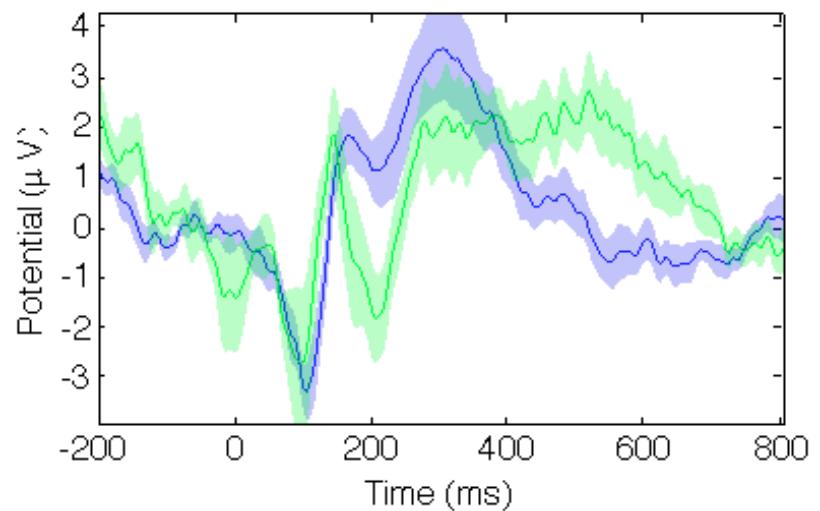
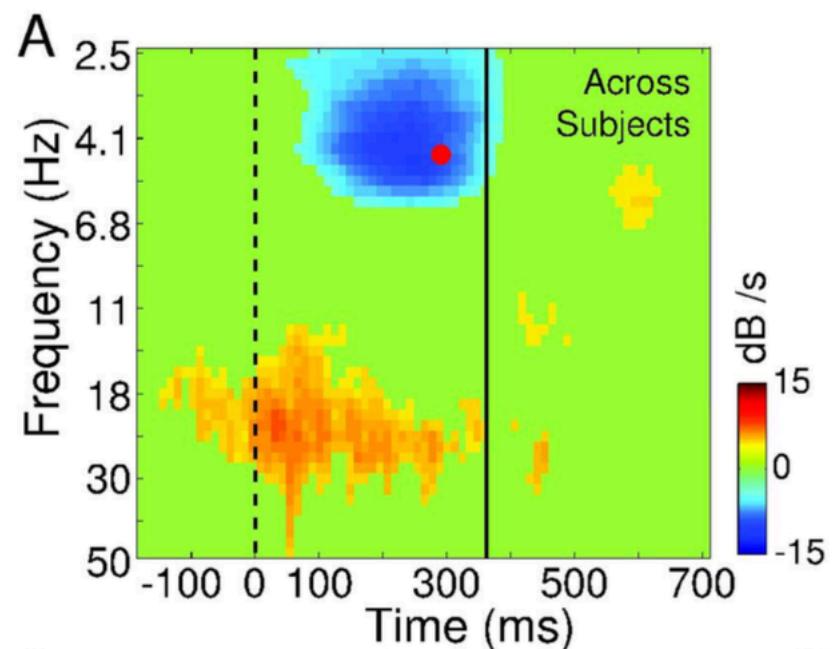
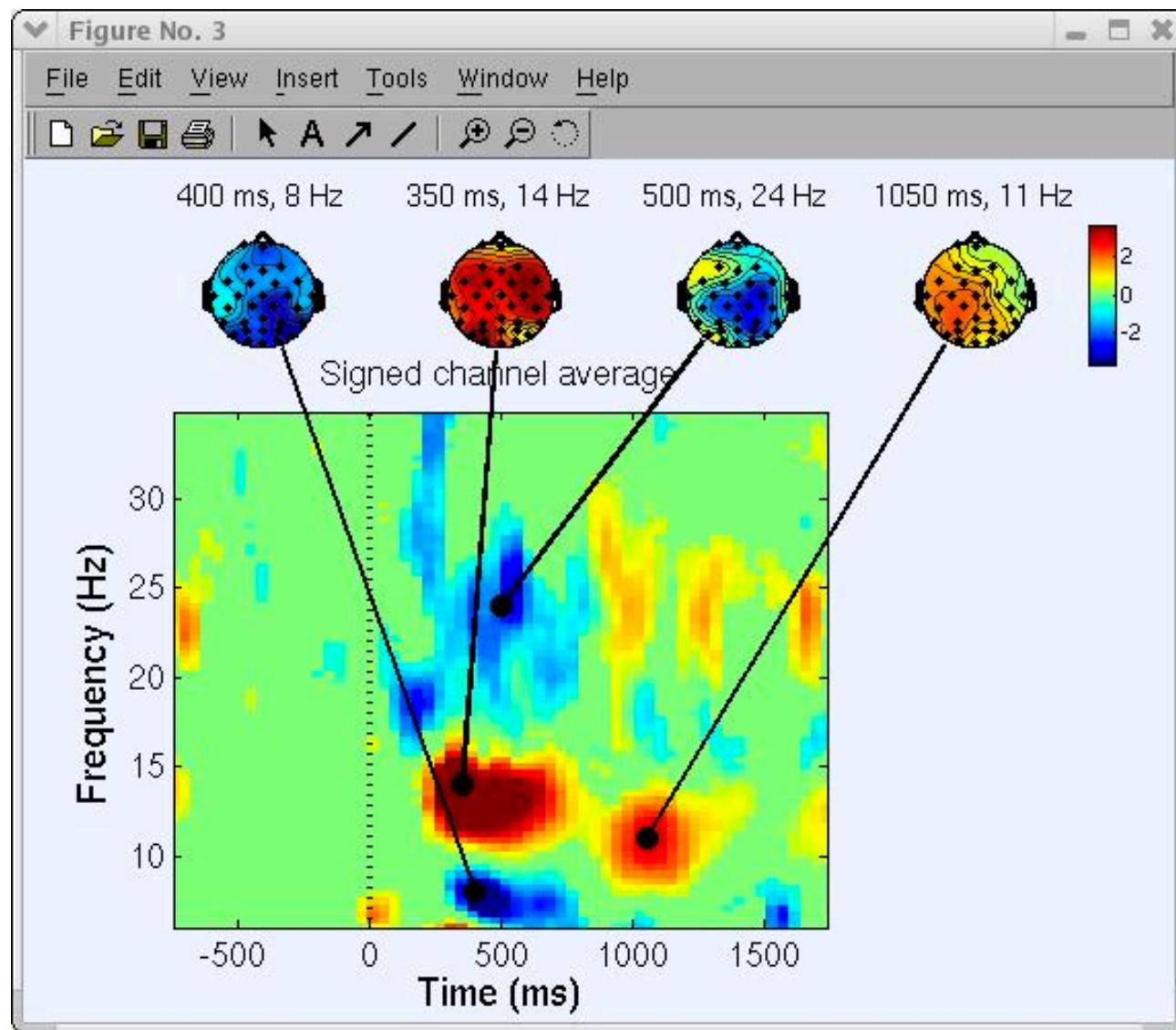


