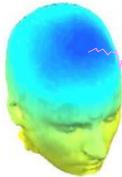


EEGLAB Data Structures



- 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. STUDY
 - .cluster - root ‘studiset’ structure
 - .cluster - component clustering substructure



EEG structure

EEG =

```
setname 'Epoched from "ee114 continuous"'
filename 'ee114sqwesepochs.set'
filepath 'C:\Users\amclint\EEG'
pnts:384
nbchan:32
trials:80
srates:128
xmin:-1
xmax:1.9922
data:[32x384x80 double]
icaWinv:[32x32 double]
icisphere:[32x32 double]
icaweights:[32x32 double]
icaact:[32x384x80 double]
event:[1x157 struct]
epoch:[1x80 struct]
chanlocs:[1x32 struct]
comments:[3x150 char]
avonet:'no'
t[ ]
eventdescription:[1x5 cell]
epochdescription:[]
specdata[]
specicaact[]
reject:[1x1 struct]
stats:[1x1 struct]
splinefile[]
ref:'common'
history:[7x138 char]
unevent:[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

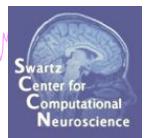
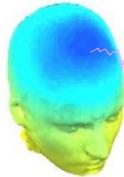
EEG =

```
setname 'Epoched from "ee114 continuous"'
filename 'ee114sqwesepochs.set'
filepath '/home/lmiller/EEG'
pnts:384
nbchan:32
trials:80
srates:128
xmin:-1
xmax:1.9922
data:[32x384x80 double]
icaWinv:[32x32 double]
icisphere:[32x32 double]
icaweights:[32x32 double]
icaact:[32x384x80 double]
event:[1x157 struct]
epoch:[1x80 struct]
chanlocs:[1x32 struct]
comments:[9x150 char]
avonet:'no'
t[ ]
eventdescription:[1x5 cell]
epochdescription:[]
specdata[ ]
specicaact[ ]
reject:[1x1 struct]
stats:[1x1 struct]
splinefile[ ]
ref:'common'
history:[7x138 char]
unevent:[1x154 struct]
times:[1x384 double]
```

The diagram illustrates the fields of an EEG structure and their descriptions. The fields are listed on the left, and their descriptions are on the right, connected by arrows. The fields are:

- pnts:384 → Number of data points per trial
- nbchan:32 → Number of channels
- trials:80 → Number of trials
- xmin:-1, xmax:1.9922 → Sampling rate
- data:[32x384x80 double] → Data
- icaWinv:[32x32 double], icisphere:[32x32 double], icaweights:[32x32 double], icaact:[32x384x80 double] → ICA scalp maps
- event:[1x157 struct], epoch:[1x80 struct] → ICA activity
- chanlocs:[1x32 struct], comments:[9x150 char], avonet:'no' → Epoch/event information
- t[], eventdescription:[1x5 cell], epochdescription:[], specdata[], specicaact[], reject:[1x1 struct], stats:[1x1 struct], splinefile[], ref:'common', history:[7x138 char], unevent:[1x154 struct], times:[1x384 double] → Channel location

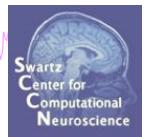
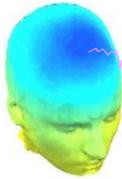
Continuous data



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



Data epochs



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

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

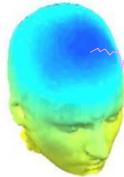
$$\begin{bmatrix} 2.1 & 3.8 & 4.9 & 5.1 & 4.8 & 3.9 & \dots \\ -1.3 & -2.4 & -0.5 & -0.3 & 1.4 & 2.5 & \dots \\ 5.2 & 4.7 & 3.3 & 1.2 & 0.7 & 1.3 & \dots \end{bmatrix} \quad \text{Trials 3: EEG.data(:,:,3)}$$

Plot ERP for your data

