Introduction to Brain-Computer Interface Design: Practicum

Christian A. Kothe
SCCN, UCSD

In the Meantime…

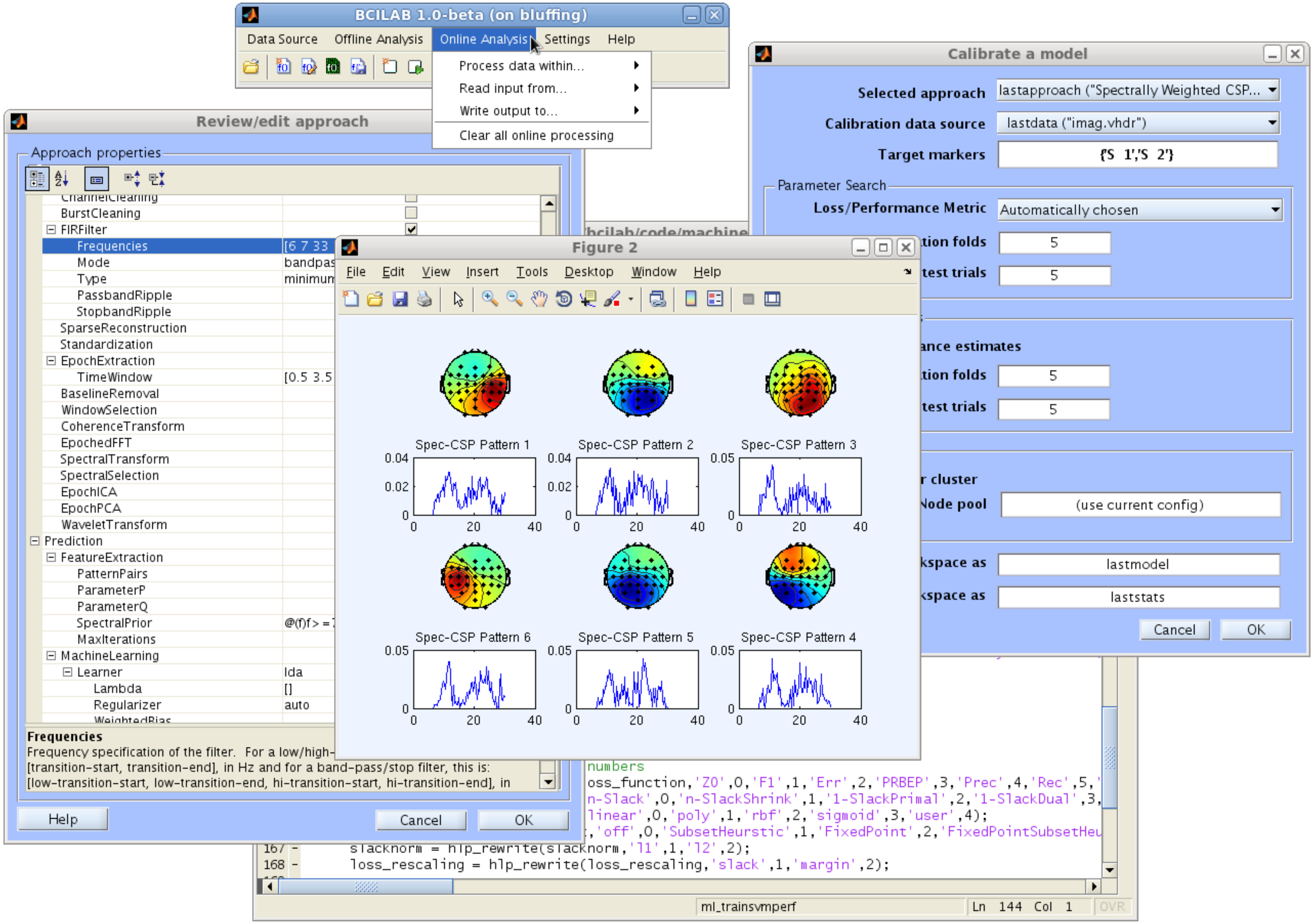
- … please start unzipping your bcilab-1.0b file to some directory on your disk if you haven't done so already.
- Should take 5-10 minutes (if you're on Windows, try to use 7-zip or WinZip/WinRAR).
- Don't put it inside the EEGLAB folder (BCILAB includes an EEGLAB distribution).

Outline

1. Toolbox Intro
2. Concepts and Components
3. Detailed GUI Walkthrough
4. Hands-On ERP Analysis
5. Hands-On Oscillatory Analysis

1 Toolbox Intro

The BCILAB Toolbox



[http://sccn.ucsd.edu/**wiki/BCILAB**](http://sccn.ucsd.edu/wiki/BCILAB)

ftp://sccn.ucsd.edu/pub/bcilab

Context

- Like EEGLAB, but for BCI (and/or cognitive state assessment)
  - Seeding a community
  - Strengthening links between BCI and Neuroscience
- SCCN's in-house tool for BCI problems
  - Main focus: Advanced cognitive monitoring
  - Part of a large US research program (CaN CTA)
  - Funded by ARL (and ONR, Swartz Foundation, …)

Software Environment For:

- **Brain-Computer Interface Design** (Cognitive Monitoring)
- **Methods Research**:
  - Design & rapid prototyping of new methods & methods from literature
  - Offline testing, performance evaluation & batch comparison
  - Simulated online testing
- **Rapid Prototyping**:
  - Real-time use
  - Prototype deployment

Facts & Figures

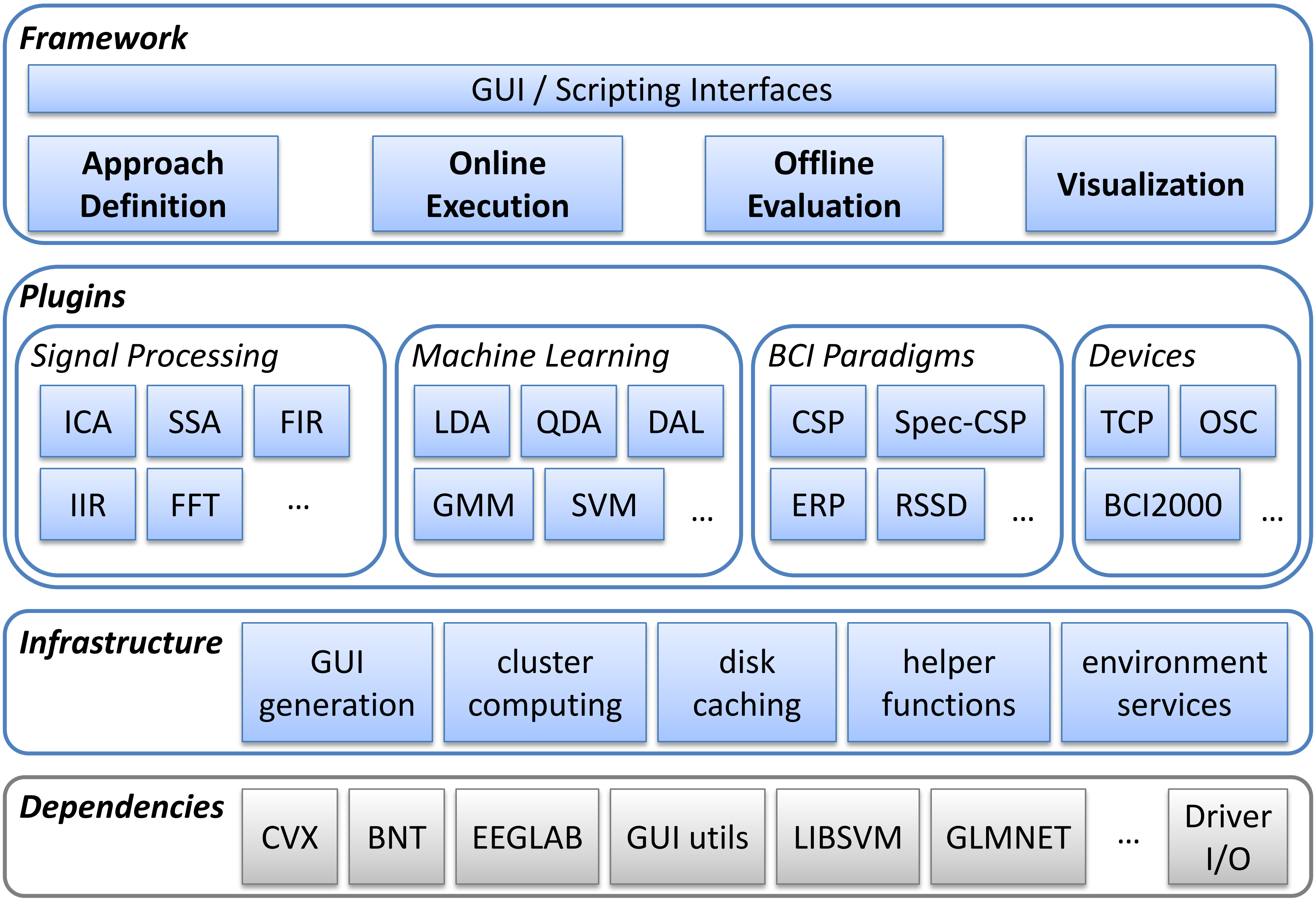
- Developed since 2010 at SCCN, UCSD (primarily by me)
- Precursor was the PhyPA toolbox (Kothe & Zander, 2006-'09)
- Built on top of EEGLAB (Delorme & Makeig, 2004)
- Currently the largest open-source BCI toolbox by methods and algorithms (100+)
- Offline and online processing both in MATLAB, same code base, cross-platform, 32/64bit

Basic Goals

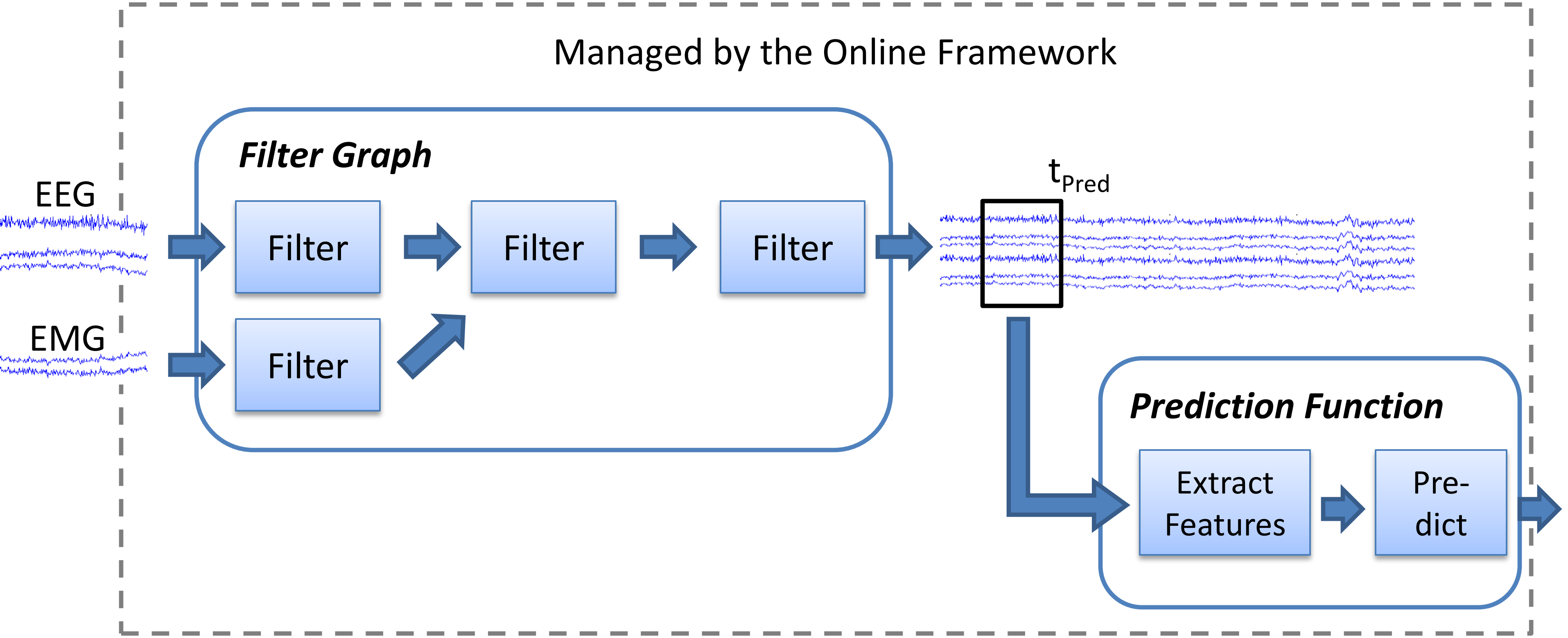
- Provide large array of existing methods to reproduce existing literature – e.g., in benchmarking and comparison studies
- Provide state-of-the-art and novel methods to rapidly set up well-performing BCIs
- Provide plugin frameworks and backend solvers to implement new methods quickly
- GUI for beginners & experimenters, scripting for experts and MATLAB veterans – largely the same feature set
- Allow for both conventional designs (e.g., data flow) and for radically new approaches

2 BCILAB Concepts and Details

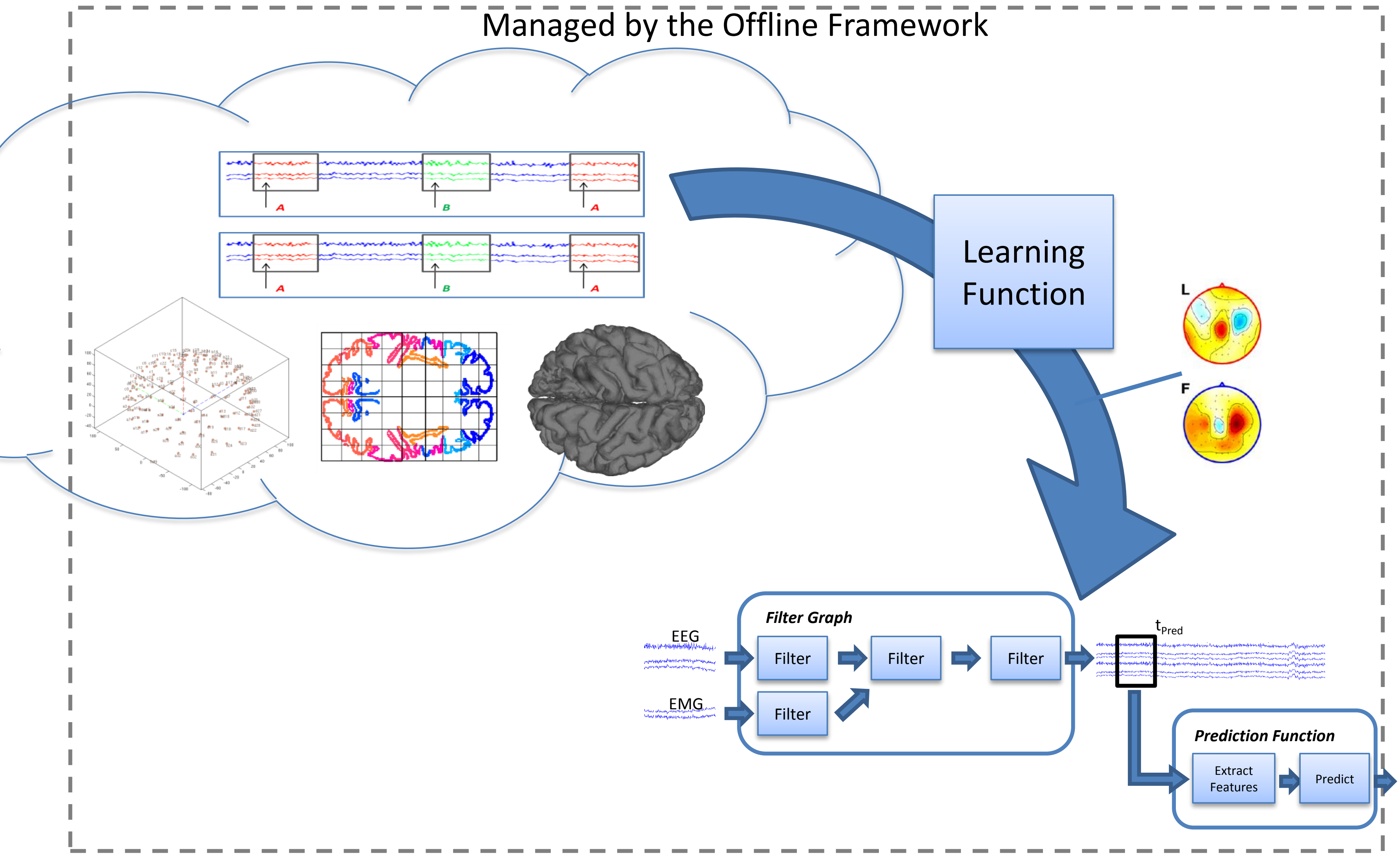
Toolbox Layers



Scope of the Online Framework

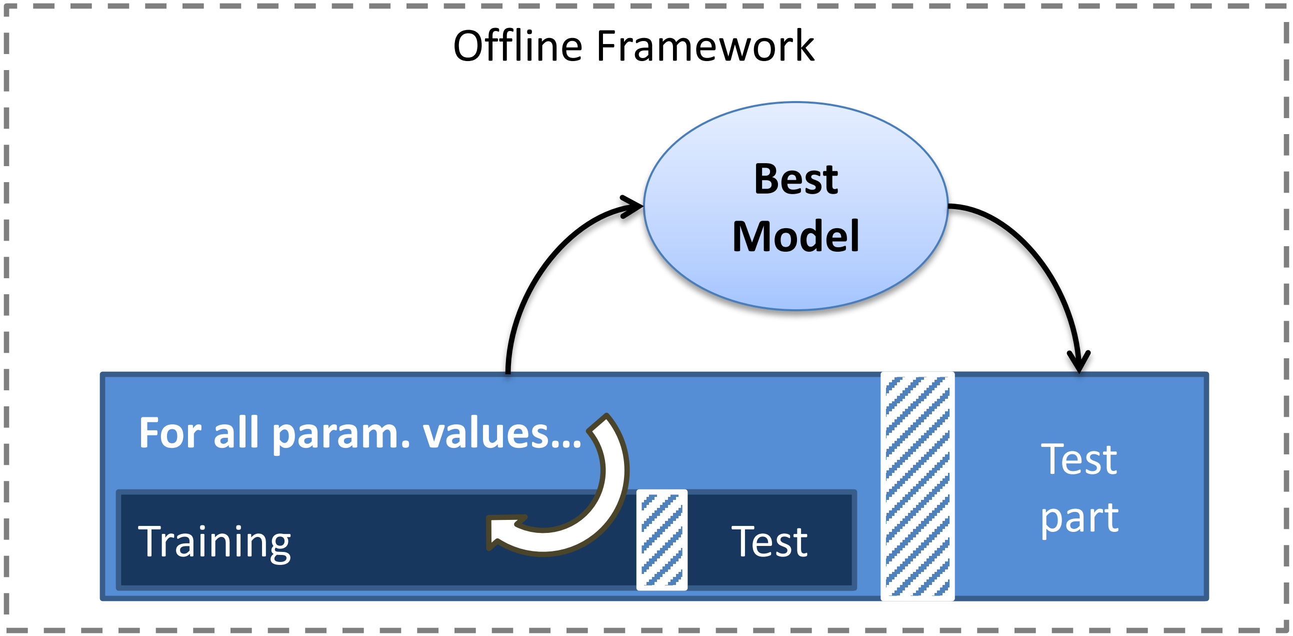


Scope of the Offline Framework



Scope of the Offline Framework

- **Also Covered:** Cross-validation, Grid Search, Nested Cross-Validation



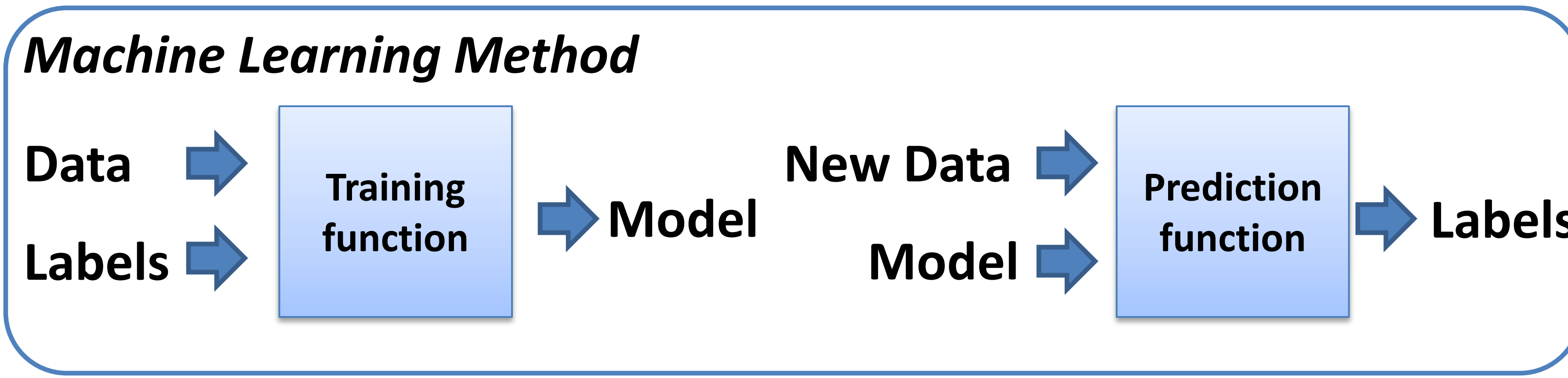
Filter Components

- Filters can operate on continuous signals…

- … or on segmented (“epoched”) signals:

Machine Learning Components

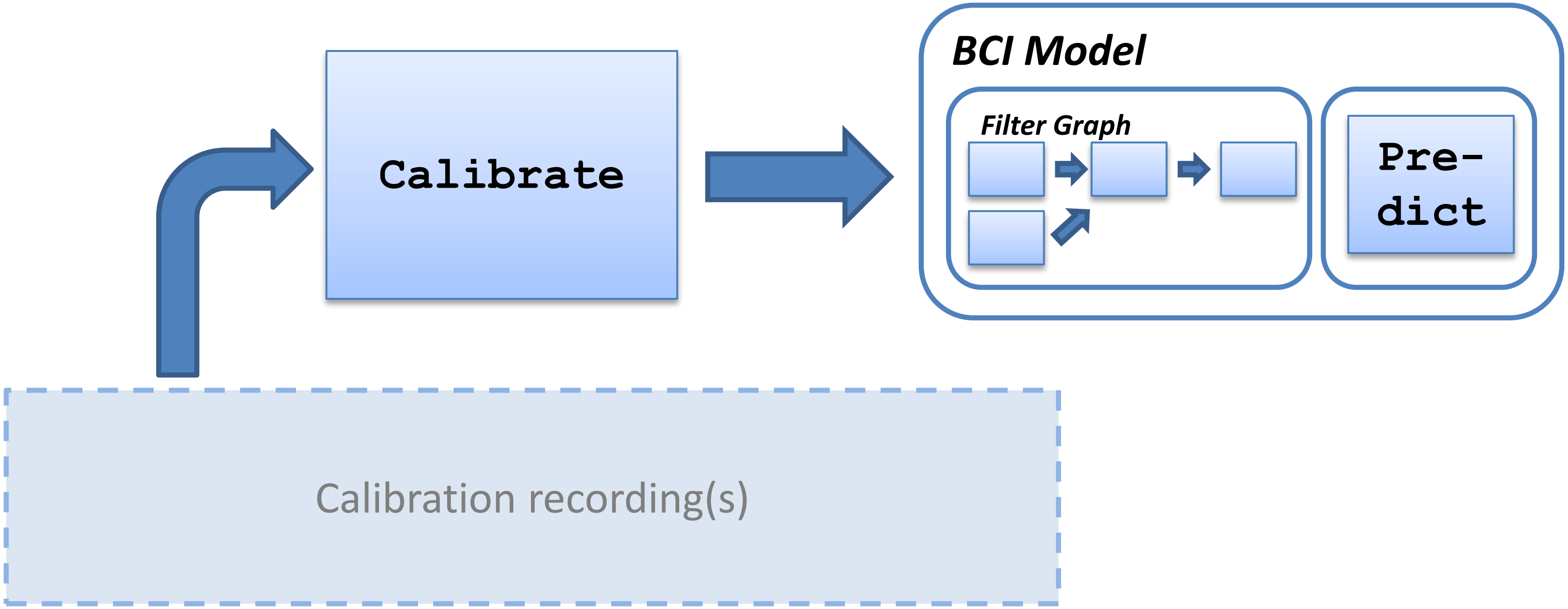
- Machine learning functions come in pairs:




```
M = ml_trainlda(X,y)
p = ml_predictlda(Xnew,M)
```

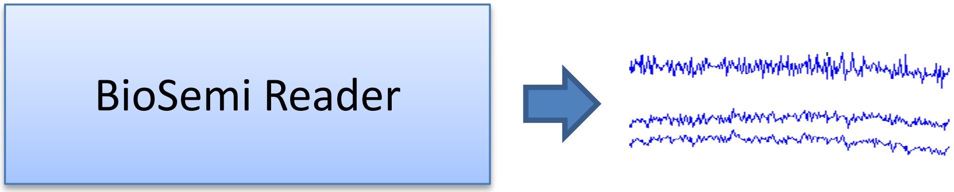
Paradigms Components

- **BCI paradigms** are the coarsest plugin type in BCILAB and *tie all parts of a BCI approach together* (signal processing, feature extraction, machine learning, ...)
- They are invoked by the offline/online framework



Online Reader Components

- Online reader plugins read signals from a source device and make them available in the MATLAB workspace:

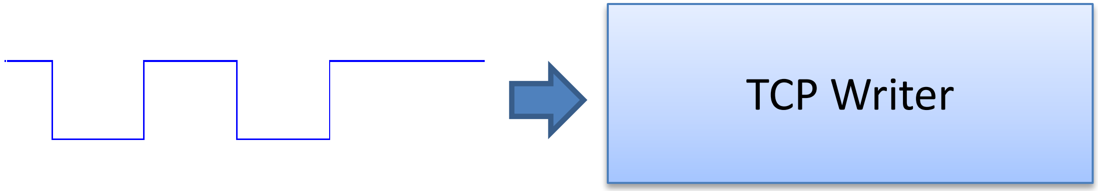


- Example:

```
run_readbiosemi();
```

Online Writer Components

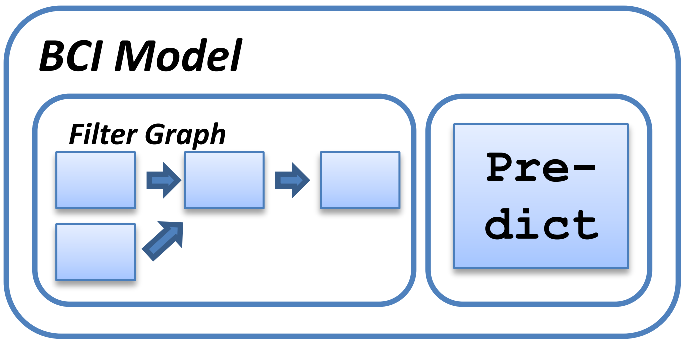
- Online writer plugins write BCI outputs (i.e., predictions) to some external destination:

