option to force 32 bits output EEG data.
  - 1.04 01-09-2011 (PEA) : fixes error when saving 16 bits data in 32 bits format (+b32 option).
  - 1.05 05-09-2011 (PEA) : changes initial conditions computation (fits to Matlab).

- ***Files***

  $ELANPATH/bin/eegfiltfilt

- ***See also***

  epfiltfilt [14], eegepochfiltfilt [22], eegresample [23]

Attachment	Size
lowpass_butter_10hz.par [24]	119 bytes
lowpass_butter_10hz_coefs.par [25]	335 bytes
bp_2-30Hz.par [26]	123 bytes

eegfiltica

- ***Description***

  Applies a decomposition matrix obtained after ICA (performed in Matlab for example) to an EEG file and creates a new (filtered) EEG file. This is used to remove some ICA components from an EEG file (e.g., eye blinks).

- ***Usage***

  eegfiltica myeegfilein.eeg matrixfile.xml myeegfileout [--keep] component_1 component_2 … component_n

  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - matrixfile.xml : XML transform matrix file (with extension), including information on matrix size (see format below).
  - myeegfileout : EEG file created after the matrix has been applied (without extension). Number of channels = number of components (matrix size) + unused EEG channels (those not involved in the matrix estimation in matlab).
  - option :
    --keep : flag used to keep the following components and to suppress all others. If omitted, the listed components are removed and the others are kept.
  - component_1 : index of the 1st component to suppress (or keep if –keep is used).
  - …
  - component_n : index of the nth component to suppress (or keep if –keep is used).

- ***Fields of parameter file and example***

- ***Examples***

- ***Comments***

- ***Current version***

  1.04 23-09-2010

- ***History***
  - 1.00 19-02-2008 (PEA) : 1st version.
  - 1.01 07-04-2008 (PEA) : add --keep option to keep components list instead of remove it.
  - 1.02 24-07-2008 (PEA) : correction for --keep option (use number of components).
  - 1.03 31-08-2009 (PEA) : correction for large files (> 2 GB).
  - 1.04 23-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.

- ***Files***

  $ELANPATH/bin/eegfiltica

- ***See also***

  eegcomponent [27], eegproject [6], matrix2p [8]

eegmval

- ***Description***

  Computes mean, minimum and maximum values of an **.eeg** file in time windows and outputs in a text file.

- ***Usage***

  eegmval *myeegfile**.eeg** myposfilein**.pos** myparfile**.par** [window_file.t.wnd]*

  with :

  - *myeegfile**.eeg***: input ***.eeg*** file to process (with extension).
  - *myposfilein**.pos***: input event file (with extension).
  - *myparfile**.par***: text parameter file (with extension) for measurements.
  - options:
    window_file.t.wnd : text file containing time window description. If omitted, time windows are created from the parameter file myparfile.par .

- ***Fields of parameter file and examples***

fileprefix myfilename_out	Prefix of the output text files.
nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself.
	List of the channels to take measurements: 1/0 for selected/unselected channels; the total

ep_channel_flag 1 1 0 1 0 0 0	number of flags is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be averaged and stored in the output .p files.
ep_channel_ref 0 5 0 6 0 0 0	List of the new reference for each channel before taking measurements (bipolar montage for instance): 0: no change of the reference, ≠0: electrode number (rank) to which the current channel should be re-referenced. The total number of values is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. This field is optional. If omitted, the channels are not modified. In this example, N=5, and channel 1 is unchanged, channel 2 is referenced to channel 5, and channel 4 re-referenced to channel 6.
eegmval_time_hw 50	Half time window (in msec) of the measurements. **This parameter is required only if no window file is provided.**
eegmval_time_step 50	Time step (in msec) for the sliding window of the measurements. **This parameter is required only if no window file is provided.**

Window file format (window_file.t.wnd) :

wnd_nb 2	Number of time windows.
wnd_list -250 -50 500 750	Beginning and end (in msec) latencies list for each window. The time 0 is at the event code. In this example, the 1st window starts at -250 ms and ends at -50 ms, and the 2nd window starts at 500 ms and ends at 750 ms.
wnd_label_list wnd_1 wnd_2	Window label list. Each label is on one line. In this example, the 1st window has label 'wnd_1', and the 2nd window has label 'wnd_2' .

- ***Examples***

- ***Comments***
  - See eegchref [2] to create a re-referenced .eeg data file (several referencing options available).

- ***Current version***

  1.04 29-09-2010

- ***History***
  - 1.02 14-01-2004 (PEA) : 1st documented version.
  - 1.03 29-11-2006 (PEA) : correction files larger than 2GB. Modification for use of rejection flag in .pos file.
  - 1.04 29-09-2010 (PEA) : update to use cmake and free release of Elan.

- ***Files***

  $ELANPATH/bin/eegmval

- ***See also***

  eegchref [2], eegstat [3], eegavg [28]

eegoffset

- ***Description***

Offset correction by substraction of the mean (by channel) of an **.eeg** file.

- ***Usage***

  eegoffset *myeegfilein**.eeg** myparfile**.par** myeegfileout**.eeg***
  or
  eegoffset** myeegfilein**.eeg** -all myeegfileout**.eeg

  with :

  - *myeegfilein**.eeg***: input ***.eeg*** file to process (with extension).
  - *myparfile**.par***: text parameter file (with extension) for measurements.
  - -all : flag to correct all channels.
  - *myeegfileout**.eeg***: output ***.eeg*** file (with extension).

  If -all is specified, no parameter file is required, and all channels are corrected.

- ***Fields of parameter file and examples***

offset_correction_channel_flag 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0	List of channel to remove offset : 1/0 for selected/unselected channels. The total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0.

- ***Examples***

- ***Comments***

- ***Current version***

  1.02 29-09-2010

- ***History***
  - 1.00 29-07-2008 (OB) : 1st version.
  - 1.01 06-02-2009 (PEA) : minor modifications.
  - 1.02 29-09-2010 (PEA) : update to use cmake and free release of Elan.

- ***Files***

  $ELANPATH/bin/eegoffset

- ***See also***

eegpos

- ***Description***

  Extracts the position and codes of all event stored in the eeg file, and creates a text event position file.

- ***Usage***

  eegpos myeegfile.eeg myposfile.pos

  with :

- myeegfile.eeg : input EEG file to process (with extension).
- myposfile.pos : event output file (with extension).

- ***Fields of parameter file and example***

- ***Examples***

- ***Comments***

- ***Current version***

  1.10 25-04-2016

- ***History***
  - 1.06 22-02-2006 (PEA) : bug fix for 32 bits events (event mask 0xffffffff instead of 0xffff).
  - 1.07 27-04-2006 (PEA) : minor modification.
  - 1.08 13-08-2007 (PEA) : minor modification.
  - 1.09 23-09-2010 (PEA) : update to use cmake and free release of Elan.
  - 1.10 25-04-2016 (PEA) : new HDF5 ELAN file format.