```
>> figure; plot(mean(EEG.data,3)');
```

```
>> figure; plot(EEG.times, mean(EEG.data,3)');
```

3 levels of functions



Administrative functions: handle EEG and ALLEEG structures

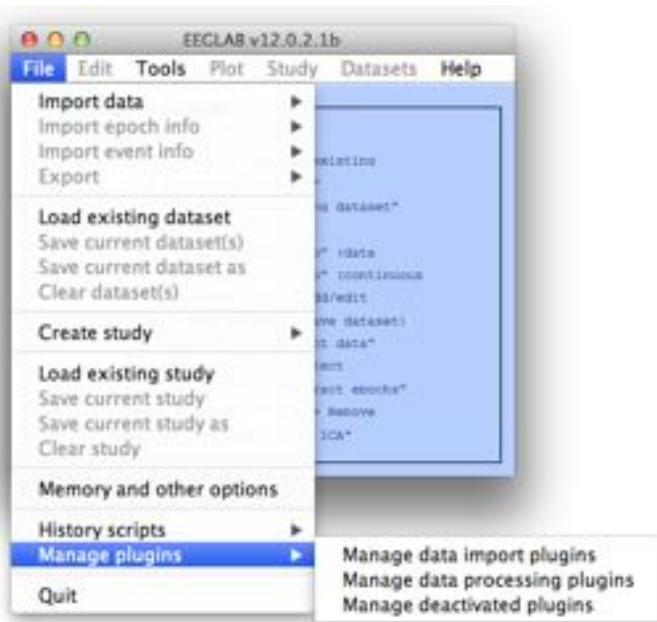
`eeglab()`, `eeg_checkset()`, `pop_delset()`, ...

Pop functions: interactive functions using EEG structure

`pop_erpimage()`, `pop_topoplot()`, `pop_envtopo()`, ...

Signal processing functions: perform signal processing

`erpimage()`, `topoplot()`, `envtopo()`, ...



Pluggings available for install on the internet

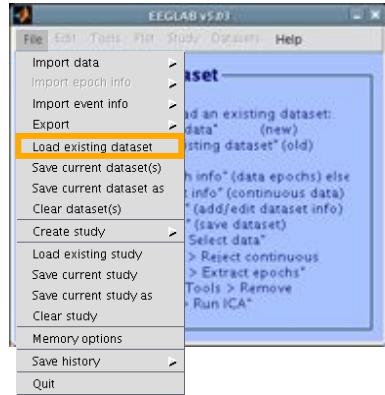
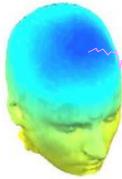
Install	Plugin	Version	Description	Doc
<input type="checkbox"/>	ERPLABmrs	1.00	Interface ERPLAB filters (requires separate ERPLAB install...)	Doc
<input type="checkbox"/>	ADJUST	1.21	Automatic artifact rejection	Doc
