STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

Exercise...
STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

Exercise...
Memory options

Memory options should change when using STUDY vs single dataset.
Build a STUDY
Build a STUDY, cont'd

![Image of EEGLAB dataset creation interface]
Edit dataset info
ICs to cluster
% Open eeglab: 

[ALLEEG EEG CURRENTSET ALLCOM] = eeglab;

% Set memory options: 

pop_editoptions( 'option_storedisk', 1, 'option_savetwofiles', 1,…
'option_saveica', 1, 'option_single', 0, 'option_memmapdata',…
0, 'option_computica', 1, 'option_scaleicaracs', 0,…
'option_rememberfolder', 1);

% saves a file 'eeg_options.m' to your current working directory

% Initialize EEGLAB/STUDY variables: 

STUDY = []; CURRENTSTUDY = 0; ALLEEG=[]; EEG=[]; CURRENTSET=[];

Most important option: 
- Allows only one dataset to be loaded at once. 
- Most STUDYs are too big to have all data loaded at once.
% Faster alternative to building a STUDY manually
% Example STUDY: 13 subjects, 3 conditions

% Define variables:

basedir = 'C:\...\EEGLAB_WORKSHOP\STUDY\';
setnames = {'Memorize.set','Ignore.set','Probe.set'};
subjs = {'S01','S02','S03','S04','S05','S06','S07',...
'S08','S09','S10','S11','S12','S13'};
studynname = 'Sternberg';
taskname = 'Sternberg';
savename = 'stern.study';
Define variables

% concatenate string variables:

[]  % strings inside brackets will be concatenated

dataset = [basedir,subjs{subj},'\',setnames{cond}];

C:\EEGLAB_Workshop\STUDY\S01\Memorize.set
% Now loop through subjects and add to the STUDY:
index = 1; % initialize STUDY index
for subj = 1:length(subjs) % for each subject
    for cond = 1:length(setnames) % for each condition
        dataset = [basedir,subjs{subj},'\',setnames{cond}];
        [STUDY ALLEEG] = std_editset( STUDY, ALLEEG,...
            'name', studyname, 'task', taskname,...
            'commands',{['index',index,'load',dataset],...
                ['dipselect',0.15],['subject',subjs{subj}],...
                ['condition',conds{cond}]}),...
            'inbrain', 'on', 'updatedat', 'off',...
            'savedat', 'off', 'filename', [basedir, savename]);
        index = index + 1;
        CURRENTSTUDY = 1; EEG = ALLEEG; CURRENTSET = [1:length(EEG)];
        [STUDY, ALLEEG] = std_checkset(STUDY, ALLEEG);
    end;
end;
eeglab redraw
STUDY structure

STUDY =

    name: 'Sternberg'
    task: 'Sternberg'
    datasetinfo: [1x39 struct]
    notes: ''
    filename: 'stern.study'
    filepath: 'C:\Users\julie\Documents\Workshops\Finland\STUDY'
    history: [1x7332 char]
    subject: {1x13 cell}
    group: {}
    session: []
    condition: {'ignore' 'memorize' 'probe'}
    setind: [3x13 double]
    etc: [1x1 struct]
    preclust: [1x1 struct]
    cluster: [1x1 struct]
    changrp: [1x71 struct]
    saved: 'yes'

>>
Subject info in STUDY structure

```matlab
>> STUDY.datasetinfo
ans =
1x39 struct array with fields:
    filepath
    filename
    subject
    session
    condition
    group
    index
    comps

>>
```

Gives information for each dataset of each subject

Each DATASET (NOT each subject) has a unique index
STUDY clustering overview

**STEP 1**
Build a STUDY

**STEP 2**
Precompute the data

**STEP 3**
Precluster the data

**STEP 4**
Cluster the data

*Exercise...*
Precompute data measures

| Study set: At
| **Study filename:** |
| **Study task name:** |
| **Nb of subjects:** |
| **Nb of conditions:** |
| **Nb of sessions:** |
| **Nb of groups:** |
| **Epoch consistency:** | yes |
| **Channels per frame:** | 31 |
| **Channel locations:** | yes |
| **Clusters:** | 1 |
| **Status:** | Ready to precluster |
| **Total size (Mb):** | 30.4 |

Precompute component measures
- Build preclustering array
- Cluster components
- Edit/plot clusters
Precompute data measures

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

```matlab
[STUDY ALLEEG] = std_precomp(STUDY, ALLEEG, 'components',...
'erp', 'on', 'rmbase',[-200 0] , 'scalp', 'on', 'spec',...
'on', 'specparams',{}, 'ersp', 'on', 'erspparams',...
{'cycles',[3 0.5] , 'nfreqs',100, 'frels',[3 70] ,...
'alpha',0.01}, 'itc', 'on');
```
STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