- ***Files***

  $ELANPATH/bin/eegpos

- ***See also***

eegproject

- ***Description***

  Applies a matrix to an eeg file and creates a new (projected) eeg file. This is a linear transformation of the original eeg signal which leads to a new eeg signal. This is used to obtain the time course of the components obtained after an ICA is performed (by a matlab program).

- ***Usage***

  eegproject myeegfilein.eeg matrixfile.xml myeegfileout
  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - matrixfile.xml : XML transform matrix file (with extension), including information on matrix size (see format below).
  - myeegfileout : EEG file created after the matrix has been applied (without extension). Number of channels = number of components (matrix size) + unused EEG channels (those not involved in the matrix transform).

- ***Fields of parameter file and examples***

- ***Example***

- ***Comments***

- ***Current version***

  1.03 03-01-2011

- ***History***
  - 1.00 15-02-2008 (PEA) : 1st version.
  - 1.01 26-02-2008 (PEA) : changes component names and numbers.
  - 1.02 23-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.
  - 1.03 03-01-2011 (PEA) : minor modification (change in description).

    file header.

- ***Files***

  $ELANPATH/bin/eegproject

- ***See also***

  eegfiltica [7], matrix2p [8]

eegrandadd

- ***Description***

  Performs a statistical randomization test without replacement between k conditions from single-trials stored in an EEG file. This program is very similar to eegranddiff [29]. It compares the sum of the first 2 conditions against the third condition.

- ***Usage***

  eegrandadd myeegfile.eeg myposfile.pos myparfile.par nb_drawings

  with :

  - myeegfile.eeg : input EEG file to process (with extension).
  - myposfile.pos : event position file.
  - myparfile.par : parameter file, similar to that used for eegavg, but with some additional parameters.
  - nb_drawings : number of drawings that are made to estimate the distribution of differences under the null hypothesis (example: 10000 drawings enables a resolution of p=0.0001).

- ***Fields of parameter file and example***

fileprefix myfilename	Prefix of the output .p files.
nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 400	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 800 800	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to average: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be averaged and stored in the output .p files.
ep_channel_ref 0 5 0 6 0 0 0	List of the new reference for each channel before averaging (bipolar montage for instance): 0: no change of the reference, ≠0: electrode number (rank) to which the current channel should be re-

	referenced. The total number of values is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. This field is optional. If omitted, the channels are not modified. In this example, N=5, and channel 1 is unchanged, channel 2 is referenced to channel 5, and channel 4 re-referenced to channel 6.
eegranddiff_baseline_eventcode 2 2	List of the event codes used to compute the baseline. 0: no baseline correction (default), ≠0: event code on which baseline will be computed. In this example, the baseline for event code 2 will be computed from event code 2, and the baseline of event code 5 from event code 2. Warning: the event-code should be chosen among those listed in list_eventcodes. This field is optional.
eegranddiff_baseline_msec_start -300 -300	List of the baseline start latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
eegranddiff_baseline_msec_stop -100 -100	List of the baseline stop latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
eegranddiff_time_hw 200 200 200	List of the time half-windows (in ms) by event code, used to defined the moving time-window on which statistical analysis will be performed. If omitted, no smoothing is applied and computation is performed on each time-sample.
eegranddiff_time_step 50 50 50	List of the time steps (in ms) by event code, used to defined the moving step of the moving time-window on which statistical analysis will be performed. If omitted, the time step is one sample.
eegranddiff_measure 1	Statistic on which the permutation test will be made : 1: Sum of squared sums of values by conditions (two-sided test) 2: Sum of values in the first condition (one-sided test), valid only if k=2 conditions. If omitted, default value is 1.
eegranddiff_compute 1	Choice for multiple test correction : 0: no multiple test correction 1: multiple test correction based on the maximum number of consecutive samples 2: multiple test correction based on the minimum significance across samples 3: same as 2 across samples and channels If omitted, default value is 0 (no correction).
eegranddiff_proba_measure 0.05	If eegrandiff_compute = 1, statistical probability threshold for the determination of significant samples.
eegranddiff_proba_compute 0.05	If eegrandiff_compute ≥ 1, statistical probability threshold for the computation of consecutive significant sample.
eegranddiff_compute_msec_start 50	If eegrandiff_compute ≥ 1, start latency (in msec) of the analysis window for the correction across samples.
eegranddiff_compute_msec_stop 50	If eegrandiff_compute ≥ 1, stop latency (in msec) of the analysis window for the correction across samples.
erpa_positivity_up 1	Value stored in all output .p file to set the default upward/downward positivity orientation for curve display (by erpa). -1 (positivity down) or 1 (positivity up) This field is optional. If omitted, erpa will display curve with positivity down.
erpa_view 4	Value stored in all output .p file. It is used by erpa to set the default mapping view. 1: right view 2: left view 3: top view 4: back view 5: large top view 6: back-top view This field is optional. If omitted, it is set to 0, and erpa default will be top view.

- ***Examples***

- ***Comments***

  - See eegchref [2] to create a re-referenced .eeg data file (several referencing options available).

- ***Current version***

1.11 15-04-2011

- ***History***

  - 1.01 30-06-2006 (JB) : 1st version, derived from eegranddiff v 2.01 .
  - 1.02 02-07-2006 (JB) : fixes smoothing (compatible with eegstat).
  - 1.03 03-07-2006 (JB) : fixes baseline correction.
  - 1.04 05-07-2006 (JB) : adds compute_msec_start and compute_msec_stop to have independant analysis window and baseline window.
  - 1.05 06-07-2006 (JB) : optimizes computing with copy of data out of permutations loop.
  - 1.06 11-12-2006 (JB) : fixes bilateral statistic (see eegranddiff [29]).
  - 1.07 24-01-2007 (JB) : fixes for event numbers. Adds unilateral test.
  - 1.08 26-01-2007 (JB) : modifies decision for loading data to memory (if data size<= whole memory, not just available).
  - 1.09 26-01-2007 (JB) : removes count of n_max out of window.
  - 1.10 28-09-2010 (PEA) : update to use cmake and free release of Elan. Changes for parameter names : compute_msec_start and compute_msec_stop become eegranddiff_compute_msec_start and eegranddiff_compute_msec_stop. Changes for baseline parameters : baseline_eventcode, baseline_msec_start and baseline_msec_stop become eegranddiff_baseline_eventcode, eegranddiff_baseline_msec_start and eegranddiff_baseline_msec_stop.
  - 1.11 15-04-2011 (PEA) : changes Shell sort to Heap sort in FDR (faster algorithm).