<input type="checkbox"/>	BERGEN	1.1	Removal of fMRI-related gradient artifacts from simultaneous...	Doc

Installed pluggings

Update	Deactivate	Plugin	Version	Description	Doc
	<input type="checkbox"/>	brainmovie	0.1	Brainmovies (command line only)	Doc
	<input type="checkbox"/>	commap	2.00	New version 1.03 available. Click update to install.	Doc
	<input type="checkbox"/>	eeg_toolbox	1.0	Interface EEG toolbox functions for ERP peak detection	Doc
	<input type="checkbox"/>	fMRIb	1.21	Remove fMRI artifacts from EEG	Doc
	<input type="checkbox"/>	MP_clustering	1.00	Measure projection clustering of ICA components	Doc
	<input type="checkbox"/>	MutualInfoClustering	1.00	Mutual information clustering	Doc
	<input type="checkbox"/>	StudyEnv topo	0.9	Add env topo capabilities to STUDY	Doc
	<input type="checkbox"/>	VisEd	1.05	New version 1.04 available. Click update to install.	Doc
	<input type="checkbox"/>	infiltr	1.02	Non linear filtering	Doc
	<input type="checkbox"/>	loneta	1.1	New version 1.0 available. Click update to install.	Doc

Cancel Ok

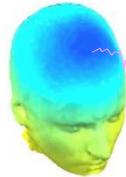
Command line tools



(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



EEGLAB

File **EEGLAB** **Tools** **Plot** **Help**

Import data

Import epoch info

Import event info

Export

Load existing dataset

Save current dataset

Save current dataset as

Clear dataset(s)

Create study

Load existing study

Save current study

Save current study as

Clear study

Memory and other options

Save history

Quit

Memory options - prep_editoptions()

STIMY options (check these checkboxes if you intend to work with studies)

If set, keep at most one dataset in memory. This allows processing hundreds of datasets within studies.