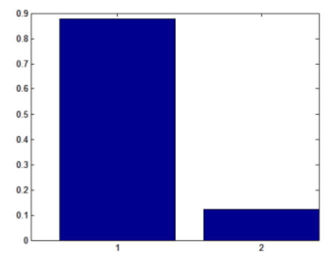


- Example:

```
run_writetcp('mdl','strm','192.168.1.5',12467)
```

Data Representations

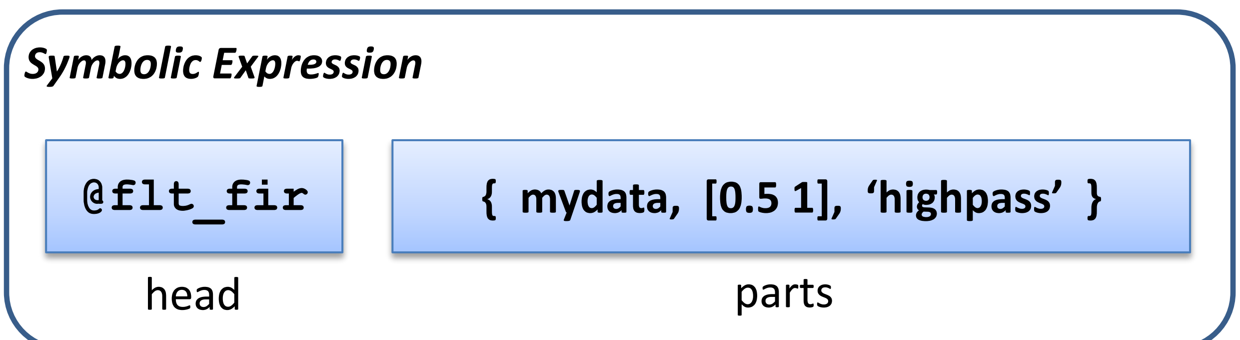




Probability Distributions

$$\begin{bmatrix} f_1 \\ f_2 \\ \vdots \end{bmatrix}$$

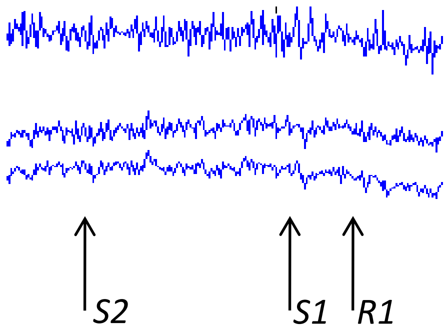
Feature Vectors



Data Representations

Signal

.data

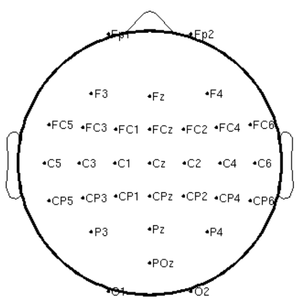


.event

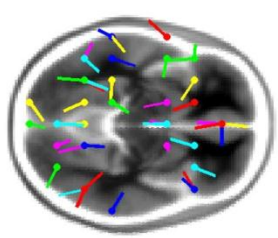
.srate *200Hz*

.xmin *0.0s*

.chanlocs



.dipfit



... (meta-data)

Signal Bundle

.streams

- Signal 1
- Signal 2
- ⋮
- Signal n

... (meta-data)

Dataset Collection

- Bundle 1
- Bundle 2
- ⋮
- Bundle n

Pipeline Notion

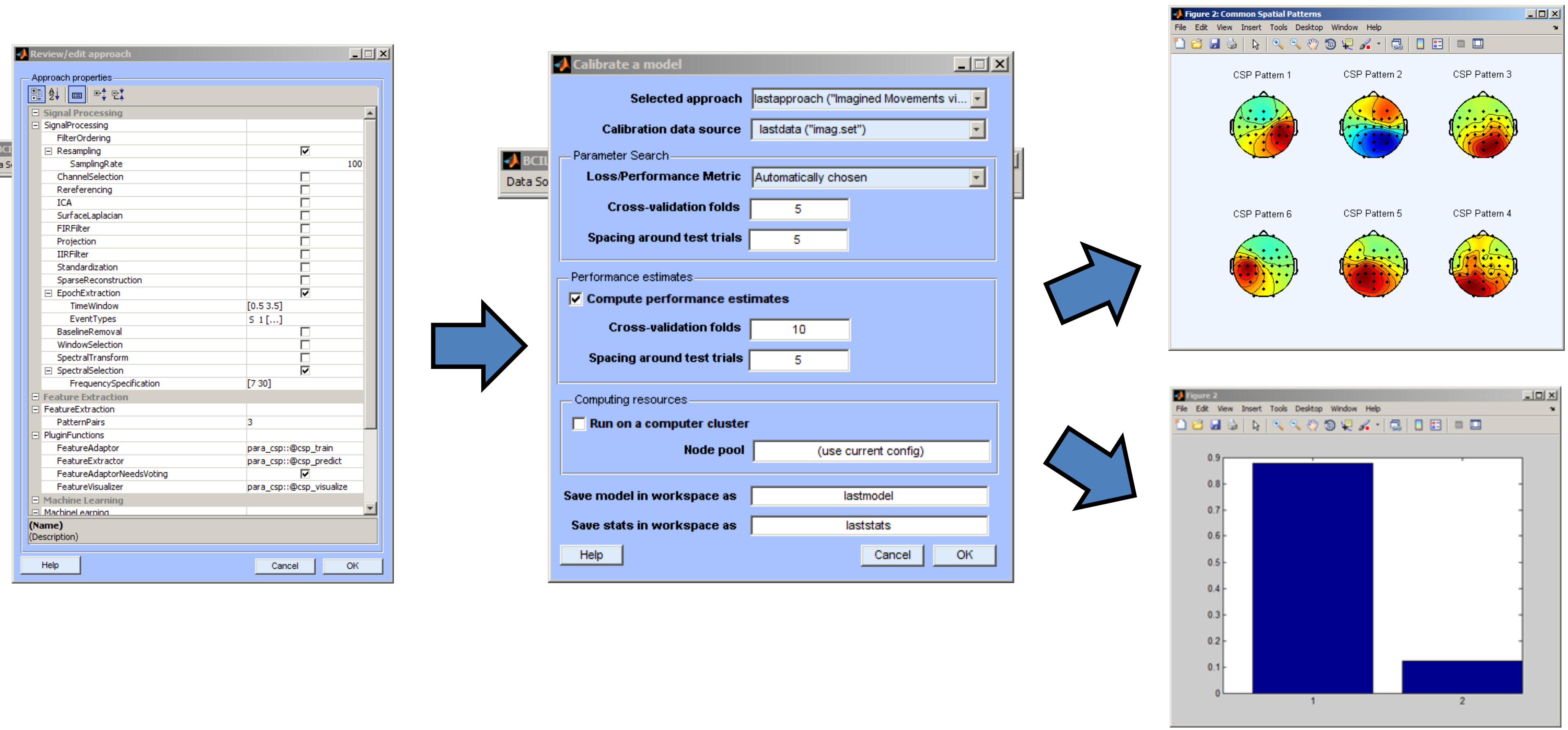
- BCILAB is a framework that resembles a processing pipeline: first configure everything, then apply it to one or more data sets
- **Configuration Inputs:**
  - Mapping between marker type strings and numeric class labels
  - Base BCI Paradigm to execute – "what to run?"
  - Custom parameters for the paradigm
  - Evaluation Scheme – "how to run it?" (e.g., what type of cross-validation)

Pipeline Processes

- **Curate:** bring the input data into standard form
- **Design:** define the computational approach
- **Train:** invoke all steps necessary for training (calibrating) a BCI and estimates performance
- **Predict:** apply a BCI to some data offline
- **Visualize:** visualize BCI model internals
- **Run Online:** apply a BCI online / incrementally
- **Batch Analysis:** perform a series of processing steps, optionally in parallel

3 Detailed GUI Walkthrough

Detailed GUI Walkthrough



Getting Started

System requirements

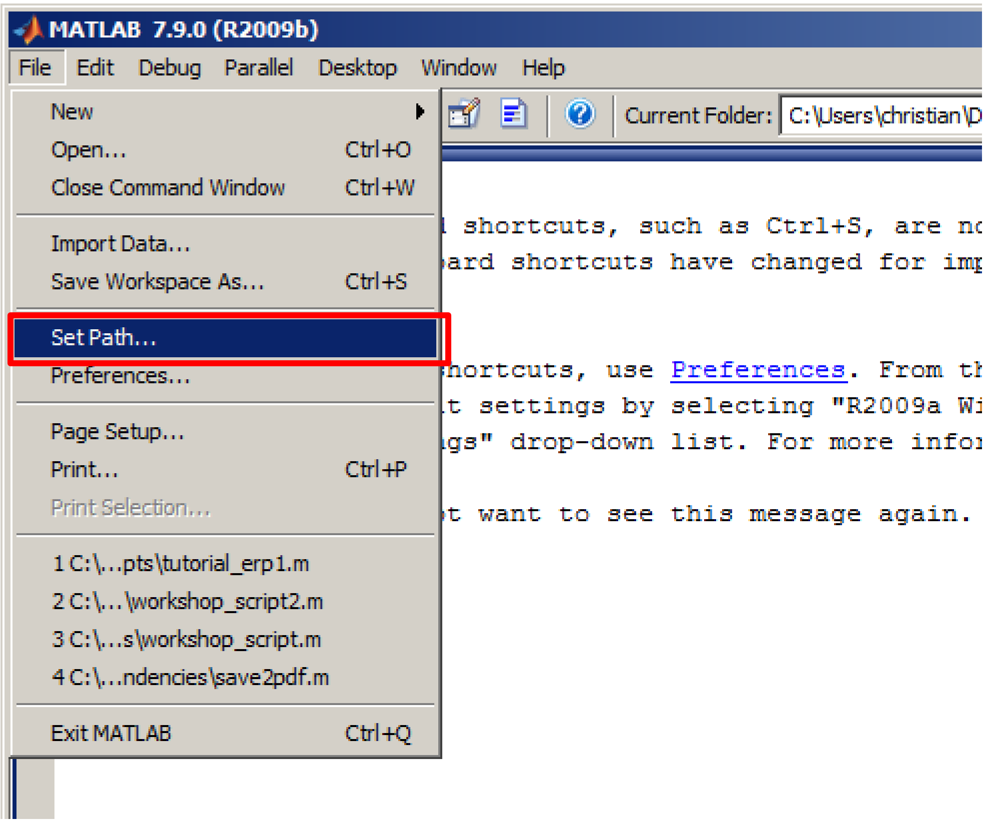
- MATLAB 2008a+
- 1GB+ RAM (better: 2GB+)
- Windows, Linux, or Mac
- For smooth workshop: **No** toolboxes in MATLAB path other than Mathworks toolboxes (or EEGLAB)
- To use certain additional features (not covered today): Signal Processing Toolbox, Statistics Toolbox, Real-time experimentation environment (DataRiver, BCI2000, OpenViBE or your own)
- To use certain advanced features (also not covered today): Correct MEX compiler setting (this requires Microsoft Visual C++ Express under Win64 and Xcode/gcc under Mac)

When Processing your own Data

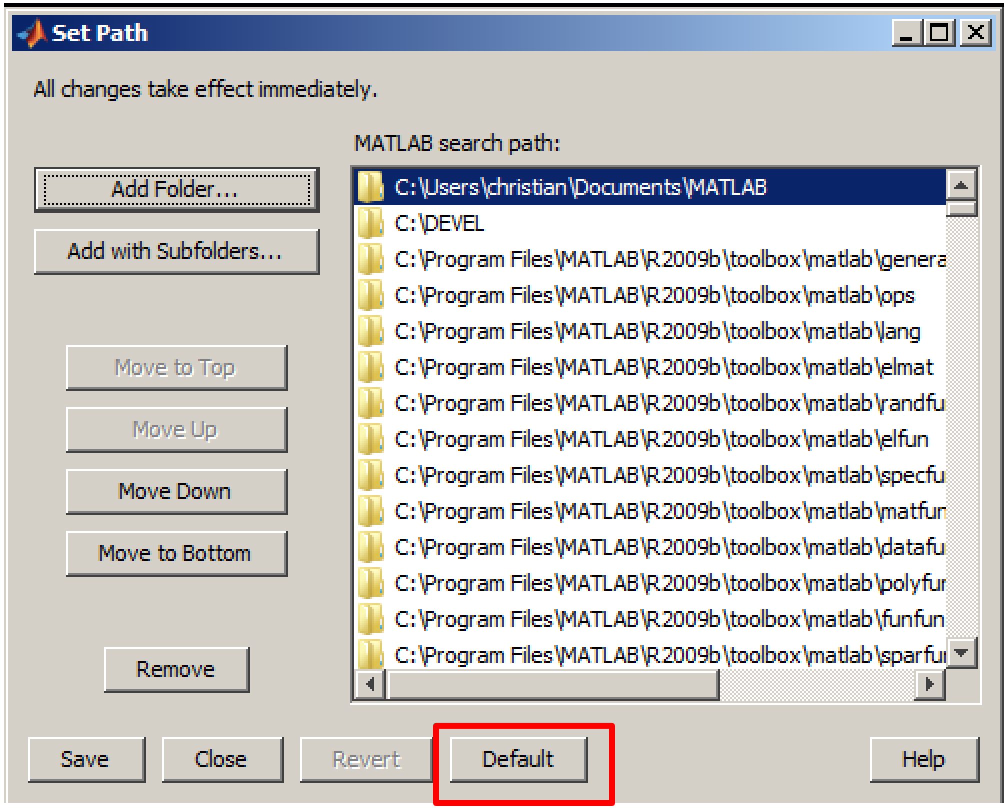
- Note the following requirements:
  - You need proper channel labels (usually the 10-20 labels); 3d locations not necessary
  - You need event markers in your data for the time points with known target condition
  - BCILAB needs raw (unprocessed) data
  - Make sure you have a file format supported by EEGLAB

Rawr!

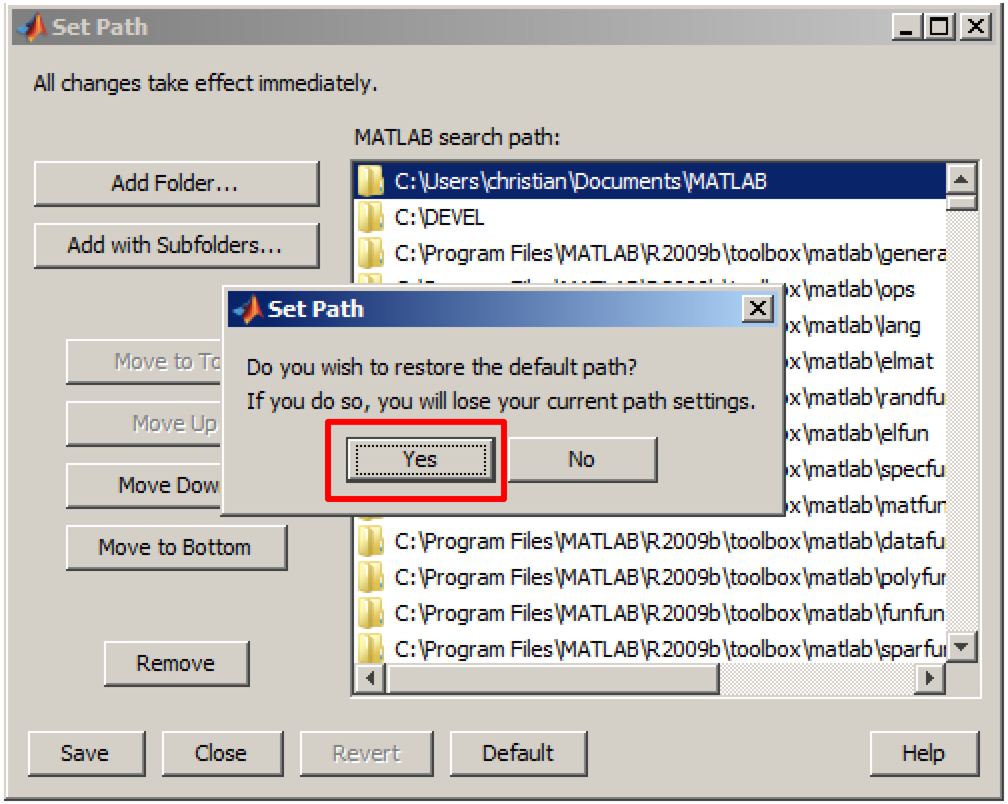
Clearing the Path



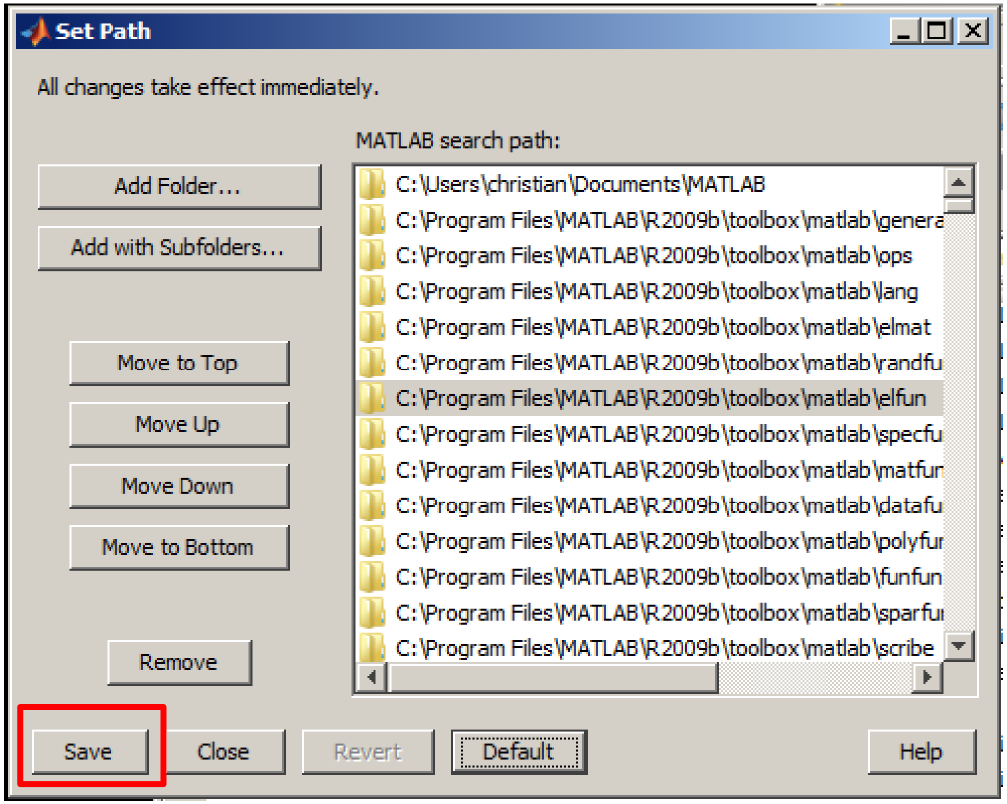
Clearing the Path



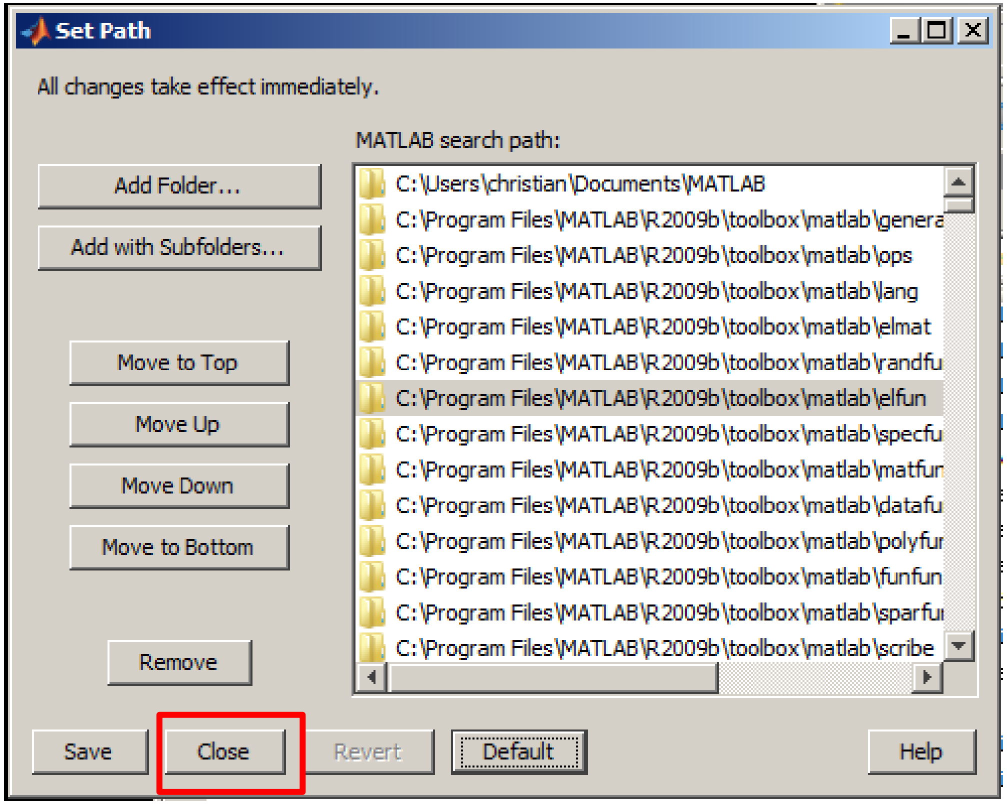
Clearing the Path



Clearing the Path

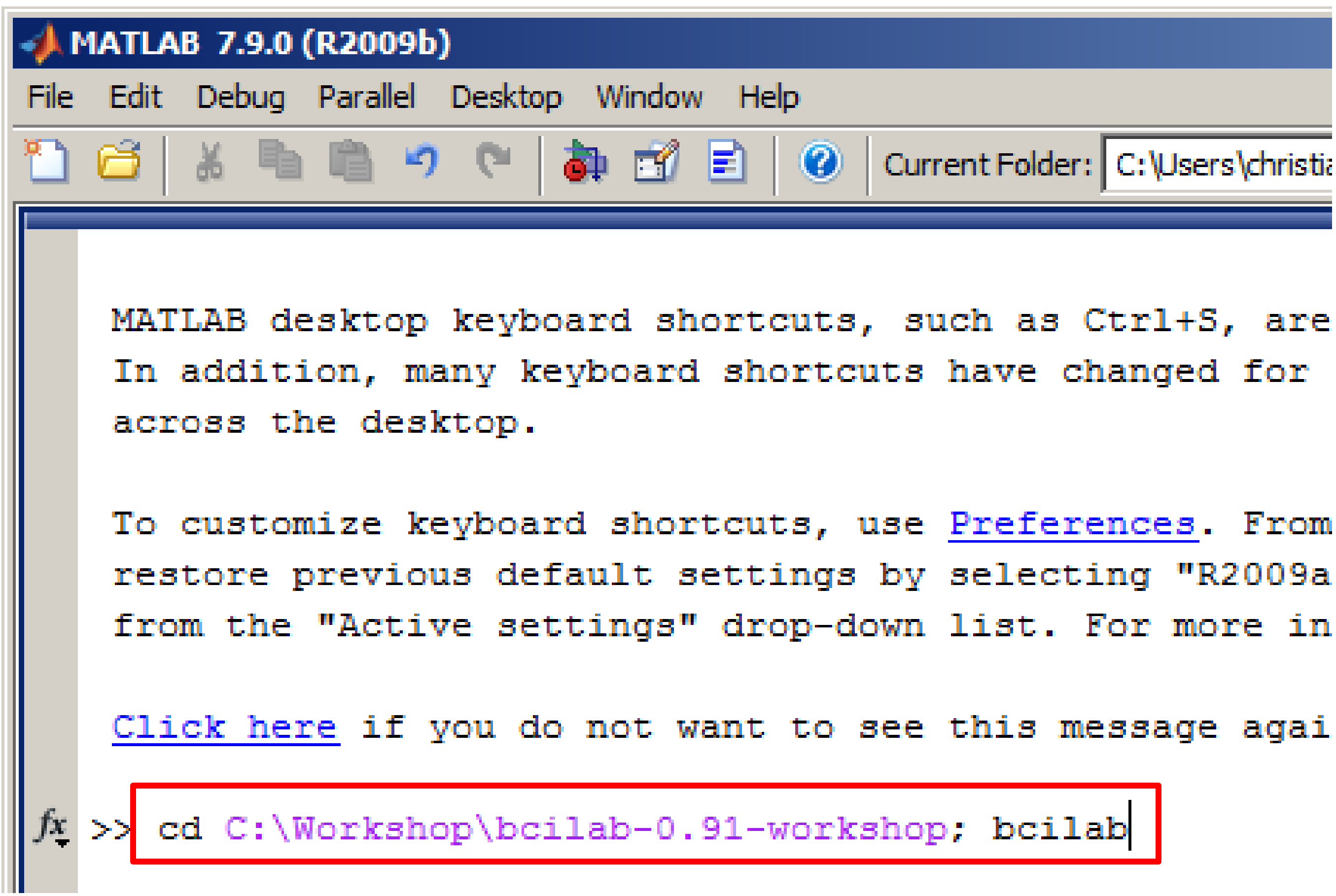


Clearing the Path



Starting the Toolbox

- Type: cd C:\your\path\to\bcilab; bcilab



Starting the Toolbox

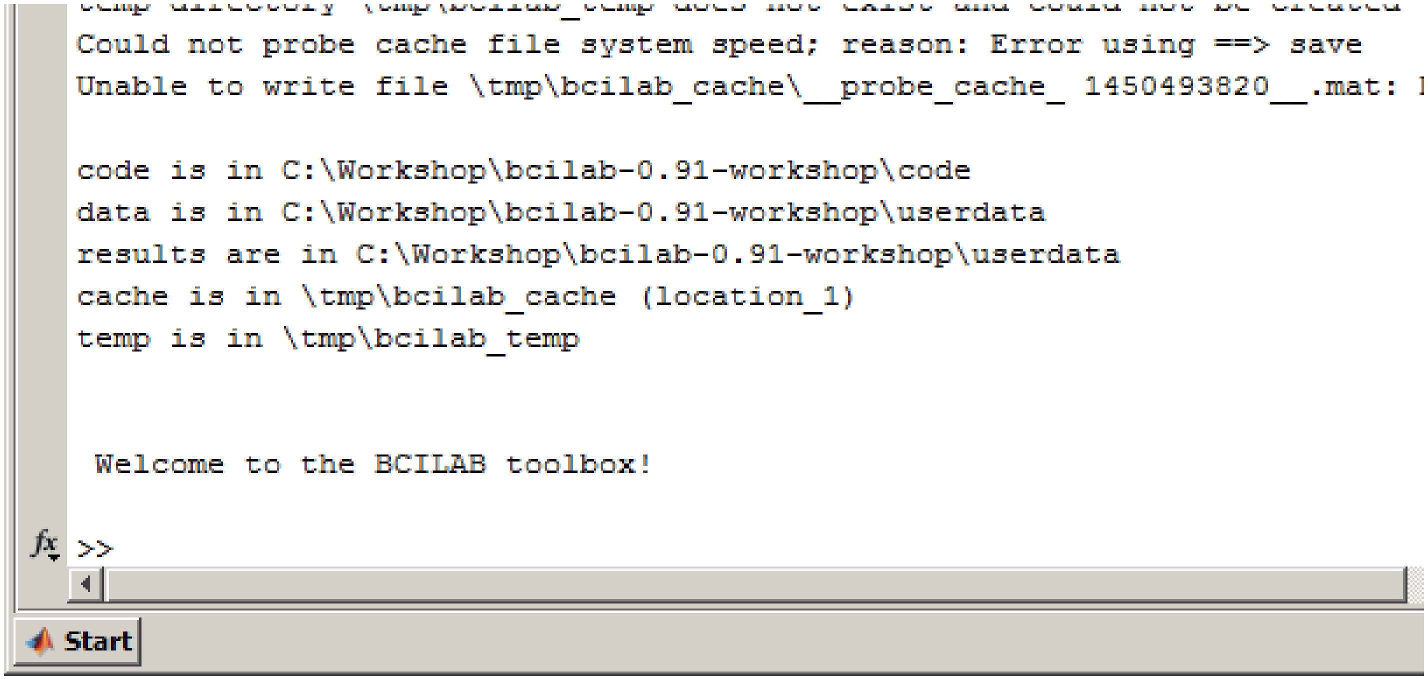
- Or if your path contains spaces, type:
  cd('C:\your\path\to\bcilab‘); bcilab