- ***Files***

$ELANPATH/bin/eegrandadd

- ***See also***

eegranddiff [30], eegstat [3], eegavg [28], eegchref [2]

eegranddiff

- ***Description***

  Performs a statistical randomization test without replacement between k conditions from single-trials stored in an EEG file.
  This programme creates a .rand.p.p file representing the probability values p that the events in k conditions are drawn from the same population of events* (the trials from the k conditions are shuffled and the distribution of an equivalent of the F statistic under the null hypothesis is estimated by randomly assigning trials to conditions), independently computed for each channel over each moving window. They can be viewed by erpa as curve plots.
  * if the null hypothesis is rejected, assuming equivalence of the variances, it can further be concluded that the k types of events are drawn from k populations of events with different means.
  Optionally, this program can take into account the number of samples of the window of analysis and correct for multiple tests in different ways:
  - The minimum number of consecutive significant samples (at a given probability threshold proba_measure) appearing in permutations with a given probability threshold proba_compute is computed independently for each channel (i.e. the maximum number of significant samples appearing with a given probability under the null hypothesis).
  - The maximum significance probability appearing in permutations with a given probability threshold proba_compute (i.e. the minimum significance probability appearing with a given probability under the null hypothesis) is computed across samples (i.e. independently for each channel) or accross channels and electrodes.
  The output files for these options are 2 statistical .p files (uncorrected and corrected) and a .txt file including the correction value for each channel.

- ***Usage***

  eegranddiff myeegfile.eeg myposfile.pos myparfile.par nb_drawings

  with :

  - myeegfile.eeg : input EEG file to process (with extension).
  - myposfile.pos : event position file.
  - myparfile.par : parameter file, similar to that used for eegavg, but with some additional parameters.
  - nb_drawings : number of drawings that are made to estimate the distribution of differences under the null hypothesis (example: 10000 drawings enables a resolution of p=0.0001).

- ***Fields of parameter file and example***

fileprefix myfilename	Prefix of the output .p files.
nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 400	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 800 800	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to average: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be averaged and stored in the output .p files.
ep_channel_ref 0 5 0 6 0 0 0	List of the new reference for each channel before averaging (bipolar montage for instance): 0: no change of the reference, ≠0: electrode number (rank) to which the current channel should be re-referenced. The total number of values is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. This field is optional. If omitted, the channels are not modified. In this example, N=5, and channel 1 is unchanged, channel 2 is referenced to channel 5, and channel 4 re-referenced to channel 6.
eegranddiff_baseline_eventcode 2 2	List of the event codes used to compute the baseline. 0: no baseline correction (default), ≠0: event code on which baseline will be computed. In this example, the baseline for event code 2 will be computed from event code 2, and the baseline of event code 5 from event code 2. Warning: the event-code should be chosen among those listed in list_eventcodes. This field is optional.
eegranddiff_baseline_msec_start -300 -300	List of the baseline start latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
eegranddiff_baseline_msec_stop -100 -100	List of the baseline stop latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
eegranddiff_time_hw 200 200 200	List of the time half-windows (in ms) by event code, used to defined the moving time-window on which statistical analysis will be performed. If omitted, no smoothing is applied and computation is performed on each time-sample.
eegranddiff_time_step 50 50 50	List of the time steps (in ms) by event code, used to defined the moving step of the moving time-window on which statistical analysis will be performed. If omitted, the time step is one sample.
eegranddiff_measure 1	Statistic on which the permutation test will be made :

	1: Sum of squared sums of values by conditions (two-sided test) 2: Sum of values in the first condition (one-sided test), valid only if k=2 conditions. If omitted, default value is 1.
eegranddiff_compute 1	Choice for multiple test correction : 0: no multiple test correction 1: multiple test correction based on the maximum number of consecutive samples 2: multiple test correction based on the minimum significance across samples 3: same as 2 across samples and channels If omitted, default value is 0 (no correction).
eegranddiff_proba_measure 0.05	If eegrandiff_compute = 1, statistical probability threshold for the determination of significant samples.
eegranddiff_proba_compute 0.05	If eegrandiff_compute ≥ 1, statistical probability threshold for the computation of consecutive significant sample.
eegranddiff_compute_msec_start 50	If eegrandiff_compute ≥ 1, start latency (in msec) of the analysis window for the correction across samples.
eegranddiff_compute_msec_stop 50	If eegrandiff_compute ≥ 1, stop latency (in msec) of the analysis window for the correction across samples.
erpa_positivity_up 1	Value stored in all output .p file to set the default upward/downward positivity orientation for curve display (by erpa). -1 (positivity down) or 1 (positivity up) This field is optional. If omitted, erpa will display curve with positivity down.
erpa_view 4	Value stored in all output .p file. It is used by erpa to set the default mapping view. 1: right view 2: left view 3: top view 4: back view 5: large top view 6: back-top view This field is optional. If omitted, it is set to 0, and erpa default will be top view.

In the example above, all trials with codes 2 and 5 will be processed on a time window ranging from 400 samples prior to 1000 samples after event code. The analysis will be performed on channels 1 (not re-referenced), 2 (re-referenced to channel 3),and 4 (re-referenced to channel 5). On each trial, the data are averaged over a moving window having 200 ms of half duration (total duration of 400 ms), moving by step of 50 ms, and with a baseline correction defined between –300 and –100 ms.

Output :

- .p file of the p values of the randomization test computed between event codes 2 and 5 :
  myfilename.2-5.rand.p.p
- For the multiple-test correction across samples : list of the maximum number of successive significant samples n_max (or the minimum probability p_min) under the null hypothesis for each channel when eegranddiff_compute is set to 1 or 2 :
  myfilename.2-5.n_max.txt
- For the multiple-test correction : only p values included in a succession of samples that is above the computed minimum number of significant samples (or smaller than the p_min value) are kept. The other samples are set to p=1 :
  myfilename.2-5.rand.mask.p.p

- ***Examples***

- ***Comments***
  - See eegchref [2] to create a re-referenced .eeg data file (several referencing options available).

- ***Current version***

2.13 15-04-2011

- ***History***
  - 1.03 18-08-2005 (PEA) : 1st documented version. Adds a smoothing temporal window (eegrand_time_hw and eegrand_time_step parameters).
  - 1.04 26-01-2006 (PEA) : minor modification.
  - 1.05 20-06-2006 (PEA-JB) : fixes statistic (unilateral or bilateral with eegranddiff_measure parameter). Changes in loop for channels.
  - 1.06 29-06-2006 (JB) : generalization for case 1 factor, k conditions (bilateral test).
  - 1.07 29-06-2006 (PEA) : reads file if channel loop to limit memory usage.
  - 2.01 30-06-2006 (JB) : adds multiple-test correction (eegranddiff_compute, eegranddiff_proba_compute and eegranddiff_proba_measure parameters).
  - 2.02 01-07-2006 (JB) : fixes smoothing (compatible with eegstat).
  - 2.03 04-07-2006 (JB) : fixes baseline correction.
  - 2.04 05-07-2006 (JB) : adds compute_msec_start and compute_msec_stop to have independant analysis window and baseline window.
  - 2.05 06-07-2006 (JB) : optimizes computing with copy of data out of permutations loop.
  - 2.06 11-12-2006 (PEA) : fixes memory size.
  - 2.07 12-12-2006 (PEA) : modifies decision for loading data to memory (if data size<= whole memory, not just available).
  - 2.08 18-12-2006 (AC - JB) : fixes bilateral statistic : sum of square sum by condition must be weighted by the size. The statitic was wrong when the sizes were very different. Adds comparison for k > 2 conditions.
  - 2.09 20-10-2009 (PEA) : changes for baseline parameters : baseline_eventcode, baseline_msec_start and baseline_msec_stop become eegranddiff_baseline_eventcode, eegranddiff_baseline_msec_start and eegranddiff_baseline_msec_stop.
  - 2.10 21-10-2009 (PEA) : changes for parameter names : compute_msec_start and compute_msec_stop become eegranddiff_compute_msec_start and eegranddiff_compute_msec_stop.
  - 2.11 26-10-2009 (PEA) : fixes reading of parameter eegranddiff_compute_msec_stop (only one value).
  - 2.12 27-09-2010 (PEA) : update to use cmake and free release of Elan.
  - 2.13 15-04-2011 (PEA) : changes Shell sort to Heap sort in FDR (faster algorithm).

