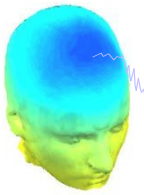


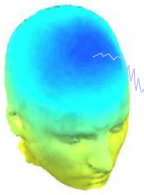
Evaluating ICA Components

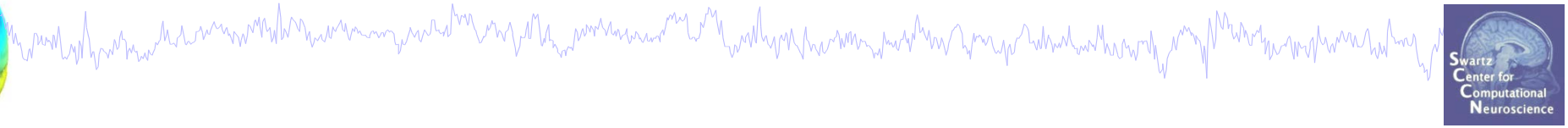
EEGLAB Workshop XXV

JAIST, Tokyo, Japan

Day 1

Independent Component Analysis



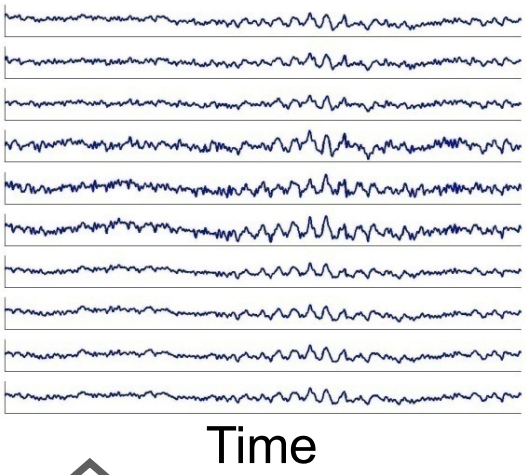


Swartz Center for Computational Neuroscience

x = scalp EEG

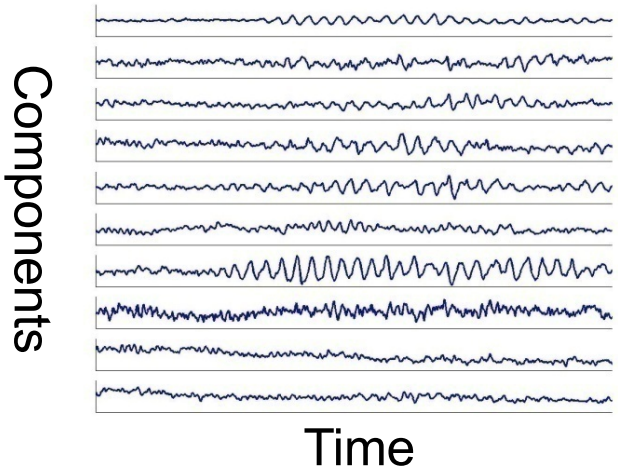
W = unmixing matrix

u = sources



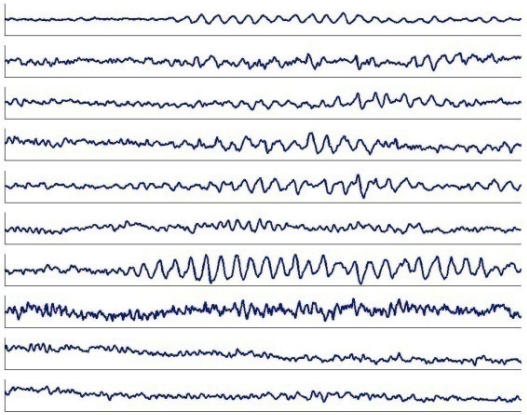
W*x = u

ICA



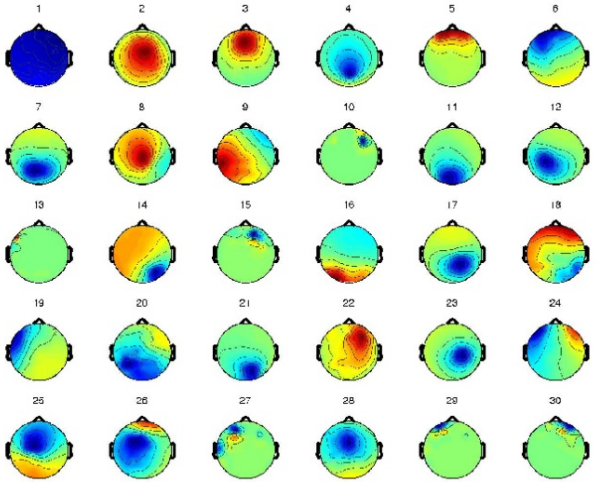
x = W^{-1}*u

u = sources

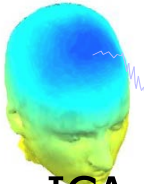


*

W^{-1} (scalp projections)

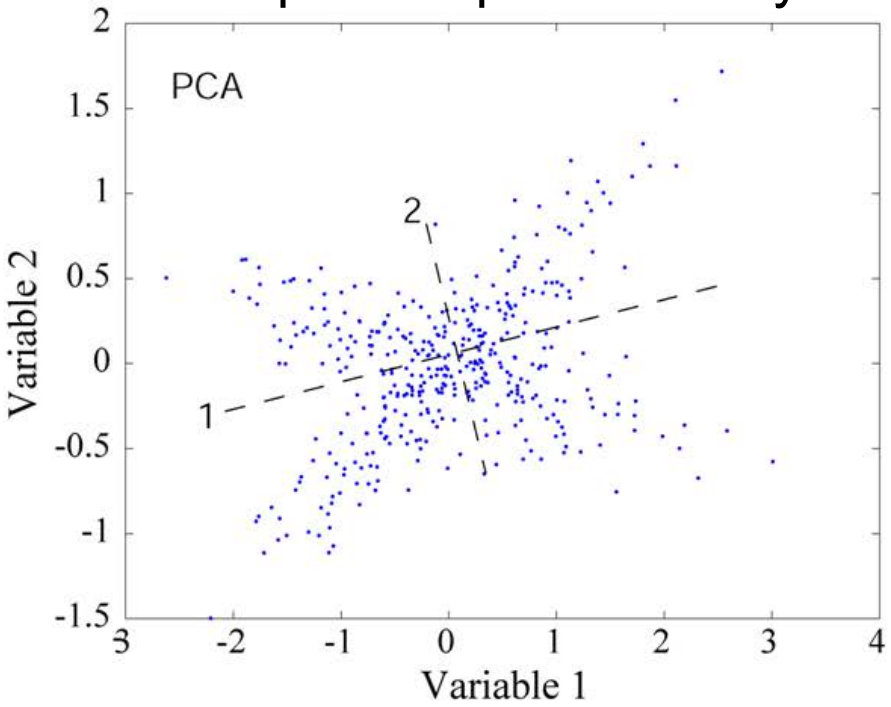


ICA and PCA

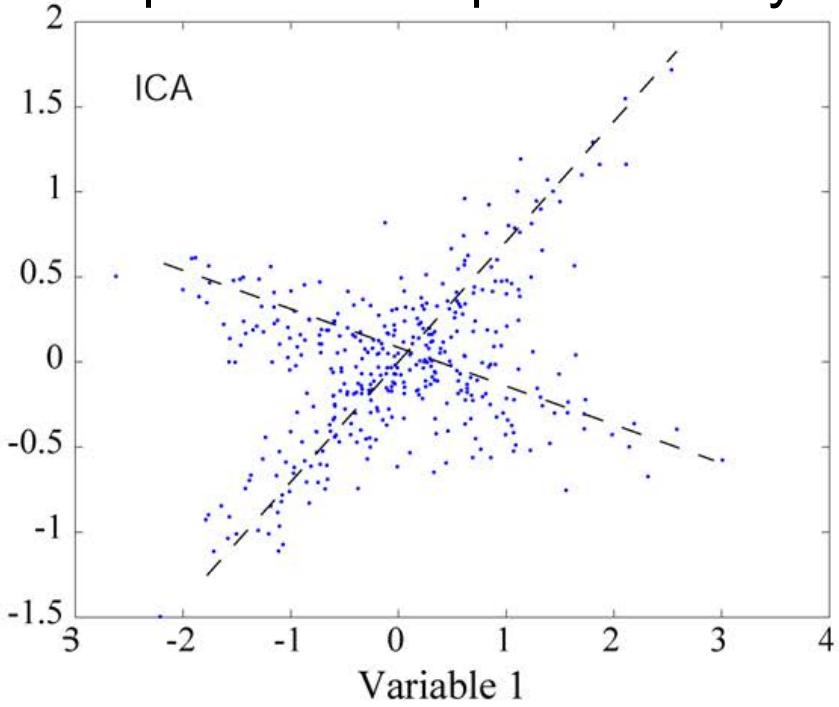


ICA is a method to recover a version of the original sources by multiplying the data by an unmixing matrix,

Principal component analysis

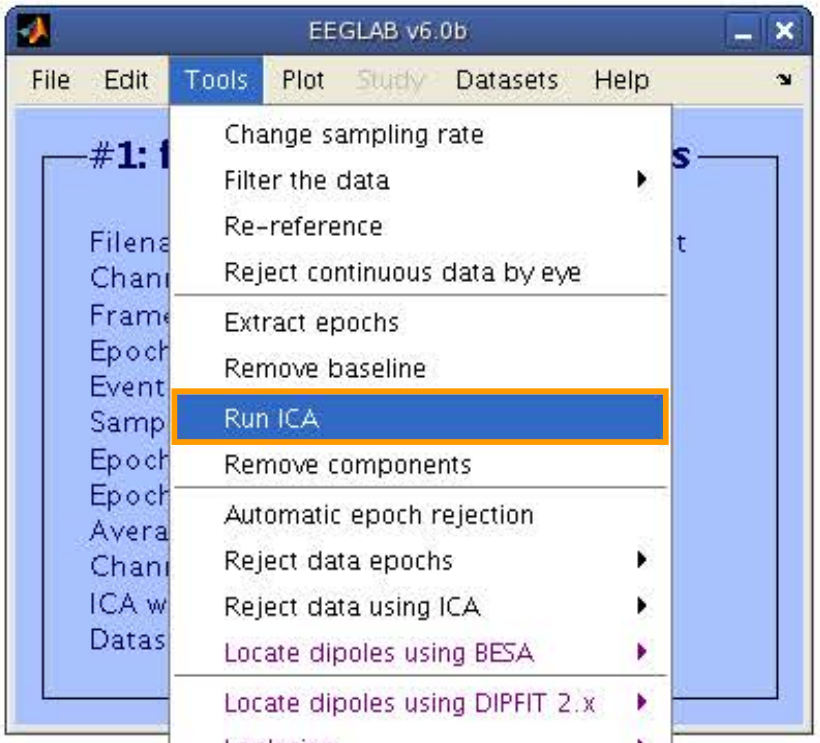


Independent component analysis



While PCA simply decorrelates the outputs (using an orthogonal mixing matrix), ICA attempts to make the outputs **statistically independent**, while placing no constraints on the mixing matrix.

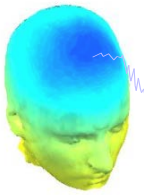
Finally: ICA options



Option	Default	Comments
'extended'	0	1 is recommended to find sub-gaussians
'stop'	1e-7	final weight change → stop
'lrate'	determined from data	too small → too long… too large → wts blow up
'maxsteps'	512	more channels → more steps
'pca'	0 or EEG.nbchan	Decompose only a principal data subspace

Other algorithms:
binica, amica, cudaica, beamica

Runica progress…



Swartz Center for Computational Neuroscience

Press Button to interupt runica

Interupt

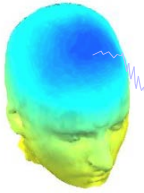
csh

```
Input data size [33,133175] = 33 channels, 133175 frames/nFinding 33 ICA components using extended ICA.
Kurtosis will be calculated initially every 1 blocks using 6000 data points.
Decomposing 122 frames per ICA weight ((1089)^2 = 133175 weights, Initial learning rate will be 0.001, block size
Learning rate will be multiplied by 0.98 whenever angledelta >= 60 deg.
More than 32 channels: default stopping weight change 1E-7
Training will end when wchange < 1e-07 or after 512 steps.
Online bias adjustment will be used.
Removing mean of each channel ...
Final training data range: -171.806 to 179.094
Computing the sphering matrix...
Starting weights are the identity matrix ...
Sphering the data ...
Beginning ICA training ... first training step may be slow ...
step 1 - lrate 0.001000, wchange 16.85061324, angledelta  0.0 deg
step 2 - lrate 0.001000, wchange 0.26760405, angledelta  0.0 deg
step 3 - lrate 0.001000, wchange 0.79058323, angledelta 104.0 deg
step 4 - lrate 0.000980, wchange 0.66700031, angledelta 147.2 deg
step 5 - lrate 0.000960, wchange 0.62849071, angledelta 146.5 deg
step 6 - lrate 0.000941, wchange 0.73967955, angledelta 150.7 deg
step 7 - lrate 0.000922, wchange 0.73727229, angledelta 151.6 deg
step 8 - lrate 0.000904, wchange 0.74051387, angledelta 137.9 deg
step 9 - lrate 0.000886, wchange 0.74536137, angledelta 156.0 deg
step 10 - lrate 0.000868, wchange 0.72101402, angledelta 143.7 deg
step 11 - lrate 0.000851, wchange 0.14690114, angledelta 102.5 deg
step 12 - lrate 0.000834, wchange 0.11822100, angledelta 114.3 deg
step 13 - lrate 0.000817, wchange 0.75552966, angledelta 100.6 deg
step 14 - lrate 0.000801, wchange 0.26739750, angledelta 109.1 deg
step 15 - lrate 0.000785, wchange 0.12123251, angledelta 94.2 deg
step 16 - lrate 0.000769, wchange 0.10285606, angledelta 110.7 deg
step 17 - lrate 0.000754, wchange 0.09770499, angledelta 118.6 deg
step 18 - lrate 0.000739, wchange 0.09544428, angledelta 117.1 deg
```

csh

```
step 241 - lrate 0.000002, wchange 0.00000082, angledelta 101.5 deg
step 242 - lrate 0.000001, wchange 0.00000061, angledelta 96.1 deg
step 243 - lrate 0.000001, wchange 0.00000057, angledelta 97.5 deg
step 244 - lrate 0.000001, wchange 0.00000054, angledelta 93.7 deg
step 245 - lrate 0.000001, wchange 0.00000055, angledelta 100.3 deg
step 246 - lrate 0.000001, wchange 0.00000047, angledelta 96.9 deg
step 247 - lrate 0.000001, wchange 0.00000046, angledelta 91.3 deg
step 248 - lrate 0.000001, wchange 0.00000045, angledelta 101.5 deg
step 249 - lrate 0.000001, wchange 0.00000041, angledelta 103.1 deg
step 250 - lrate 0.000001, wchange 0.00000036, angledelta 95.5 deg
step 251 - lrate 0.000001, wchange 0.00000033, angledelta 92.1 deg
step 252 - lrate 0.000001, wchange 0.00000029, angledelta 97.4 deg
step 253 - lrate 0.000001, wchange 0.00000030, angledelta 95.8 deg
step 254 - lrate 0.000001, wchange 0.00000023, angledelta 94.2 deg
step 255 - lrate 0.000001, wchange 0.00000023, angledelta 97.6 deg
step 256 - lrate 0.000001, wchange 0.00000023, angledelta 97.1 deg
step 257 - lrate 0.000001, wchange 0.00000021, angledelta 92.0 deg
step 258 - lrate 0.000001, wchange 0.00000020, angledelta 99.1 deg
step 259 - lrate 0.000001, wchange 0.00000019, angledelta 95.0 deg
step 260 - lrate 0.000001, wchange 0.00000015, angledelta 98.3 deg
step 261 - lrate 0.000001, wchange 0.00000014, angledelta 99.0 deg
step 262 - lrate 0.000001, wchange 0.00000014, angledelta 94.3 deg
step 263 - lrate 0.000001, wchange 0.00000013, angledelta 95.4 deg
step 264 - lrate 0.000001, wchange 0.00000012, angledelta 94.1 deg
step 265 - lrate 0.000001, wchange 0.00000011, angledelta 96.1 deg
step 266 - lrate 0.000001, wchange 0.00000010, angledelta 94.8 deg
step 267 - lrate 0.000001, wchange 0.00000010, angledelta 94.5 deg
step 268 - lrate 0.000001, wchange 0.00000010, angledelta 97.7 deg
step 269 - lrate 0.000001, wchange 0.00000008, angledelta 95.1 deg
Sorting components in descending order of mean projected variance ...
Permuting the activation wave forms ...
>>
>>
```

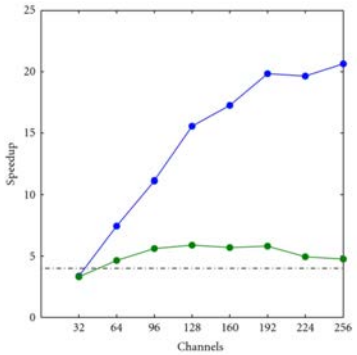
Alternatives to runica



Infomax ICA

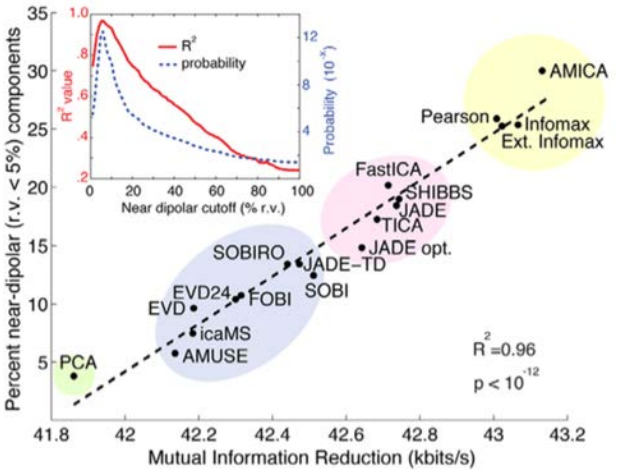
- `runica` matlab implementation
- `binica` compiled version; fast
- `cudaica` GPU version

Raimondo, et al, 2012, `https://liaa.dc.uba.ar/node/13`

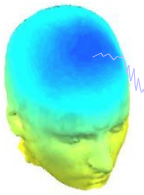


AMICA

- Best at extracting dipolar ICs
- Multiple-model support



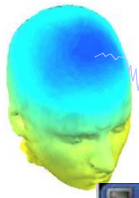
Review: ICA in Plain English

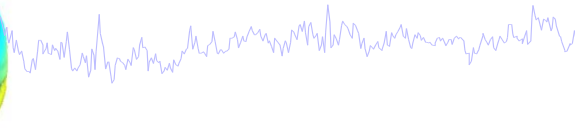


Source activation = unmixing * Channel data

Channel data = mixing (topo) * Source activation

Results of ICA Decomposition in EEG struct

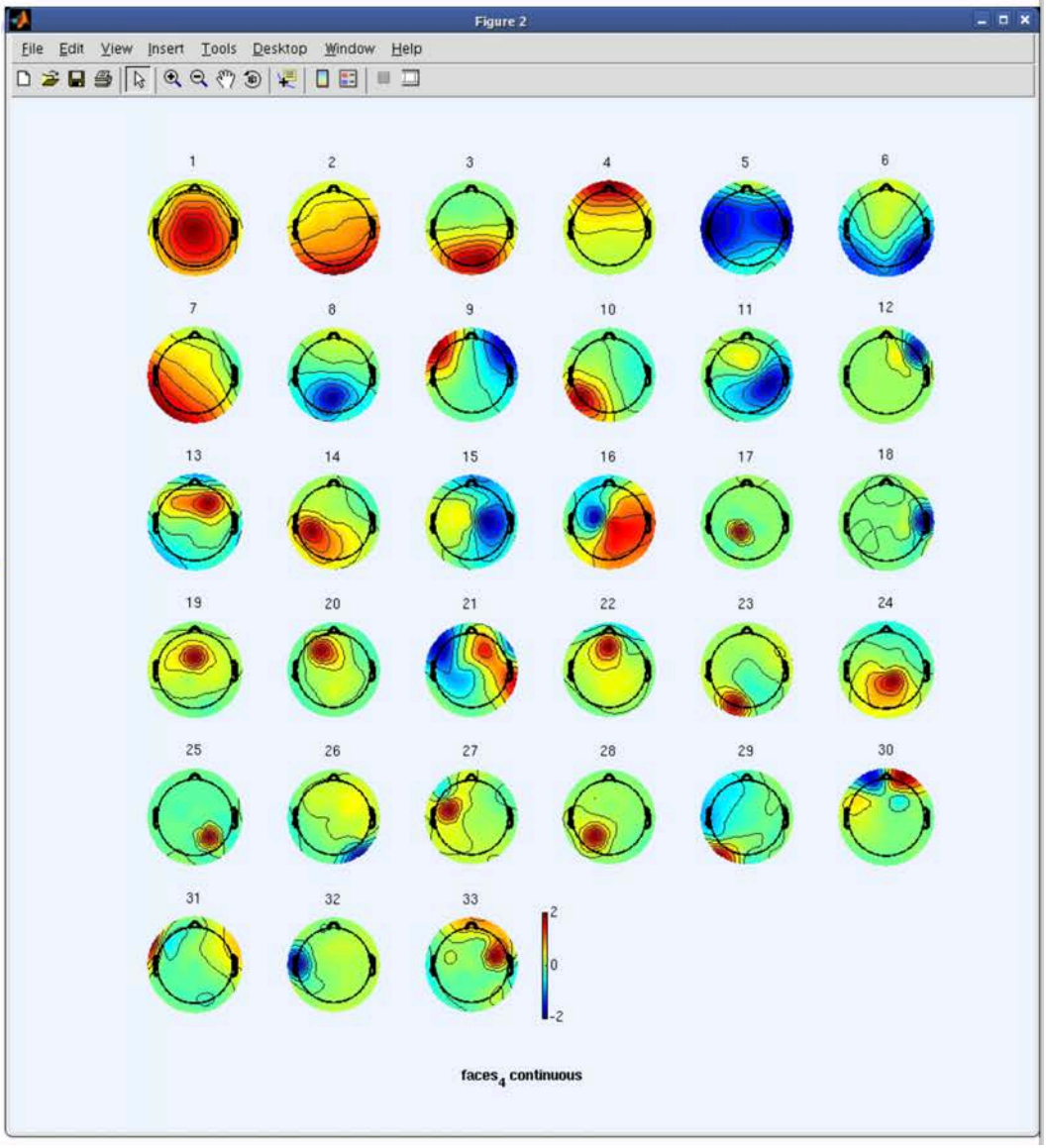




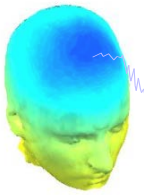
Terminal

File Edit View Terminal Tabs Help

```
>> EEG

EEG =

            setname: 'faces_4 continuous'
           filename: 'faces_4.set'
           filepath: '/home/julie/workshop06/'
            subject: ''
              group: ''
          condition: ''
            session: []
           comments: [15x48 char]
             nbchan: 33
             trials: 1
               pnts: 133175
              srate: 250
               xmin: 0
               xmax: 532.6960
              times: []
               data: [33x133175 single]
             icaact: [33x133175 single]
          * icawinv: [33x33 double]
          icasphere: [33x33 double]
        icaweights: [33x33 double]
        icachansind: [1x33 double]
           chanlocs: [1x33 struct]
         urchanlocs: []
           chaninfo: [1x1 struct]
                ref: 'common'
              event: [1x731 struct]
            urevent: [1x731 struct]
  eventdescription: {[]  []}
              epoch: []
  epochdescription: {}
             reject: [1x1 struct]
              stats: [1x1 struct]
           specdata: []
         specicaact: []
         splinefile: ''
      icasplinefile: ''
             dipfit: [1x1 struct]
            history: [1x1633 char]
              saved: 'no'
                etc: []

>>
```



English → MATLAB

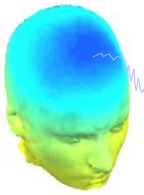


Source activation = unmixing * Channel data

Channel data = mixing (topo) * Source activation

```
EEG.icaact = (EEG.icaweights*EEG.icasphere) * EEG.data

EEG.data = EEG.icawinv * EEG.icaact
```

Now what…?



Part 1

Getting an overview of your ICs

Part 2

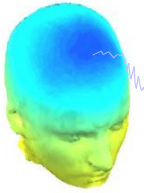
Classifying/Evaluating ICs

Part 3

Detailed look at IC properties

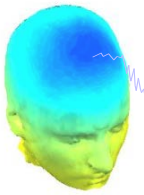
- ERP
- Spectrum
- ERP images
- ERSP

IC Evaluation Practicum (Day 1)



- **ICA Component Classifier Competition**
- **Traditional Practicum using faces_4.set**

Now what…?



Part 1

Getting an overview of your ICs

Part 2

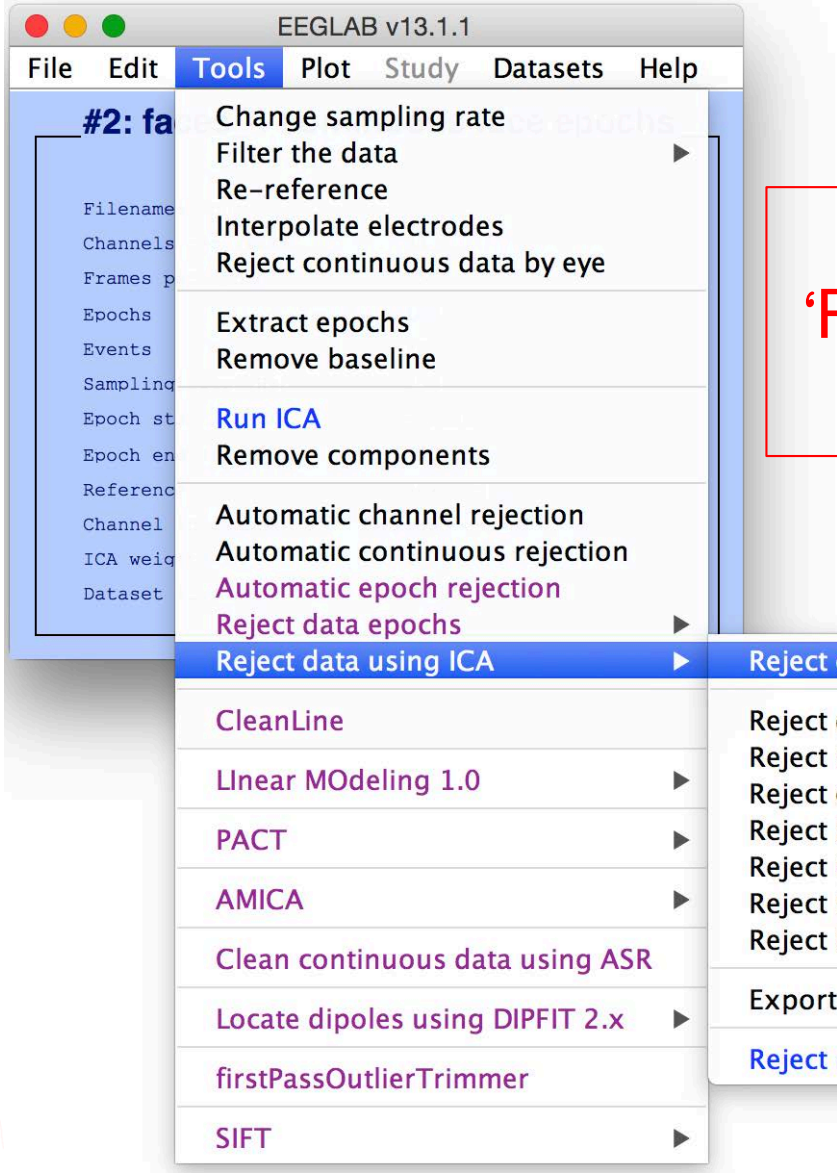
Classifying/Evaluating ICs

Part 3

Detailed look at IC properties

- ERP
- Spectrum
- ERP images
- ERSP

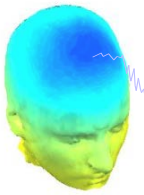
A convenient ‘trick’…

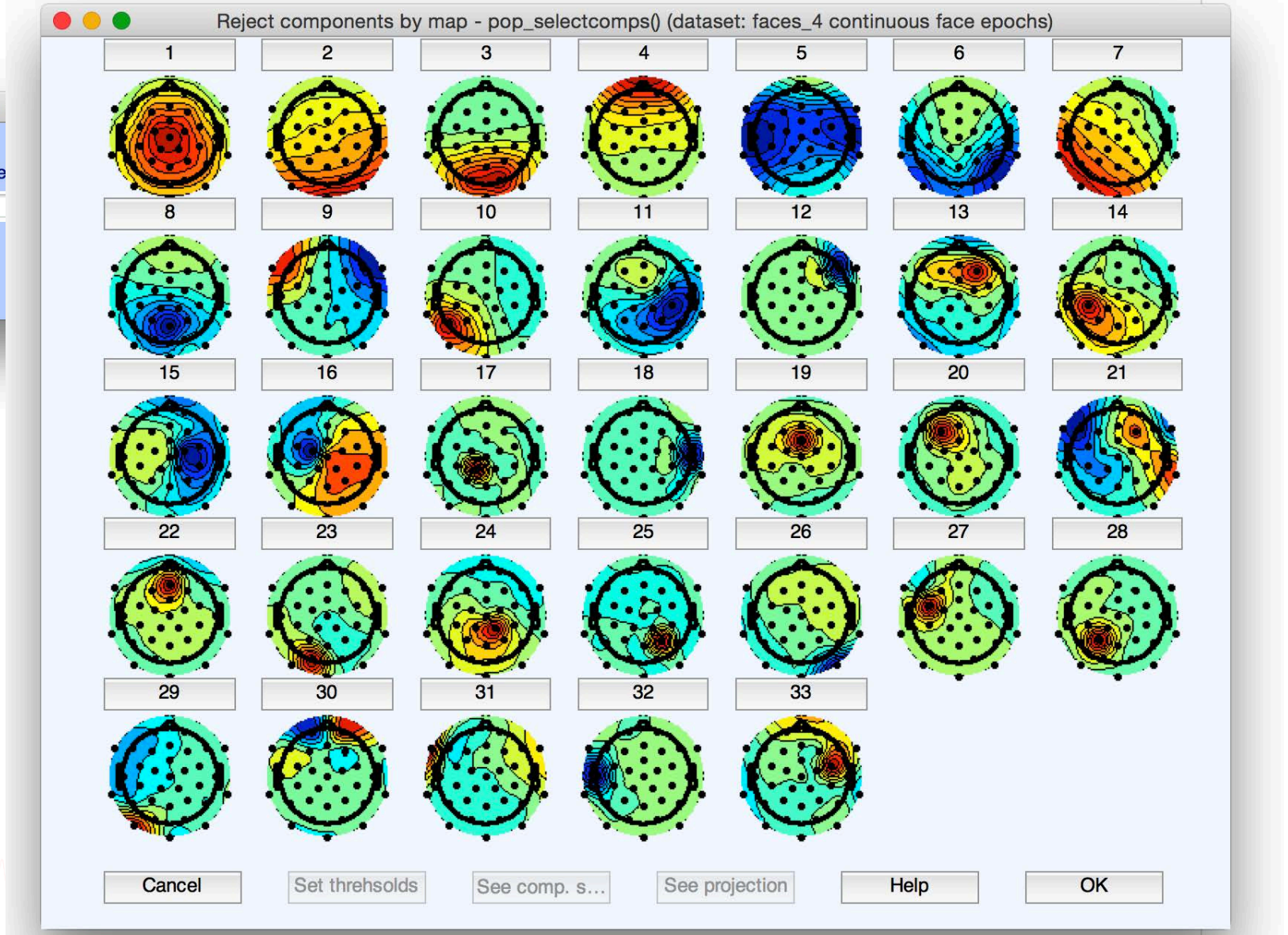


Use
‘Reject components by map’
to survey components

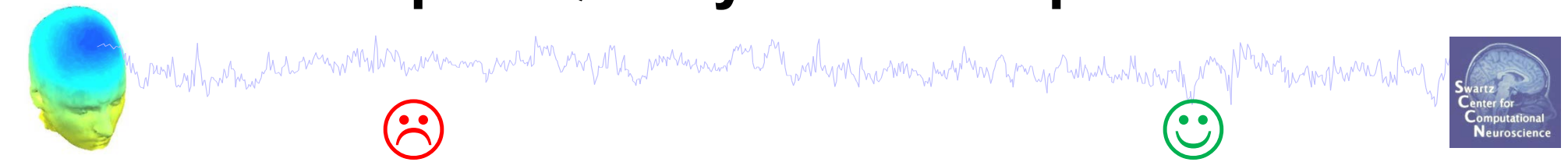
NB: A new plugin
"viewprops"
is available!