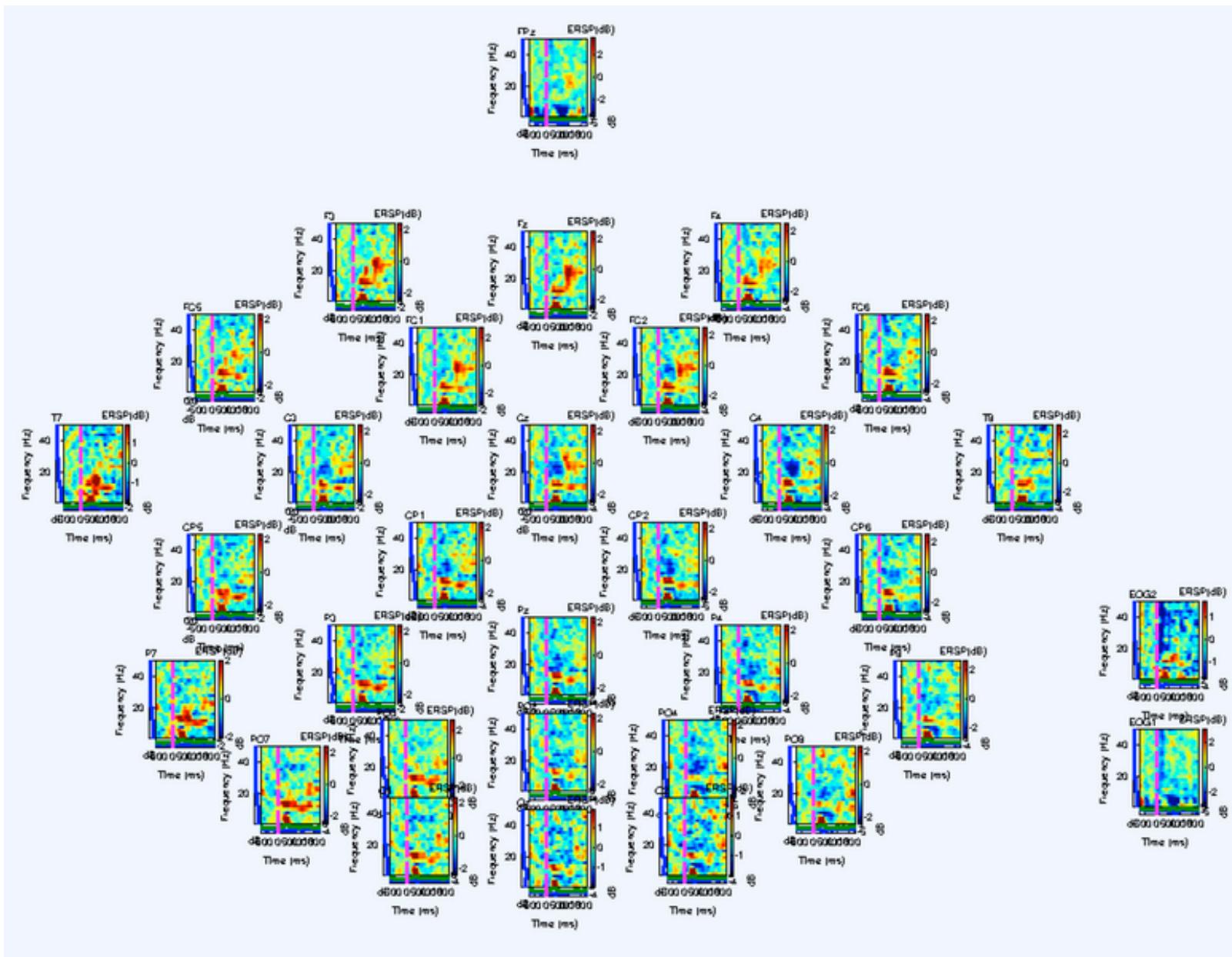
Basic Scripting in EEGLAB

Why scripting?

- corimage.m
- eeg_regePOCHS.m
- eegmovie.m
- fastregress.m
- fieldtrip2eeglab.m
- fillcurves.m
- compile_eeglab.m

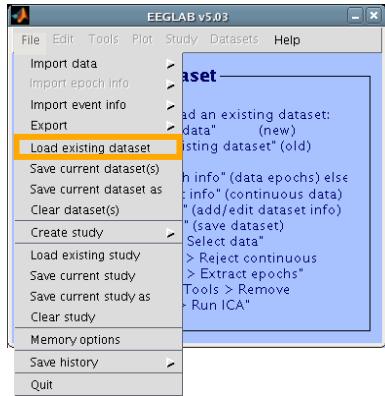






Metaplotopo function

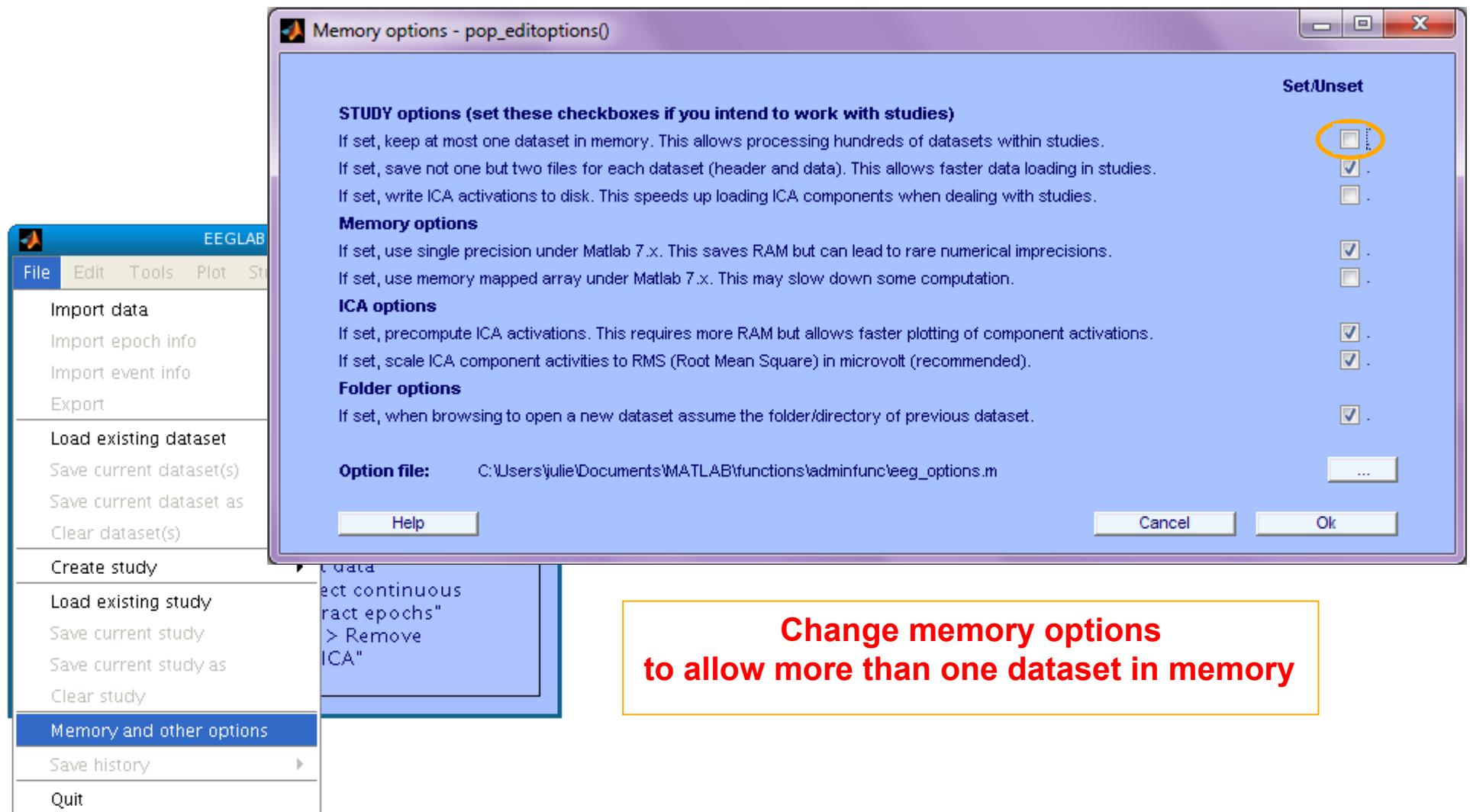
Command line tools



(Menus write both dataset and global history)

