STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

STEP 5
Edit/view the clusters
STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

STEP 5
Edit/view the clusters
Build a STUDY
Build a STUDY, cont'd

Create a new STUDY set
STUDY set name:
STUDY set task name:
STUDY set notes:

<table>
<thead>
<tr>
<th>dataset filename</th>
<th>browse</th>
<th>subject</th>
<th>session</th>
<th>condition</th>
<th>group</th>
<th>Select by r.v.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>2</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>3</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>4</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>5</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>6</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>7</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>8</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>9</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
<tr>
<td>10</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Clear</td>
</tr>
</tbody>
</table>

Important note: Removed datasets will not be saved before being deleted from EEGLAB memory.

Update dataset info – datasets stored on disk will be overwritten (unset = retain original dataset info). Delete cluster information to allow loading new datasets, set new comment.

Save this study to a disk file named:

[Choose dataset to add to STUDY dialog box]

- S01_attend1_pos1.set
- S01_attend1_pos3.set
- S01_attend5_pos1.set
- S01_attend5_pos3.set

File Name: S01_attend1_pos1.set
Files of Type: (*.set, *set)

[Dialog box options: Open, Cancel]
Edit dataset info

Create a new STUDY set – pop_study()
ICs to cluster
STUDY structure

STUDY =

name: 'Synonyms'
task: 'Word Recognition'
notes: ''
filename: 'workshop.study'
cluster: [1x1 struct]
history: [1x6654 char]
datasetinfo: [1x10 struct]
filepath: '/data/STUDY'
subject: {'S02' 'S05' 'S07' 'S08' 'S10'}
group: {}
session: []
condition: {'non-synonyms' 'synonyms'}
setind: [2x5 double]
etc: [1x1 struct]
preclust: [1x1 struct]
saved: 'no'
changrp: []

>>
Subject info in STUDY structure

>> STUDY.datasetinfo

ans =

1x10 struct array with fields:
    filepath
    filename
    subject
    session
    condition
    group
    comps
    index

>>

Gives information for each subject
STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

STEP 5
Edit/view the clusters
Precompute data measures
Precompute data measures

TIP: Compute all measures and so you can test different combinations for later clustering

Recommend: 'alpha', .01
STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

STEP 5
Edit/view the clusters
Precluster the data
Precluster the data

Build pre-clustering matrix for STUDY 'Attention'
Select the cluster to refine during sub-clustering (any existing sub-hierarchy will be overwritten)

ParentCluster 1 (181 ICs)

Load
- spectra
- ERPs
- dipoles
- scalp maps
- ERSPs
- ITCs

Dims. Norm. Rel. Wt. Freq. range [Hz] Time range [ms]
- 10
- 10
- 3
- 10
- 20
- 10

Use channel values
- Time range [ms]
- Time range [ms]
- 0 1500
- 0 600

Save STUDY to file
/home/julie/WorkshopSD2007/STUDY/attention.study

Cancel Help Ok
STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

STEP 5
Edit/view the clusters
Cluster components

Tip: start with a lot of clusters (~25) and work down if necessary.
STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

STEP 5
Edit/view the clusters
View and edit clusters
View and edit clusters

LEFT SIDE: Plot measures of full clusters

RIGHT SIDE: Plot measures of individual components
Plot cluster data

Plot mean scalp maps for easy reference
Plot cluster data

Choose which cluster

Choose which components
Plot clusters

Study 'Attention': 181 of 181 components clustered

Select cluster to plot
- Cis 5 (3 ICs)
- Cis 7 (10 ICs)
- Cis 8 (5 ICs)
- Cis 11 (2 ICs)

- Plot scalp maps
- Plot dipoles
- Plot ERPs
- Plot spectra
- Plot ERSPs
- Plot ITCs
- Plot cluster properties

Select component(s) to plot
- All components
- S01 IC6
- S05 IC9
- S06 IC12

- Plot scalp map(s)
- Plot dipole(s)

- Params
- Plot ERPs
- Plot spectra
- Plot ERSPs
- Plot ITCs
- Plot component

Create new cluster
- Rename selected cluster
- Merge clusters

- Save STUDY set to disk
- /home/julie/WorkshopSD2007/

- Cancel
- Help
Plot ERPs

Study 'Attention': 181 of 181 components clustered

Select cluster to plot:
- Cls 6 (3 ICs)
- Cls 7 (10 ICs)
- Cls 8 (5 ICs)
- Cls 9 (12 ICs)