If set, save only one header file for each dataset (header and data). This allows faster data loading in studies.

If set, write ICA activations to disk. This speeds up loading ICA components when dealing with studies.

Memory options

If set, use single precision under Matlab 7.x. This saves RAM but can lead to more numerical imprecisions.

If set, use memory mapped array under Matlab 7.x. This may slow down some computation.

ICA options

If set, precompute ICA activations. This requires more RAM but allows faster plotting of component activation.

If set, scale ICA component activities to RMS (Root Mean Square) in microvolts (recommended).

Folder options

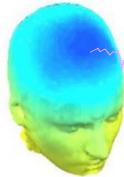
If set, when choosing to open a new dataset assume the folder directory of previous dataset.

Option file: C:\Klaire\file\Concurrent\MATLAB\functions\eegelectro\prep_options.m

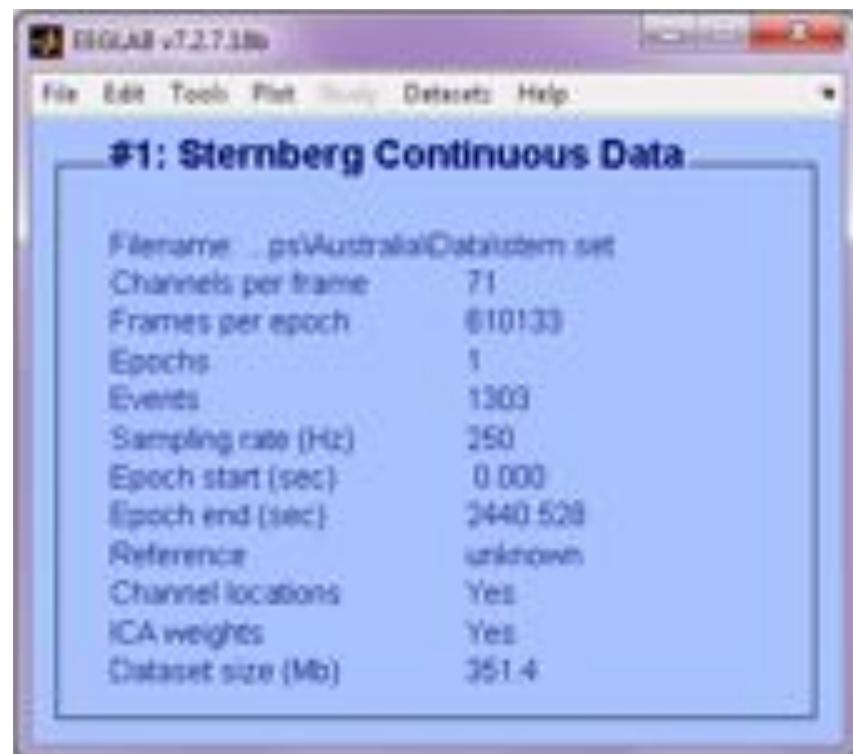
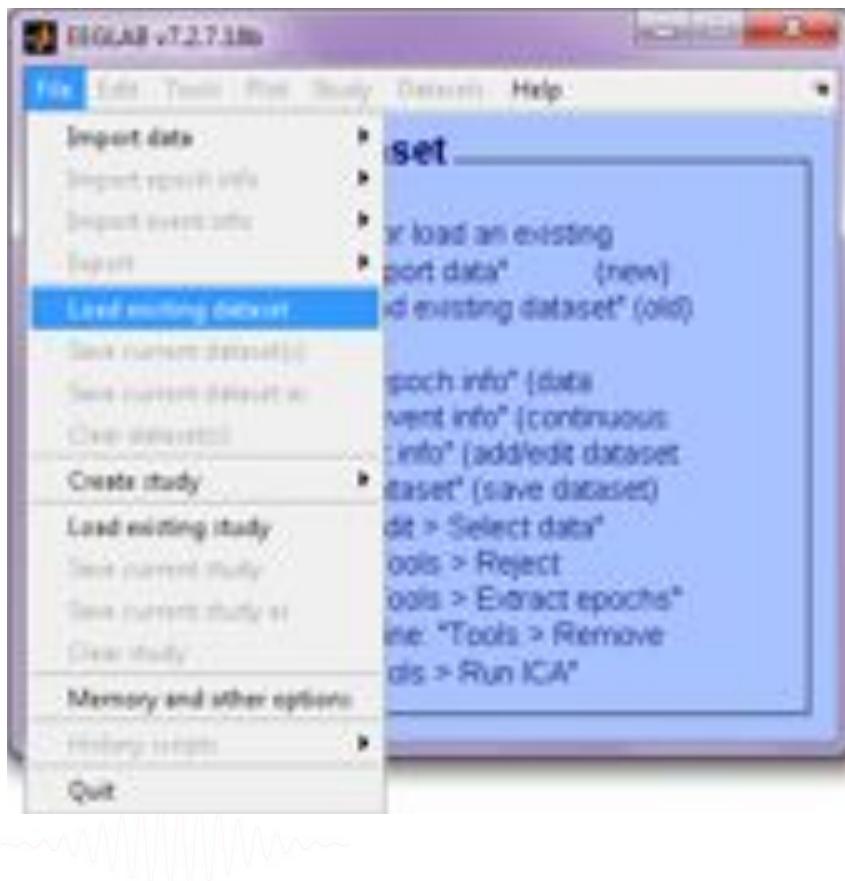
Help **CANCEL** **OK**

**Change memory options
to allow more than one dataset in memory**

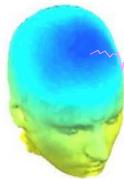
Create a script from ‘eegh’ output



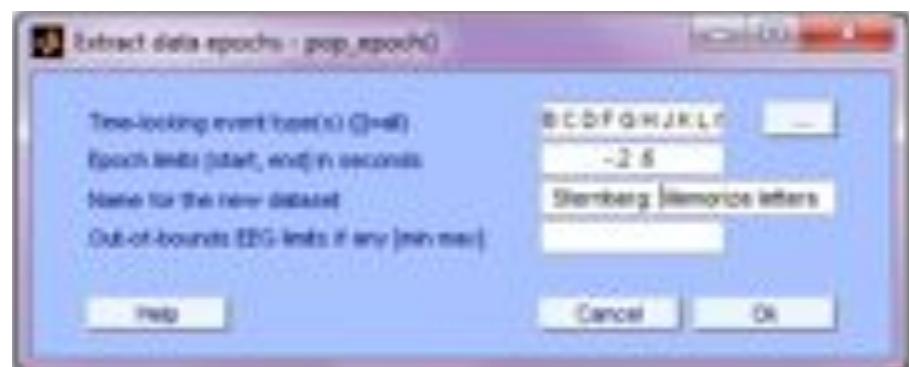
Start by loading a continuous dataset



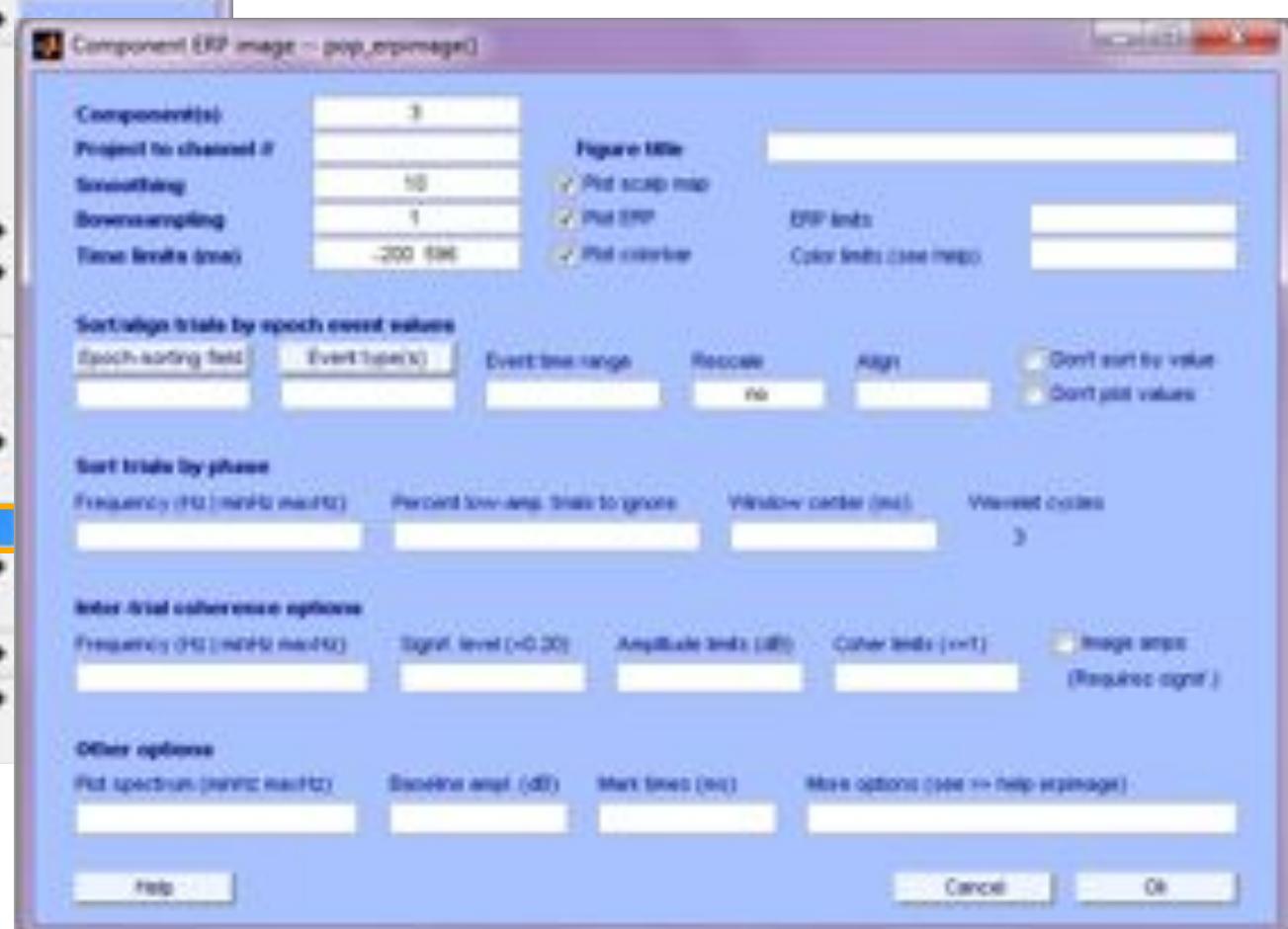
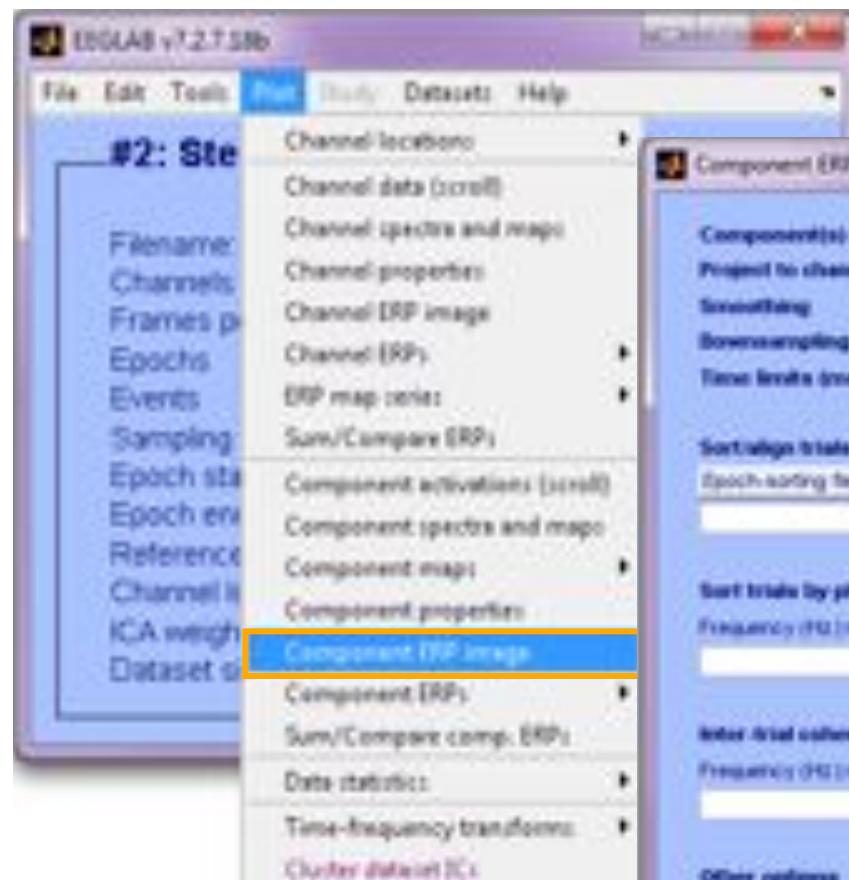
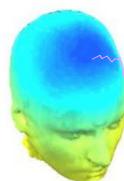
Create a script from ‘eegh’ output



Epoch on Memorize letters

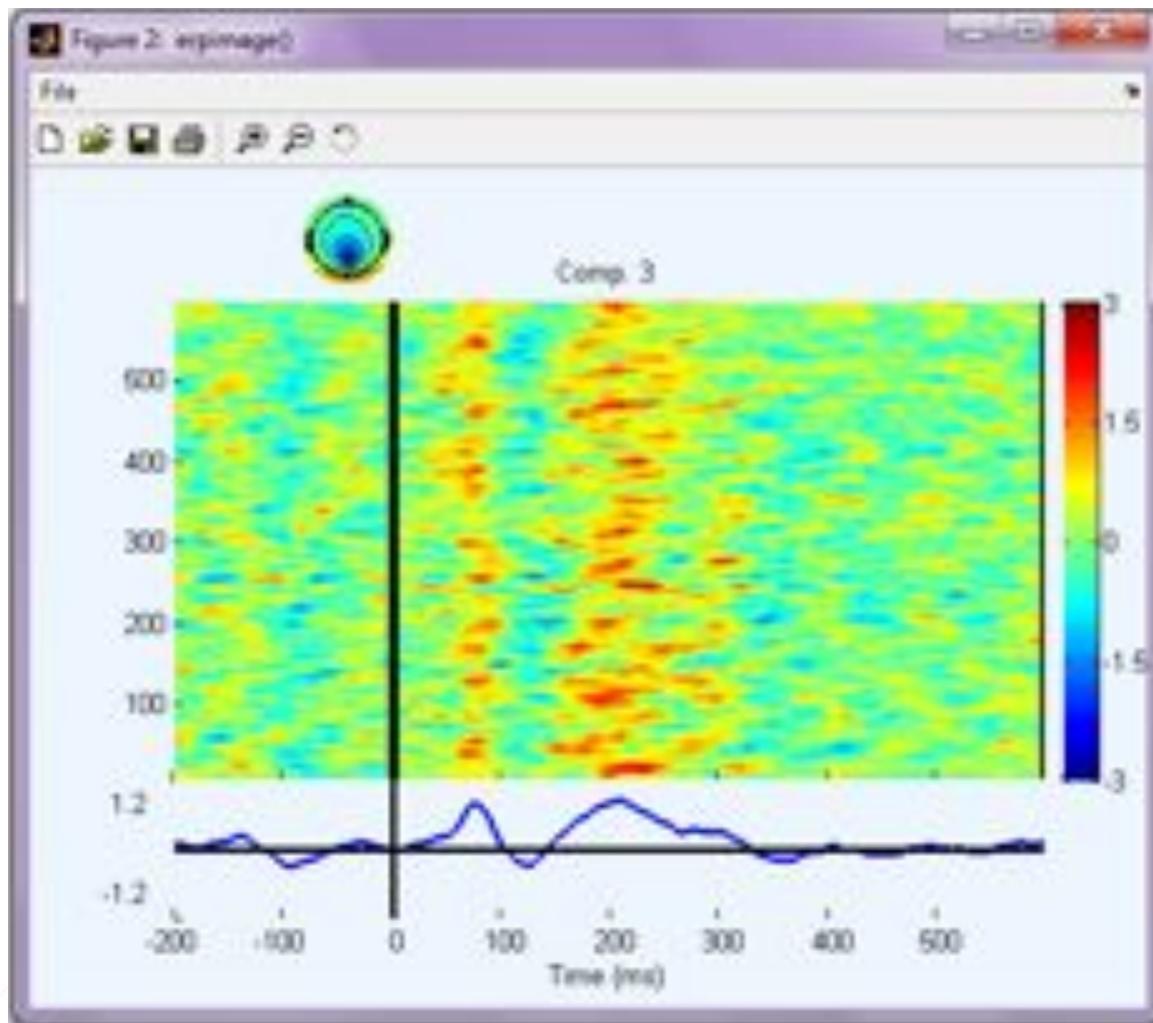
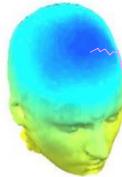


Create a script from ‘eegh’ output

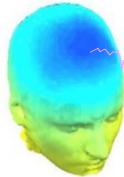


Plot an IC ERP image

Create a script from ‘eegh’ output



Retrieve commands from eegh

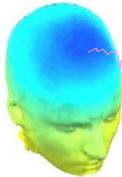


Write a script to do this:

```
>> 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');
```

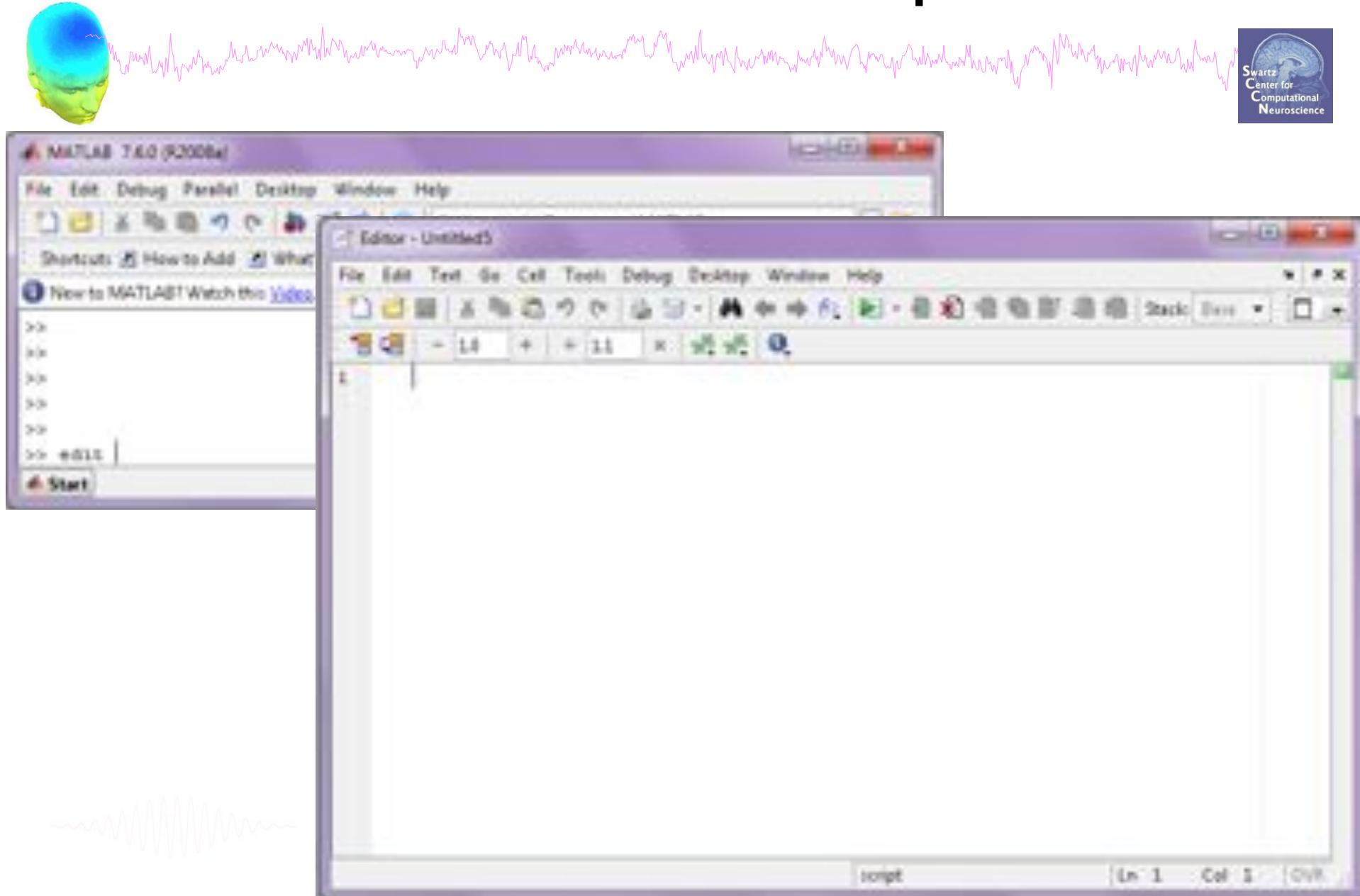
```
[ALLEEG EEG CURRENTSET] = eeg_store(ALLEEG, EEG, 1);
```

