Advanced IC analysis

Task 1
Search EEG.event structure

Task 2
Use `newtimef()` to compare conditions

Task 3
Plot a RT-sorted component ERP image
Plot a type-sorted component ERP image
Plot a load-sorted component ERP image

Task 4
Use outputs from commandline ERP image

Exercise...
Advanced IC analysis

Task 1
Search EEG.event structure

Task 2
Use `newtimef()` to compare conditions

Task 3
Plot a RT-sorted component ERP image
Plot a type-sorted component ERP image
Plot a load-sorted component ERP image

Task 4
Use outputs from commandline ERP image

Exercise...
Set memory options

% you will need memory options to keep more than one dataset in memory at once:
pop_editoptions( 'option_storedisk', 0, 'option_savetwofiles', 1,...
    'option_saveica', 0, 'option_single', 1, 'option_memmapdata', 0,...
    'option_computeica', 1, 'option_scaleicarms', 1, 'option_rememberfolder', 1);
%--------------------------------------------------------------------
The example data: Sternberg working memory

File

../Data/stern.set

Data

Continuous data (not epoched), ref’d to right mastoid

Task

3-7 letters to memorize, among 1-5 letters to ignore
50% chance of probe letter being ‘in-set’

Fixation

(5 sec)  

Memorize  +  M  L  T  G  P  Y  Q  W  -  T

Ignore

SOA

(1.4 sec)

Was this letter in the memorized set?

Maintenance

(2-4 sec)  

Probe

RT

RESPONSE
%% Color-coding for tutorial scripts:

% Green text is comments

myvariable (bold, red) = pre-defined variable

for ep ... end = ... (bold, blue) = ‘for loop’ variable

if ... end = (bold, cyan) = ‘if loop’ statement

newtimef() (bold, purple) = function call

[outdata, outvar, outtrials, ...] (brown, in brackets) = function output variables
% OBJECTIVES:
% 1) Find all Memorize letters that were preceded by an ignore letter
% 2) Find all Memorize letters that were preceded by a memorize letter

% hint: ‘memorize’ event codes are single letters

epochidxM = []; % Mem preceded by a mem letter
epochidxG = []; % Mem preceded by an ignore letter

for ev = 2:length(EEG.event)
    if length(EEG.event(ev).type)==1 & length(EEG.event(ev-1).type)==1
        epochidxM = [epochidxM, ev]; % save this event
    elseif length(EEG.event(ev).type)==1 & EEG.event(ev-1).type(1)=='g'
        epochidxG = [epochidxG, ev]; % save this event
    end;
end;
Epoch on selected events

% Epoch continuous data around selected events
%-------------------------------------------------------------------

[EEG, indices] = pop_epoch( EEG, [], [-2 2], 'eventindices', epochidxG);

[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1, ...
    'setname','Mem after Ignore letter','gui', 'off');

EEG = pop_autorej(EEG, 'nogui', 'on'); % Auto-reject noisy epochs

[ALLEEG EEG CURRENTSET]=pop_newset(ALLEEG,EEG,CURRENTSET,'retrieve',1);

[EEG, indices] = pop_epoch( EEG, [], [-2 2], 'eventindices', epochidxM);

[ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 1, ...
    'overwrite','on', 'setname','Mem after Mem letter','gui', 'off');

EEG = pop_autorej(EEG, 'nogui', 'on'); % Auto-reject noisy epochs

eeglab redraw
Confirm datasets contain expected epochs

```
>> [ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, CURRENTSET, 'retrieve',1);
>> EEG.epoch(2)  %--- Select several random epochs
ans =

    event: [4 5 6]
    eventlatency: {[-1.4400e+003]  [0]  [1.4440e+003]}
    eventload: {[1]  [2]  [3]}
    eventtype: {'R'  'N'  'Z'}
    eventurevent: {[5]  [6]  [7]}
>> [ALLEEG EEG CURRENTSET] = pop_newset(ALLEEG, EEG, 2, 'retrieve',2);
>> EEG.epoch(2)
ans =

    event: [4 5 6]
    eventlatency: {[-1.4400e+003]  [0]  [1.4440e+003]}
    eventload: {[0]  [0]  [1]}
    eventtype: {'gC'  'Z'  'L'}
    eventurevent: {[15]  [16]  [17]}
```
Advanced IC analysis

Task 1
Search EEG.event structure

Task 2
Use \texttt{newtimef()} to compare conditions

Task 3
Plot a RT-sorted component ERP image
Plot a type-sorted component ERP image
Plot a load-sorted component ERP image