Starting the Toolbox

- If you have an old (32-bit) Macintosh laptop, BCILAB might ask you some question about compiling functions
  - Just type n (for no) to continue
- If you have things on your MATLAB path that override BCILAB function names, you will get some warnings about it (it's best to remove them from the path)

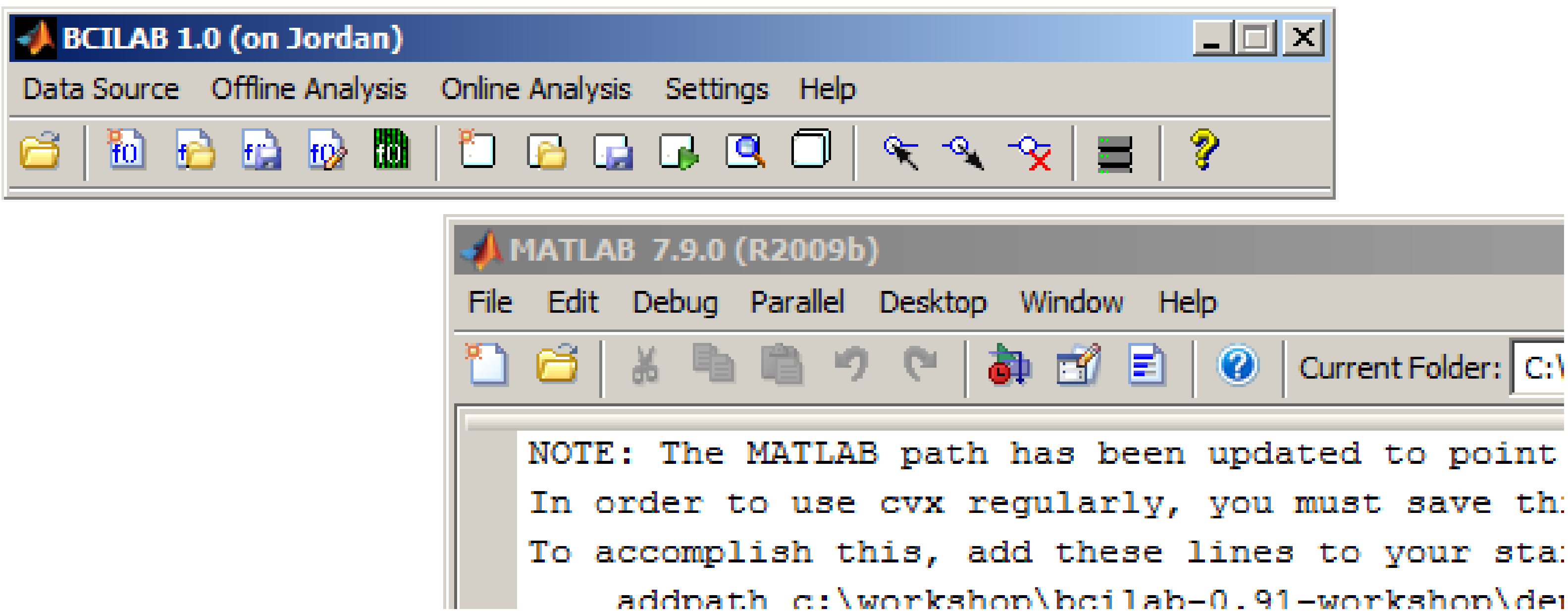
Starting the Toolbox

- You should now see the welcome message

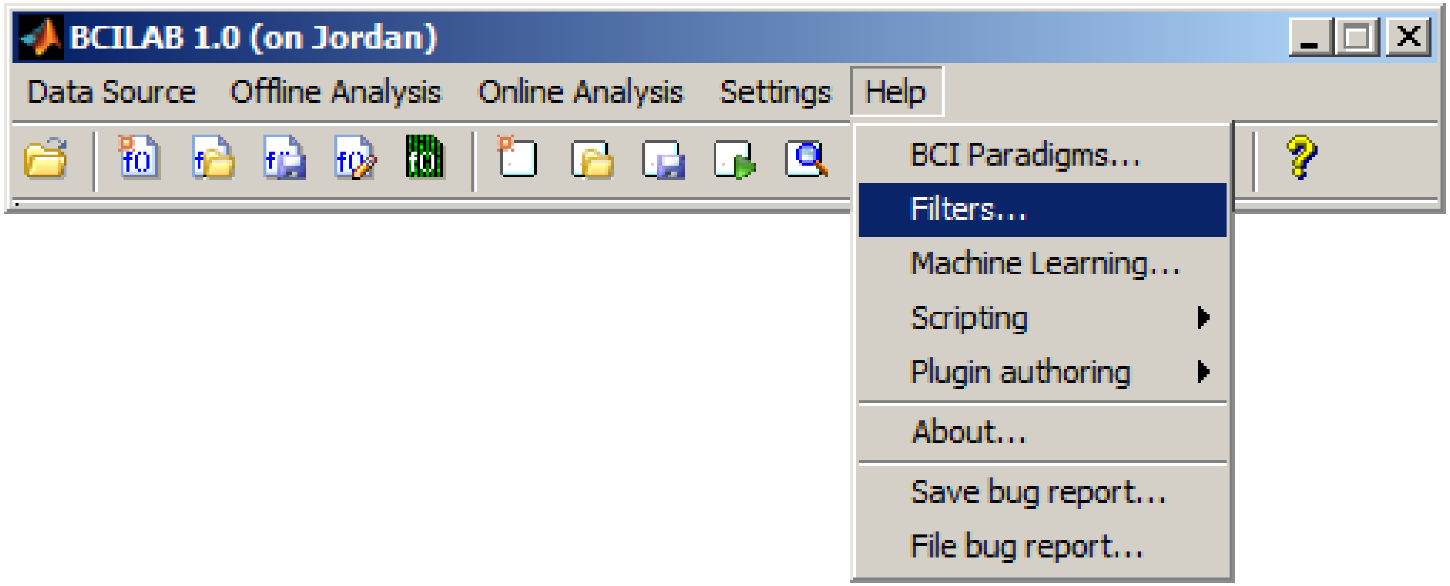


Starting the Toolbox

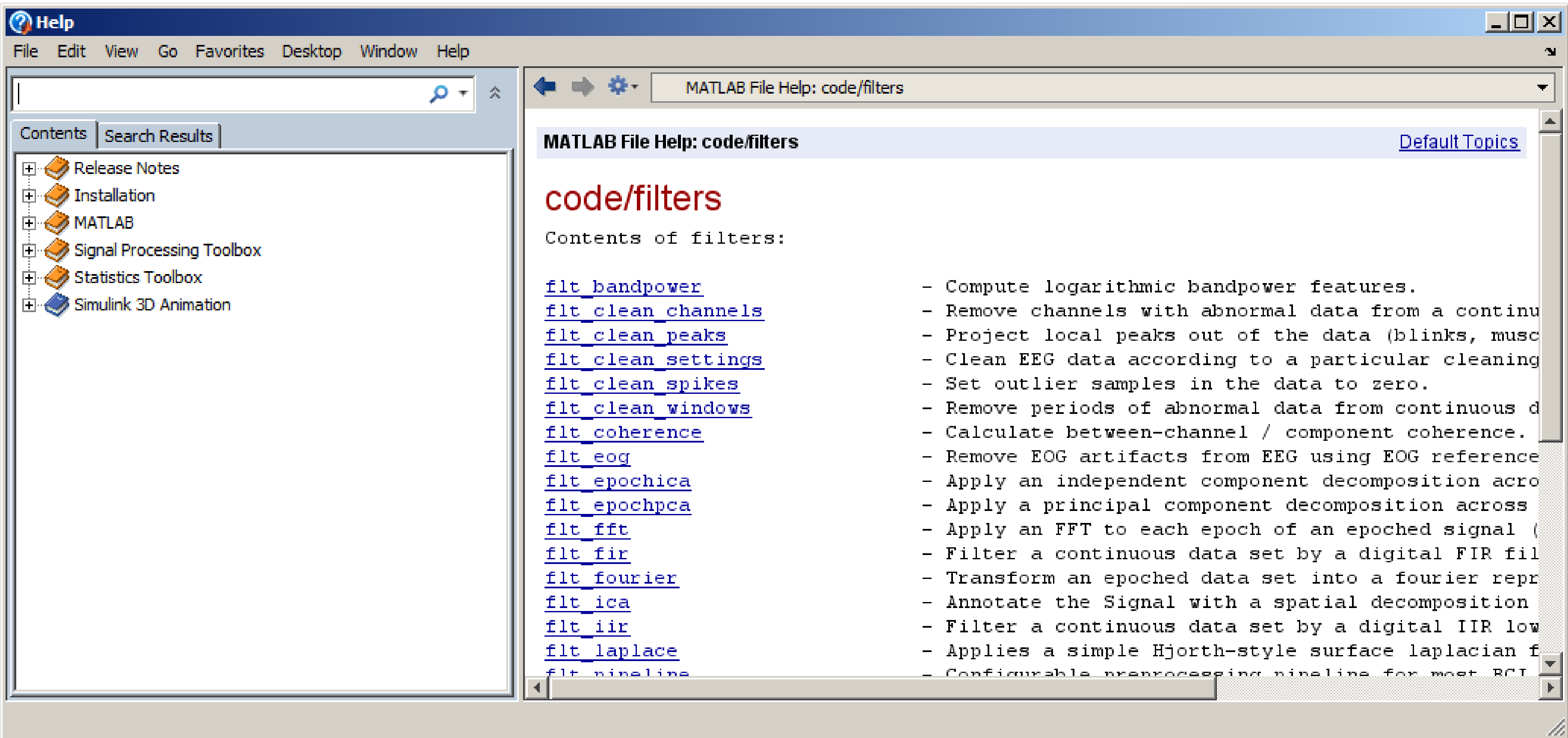
- … and the main menu



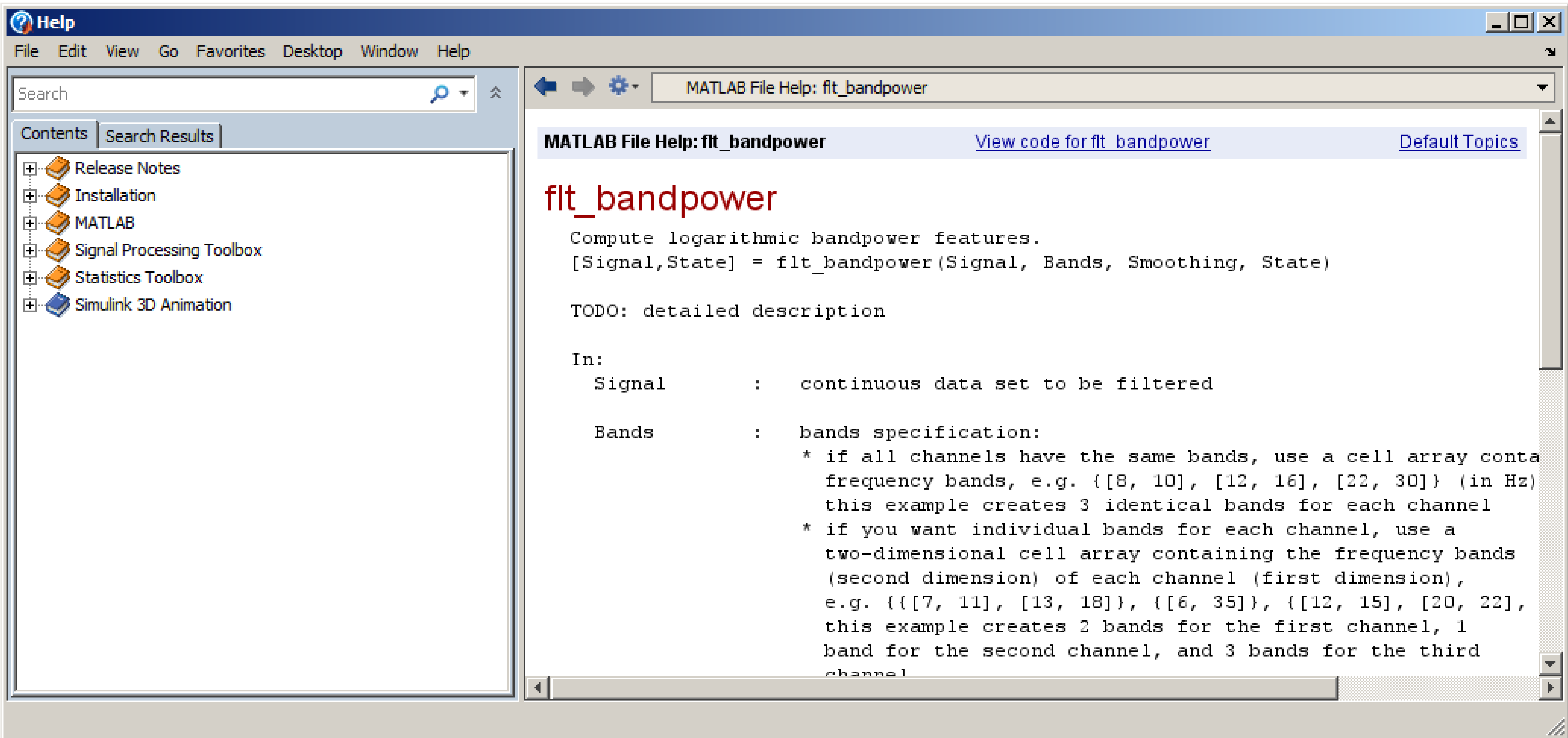
Getting help (if needed)



Getting help (if needed)



Getting help (if needed)



3 Hands-On ERP Analysis

The Data

- Provided by Grainne McLoughlin
- Contains data from a Flanker task
- Two groups of markers:
  - S101, S102: person presses a button and **commits no error**
  - S201, S202: person presses a button and **commits an error**

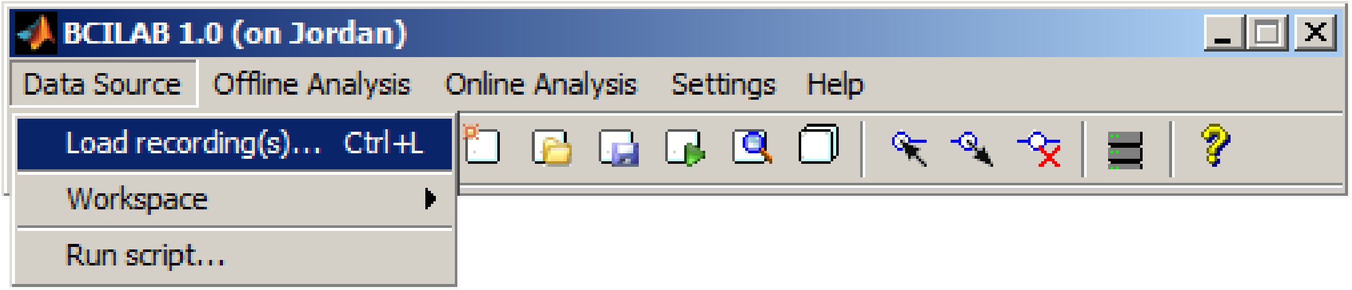
Experimental Task

- **Flanker Task:** The experiment consists of a sequence of ca. 330 trials with inter-trial interval of 2s +/- 1.5s
- At the beginning of each trial, an arrow is presented centrally (pointing either left or right)
- The arrow is flanked by congruent or incongruent “flanker” arrows:

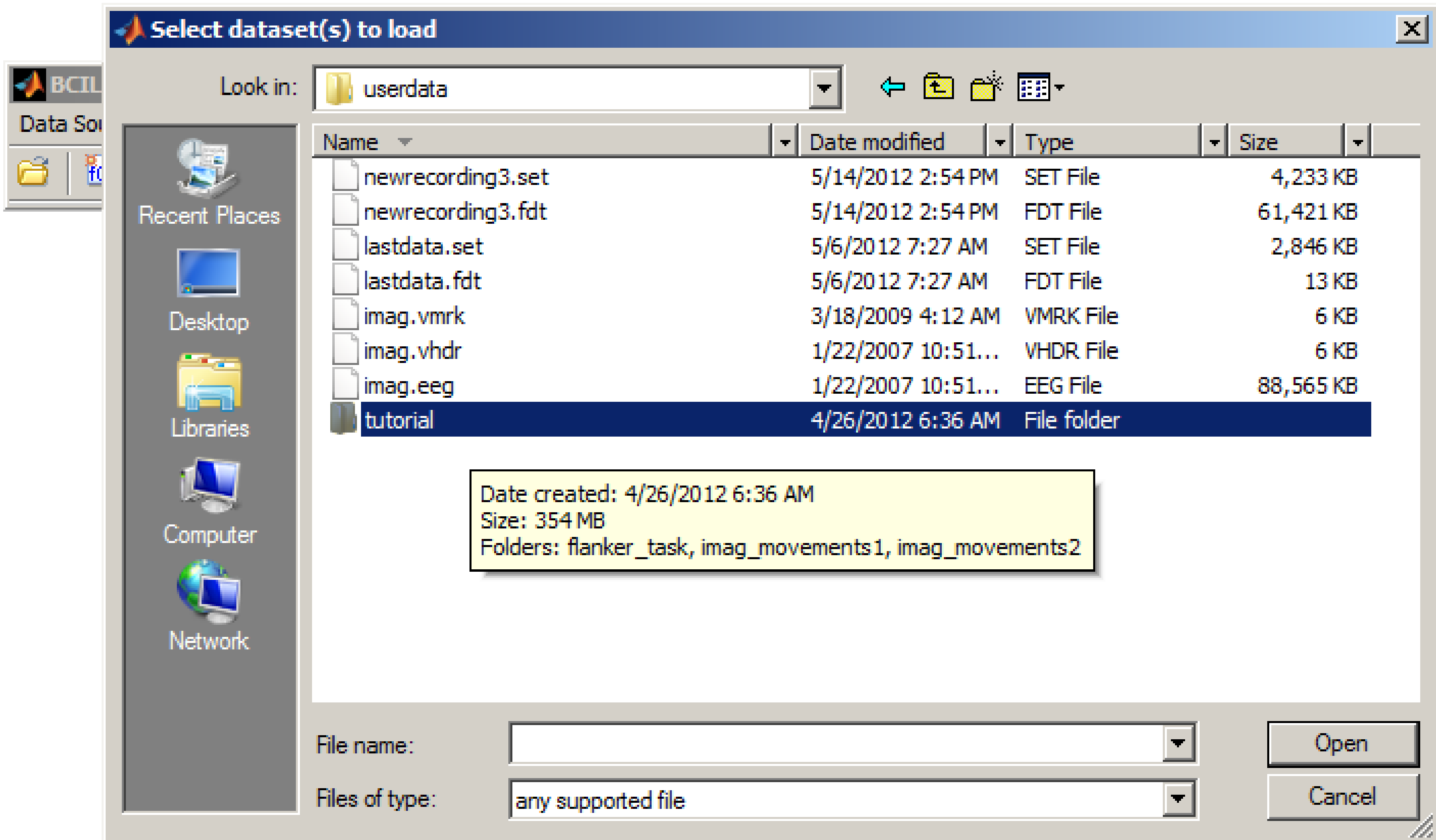
← ← → ← ←

- The subject is asked to press the left/right button, according to the central arrow, and makes frequent errors (25%)