```
EEG = pop_rmbase( EEG, [-200 0]);
```

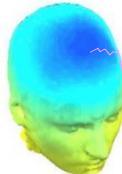
```
[ALLEEG EEG] = eeg_store(ALLEEG, EEG, CURRENTSET);
```

```
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

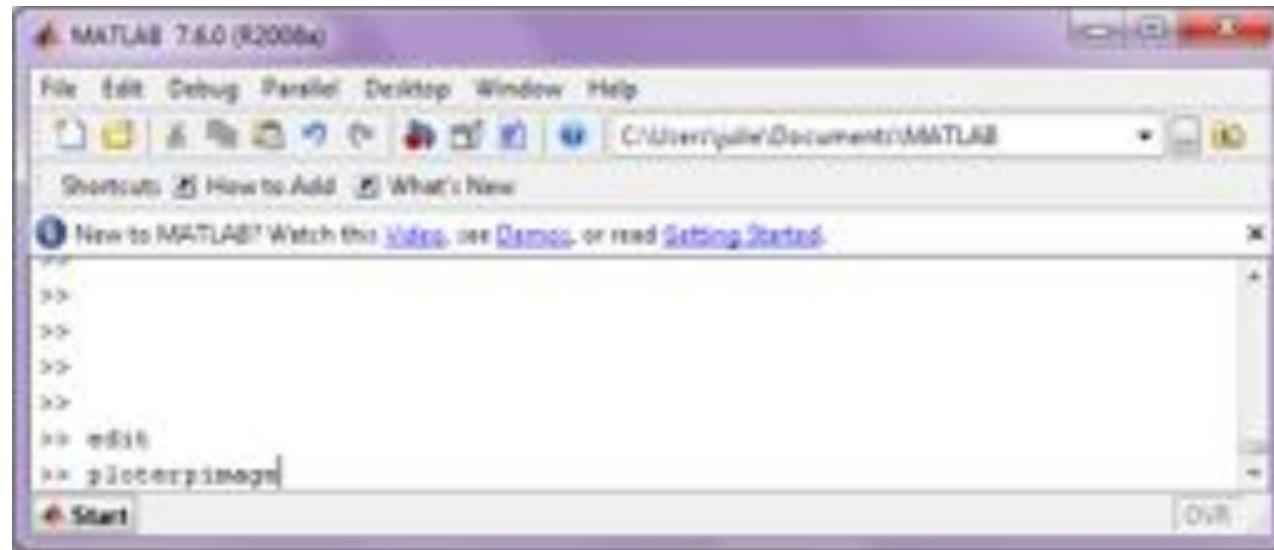
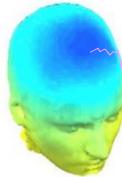


Copy and paste from Matlab window:

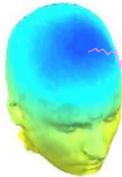
```
Editor - Untitled2>
File Edit Test Go Cell Tools Debug Drifter Window Help
New Open Save All Recent File View Insert Run Cell Window Help
Editor - Untitled2>
1 %ALLEEG EEG CURRENTSET ALLCOM = eeglab;
2 EEG = pop_loadset('filename', 't1023.net', 'filepath', 'C:\Users\lspade\Documents\EEG');
3 [ALLEEG, EEG, CURRENTSET] = eeg_picks(ALLEEG, EEG, 0);
4 EEG = pop_epoch(EEG, 1, 'B1'-'Cz', 'F1'-'G1', 'F3'-'H1', 'F4'-'L1', 'T3'-'R1');
5 [ALLEEG EEG CURRENTSET] = pop_reject(ALLEEG, EEG, 1, 'quiet', 'indef');
6 EEG = pop_endass(EEG, [-200 0]);
7 [ALLEEG EEG] = eeg_stacks(ALLEEG, EEG, CURRENTSET);
8 figure; pop_erpimage(EEG, 0, [3], [], 'Comp', 3, SD, 1, 0, [], 'yrep1.htm', 'fro-
```

Save as 'ploterpimage.m'
In MATLAB folder

Run your new script



Exercise



```
>> eeglab

% load dataset,
% 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
```