- ***Files***

$ELANPATH/bin/eegranddiff

- ***See also***

eegrandadd [31], eegstat [3], eegavg [28], epranddiff [32], eegchref [33]

eegread

- ***Description***

  Reads data of one channel from an EEG file and displays to screen as text values.

- ***Usage***

  eegread myeegfilein.eeg channel_number sample_number

  with :

  - myeegfilein.eeg : input EEG file to read (with extension).
  - channel_number : indice of channel to read (starting from 1).
  - sample_number : number of samples to read and print (starting from 1).

- ***Fields of parameter file and example***

- ***Examples***

- ***Comments***

- ***Current version***

  1.01 27-09-2010

- ***History***
  - 1.00 31-10-2008 (PEA) : 1st version.
  - 1.01 27-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.

- ***Files***

  $ELANPATH/bin/eegread

- ***See also***

eegregress

- ***Description***

  Removes artefact (usually eye blink or movement) on continuous EEG (MEG) data by computing linear regression. This programs first detects for artefact on the regressor channel(s). If an artefact is detected in the epoch, the linear regression is applied. The detection and correction are done on epochs defined by the parameter and event files. The output EEG file contains all concatenated epochs (corrected or not if no artefact detection). The epoch from rejected events are excluded from output file.

- ***Usage***

  eegregress *myeegfilein**.eeg** myeventfile**.pos** myparameterfile**.par** myeegfileout**.eeg***

  with :

  - *myeegfilein**.eeg***: input ***.eeg*** file to correct (with extension).
  - *myeventfile**.pos***: input event file (with extension) used for data epoching.
  - *myparameterfile**.par***: text file (with extension) containing computing parameters.
  - *myeegfileout**.eeg***: output EEG file (with extension) containing concatenated corrected epoched data.

- ***Fields of parameter file and examples***

regress_channel 0 1 1 1 1 1 1 1 0 0	List of the channels to correct: 1/0 for select/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0. In this example, N=8, and channels number 2 to 8 will be corrected and stored in the output EEG file.
regressor_channel 1 0 0 0 0 0 0 0 0 0	List of the channels to use as regressor: 1/0 for select/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0. In this example, N=8, and channel number 1 (EOG) is used as regressor. It is not modified in the output EEG file.
nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself.
baseline_msec_start -200 -	List of the baseline start latencies (in ms); one value for each event code.

200	If omitted, no baseline value is computed.
baseline_msec_stop -50 -50	List of the baseline stop latencies (in ms); one value for each event code. If omitted, no baseline value is computed.
center_regressor_chan 1	Flag to center the regressor channel(s) (substract the mean value, computed on the whole epoch). Possible values are : 0 : do not center regressor channel(s). 1 : center regressor channel(s).
range_threshold 125	Threshold value (in physical unit) used for artefact detection. If the difference between the maximum and minimum values on the whole epoch of one regressor channel is larger than this value, an artefact is detected.
smoothing_halfwnd_nbsample 20	Number of samples of the smoothing half-window. The smoothing is applied before the derivative for the artefact detection.
derivative_len_msec 50	Length in ms of the derivative.
derivative_threshold 1	Threshold value (in physical unit/s e.g. µV/s) used for artefact detection. This threshold is applied to the derivative. If the absolute value of the derivative of one regressor channel is larger than this value, an artefact is detected.

- ***Examples***

- ***Comments***

- ***Current version***

  1.01 30-06-2011

- ***History***
  - 1.00 02-03-2011 (PEA) : 1st version.
  - 1.01 30-06-2011 (PEA) : fixes label of post-stimulus number of samples (poststim_nbsamp to poststim_nbsample).

- ***Files***

  $ELANPATH/bin/eegregress

- ***See also***

eegresample

- ***Description***

  Tool for resampling continuous data.

- ***Usage***

  eegresample myeegfile_in.eeg myeegfile_out.eeg new_sf

  with :

  - myeegfile_in.eeg : input file to change reference (with extension).
  - myeegfile_out.eeg : output file (with extension).
  - new_sf : new sampling frequency (Hz).

- ***Fields of parameter file and example***

- ***Example***

- ***Comments***
  - When decimating data, to ensure that the sampling theorem is satisfied, a low-pass filter is used as an anti-aliasing filter to reduce the bandwidth of the signal before the signal is downsampled. For this, you must use eegfiltfilt [15] to filter EEG data before downsampling. The filter is a low-pass with a cutoff frequency less than new_sf/2 .
  - The filtering is not computed by eegresample. You must first call eegfiltfilt [15] before downsampling data.

- ***Current version***

  3.03 04-05-2011

- ***History***
  - 3.02 29-09-2010 (PEA) : 1st documented version.
  - 3.03 04-05-2011 (PEA) : adds more precision (10 decimals) in for saving sampling period in output header file.

- ***Files***

  $ELANPATH/bin/eegresample

- ***See also***

  eegfiltfilt [15]

eegresyncchan

- ***Description***

  Removes time offset of a list of channels of an EEG file.

- ***Usage***

  eegresyncchan myeegfilein.eeg myparameterfile.par myeegfileout.eeg

  with :

  - myeegfilein.eeg : input EEG file to modify (with extension).
  - myparameterfile.par : filtering parameter file (with extension).
  - myeegfileout.eeg : output EEG file (with extension).

- ***Fields of parameter file and example***

  channel_offset_samples 0 0 0 0 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0
  0 0 0 0 0 0 0 0 0 0
  0 0 0 0
  2 2 2 2 2 2 2 2 2 2
  2 2 2 2 2 2 2 2 2 2
  2 2 2 2 2 2 2 2 2 2
  2 2 2 2 2 2 2 2 2 2
  2 2 2 2 2 2 2 2 2 2
  2 2 2 2 2 2 2 2 2 2
  2 2 2 2 2

0 0
List of the actual time offsets in samples. One value for each channels. Offsets are substracted. The total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 channels correspond to event channels.
In this example, N=129, and channels 1 to 64 have no time offset, as channel 64 to 129 have a time offset of 2 samples. The last 2 channels (events and counter) are not modified.