Task 4
Use outputs from commandline ERP image

Exercise...
Get `newtimef()` command from GUI call
Use `newtimef()` to compare conditions

```matlab
>> eegh

figure; pop_newtimef( EEG, 0, 4, [-2000 1996], [3 0.5] ,
    'topovec', EEG.icawinv(:,4), 'elocs', EEG.chanlocs,
    'chaninfo', EEG.chaninfo, 'baseline', [-200 0], 'alpha', .01,
    'freqs', [3 50], 'plotphase', 'off', 'padratio', 1);

>> help newtimef

Example using data from two conditions (EEG versus ALLEEG(2)):

```matlab
>> [ersp, itc, powbase, times, freqs, erspboot, itcboot] = ...
    newtimef({EEG.data(chan,:,:) ALLEEG(2).data(chan,:,:)}, ...
    EEG.pnts, [EEG.xmin EEGxmax]*1000, EEG.srate, cycles);
```
Task 3: Use `newtimef()` to compare conditions

% adapt to your script:-----------------------------

% data from datasets 1 (mem after mem)
% and 2 (mem after ignore)

ic = 4; % choose a component

[ersp, itc, powbase, times, freqs, erspboot, itcboot] = ... data
newtimef({ALLEEG(1).icaact(ic,:), ALLEEG(2).icaact(ic,:)}, ... EEG.pnts, [EEG.xmin EEG.xmax]*1000, EEG.srate, [3 .5],...
'type', 'phasecoher', 'topovec', EEG.icawinv(:,ic), ...
'elocs', EEG.chanlocs, 'chaninfo', EEG.chaninfo, ...
'title', {[\'IC ', int2str(ic),\' M a M\'],...
condition 1
'title', {[\'IC ', int2str(ic),\' M a Ig\'],...
condition 2
'baseline', [-200 0], 'alpha', .01, 'padratio', 1,...
'plotphase', 'off', 'freqs', [3 50]);
Compare conditions with \textit{newtimef()}.

Higher frontal theta during the second consecutive Memorize letter.
Compare conditions with `newtimef()`

Less parietal alpha power during the second consecutive Memorize letter.
Compare conditions with `newtimef()`.

Less occipital alpha power during the second consecutive Memorize letter.
Compare conditions with `newtimef()`

Less visual evoked potential to the second consecutive Memorize letter
Advanced IC analysis

Task 1
Search EEG.event structure

Task 2
Use `newtimef()` to compare conditions

Task 3
Plot a RT-sorted component ERP image
Plot a type-sorted component ERP image
Plot a load-sorted component ERP image

Task 4
Use outputs from commandline ERP image

Exercise...
Sort ERP image by RT

Component ERP image -- pop_erpimage()

- Project to channel #: 17
- Smoothing: 5
- Downsampling: 1
- Time limits (ms): -3000 3000
- Figure title
- ERP limits
- Color limits (see Help)
- Sort/align trials by epoch event values
  - Epoch-sorting field: latency, in'out
  - Event type(s):
  - Event time range
  - Rescale: no
  - Align
  - Don't sort by value
  - Don't plot values
- Sort trials by phase
  - Frequency (Hz | minHz maxHz)
  - Percent low-amp. trials to ignore
  - Window center (ms)
  - Wavelet cycles: 3
- Inter-trial coherence options
  - Frequency (Hz | minHz maxHz)
  - Signif. level (<.020)
  - Amplitude limits (dB)
  - Coher limits (<=1)
  - Image amps
  (Requires signif.)
- Other options
  - Plot spectrum (minHz maxHz)
  - Baseline ampl. (dB)
  - Mark times (ms)
  - More options (see >> help erpimage)

EEGLAB Workshop X, June 14-17, 2010, Jyväskylä, Finland: Julie Onton – Advanced IC analysis
Sort ERP image by RT
Sort ERP image by response type

Labels were added for clarity (not plotted by ERP image)
Sort ERP image by response type
Sort ERP image by response type (2\textsuperscript{nd} example)
Sort ERP image by memory load

Figure 2: erpimage()

Component ERP image -- pop_erpimage()

- Component(s): 4
- Project to channel #: blank
- Smoothing: 5
- Downsampling: 1
- Time limits (ms): -3000 500
- Figure title: blank
- Epoch-sorting field: load
- Event type(s): r'B' 'C' 'D' 'F' 'G' 'r'
- Event time range: blank
- Rescale: no
- Sort/align trials by epoch event values: blank
- Frequency (Hz): blank
- Percent low-amp. trials to ignore: blank
- Window: blank
- Inter-trial coherence options:
  - Frequency (Hz): 8 12
  - Signif. level (<0.01): 0.01
  - Amplitude limits (dE): blank
  - Cohan limits (<1): blank
- Image amps (Requires signif.): blank
- Other options:
  - Plot spectrum (minHz maxHz): blank
  - Baseline ampl. (dB): blank
  - Mark times (ms): blank
  - More options (see >> help erpimage): blank

Comp. 4

Load 7
Load 5
Load 3

ERSP

ITC

-1.693 dB
9.644 Hz
Sort ERP image by memory load

Component ERP image -- pop_erpmage()

- Component(s): 17
- Project to channel #: 5
- Smoothing: 1
- Downsampling: -3000 to 2500

Sort/align trials by epoch event values:
- Epoch-sorting field: load
- Event type(s): tB'tY'c'Y'd'tF'ty'G't'y'
- Event time range: no

Sort trials by phase:
- Frequency (Hz): 8 to 12
- Percent low-amp. trials to ignore: 0.01
- Window center: 0 Hz

Inter-trial coherence options:
- Frequency (Hz): 8 to 12
- Signif. level (>=0.20): 0.01
- Amplitude limits (dB): 0.01
- Coher limits (<=1): 0.01
- Image amps: (Requires signif.)

Other options:
- Plot spectrum (mHz): 0.01
- Baseline ampl. (dB): 0.01
- Mark times (ms): 0.01
- More options (see >> help erpmage)

Figure 4

Comp. 17

- Load 7
- Load 5
- Load 3

ITC

- 11.11 Hz

ERSP

- -3.408 dB

Time (ms)

-3000 to 2500
Advanced IC analysis

Task 1
Search EEG.event structure

Task 2
Use `newtimef()` to compare conditions

Task 3
Plot a RT-sorted component ERP image
Plot a type-sorted component ERP image
Plot a load-sorted component ERP image

Task 4
Use outputs from commandline ERP image

Exercise...
ERP image from the commandline

Command executed by `pop_erpimage`:

```matlab
erpimage(EEG.icaact([8], :) , ones(1, EEG.trials)*EEG.xmax*1000, linspace(EEG.xmin*1000, EEG.xmax*1000, EEG.pnts), 'Comp. 8', 10, 1, 'yerplabel', '', 'tocpo', [ EEG.icainv(:,8) EEG.chanlocs EEG.chaninfo ], 'erp', 'cbar');
```

Plotting input data as 359 epochs of 750 frames sampled at 250.0 Hz.

Sorting data on input sortvar.

Smoothing the sorted epochs with a 10-epoch moving window.

and a decimation factor of 1

Output data will be 750 frames by 349 smoothed trials.

Ourtials: 6.00 to 354.00

The axis range will be the sym. abs. data range -> [-1.35755,1.35755].

Data will be plotted between -1000 and 1996 ms.

Overplotting sorted sortvar on data.

Plotting the ERP trace below the ERP image

Plotting a topo map in upper left.

Done.
Task 1: Retrieve \textit{erpimage()} call

Command executed by \texttt{pop_erpimage}:

\begin{verbatim}
\texttt{erpimage( EEG.icaact([8], :), ones(1, EEG.trials)*EEG.xmax*1000, ...}
\end{verbatim}

\begin{verbatim}
\hspace{1cm} EEG.times \hspace{1cm} title \hspace{1cm} smooth/decimate
\end{verbatim}

\begin{verbatim}
\hspace{1cm} linspace(EEG.xmin*1000, EEG.xmax*1000, EEG.pnts),'Comp. 8', 10, 1, ...
\end{verbatim}

\begin{verbatim}
\hspace{1cm} 'yerplabel','','topo', { EEG.icawinv(:,8) EEG.chanlocs EEG.chaninfo },...
\end{verbatim}

\begin{verbatim}
\hspace{1cm} 'erp','cbar');
\end{verbatim}

Plotting options/scalp map

\begin{verbatim}
Plot ERP and colorbar
\end{verbatim}

To adapt this command to include more \textit{erpimage()} options:

\begin{verbatim}
>> help erpimage
\end{verbatim}
Use ‘help’ command to build script

>> help erpimage

```
help erpimage

erpimage() - Image a collection of single-trial data epochs, optionally sorted on
and/or aligned to an input sorting variable and smoothed across trials
with a moving-average. (To return event-aligned data without plotting,
use eventlock()). Optionally sort trials on value, amplitude or phase
within a specified latency window. Optionally plot the ERP mean and
std. dev. and moving-window spectral amplitude and inter-trial coherence
at a selected or peak frequency. Click on individual figures parts to
examine them separately and zoom (using axcopy()).