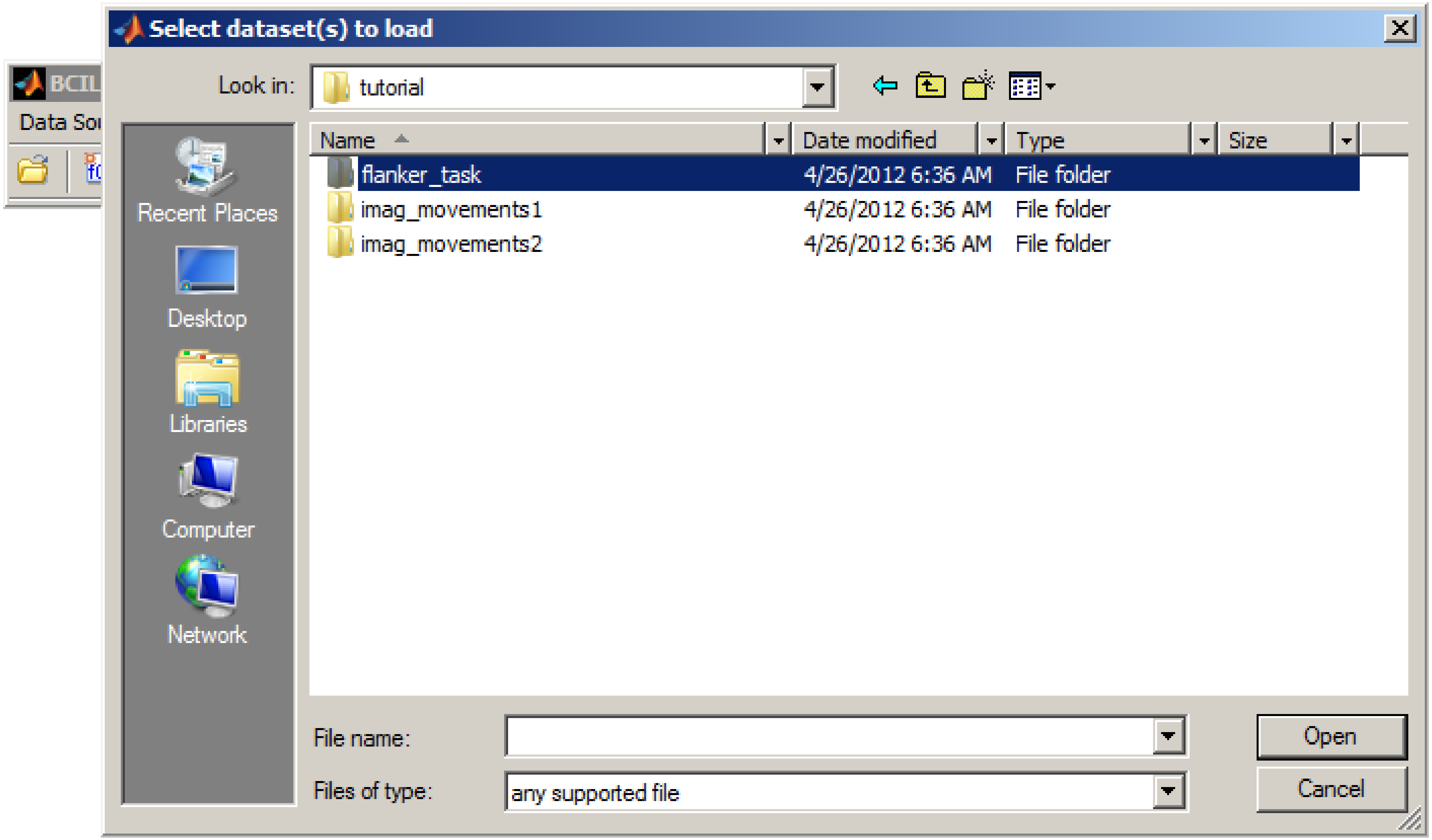
Loading the Data



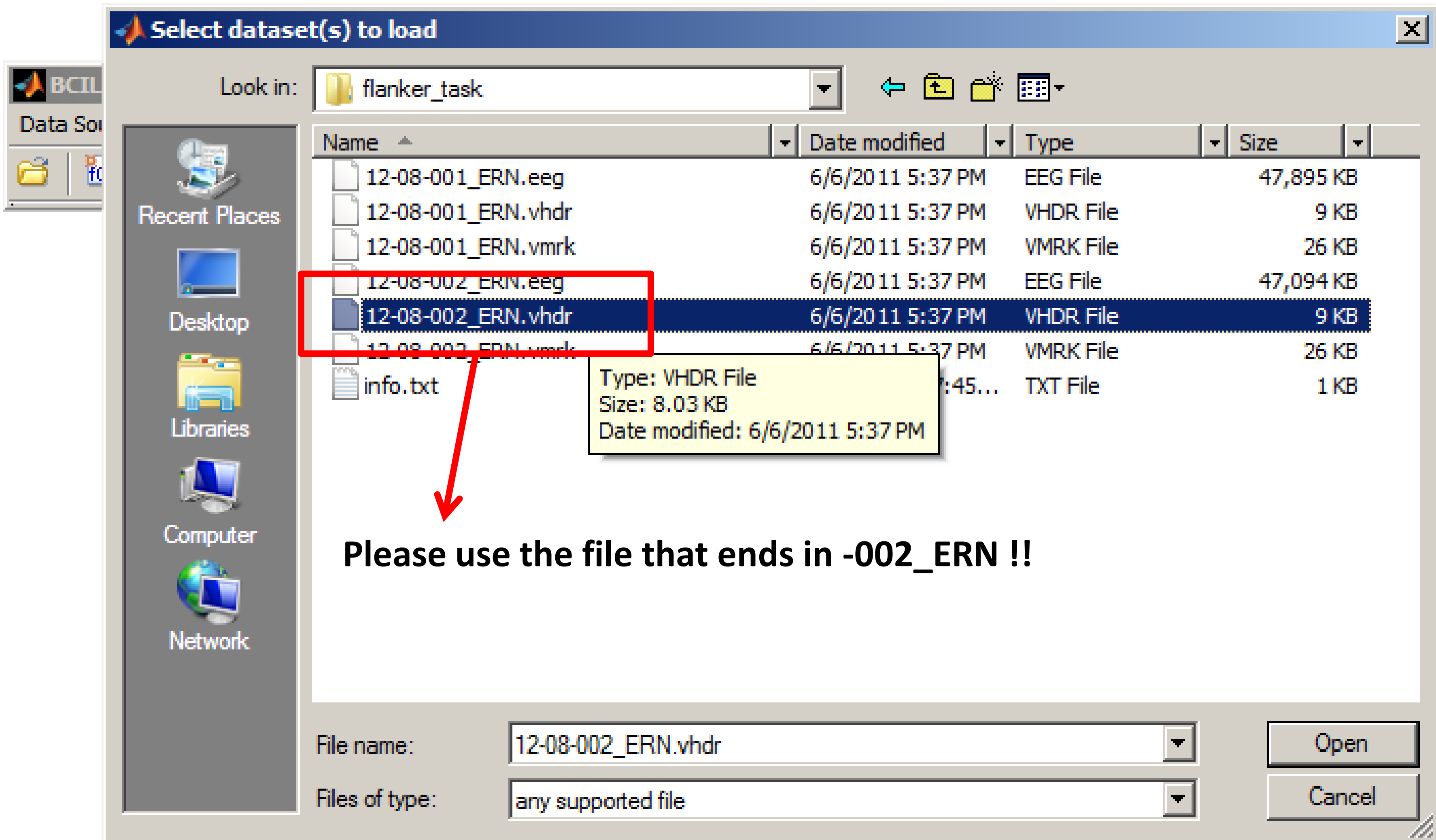
Loading the Data



Loading the Data

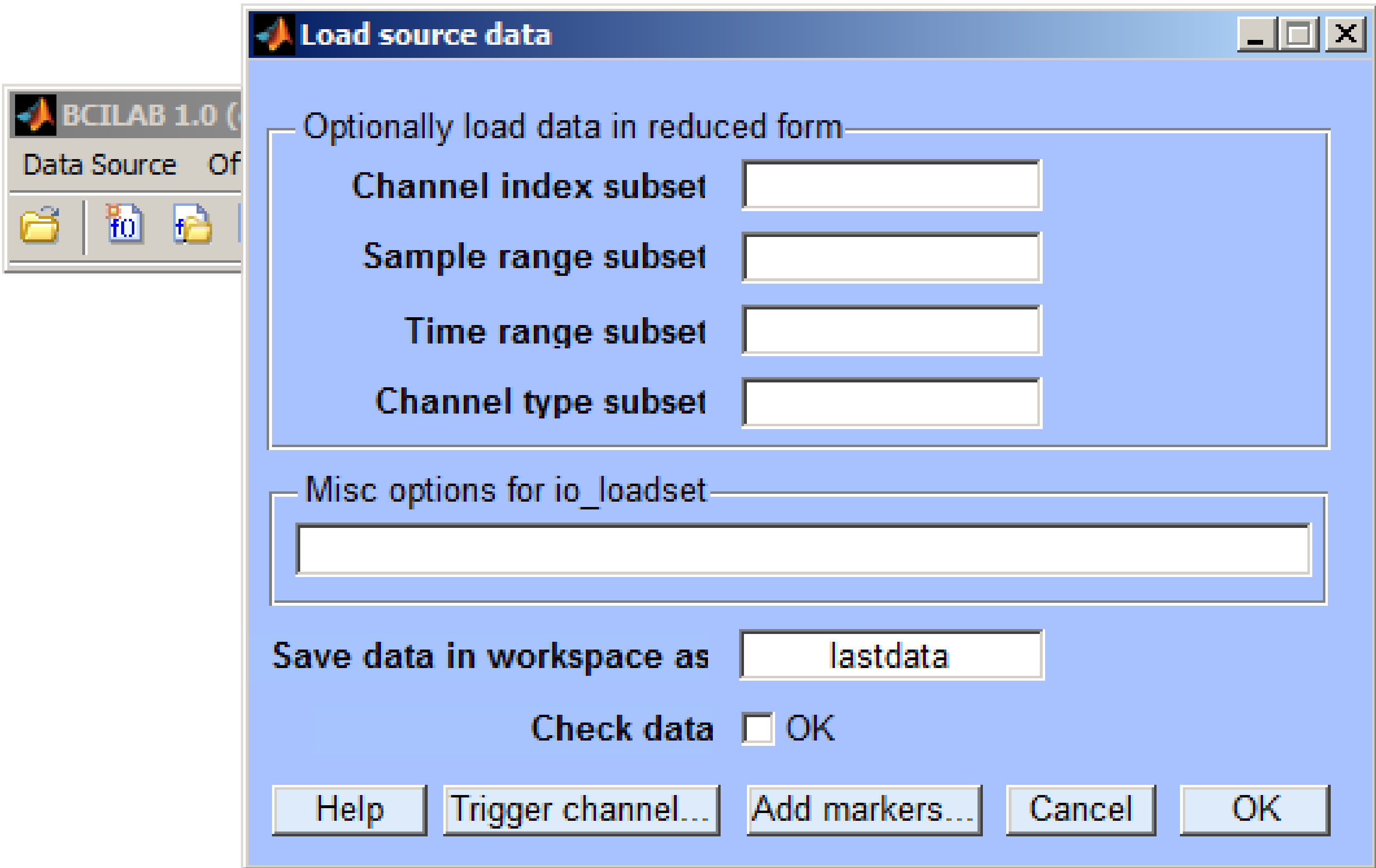


Loading the Data

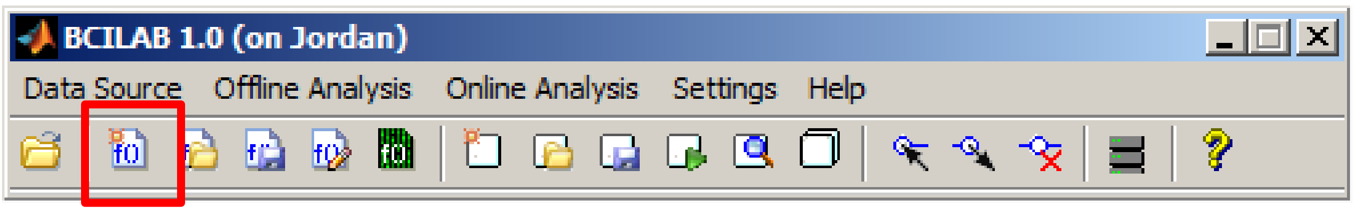


Please use the file that ends in -002_ERN !!

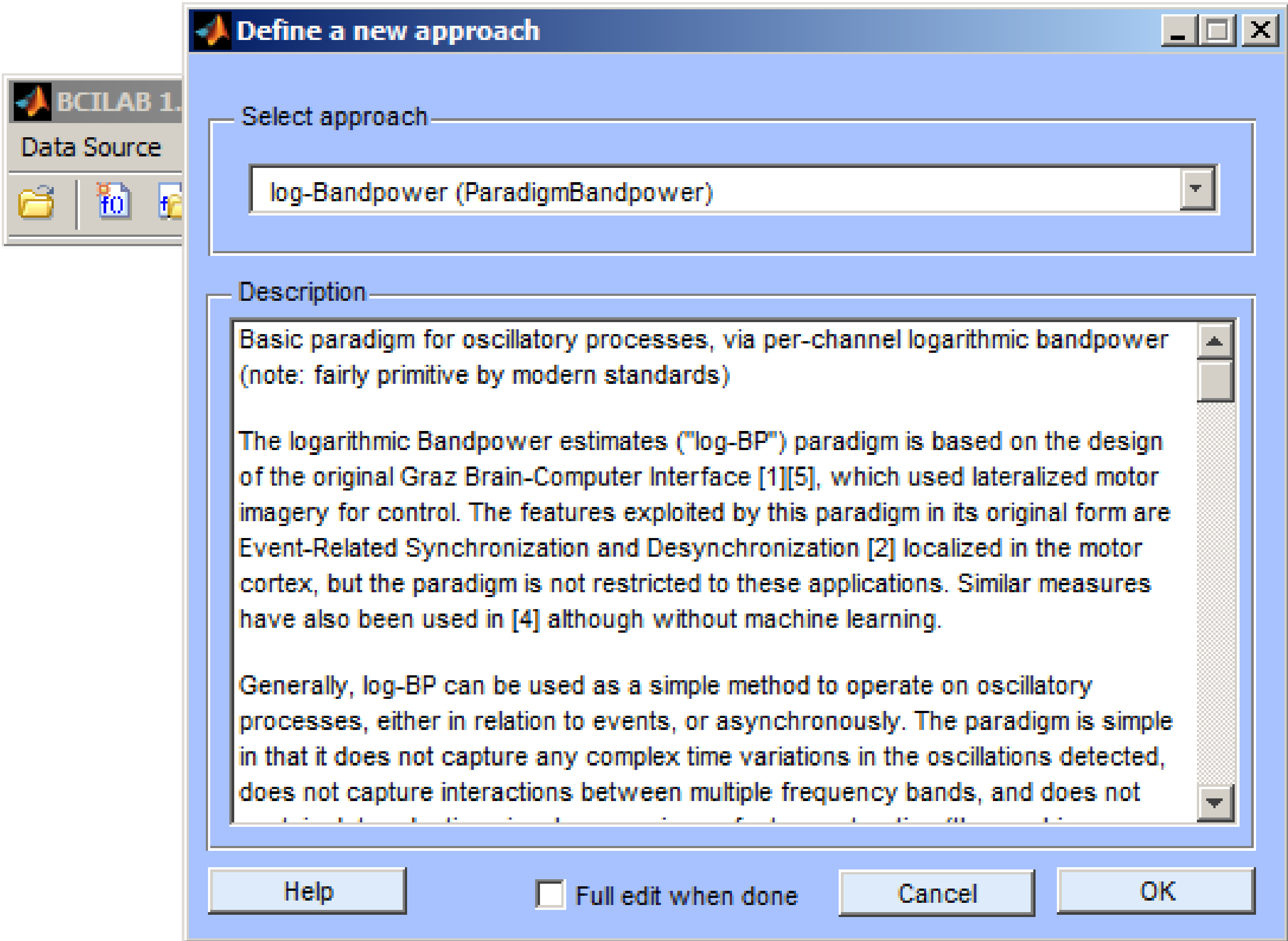
Confirming Import Options



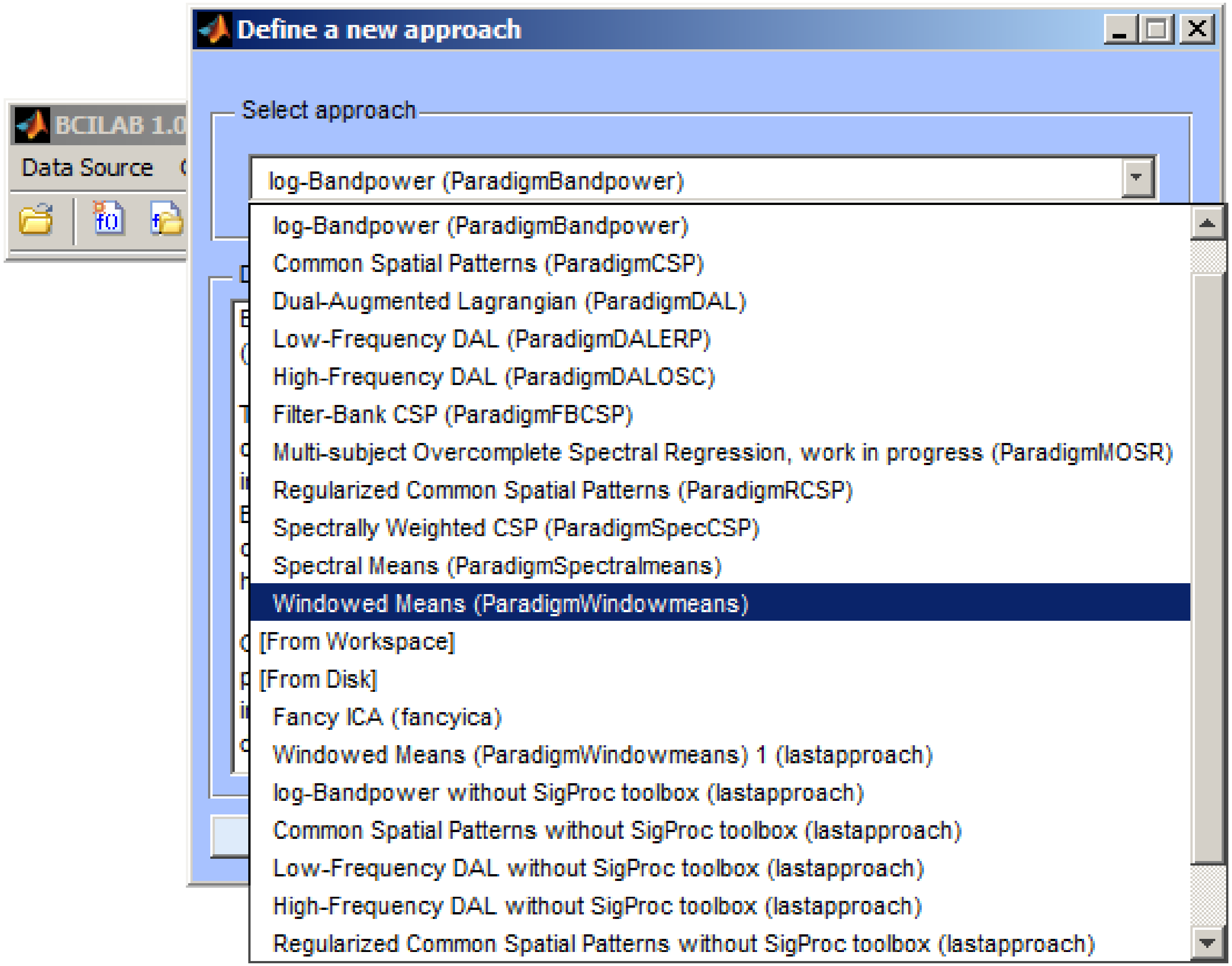
Creating a New Approach



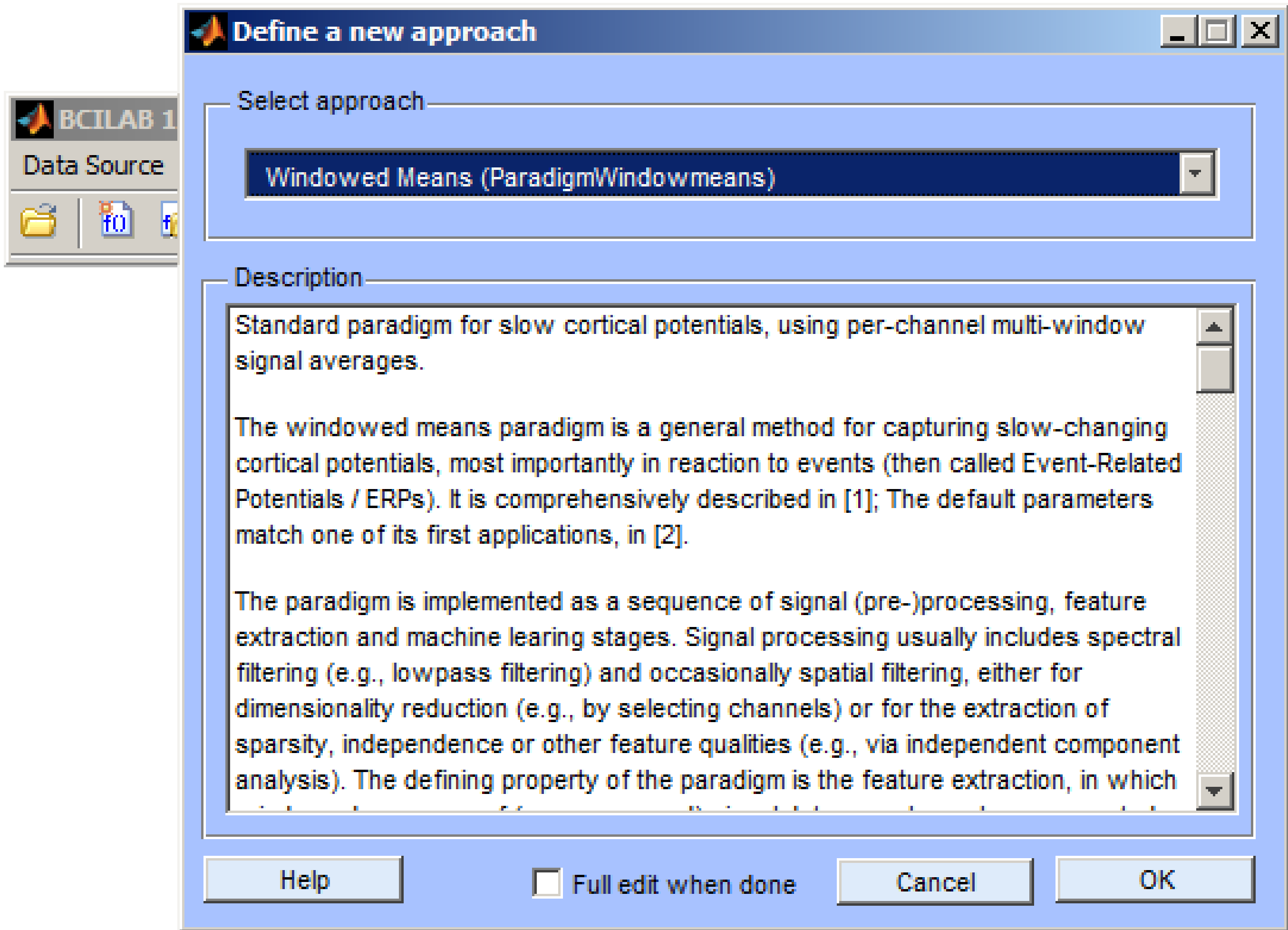
Creating a New Approach



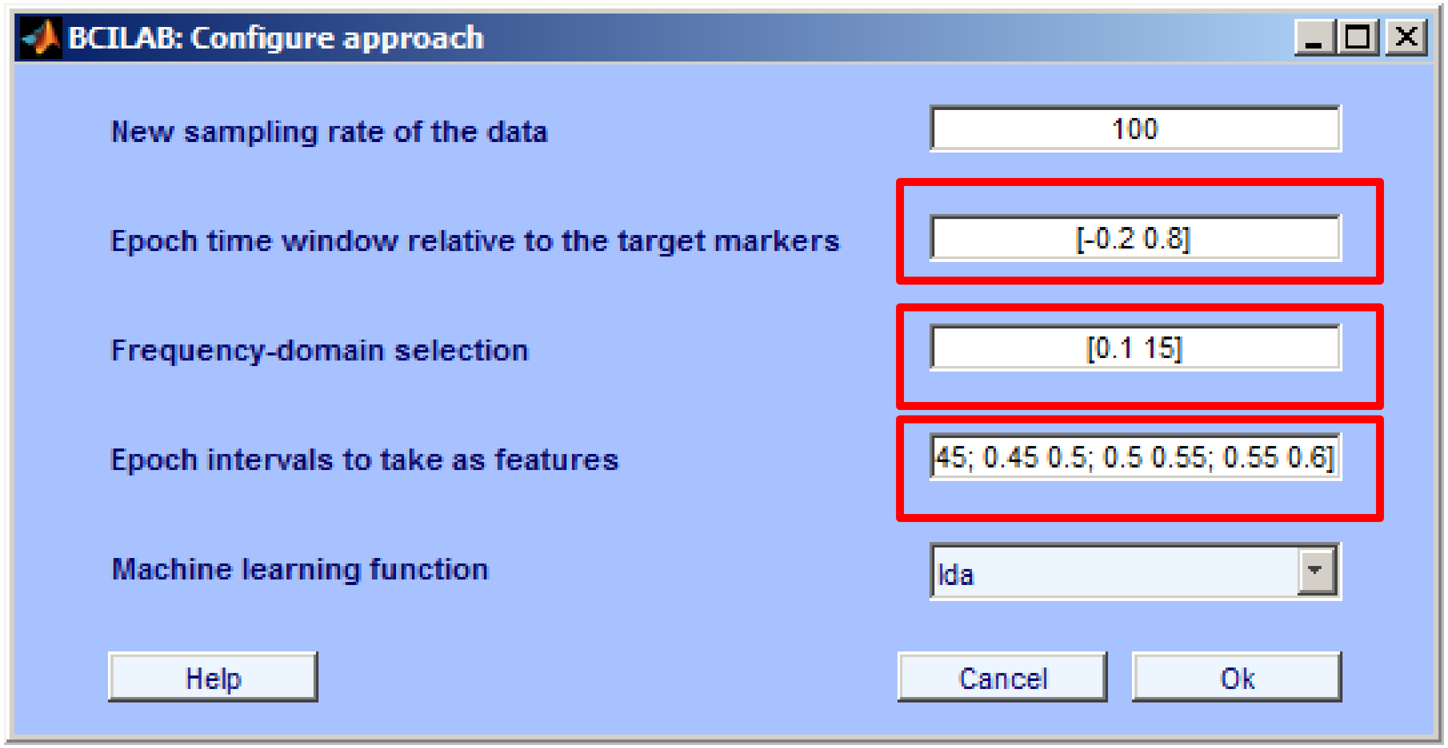
Select an ERP Paradigm



Select an ERP Paradigm

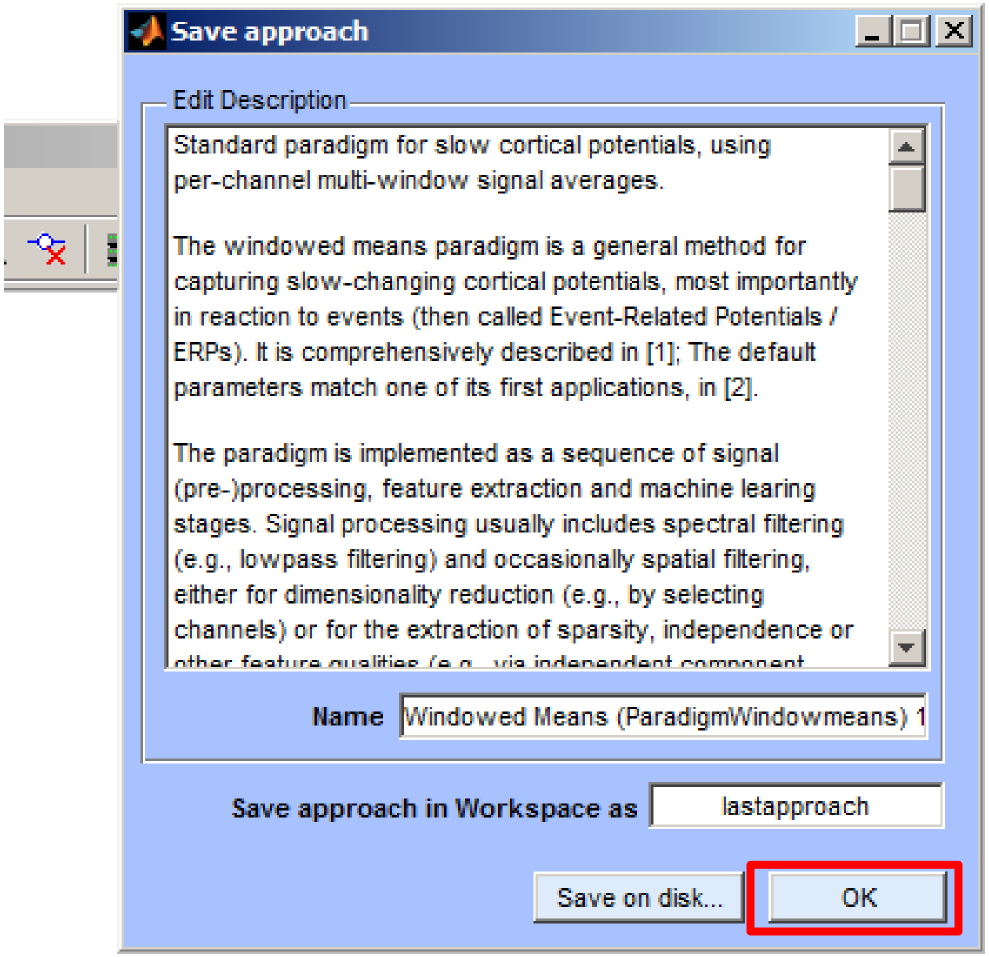


Configuring the Approach

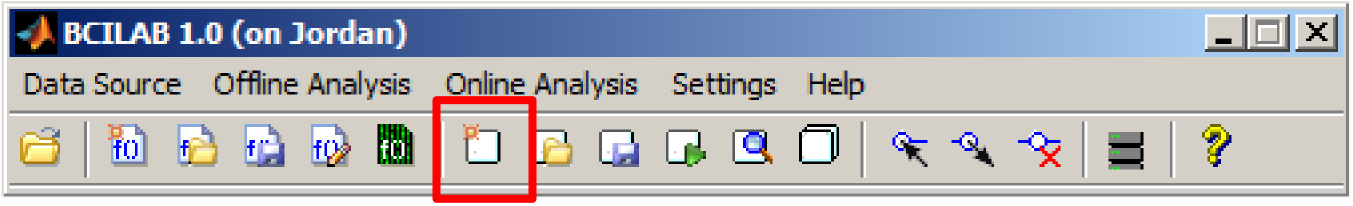


Type into the lowest of the 3 highlighted fields:
[0.25 0.3; 0.3 0.35; 0.35 0.4; 0.4 0.45; 0.45 0.5; 0.5 0.55; 0.55 0.6]