- ***Examples***
  1. The command line below illustrates the above example :

     eegresyncchan eeg_1.eeg resync.par eeg_1.resync.eeg
     with
     eeg_1.eeg : the EEG file to correct.
     resync.par available to download here [34].
     eeg_1.resync.eeg : the modified EEG file.

- ***Comments***

- ***Current version***

  1.01 25-03-2011

- ***History***
  - 1.00 03-03-2011 (PEA) : 1st version.
  - 1.01 25-03-2011 (PEA) : fixes large file reading for 32 bits version.

- ***Files***

  $ELANPATH/bin/eegresyncchan

- ***See also***

Attachment	Size
resync.par [35]	630 bytes

eegrms

- ***Description***

  Computes sum of RMS values across trials from an EEG file.

- ***Usage***

  eegrms myeegfilein.eeg myeventfile.pos myparameterfile.par [+v]

  with :

  - myeegfilein.eeg : input EEG file (with extension).
  - myeventfile.pos : event file to process (with extension).
  - myparameterfile.par : parameter file (with extension) containing parameters.
  - options :
    - +v : verbose mode on. If omitted, verbose mode is off.

- ***Fields of parameter file and example***

fileprefix myfilename.rms	Prefix of the output files.

nb_eventcode 3	Number of event codes to process.
list_eventcode 10 12 19	List of the event codes to process.
prestim_nbsample 500 500 500	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1000 1000	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to compute: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be computed and stored in the output files.
baseline_eventcode 2 2	List of the event codes used to compute the baseline. 0: no baseline correction (default), ≠0: event code on which baseline will be computed. In this example, the baseline for event code 2 will be computed from event code 2, and the baseline of event code 5 from event code 2. This field is optional.
baseline_msec_start -200 -200	List of the baseline start latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
baseline_msec_stop -50 -50	List of the baseline stop latencies (in ms); one value for each event code; required only if baseline_eventcode≠0. If omitted and baseline_eventcode≠0, the baseline value is computed on the whole epoch.
erpa_view 4	Value stored in all output .p file. It is used by erpa to set the default mapping view (see eegavg [36]). This field is optional. If omitted, it is set to 0, and erpa default will be top view.
erpa_positivity_up	Value stored in all output .p file to set the default upward/downward positivity orientation for curve display (by erpa). -1 (positivity down) or 1 (positivity up) This field is optional. If omitted, erpa will display curve with positivity down.

- ***Example***

- ***Comments***

- ***Current version***

  1.13 27-09-2010

- ***History***
  - 1.12 12-01-2004 (PEA/CTB) : 1st documented version.
  - 1.13 27-09-2010 (PEA) : update to use cmake and free release of Elan. Remove all static allocations. Change messages language to english.

- ***Files***

  $ELANPATH/bin/eegrms

- ***See also***

  eegavg

eegspectrum

- ***Description***

  Computes FFT on an EEG file.

- ***Usage***

  eegspectrum myeegfilein myparameterfile.par myeventfile.pos [+std]

  with :

  - myeegfilein : input EEG file (without extension).
  - myparameterfile.par : parameter file (with extension) containing parameters for FFT computation.
  - myeventfile.pos : event file to process (with extension).
  - options :
    - +std : computes an ep file with standard deviation of FFT, stored in filename.std.p .

- ***Fields of parameter file and example***

fileprefix myfilename.fft	Prefix of the output files.
nb_eventcode 3	Number of event codes to process.
list_eventcode 10 12 19	List of the event codes to process.
prestim_nbsample 2000 2000 2000	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 3000 3000 3000	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself
fft_channel_flag 1 1 0 1 0 0 0	List of the channels to compute: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be computed and stored in the output files.
fft_channel_ref 0 5 0 6 0 0 0	List of the new reference for each channel before averaging (bipolar montage for instance): 0: no change of the reference, ≠0: electrode number (rank) to which the current channel should be re-referenced. The total number of values is N+2, N being the number of recorded channels in myeegfilein file; the last 2 flags should be set to 0. This field is optional. If omitted, the channels are not modified. In this example, N=5, and channel 1 is unchanged, channel 2 is referenced to channel 5, and channel 4 re-referenced to channel 6.
fft_center_signal 1 1 1	Flag to center signal before computing FFT (0=center signal, 1=no centering), by substracting mean value of sliding window . This field is optional. If omitted, no centering is done.
fft_nb_sample_blackman 100 100 100	Number of samples of the Blackman window applied to signal before computing FFT. This field is optional. If omitted, the default value is 100 samples.
fft_nb_samples 1000 1000 1000	Number of samples of the sliding window of FFT. This field is optional. If omitted, the window size corresponds to the whole event window.
fft_step_nb_samples 100 100 100	Step of the sliding window of FFT. This field is optional. If omitted, the step is half of the sliding window.
erpa_view 4	Value stored in all output .p file. It is used by erpa to set the default mapping view (see eegavg [36]). This field is optional. If omitted, it is set to 0, and erpa default will be top view.

- ***Example***

- ***Comments***

- ***Current version***

  1.07 12-04-2013

- ***History***
  - 1.00 19-03-2008 (PEA) : 1st version.
  - 1.01 21-03-2008 (PEA) : adds parameter fft_step_nb_samples for adjusting step of sliding window.
  - 1.02 25-03-2008 (PEA) : corrections for last sliding window and reference of channels.
  - 1.03 14-04-2008 (PEA) : correction for mean (number of FFT out of square root).
  - 1.05 26-11-2008 (PEA) : correction for signal center (sliding window instead of event window).
  - 1.06 27-09-2010 (PEA) : update to use cmake and free release of Elan.
  - 1.07 12-04-2013 (PEA) : minor modification for FFT initialization.

- ***Files***

  $ELANPATH/bin/eegspectrum

- ***See also***

  eegavg

eegspline

- ***Description***

  Computes spline interpolation from an ***.eeg*** file and creates a new ***.eeg*** file on any user-defined sensor montage. Resulting ***.eeg*** file consists in either interpolated potential values or scalp current density values (surface Laplacian), computed on a spherical surface.

- ***Usage***

  eegspline *myeegfile**.eeg** myparfile**.par** myeegfileout**.eeg** [myelecpositionin.rtp]*

  with :

  - *myeegfile**.eeg***: input ***.eeg*** file to process (with extension).
  - *myparfile**.par***: text parameter file (with extension).
  - *myeegfileout**.eeg***: output file (with extension) after processing by spline computation.
  - options:
    myelecpositionin.rtp : text file with the list of R, theta, phi coordinates of the recorded channels (channels in myeegfilein.eeg). Only theta and phi values will be considered for spline computation (normalized sphere with radius=1). If omitted, electrode coordinates (theta, phi) are read from elec.dat file.