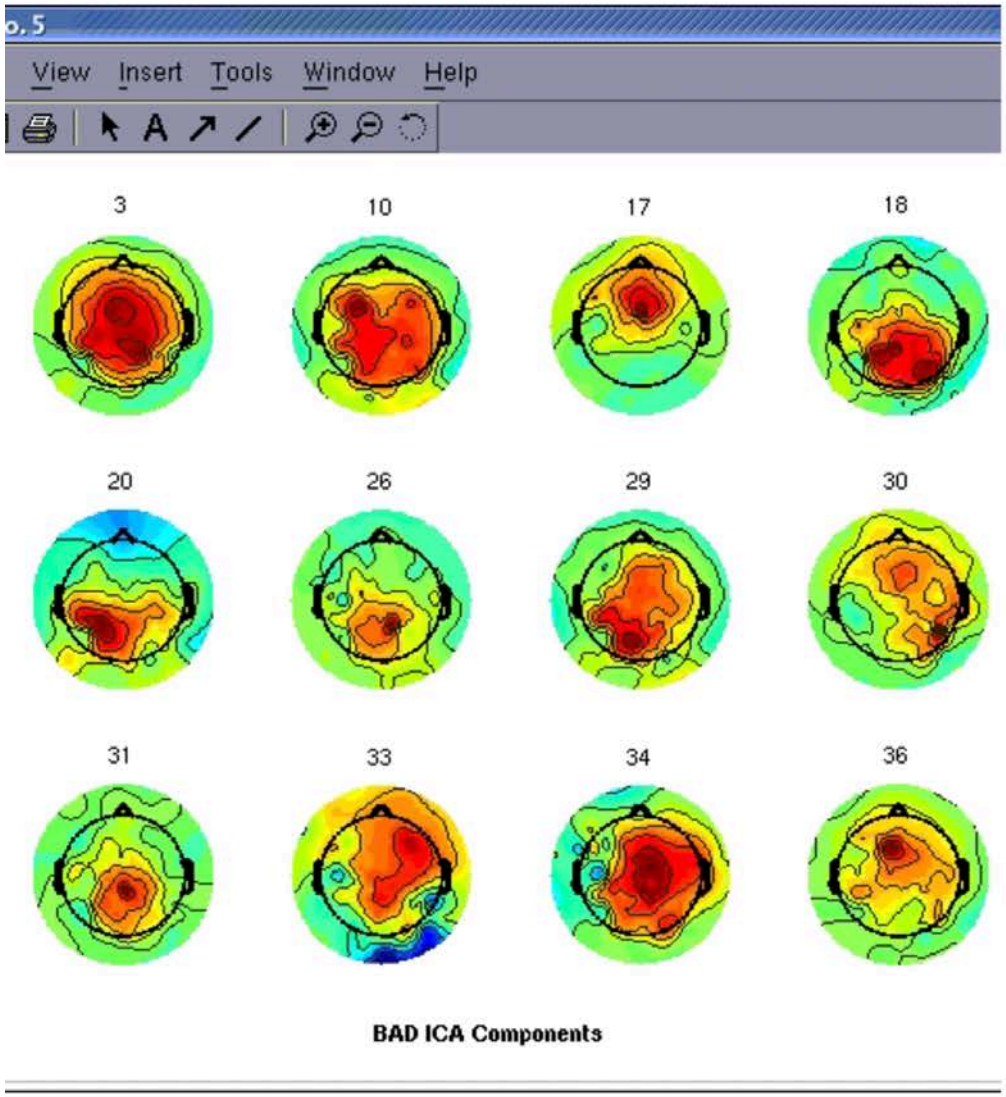
An interactive overview of ICs

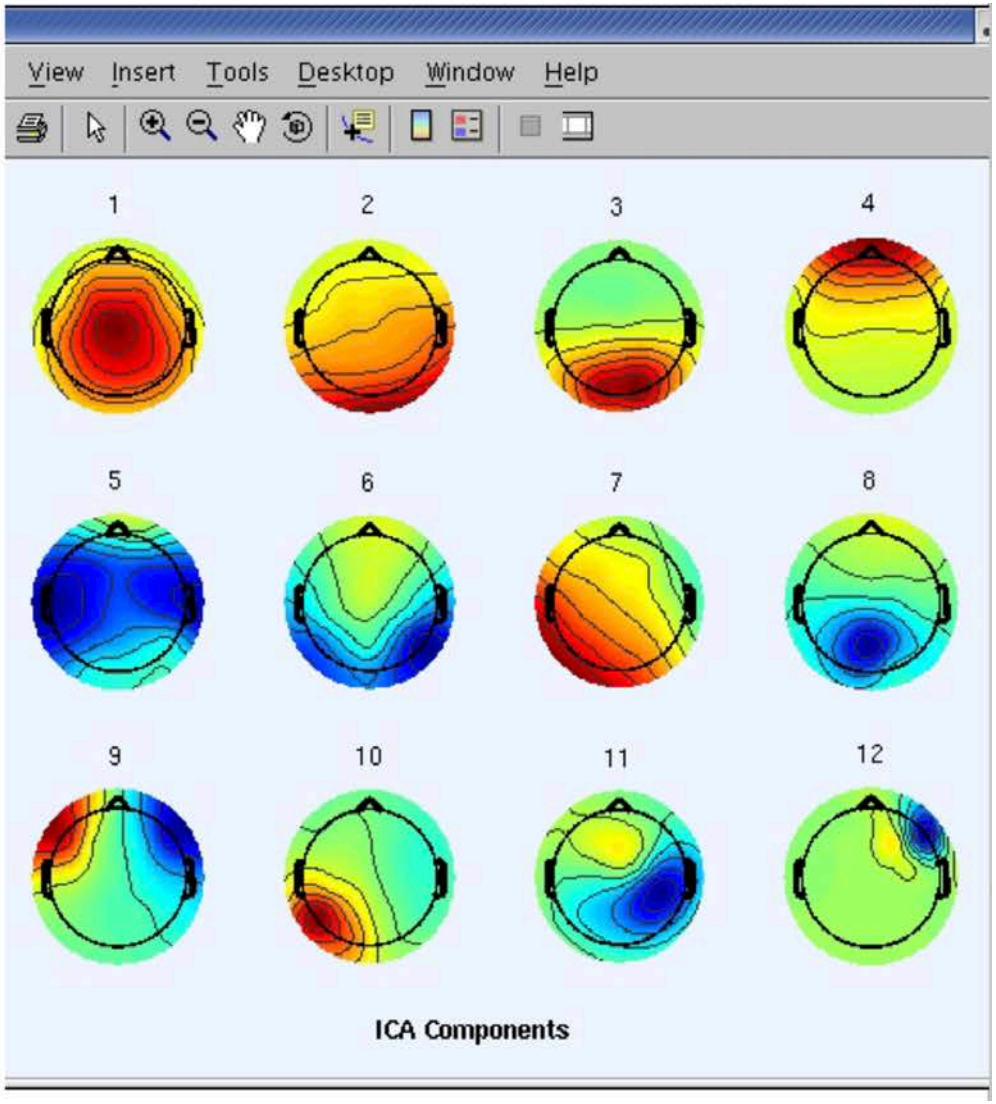




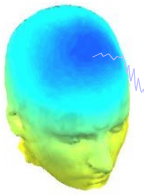
Step 0: Quality of Decomposition

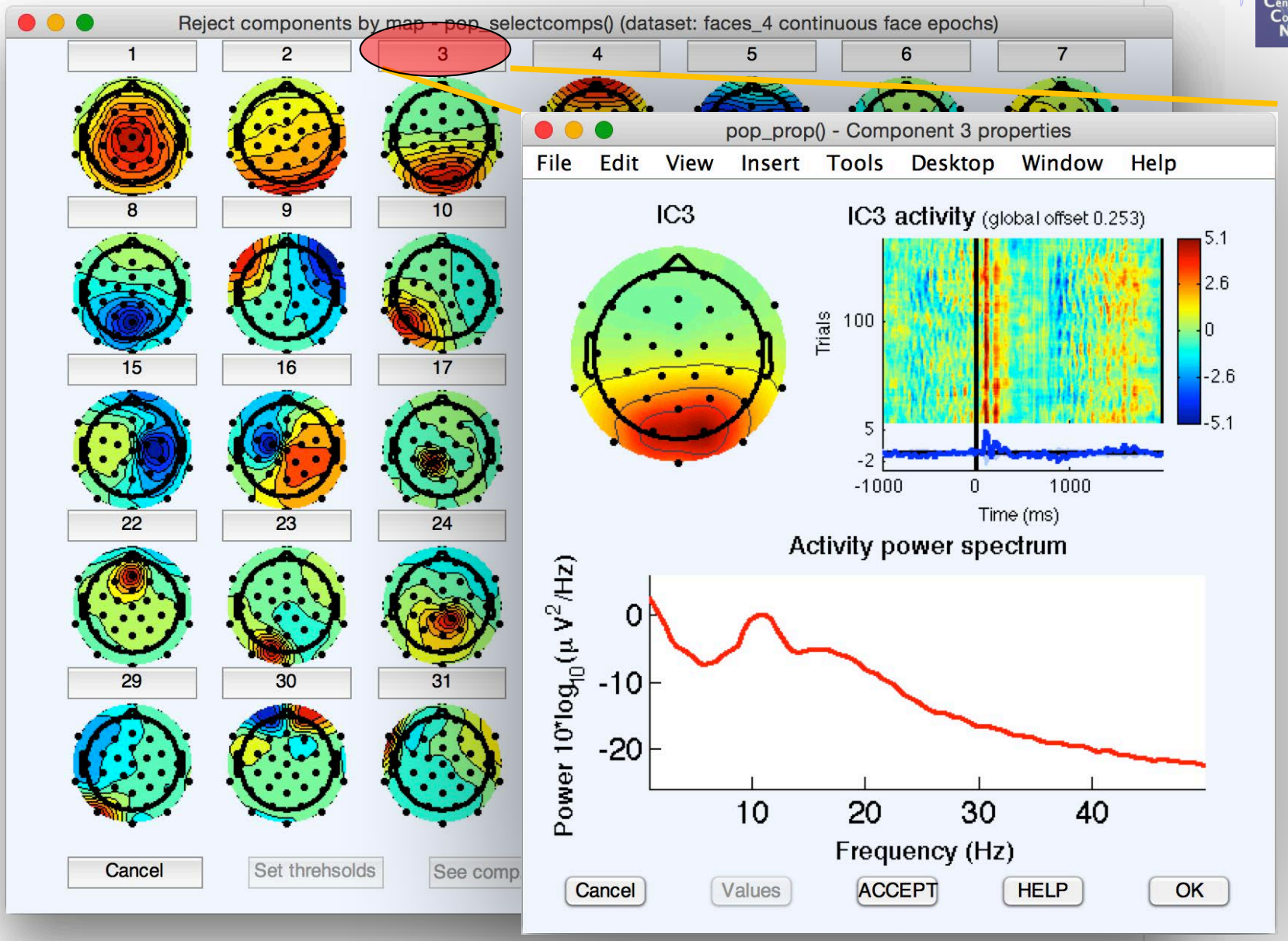




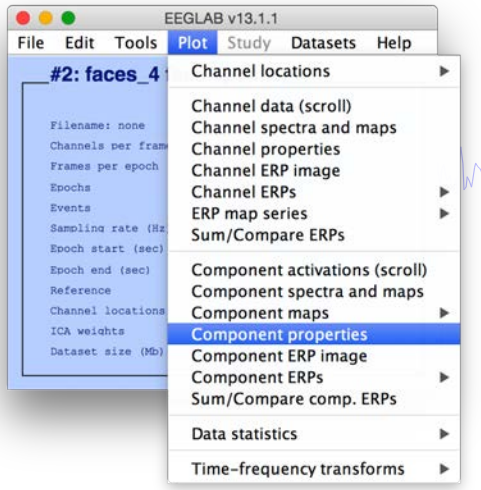


Examining IC Properties



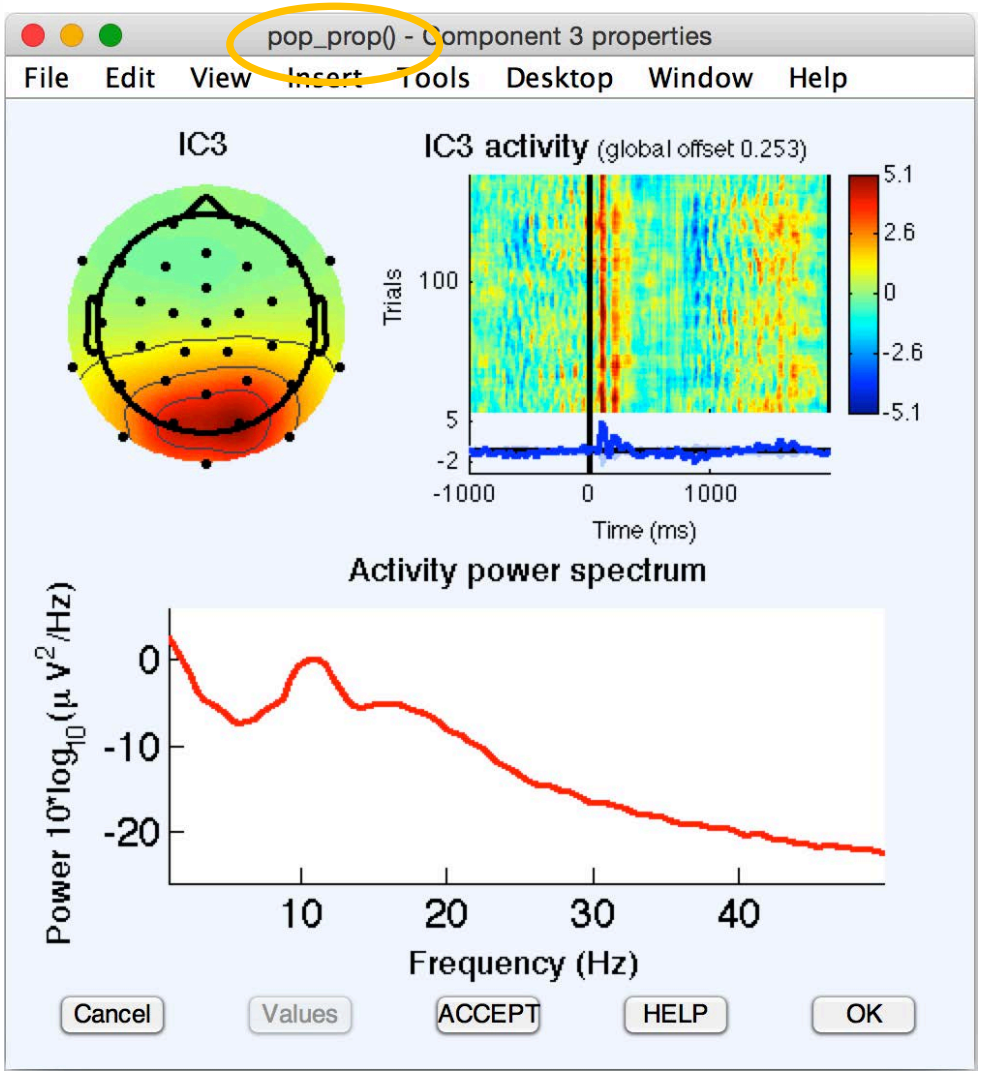


IC Properties



IC Topography

`topoplot()`



ERP Image

&

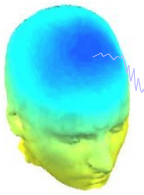
ERP

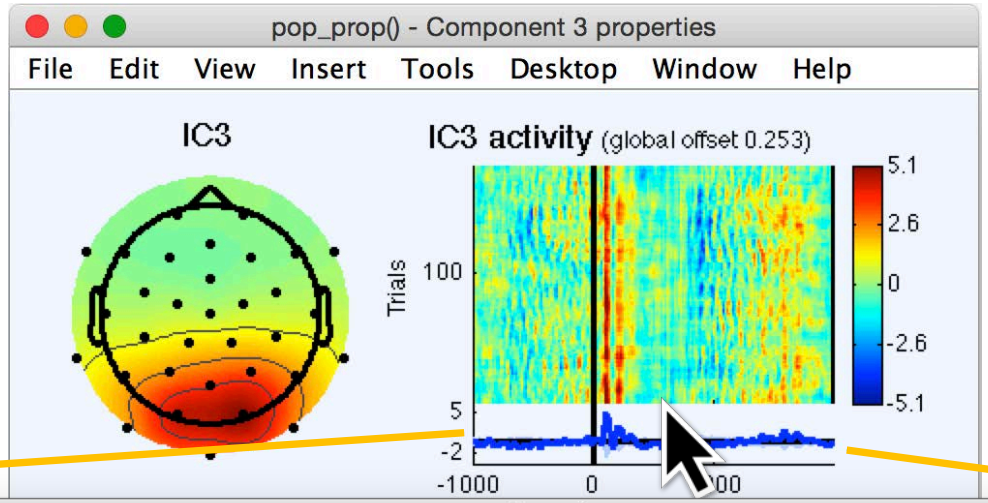
`erpimage()`

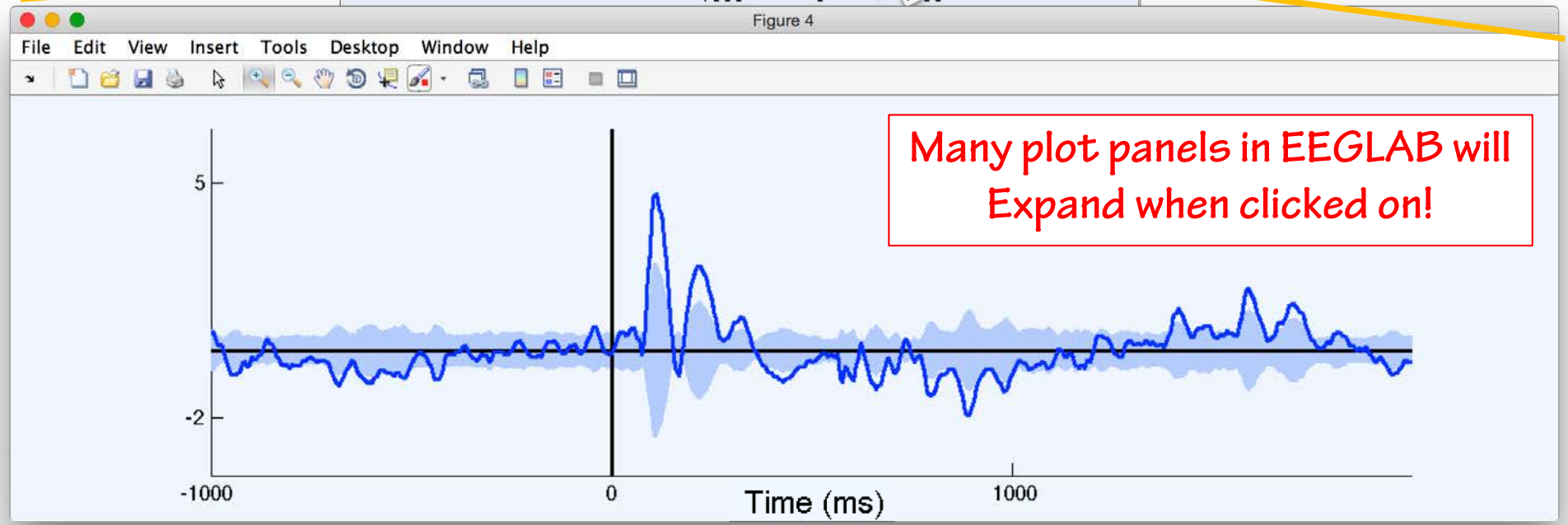
Power Spectrum

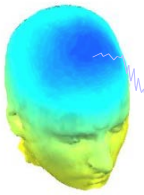
`spectopo()`

Click to expand…









Part 1

Getting an overview of your ICs

Part 2

Classifying/Evaluating ICs

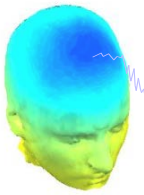
- Eye Artifacts
- Muscle Artifacts
- Other Artifacts
- Brain ICs

Part 3

Detailed look at IC properties

- ERP
- Spectrum
- ERP images
- ERSP

Evaluating ICs



Over time, most EEGLAB users develop a *heuristic* sense of which ICs might be brain vs. artifact.

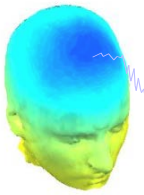
Heuristics are generally based on:

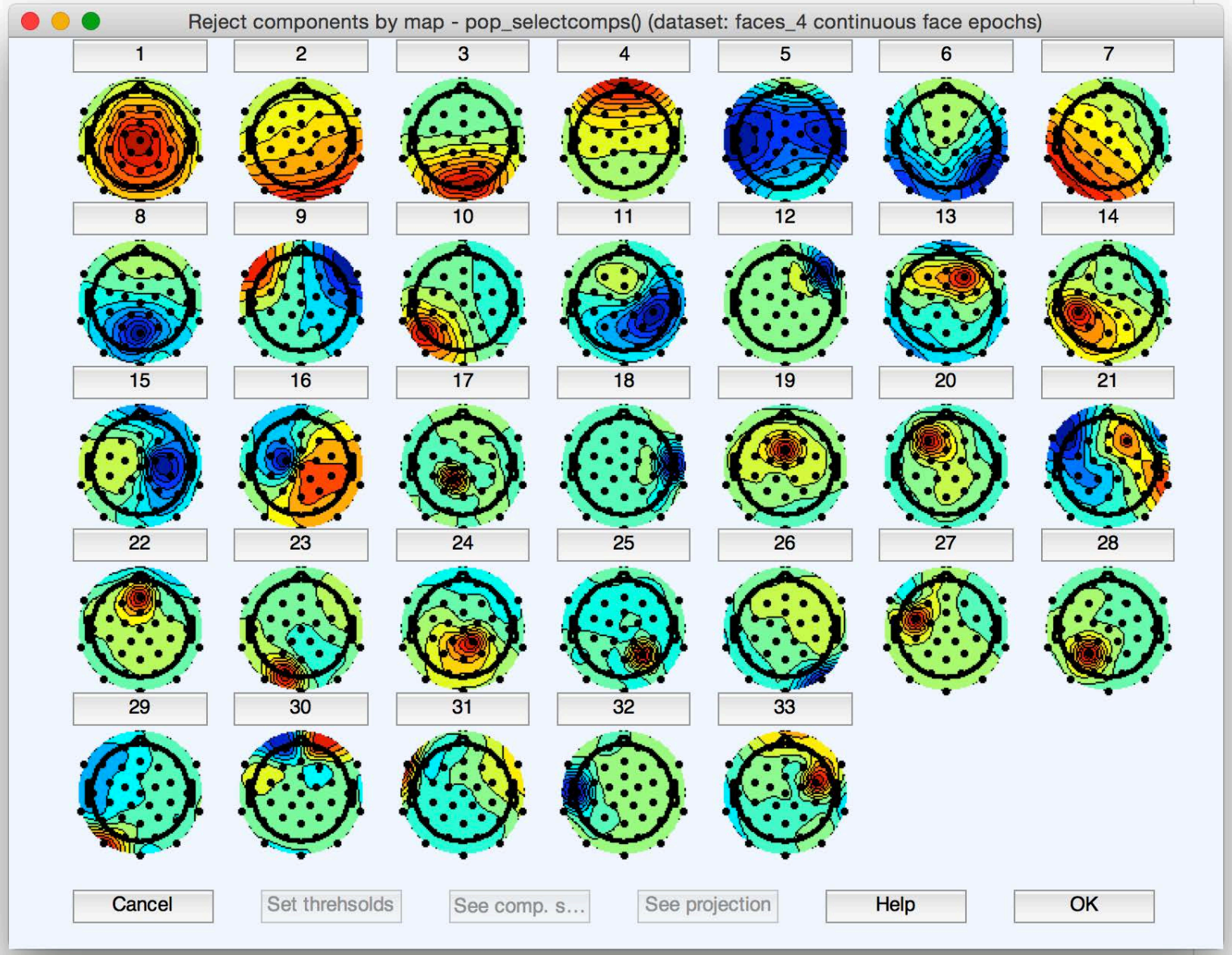
- Topography
- Component Activities (scroll)
- ERP
- Power Spectrum

IC Classification can be used to 'clean' data—study likely brain activity without artifacts

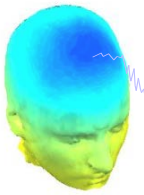
There are new efforts to automate this process, but doing it by hand is a good place to start to build intuition

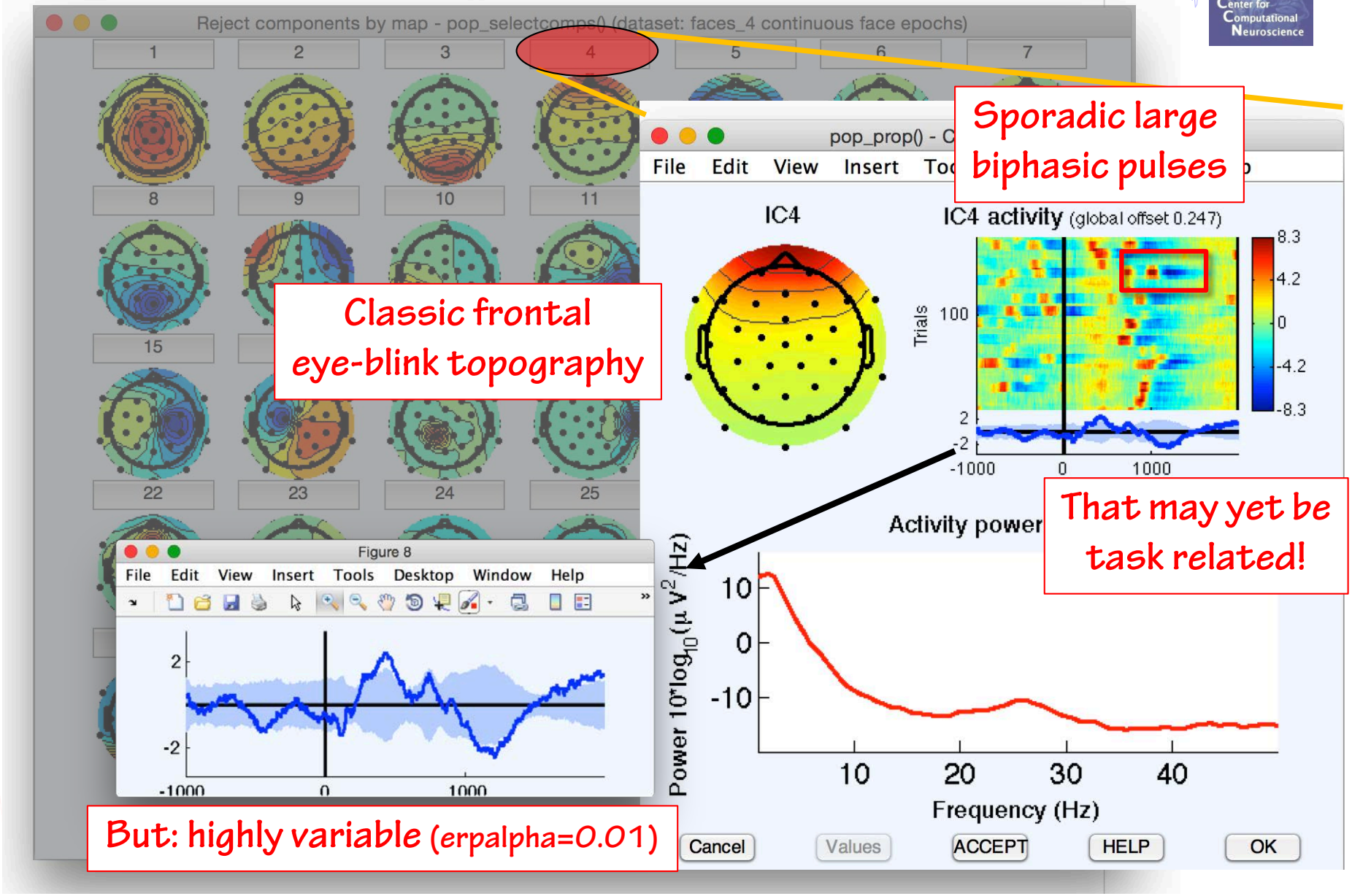
Topography




Reject components by map - pop_selectcomps() (dataset: faces_4 continuous face epochs)
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
Cancel
Set threhsolds
See comp. s...
See projection
Help
OK

IC 4 – eyeblink





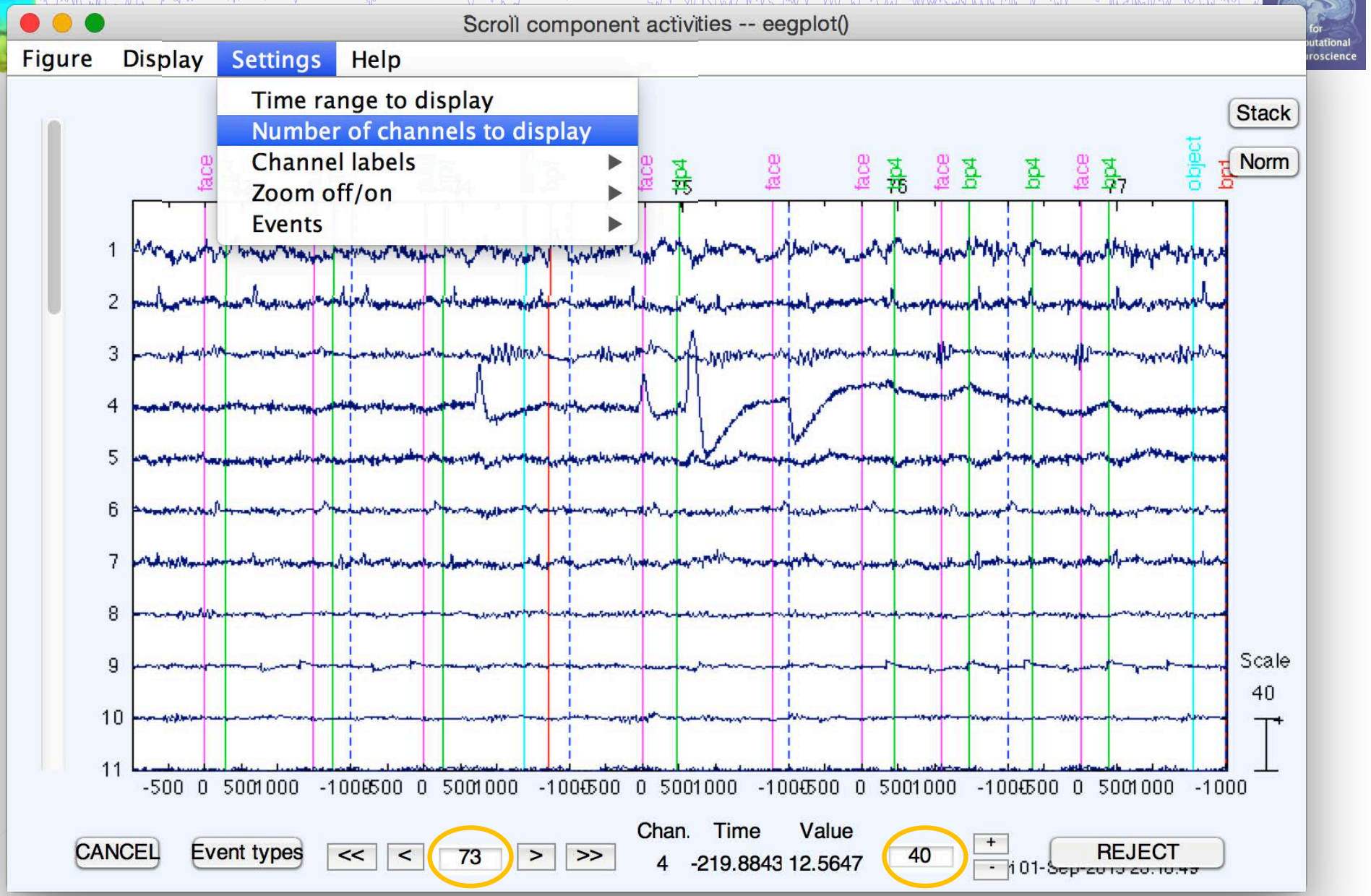
Sporadic large biphasic pulses

Classic frontal eye-blink topography

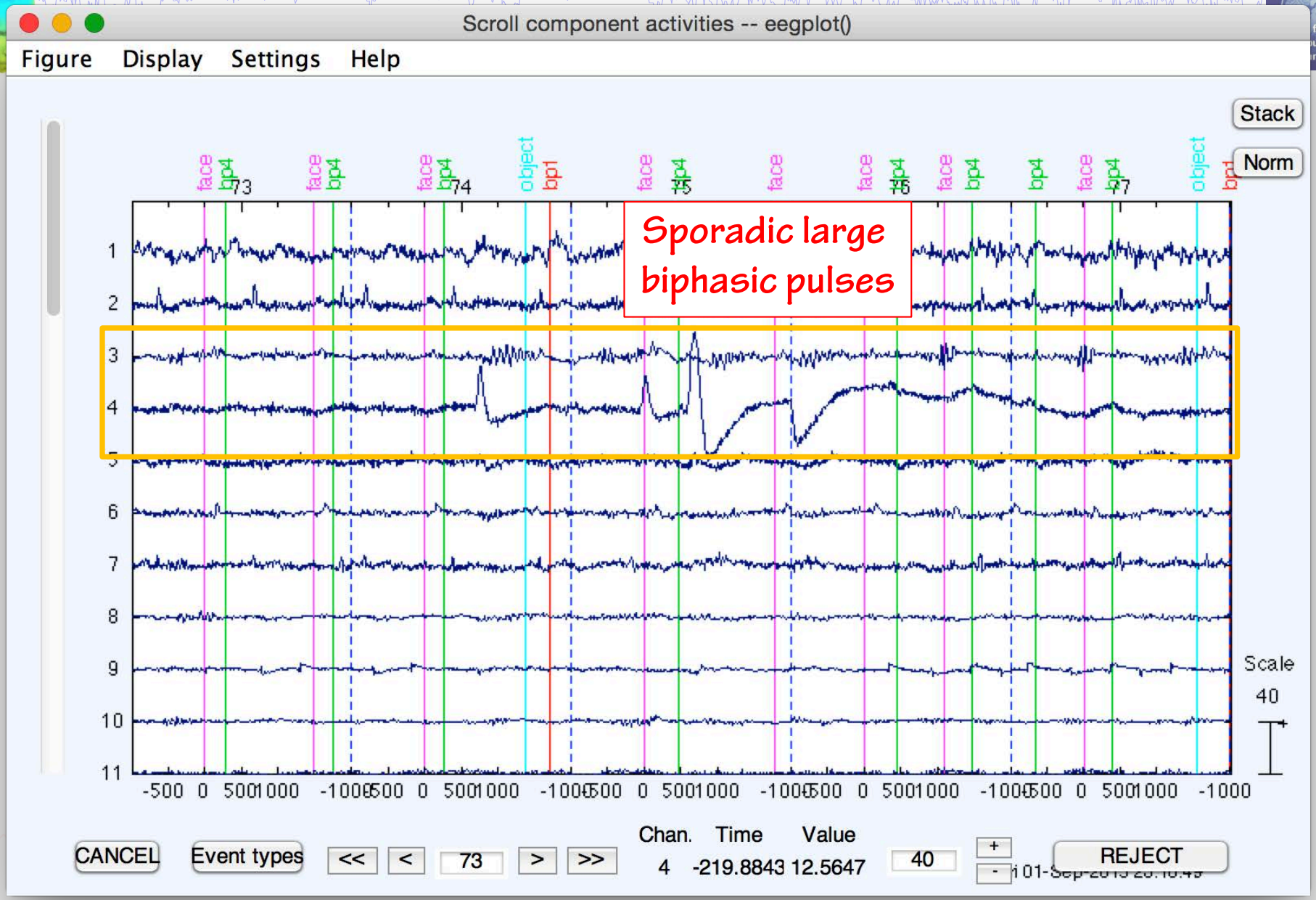
That may yet be task related!