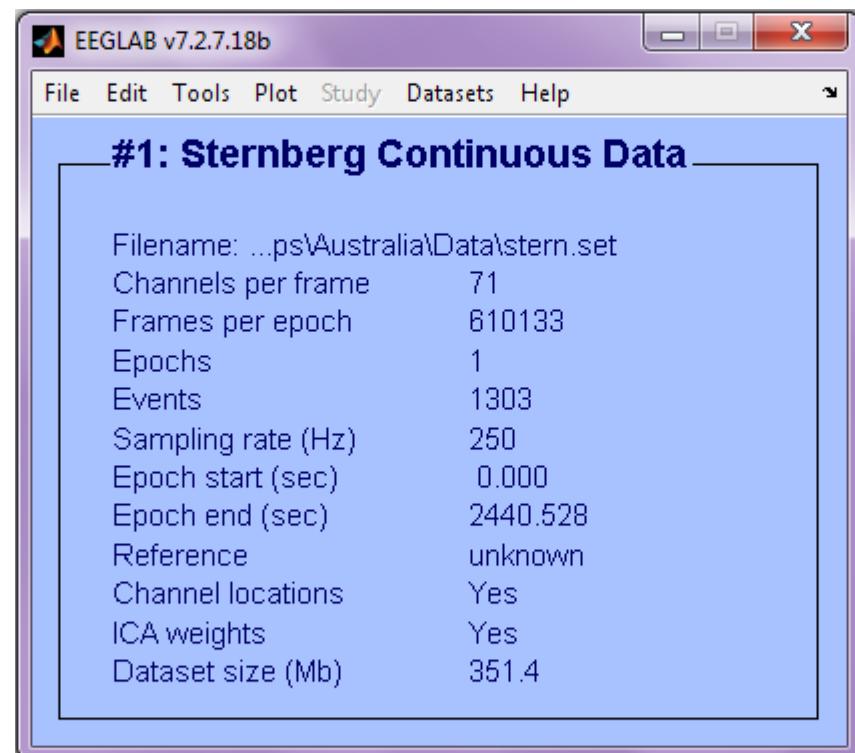
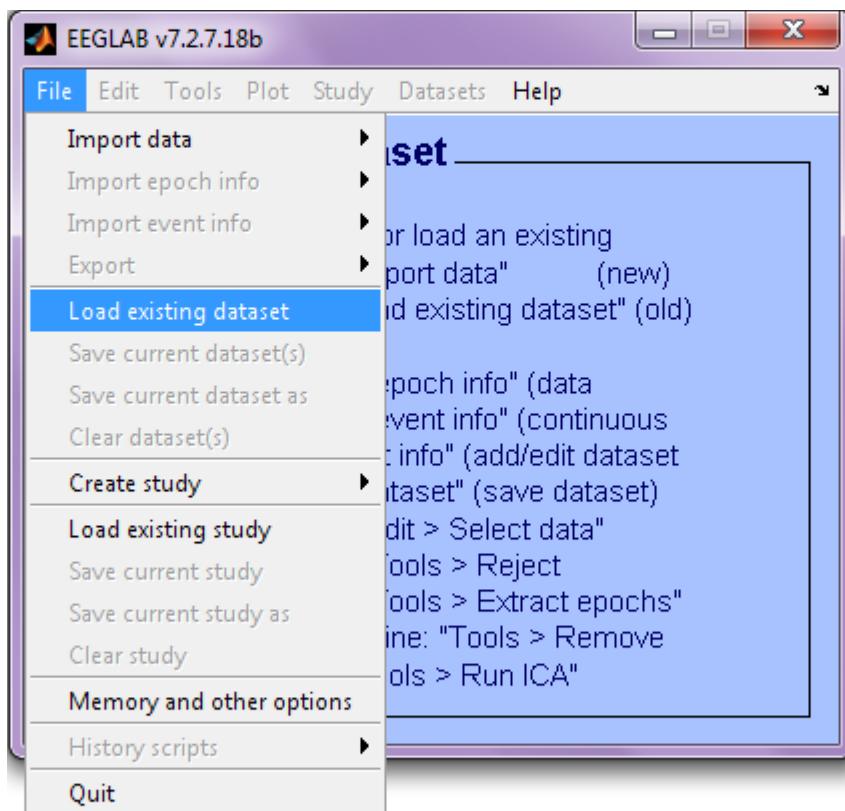
- Automated processing on groups of subjects (possibly on several processors).
- Richer options for plotting and processing functions (time-frequency decompositions, ...)
- Selecting data/epoch based on event context
- Custom processing...