- ***Fields of parameter file and examples***

spline_channel_flag 1 1 0 1 0 0 0 0	list of the channels to include in the spline computation: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be included for computation.
spline_channel_out_nb 6	Number of output channels (interpolated or not).
spline_channel_out_list 13 5 8 10 35 -1	List of output electrode numbers (according to elec.dat) on which spline interpolated values (potential or SCD) are computed. The electrodes with number -1 are not interpolated (e.g. EOG or EMG channels because they have no coordinates on head). To keep the original data in output file, use the -1 and the original_channel_out_nb and original_channel_out_list fields (see below).

original_channel_out_nb 1	Number of output channels with original data (no interpolation).
original_channel_out_list 5	List of electrode indices in the original input EEG file (first is number 1) of the output channel with original data. This electrodes are added at the -1 positions of spline_channel_out_list label with the same order. In this example, the 5 first channels (13, 5, 8, 10 and 35 are corresponding to F7, Pz, T3, Cz and POz in elec.dat) are interpolated from channels 1, 2 and 4. The 6th channel is the copy of the 5th channel of the original file.
spline_scd 1	Flag for computing SCD or potentials: 0 : potential interpolation 1 : SCD interpolation If omitted, spline_scd = 0.
spline_order 3	Spline interpolation order (usually 3 or 4). Suggestion: if regularization is used for potential or SCD, take spline order = 3, otherwise, take spline order = 4.
spline_regul 1	Type of regularization for spline computation: 0 : no regularization 1 : optimized (Tikhonov) 2 : user-defined lambda (Tikhonov) 3 : user-defined lambda (Wahba)
spline_regul_lambda 0.001	Regularization coefficient for spline interpolation. Required only when spline_regul = 2 or 3.
spline_signal_offset 1	Flag for offset correction of all selected input channels (for each channel, the mean value over the entire recording period is substracted from the raw signal): 0 : no offset correction 1 : offset correction If omitted, spline_signal_offset = 0.

Format of myelecposition.rtp file :
Text file containing the label 'r_theta_phi' followed by the R, Theta, Phi spherical coordinates of the sensors recorded in the input .eeg file myeegfilein.eeg (one line per sensor following the same order as in myeegfilein.eeg). The R value is not read by the programme, sensors are considered to be on a normalized sphere.

- ***Example***
- ***Comments***
- ***Current version***

  1.17 30-07-2015

- ***History***
  - 1.04 25-05-2005 (OB) : changes label of SCD computing flag (becomes spline_scd, was spline_pot_scd).
  - 1.06 04-04-2008 (PEA) : changes label of coordinates file (becomes r_theta_phi field).
  - 1.08 27-11-2009 (PEA) : changes in progress display.
  - 1.09 27-11-2009 (PEA) : reads data by blocks instead of samples.
  - 1.10 30-11-2009 (PEA) : changes for temporary file name (allows many instances of eegspline in one directory).
  - 1.11 23-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.
  - 1.12 21-02-2011 (PEA) : adds original_channel_out_nb and original_channel_out_list parameters to keep original channel which have no coordinates (e.g. EOG, EMG, ...).
  - 1.13 17-11-2011 (PEA) : minor modification.
  - 1.14 07-02-2012 (PEA) : removes maximum number of channel test. Removes exit when original_channel_out_nb field is not present.
  - 1.15 26-09-2014 (PEA) : change temporary file creation to output directory (was system temporary directory).
  - 1.16 30-09-2014 (PEA) : fix reading temporary file error (due to previous change).
  - 1.17 30-07-2015 (PEA) : fix rCSL constant initialization.

- ***Files***

  $ELANPATH/bin/eegspline

- ***See also***

  epscd

eegstat

- ***Description***

  Performs a statistical analysis from single-trials stored in an ***.eeg*** file. Two statistical tests are proposed on the signal values averaged over moving time-windows.
  - a Wilcoxon test (non-parametric paired comparison) for the detection of an emerging component with respect to a baseline period. The test is performed indepedantly for each event code and each channel.
  - a Kruskal-Wallis test (one-way analysis of variance by ranks) for the statistical comparison of the responses across different event codes.
  This program creates ***.p*** files representing the statistical values (Z for Wilcoxon, H for Kruskal-Wallis and their related probability values p) computed for each channel over each moving window. They can be viewed by erpa as curve plots.

- ***Usage***

  eegstat *myeegfile****.eeg*** *myposfile****.pos*** *myparfile****.par***

  with :

  - *myeegfile****.eeg***: input ***.eeg*** file to process (with extension).
  - *myposfile****.pos***: input event file (with extension).
  - *myparfile****.par***: text parameter file (with extension), similar to that used for eegavg, but with some additional parameters.

- ***Fields of parameter file and examples***

fileprefix myfilename	Prefix of the output .p files.
nb_eventcode 3	Number of event codes to process.
list_eventcode 2 5 8	List of the event codes to process.
prestim_nbsample 400 400 400	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 800 800 800	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself.
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to average: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be averaged and stored in the output .p files.
ep_channel_ref 0 5 0 6 0 0 0	List of the new reference for each channel before averaging (bipolar montage for instance): 0: no change of the reference, ≠0: electrode number (rank) to which the current channel should be re-referenced. The total number of values is N+2, N being the number of recorded channels in myeegfile.eeg file; the last 2 flags should be set to 0. This field is optional. If omitted, the channels are not modified. In this example, N=5, and channel 1 is unchanged, channel 2 is referenced to channel 5, and channel 4 re-referenced to channel 6.
erpa_positivity_up 1	Value stored in all output .p file to set the default upward/downward positivity orientation for curve display (by erpa). -1 (positivity down) or 1 (positivity up) This field is optional. If omitted, erpa will display curve with positivity down.

erpa_view 4	Value stored in all output .p file. It is used by erpa to set the default mapping view. 1: right view 2: left view 3: top view 4: back view 5: large top view 6: back-top view This field is optional. If omitted, it is set to 0, and erpa default will be top view.
eegstat_type_stat 2	Type of statistical analysis based on single-trial data: 1: Wilcoxon test on each moving time-window with respect to a baseline period 2: Kruskal-Wallis test on each moving time-window between the different event codes.
eegstat_time_hw 200 200 200	List of the time half-windows (in ms) by event code, used to defined the moving time-window on which statistical analysis will be performed. If the value is 0, the analysis is performed on single point. If omitted, all time points are considered in the analysis.
eegstat_time_step 50 50 50	List of the time steps (in ms) by event code, used to defined the moving step of the moving time-window on which statistical analysis will be performed. Optional if eegstat_time_hw is omitted.
eegstat_baseline_start -300 -300 -300	List of the baseline start latencies (in ms) by event code. Required when eegstat_type_stat=1 (Wilcoxon test) or when eegstat_type_stat=2 and eegstat _baseline_flag=1. If omitted, the start latency is the first point of the pre-stimulus baseline defined by prestim_nbsample.
eegstat_baseline_stop -100 -100 -100	List of the baseline end latencies (in ms) by event code. Required when eegstat_type_stat=1 (Wilcoxon test) or when eegstat_type_stat=2 and eegstat _baseline_flag=1. If omitted, the start latency is the first point of the pre-stimulus baseline defined by prestim_nbsample.
eegstat_baseline_flag 1 1 1	List of flags by event code for applying a baseline correction on single-trials prior to the Kruskal-Wallis test. 0: no baseline correction 1: baseline correction applied. If omitted, the default value is 0.
eegstat_flag_fdr 0	Flag allowing to compute False Detection Rate (FDR) after the statistical test. 0: no FDR computation 1: FDR computation, and generation of a *.p file with a mask of the statistical results (Z or H) with the computed FDR (see output files below). This requires a probability (p) threshold value (tfstat_threshold_fdr). This field is optional. If omitted, the default value is 0.
eegstat_threshold_fdr 0.05	To be used in case of tfstat_flag_fdr = 1. Defines the probability (p) threshold value to be used for masking the statistical results (Z or H) by the thresholded FDR statistics.
eegstat_kruskal_modified 0	In case of Kruskal-Wallis test with 2 conditions (2 event codes), this flag allows to indicate that all windows related to event code #2 is compared to a baseline window related to event code #1. For the definition of the baseline, see eegstat_kruskal_modified_baseline_beg_msec and eegstat_kruskal_modified_baseline_end_msec. 0: standard case (no comparison of event code #2 to the baseline of even code #1) 1: comparison of event code #2 to the baseline of event code #1 This field is optional. If omitted, the default value is 0.
eegstat_kruskal_modified_baseline_beg_msec -300	To be used if eegstat_kruskal_modified = 1. Latency of the beginning of the baseline period (in msec) relative to event code #1.
eegstat_kruskal_modified_baseline_end_msec	To be used if eegstat_kruskal_modified = 1.