Usage:

>> [outdata, outvar, outtrials, limits, axhndls, erp, ...]
   amps, cohers, cohsig, ampsig, outamps, phsangls, phsamp, sortidx, erpsig] ...
   = erpimage(data, sortvar, times, 'title', avewidth, decimate, ...
               flag1, arg1, flag2, arg2, ...);

Necessary inputs:

data - [vector or matrix] Single-channel input data to image.
       Formats (1,frames*trials) or (frames,trials)

Optional ordered inputs (with defaults):

sortvar - [vector | []] Variable to sort epochs on (length(sortvar) = nepochs)
           Example: sortvar may by subject response time in each epoch (in ms)
           (default []) plot in input order

times - [vector | []] of latencies (ms) (length(times) = frames)
        else [starts times srate] Give start latency (ms), time points
        (i.e. frames) per epoch, sampling rate (Hz), (default []) 0:nframes-1
		'title' - [string] Plot title (default: none)

'avewidth' - Number of trials to smooth with a moving-average (may be non-integer)
             (default: 1)

decimate - Factor to decimate ntrials out by (may be non-integer) (default: 0:1)
           If this is large (> sqrt(num. trials)), output this many trials.

Unordered options ('keyword',argument pairs):
```

EEGLAB Workshop X, June 14-17, 2010, Jyväskylä, Finland: Julie Onton – Advanced IC analysis
ERP image sorted by activation value

[valsort] - [startms endms direction] Sort data by (mean) activation value between startms and endms. Direction is 1 or -1. If -1, plot max-value epoch at bottom

%% VARIABLES
comp1 = 8; % Comp number to plot
data = squeeze(EEG.icaact(comp1,:,:));

[sortvar] = []; % no sorting
startms = 580; % ms
dat
endms = 620; % ms
smoothby = 1;

%% PLOT ERPIMAGE
figure;
[outdata,outvar,outtrials,limits,axhndls,....
erp,amps,coher,cohsig,ampsig,outamps,....
phsangls,phsamp,sortidx,erpsig] = ...

[sortidx] = erpimage(data, sortvar, EEG.times,...

avnwidth, decimate, flag1, arg1,...);
Matlab index definition

>> my_numbers = [101,102,103,104,105,106,107,108,109,110];
my_numbers =

101 102 103 104 105 106 107 108 109 110

>> new_order = [8,2,5,1,10,9,4,6,3,7];  % analogous to sortidx
>> my_numbers(new_order)

ans =

108 102 105 101 110 109 104 106 103 107

...
Use sort index to sort a new ERP image

%%% VARIABLES
comp1 = 8;
data = squeeze(EEG.icaact(comp1,:,:));

sortvar = [];
startms = 580;
endms = 620;
smoothby = 1;

%%% 1st ERPIMAGE

figure;
[outdata,outvar, outtrials,limits, axhndls, erp, ...
amps, coher, cohsig, ampsig, outamps, phsangls, ...
phsamps,sortidx, erpsig] = erpimage(data,sortvar,...
EEG.times,'',smoothby, 1,'valsort',[startms endms]);

%%% 2nd ERPIMAGE

%%% Sort by previous sort order
comp2 = 16;
data2 = squeeze(EEG.icaact(comp2,:,
sortidx));
minfrq = 9;  % specify freq range for
maxfrq = 12;  % amplitude plot
smoothby = 20;
figure;
[outdata,outvar, outtrials,limits, axhndls, erp, amps,...
coher, cohsig, ampsig, outamps, phsangls,phsamps,sortidx2, erpsig] ...
= erpimage( data2, sortvar, EEG.times, ['Component ',int2str(comp2)],...
smoothby, 1, 'coher', [minfrq maxfrq .01], 'plotamps');

Objective: Use sort order (sortidx) from 'valsort' of comp1 to create a new ERP image of another component with the same sort order.
Phase-sort applied to second IC

No sort

Phase-sort -75 ms center

Sorted by IC 5 phase-sort
Exercise

• Intermediate:
  • Use EEG.event structure to select specific trial types.
    - Epoch on these trials and plot ERSP and/or ERP images
  • Plot a two-condition ERSP of a chosen IC (start with loading continuous data, epothing, etc)
  • Start with GUI call and adapt a script using ‘eegh’
  • Collect RTs and include in ERP image plots

• Advanced:
  • From a ‘valsort’ ERP image, collect sortidx output
  • Apply sort order to an ERP image of another component (try different smoothing factors)

** Example scripts for exercises can be found in
…/workshop/Scripts/Tutorial_6_ICanalysis.m