But: highly variable (erpalpha=0.01)

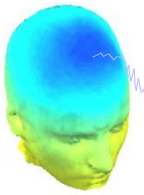
Plot → Component Activations (scroll)



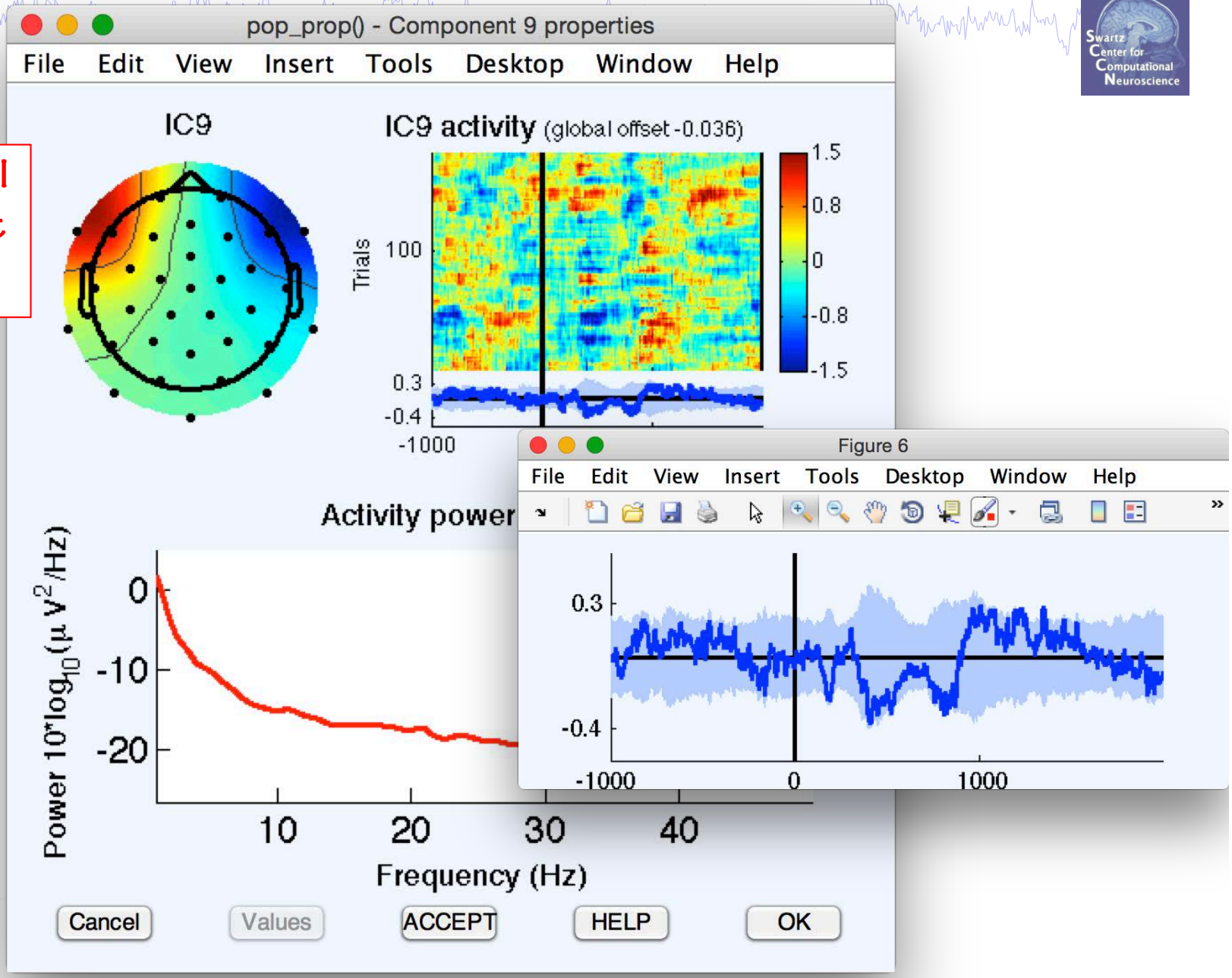
IC 4 Activation – eyeblink



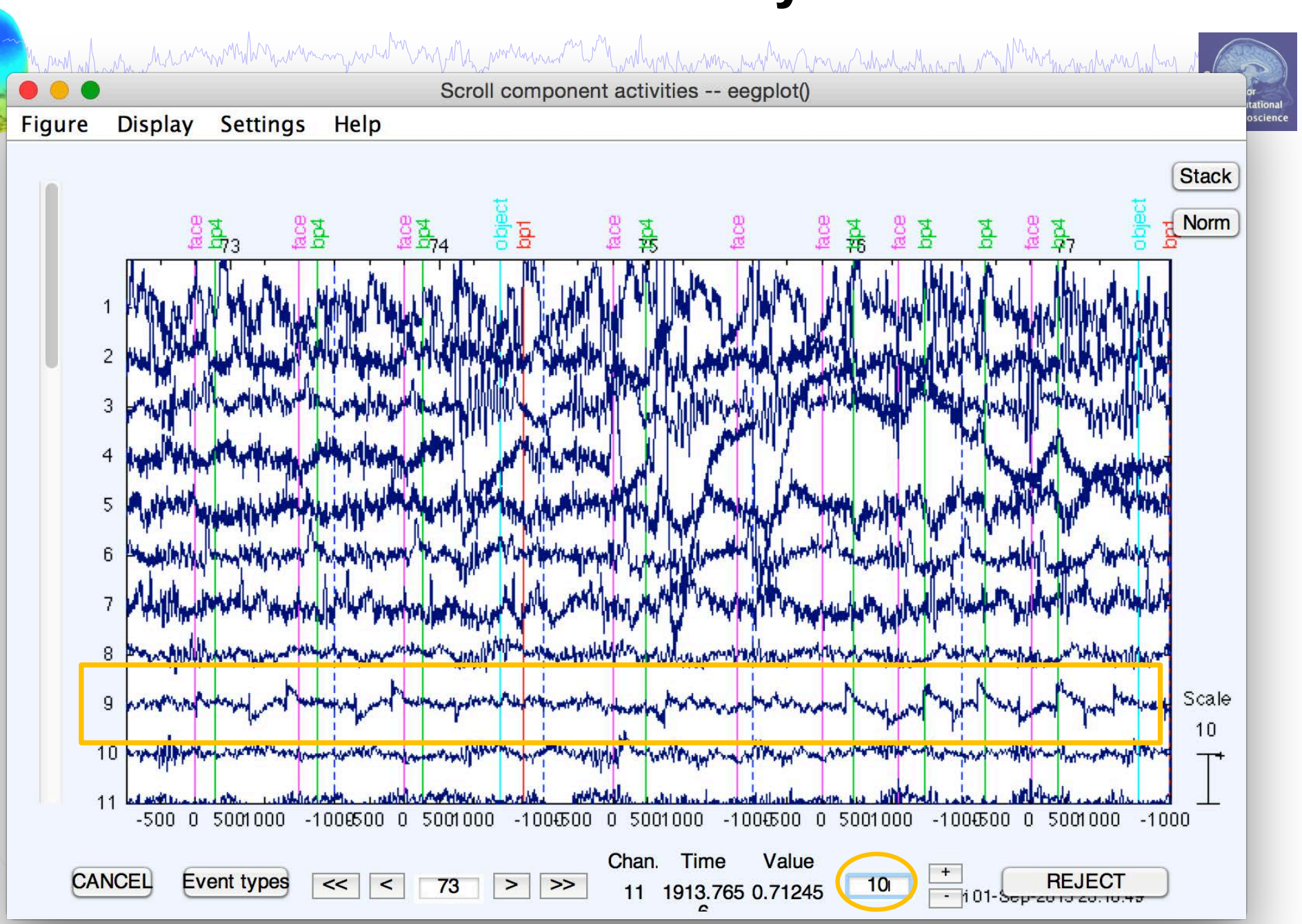
IC 9 – lateral eye movement



Classic frontal eye-movement topography

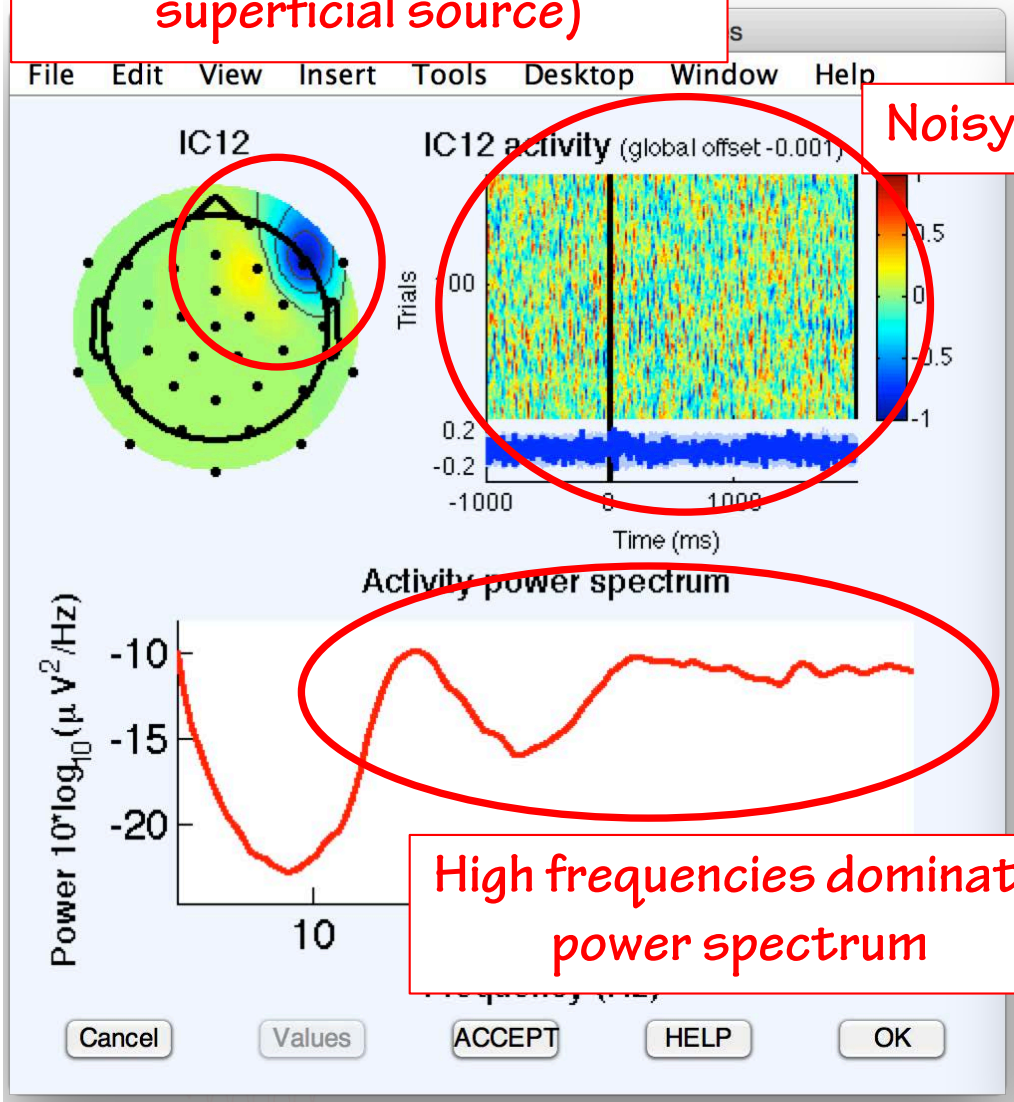


IC 9 Activation – lateral eye movement

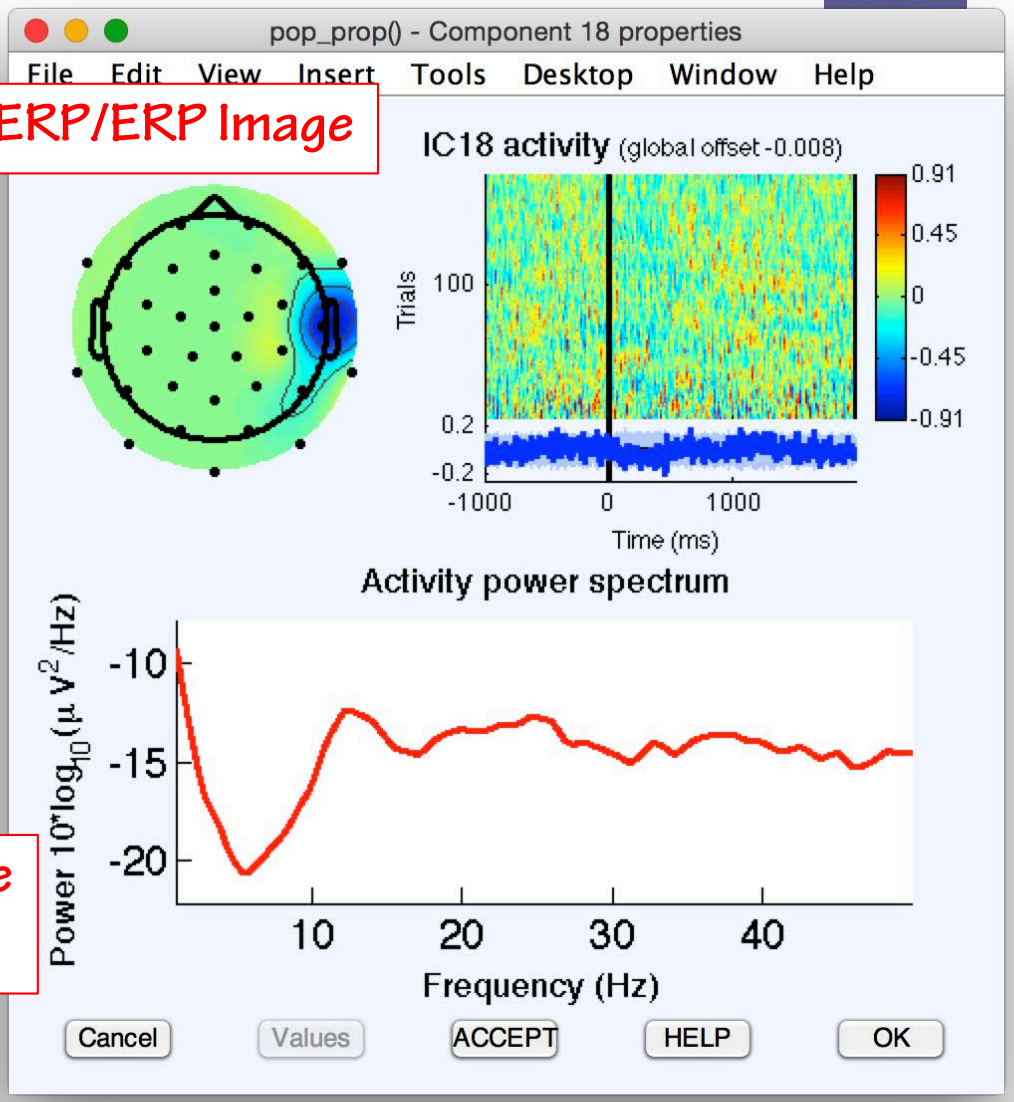


IC 12, 18 – Muscle

Narrowly spaced dipolar topography (consistent with superficial source)



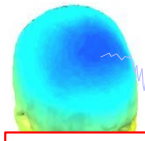
Noisy ERP/ERP Image



High frequencies dominate power spectrum

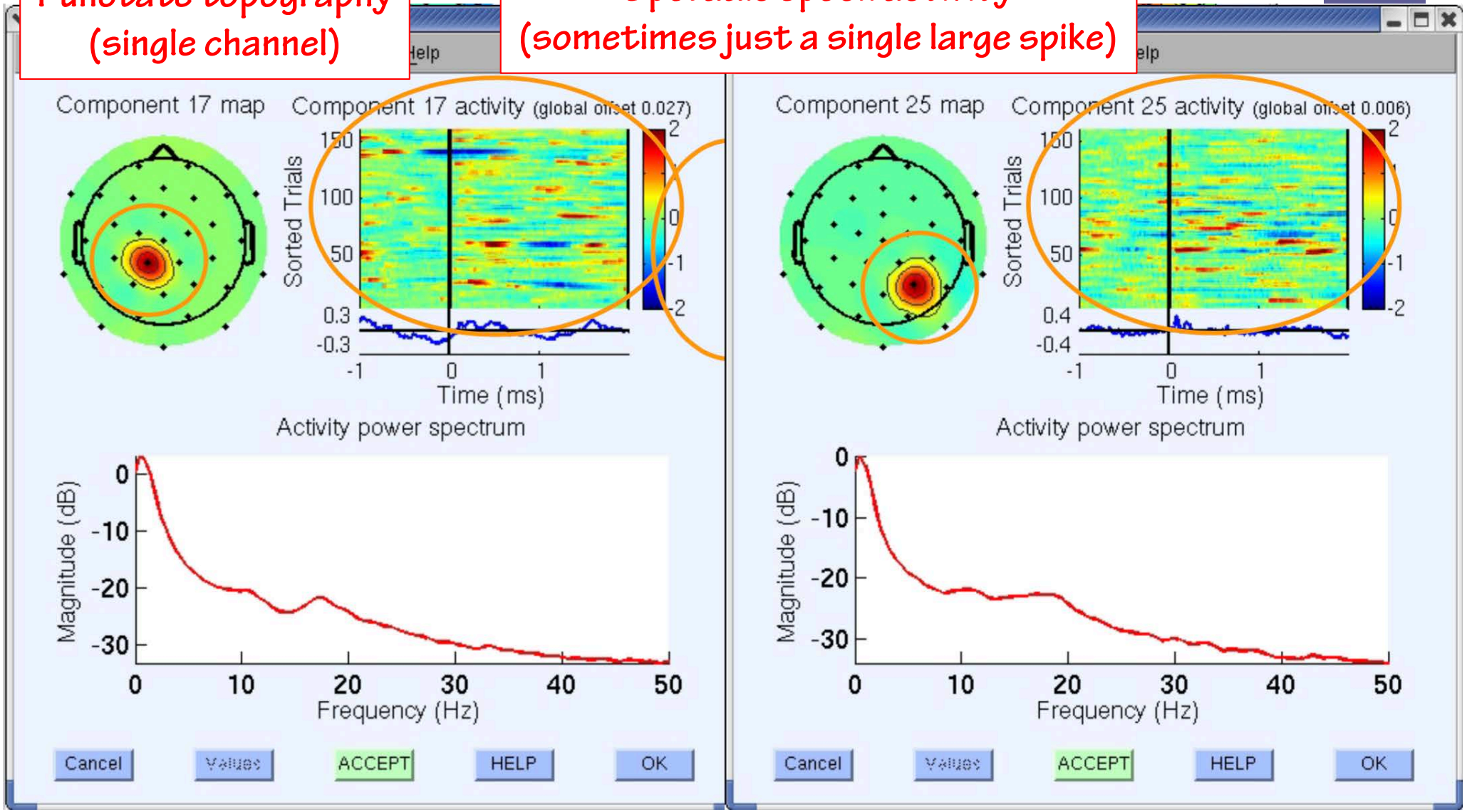
IC 12, 18 Activation – Muscle

IC 17, 25 – Bad channels

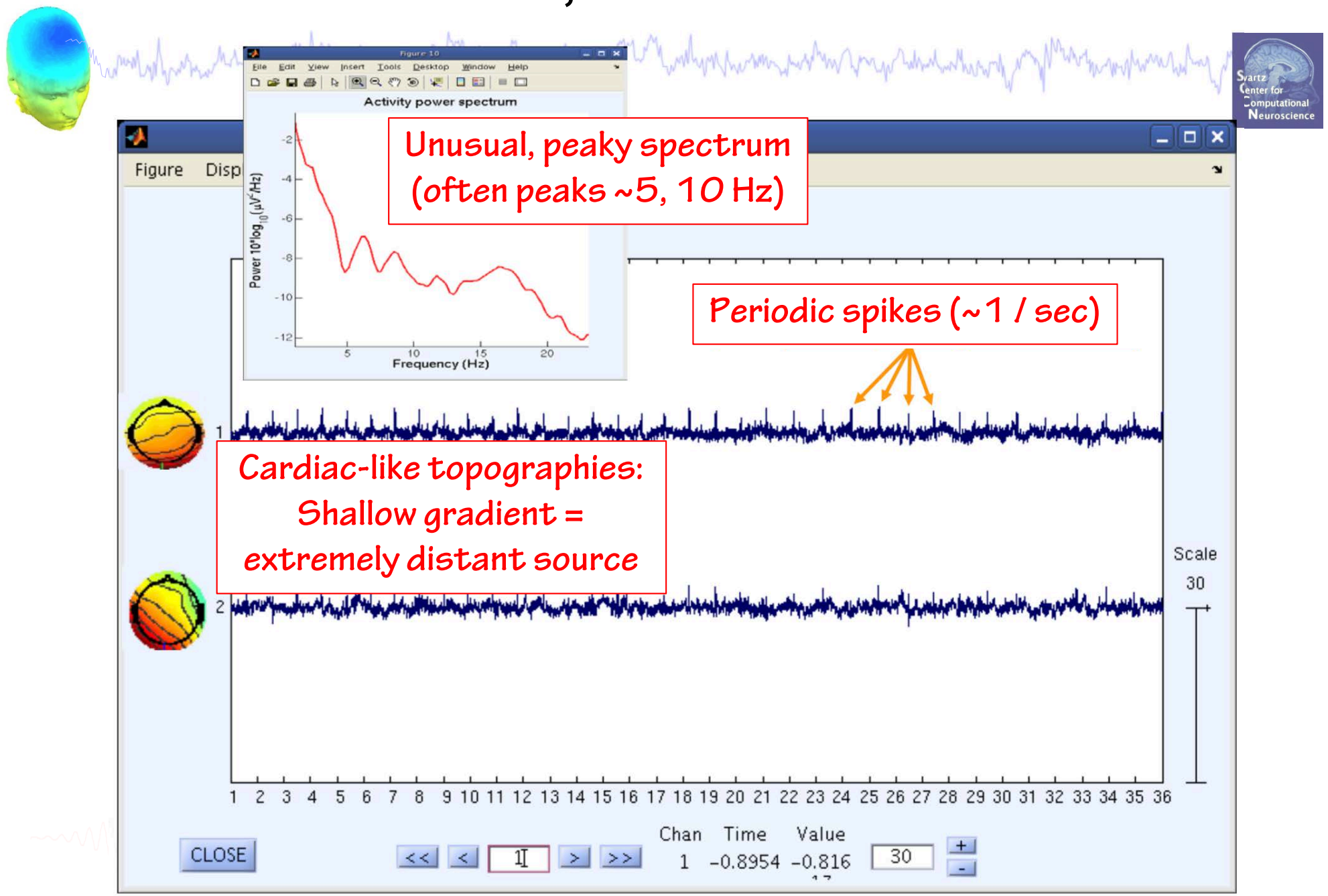


Punctate topography (single channel)

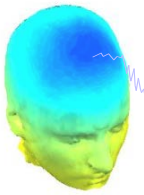
Sporadic epoch activity (sometimes just a single large spike)

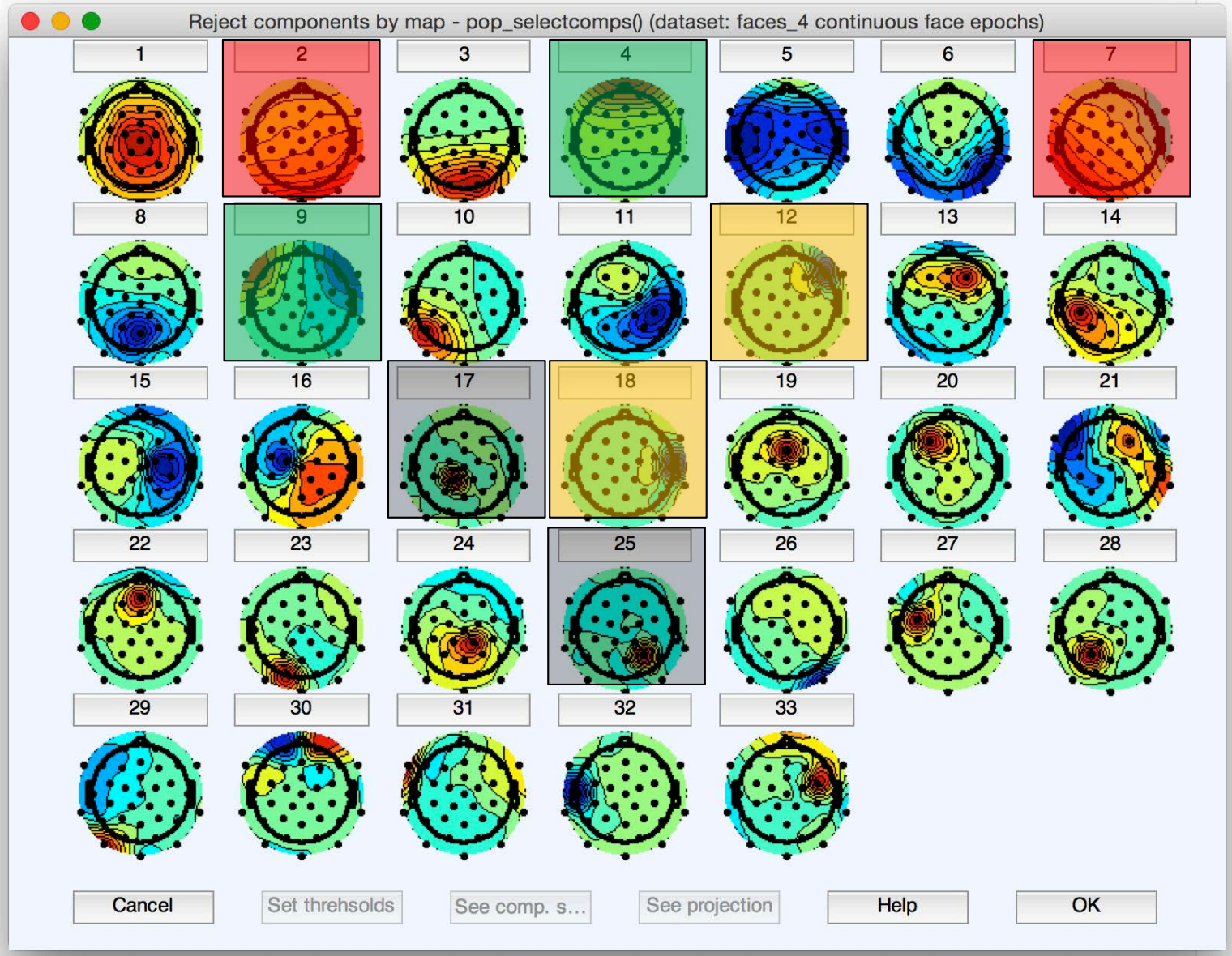


IC 2, 7 – Cardiac



Artifacts





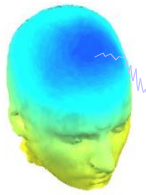
Eye

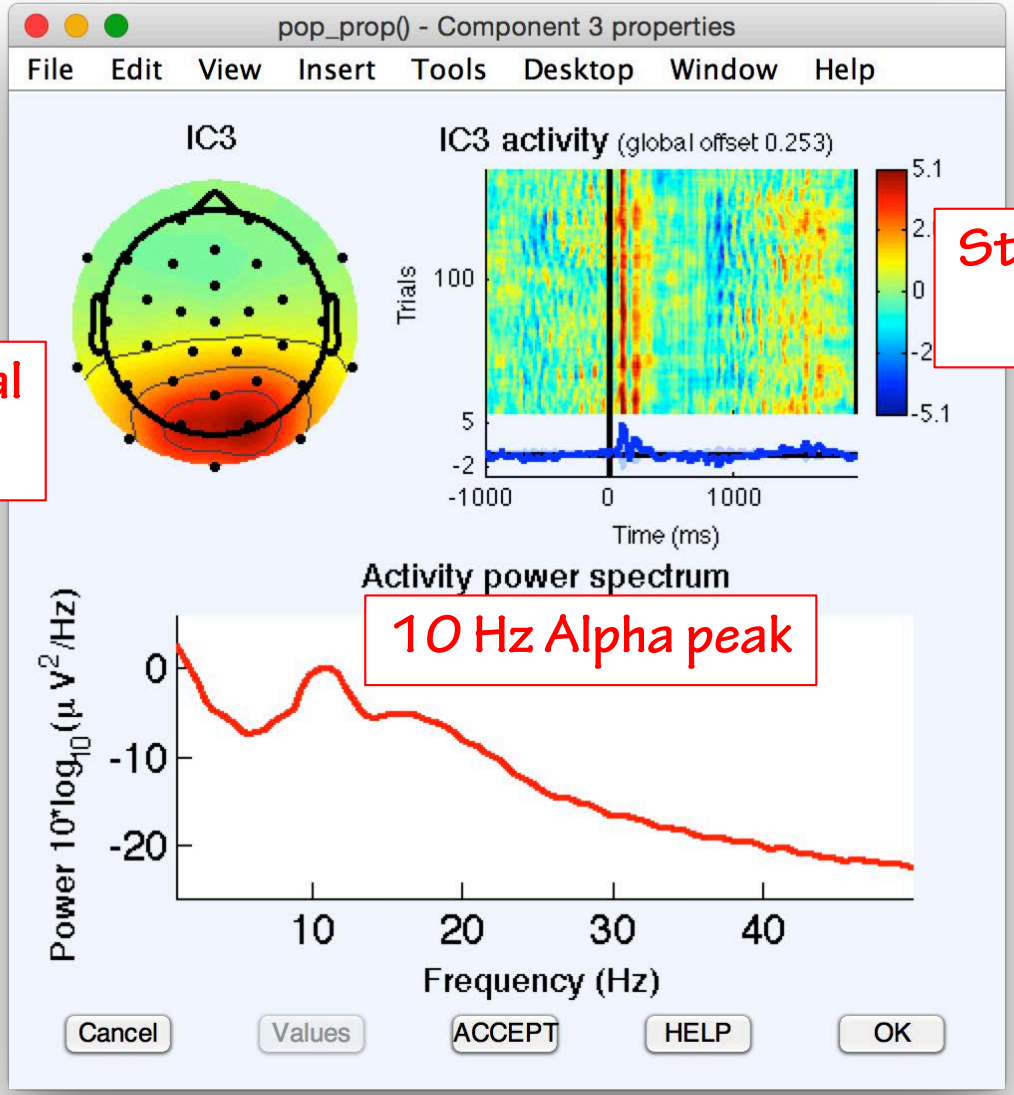
Muscle

Cardiac

Badchan

Brain ICs

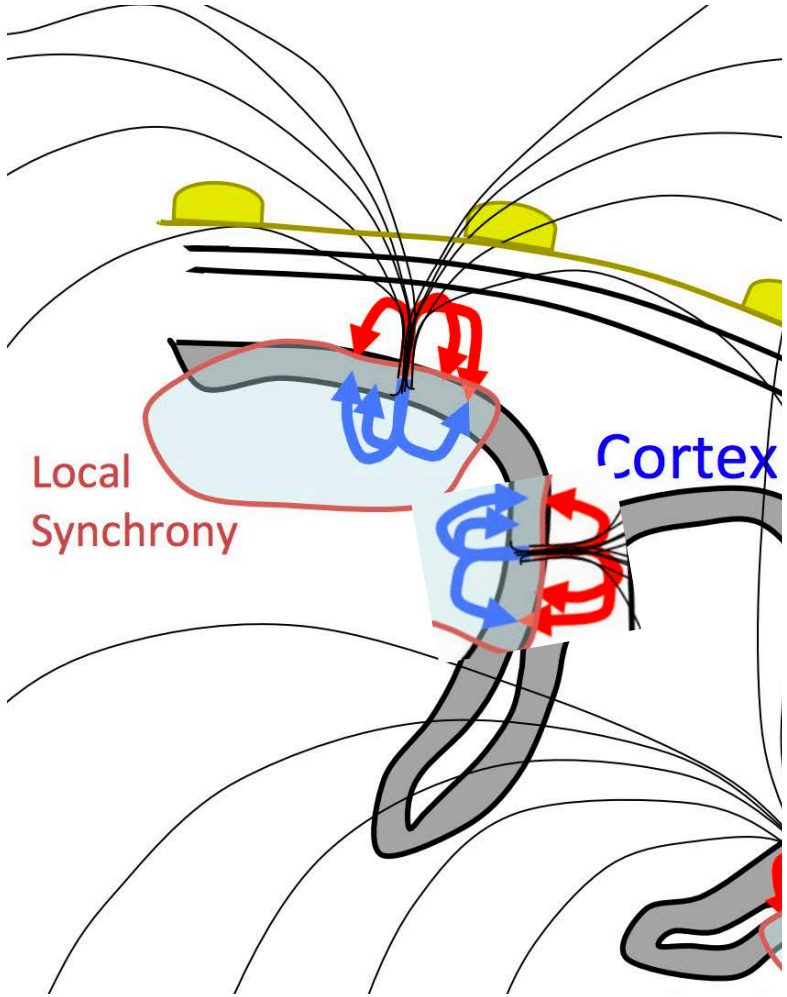




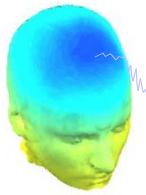
Classic occipital topography

Strongly task-related ERP Image & ERP

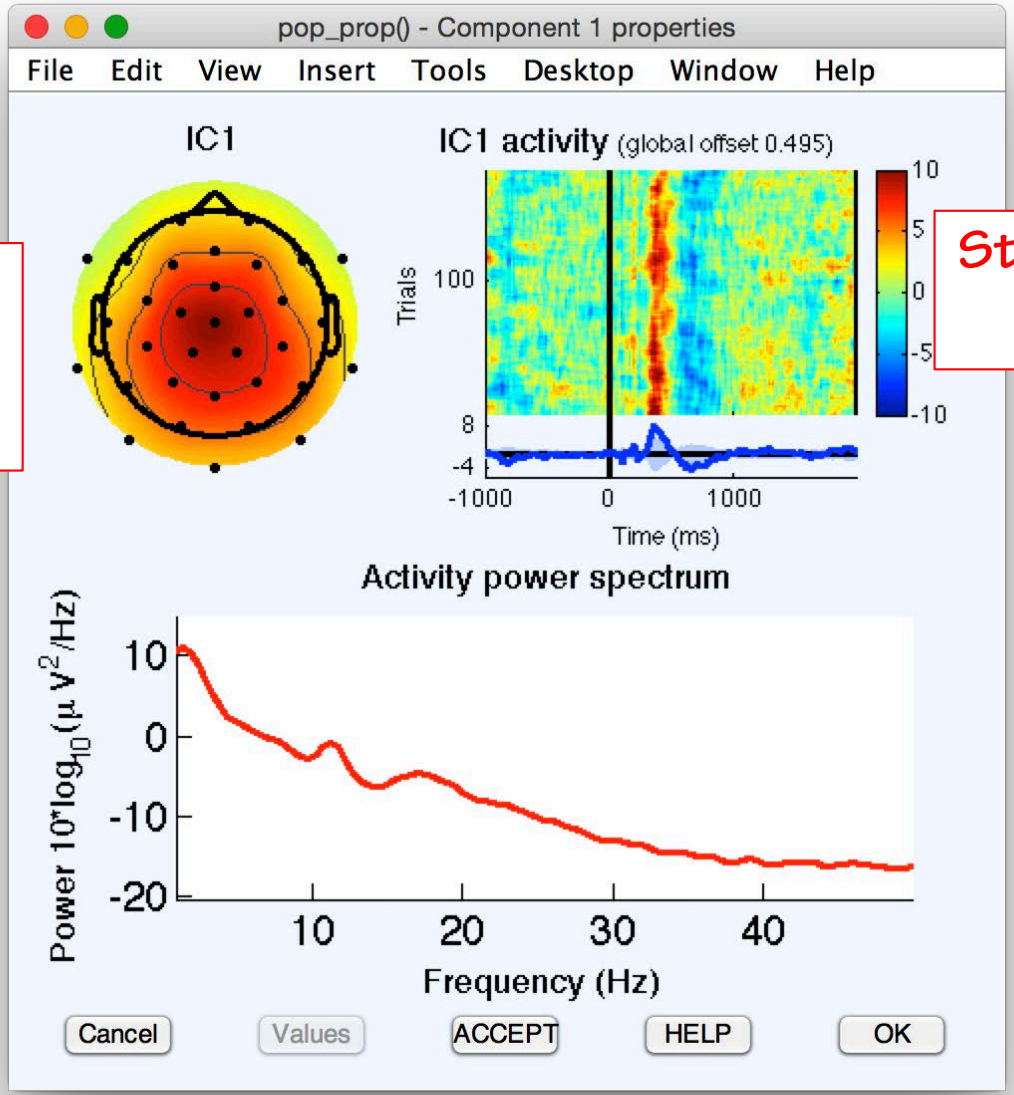
Dipole orientation matters



Brain ICs

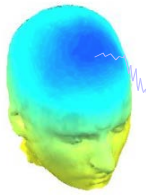


Classic radial-dipole source topography



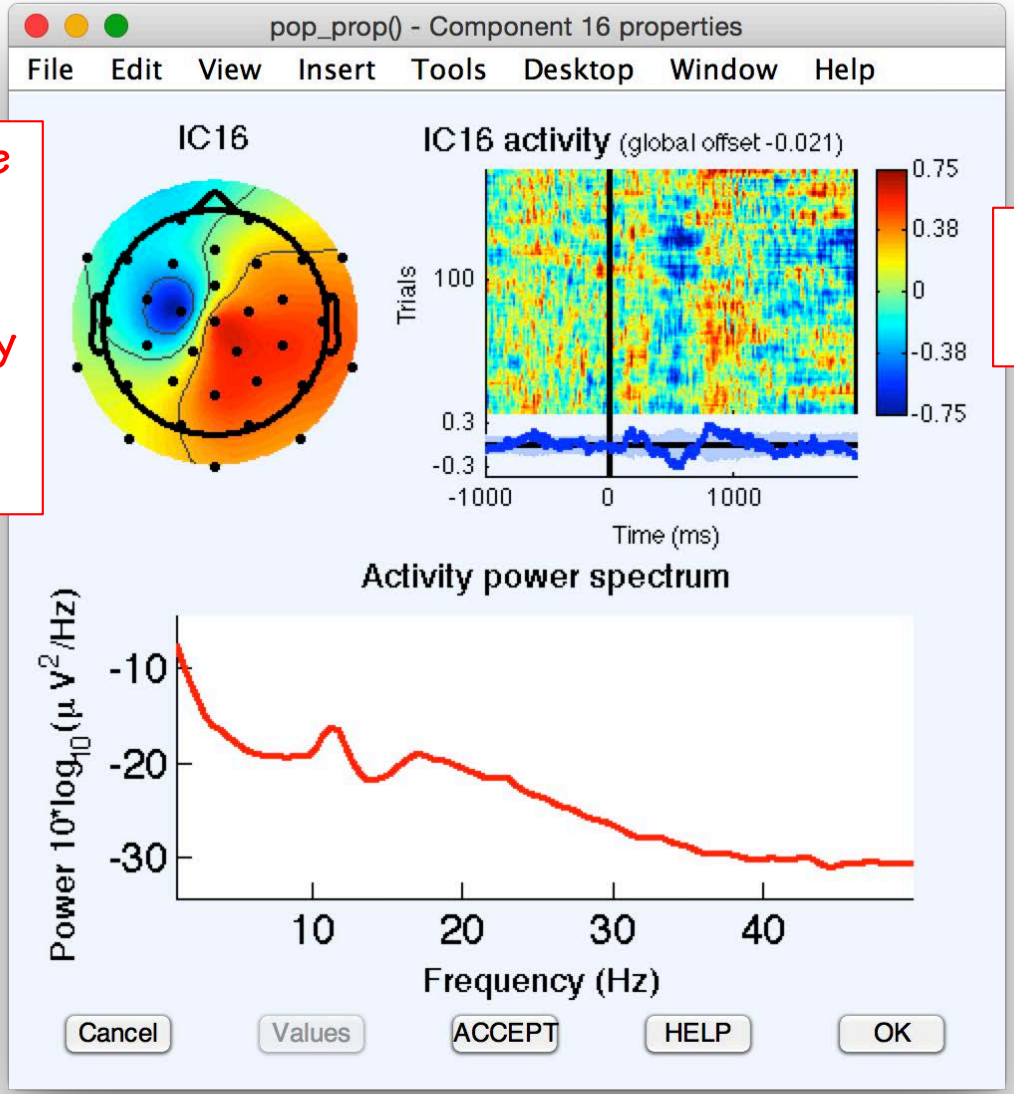
Strongly task-related ERP Image & ERP

Brain ICs



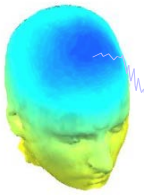
Classic tangential-dipole source topography