Select component(s) to plot:
- All components
- S01 IC6
- S05 IC9
- S06 IC12

Set parameters for plotting ERPs -- pop_erpparams()
Plot cluster ERP

Each blue trace is the ERP of a different component
Plot cluster spectra

Study 'Attention': 181 of 181 components clustered

Select cluster to plot
- Cls 6 (3 ICs)
- Cls 7 (10 ICs)
- Cls 9 (12 ICs)

Select component(s) to plot
- All components
- S01 IC6
- S05 IC9
- S06 IC12

Set parameters for plotting specs -- pop_specparams()

- Frequency [low_Hz high_Hz]
- Plot scalp map at freq. [Hz]
- Subtract individual subject mean spectrum
- Compute condition statistics
- Compute group statistics
- Use False Discovery Rate to correct for multiple comparisons

Cancel  Help  Ok
Plot cluster spectra

Each blue trace is the power spectrum of a different component
Plot cluster ERSPs

Study 'Attention': 181 of 181 components clustered

Select cluster to plot

Cis 6 (3 ICs)
Cis 7 (10 ICs)
Cis 8 (5 ICs)
Cis 9 (12 ICs)

Plot scalp maps
Plot dipoles
Plot ERPs
Plot spectra
Plot ERSPs
Plot ITCs
Plot cluster properties

Select component(s) to plot

All components
S01 IC6
S05 IC9
S06 IC12

Plot scalp map(s)
Plot dipole(s)
Plot ERP(s)
Plot spectra
Plot ERSPs
Plot ITC(s)

Set ERSP | ITC plotting parameters -- pop_erspparams()

Time range in ms [Low High]
Freq. range in Hz [Low High]
Power limits in dB [Low High]
ITC limit (0-1) [High]

Compute ERSP baseline across conditions

Statistics
Permutation
compute condition statistics
compute group statistics
Mask non-significant data (only when threshold is set)
Use False Discovery Rate to correct for multiple comparisons

Cancel
Help
Ok
Plot cluster ERSPs

ERSP (non-sig voxels across conditions are green)

ITC (non-sig voxels across conditions are green)
Remove outlier components
Remove outlier components
Remove outlier components

### Study settings

**Select cluster to plot**
- Cls 16 (6 ICs)
- Cls 17 (3 ICs)
- Cls 18 (2 ICs)
- Cls 19 (14 ICs)
- Outliers Cls 17, 20 (1 ICs)

**Select component(s) to plot**
- All components
- S10TC60

### Plotting options

- Plot dipoles
- Plot ERPs
- Plot spectra
- Plot ERSPs
- Plot ITCs
- Plot cluster properties
- Create new cluster
- Rename selected cluster
- Merge clusters

**Save STUDY set to disk**
/home/julie/workshop06/5subjects/WS/study.study

**Options**
- Cancel
- Help
- Ok
Reassign component

If you want to manually reassign a component to another cluster...
Reassign component

Study 'Attention': 181 of 181 components clustered

Select cluster to plot
- CIs 18 (4 ICs)
- CIs 19 (10 ICs)
- CIs 20 (17 ICs)
- CIs 21 (6 ICs)
- Plot scalp maps
- Plot dipoles
- Plot ERPs
- Plot spectra
- Plot ERSPs
- Plot ITCs
- Plot cluster properties
- Create new cluster
- Rename selected cluster
- Merge clusters
- Save STUDY set to disk /home/julie/WorkshopSDz007/STUDY

Select component(s) to plot
- All components
- S02 IC12
- S02 IC14
- S03 IC1
- Plot scalp map(s)
- Plot dipole(s)
- Plot ERP(s)
- Plot spectra
- Plot ERSP(s)
- Plot ITC(s)
- Plot component properties
- Reassign selected component(s)
- Remove selected outlier comps.
- Auto-reject outlier components

Reassign currently selected component from CIs 19 to the cluster selected below
- CIs 4
- CIs 5
- CIs 6
- CIs 7