Memory options

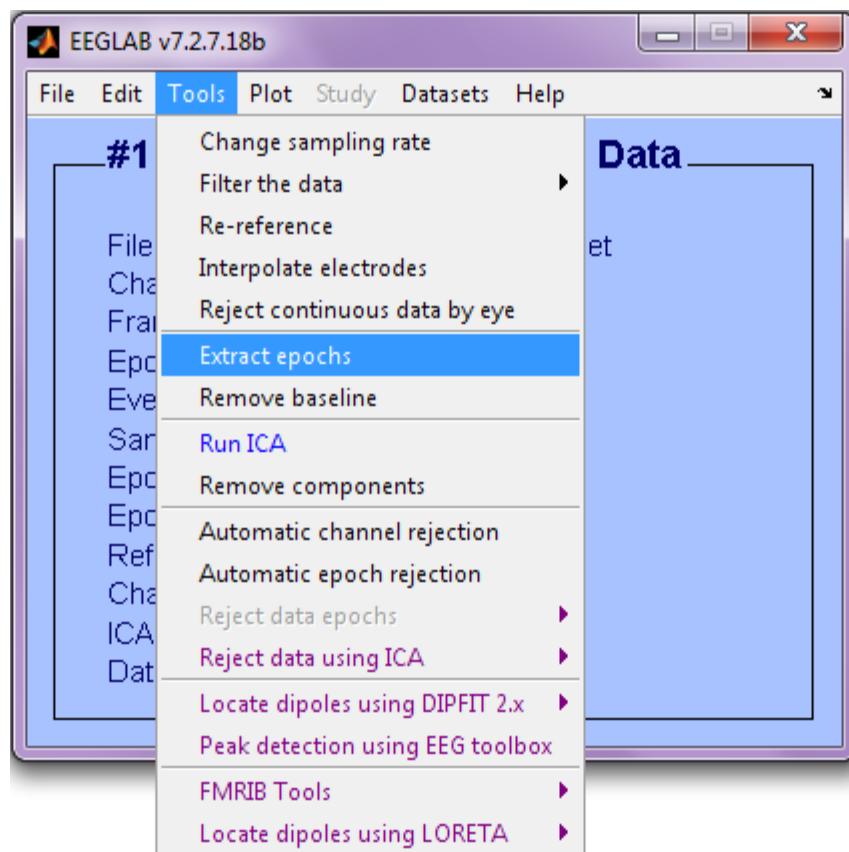


Create a script from ‘eegh’ output

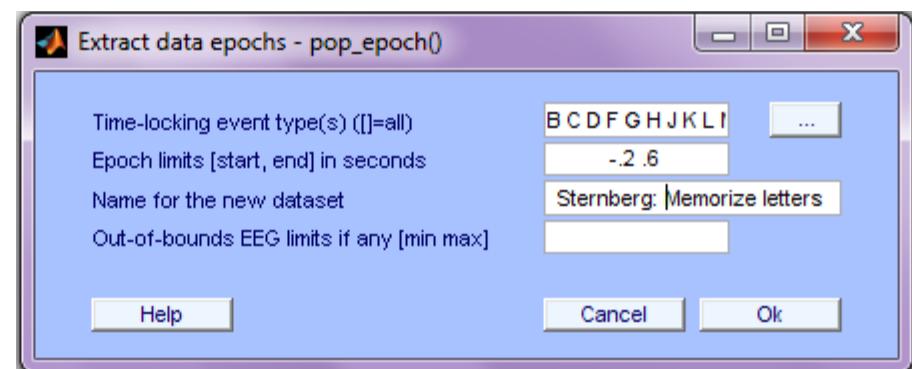
Start by loading a continuous dataset



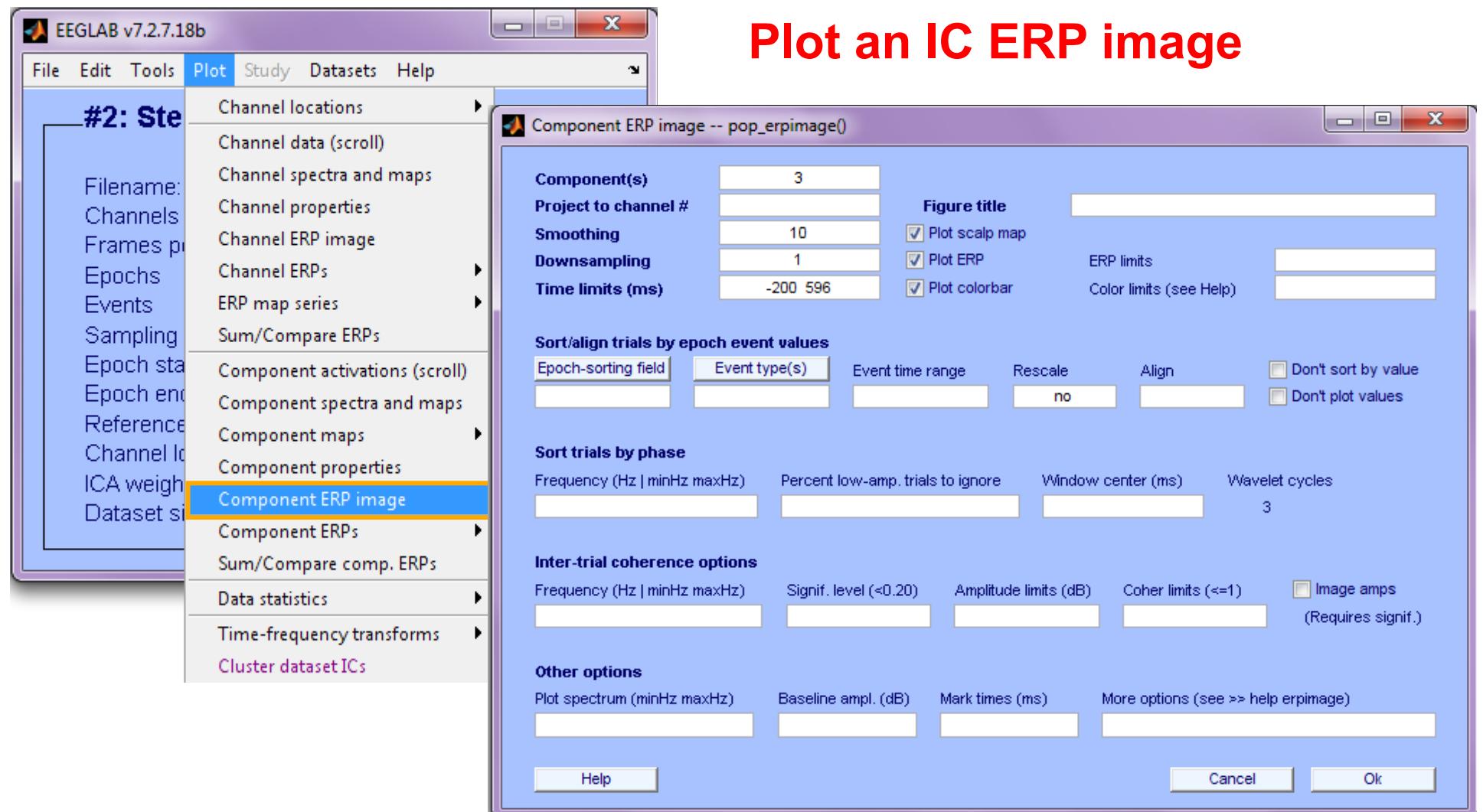
Create a script from ‘eegh’ output



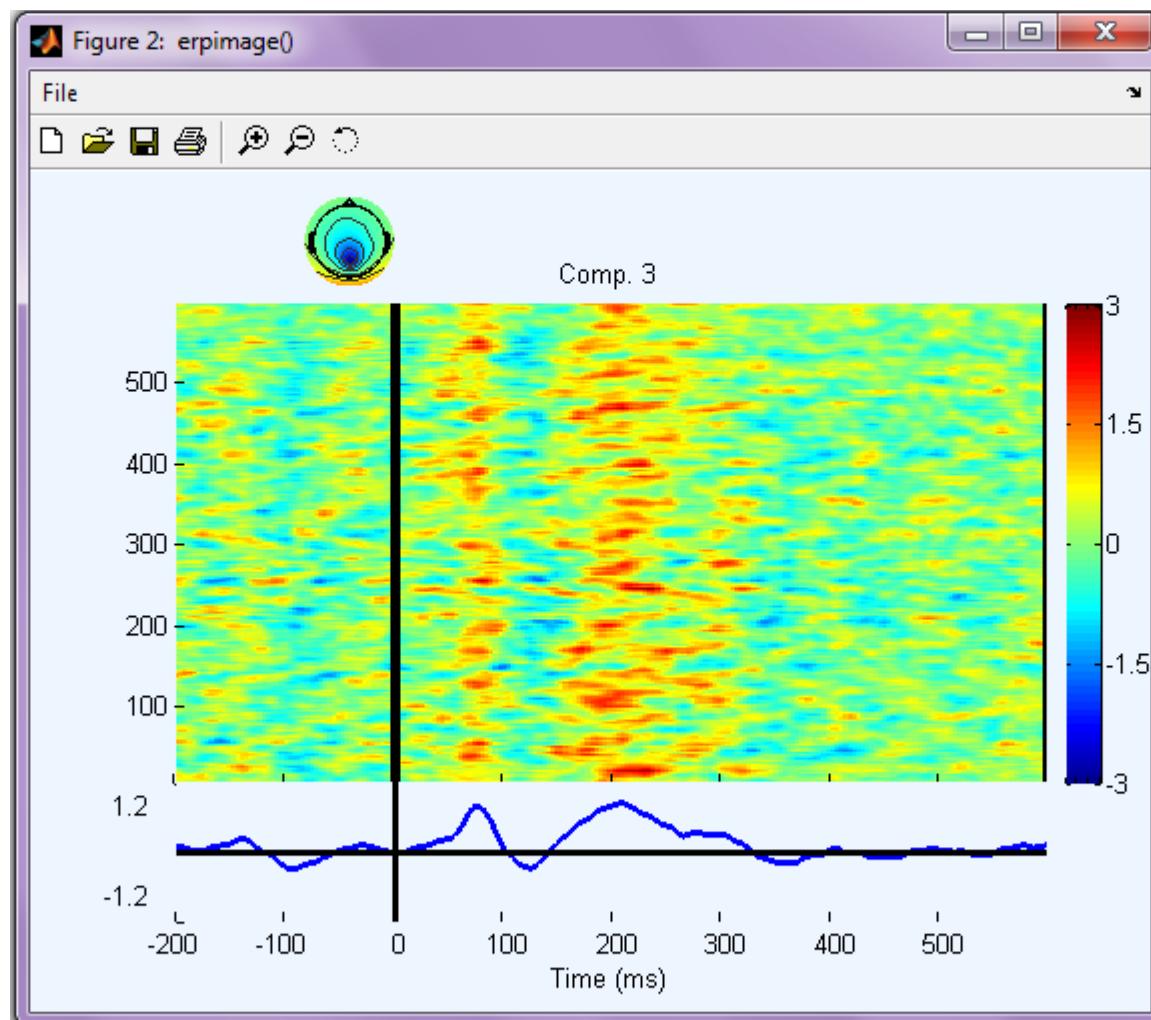
Epoch on Memorize letters



Create a script from ‘eegh’ output



Create a script from ‘eegh’ output



Retrieve commands from eegh

Write a script to do this:

`>> eegh`

Retrieve commands from eegh

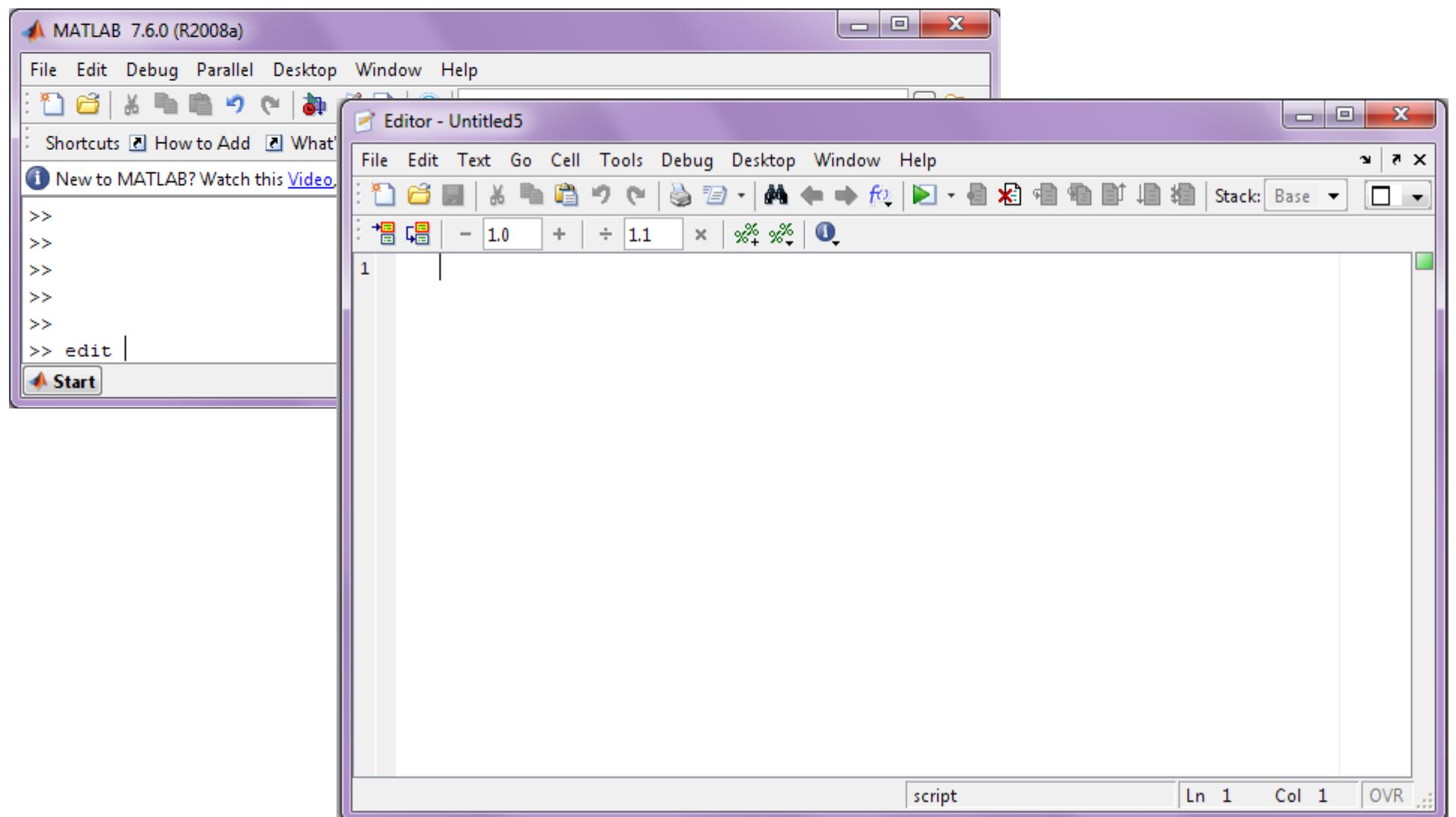
```
>> eegh
[ALLEEG EEG CURRENTSET ALLCOM] = eeglab;

EEG = pop_loadset('filename', 'stern_125Hz.set');
[ALLEEG EEG CURRENTSET] = eeg_store(ALLEEG, EEG, 0);

EEG = pop_epoch( EEG, {'B' 'C' 'D' ... }, [-0.2 0.6], 'newname',
'Memorize epochs', 'epochinfo', 'yes');
[ALLEEG EEG CURRENTSET] = eeg_store(ALLEEG, EEG, 1);
EEG = pop_rmbase( EEG, [-200 0]);
[ALLEEG EEG] = eeg_store(ALLEEG, EEG, CURRENTSET);

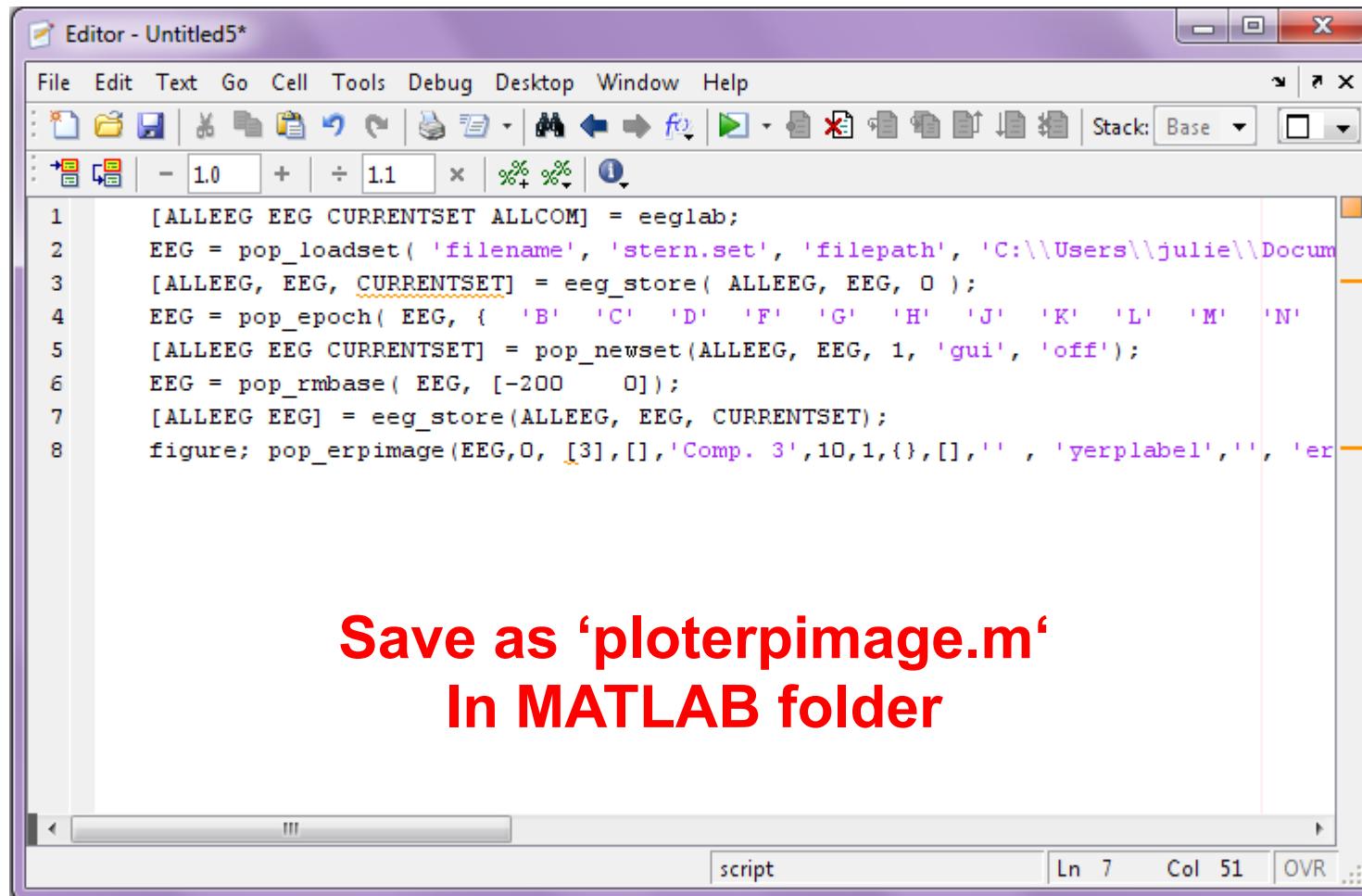
figure; pop_erpimage(EEG,0, [3],[],'Comp. 3',10,1,{},
[],'', 'yerplabel', '', 'erp', 'on', 'cbar', 'on','topo',
{mean(EEG.icawinv(:,[3]),2) EEG.chanlocs EEG.chaninfo});
```