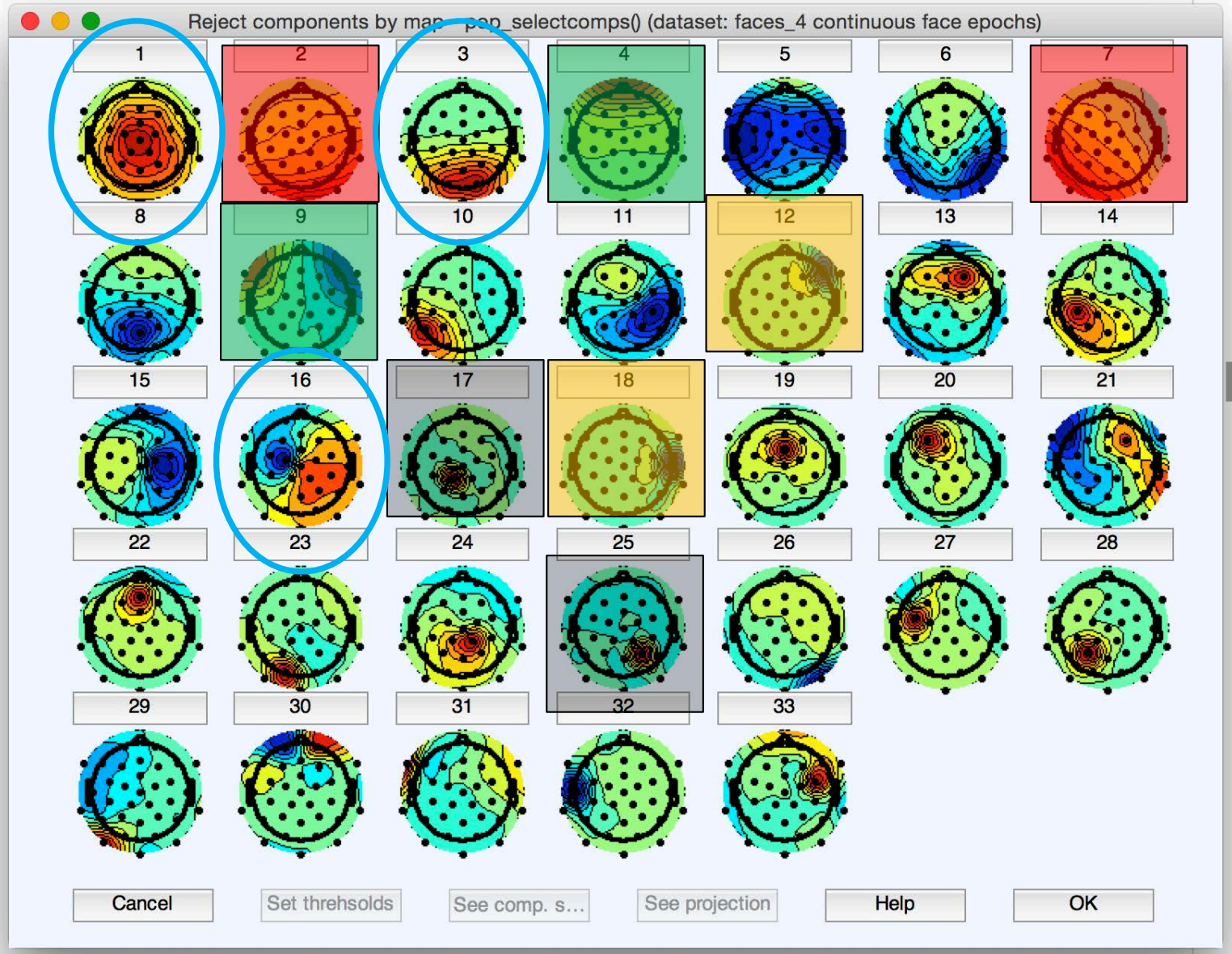
two peaks, not as closely spaced as muscle: deeper



Task-related ERP Image & ERP

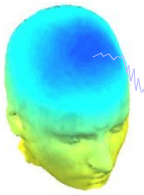
IC Classification…so far





Eye
Muscle
Cardiac
Badchan
Brain

Now what…?



Part 1

Getting an overview of your ICs

Part 2

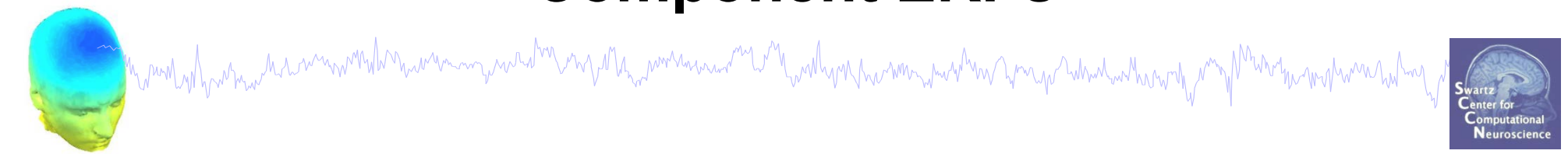
Classifying/Evaluating ICs

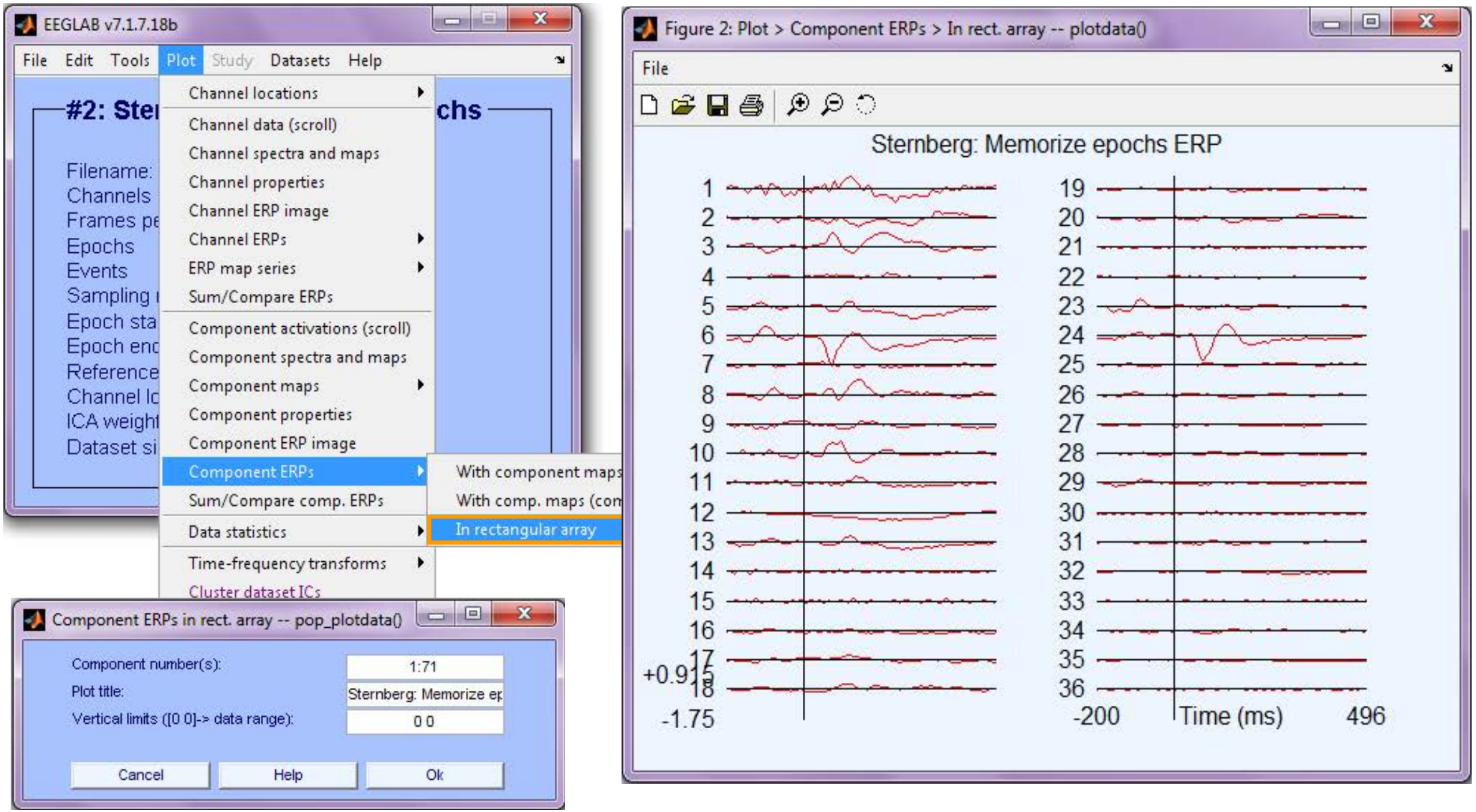
Part 3

Detailed look at IC properties

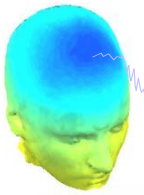
- ERP
- Spectrum
- ERP images
- ERSP

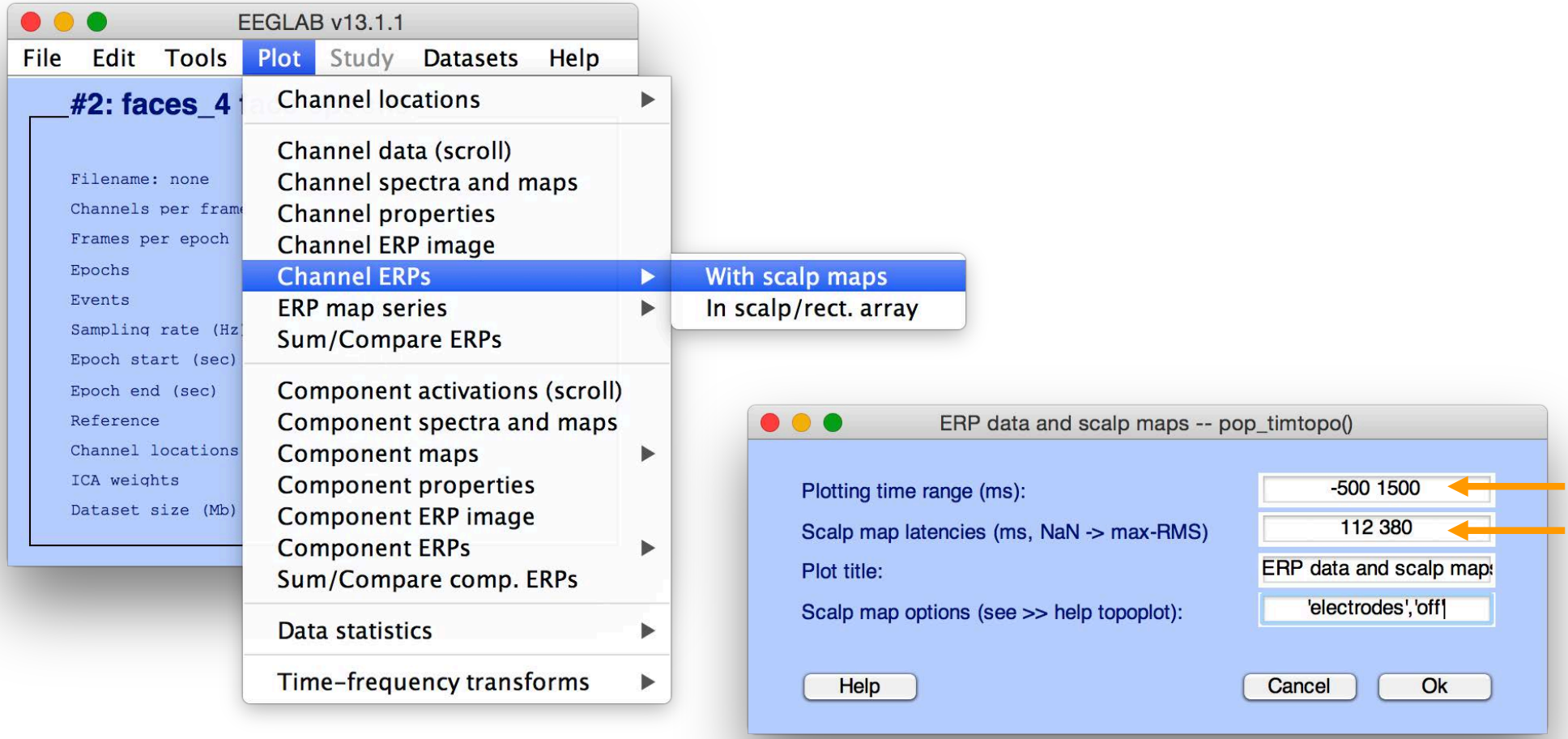
Component ERPs



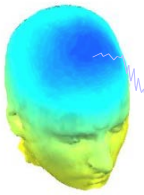


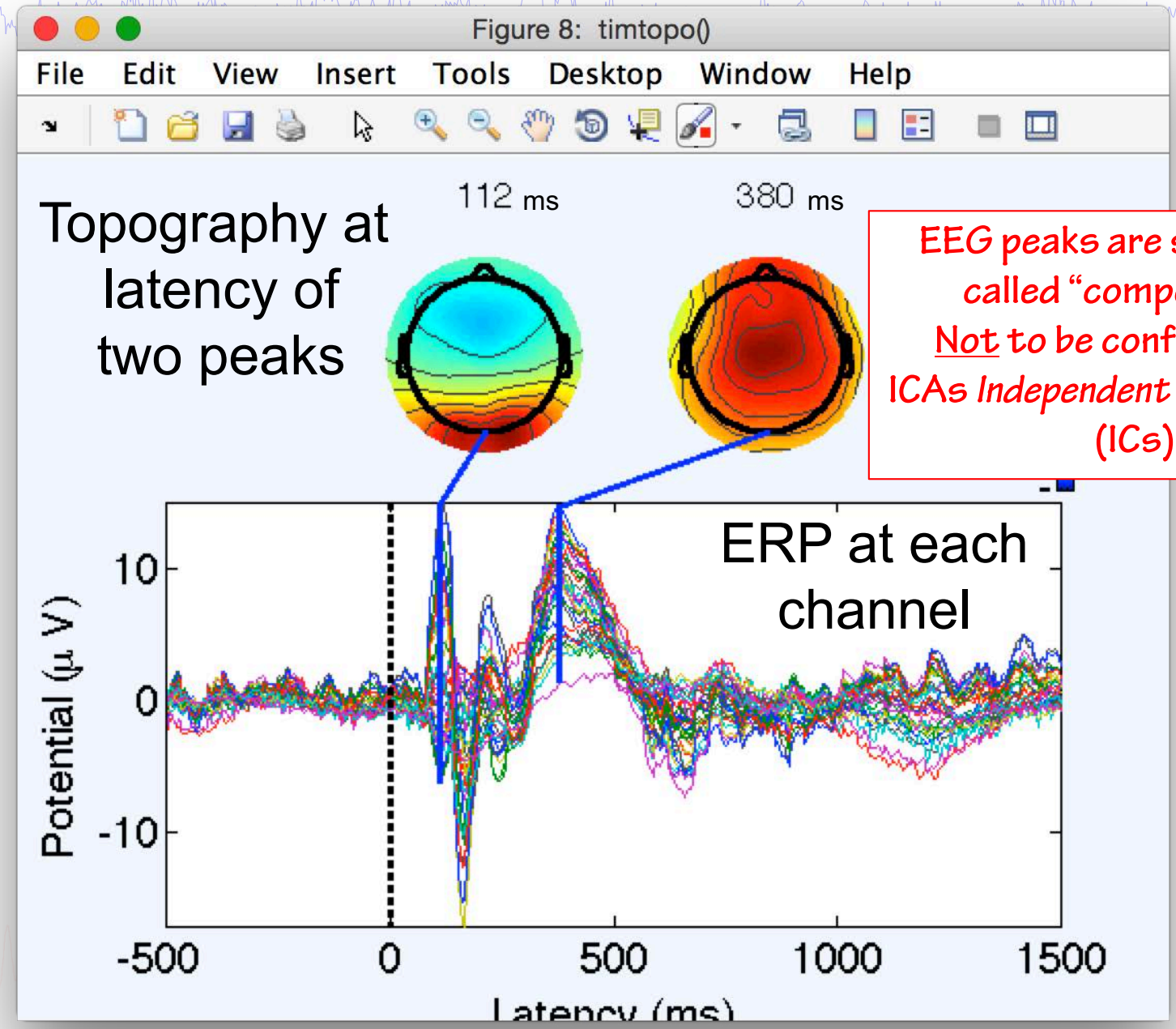
A step back: Electrode-level ERP





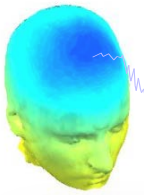
Traditional ERP: Time-locked activity at each channel

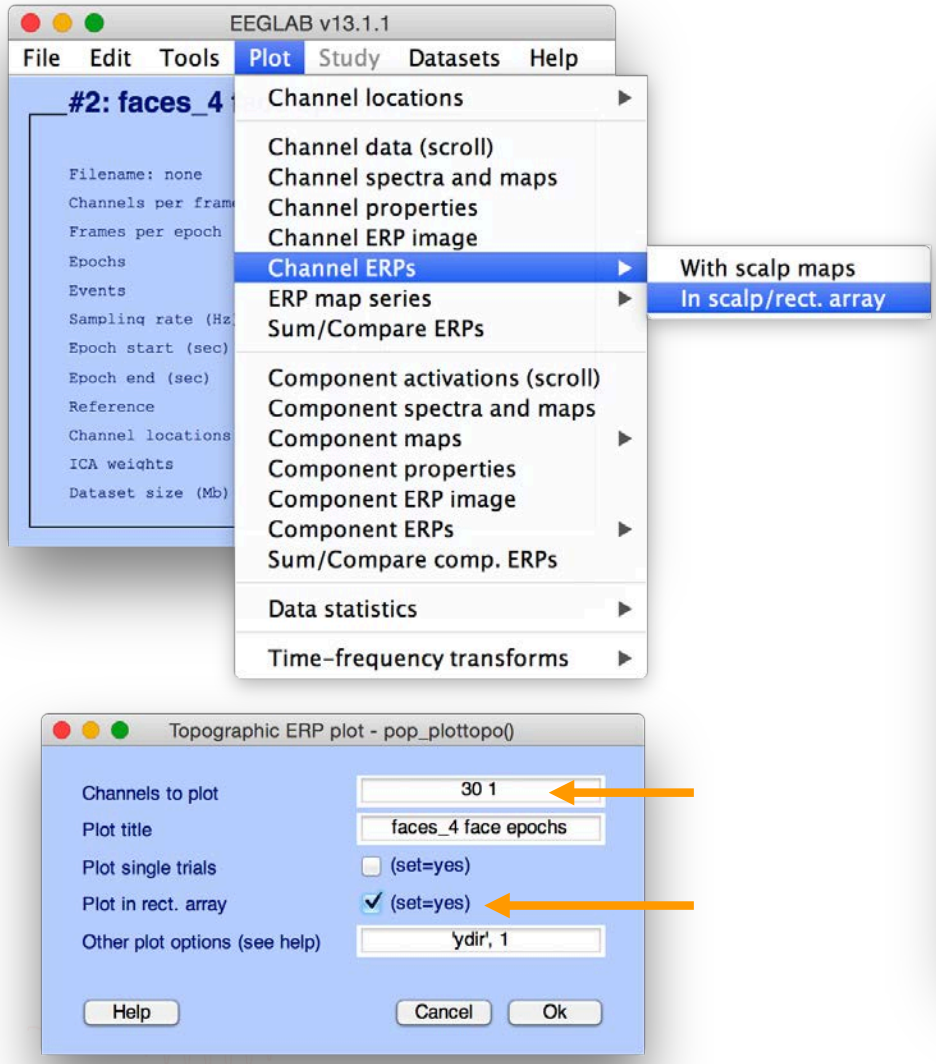


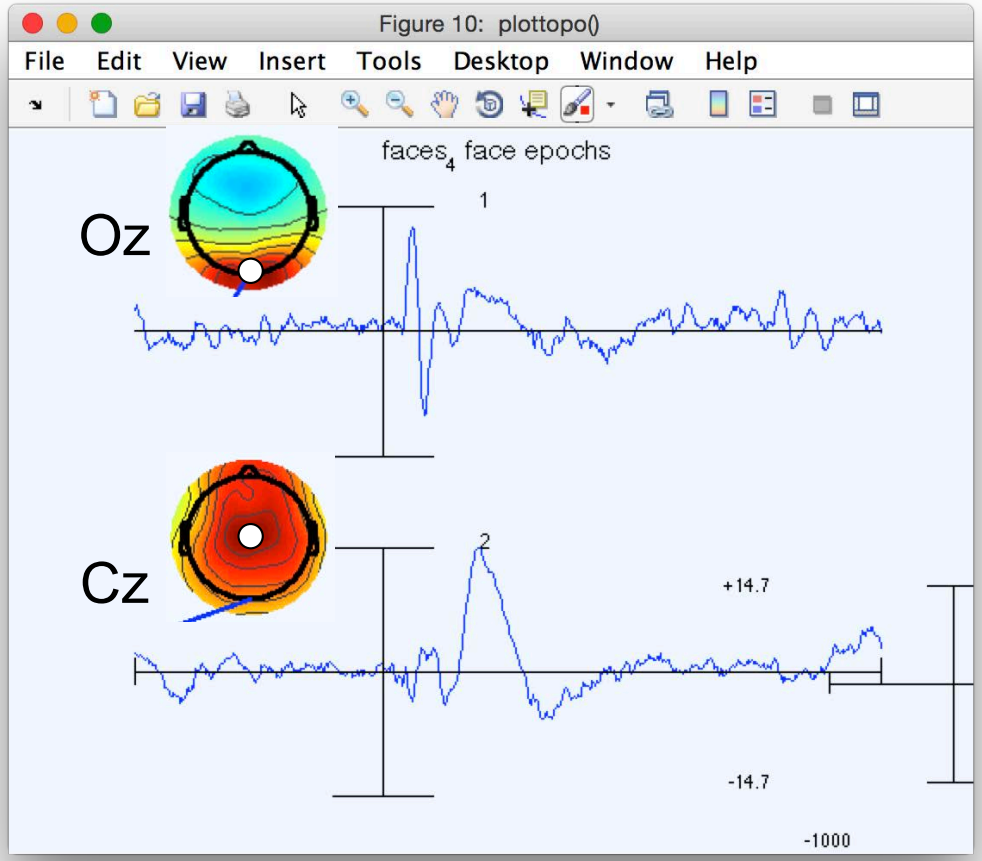


EEG peaks are sometimes called "components."
Not to be confused with ICAs *Independent Components* (ICs)!

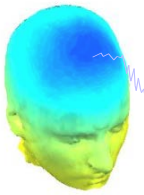
ERP at two channels

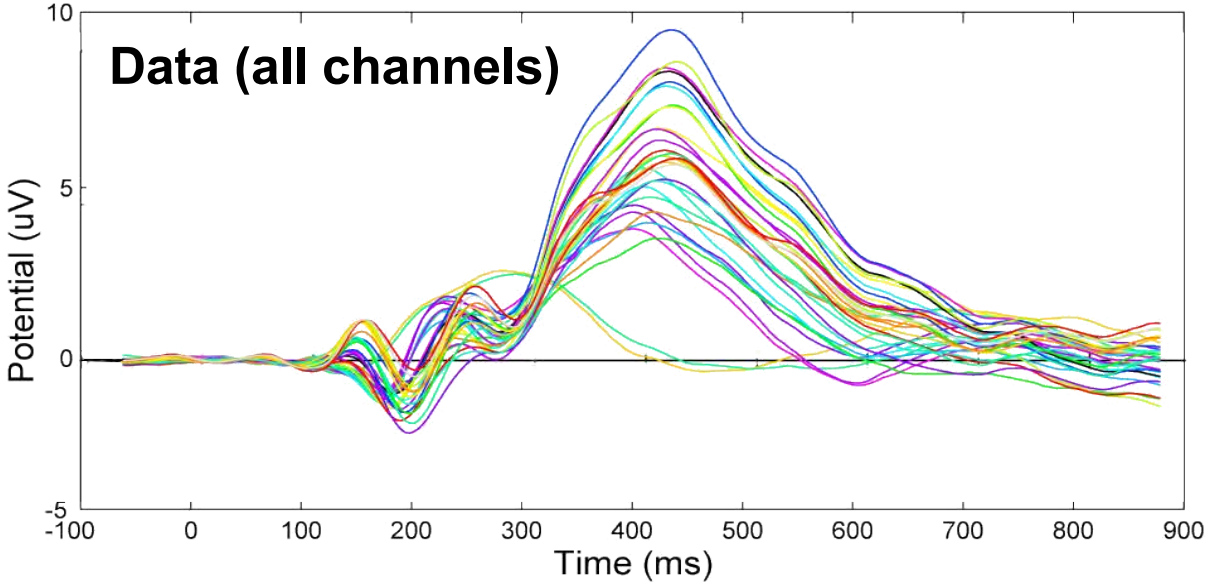


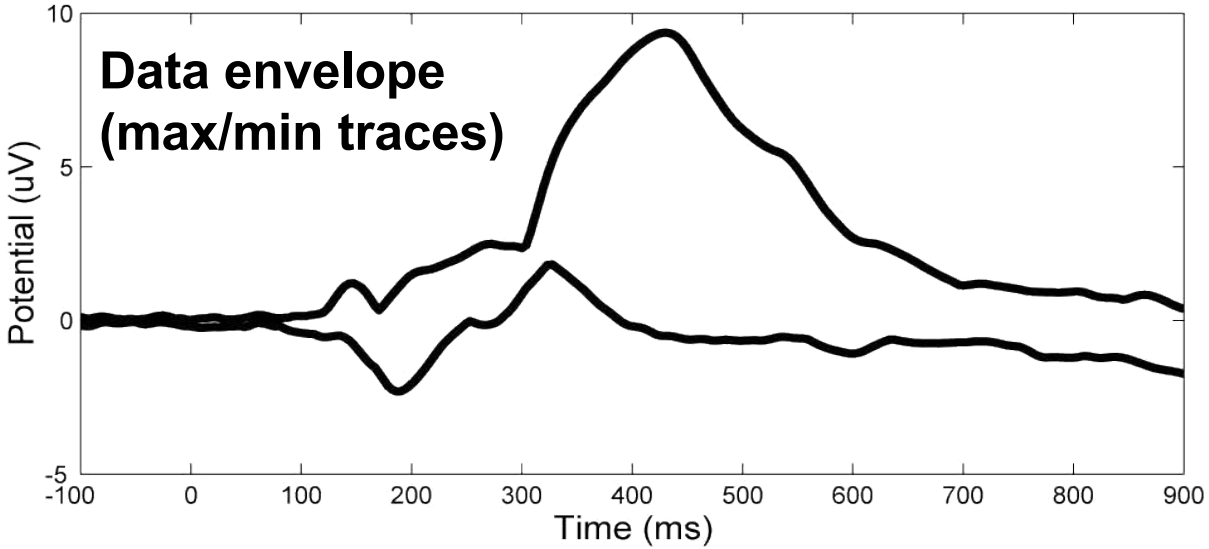




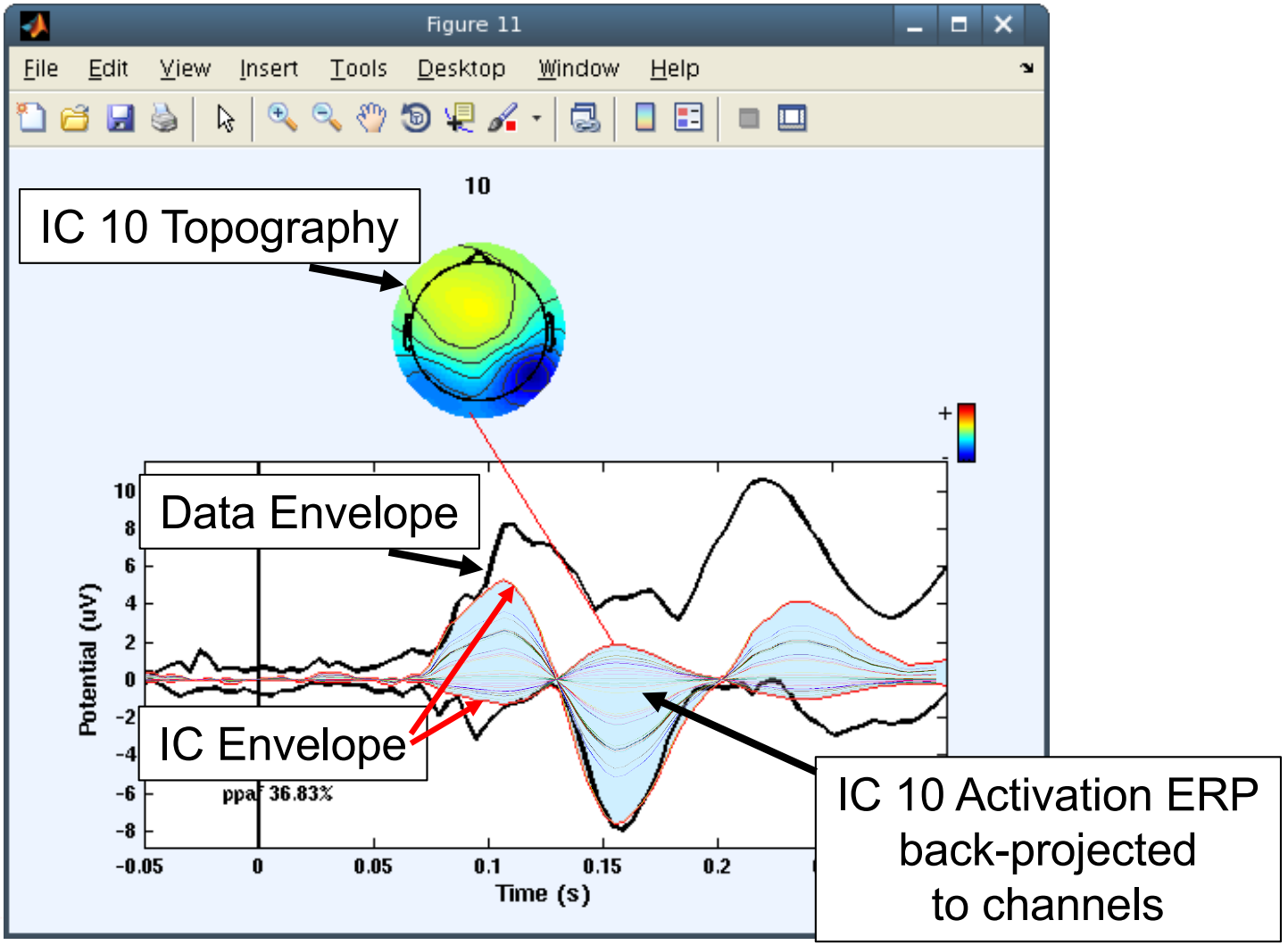
Definition: The data envelope



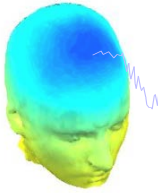


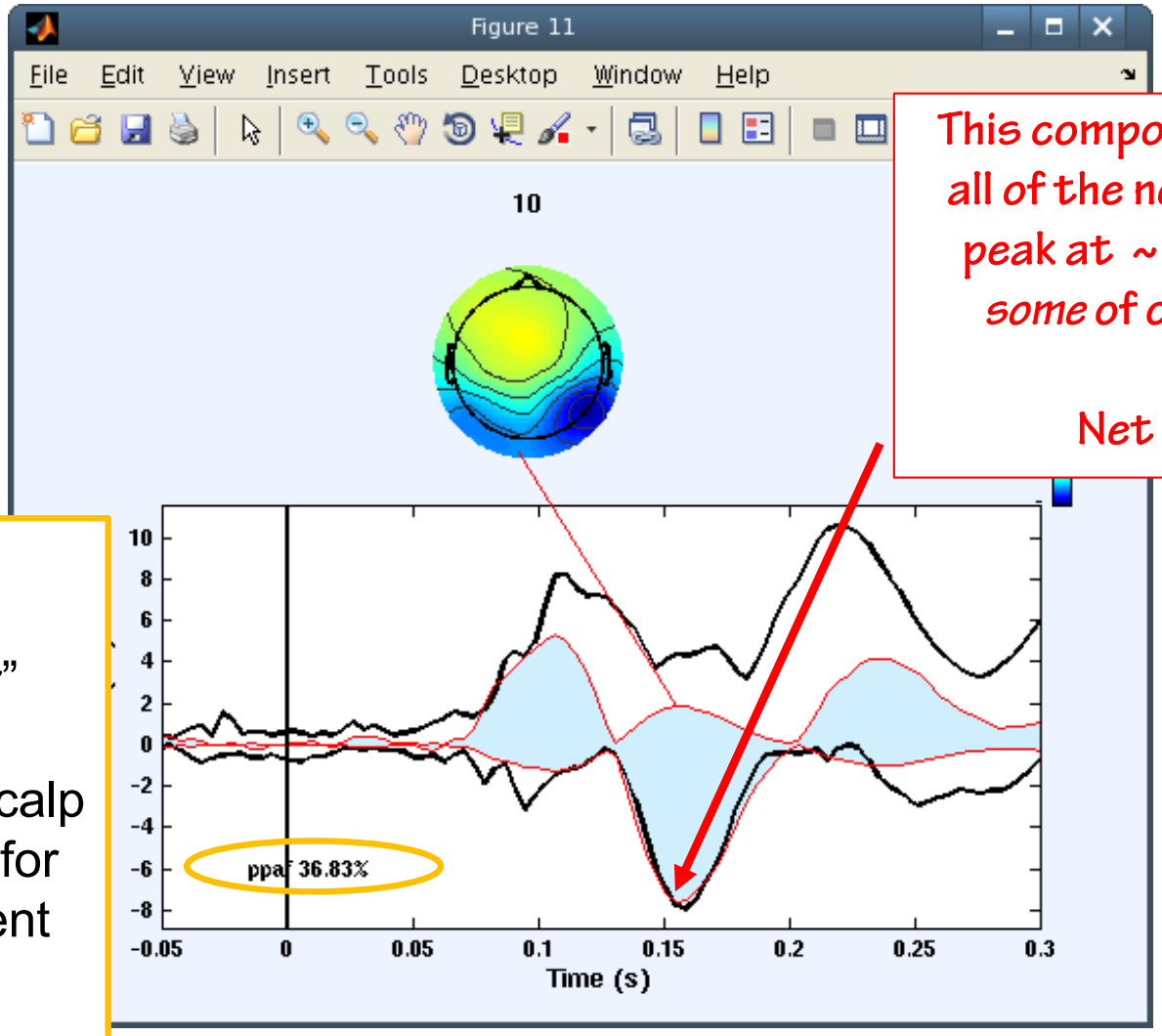


Definition: IC Envelope



Key: Scalp ERP peaks are often the sum of multiple independent source processes





This component accounts for all of the negative scalp ERP peak at ~150 ms, but *only some* of other ERP peaks