[Ok] [Cancel]
Reassign component

Successful reassignment
Rename a cluster

Name your cluster of interest
Create a new cluster

You found a bunch of 'outliers' that seem well-matched

Create a new cluster...
Channel plotting
Exercise

• **Novice**
  - Use the GUI to build a STUDY (for practice, try just a few subjects).
  - Choose 'precompute' options (but do not recalculate for the sake of time).
  - Choose 'precluster' options and cluster.
  - Use the GUI to plot cluster and component data using default parameters

• **Intermediate**
  - Create a Study and add a single subject. Get the history from the command line and script a loop to build a STUDY with all subjects from the commandline
  - Precluster (pre-computation already done) and cluster components using measures of your choice.
  - Use the GUI to plot cluster and component data trying out different plotting parameters such as x/y-axis limits, and color scale limits to compare absolute values across clusters.
  - Apply statistical thresholds of your choice
  - Create and name a new cluster, fill with your choice of ICs
STUDY analysis, Part 1

Task 1
Script STUDY plotting functions

Task 2
STUDY structure details

Task 3
Load/plot data from commandline

Task 4
Apply significance limits

Task 5
Time/frequency analysis within cluster

Exercise...
STUDY analysis, Part 1

Task 1

Script STUDY plotting functions

Task 2

STUDY structure details

Task 3

Load/plot data from commandline

Task 4

Apply significance limits

Task 5

Time/frequency analysis within cluster

Exercise...
Task 1: Plot cluster ERSP

% plot all cluster ERSPs with variable parameters using a for loop to compare power across clusters:---------------------

% Define variables:------------------------------------------
frqlim = [3 30]; % ERSP freq range (Hz)
tmlim = [0 1000]; % ERSP time range (ms)
maxdb = 4; % set color limit for dB power from baseline %------------------------------------------------------------

for clust = 3:length(STUDY.clusters)
    STUDY = std_erspplot(STUDY, ALLEEG, 'clusters', clust, ...
    'plotsubject', 'on', 'statistics', 'perm', ...
    'timerange', tmlim, 'freqrange', frqlim, 'threshold', 0.01, ...
    'ersplim', [-maxdb maxdb]);
end;
Task 1: Cluster ERSP plotting

Directly compare event-related power changes across clusters
STUDY analysis, Part 1

Task 1
Script STUDY plotting functions

Task 2
STUDY structure details

Task 3
Load/plot data from commandline

Task 4
Apply significance limits

Task 5
Time/frequency analysis within cluster

Exercise...
Task 2: STUDY structure details

Question:

I want to know which ICs from which subjects are in a particular cluster.

Where in the STUDY structure can I find this information?
Task 2: Understanding STUDY structure

Choose a cluster:  >> STUDY.cluster(clust)
ans =
   name: 'Cls 3'
   parent: {'ParentCluster 1'}
   child: []
   comps: [20 19 10 9 1 4]
   sets: [4x6 double]
   algorithm: {'Kmeans' [25]}
   centroid: []
   preclust: [1x1 struct]
   selected: 1
   allinds: {4x1 cell}
   setinds: {4x1 cell}
   erspbase: {4x1 cell}
   erpdata: {4x1 cell}
   erptimes: [768x1 double]
   topo: [67x67 double]
   topox: [67x1 double]
   topoy: [67x1 double]
   topoall: {1x6 cell}
   topopol: [-1 1 -1 1 1 1]
   erspdata: {4x1 cell}
   erspfreqs: [1x126 double]
   ersptimes: [1x200 double]

>> STUDY.cluster
ans =
   27 total clusters
1x27 struct array with fields:
   name
   parent
   child
   comps
   sets
   algorithm
   centroid
   preclust
   selected
   allinds
   setinds
   erspbase
   erpdata
   erptimes
   topo
   topox
   topoy
   topoall
   topopol

4 = # conditions
Task 2: Understanding STUDY structure

>> STUDY.cluster(clust)
an =
    name: 'Cls 3'
    parent: {'ParentCluster 1'}
    child: []
    comps: [20 19 10 9 1 4]
    sets: [4x6 double]
    algorithm: {'Kmeans' [25]}
    centroid: []
    preclust: [1x1 struct]
    selected: 1
    allinds: {4x1 cell}
    setinds: {4x1 cell}
    erspbase: {4x1 cell}
    erpdata: {4x1 cell}
    erptimes: [768x1 double]
    topo: [67x67 double]
    topox: [67x1 double]
    topoy: [67x1 double]
    topoall: {1x6 cell}
    topopol: [-1 1 -1 1 1 1]
    erspdata: {4x1 cell}
    erspfreqs: [1x126 double]
    ersptimes: [1x200 double]

>> STUDY.cluster(clust).setinds{cond}
an =
    14 38 46 50 54 54

Dataset 38!!