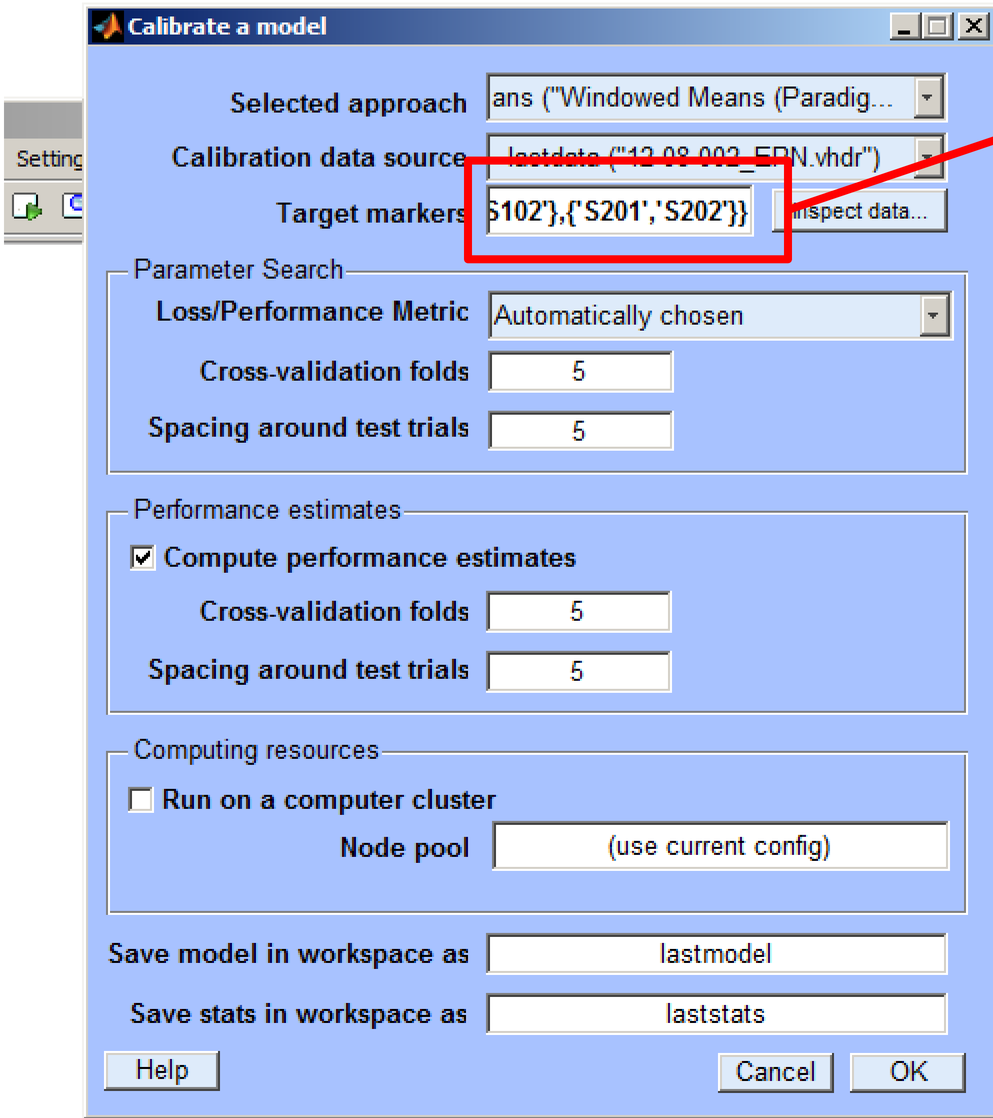
Saving to the Workspace



Calibrating a New Model



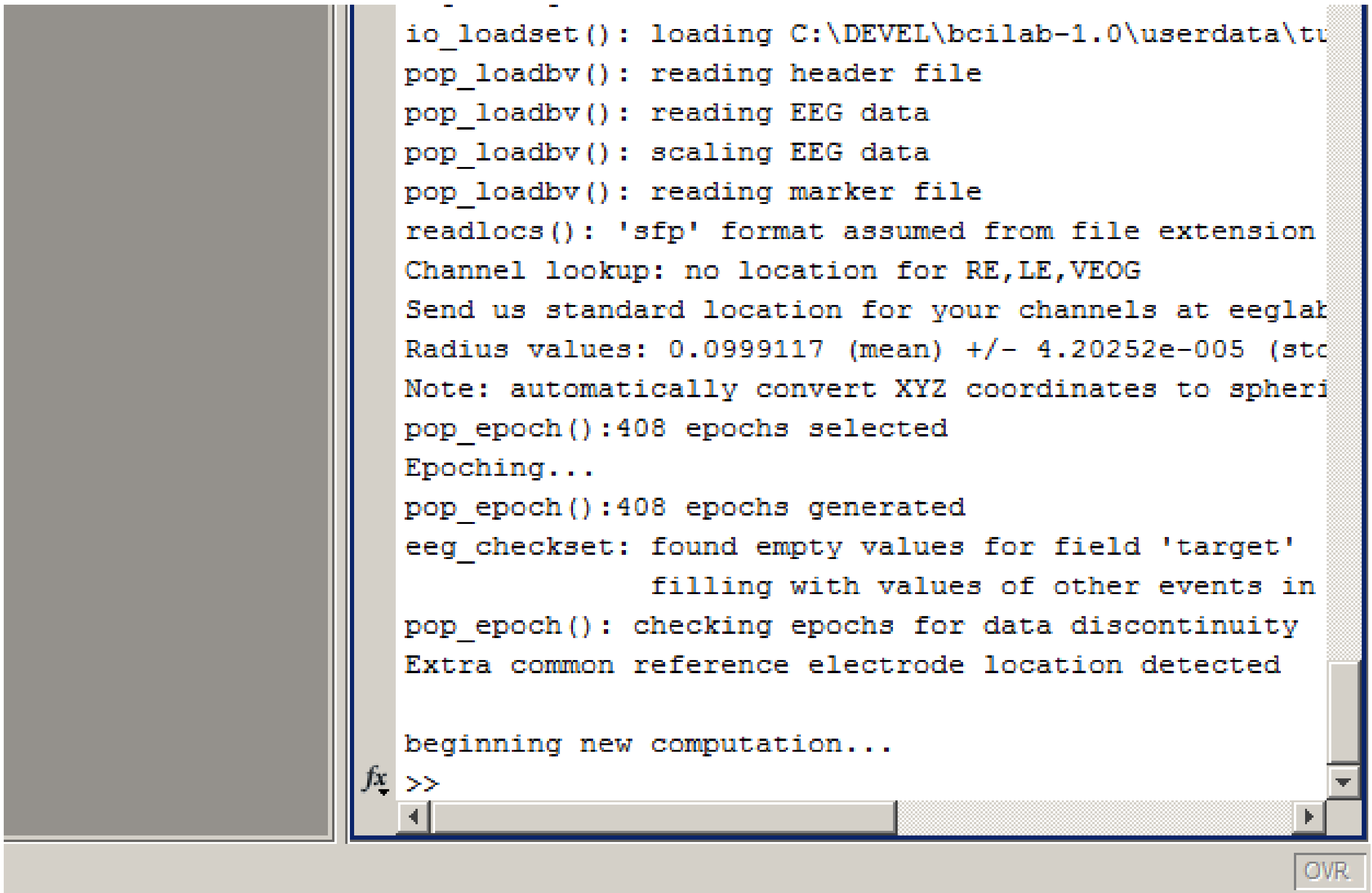
Calibrating a New Model



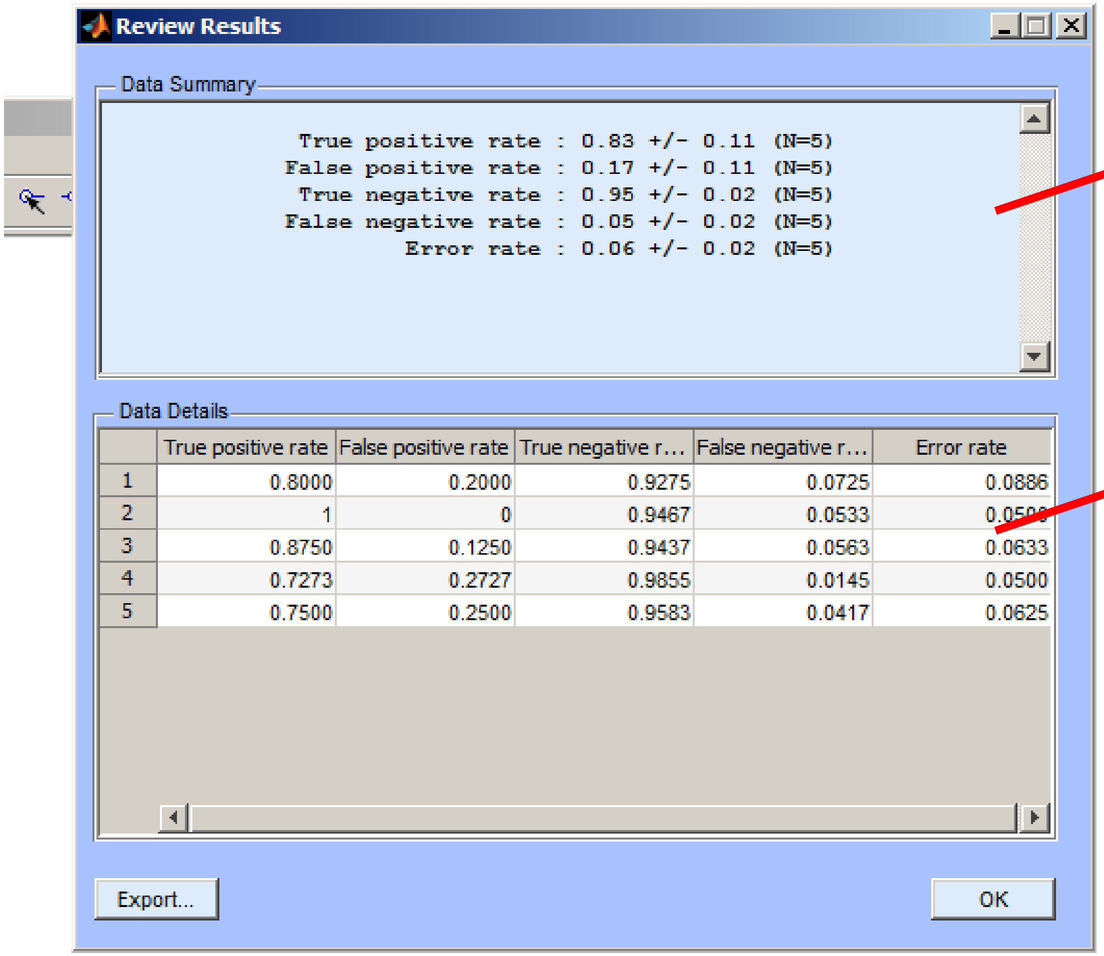
This is the set of marker labels that determine our two possible error conditions. For each of the two conditions, there is a group of multiple markers (different types of errors and non-errors).

Type the following here:
{{'S101','S102'},{'S201','S202'}}

Watching the Computation…



Reviewing Results

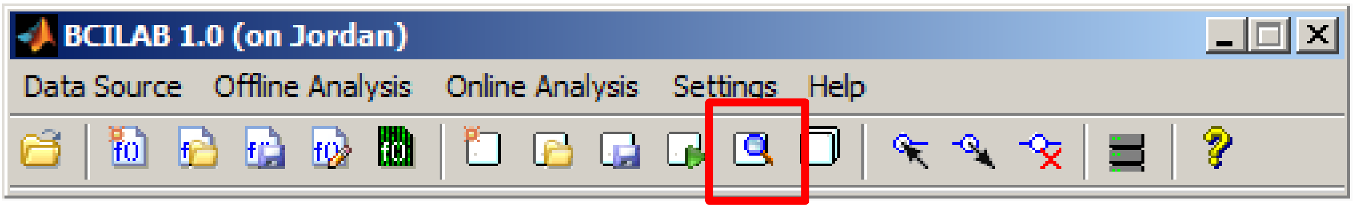


	True positive rate	False positive rate	True negative r...	False negative r...	Error rate
1	0.8000	0.2000	0.9275	0.0725	0.0886
2	1	0	0.9467	0.0533	0.0500
3	0.8750	0.1250	0.9437	0.0563	0.0633
4	0.7273	0.2727	0.9855	0.0145	0.0500
5	0.7500	0.2500	0.9583	0.0417	0.0625

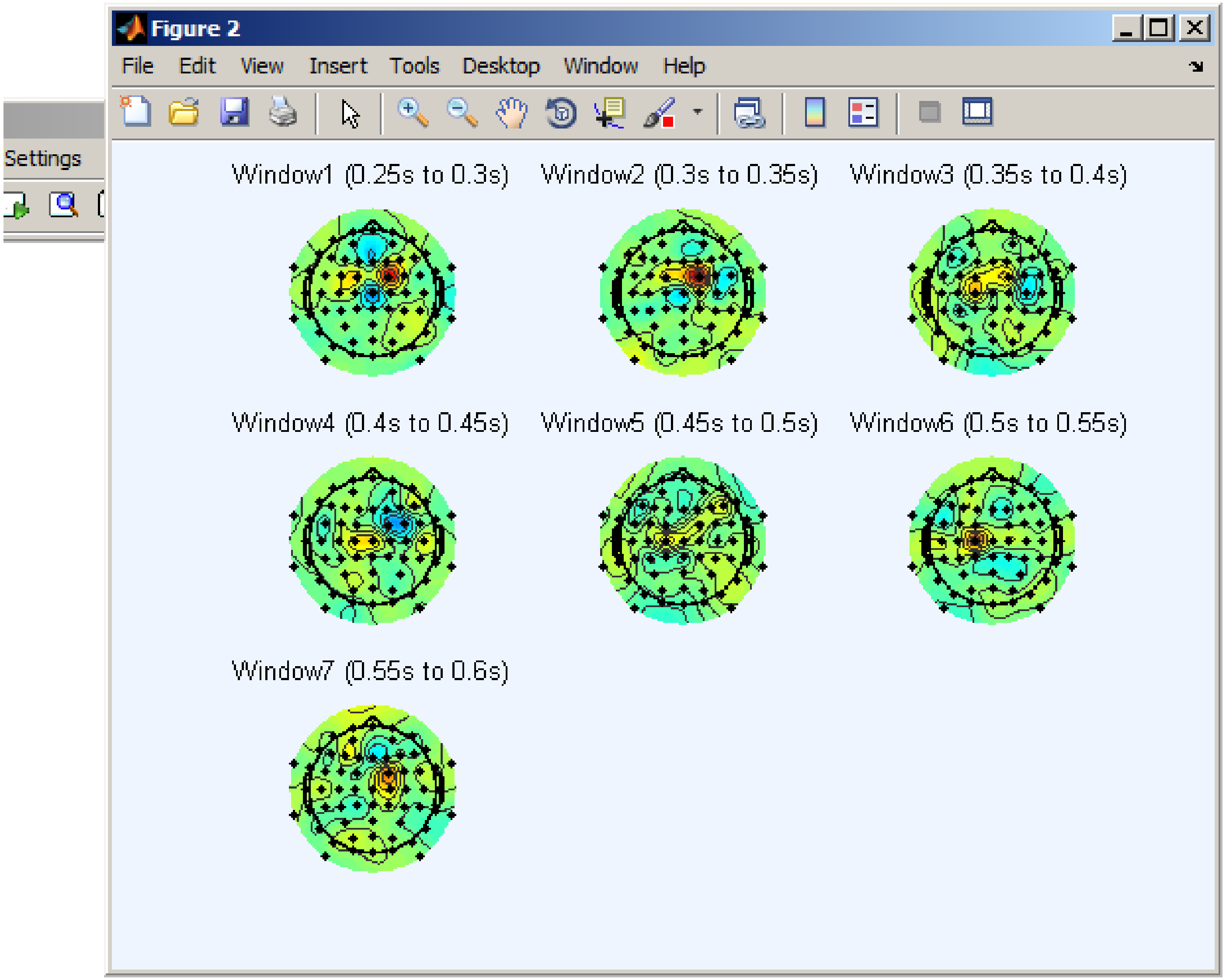
Summary Statistics

Statistics for each fold of the Cross-Validation (here 5x).

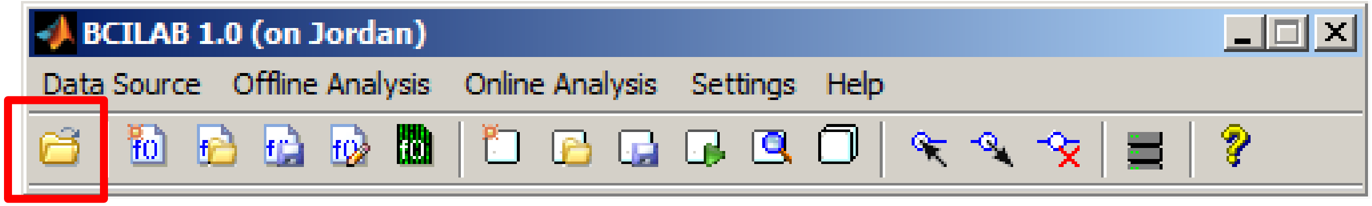
Visualizing the Model



Visualizing the Model

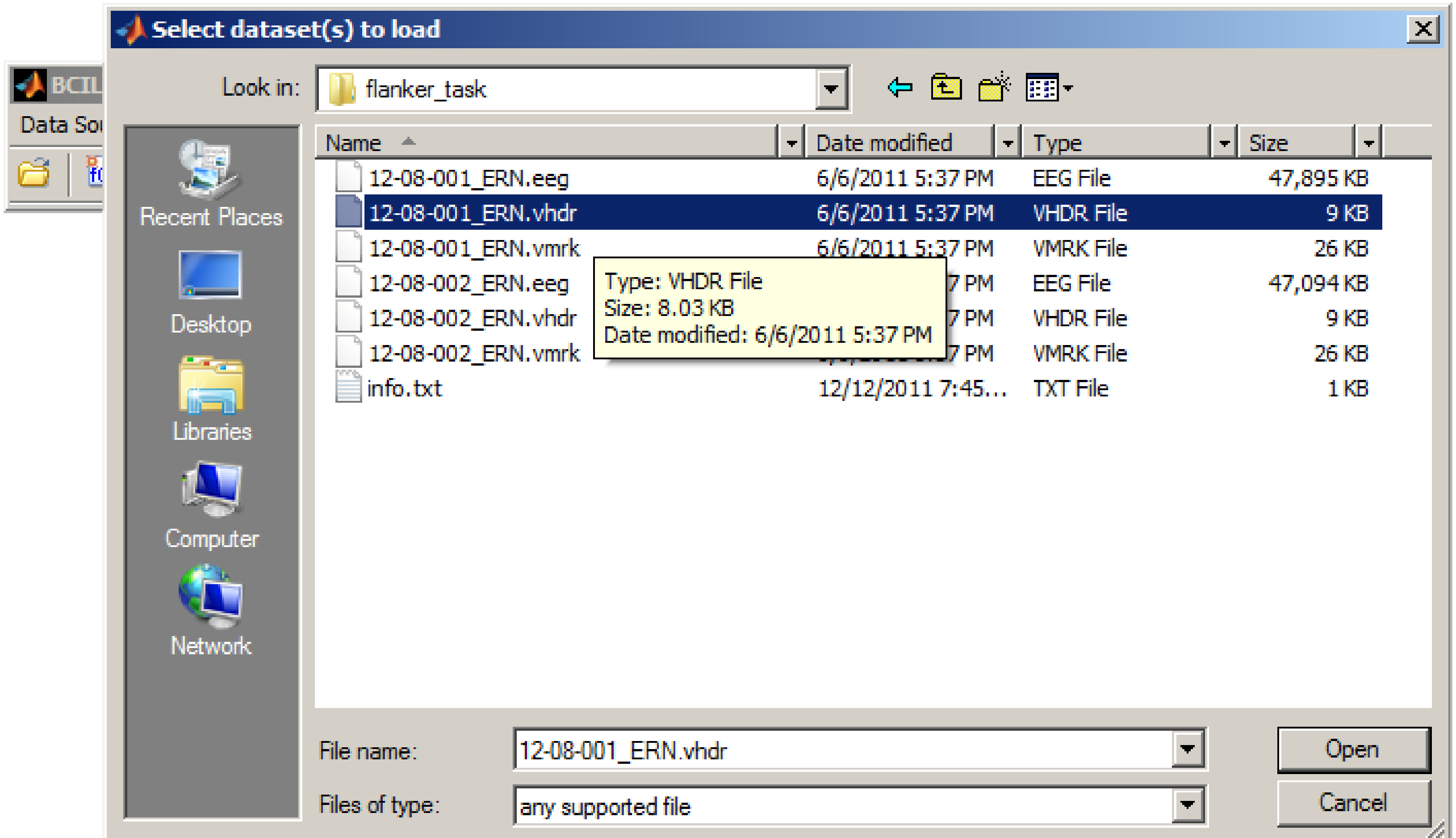


Loading a Separate Test Set

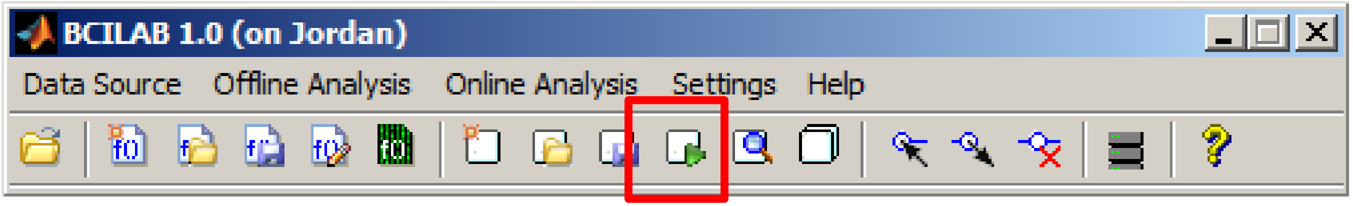


Loading a Separate Test Set

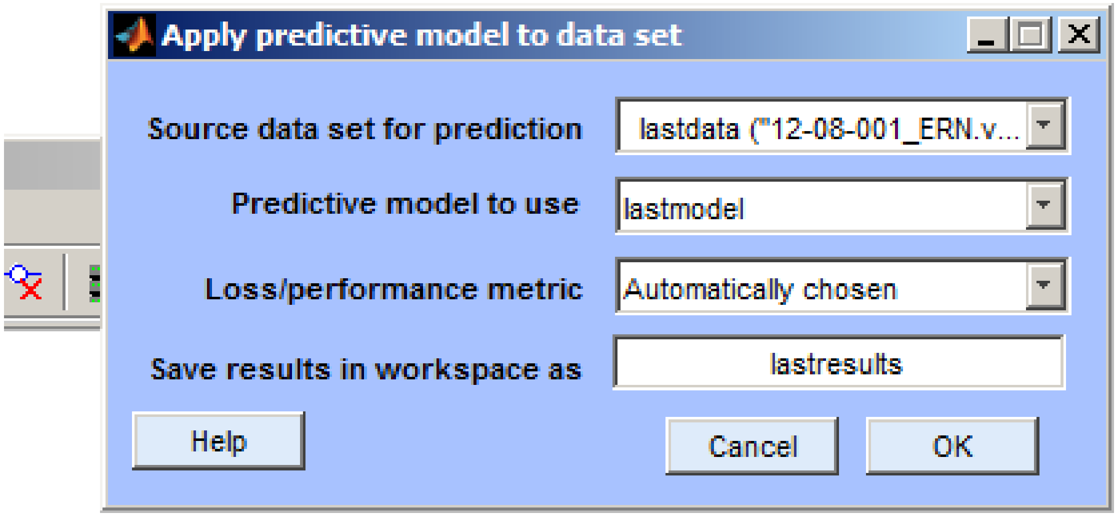
Note: This data set is from an identical twin doing the same task.