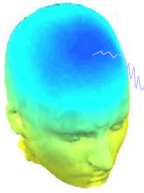
Net PVAF: 37%

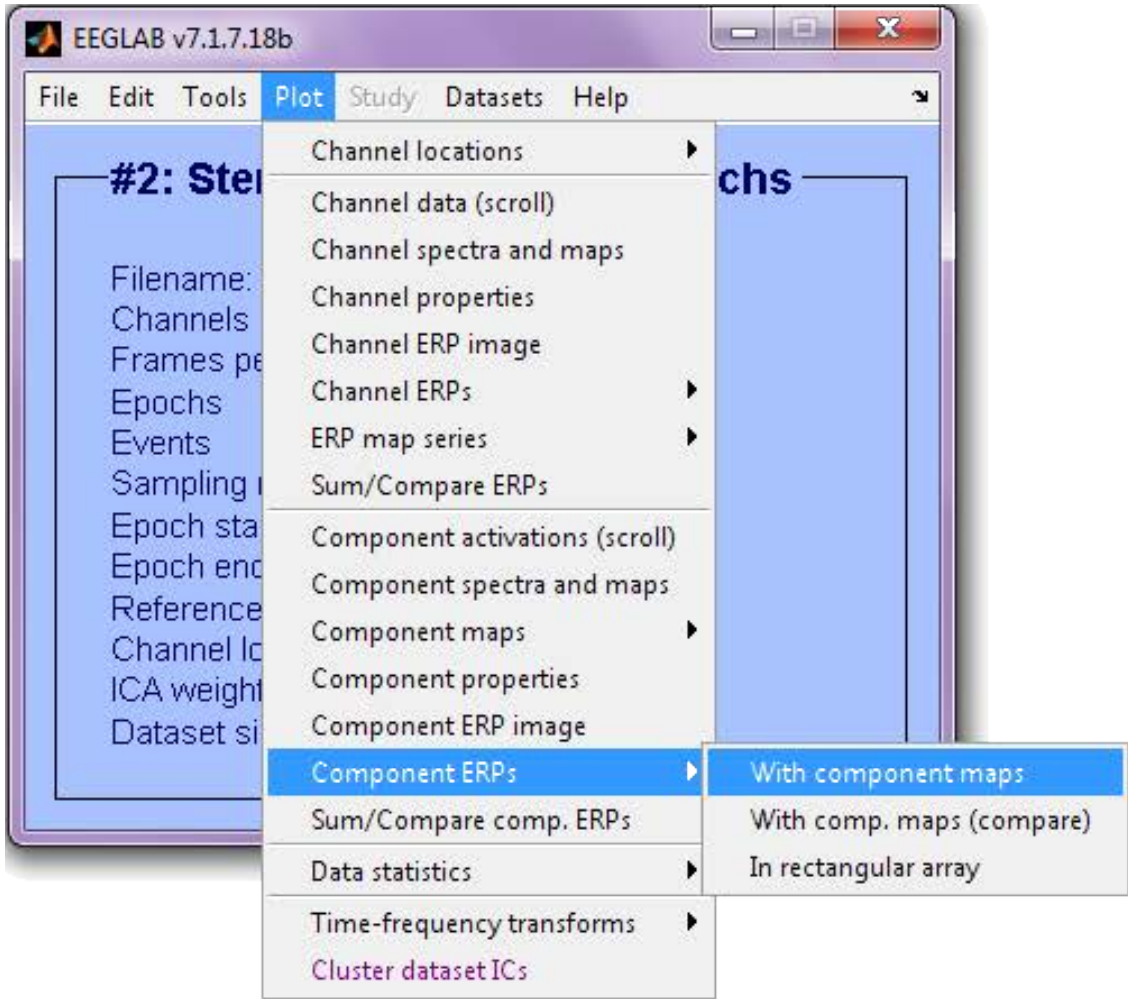
PVAF
"% Variance Accounted For"

The variance of scalp EEG accounted for by this component

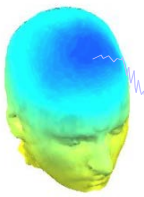
*ppaf

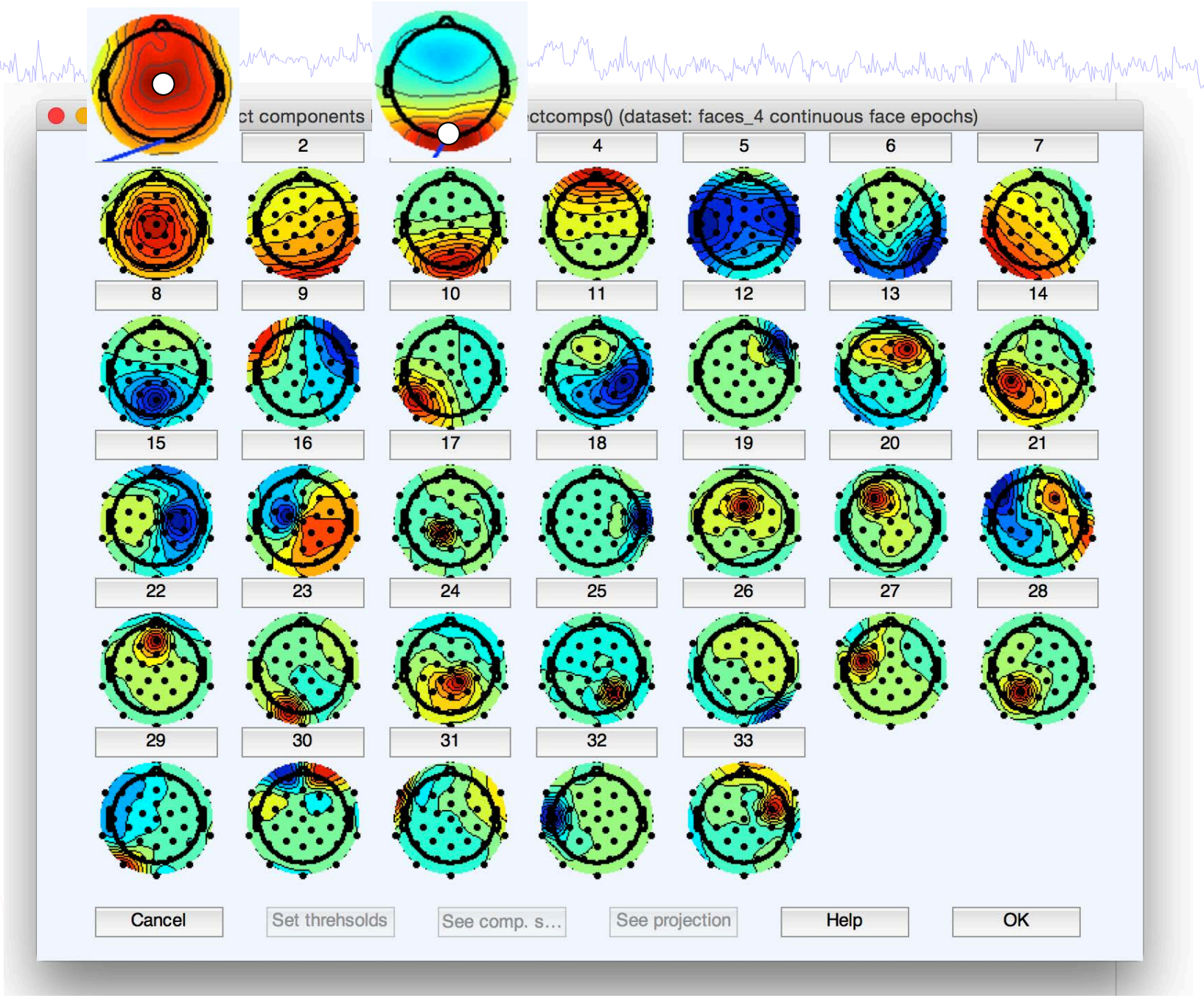
Component ERP envelope



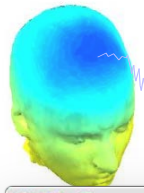


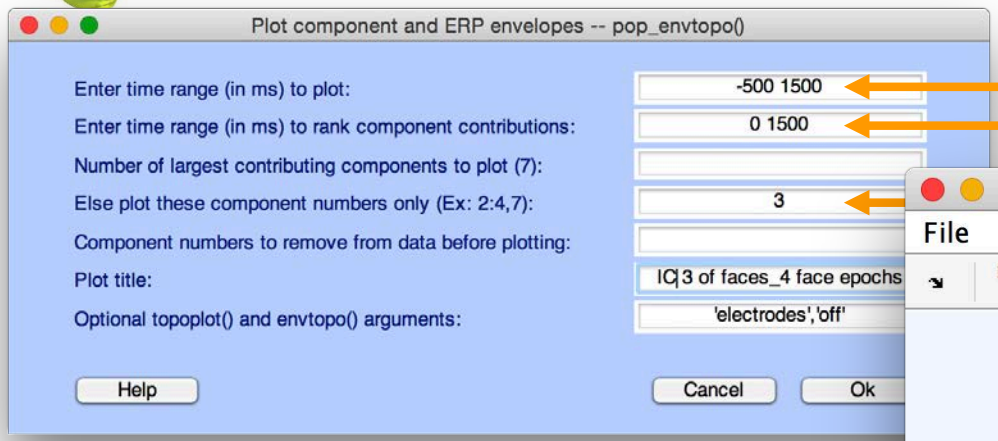
ERP peak- and IC Component-topographies

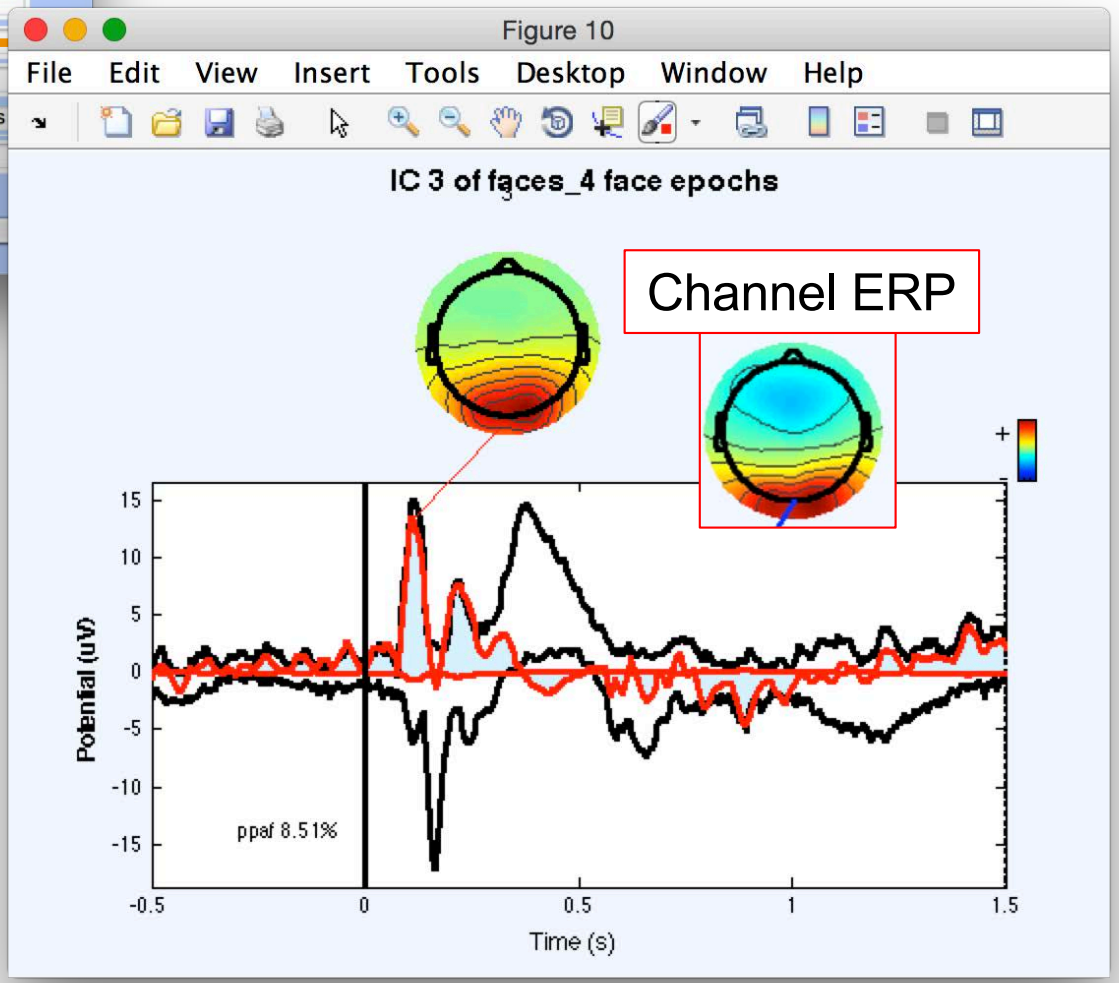




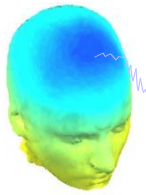
Component 3 ERP envelope

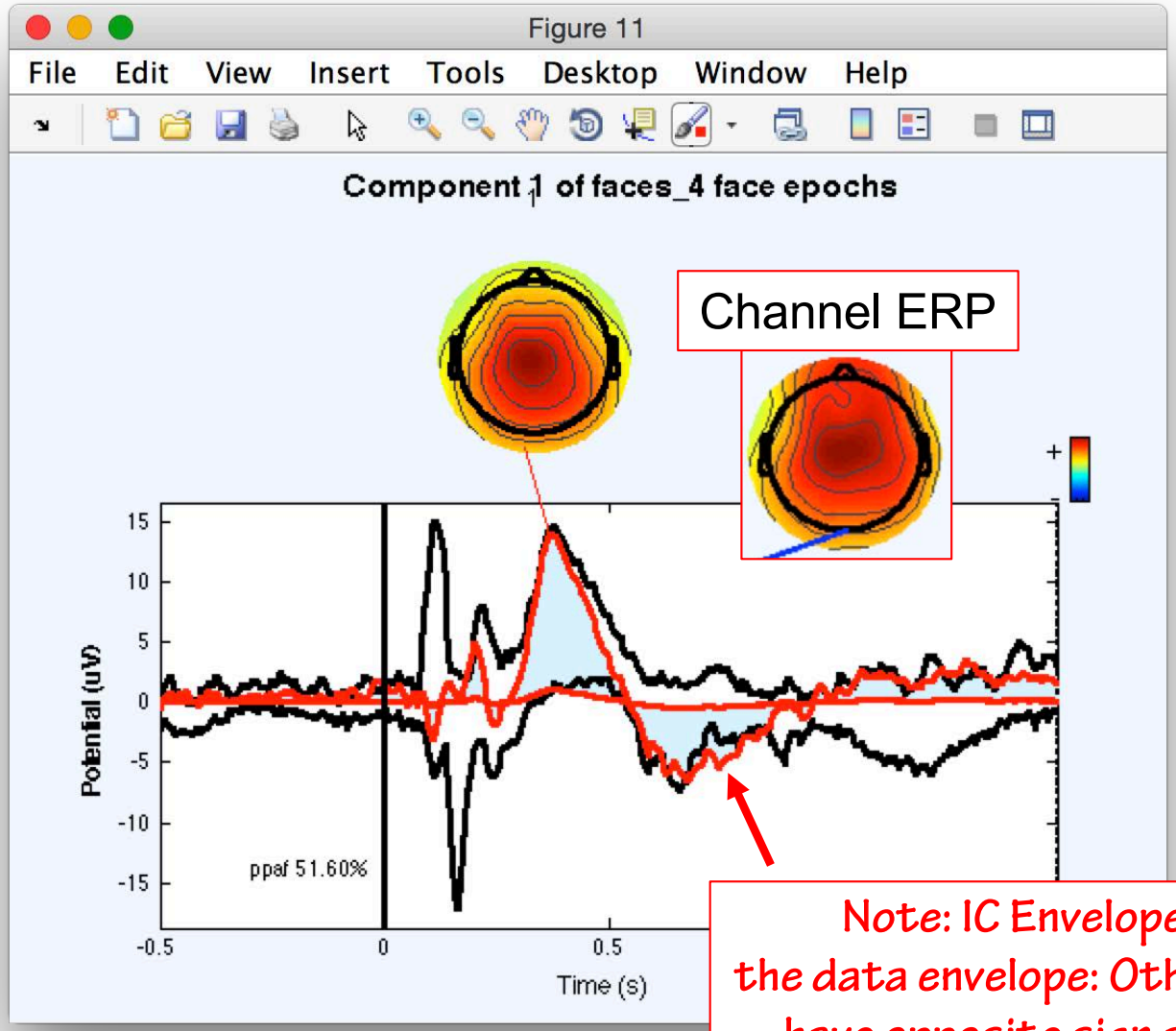






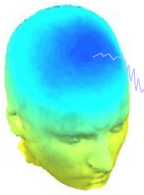
Component 1 ERP envelope



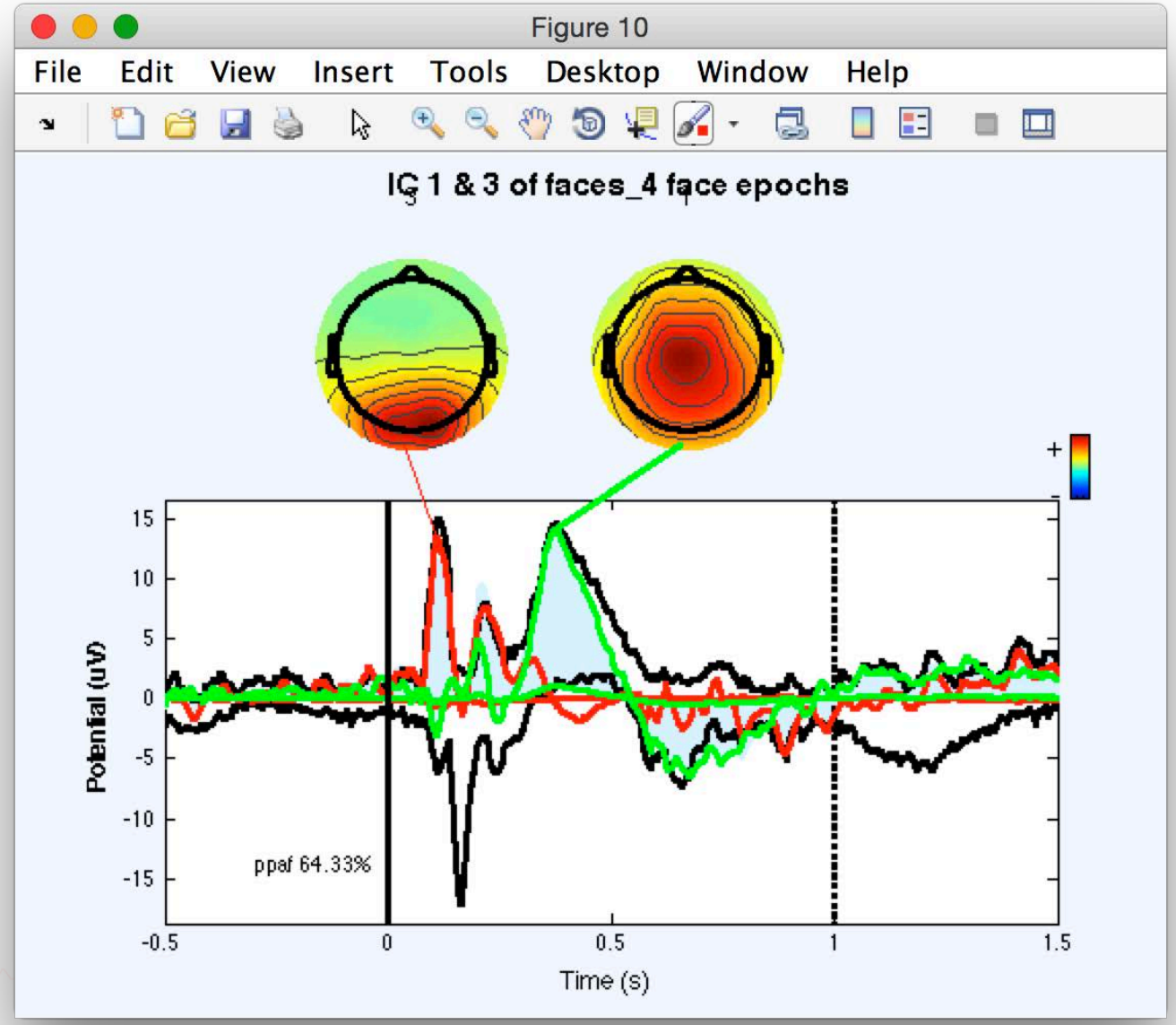


Note: IC Envelope can *exceed* the data envelope: Other component(s) have opposite sign at this latency.

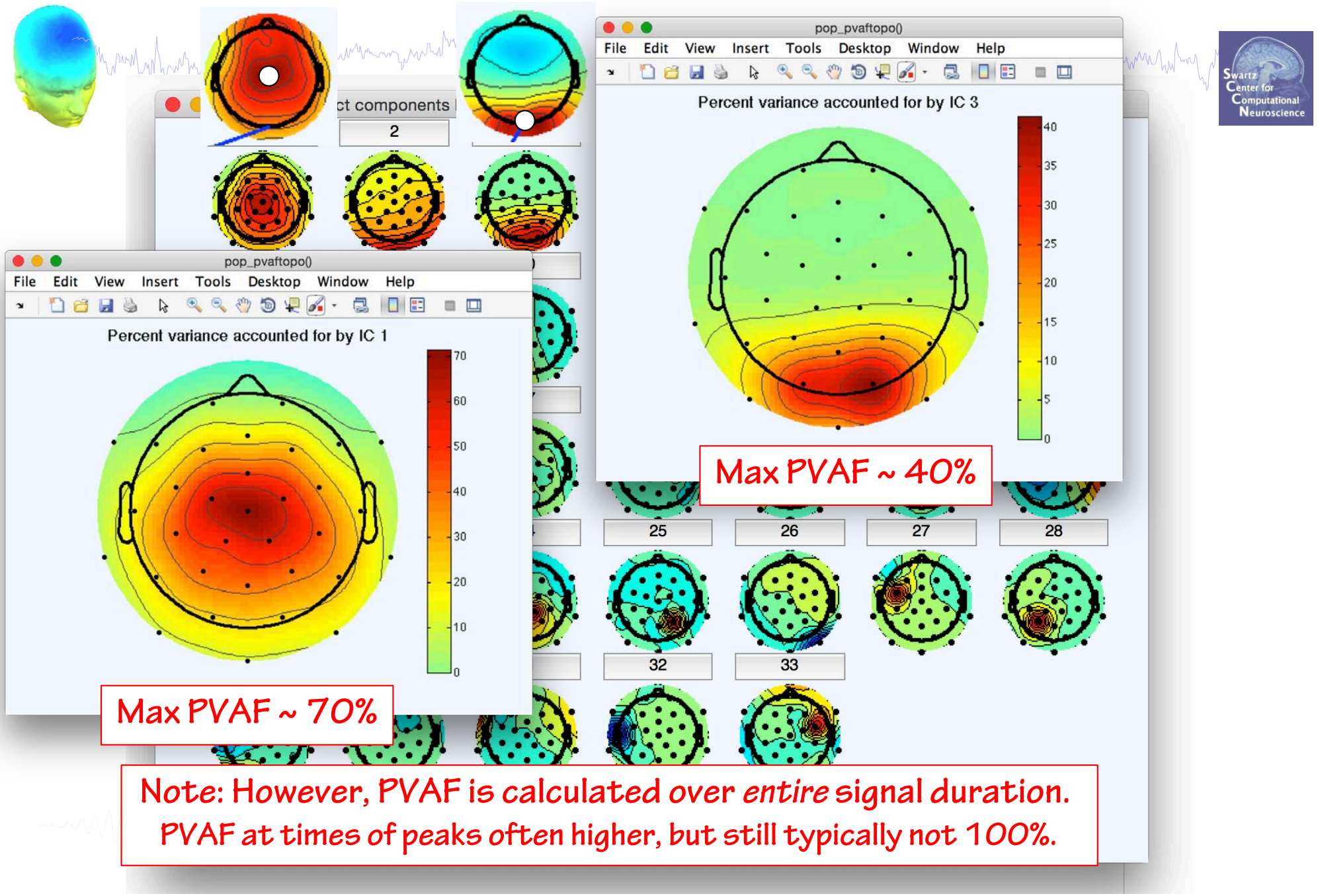
Component 1 + 3 ERP envelope



Else plot these component numbers only (Ex: 2:4,7): 1 3

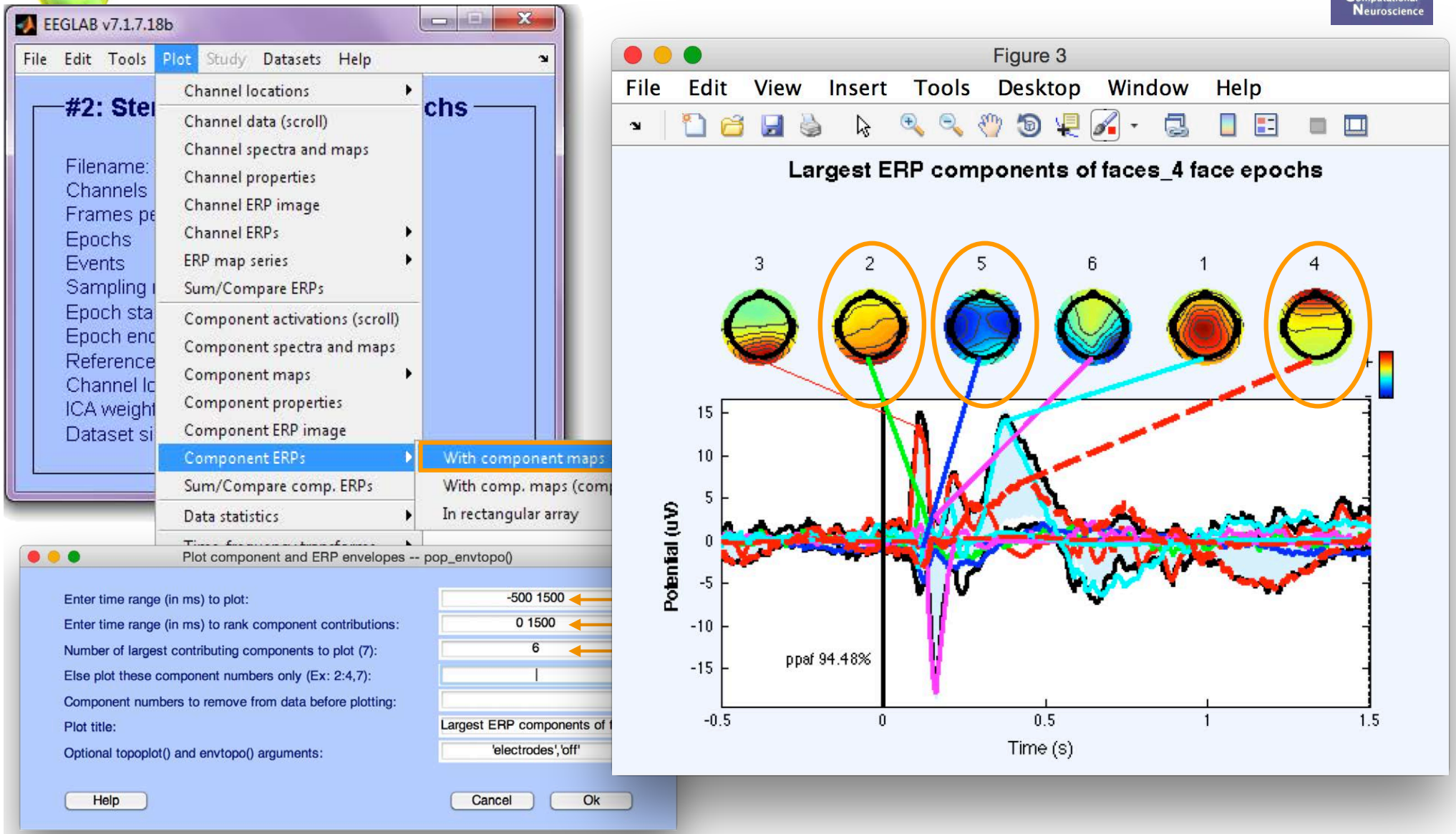


pvaftopo plugin (Makoto Miyakoshi)

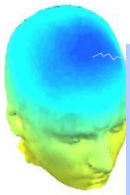


Note: However, PVAF is calculated over *entire* signal duration.
PVAF at times of peaks often higher, but still typically not 100%.