eegstat_kruskal_modified_baseline_end_msec -50	Latency of the end of the baseline period (in msec) relative to event code #1.

- ***Example***

In the example above, all trials with codes 2, 5 and 8 will be processed on a time window ranging from from 400 samples prior to 1000 samples after event code. The analysis will be performed on channels 1 (not re-referenced), 2 (re-referenced to channel 3),and 4 (re-referenced to channel 5). The statistical analysis is here a Kruskal-Wallis test between event codes 2, 5 and 8. On each trial, the data are averaged over a moving window having 200 ms of half duration (total duration of 400 ms), moving by step of 50 ms, and with a baseline correction defined between –300 and –100 ms.
Output files :

myfilename.2.wil.Z.p myfilename.5.wil.Z.p myfilename.8.wil.Z.p	For the Wilcoxon test: p files of the Z value of the Wilcoxon test for each event code (2, 5, 8).
myfilename.2.wil.p.p myfilename.5.wil.p.p myfilename.8.wil.p.p	For the Wilcoxon test: p files of the probability value (p) of the Wilcoxon test for each event code.
myfilename.2.5.8.kw.H.p	For the Kruskal-Wallis test: p files of the H value of the Kruskal-Wallis test computed on the 3 event codes (2, 5, 8)
myfilename.2.5.8.kw.p.p	For the Kruskal-Wallis test: p files of the probability value (p) of the Kruskal-Wallis test computed on the 3 event codes (2, 5, 8)
myfilename.2.Z.fdr.p	For the Wilcoxon test: p files of the Wilcoxon Z value masked by the thresholded probability value (p) of the FDR procedure.
myfilename.2.5.8.H.fdr.p	For the Kruskal-Wallis test: p files of the Kruskal-Wallis H value masked by the thresholded probability value (p) of the FDR procedure.

- ***Comments***
  1. eegstat_kruskal_modified : This option allows, for instance, to compare each position in time in an active condition (related to event code #2) to a resting state defined by the mean value computed in a given time-window (related to event code #1).
  2. eegstat_flag_fdr : This option allows to compute the False Detection Rate statistics after a Wilcoxon or a Kruskal-Wallis test. This is a possible solution for multiple testing problem. See: Genovese, C. R., N. A. Lazar, et al. (2002). "Thresholding of statistical maps in functional neuroimaging using the false discovery rate." Neuroimage 15(4): 870-8.

     Warning : no control of the compatibility of the prestim/post-stim duration and the baseline for statistics is performed. Incorrect results in case of a too long baseline.

  3. See eegchref [2] to create a re-referenced .eeg data file (several referencing options available).

- ***Current version***

1.24 10-04-2014

- ***History***
  - 1.12 25-08-2005 (PEA) : modification for time window : same as eegranddiff.
  - 1.13 07-02-2006 (PEA) : correction in baseline computation.
  - 1.14 08-02-2006 (PEA) : adds computation on all samples : eegstat_time_hw and eegstat_time_step become optional. If omitted, computation is done on all samples.
  - 1.15 18-04-2006 (OB) : correction for exact time window values (not rounded to 1 msec anymore). The output sampling step is exactly the one given as parameter with eegstat_time_step.
  - 1.16 13-08-2008 (PEA) : minor modification.
  - 1.17 09-09-2008 (PEA) : adds kruskal_wallis_modified case : for 2 event types, the baseline is defined in the first event type window. Adds the following fields to the parameter file :
    eegstat_kruskal_modified (optional : value 1 or 0, 0 if omitted) : if 1, Kruskal-Wallis is computed as defined above.
    eegstat_kruskal_modified_baseline_beg_msec (required only if eegstat_kruskal_modified=1) : beginning (in msec) of the baseline.
    eegstat_kruskal_modified_baseline_end_msec (required only if eegstat_kruskal_modified=1) : end (in msec) of

the baseline.
Adds masking with FDR result and creation of new output files (.Z.fdr.p (Wilcoxon) ou .H.fdr.p (Kruskal-Wallis)).
Adds the following fields to the parameter file :
eegstat_flag_fdr (optional : value 1 or 0, 0 if omitted) : if 1, FDR is computed.
eegstat_threshold_fdr (required only if eegstat_flag_fdr=1) : probability (p) threshold value to be used for masking the statistical results (Z or H) by the thresholded FDR statistics.

- 1.18 14-10-2009 (PEA) : correction baseline computation (Kruskal-Wallis).
- 1.19 23-09-2010 (PEA) : update to use cmake and free release of Elan.
- 1.20 15-04-2011 (PEA) : changes Shell sort to Heap sort in FDR (faster algorithm).
- 1.21 07-10-2011 (PEA) : adds test for at least 2 samples to test in Wilcoxon test.
- 1.22 17-07-2012 (PEA) : fixes an error in wilcox function of libstat when all samples are ties.
- 1.23 21-03-2013 (PEA) : fixes an error of parameter file reading for eegstat_time_hw and eegstat_time_step parameters when working on all samples.
- 1.24 10-04-2014 (PEA) : fixes memory desallocation in parameter reading. Fixes output FDR filename creation for Kruskal-Wallis.

- ***Files***

  $ELANPATH/bin/eegstat

- ***See also***

  eegavg [28], eegranddiff [30], eegrandadd [31], eegchref [2]

eegswapchan

- ***Description***

  Swaps two channels in an EEG file (change order of channels) and saves in a new EEG file.