Create a Matlab script



Create a Matlab script

Copy and paste from Matlab window:



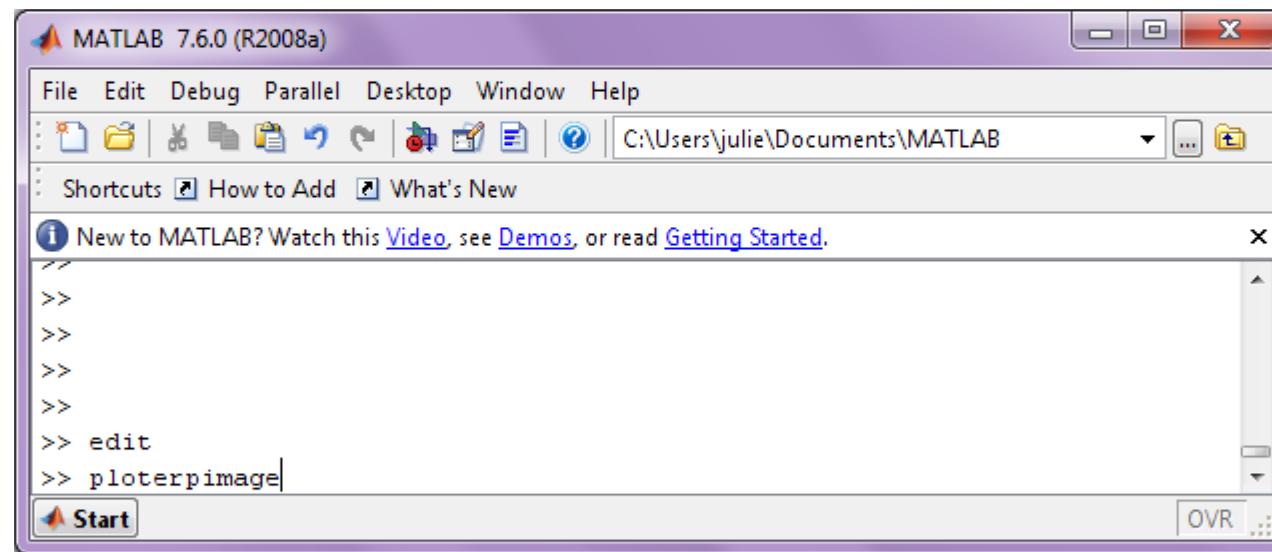
The screenshot shows the MATLAB Editor window titled "Editor - Untitled5*". The window contains the following MATLAB script:

```
1 [ALLEEG EEG CURRENTSET ALLCOM] = eeglab;
2 EEG = pop_loadset( 'filename', 'stern.set', 'filepath', 'C:\\\\Users\\\\julie\\\\Docum
3 [ALLEEG, EEG, CURRENTSET] = eeg_store( ALLEEG, EEG, 0 );
4 EEG = pop_epoch( EEG, { 'B' 'C' 'D' 'F' 'G' 'H' 'J' 'K' 'L' 'M' 'N'
5 [ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1, 'gui', 'off');
6 EEG = pop_rmbase( EEG, [-200 0]);
7 [ALLEEG EEG] = eeg_store(ALLEEG, EEG, CURRENTSET);
8 figure; pop_erpimage(EEG,0, [3],[],'Comp. 3',10,1,{},[],'', 'yerplabel','','er
```

Below the code, red text instructions read:

**Save as 'ploterpimage.m'
In MATLAB folder**

Run your new script



Exercise page 1

```
>> eeglab

% load dataset stern_125.set
% epoch on 'memorize' letter' B, C, etc...
% plot erpimage for component 3

>> eegh

% open Matlab editor

>> edit

% copy & paste eegh results into a new
% file and save it (ploterpimage.m)

>> clear
>> close all
>> ploterpimage
>> eeglab redraw
```

Using EEGLAB history for basic scripting

Task 1

Create a script from ‘eegh’ output

Task 2

Adapt your script with variables

Task 3

Create a Matlab function

Task 4

Exercise...

Matlab basics -- Briefly

Variable = word with an assigned value (type 'whos')

Examples:

```
% vector of numbers:  
mynumbers = [1, 2, 3, 5:10];  
          (Square brackets: concatenate anything within)  
% access vector elements:  
>> mynumbers(2)  
ans =  
     2  
% cell array of strings:  
mylabels = {'stimulus','response'};  
% access cell array elements:  
>> mylabels{2}  
ans =  
    response
```

Parameterize a script

```
>> eegh  
[ALLEEG EEG CURRENTSET ALLCOM] = eeglab;  
  
EEG = pop_loadset('filename', 'stern.set','filepath',...  
'..\EEGLAB_Workshop\Data\');  
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 0);  
  
EEG = pop_epoch( EEG, {'B' 'C' 'D' ...}, [-0.2 0.6],...  
'newname', 'Memorize epochs', 'epochinfo', 'yes');  
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1);  
EEG = pop_rmbase( EEG, [-200 0]);  
[ALLEEG EEG] = eeg_store(ALLEEG, EEG, CURRENTSET);  
  
figure; pop_erpimage(EEG,0, [3],[],'Comp. 3',10,1,{},[],...  
'', 'yerplabel', '', 'erp', 'on', 'cbar', 'on', 'topo',...  
{mean(EEG.icawinv(:,[3]),2) EEG.chanlocs EEG.chaninfo});
```

Parameterize a script

```
% Variables-----
dataset = 'stern.set';
datpath = 'C:\MATLAB\...\EEGLAB_Workshop\Data\' ;
epochletts = {'B' 'C' 'D' 'E' 'F' 'G' 'H' 'I' 'J' 'K' 'L' 'M' 'N' 'O'...} ;
datsetname = 'Memorize epochs' ;
comp = [3] ;

[ALLEEG EEG CURRENTSET ALLCOM] = eeglab;
EEG = pop_loadset('filename', dataset,'filepath', datpath) ;
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 0) ;

EEG = pop_epoch( EEG, epochletts , [-0.2 0.6],...
'newname', datsetname, 'epochinfo', 'yes') ;
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1) ;
EEG = pop_rmbase( EEG, [-200 0]) ;
[ALLEEG EEG] = eeg_store(ALLEEG, EEG, CURRENTSET) ;

figure; pop_erpimage(EEG,0, [comp],[],['Comp.',int2str(comp)],...
10,1,{},[],'',yerplabel,'','erp','on','cbar','on','topo',...
{mean(EEG.icawinv(:,[comp ]),2),EEG.chanlocs EEG.chaninfo});
```

Exercice page 2

- 1) Open stern125Hz.set (eeglab_data folder)
- 2) Epoch on memorize letters (all single capital letters - excluding WM, gL where L is letter)
- 3) Plot -> ERPimage for component 3
- 4) Save a script using the history (eegh) and execute it
- 5) Parametrize the script and execute it**
- 6) Write a function instead of a script and call it from the Matlab command line

Using EEGLAB history for basic scripting

Task 1

Create a script from ‘eegh’ output

Task 2

Adapt your script with variables

Task 3

Create a Matlab function

Task 4

Exercise...