Top 6 IC contributions to data ERP envelope



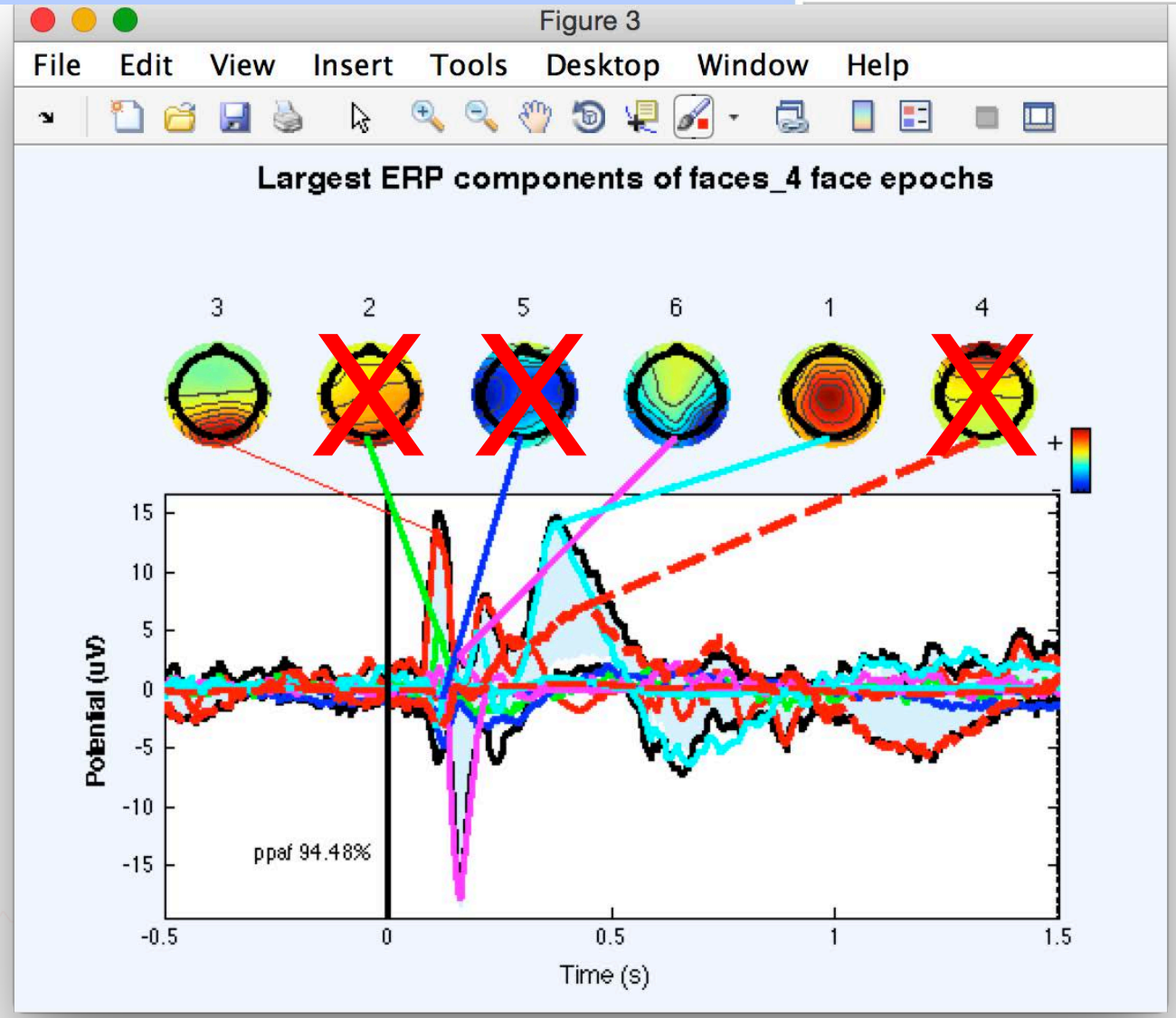
Non-artifact IC contrib. to data ERP envelope



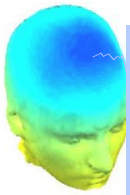
Artifact Components

Component numbers to remove from data before plotting: 2 4 5 7 9 12 17 18 25

Plot title: Largest non-artifact ERP compor



Non-artifact IC contrib. to data ERP envelope



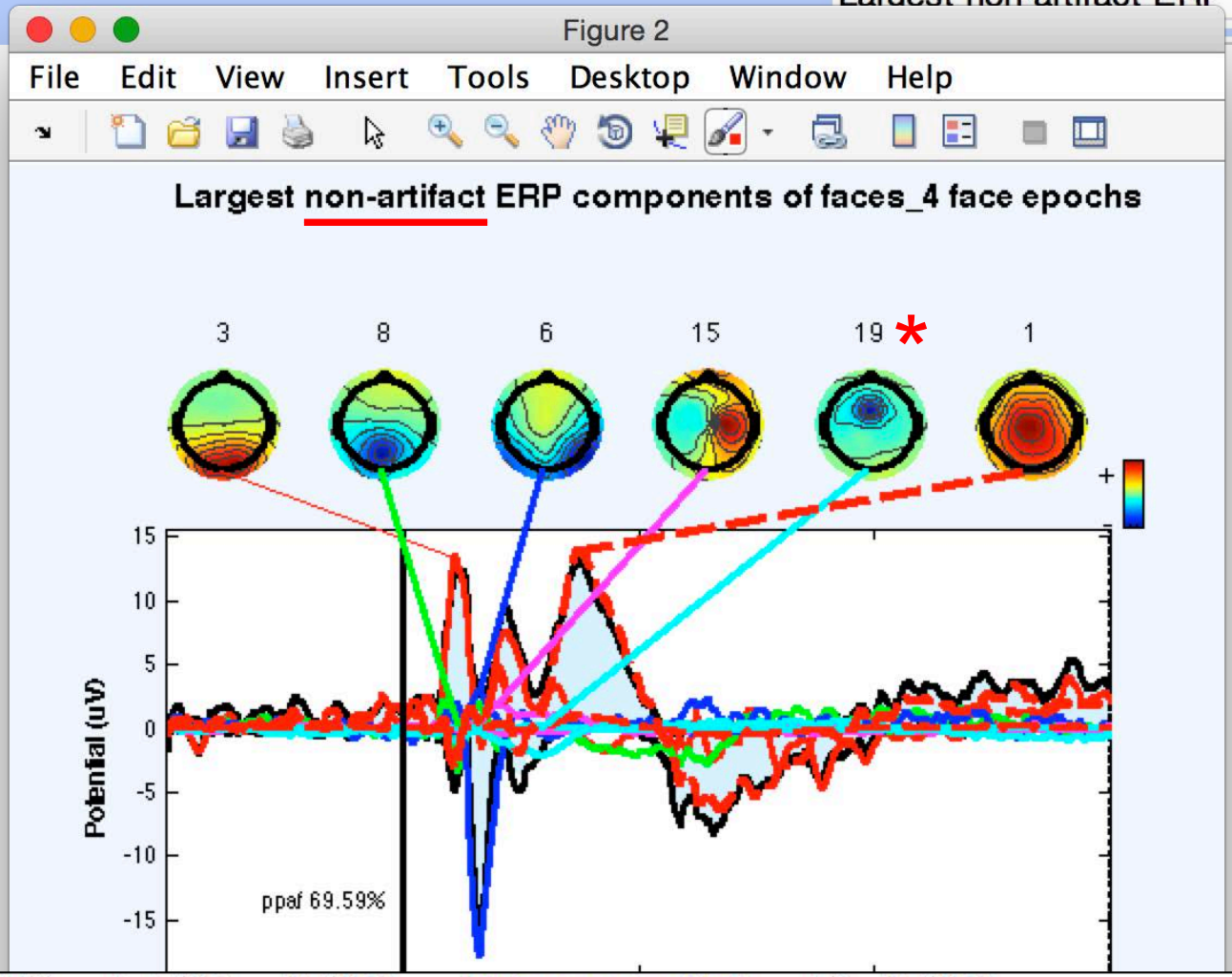
Else plot these component numbers only (Ex: 2:4,7):

Component numbers to remove from data before plotting: 2 4 5 7 9 12 17 18 25

Plot title: Largest non-artifact ERP compor

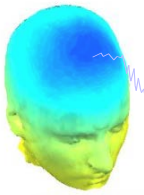
Artifact Components

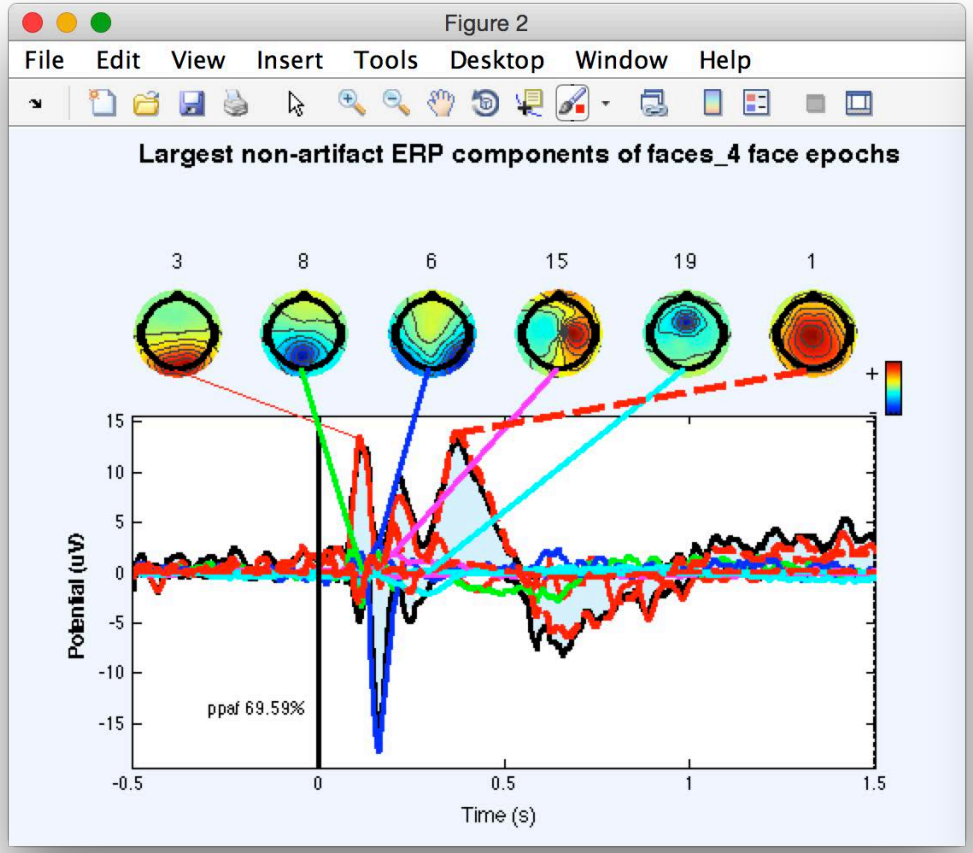
Center for Computational Neuroscience

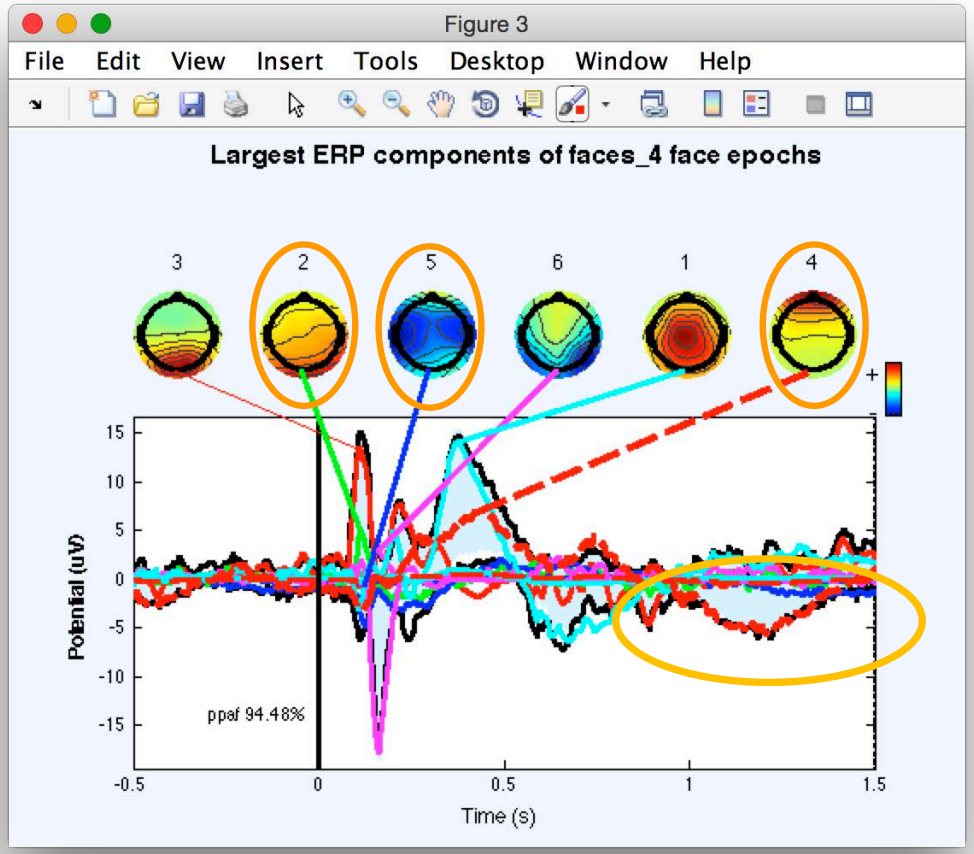



```
pop_envtopo(EEG, [-500  1500] ,'limcontrib', [0 1500], ...
'compsplot',[6],'subcomps',[2 4 5 7 9 12 17 18 25], ...
'title', 'Largest non-artifact ERP components of faces_4 face epochs',...
'electrodes','off');
```

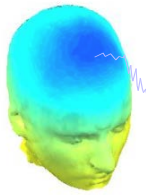
Compare: Effect of removing artifacts



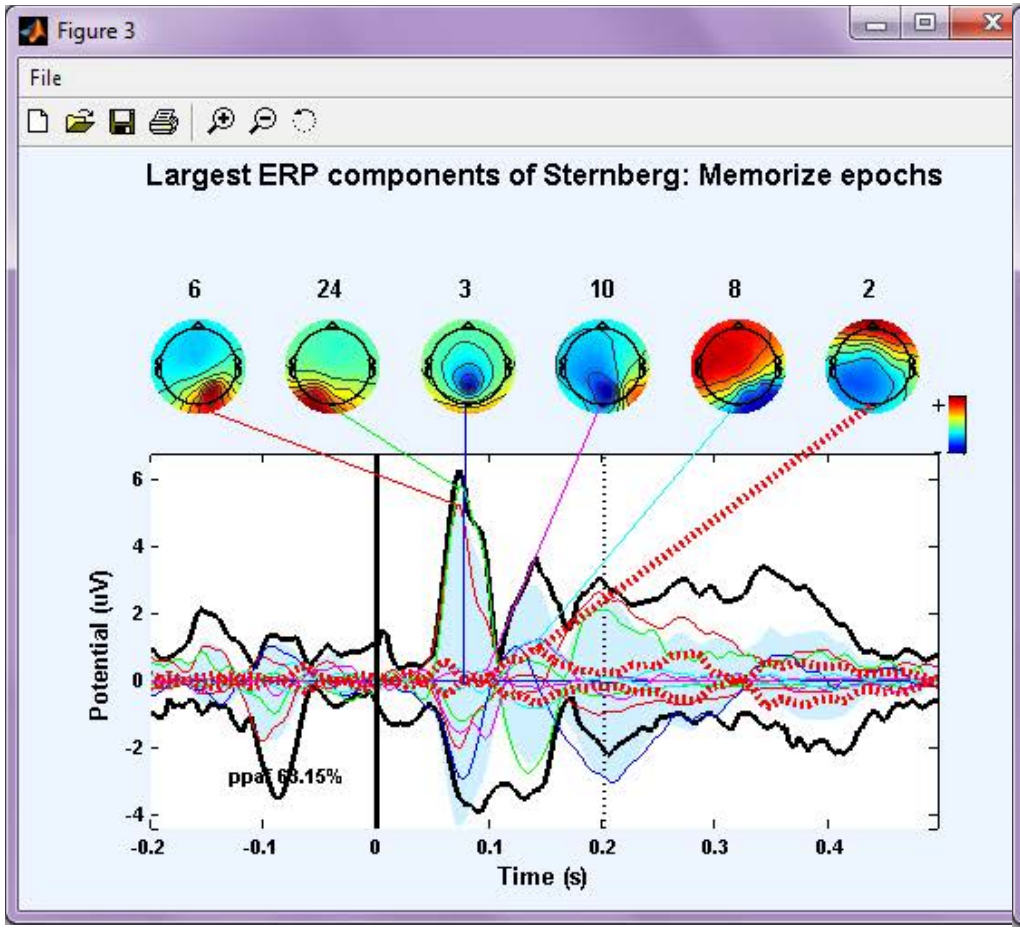


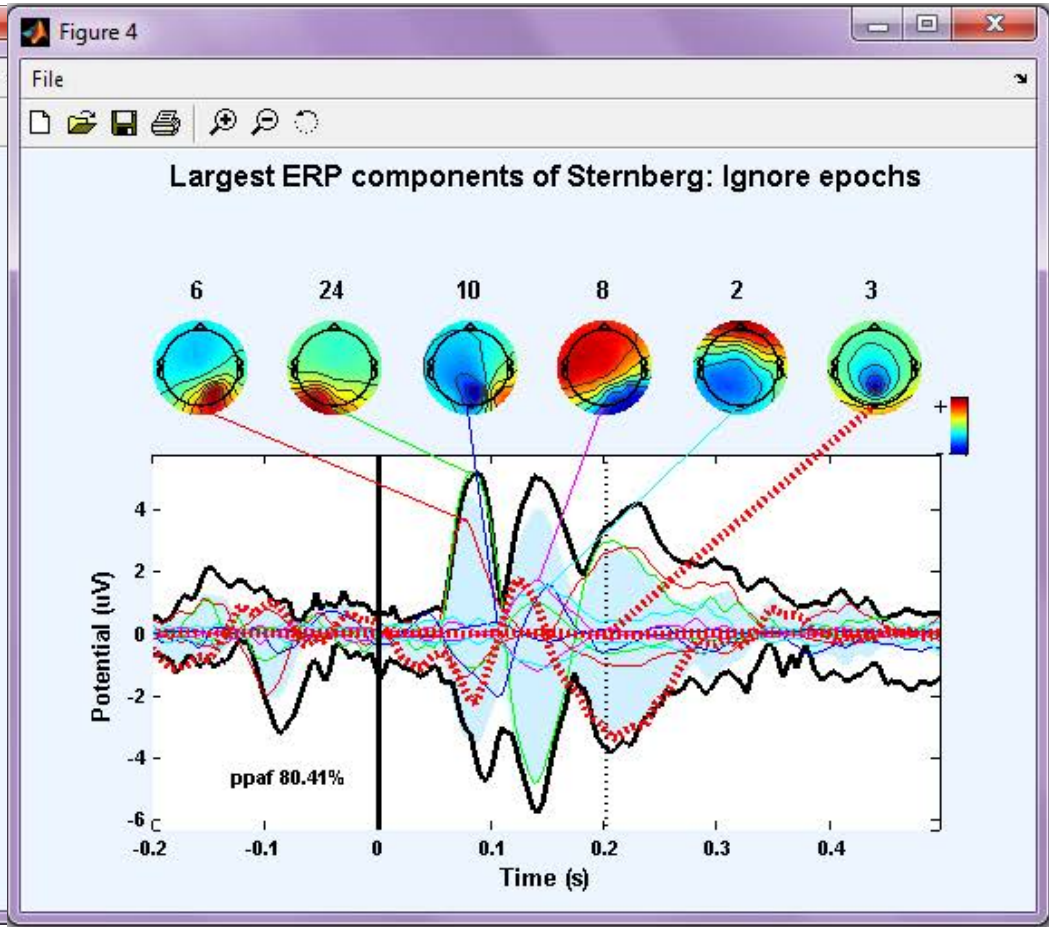


IC ERP difference



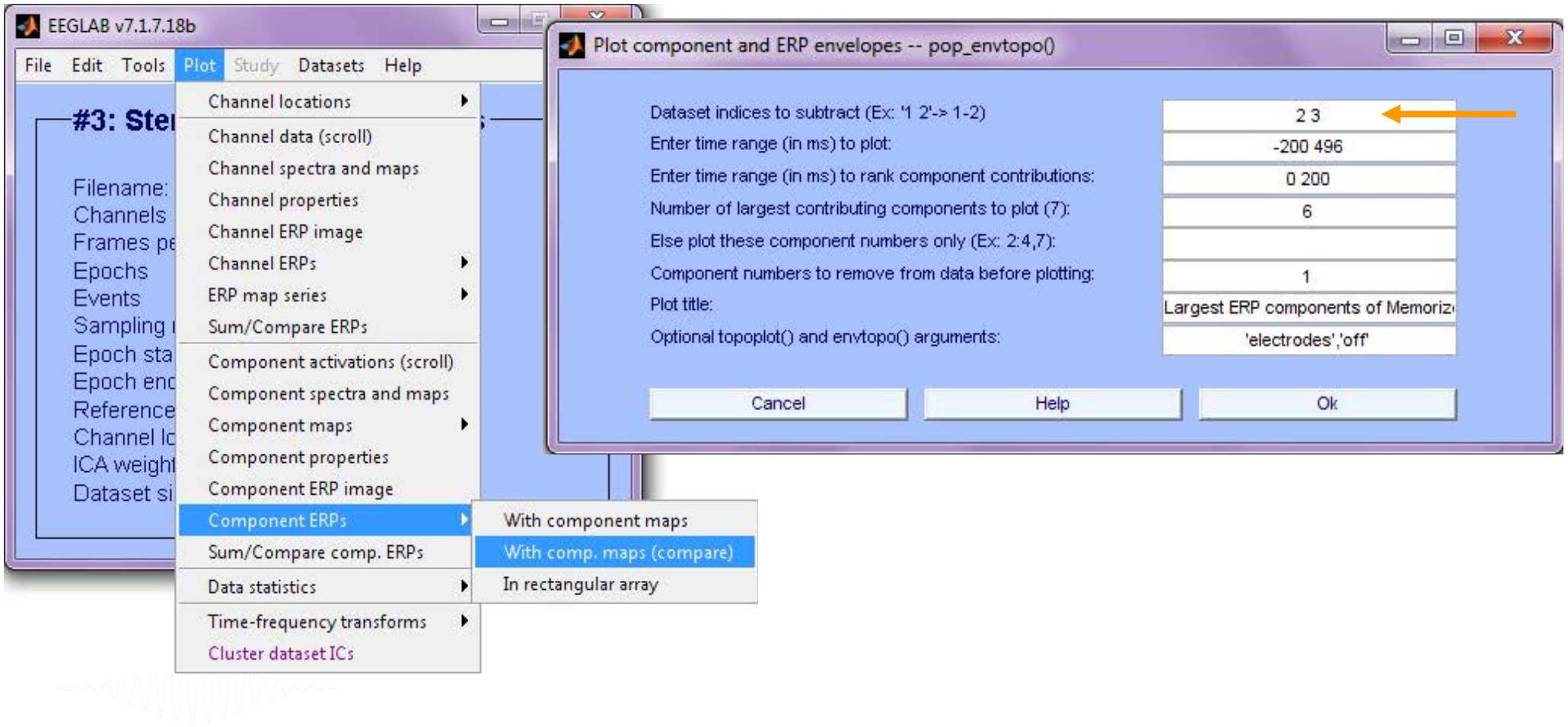
What is the IC ERP difference between these 2 conditions?



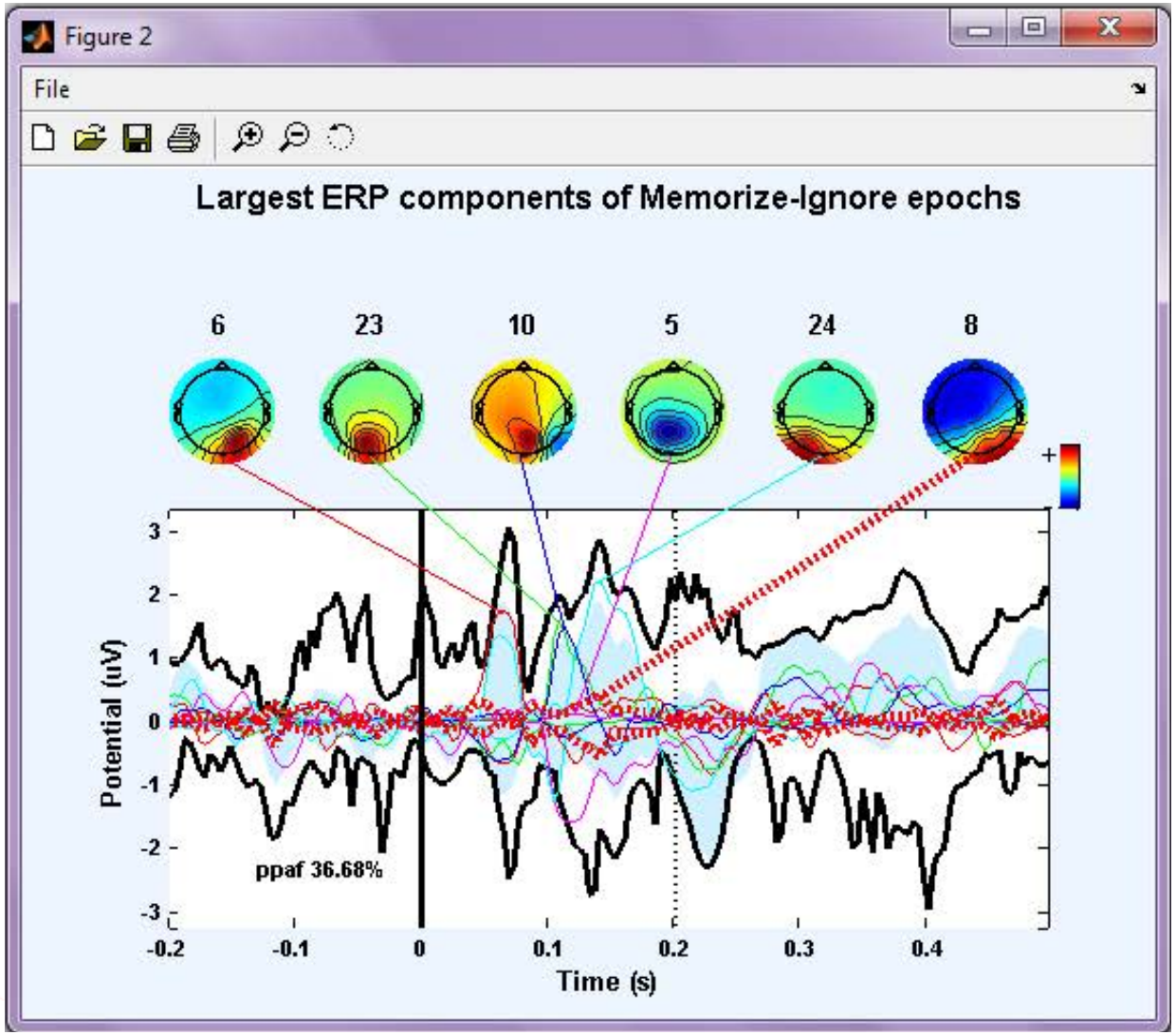


(Data: stern_125Hz.set)

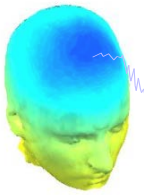
IC ERP difference



IC ERP difference



Now what…?



Part 1

Getting an overview of your ICs

Part 2

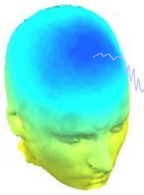
Classifying/Evaluating ICs

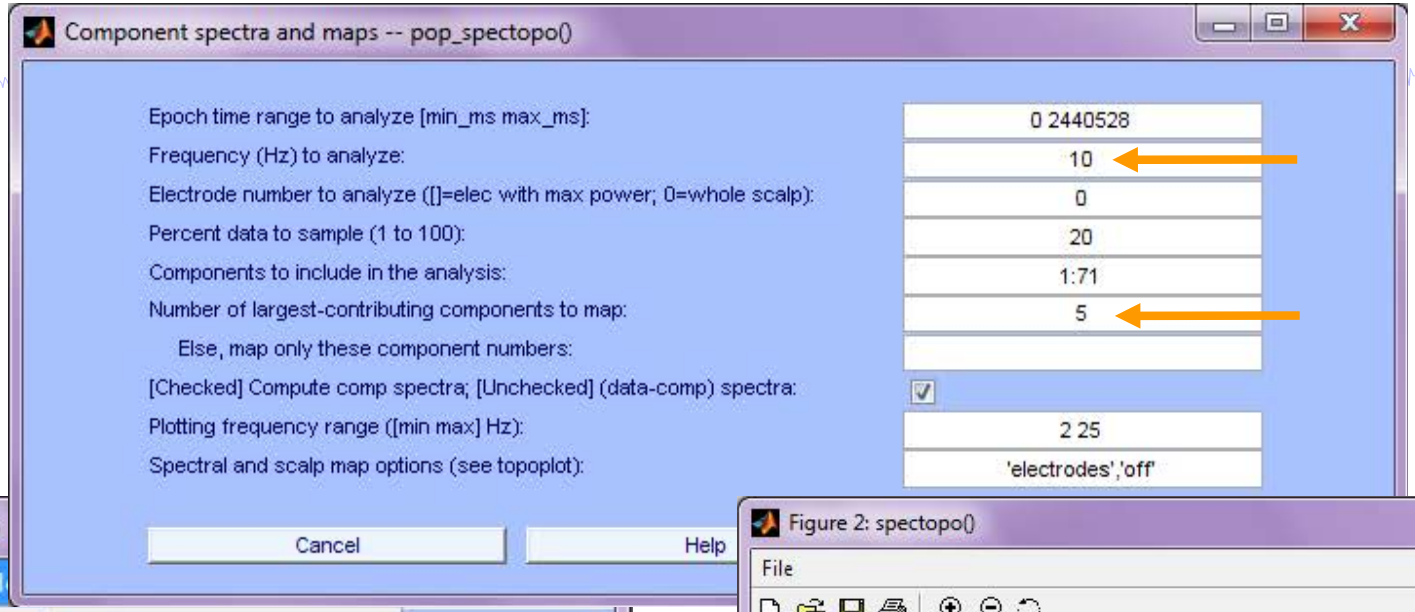
Part 3

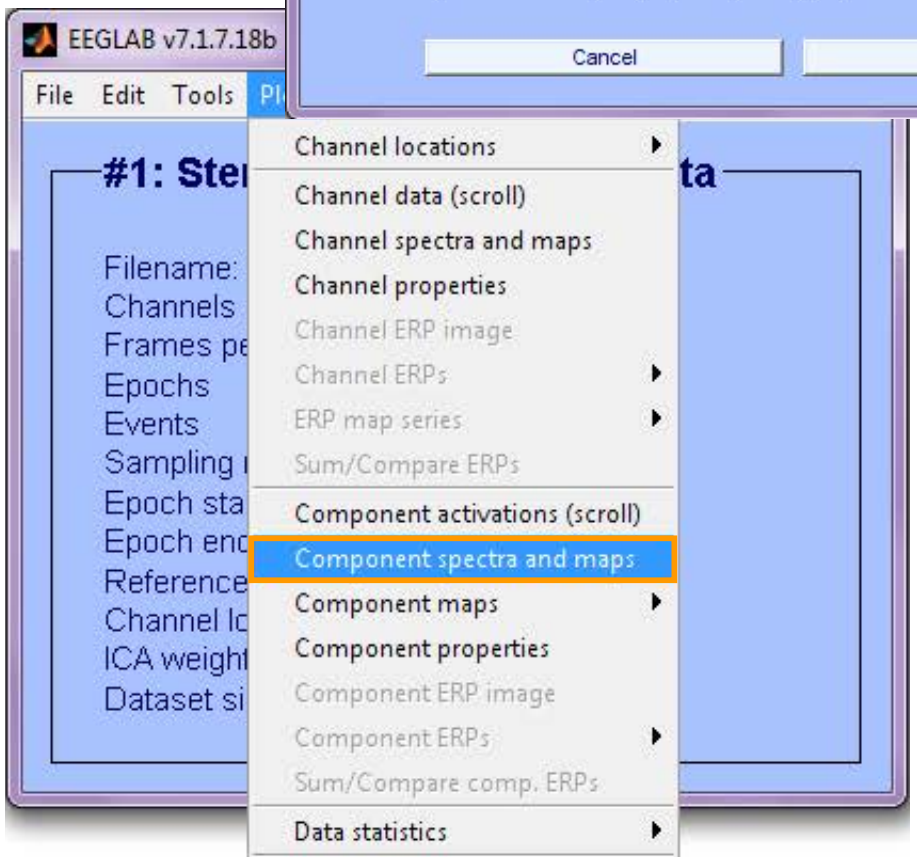
Detailed look at IC properties

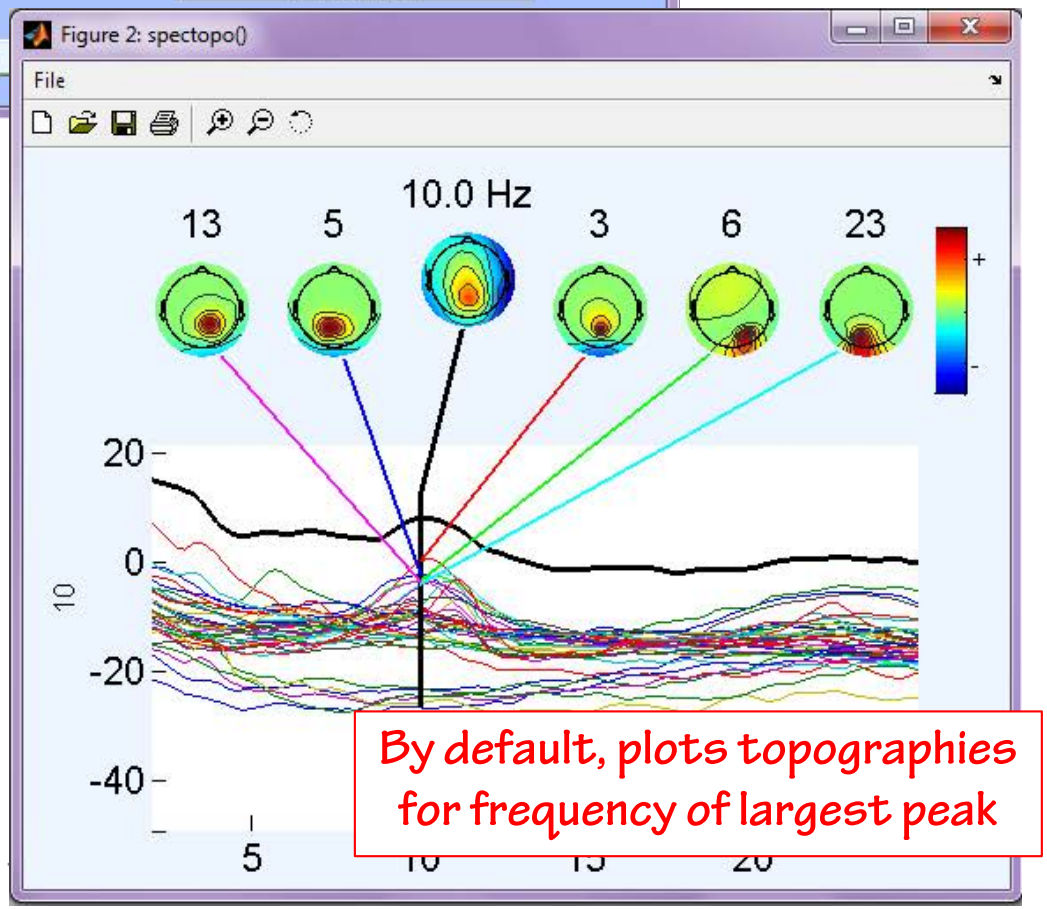
- ERP
- Spectrum (see next lecture)
- ERP images
- ERSP

Plot component power spectrum



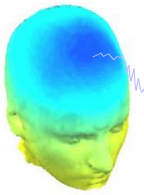


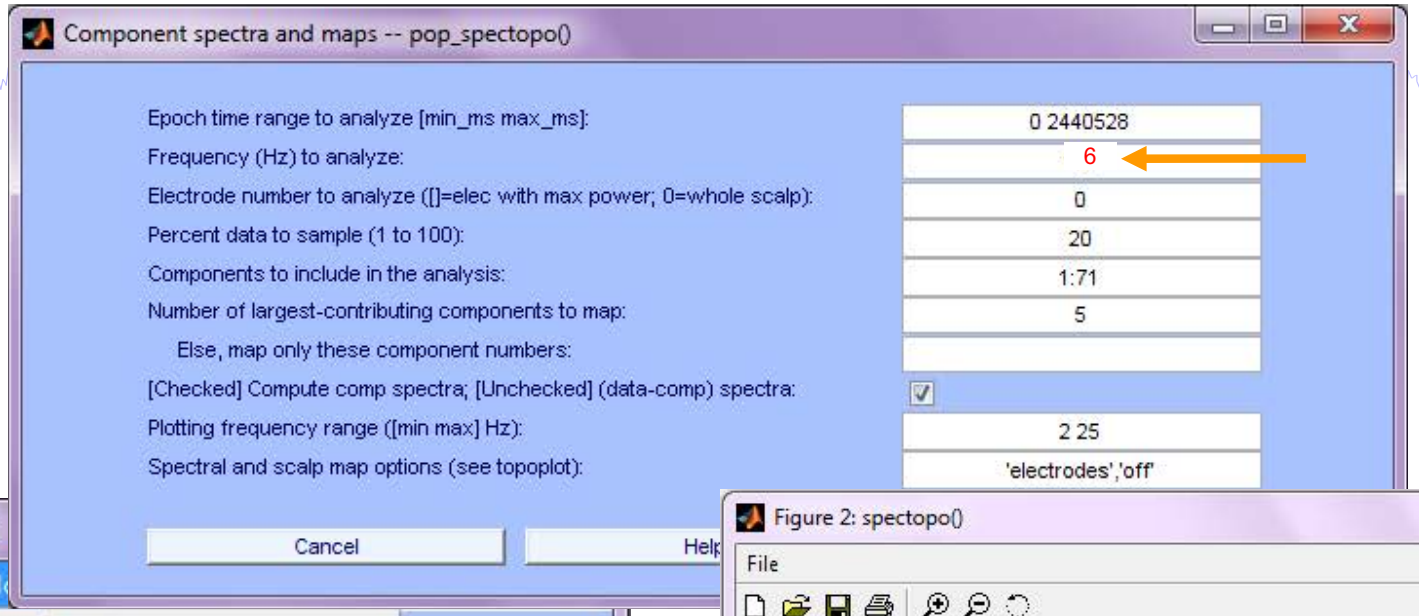


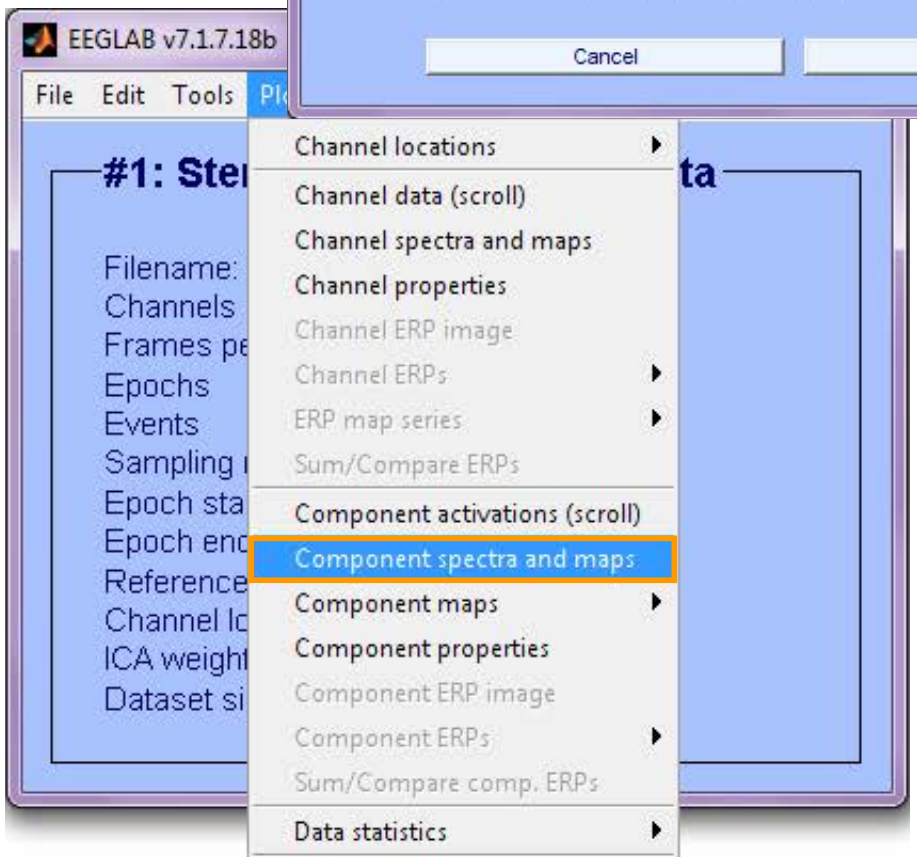


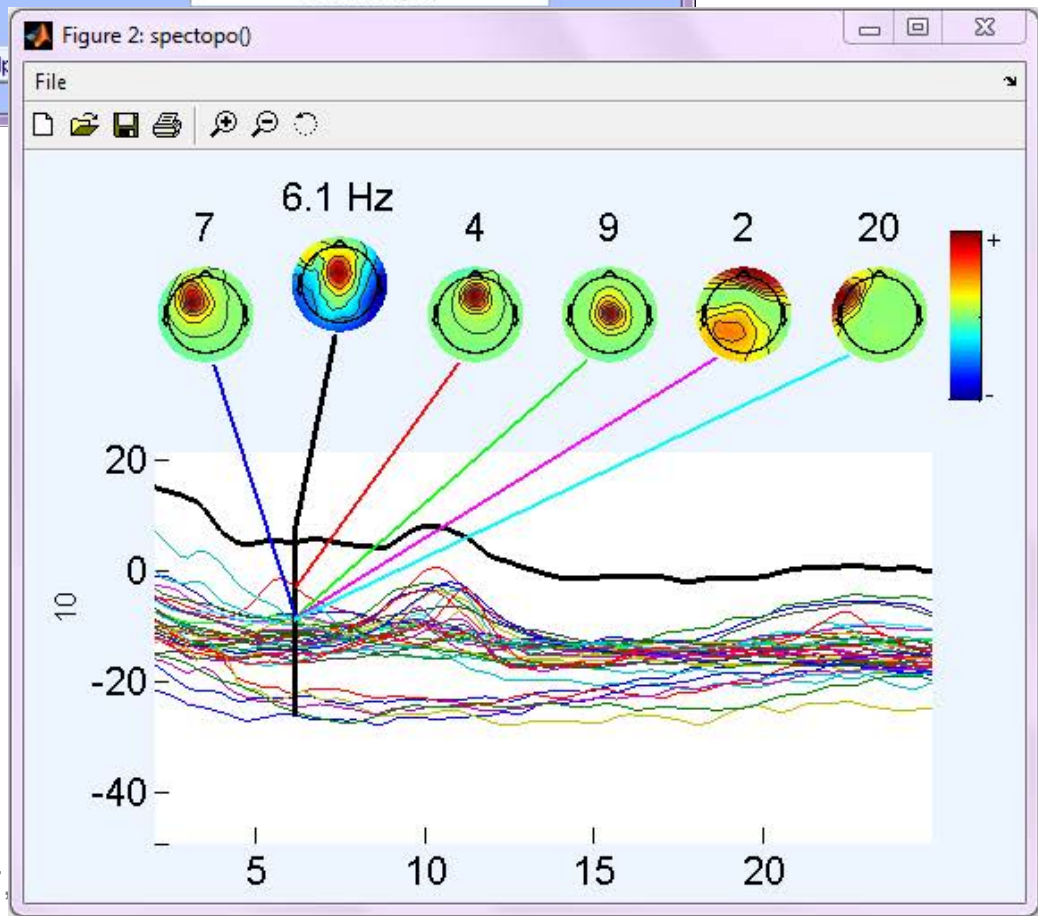
By default, plots topographies for frequency of largest peak

Select the frequency for topographies

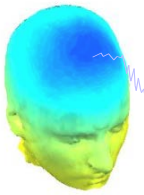








Now what…?