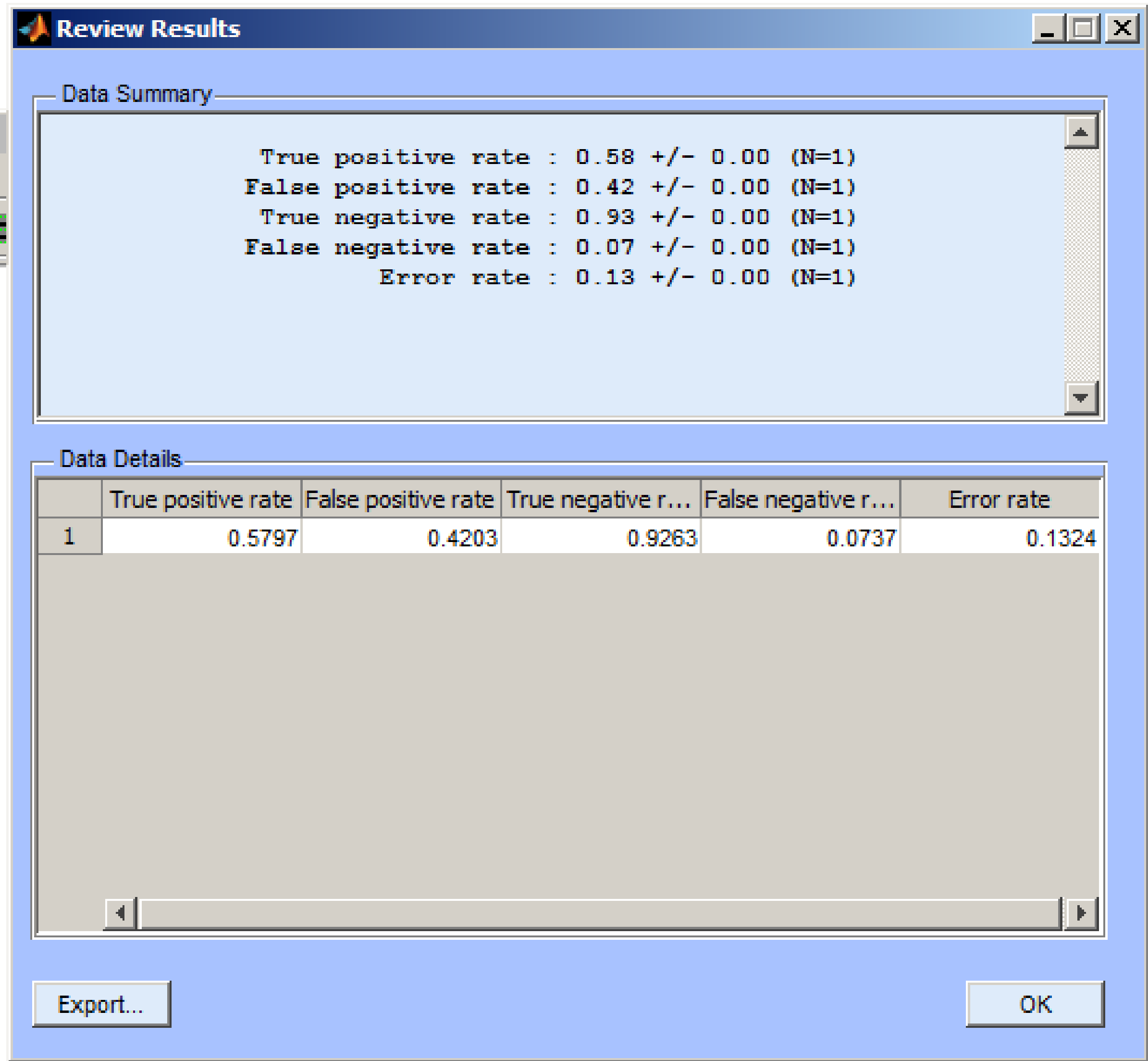
Applying the Model to Test Data



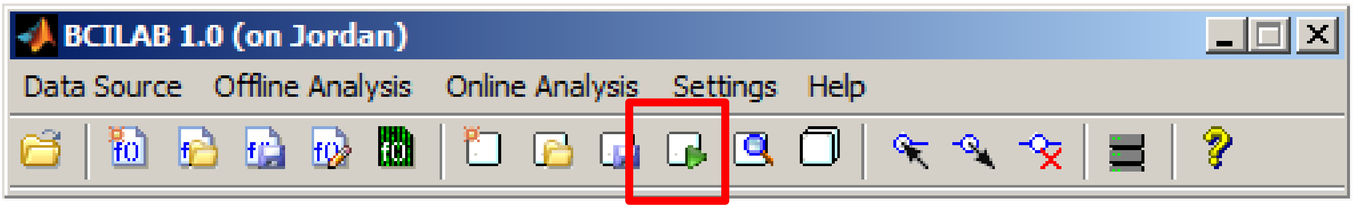
Applying the Model to Test Data



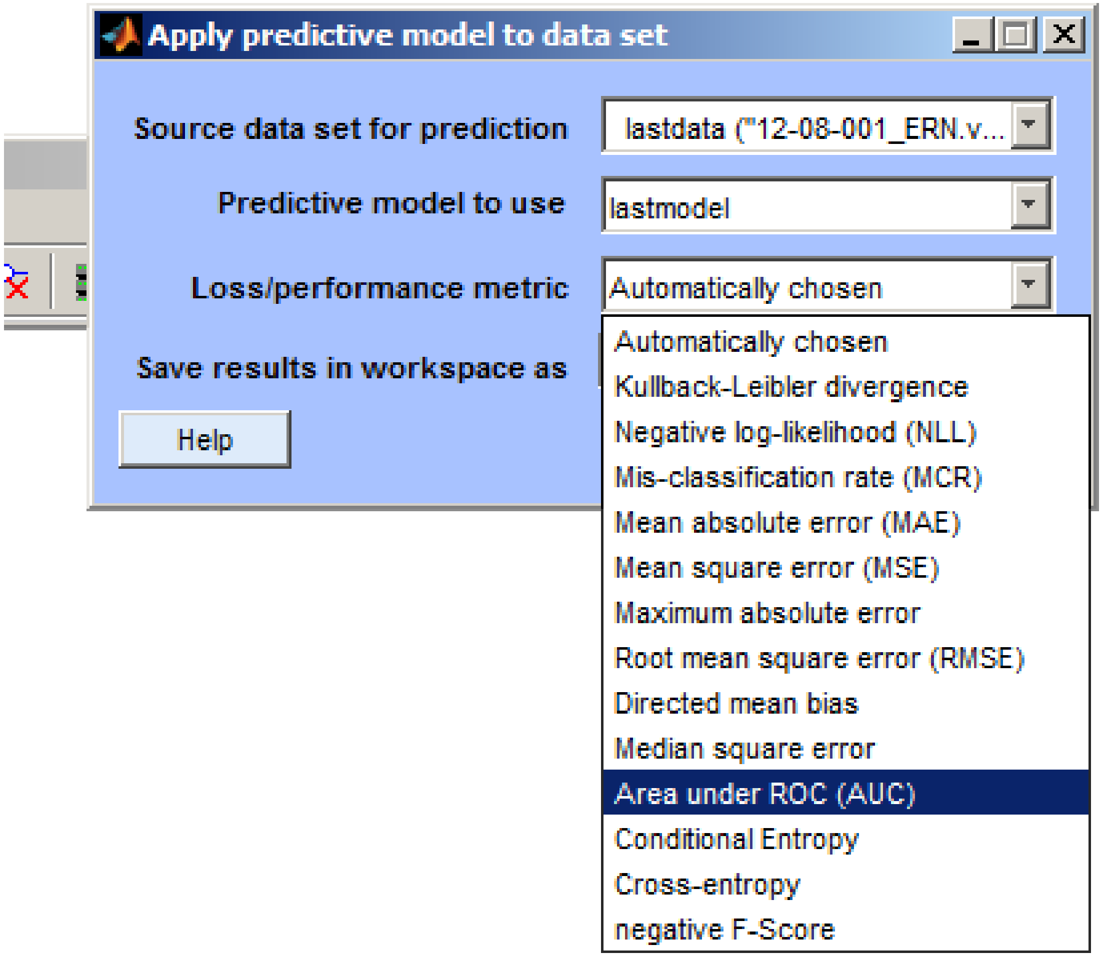
Reviewing Statistics



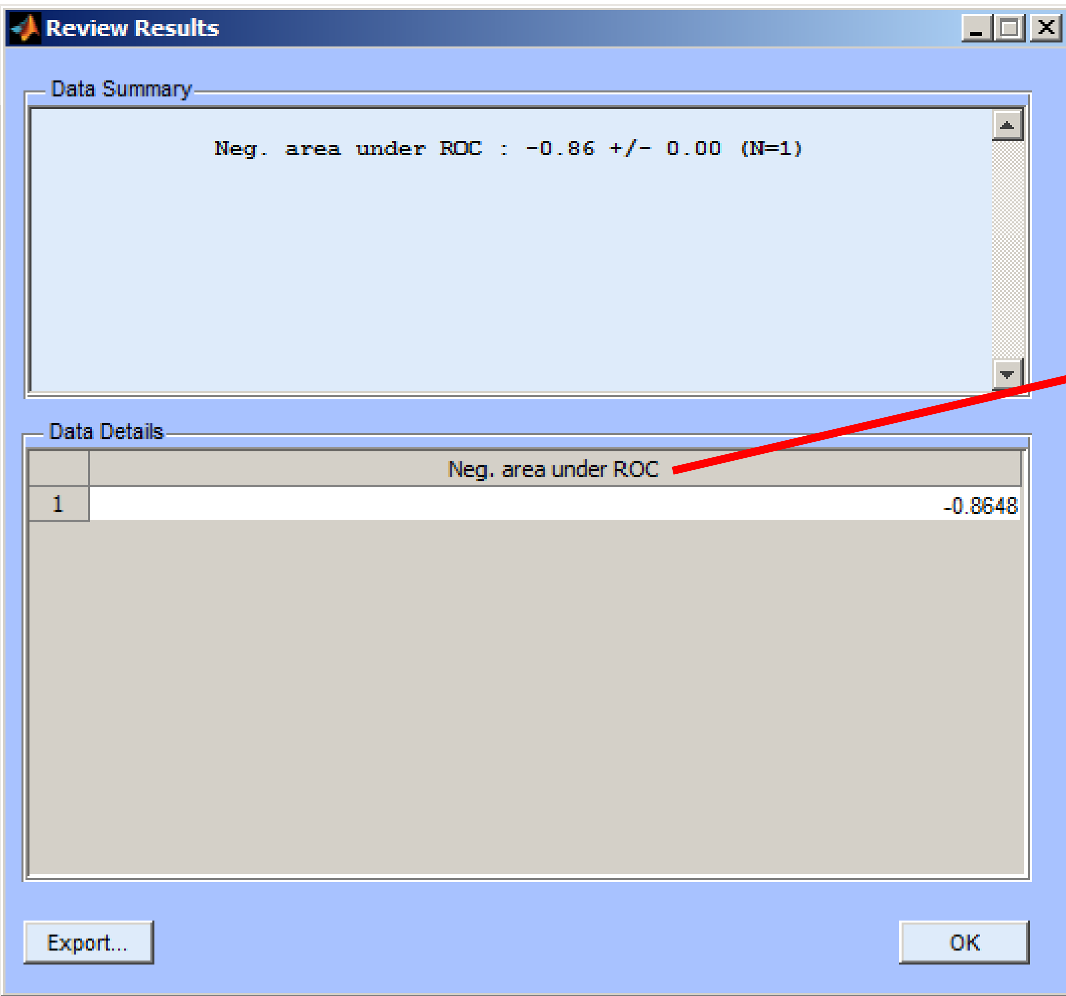
Using Another Loss Measure

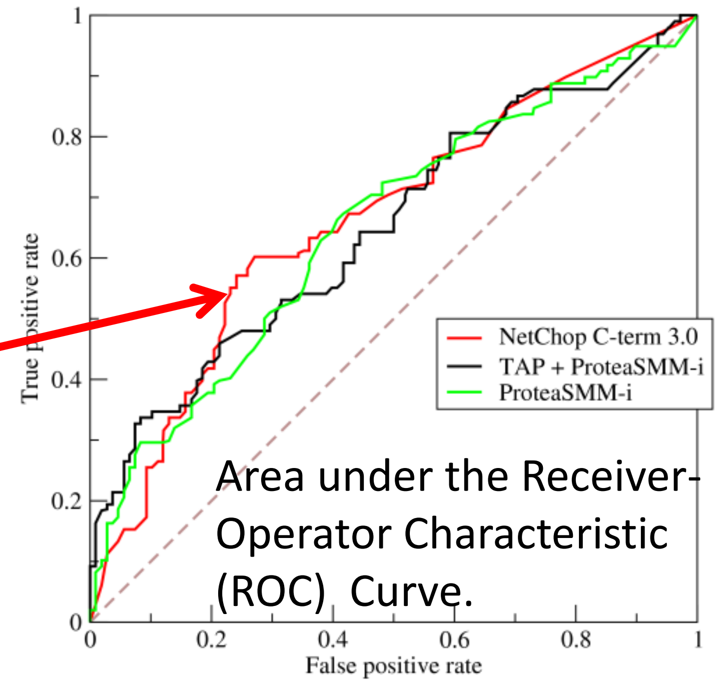


Using Another Loss Measure



Using Another Loss Measure

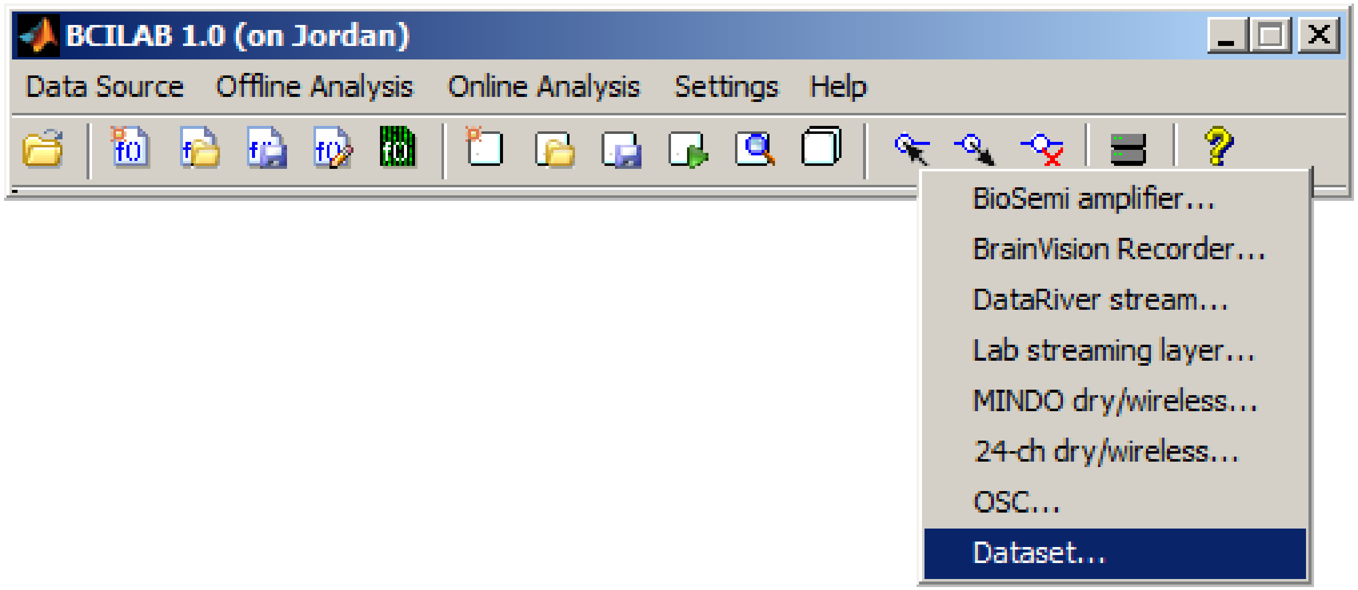




Area under the Receiver-Operator Characteristic (ROC) Curve.

Online Analysis

Starting an Online Data Stream



The selected stream will be played back in the background.
In this course we'll be playing back the test data set in real time (instead of reading from an actual device).

Configuring the Online Stream

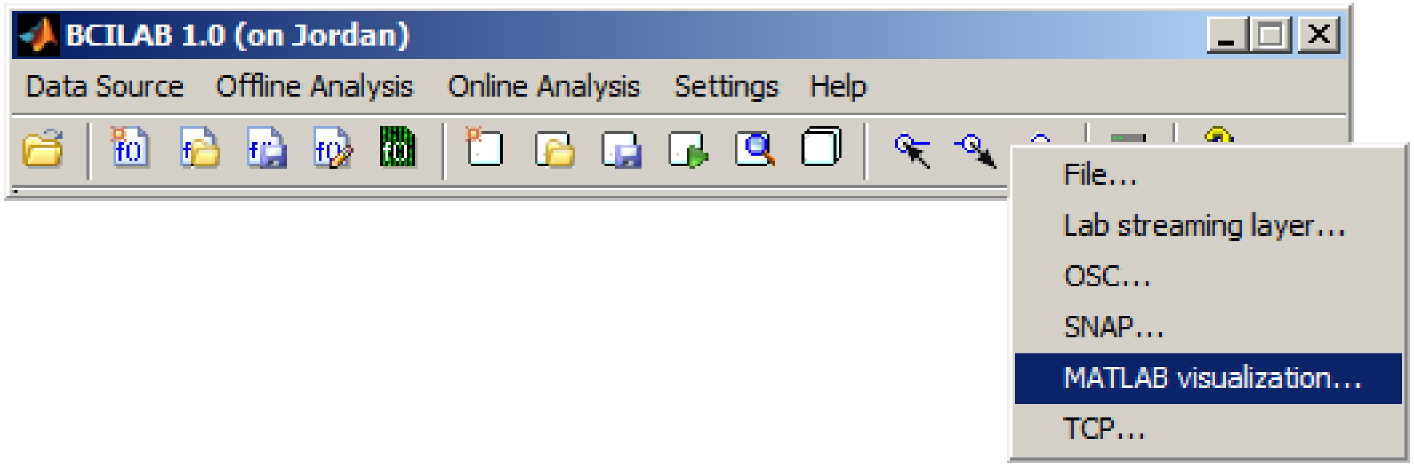
Meanwhile in the MATLAB Workspace…

If you type **whos** you could see the data structure (laststream) that is updated in the background.

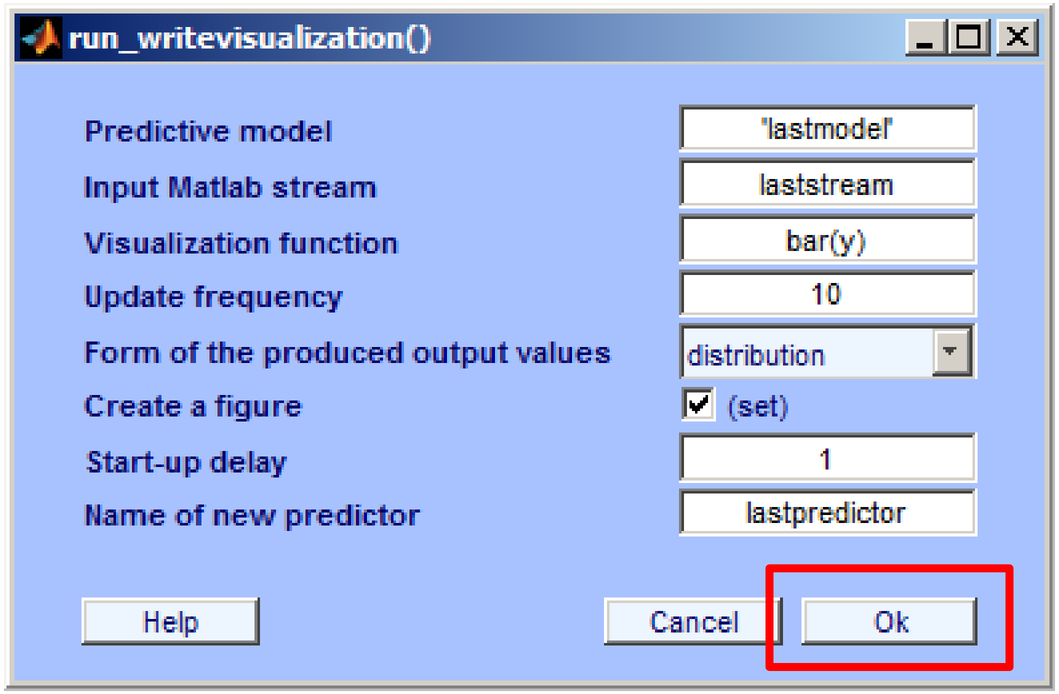
```
beginning new computation...
beginning evaluation...
Extra common reference electrode location detected
>> whos
  Name                 Size                 Bytes  Class

  ans                  1x1                  38164  struct
  f                    1x1                      8  double
  lastapproach         1x1                  38164  struct
  lastchunk           64x15                  3840  single
  lastdata             1x1                    874  struct
  lastmodel            1x1                 356657  struct
  lastresults          1x1                  12543  struct
  laststats            1x1                 418464  struct
  laststream           1x1                7804880  struct
  y                    1x2                     16  double

>> 
```

OVR

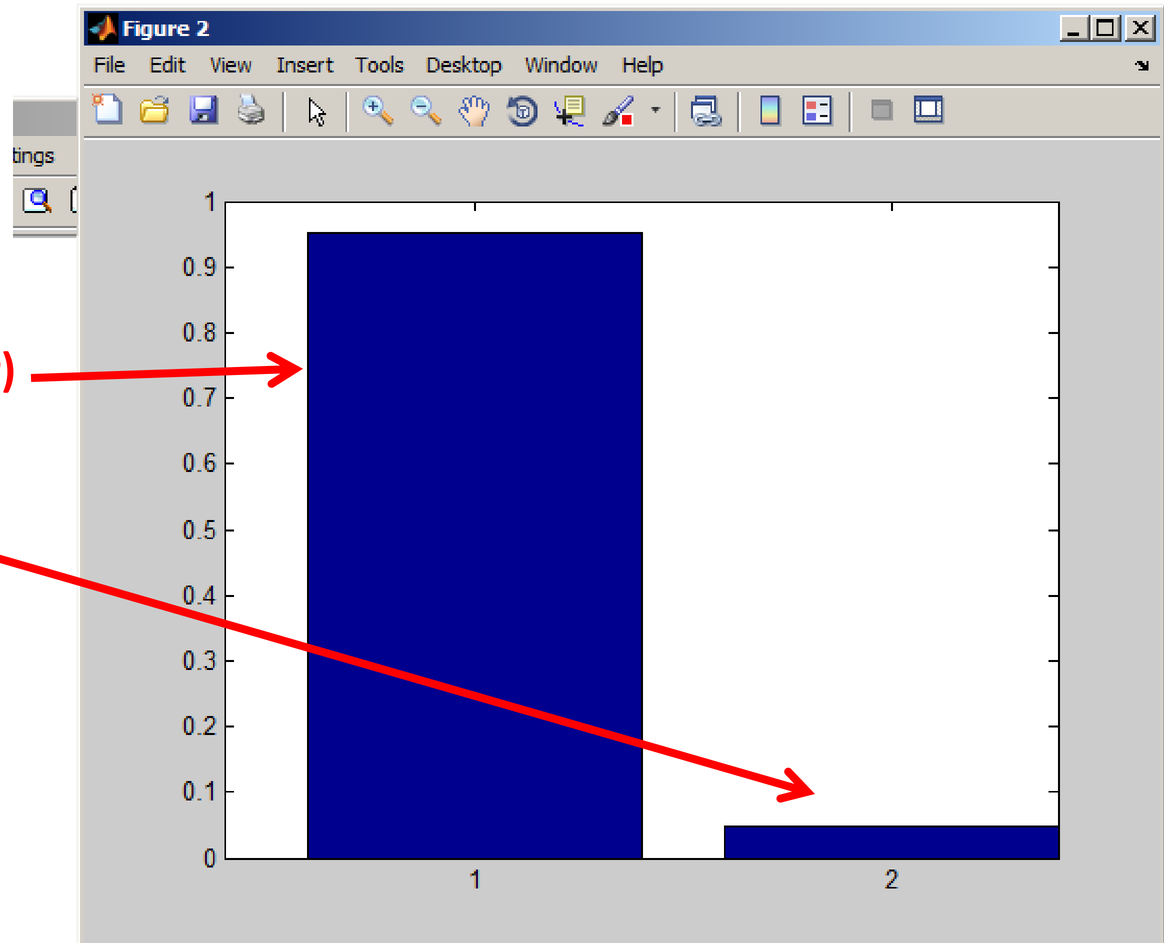
Selecting the Destination for BCI Outputs



Selecting the Destination for BCI Outputs



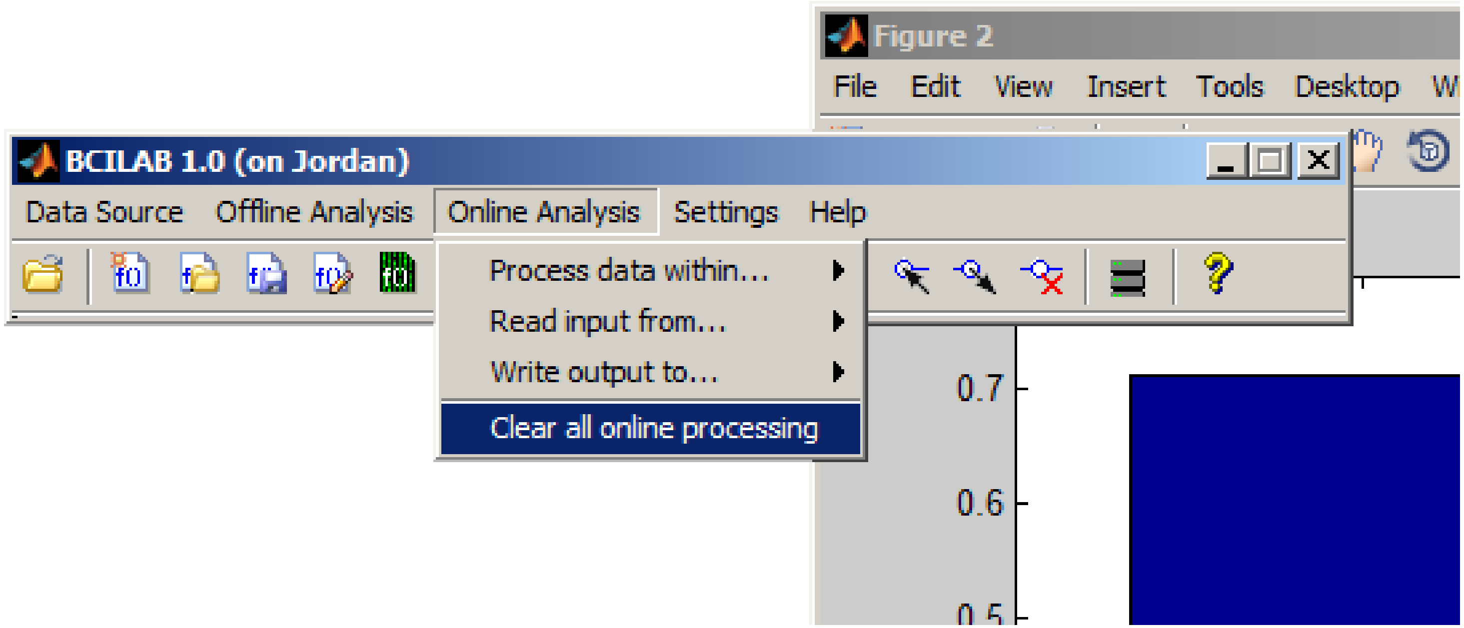
Visualized Real-Time Outputs



Probability of class 1 (no error)