Exercise...
Precluster the data

![EEGLAB v6.0b interface showing study settings and preclustering options](image)
parentclust = 1; % cluster 1 is always full parent cluster
[STUDY ALLEEG] = std_preclust(STUDY, ALLEEG, parentclust,{'spec','npca',5,...
    'norm',1,'weight',1,'freqrange',[3 25]},{'erp','npca',6,'norm',1,...
    'weight',1, 'timewindow',[0 400]},{'scalp','npca',10,'norm',1,'weight',1,...
    'abso',1},{'dipoles','norm',1,'weight',10},{'ersp','npca',20,...
    'freqrange',[3 30], 'timewindow',[0 600], 'norm',1,'weight',1},{'itc',...
    'npca',6,'freqrange',[3 30], 'timewindow',[0 400], 'norm',1, 'weight',1});
Choosing data measures

What measure(s) should you use?

• It depends on your final cluster criteria…
  - If for example, your priority is dipole location,
    then cluster only based on dipole location…

But consider:

  - What is the difference between these two components?

![Image of brain头皮图](image1.png)
Choosing data measures

Similar dipole location, very different orientation.

Obvious dramatic effect on scalp map topography:

But, do they perform the same functions?
Choosing data measures

ERPs seem different...
Choosing data measures

Spectra are similar, but they have variable responses to different conditions...
Choosing data measures

ERSPs have some similar features…
Choosing data measures

What data measures should you use?

It depends…

• broadly-matched ICs: use many/all of the measures.

• specifically-matched ICs: use one/few of the measures.
What should clusters look like?

ICs clustered by dipole location and ERSP activity.
Plot STUDY dipoles

% std_dipoleclusters() variables:

clusters = [3:length(STUDY.cluster)]; % clusters to plot
title = 'Cluster Dipoles'; % figure title
plot_params = [2,2,1]; % [nrows,ncols,subplot]
views = [1,2,3,4]; % 1=top,2=side,3=rear,4=oblique
cols = hsv(length(clusters));

% std_dipoleclusters function call:

std_dipoleclusters(STUDY,ALLEEG,'clusters',clusters,...
              'title',title,'viewnum',views,'rowcolplace',plot_param,...
              'centroid','off','colors',cols);
Precluster schematic

Mean ERSPs
ICs (all subj)

Mean spectra
ICs (all subj)

Power spectrum
ICs (all subj)

PCA

PC templates
ICs (all subj)

PC weights
concatenate

3D dipole position
ICs (all subj)

Rel. weight

ERSP (time/freq)

PC weights

# PCs
Precluster: Use singular values from PCA

%% Do it yourself:
%% Load all ERSP data
%% decompose with PCA
%% plot singular values
(See code in ‘Tutorial_7_BuildSTUDY.m’)

Normalized singular values

~ relative variance of principal components

10% of max singular value
STUDY clustering overview

STEP 1
Build a STUDY

STEP 2
Precompute the data

STEP 3
Precluster the data

STEP 4
Cluster the data

Exercise...
nclusts = 25; % choose # of clusters to create

[STUDY] = pop_clust(STUDY, ALLEEG,'algorithm','kmeans','clus_num',nclusts);
Plot/edit clusters

Study name: 'Sternberg' (336 of 336 components clustered)

Select cluster to plot
- All cluster centroids
- ParentCluster 1 (336 ICs)
  - Cls 2 (17 ICs)
  - Cls 3 (6 ICs)
- Plot scalp maps
- Plot dipoles
- Plot ERP(s)
- Plot spectra
- Plot ERSPs
- Plot ITCs
- Plot cluster properties

Select component(s) to plot
- 'Cls 2' comp. 1 (S01 IC21)
- 'Cls 2' comp. 2 (S03 IC21)
- 'Cls 2' comp. 3 (S03 IC25)
- 'Cls 2' comp. 4 (S04 IC19)
- Plot scalp map(s)
- Plot dipole(s)
- Plot ERP(s)
- Plot spectra
- Plot ERSP(s)
- Plot ITC(s)
- Plot component properties

- Create new cluster
- Rename selected cluster
- Merge clusters

- Reassign selected component(s)
- Remove selected outlier comps.
- Auto-reject outlier components

Help
Cancel
Ok
Plot cluster data

Plot mean scalp maps for easy reference.
Exercise

• **Novice**
  - Open stern.study and practice plotting the existing clusters

• **Intermediate**
  - Script a loop to build a STUDY from the commandline
  - Precluster (pre-computation already done) and cluster components using measures of your choice.

• **Advanced**
  - Load raw data measures and run PCA to determine the relative size of PCA dimensions for each data measure.
  - Try preclustering/clustering based on your observations

** All scripts for exercises can be found in:
  ../Scripts/Tutorial_7_BuildSTUDY.m