>> STUDY.cluster(clust).allinds{cond}
an =
    20 19 10 9 1 4

>> STUDY.datasetinfo(38) % access dataset 38
ans =
   _filepath: '../Workshop/STUDY/S10'
   _filename: 'S10_attend1_pos5.set'
   _subject: 'S10'
   _session: 1
   _condition: 'NONTargAttnL'
   _group: 'normals'
   _index: 38
   _comps: [1 2 3 4 5 6 7 ...]
STUDY analysis, Part 1

Task 1
Script STUDY plotting functions

Task 2
STUDY structure details

Task 3
Load/plot data from commandline

Task 4
Apply significance limits

Task 5
Time/frequency analysis within cluster

Exercise...
Task 3: Load data from commandline

** Where is the raw data stored?**
Data for each subject is stored in the file path of that subject (STUDY.datasetinfo(subj).filepath)

** How to load it from the commandline:**
File name format: 'setname.extension'
extension = '.ica*' or '.dat*' (for channel data)

for example:
S01_attend1_pos1.icaerp % ERP data
S01_attend1_pos1.icaersp % ERSP data
S01_attend1_pos1.icaitc % ITC data
S01_attend1_pos1.icaspec % Power spectrum data
S01_attend1_pos1.icatopo % Scalp map data

% Example of channel data file name:
S01_attend1_pos1.daterp % ERP data
Task 3: Load individual ERSPs

% call in ERSP data for all ICs in a single cluster:
clust = 5; % choose a cluster
cond = 1; % choose experimental condition
tmlims = [0 1000]; % time limits (ms)
frqlims = [0 40]; % frequency limits (Hz)

for ic = 1:size(STUDY.cluster(clust).sets,2)
    setidx = STUDY.cluster(clust).setind{cond}{ic};
    comp = STUDY.cluster(clust).allinds{cond}{ic};
    [logersp(:, :, ic), logfreqs, timevals, params, baseersp] = ...
    std_readersp(ALLEEG, setidx, comp, tmlims, frqlims);
end;
Task 3: Load individual ERSPs

% Check imported variables in workspace:

>> whos logersp logfreqs timevals params baseersp

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Bytes</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseersp</td>
<td>91x1</td>
<td>728</td>
<td>double</td>
</tr>
<tr>
<td>logersp</td>
<td>91x106x7</td>
<td>540176</td>
<td>double</td>
</tr>
<tr>
<td>logfreqs</td>
<td>1x91</td>
<td>728</td>
<td>double</td>
</tr>
<tr>
<td>params</td>
<td>1x1</td>
<td>4432</td>
<td>struct</td>
</tr>
<tr>
<td>timevals</td>
<td>1x106</td>
<td>848</td>
<td>double</td>
</tr>
</tbody>
</table>
Task 3: PLOT individual ERSPs

% Plot the imported data:--------

figure; clim = 3; % standardize color limits

for ic = 1:size(logersp,3)
    sbplot(row,col,ic);
    imagesclogy(timevals, logfreqs, logersp(:,:,ic)); % see tftopo
    set(gca,'clim', [-clim clim]); % adjust the color limits
    set(gca,'ydir','norm'); % plot low freqs at the bottom
end;

cbar; % include a colorbar

% plot the mean for comparison:
    sbplot(row,col,ic+1)
    imagesclogy(timevals, logfreqs, mean(logersp,3));
Task 3: PLOT individual ERSPs

Single-condition ERSPs from all ICs in a single cluster
STUDY analysis, Part 1

Task 1
Script STUDY plotting functions

Task 2
STUDY structure details

Task 3
Load/plot data from commandline

Task 4
Apply significance limits

Task 5
Time/frequency analysis within cluster

Exercise...
Task 4: Requires access to .mat files

% Load ERSP data as a .mat file

% Variables:
cond = 1;  % which experimental condition
subj = 1;  % which subject