Write a Matlab function

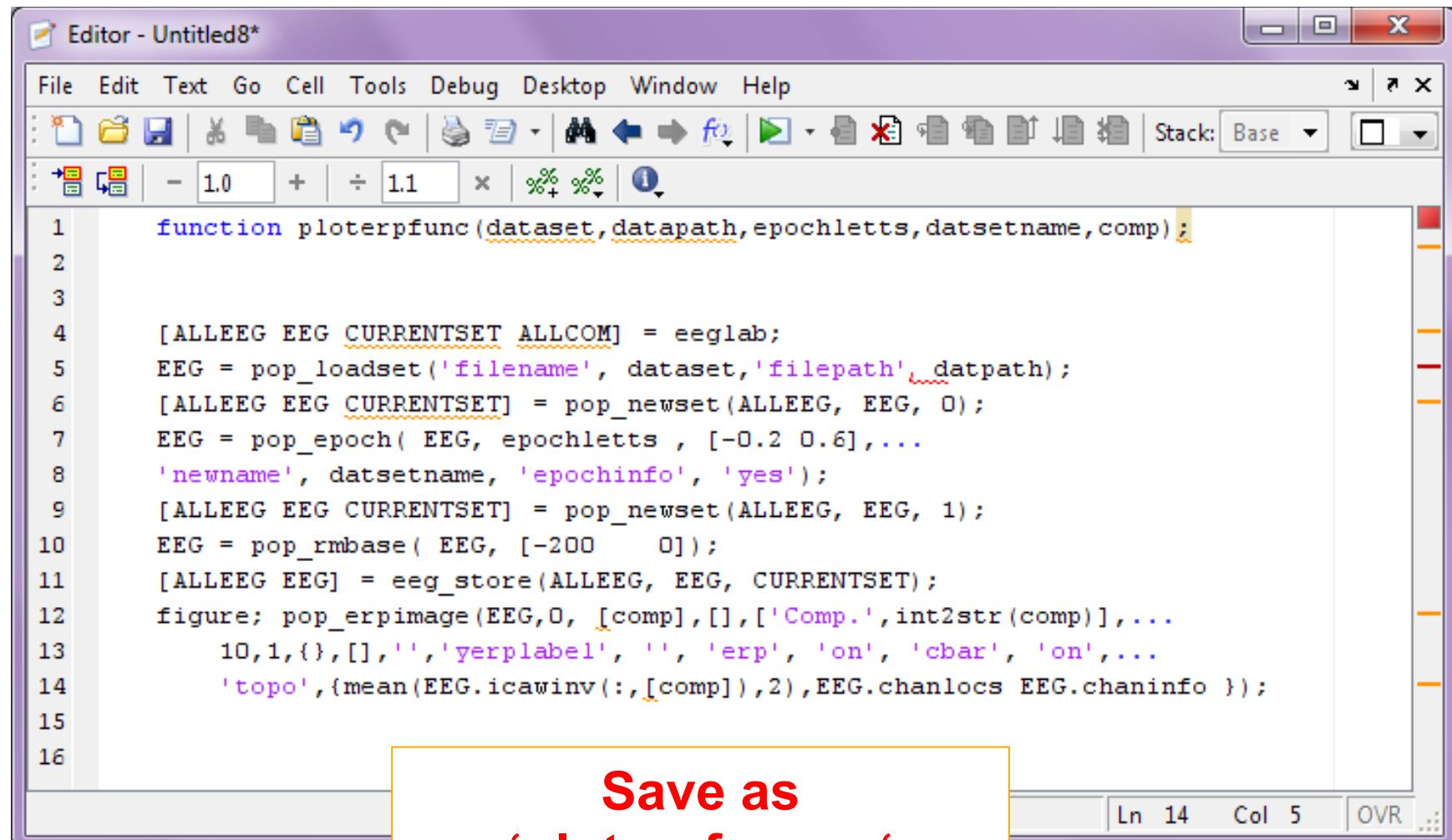
Matlab functions:

1. Take arguments
2. Can return variables
3. Do not draw variables from the global workspace
Need all variables called internally or passed as arguments

Example function

```
% Variables-----  
  
dataset = 'stern.set';  
datapath = '...\\EEGLAB_Workshop\\Data\\';  
epochletts = {'B'    'C'    'D' ...};  
datsetname = 'Memorize epochs';  
comp = [3];  
  
ploterpfnc(dataset, datapath, epochletts, datsetname,  
comp);
```

Example function

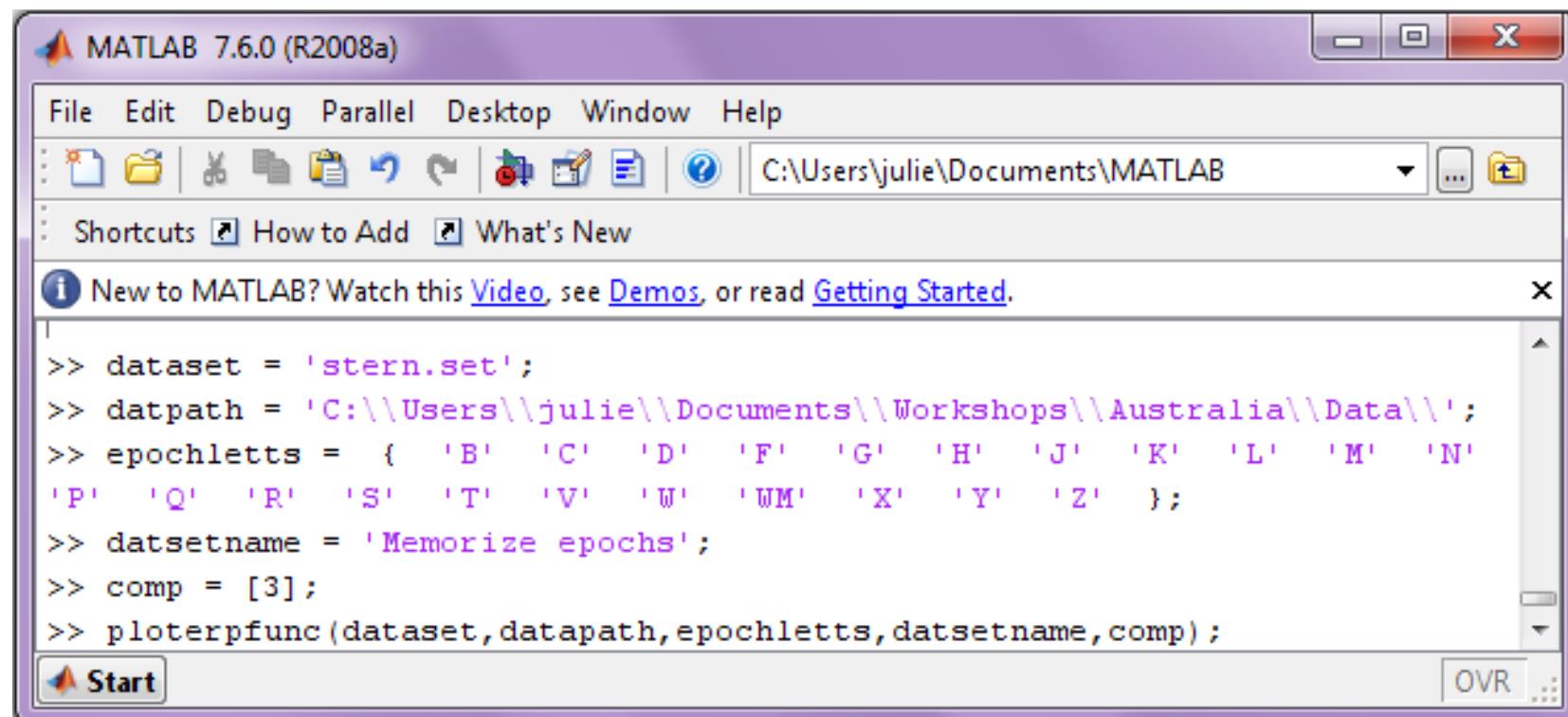


The screenshot shows the MATLAB Editor window titled "Editor - Untitled8*". The window contains a function definition:

```
function ploterpfnc(dataset,datapath,epochletts,datsetname,comp);  
%  
[ALLEEG EEG CURRENTSET ALLCOM] = eeglab;  
EEG = pop_loadset('filename', dataset,'filepath'\datapath);  
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 0);  
EEG = pop_epoch( EEG, epochletts , [-0.2 0.6],...  
'newname', datsetname, 'epochinfo', 'yes');  
[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1);  
EEG = pop_rmbase( EEG, [-200 0]);  
[ALLEEG EEG] = eeg_store(ALLEEG, EEG, CURRENTSET);  
figure; pop_erpimage(EEG,0, [comp],[],['Comp.',int2str(comp)],...  
10,1,{},[],'', 'yerplabel', '', 'erp', 'on', 'cbar', 'on',...
'topo',{mean(EEG.icawinv(:,:,comp)),2},EEG.chanlocs EEG.chaninfo );
```

Save as
'ploterpfnc.m'
In MATLAB folder

Run your function in Matlab



Using EEGLAB history for basic scripting

Task 1

Create a script from ‘eegh’ output

Task 2

Adapt your script with variables

Task 3

Create a Matlab function

Task 4

Exercise...