- ***Usage***

  eegswapchan myeegfilein myeegfileout rank_chan_1 rank_chan_2

  with :

  - myeegfilein : input EEG file to process (without extension).
  - myeegfileout : output EEG file (without extension).
  - rank_chan_1 : indice of 1st channel to swap (starting from 1).
  - rank_chan_2 : indice of 2nd channel to swap (starting from 1).

- ***Fields of parameter file and example***

- ***Examples***

- ***Comments***

- ***Current version***

  1.03 24-09-2010

- ***History***
  - 1.02 13-08-2007 (PEA) : 1st documented version.
  - 1.03 24-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header.

- ***Files***

  $ELANPATH/bin/eegswapchan

- ***See also***

eegval

- ***Description***

  Compute mean value of temporal window on each single trials. Values are written in text files (one file containing all single trials for each event code).

- ***Usage***

  eegval myeegfilein.eeg myeventfile.pos myparameterfile.par [-e] [-c] [-dc]

  with :

  - myeegfilein.eeg : input EEG file (with extension).
  - myeventfile.pos : event file to process (with extension).
  - myparameterfile.par : parameter file (with extension) containing parameters for value extraction.
  - options :
    - -e : omit column titles in output file.
    - -c : compute mean of squares.
    - -dc : substract baseline mean value before square.

- ***Fields of parameter file and example***

fileprefix myfilename	Prefix of the output text files (suffix .single.val).
nb_eventcode 2	Number of event codes to process.
list_eventcode 2 5	List of the event codes to process.
prestim_nbsample 400 800	List of the numbers of samples in the prestimulus period; one value for each event code.
poststim_nbsample 1000 1200	List of the numbers of samples in the poststimulus period; one value for each event code; the total number of samples of the analysis is prestim_nbsample + poststim_nbsample + 1, the extra sample corresponds to the event itself
ep_channel_flag 1 1 0 1 0 0 0	List of the channels to compute: 1/0 for selected/unselected channels; the total number of flags is N+2, N being the number of recorded channels in myeegfilein.eeg file; the last 2 flags should be set to 0. In this example, N=5, and only channels number 1, 2, 4 will be computed and stored in the output text files.
single_lat_min 100 200	Value in msec of the beginning of the window to average for each event.
single_lat_max 150 300	Value in msec of the end of the window to average for each event.
single_base_min -100 -100	Value in msec of the beginning of the baseline for each event (even if it is not used). The baseline is substracted only in case of -dc option is set.
single_base_max -50 -50	Value in msec of the end of the baseline for each event (even if it is not used). The baseline is substracted only in case of -dc option is set.

- ***Example***

- ***Comments***

- ***Current version***

  3.13 24-09-2010

- ***History***
  - 3.12 13-01-2004 (PEA) : 1st version with documentation.
  - 3.13 24-09-2010 (PEA) : update to use cmake and free release of Elan. Remove all static allocations. Change messages language to english.

- ***Files***

  $ELANPATH/bin/eegval

- ***See also***

eegwritetrig

- ***Description***

  Writes events from text file to EEG digital channels.

- ***Usage***

  eegwritetrig myeegfilein.eeg myeventfile.pos myeegfileout.eeg

  with :

  - myeegfilein.eeg : input EEG file to process (with extension).
  - myeventfile.pos : event file to process (with extension).
  - myeegfileout.eeg : output EEG file (with extension) containing events from myeventfile.pos .

- ***Fields of parameter file and example***

- ***Examples***

- ***Comments***

- ***Current version***

  1.04 24-09-2010

- ***History***
  - 1.03 19-11-2003 (CTB) : 1st documented version.
  - 1.04 24-09-2010 (PEA) : update to use cmake and free release of Elan. Remove static allocation for reading EEG file header. Change messages language to english.

- ***Files***

  $ELANPATH/bin/eegwritetrig

- ***See also***

Lyon Neuroscience Research Center - Brain Dynamic and Cognition team

CRNL

Source URL: http://elan.lyon.inserm.fr/?q=ref_eeg_proc_tool

Links:
[1] http://elan.lyon.inserm.fr/?q=sites/default/files/ctf275_meg.par
[2] http://elan.lyon.inserm.fr/?q=eegchref
[3] http://elan.lyon.inserm.fr/?q=eegstat
[4] http://elan.lyon.inserm.fr/sites/default/files/ctf275_meg.par
[5] http://research.ics.tkk.fi/ica/fastica/
[6] http://elan.lyon.inserm.fr/?q=eegproject
[7] http://elan.lyon.inserm.fr/?q=eegfiltica
[8] http://elan.lyon.inserm.fr/?q=matrix2p
[9] http://elan.lyon.inserm.fr/?q=eegpos
[10] http://elan.lyon.inserm.fr/sites/default/files/fig_eegdetecttrigl.png
[11] http://elan.lyon.inserm.fr/?q=sites/default/files/lowpass_butter_10hz_epoch.par
[12] http://elan.lyon.inserm.fr/?q=sites/default/files/lowpass_butter_10hz_coefs_epoch.par
[13] http://elan.lyon.inserm.fr/?q=sites/default/files/bp_2-30Hz_epoch.par
[14] http://elan.lyon.inserm.fr/?q=epfiltfilt
[15] http://elan.lyon.inserm.fr/?q=eegfiltfilt
[16] http://elan.lyon.inserm.fr/sites/default/files/bp_2-30Hz_epoch.par
[17] http://elan.lyon.inserm.fr/sites/default/files/lowpass_butter_10hz_coefs_epoch.par
[18] http://elan.lyon.inserm.fr/sites/default/files/lowpass_butter_10hz_epoch.par
[19] http://elan.lyon.inserm.fr/?q=sites/default/files/lowpass_butter_10hz.par
[20] http://elan.lyon.inserm.fr/?q=sites/default/files/lowpass_butter_10hz_coefs.par
[21] http://elan.lyon.inserm.fr/?q=sites/default/files/bp_2-30Hz.par
[22] http://elan.lyon.inserm.fr/?q=eegepochfiltfilt
[23] http://elan.lyon.inserm.fr/?q=eegresample
[24] http://elan.lyon.inserm.fr/sites/default/files/lowpass_butter_10hz.par
[25] http://elan.lyon.inserm.fr/sites/default/files/lowpass_butter_10hz_coefs.par
[26] http://elan.lyon.inserm.fr/sites/default/files/bp_2-30Hz.par
[27] http://elan.lyon.inserm.fr/?q=eegcomponent
[28] http://elan.lyon.inserm.fr/?q=eegavg
[29] http://elan.lyon.inserm.fr/?q=node/72
[30] http://elan.lyon.inserm.fr/?q=eegranddiff
[31] http://elan.lyon.inserm.fr/?q=eegrandadd
[32] http://elan.lyon.inserm.fr/?q=epranddiff
[33] http://elan.lyon.inserm.fr/?q=eechref
[34] http://elan.lyon.inserm.fr/?q=sites/default/files/resync.par
[35] http://elan.lyon.inserm.fr/sites/default/files/resync.par
[36] http://elan.lyon.inserm.fr/?q=node/56