Part 1

Getting an overview of your ICs

Part 2

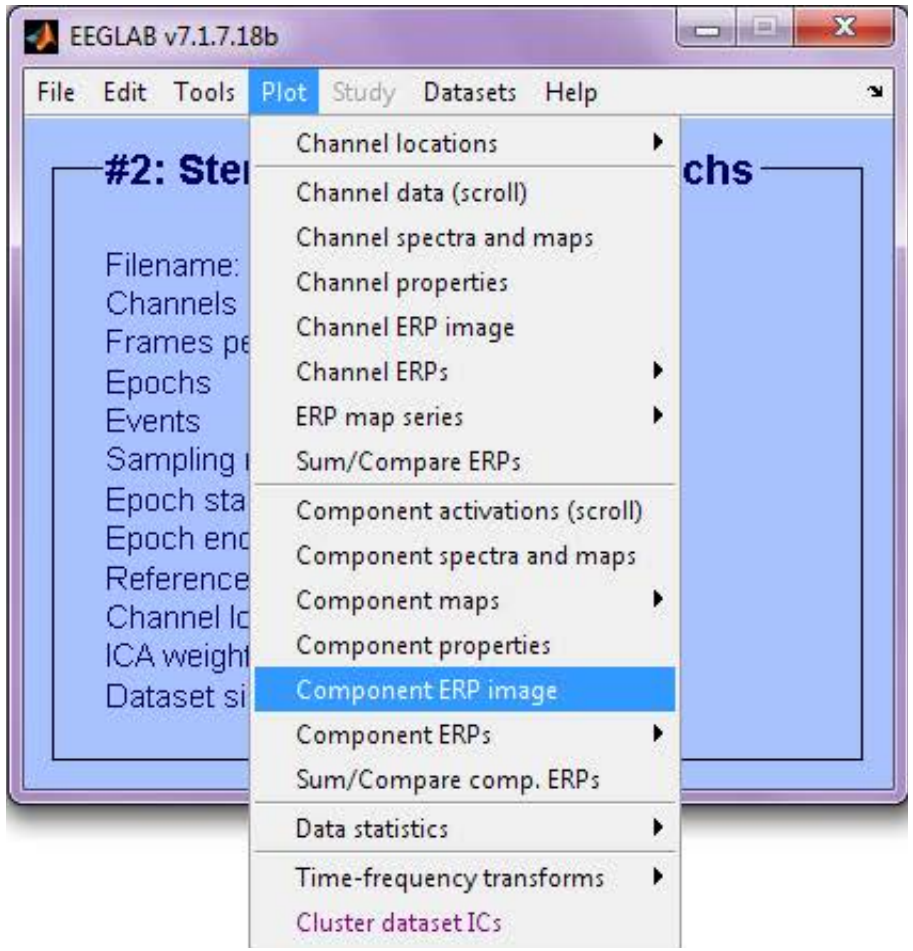
Classifying/Evaluating ICs

Part 3

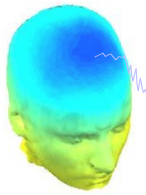
Detailed look at IC properties

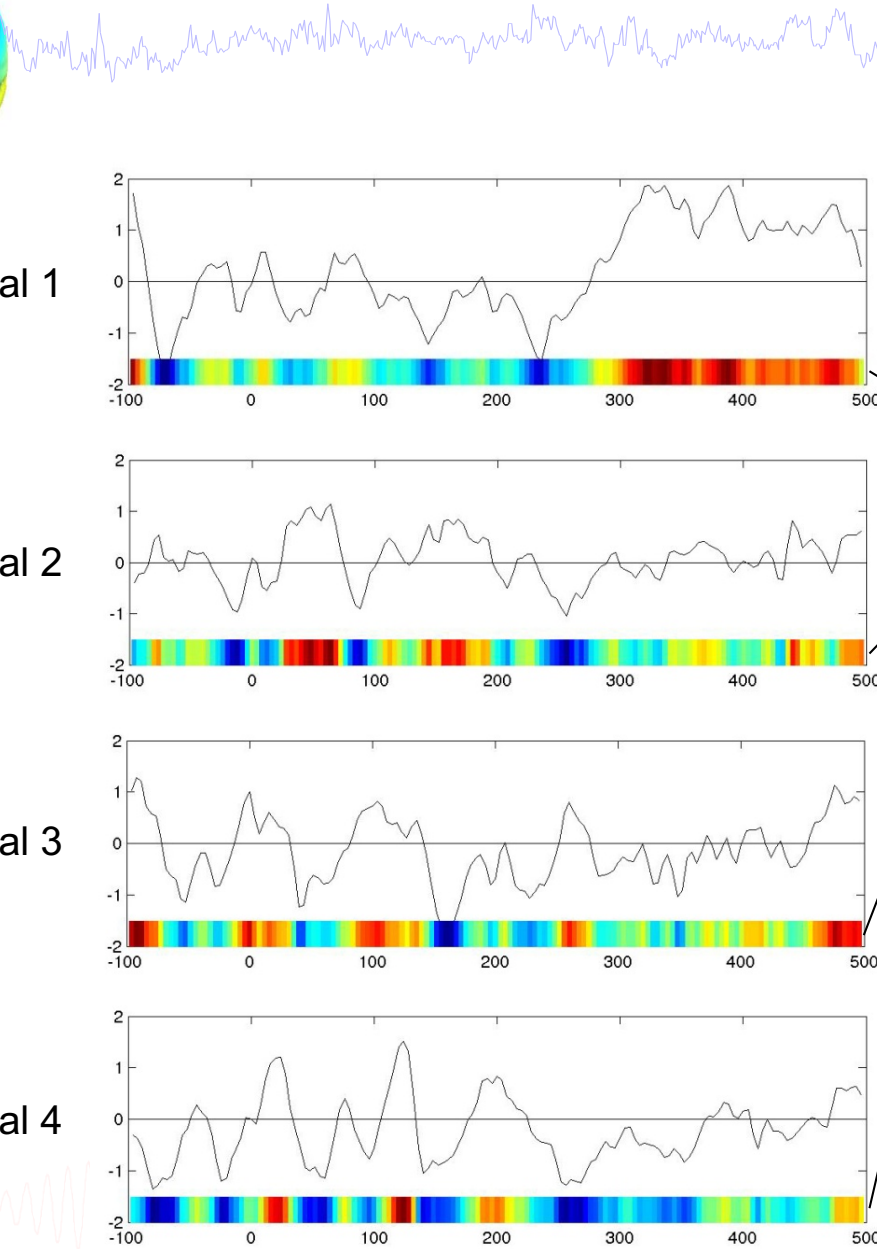
- ERP
- Spectrum
- ERP images
- ERSP

Component ERP image

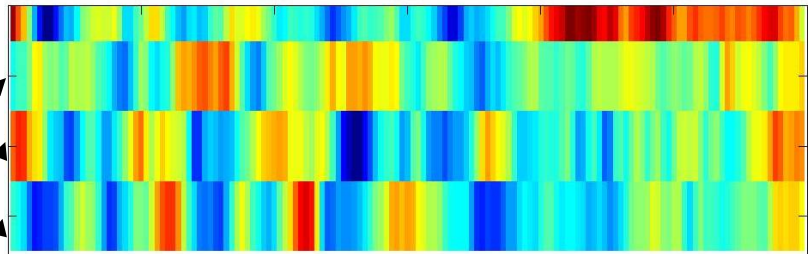


ERP Image basics



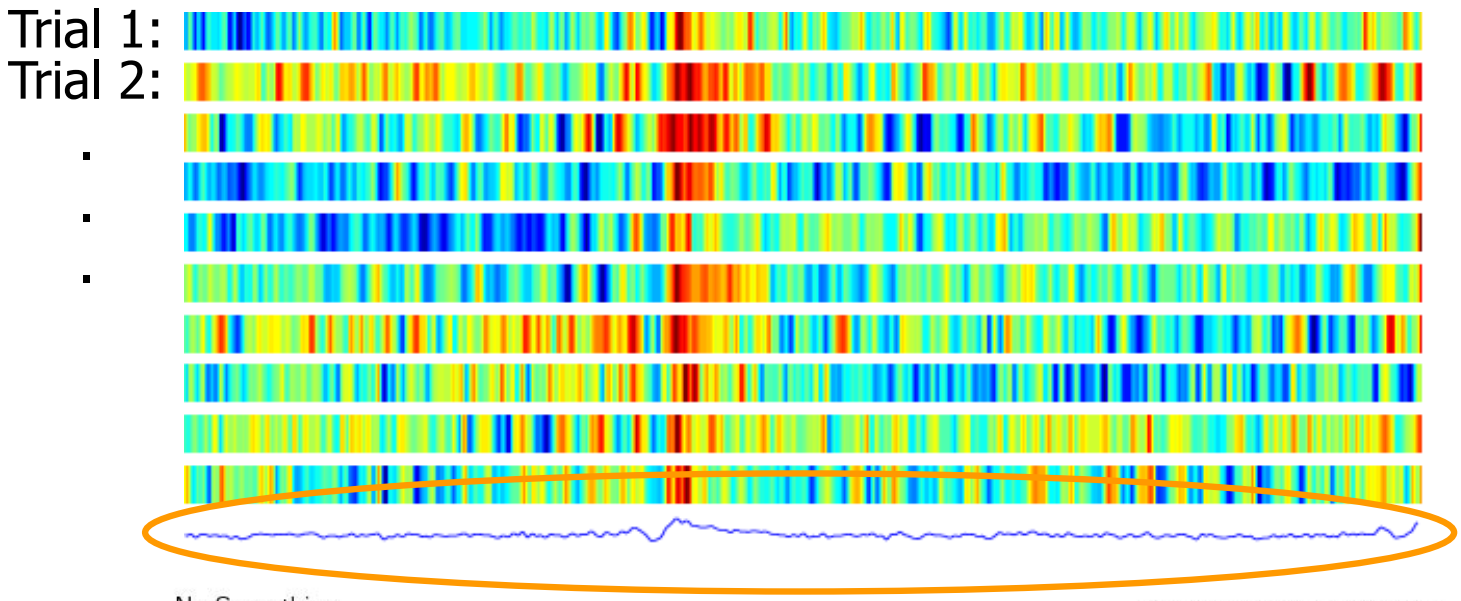


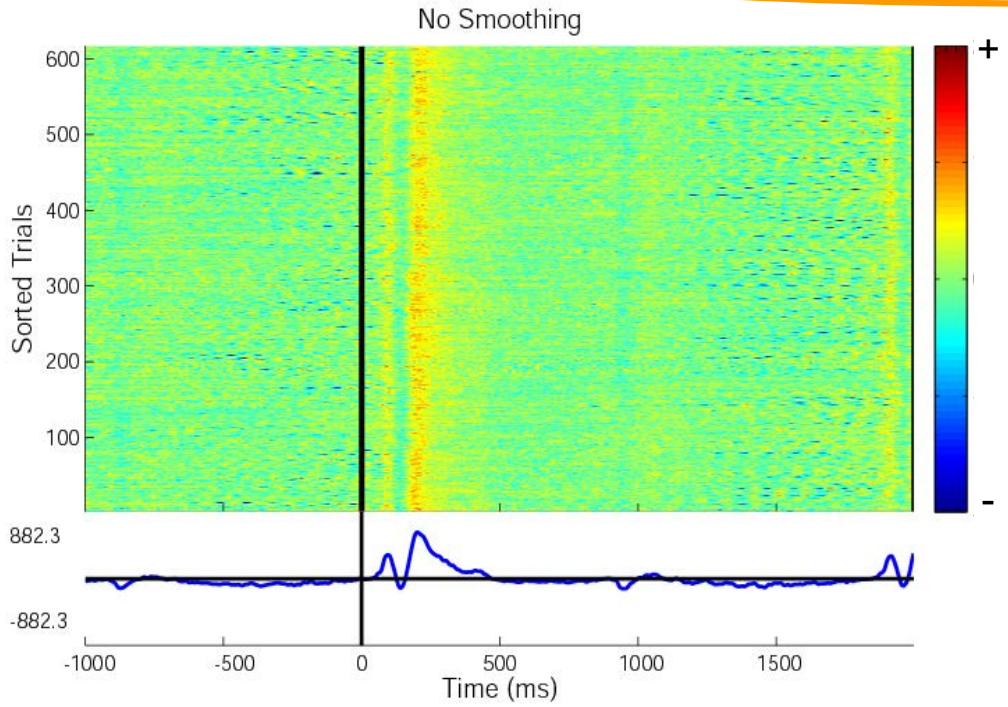
ERP Image

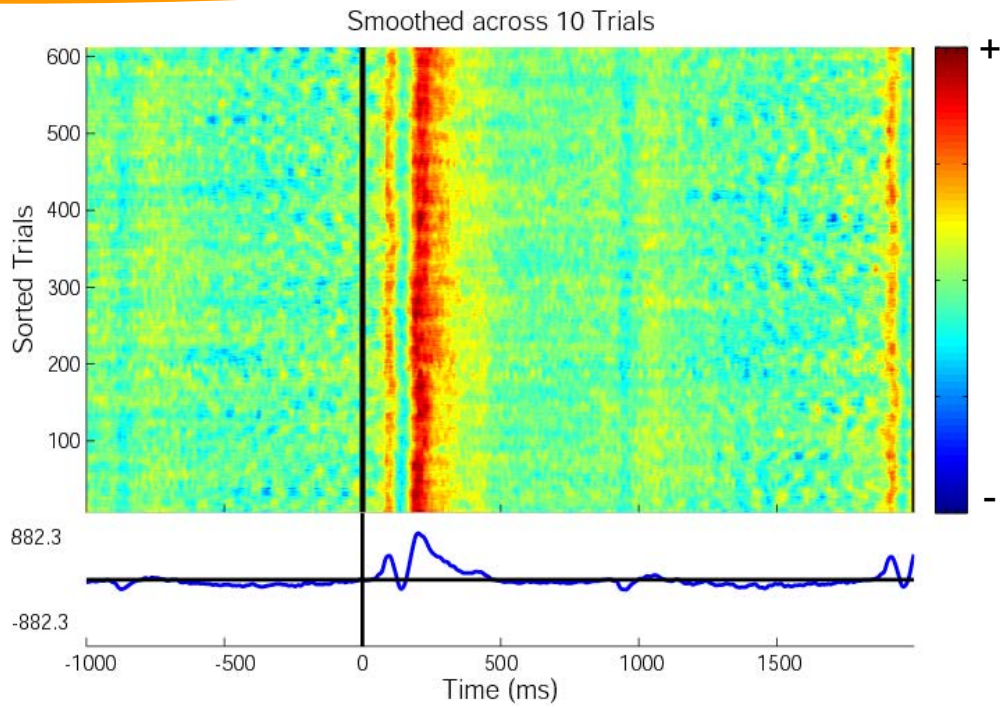


by default, sorted by time-on-task (1st trial, 2nd trial, ...)

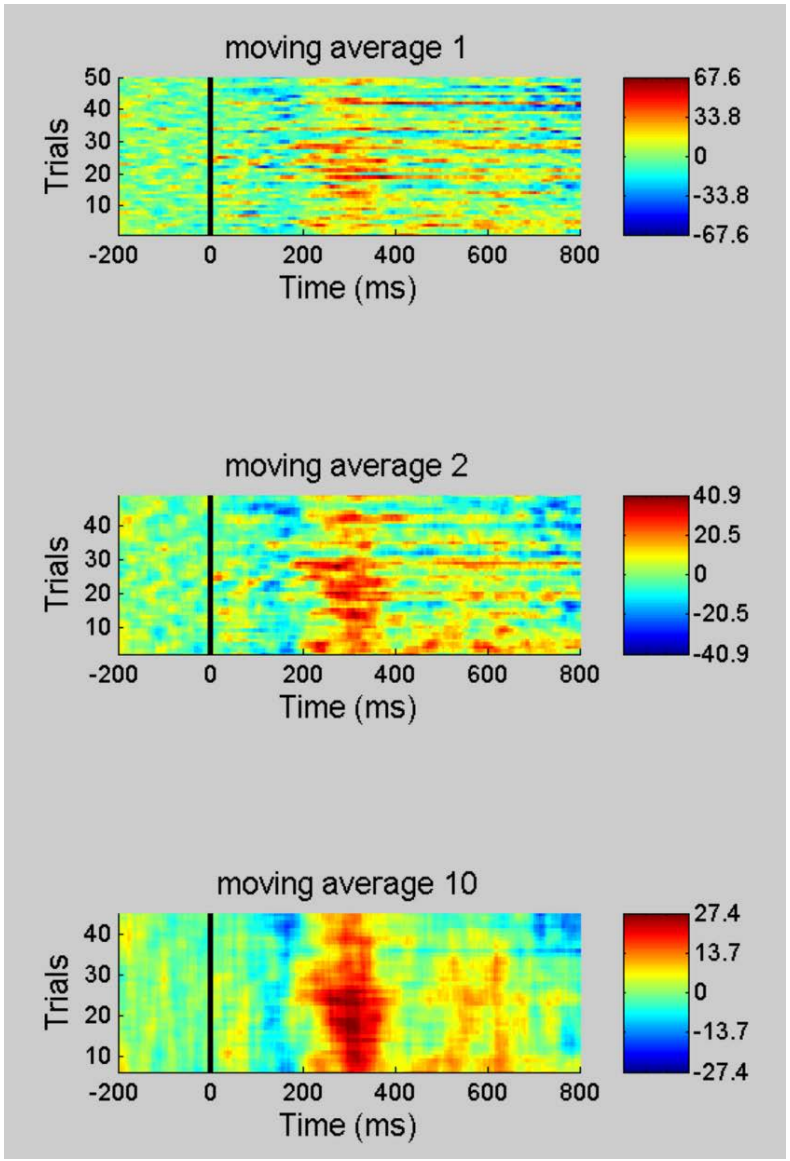
ERP Image basics



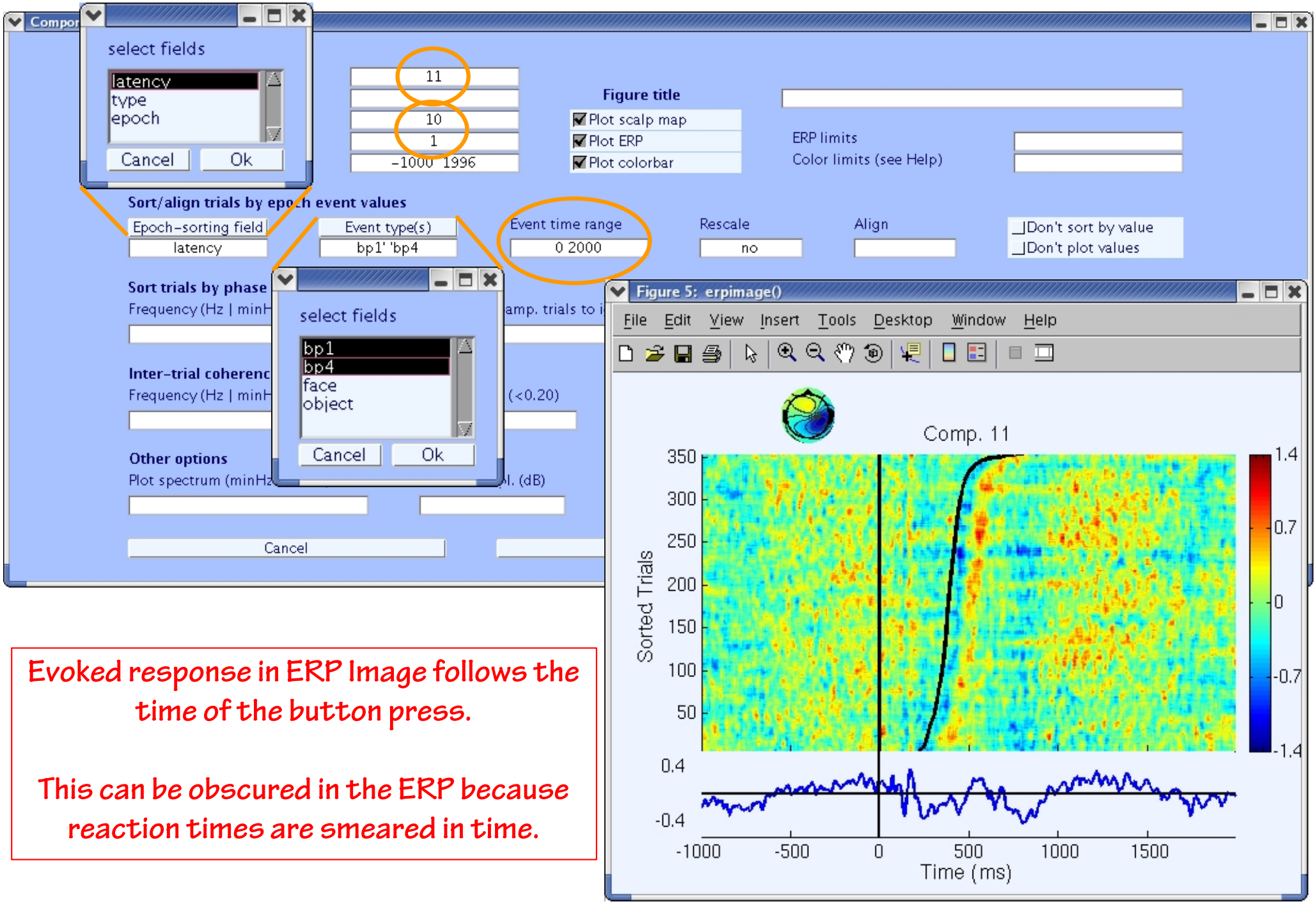




ERP Images: smoothing across trials



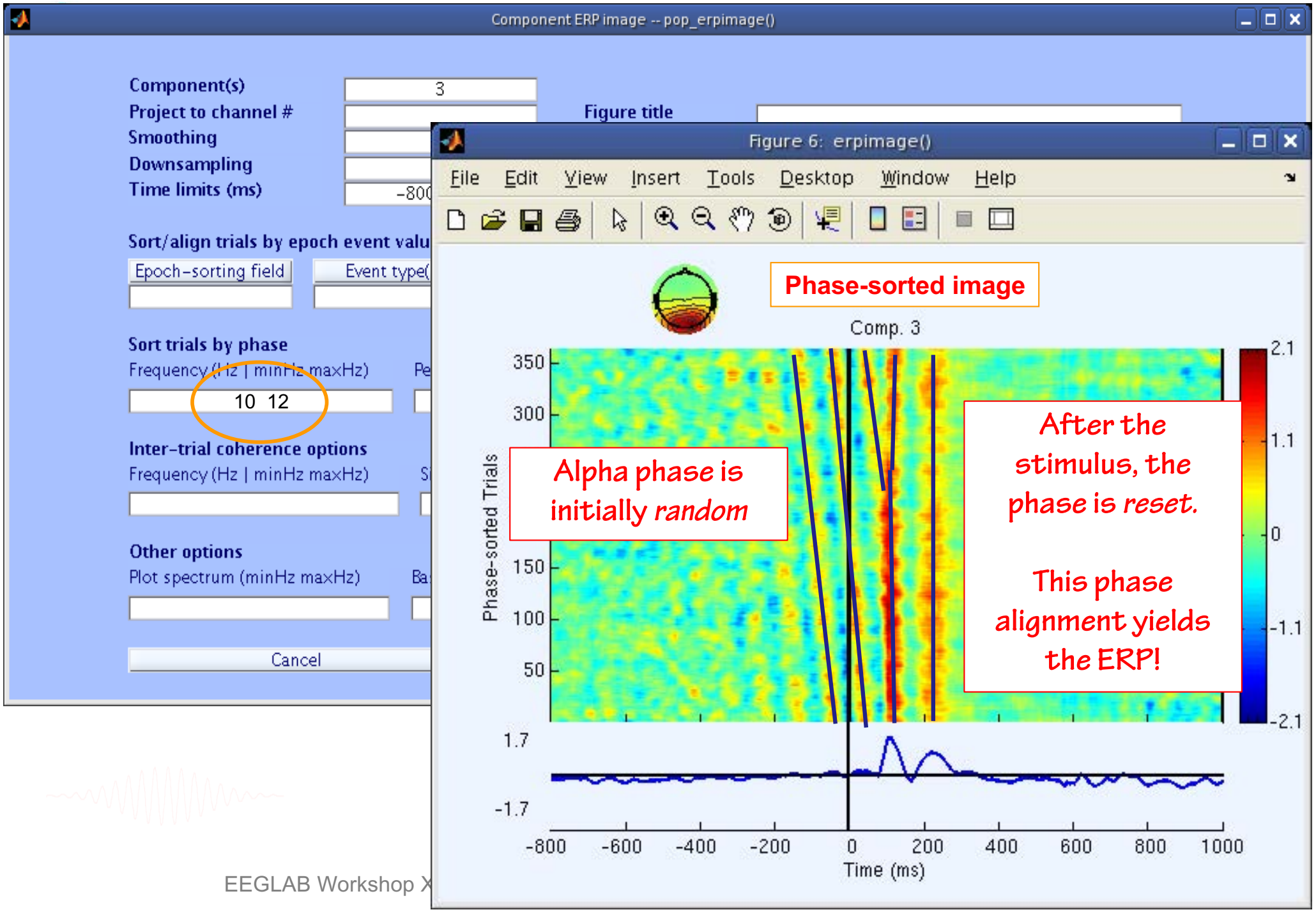
Component ERP Image: Sort by RT



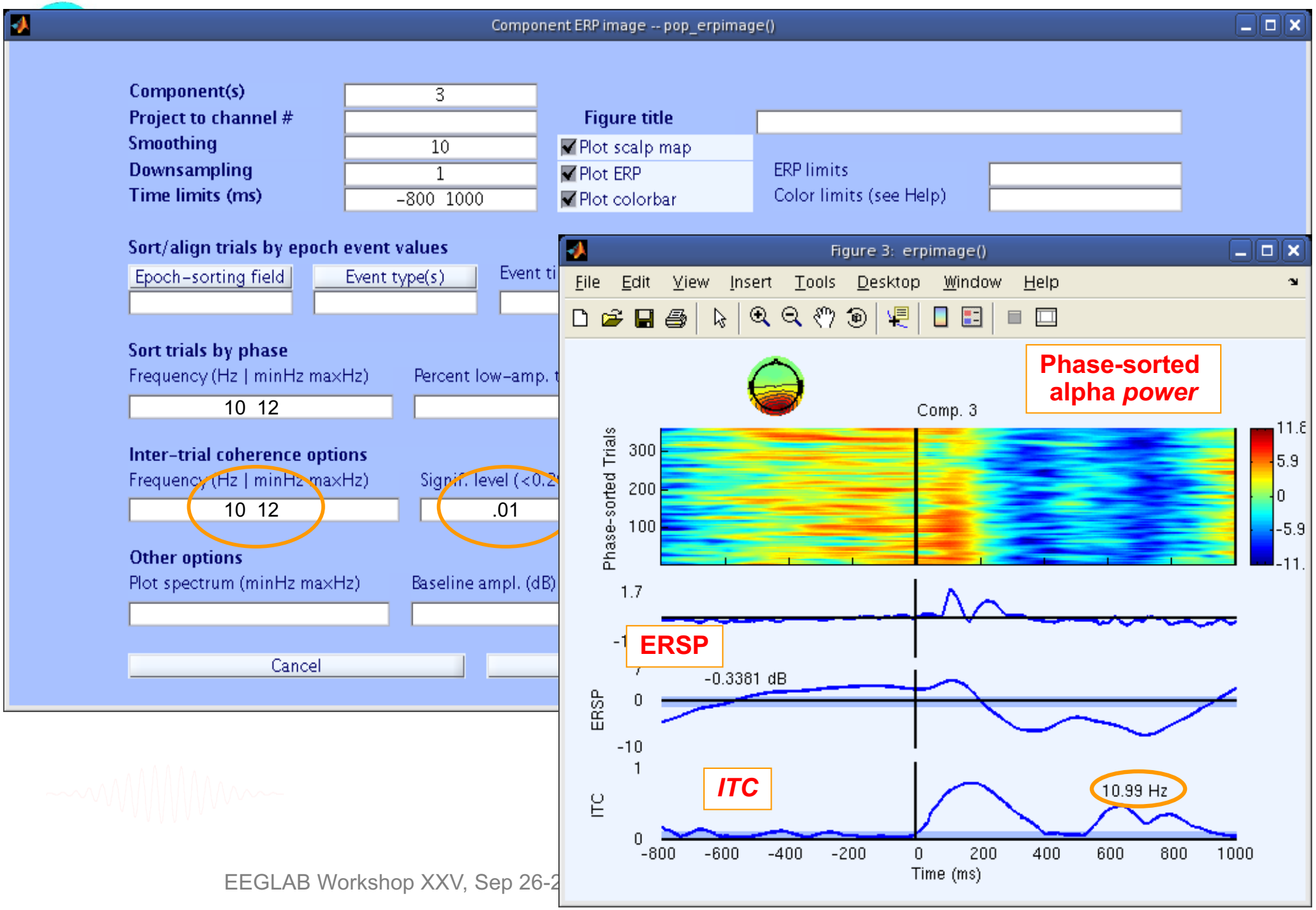
Evoked response in ERP Image follows the time of the button press.

This can be obscured in the ERP because reaction times are smeared in time.