Exercice page 2

- 1) Open stern125Hz.set (eeglab_data folder)
- 2) Epoch on memorize letters (all single capital letters - excluding WM, gL where L is letter)
- 3) Plot -> ERPimage for component 3
- 4) Save a script using the history (eegh) and execute it
- 5) Parametrize the script and execute it
- 6) Write a function instead of a script and call it from the Matlab command line

EEGLAB Data Structures

1. EEG
 - .data - root 'dataset' structure
 - .chanlocs - the dataset data (2-D, 3-D matrix)
 - .event - channel locations substructure
 - .epoch - data events substructure
 - .epoch - data epochs substructure
2. ALLEEG - vector of loaded EEG datasets
3. CURRENTSET - index in ALLEEG of current EEG dataset

$$\text{ALLEEG}(\text{CURRENTSET}) = \text{EEG}$$

In practice

`[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, x)`
or `[ALLEEG EEG CURRENTSET] = eeg_store(ALLEEG, EEG, x)`

EEGLAB Data Structures

1. EEG
 - .data - root 'dataset' structure
 - .chanlocs - the dataset data (2-D, 3-D matrix)
 - .event - channel locations substructure
 - .epoch - data events substructure
 - .epoch - data epochs substructure
2. ALLEEG - vector of loaded EEG datasets
3. CURRENTSET - index in ALLEEG of current EEG dataset

4. CURRENTSTUDY - 0 or 1 based on whether a STUDY is being processed
5. STUDY
 - .cluster - root 'studyset' structure
 - .cluster - component clustering substructure

If CURRENTSTUDY is 1 process STUDY

If CURRENTSTUDY is 0 process datasets

EEGLAB Data Structures

1. EEG
 - .data - root 'dataset' structure
 - .chanlocs - the dataset data (2-D, 3-D matrix)
 - .event - channel locations substructure
 - .epoch - data events substructure
 - .epoch - data epochs substructure
2. ALLEEG - vector of loaded EEG datasets
3. CURRENTSET - index in ALLEEG of current EEG dataset
4. CURRENTSTUDY - 0 or 1 based on whether a STUDY is being processed
5. STUDY
 - .cluster - root 'studyset' structure
 - .cluster - component clustering substructure
6. LASTCOM - last command
7. ALLCOM - all command (history)
8. PLUGINLIST - list of all plugins

EEG structure

EEG =

```
setname:'Epoched from "ee114 continuous"'  
filename:'ee114squaresepochechs.set'  
filepath:'/home/arno/ee114/'  
pnts:384  
nbchan:32  
trials:80  
srate:128  
xmin:-1  
xmax:1.9922  
data:[32x384x80 double]  
icawinv:[32x32 double]  
icasphere:[32x32 double]  
icaweights:[32x32 double]  
icaact:[32x384x80 double]  
event:[1x157 struct]  
epoch:[1x80 struct]  
chanlocs:[1x32 struct]  
comments:[8x150 char]  
averref:'no'  
rt:[]  
eventdescription:{1x5 cell}  
epochdescription:{}  
specdata:[]  
specicaact:[]  
reject:[1x1 struct]  
stats:[1x1 struct]  
splinefile:[]  
ref:'common'  
history:[7x138 char]  
urevent:[1x154 struct]  
times:[1x384 double]
```

Number of data points per trial

Number of channels

Number of trials

Sampling rate

Time limits

Data

ICA scalp maps

ICA activity

Epoch/event information

Channel location

EEG structure

The EEG
structure can
be extended to
include new
fields

store
information for
future access

EEG =

```
setname:'Epoched from "ee114 continuous"'  
filename:'ee114squaresepochechs.set'  
filepath:'/home/arno/ee114/'  
pnts:384  
nbchan:32  
trials:80  
srate:128  
xmin:-1  
xmax:1.9922  
data:[32x384x80 double]  
icawinv:[32x32 double]  
icasphere:[32x32 double]  
icaweights:[32x32 double]  
icaact:[32x384x80 double]  
event:[1x157 struct]  
epoch:[1x80 struct]  
chanlocs:[1x32 struct]  
comments:[8x150 char]  
averref:'no'  
rt:  
eventdescription:{1x5 cell}  
epochdescription:{}  
specdata:{}  
specicaact:{}  
reject:[1x1 struct]  
stats:[1x1 struct]  
splinofile:{}  
ref:'common'  
history:[7x138 char]  
urevent:[1x154 struct]  
times:[1x384 double]
```

The diagram illustrates the fields of the EEG structure and their descriptions. Arrows point from specific field names to their corresponding descriptions on the right.

- pnts**: Number of data points per trial
- nbchan**: Number of channels
- trials**: Number of trials
- xmin**, **xmax**: Sampling rate
- data**: Time limits
- icawinv**, **icasphere**, **icaweights**: Data
- icaact**, **event**, **epoch**: ICA scalp maps
- chanlocs**: ICA activity
- eventdescription**, **epochdescription**, **specdata**, **specicaact**, **reject**, **stats**: Epoch/event information
- chanlocs**: Channel location

Continuous data

EEG.data = $\begin{bmatrix} 2.1 & 3.8 & 4.9 & 5.1 & 4.8 & 3.9 & \dots \\ -1.3 & -2.4 & -0.5 & -0.3 & 1.4 & 2.5 & \dots \\ 5.2 & 4.7 & 3.3 & 1.2 & 0.7 & 1.3 & \dots \end{bmatrix}$



Data epochs

$$\text{EEG.data} = \begin{bmatrix} 2.1 & 3.8 & 4.9 & 5.1 & 4.8 & 3.9 & \dots \\ -1.3 & -2.4 & -0.5 & -0.3 & 1.4 & 2.5 & \dots \\ 5.2 & 4.7 & 3.3 & 1.2 & 0.7 & 1.3 & \dots \end{bmatrix} \quad \text{Trials 1: EEG.data(:,:,1)}$$

$$\begin{bmatrix} 2.1 & 3.8 & 4.9 & 5.1 & 4.8 & 3.9 & \dots \\ -1.3 & -2.4 & -0.5 & -0.3 & 1.4 & 2.5 & \dots \\ 5.2 & 4.7 & 3.3 & 1.2 & 0.7 & 1.3 & \dots \end{bmatrix} \quad \text{Trials 2: EEG.data(:,:,2)}$$

$$\begin{bmatrix} 2.1 & 3.8 & 4.9 & 5.1 & 4.8 & 3.9 & \dots \\ -1.3 & -2.4 & -0.5 & -0.3 & 1.4 & 2.5 & \dots \\ 5.2 & 4.7 & 3.3 & 1.2 & 0.7 & 1.3 & \dots \end{bmatrix} \quad \text{Trials 3: EEG.data(:,:,3)}$$

eeg_checkset function

```
EEG = eeg_checkset(EEG); % check EEG structure consistency  
EEG = eeg_checkset(EEG, 'eventconsistency'); % check event consistency  
EEG = eeg_checkset(EEG, 'loaddata'); % load data if necessary
```

Equivalent for studies

eeglab

...

```
CURRENTSTUDY = 1; EEG = ALLEEG; CURRENTSET = [1:length(EEG)];  
[STUDY, ALLEEG] = std_checkset(STUDY, ALLEEG);  
eeglab redraw
```

Exercise

1. Load stern_125.set
2. Plot channel 1
3. Plot channel 1 with correct time axis (EEG.times)
4. Extract data epochs from the graphic interface on letters A, B, C...
5. Plot the ERP for all channels
6. Use EEG.times to correct time axis