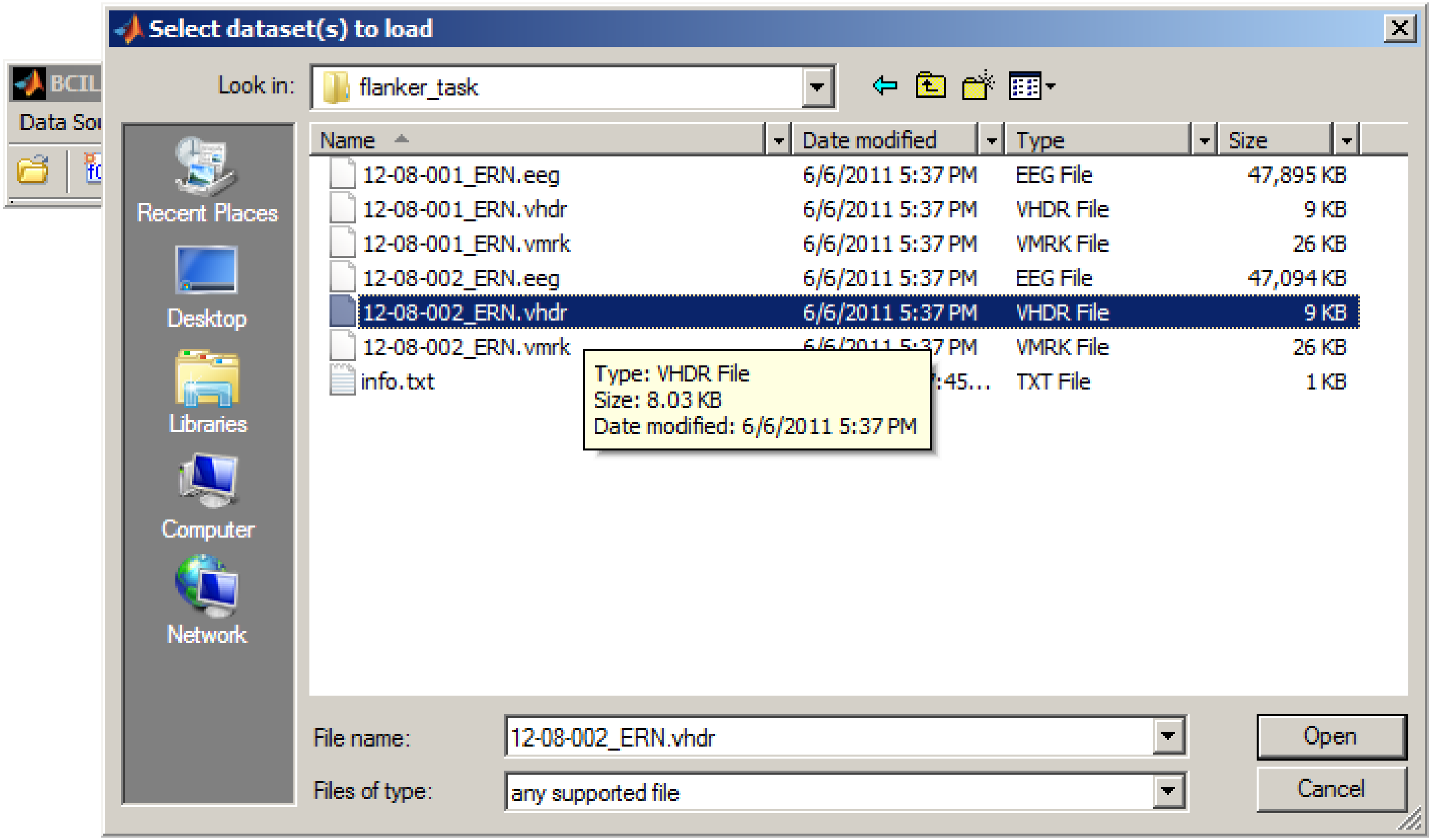
Probability of class 2 (error)

Stopping the Online Processing

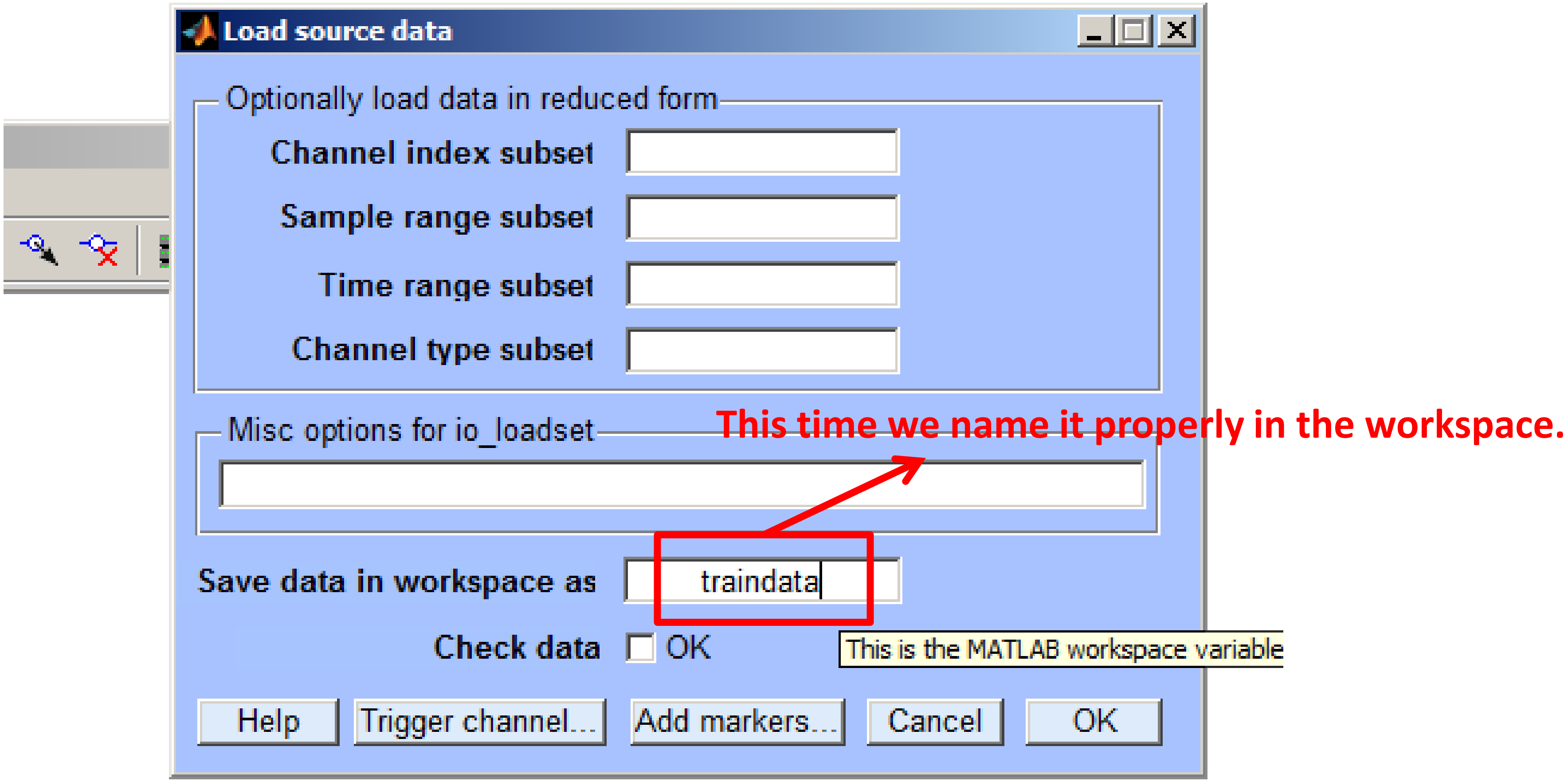


Customizing Approaches

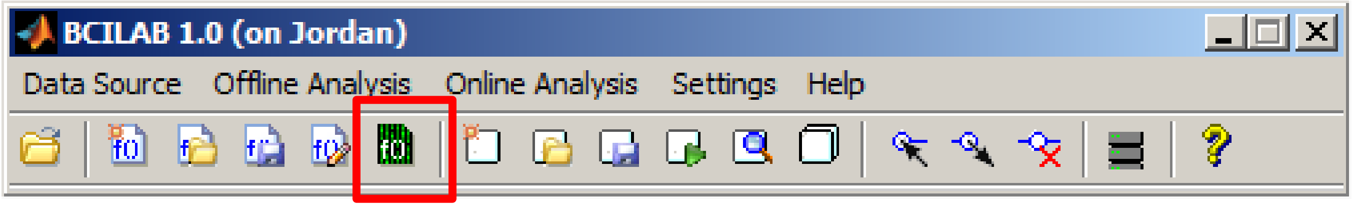
Loading the Training Data Again



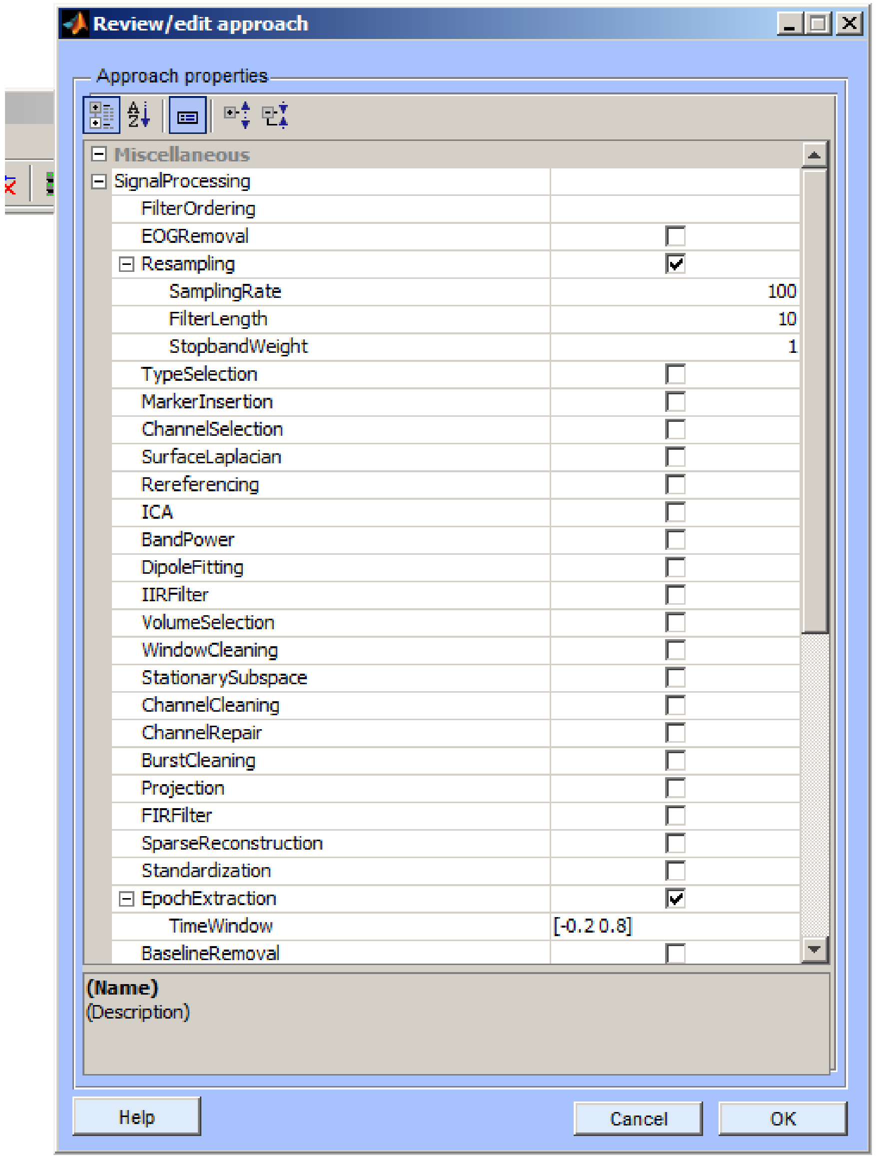
Loading the Training Data Again



Editing the Previous Approach In Detail

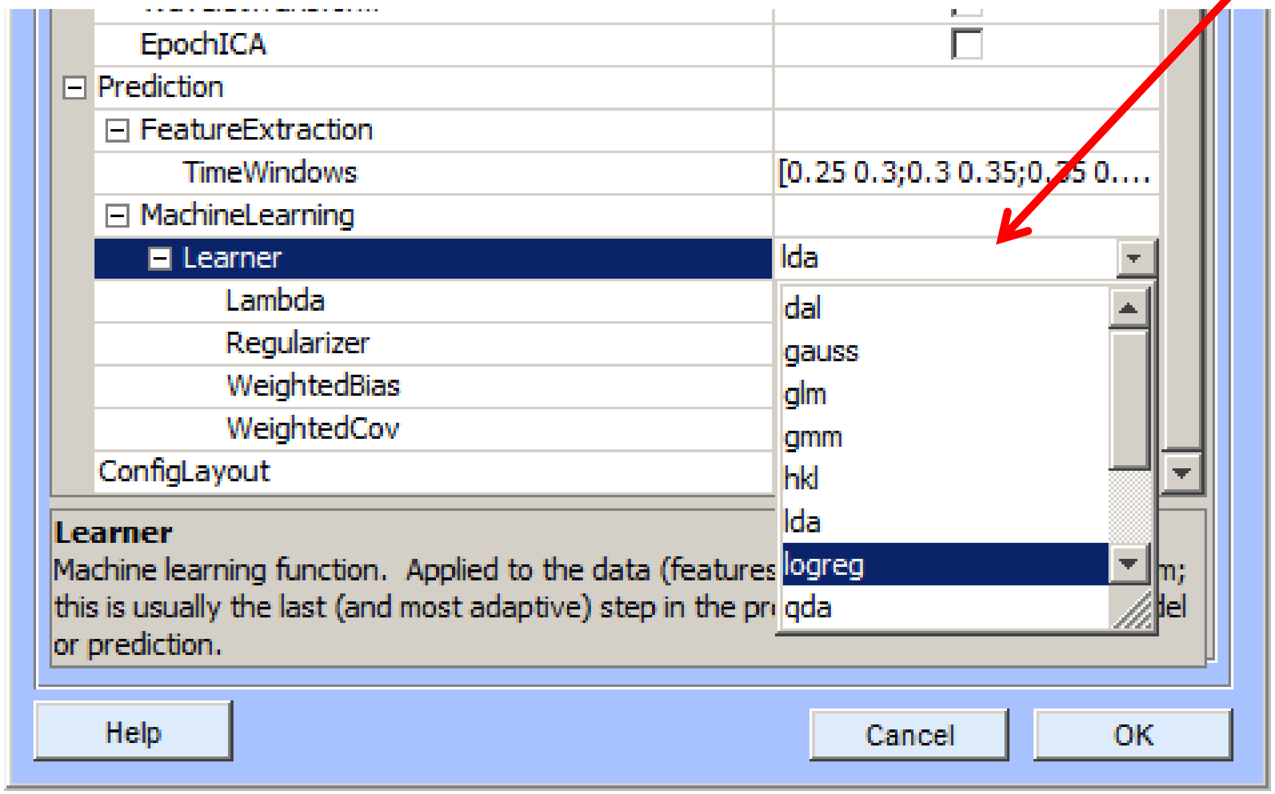


Editing the Previous Approach In Detail



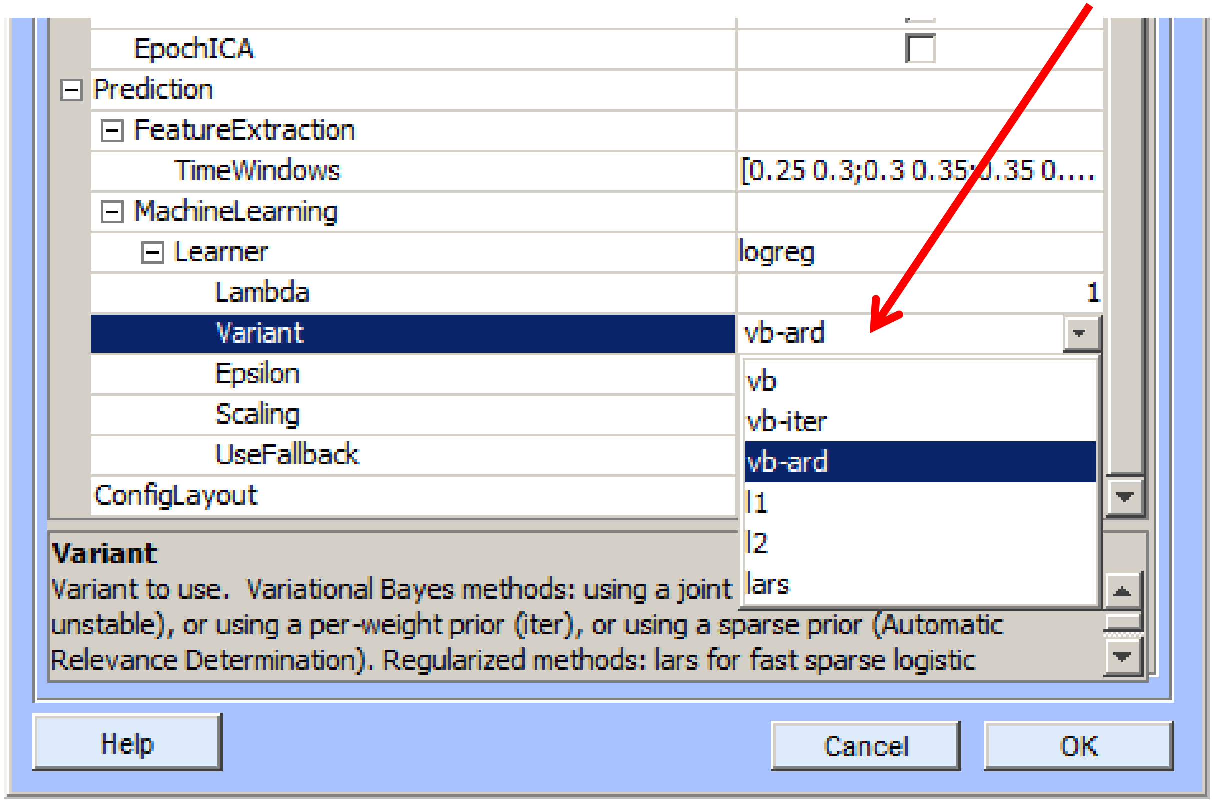
Changing the Classifier

Instead of LDA we choose logistic regression.

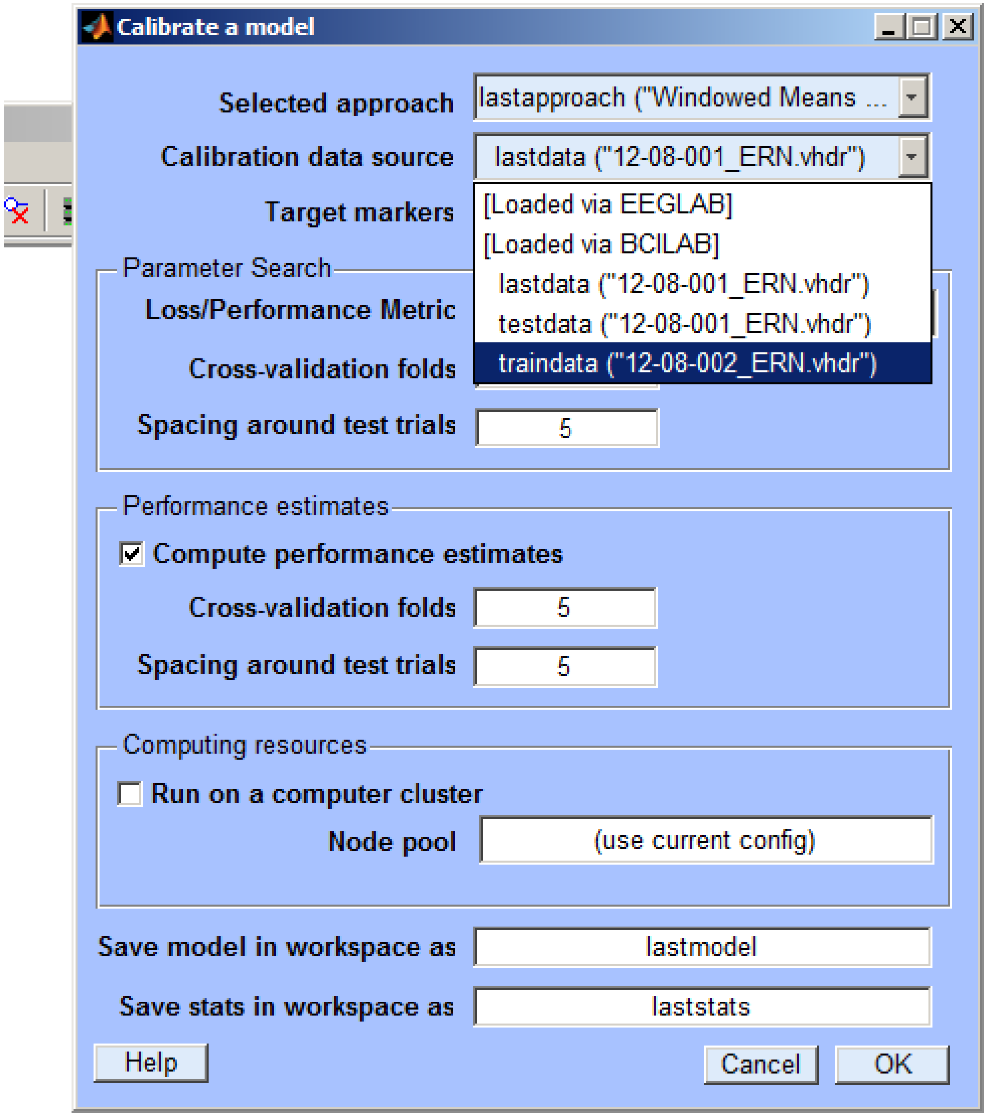


Changing the Classifier

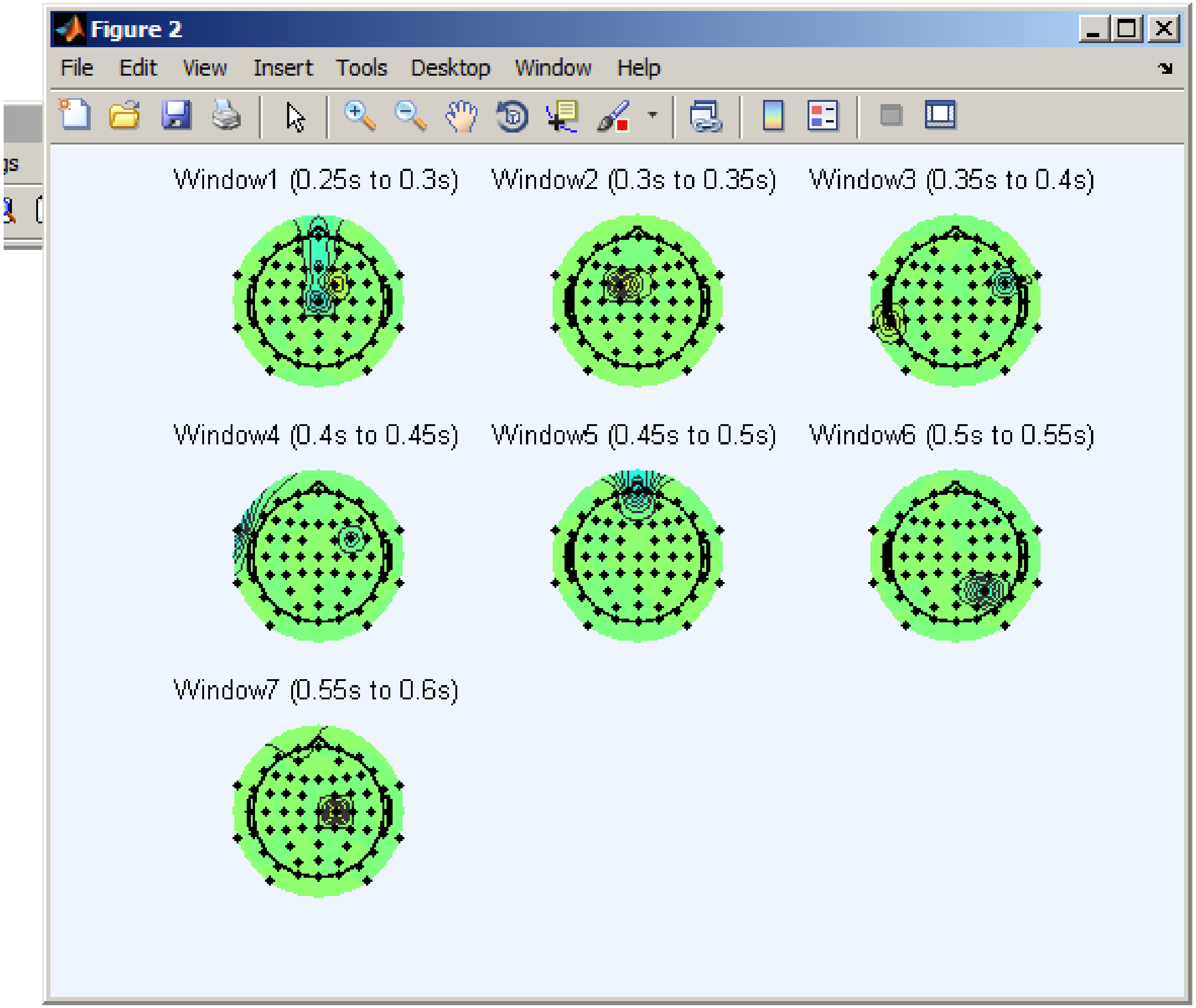
We don't use the VB (Variational Bayes) variant but the sparse version (with Automatic Relevance Determination).



Learning a New Model…

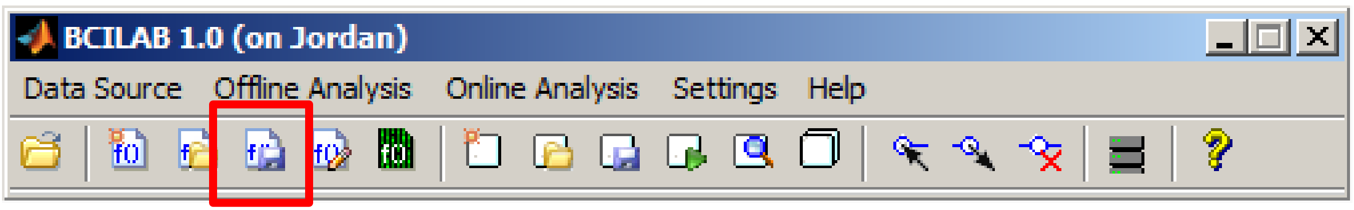


Visualizing The Model

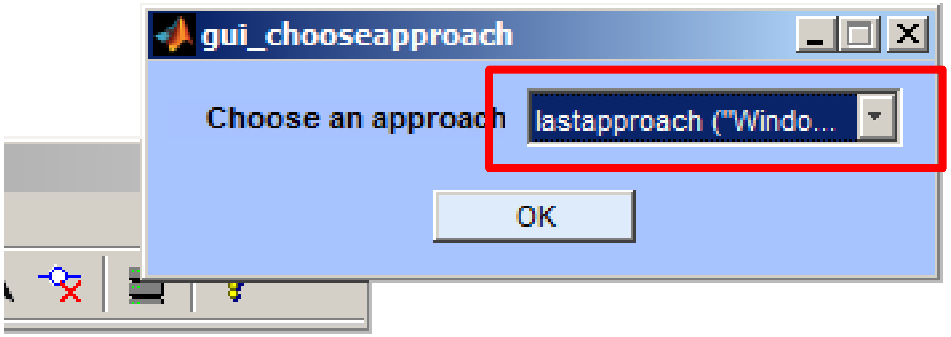


This model uses a minimal subset of channels.