% files are all .mat format:
load_string = [basedir,subjs{subj},'/',...
    subjs{subj},'_',setnames{cond}(1:end-4),'.icaersp'];

% actual string: '.../STUDY/S01/S01_attend1_pos1.icaersp'

ERSPdata = load('-mat',load_string);
Task 4: Raw data structure

>> ERSPdata

ERSP dB data
- comp1_ersp: [126x200 single]
- comp2_ersp: [126x200 single]

dB baseline
- comp1_erspbase: [1x126 single]
- comp2_erspbase: [1x126 single]

Bootstrap limits
- comp1_erspboot: [126x2 single]
- comp2_erspboot: [126x2 single]

126 frequency bins
- freqs: [1x126 double]
- times: [1x200 double]

Datatype: 'ERSP'
- datatype: 'ERSP'

Parameters: {1x26 cell}
- parameters: {1x26 cell}

Data file:
- datafile: [1x57 char]
Task 4: ERSP significance limits

>> ERSPdata
comp1_ersp: [126x200 single]
comp1_erspbase: [1x126 single]
comp1_erspboot: [126x2 single] % (freqs x lower/upper limits)
%------------------------------------------------------------
% create min and max limit matrices:-------------------------

minmask = repmat(ERSPdata.comp1_erspboot(:,1),... 
[1 size(ERSPdata.comp1_ersp,2))]);

maxmask = repmat(ERSPdata.comp1_erspboot(:,2),... 
[1 size(ERSPdata.comp1_ersp,2))]);
%------------------------------------------------------------

sig_ersp = ERSPdata.comp1_ersp; % extract relevant ERSP

% zero out values within significance limits:--------------
sig_ersp(find(sig_ersp > minmask & sig_ersp < maxmask)) = 0;

% plot the results------------------------------------------
figure; imagesclogy(ERSPdata. times, ERSPdata. freqs, sig_ersp);
%
Task 4: Plot a single IC ERSP

UN-masked

Masked (p = .01)
STUDY analysis, Part 1

Task 1
Script STUDY plotting functions

Task 2
STUDY structure details

Task 3
Load/plot data from commandline

Task 4
Apply significance limits

Task 5
Time/frequency analysis within cluster

Exercise...
Task 5: Mean theta power across conditions

```matlab
clust = 13;  % cluster to analyze
tlims = [250 500];  % time limits (ms)
flims = [3 6];  % frequency limits (Hz)

for cond = 1:4
    for ic = 1:length(STUDY.cluster(clust).comps)
        setidx = STUDY.cluster(clust).setinds{cond}(ic);
        comp = STUDY.cluster(clust).comps(ic);
        [logersp(:, :, ic), logfreqs, timevals, params, baseersp] = ...
            std_readersp(ALLEEG, setidx, comp, tlims, flims);
        tfdat(cond, ic) = mean(mean(logersp(:, :, ic)));
        leg{ic} = ['IC ', num2str(STUDY.cluster(clust).comps(ic))];
    end;
end;
```
Task 5: Mean theta power across conditions

```matlab
figure;
ph = bar([1:4], tfdat);
set(gca, 'xticklabel', STUDY.condition);
legend(leg)
title('Theta power between 250 and 500 ms post-stimulus');
```

Quickly assess inter-subject variability and condition effects
Exercise

- **Novice**
  - Script a loop through clusters to plot an activity measure(s) of your choice (ie, component properties, ERSP, spectra, ERP, etc)
  - use 'eegh' to recover EEGLAB plotting function commands

- **Intermediate**
  - Plot an activity measure of your choice (ie, spectra, ERSPs, ITC, etc.) for all members of a cluster from the commandline:
    - Use `std_readersp()` or analogous STUDY function

- **Advanced**
  - Plot masked ERSPs for all members of a cluster
    - use the `load('-mat',load_string)` command
  - Plot mean power in a small time/frequency window across all ICs and conditions for a single cluster

** SCRIPT AVAILABLE ON THE COMMAND LINE
practicum_10.m**