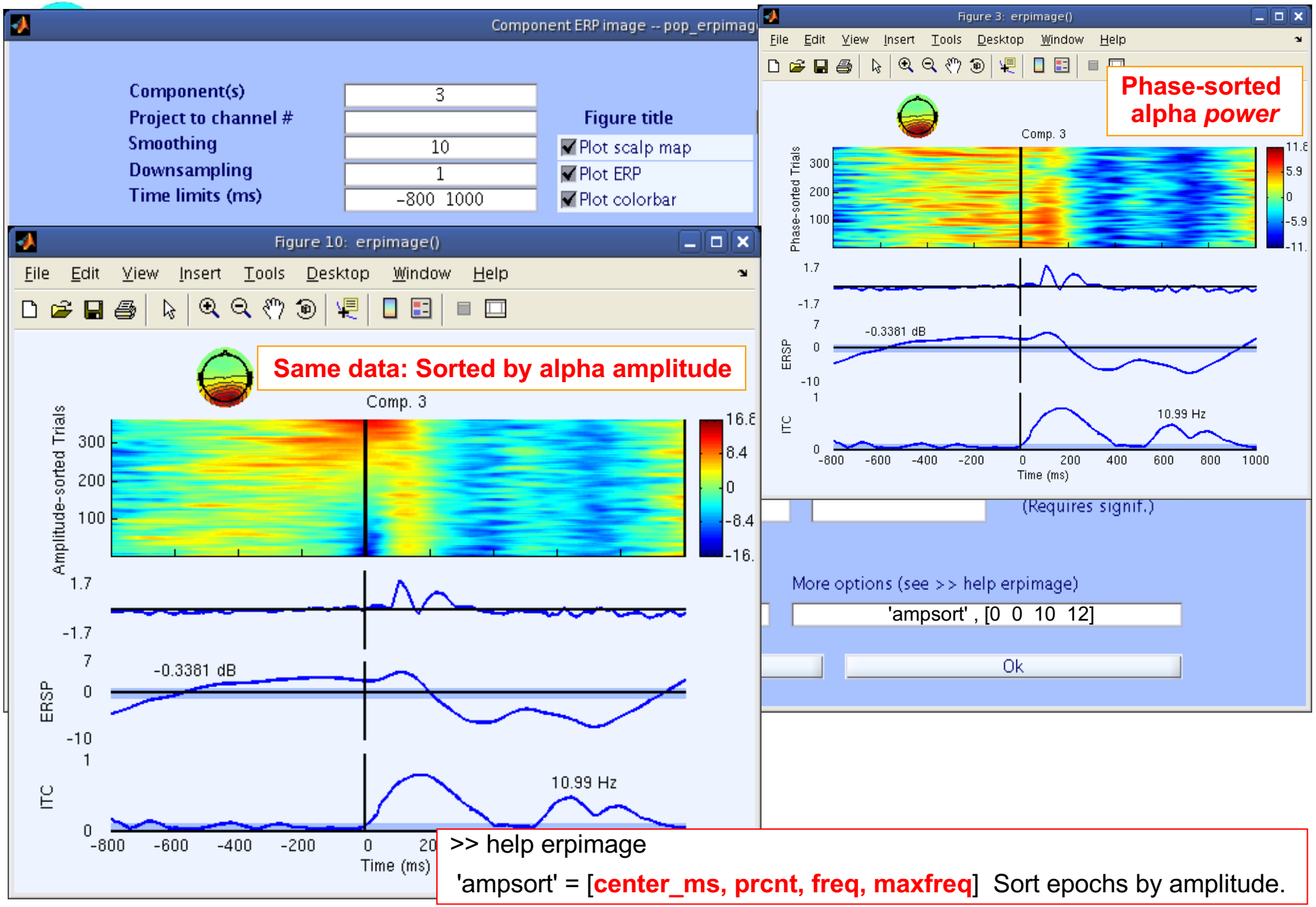
Component ERP Images: Sort by phase



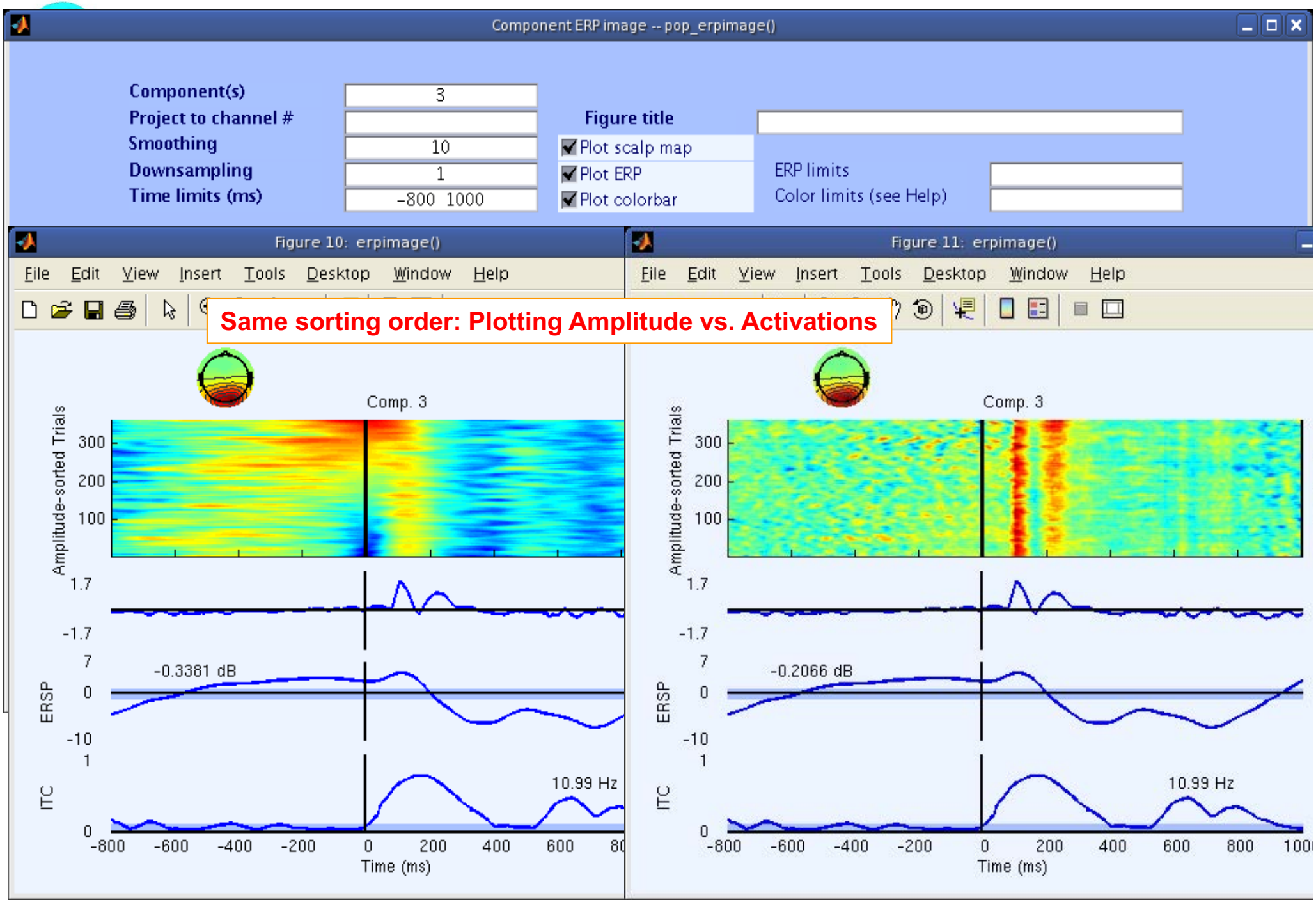
Component ERP Images: ITC



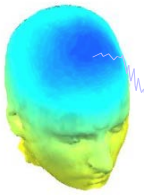
Component ERP Images: Sort by amplitude



Component ERP Images: Amplitude vs. Activations



Now what…?



Part 1

Getting an overview of your ICs

Part 2

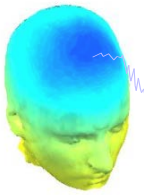
Classifying/Evaluating ICs

Part 3

Detailed look at IC properties

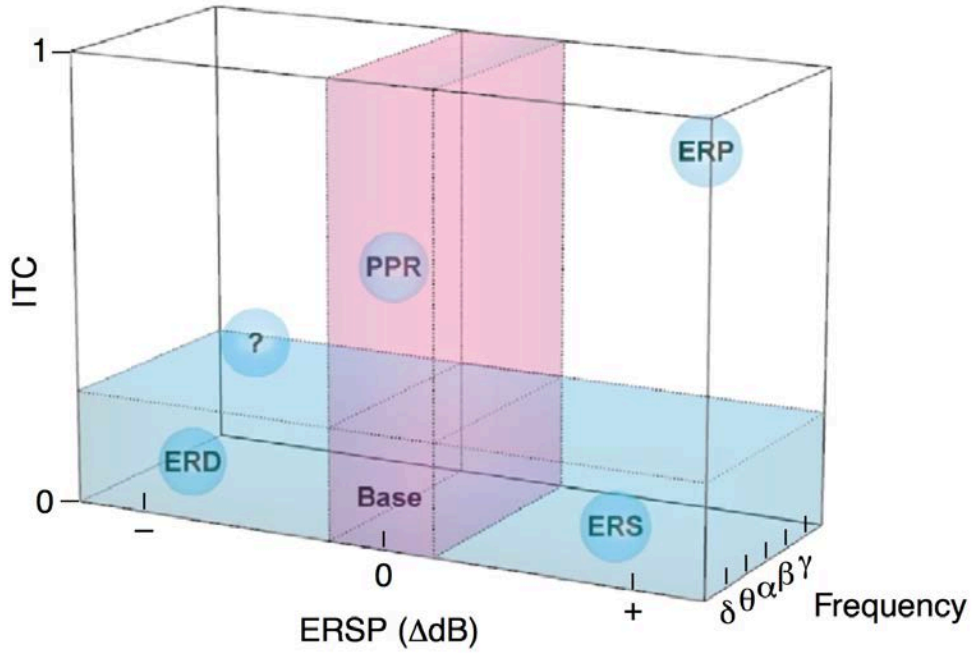
- ERP
- Spectrum
- ERP images
- ERSP (see next lecture)

Definition: ERSP

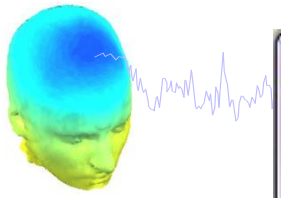


Event Related Spectral Perturbation

Change in power in different frequency bands relative to a baseline. ERS , ERD



Plot IC ERSP



Plot component time frequency -- pop_newtimef()

Component number 3

Sub epoch time limits [min max] (msec) -1000 1996

Frequency limits [min max] (Hz) or sequence

Baseline limits [min max] (ms

Wavelet cycles [min max/fact]

ERSP color limits [max] (min=

ITC color limits [max]

Bootstrap significance level (

Optional newtimef() argumen

Use 200 time points

Use limits, padding 1

Log spaced

Plot Event Related Spectral Po

Cancel

EEGLAB v7.1.7.18b

File Edit Tools Plot

#2: Ste

Filename:
Channels
Frames pe
Epochs
Events
Sampling
Epoch sta
Epoch enc
Reference
Channel lc
ICA weight
Dataset si

Sum/Compare ERPs

Component activations (scroll)

Component spectra and maps

Component maps

Component properties

Component ERP image

Component ERPs

Sum/Compare comp. ERPs

Data statistics

Time-frequency transforms

Cluster dataset ICs

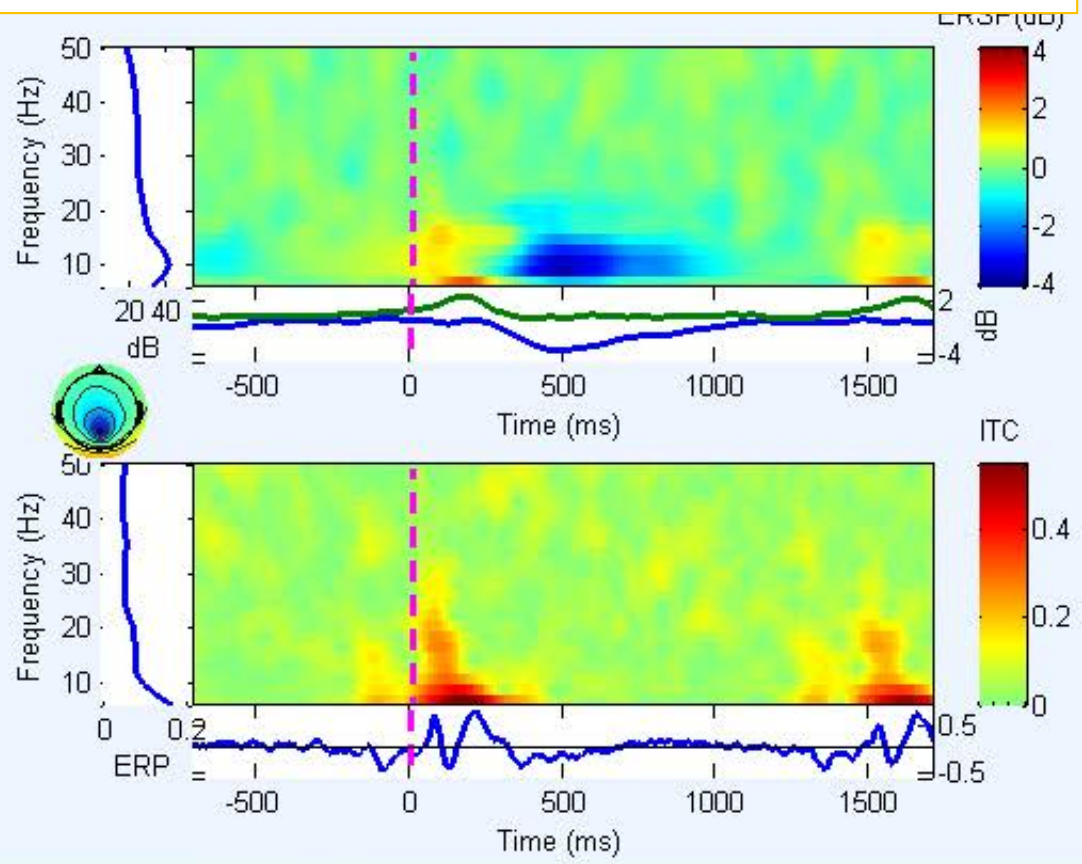
Channel t

Channel c

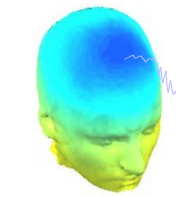
Compone

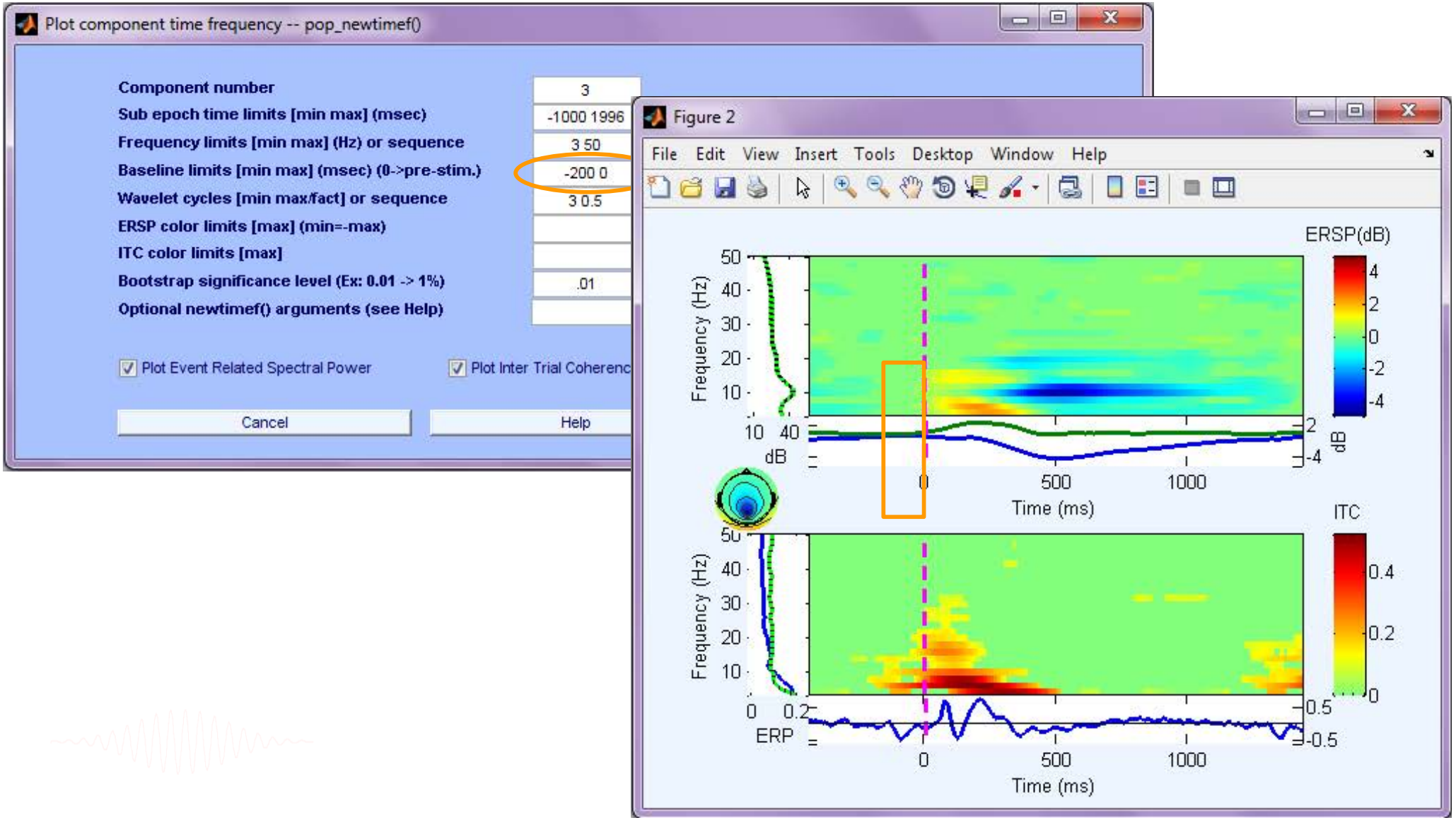
Compone

Time points removed at beginning and end of epoch to avoid edge effects of wavelet transoform.

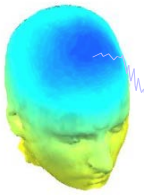


Plot IC ERSP




Plot component time frequency -- pop_newtimef()
Component number
3
Sub epoch time limits [min max] (msec)
-1000 1996
Frequency limits [min max] (Hz) or sequence
3 50
Baseline limits [min max] (msec) (0->pre-stim.)
-200 0
Wavelet cycles [min max/fact] or sequence
3 0.5
ERSP color limits [max] (min=-max)
ITC color limits [max]
Bootstrap significance level (Ex: 0.01 -> 1%)
.01
Optional newtimef() arguments (see Help)
Plot Event Related Spectral Power
Plot Inter Trial Coherenc
Cancel
Help
Figure 2
File Edit View Insert Tools Desktop Window Help
ERSP(dB)
Frequency (Hz)
Time (ms)
dB
ITC
ERP

Further Resources



Some attempts to automate the IC classification:

"Automatic Classification of Artifactual ICA-Components for Artifact Removal in EEG Signals"

Irene Winkler, Stefan Haufe and Michael Tangermann (2011)

http://www.behavioralandbrainfunctions.com/content/7/1/30

Bigdely-Shamlo's EyeCatch (2013)

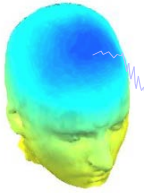
https://www.researchgate.net/publication/257602145_EyeCatch_Data-mining_over_half_a_million_EEG_independent_components_to_construct_a_fully-automated_eye-component_detector

Luca Pion-Tonachini (ongoing)

Crowd-sourcing heuristic knowledge about IC components to build automatic classifier

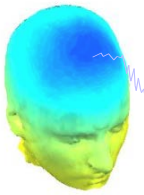
We'll play the game later: http://reaching.ucsd.edu:8000

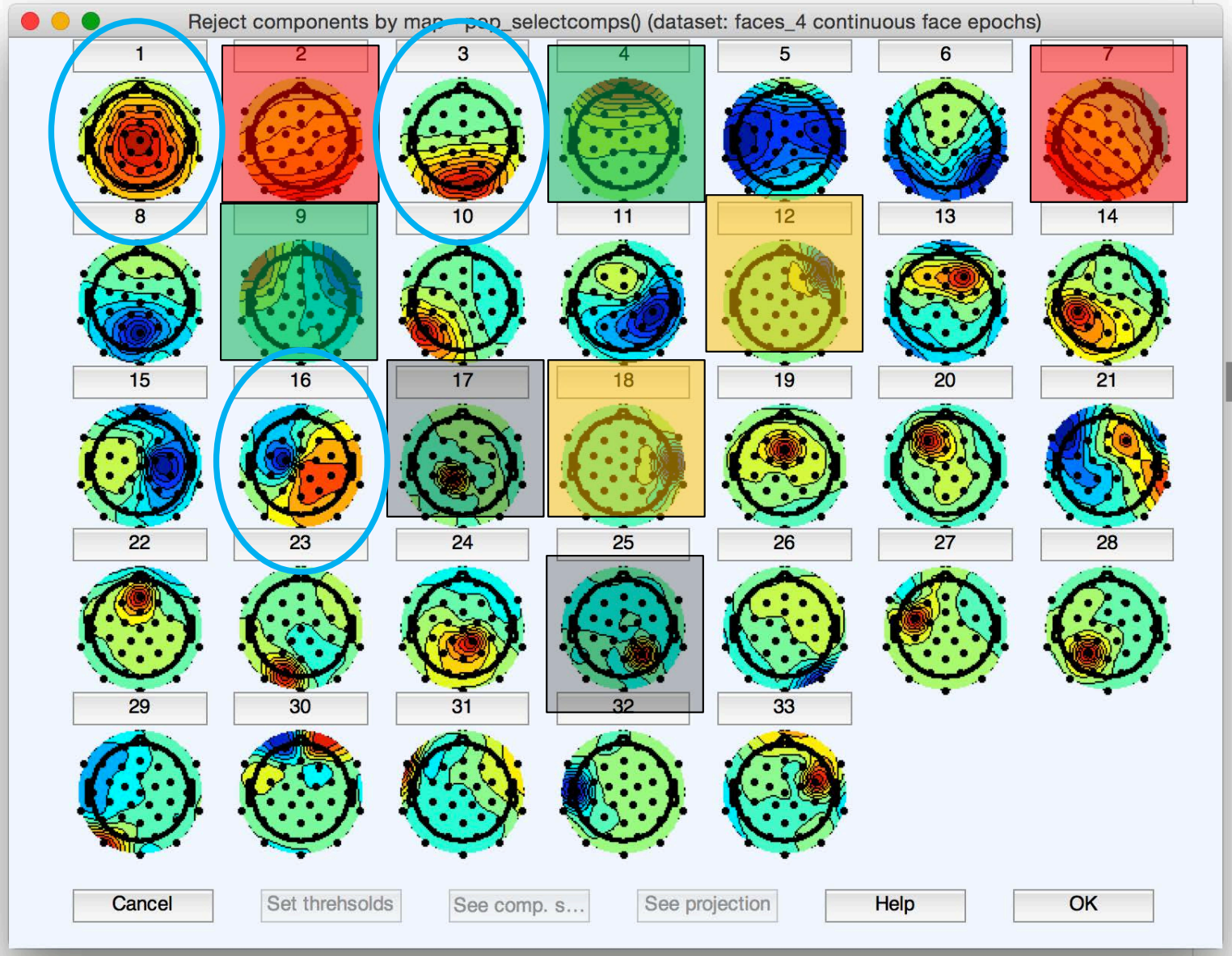
IC Evaluation Practicum (Day 1)



- **ICA Component Classifier Competition**
- **Traditional Practicum using faces_4.set**

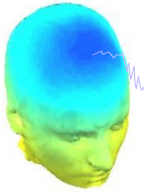
IC Classification…so far





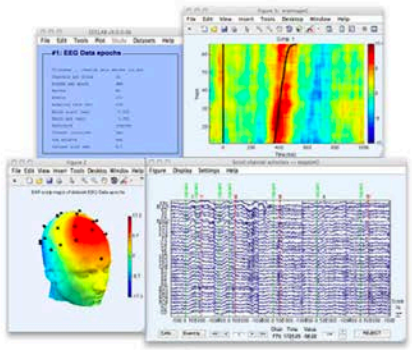
Eye
Muscle
Cardiac
Badchan
Brain

IC Evaluation Practicum (Day 1)



https://sccn.ucsd.edu/eeglab

What is EEGLAB?



EEGLAB is an interactive Matlab toolbox for processing continuous and event-related EEG, MEG and other electrophysiological data incorporating independent component analysis (ICA), time/frequency analysis, artifact rejection, event-related statistics, and several useful modes of visualization of the averaged and single-trial data. EEGLAB runs under Linux, Unix, Windows, and Mac OS X.

EEGLAB provides an interactive graphic user interface (GUI) allowing users to flexibly and interactively process their high-density EEG and other dynamic brain data using independent component analysis (ICA) and/or time/frequency analysis (TFA), as well as standard averaging methods. EEGLAB also incorporates extensive tutorial and help windows, plus a command history function that eases users' transition from GUI-based data exploration to building and running batch or custom data analysis scripts. EEGLAB offers a wealth of methods for visualizing and modeling event-related brain dynamics, both at the level of individual EEGLAB 'datasets' and/or across a collection of datasets brought together in an EEGLAB 'studyset.'

For experienced Matlab users, EEGLAB offers a structured programming environment for storing, accessing, measuring, manipulating and visualizing event-related EEG data. For creative research programmers and methods developers, EEGLAB offers an extensible, open-source platform through which they can share new methods with the world research community by publishing EEGLAB 'plug-in' functions that appear automatically in the EEGLAB menu of users who download them. For example, novel EEGLAB plug-ins might be built and released to 'pick peaks' in ERP or time/frequency results, or to perform specialized import/export, data visualization, or inverse source modeling of EEG, MEG, and/or ECOG data.

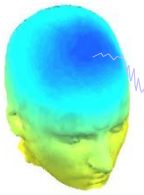
EEGLAB Features

- Graphic user interface
- Multiformat data importing
- High-density data scrolling
- Interactive plotting functions
- Semi-automated artifact removal
- ICA & time/frequency transforms
- Event & channel location handling
- Forward/inverse head/source modeling
- Defined EEG data structure
- Many advanced plug-in/extension toolboxes

Q: I've used ICA to decompose my data -- now how can I learn to recognize which independent component processes of EEG data represent brain sources activity and which capture activity from other non-brain sources?

Try using ICLabel to first learn about and then practice labeling EEG independent components (ICs). You can also help to create a more accurate automated IC classifier using machine learning algorithms on crowd-sourced data. The results are being incorporated into a self-updating EEGLAB plug-in that will become more accurate the more labels you and others contribute.

Automating IC Identification



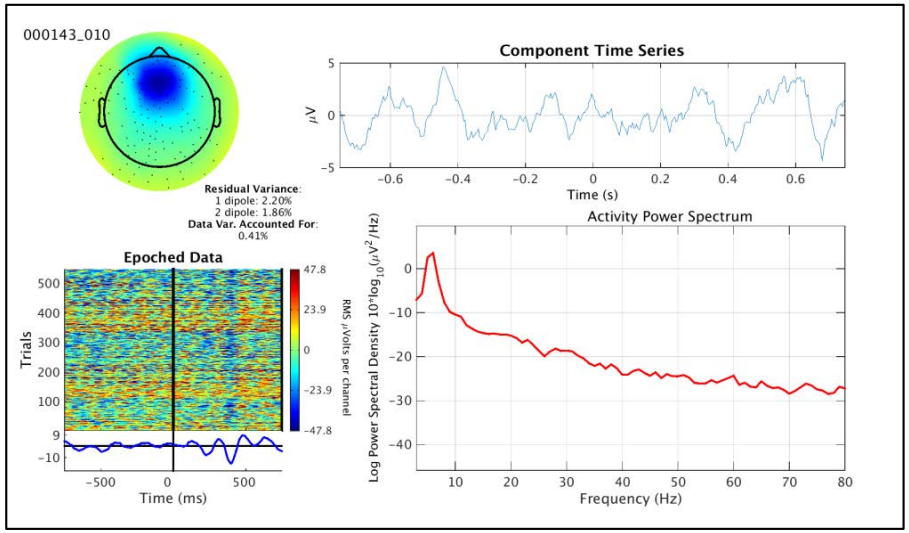
Luca Pion-Tonachini (lpionton@ucsd.edu)

Goal: Create an automated, high confidence EEG component labeler.

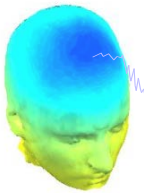
Motivation: Typically we rely on expert knowledge to pick which components to work with, but can be very time consuming with large datasets or inconvenient / infeasible when automation is the goal (BCI).

Plan:

1. Aggregate Data
2. Gather Labels
   reaching.ucsd.edu:8000/tutorial
3. Process Labels
4. Train Classifiers
   Real-time and offline versions



– let's play…



Tutorial: EEG Independent Component Labeling

Overview
Why Help Us?
How To Label
Telling Components Apart

Practice Labeling
Leave A Comment

Return To Labeling

Overview

We would like you to help us label independent components from EEG datasets to create an automated classifier. For more information, see Why Help Us?.

Steps to doing so

1. Register or Log In.
2. Look at the image presented. For help reading it, see How To Label. It is essential that you go over the instructions before you start.
3. For each component presented, try to decide what type of component you are looking at. To learn how to do this, see Telling Components Apart and perform Practice Labeling.
4. Click the appropriate button or buttons to label the component. For help with our categories, first read How To Label.
5. Then click on "Next" to view a new component.

That's it! Please read the text in all the links above and perform some Practice Labels. Then click Begin Labeling.

If you have any suggestions, please Leave A Comment. Also – we have a Leaderboard!

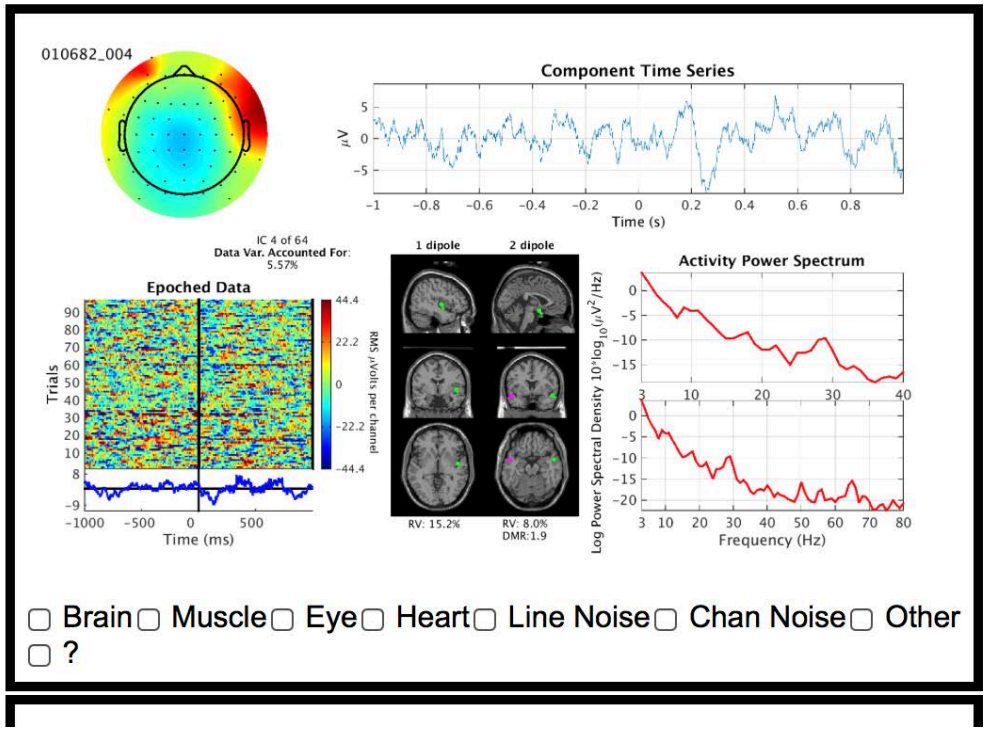
Practice First…

Practice

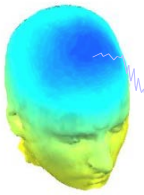
Example component images are provided below. Click on all the labels your fe
follows:

- White: no labels
- Grey: correct but insufficient labels
- Green: all labels correct
- Red: one or more labels are incorrect

Marking "?" is ignored here as that category is user dependent.



Practicum



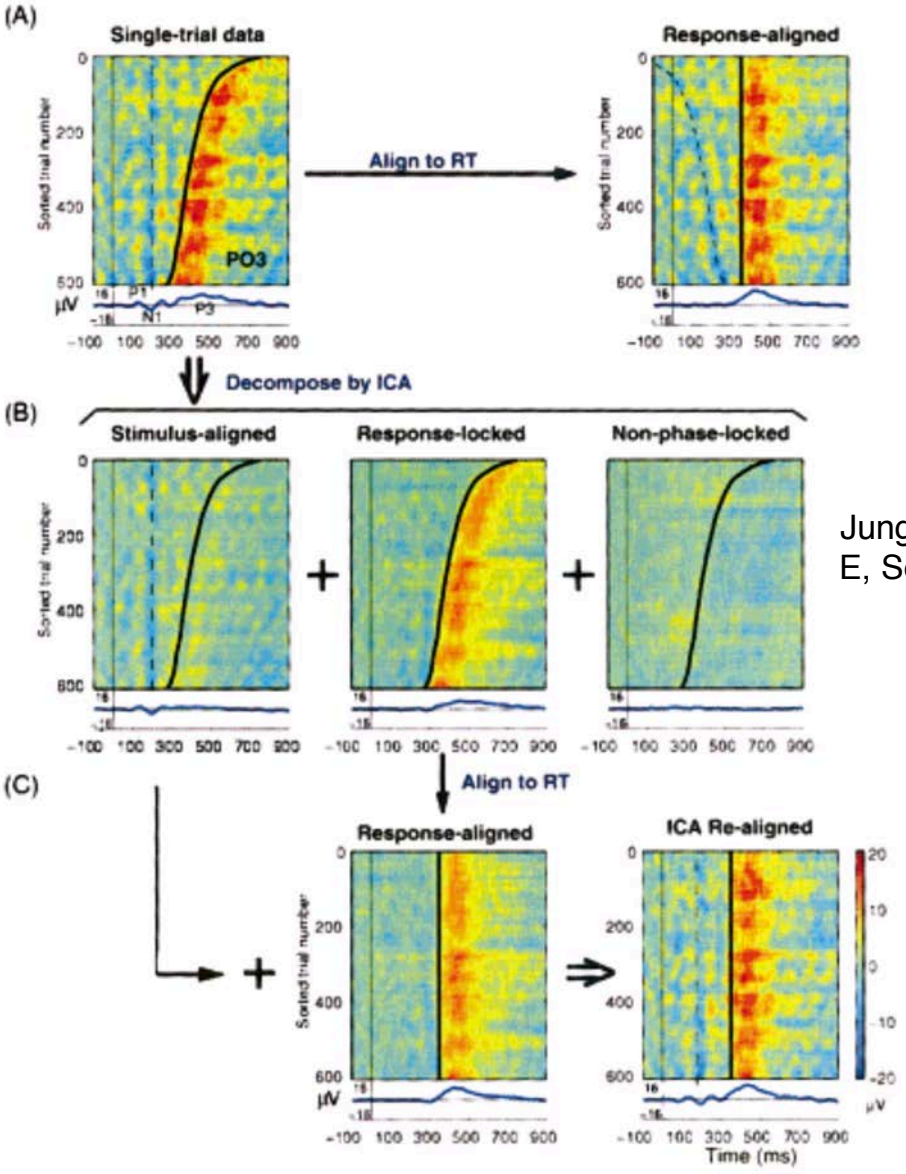
- **ALL**
  - Download then load faces_4.set, epoch on face

- **Novice, Intermediate**
  - From the GUI, open the 'Reject component by map' interface
  - Explore and classify several additional ICs: muscle, channel, brain
    ~ Justify your classification
  - Redo the "Plot → Component ERPs → With component maps" excluding your additional artifacts. What change do you observe?
  - Pick a brain IC. Plot an ERP Image
    ~ Try sorting by phase, is there any relationship to the IC activation pattern? What about power in a frequency band of choice?

- **Intermediate**
  - Plot ERP Image sorted by response latency
    ~ Figure out how to realign trials to response latency instead (Hint 'Align')
  - Plot ERSPs for selected ICs
    ~ Explore parameter options. Why is each useful?
  - Plot component cross-coherence for pairs of ICs

- **ALL (Time permitting)**
  - Create second dataset, epoched on object
  - Examine ERP differences between the conditions using "Plot → Component ERPs → With component maps (compare)"
  - For ICs most different between conditions, compare ERP Image, ERSP

Realigning Trials: Stimulus vs. Response



Jung T-P, Makeig SD, Westerfield M, Townsend J, Courchesne E, Sejnowski TJ (2001) Analysis and visualization of single-trial event-related potentials. Hum Brain Mapp 14:166–85.

Nice method for generating dual ERPs