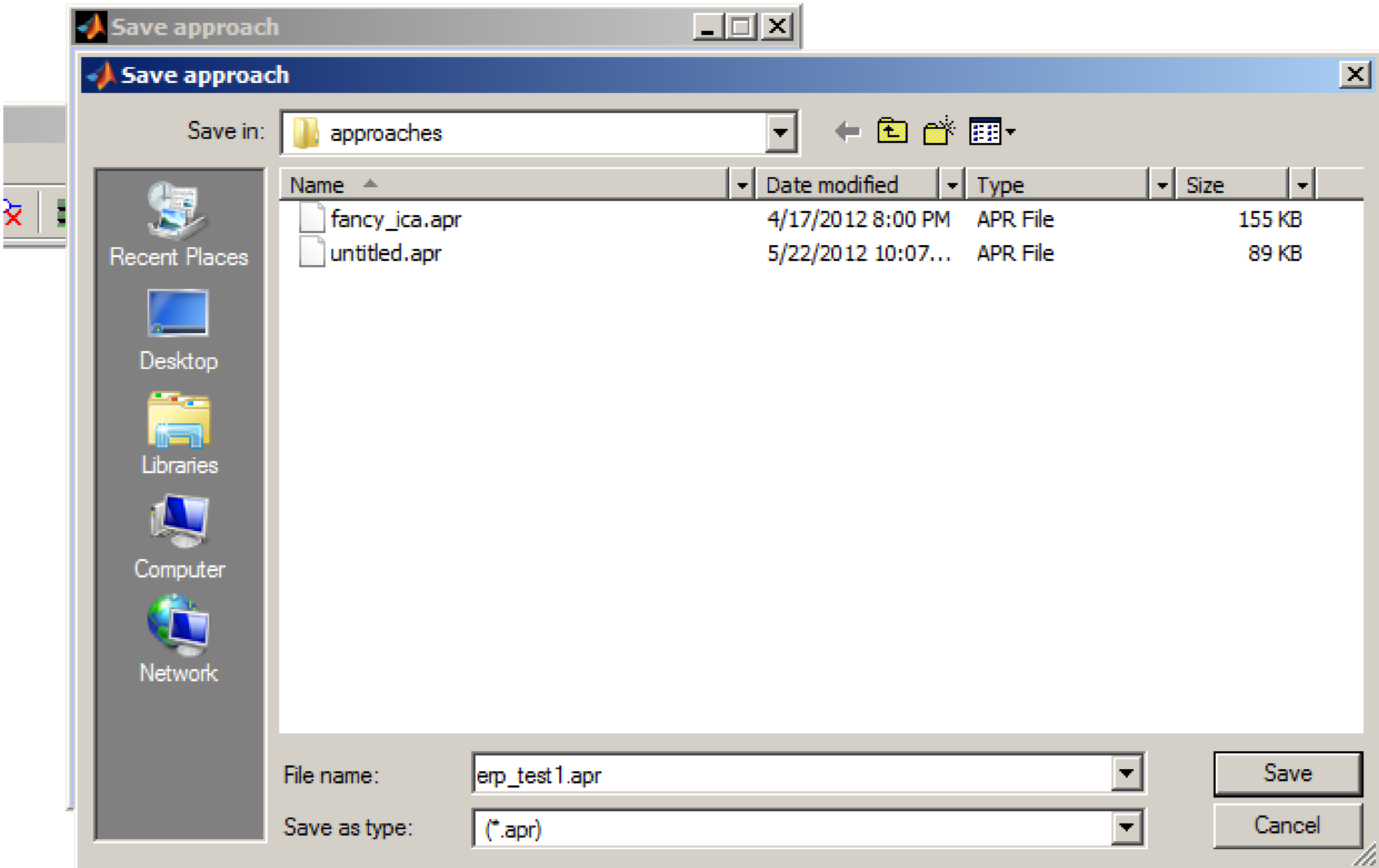
Saving the Approach for Later



Saving the Approach for Later



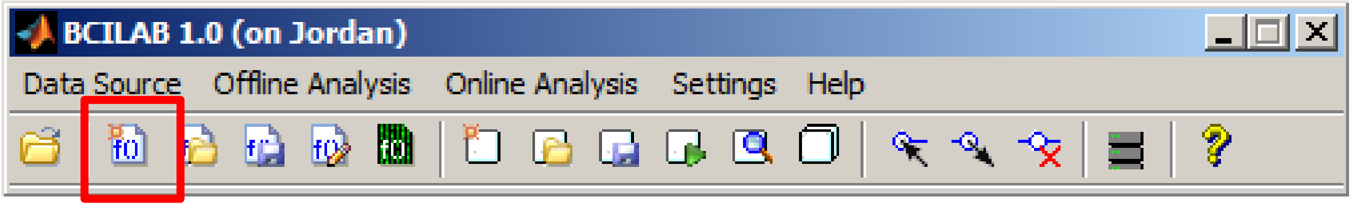
Saving the Approach for Later



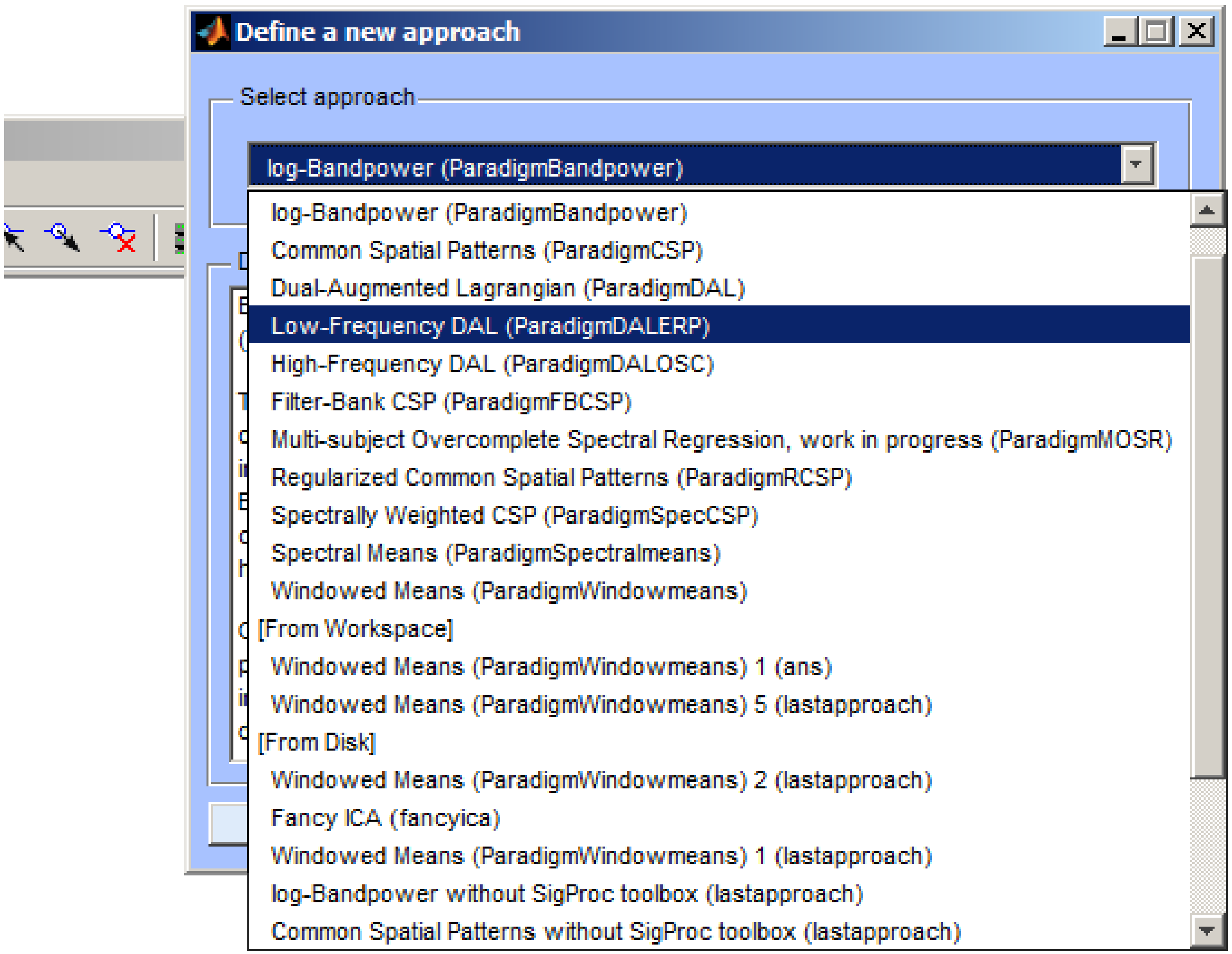
Using a Different BCI Paradigm

(state of the art – this is optional for people with a fast computer)

New Approach….

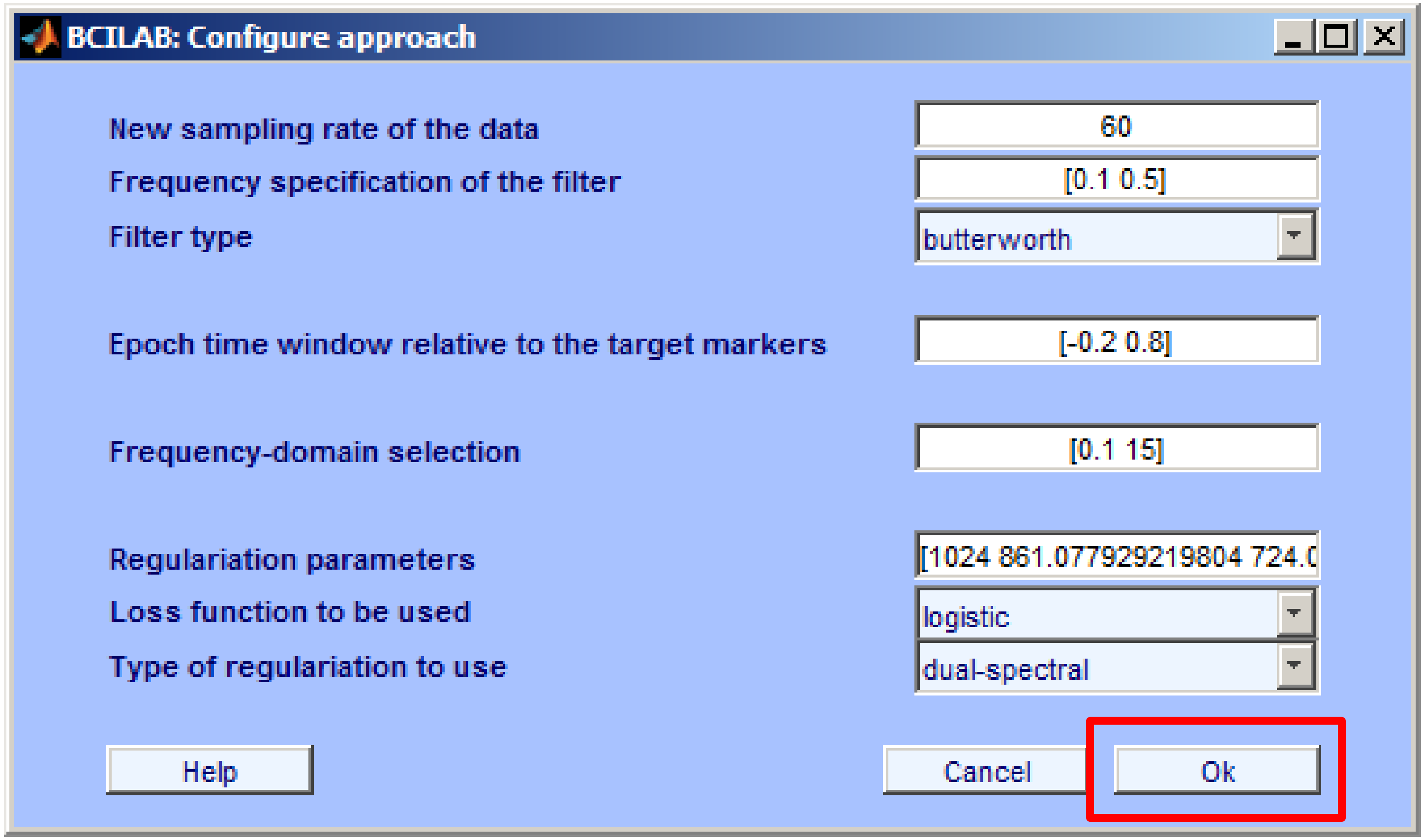


Selecting DAL-ERP

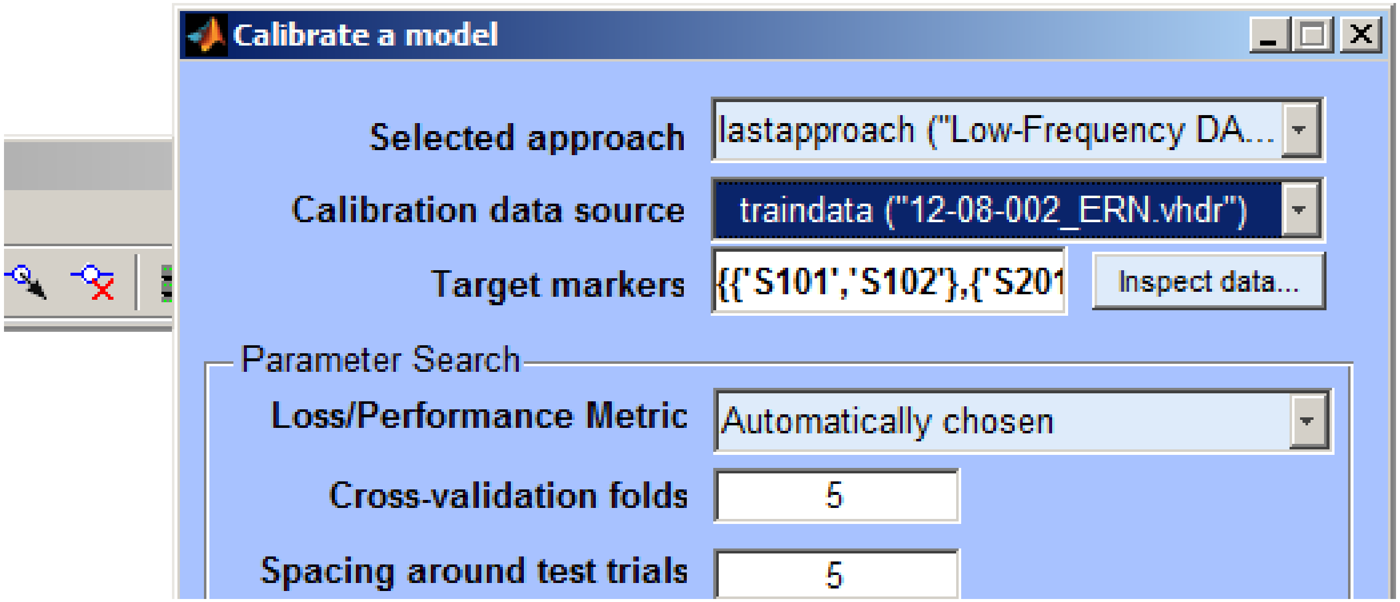


This is one of the best known approaches for ERP-based BCIs. It assumes that there is a small set of latent spatial sources with their own characteristic time course weights, and learns *both simultaneously*.

Configuring DAL-ERP



Calibrate Model…

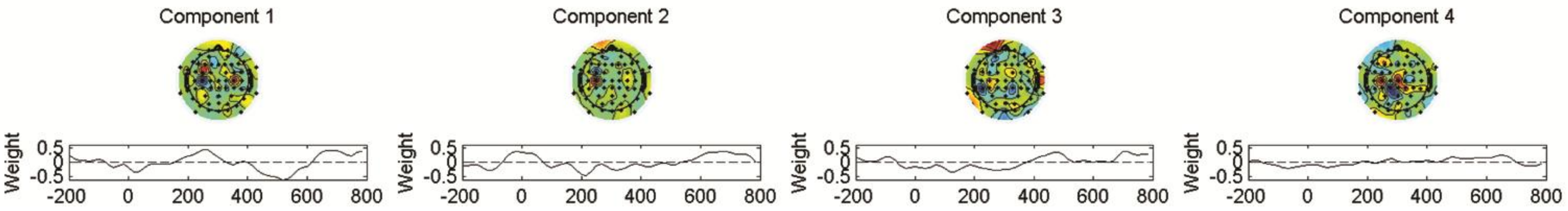


Wait for a Few Minutes…

```
beginning evaluation...

beginning new computation...
pop_epoch():398 epochs selected
Epoching...
pop_epoch():398 epochs generated
eeg_checkset: found empty values for field 'target'
              filling with values of other events in
pop_epoch(): checking epochs for data discontinuity
learning ensemble...
  scanning lambda = 1024.000000... model rank = 0
  scanning lambda = 861.077929... model rank = 0
  scanning lambda = 724.077344... model rank = 0
  scanning lambda = 608.874043... model rank = 0
  scanning lambda = 512.000000... model rank = 0
  scanning lambda = 430.538965... model rank = 1
  scanning lambda = 362.038672... model rank = 1
  scanning lambda = 304.437021... model rank = 1
  scanning lambda = 256.000000... model rank = 1
  scanning lambda = 215.269482... model rank = 1
  scanning lambda = 181.019336... model rank = 1
  scanning lambda = 152.218511... model rank = 1
  scanning lambda = 128.000000... model rank = 2
  scanning lambda = 107.634741... model rank = 3
  scanning lambda = 90.509668... model rank = 3
  scanning lambda = 76.109255... model rank = 3
  scanning lambda = 64.000000... model rank = 3
  scanning lambda = 53.817371... model rank = 3
  scanning lambda = 45.254834... model rank = 4
  scanning lambda = 38.054628... model rank = 6
fx  scanning lambda = 32.000000...>>
```

OVR

Some of the Resulting Components



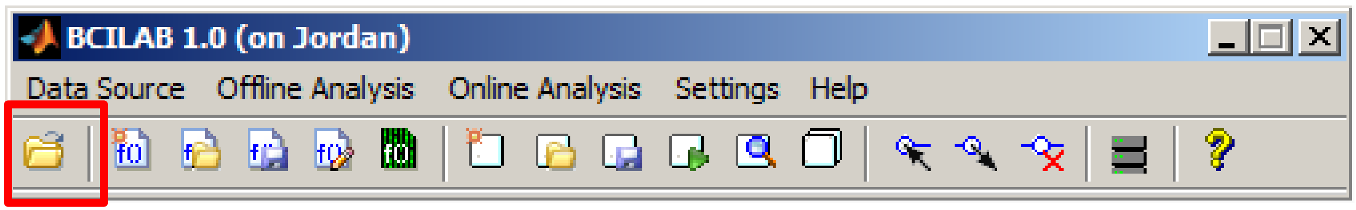
Note that these are the spatial filters rather than the forward projections!

5 Hands-On Oscillatory Process Analysis

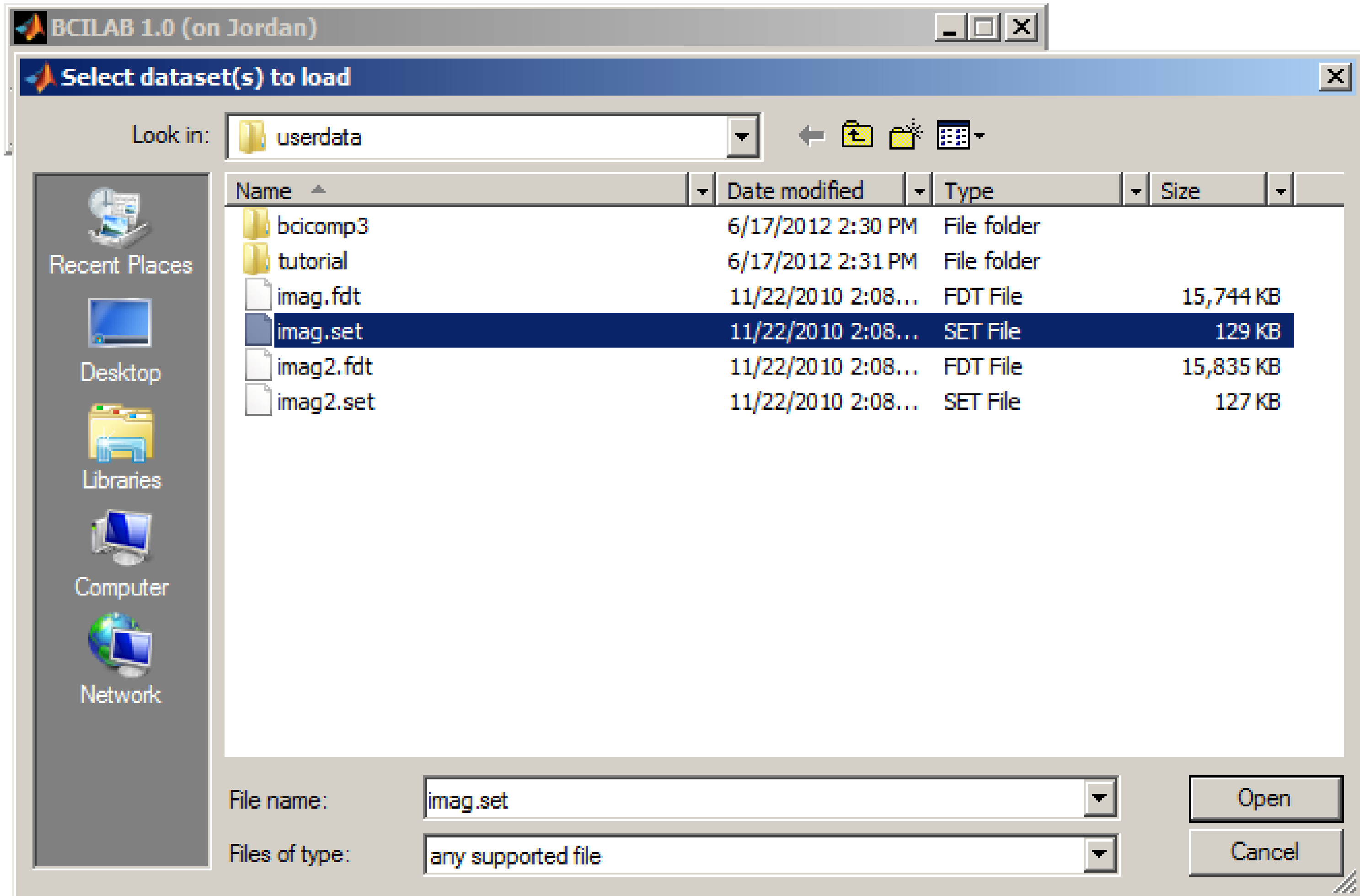
The Data

- The experiment consists of 160 trials (pause at ½ the experiment) . Each trial begins with a letter (either L or R) displayed for 3s. The subject is instructed to subsequently **imagine either a left-hand or a right-hand movement**. Each trial ends with a blank screen displayed for 3.5s.

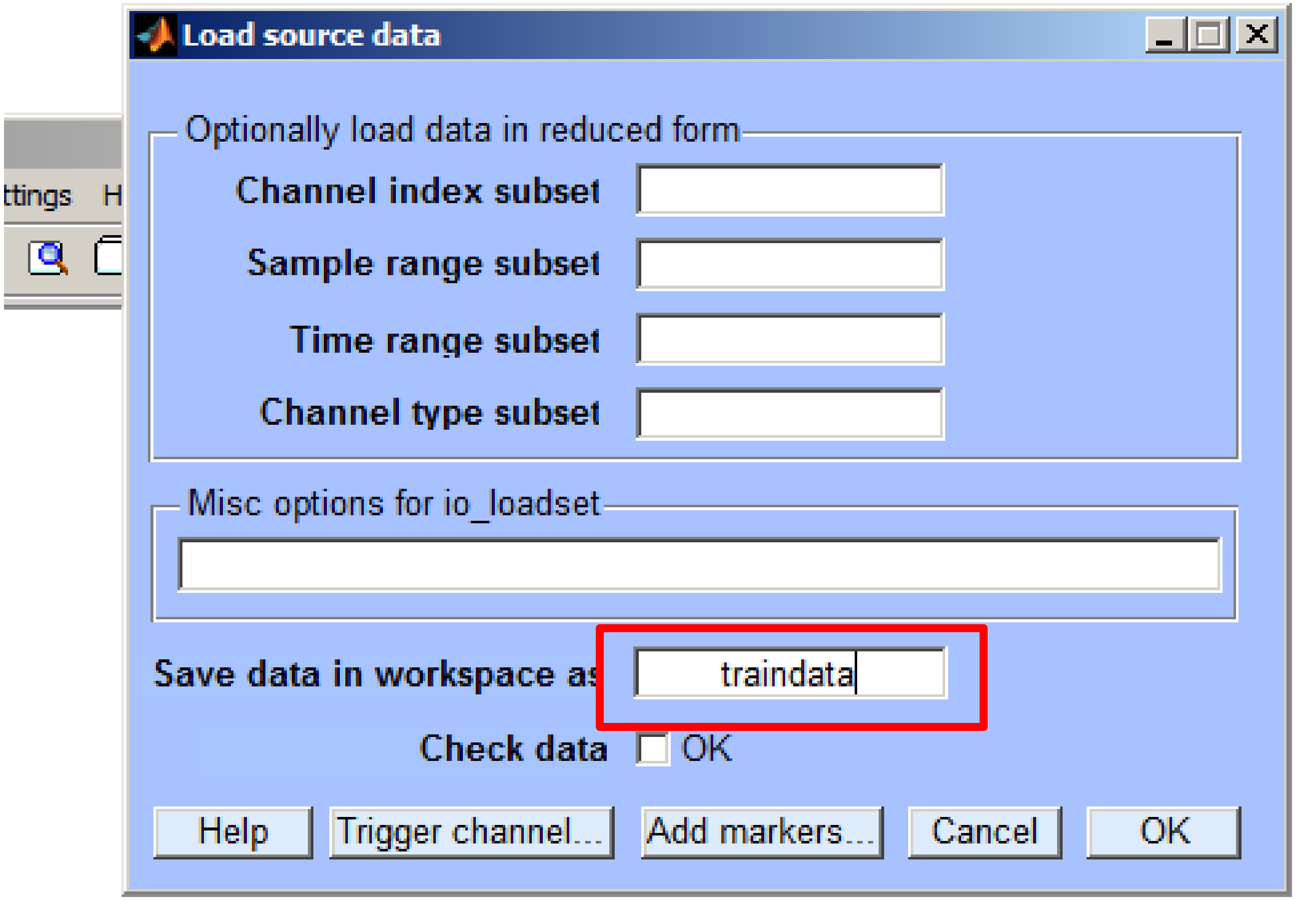
Loading the Data….



Loading the Data…

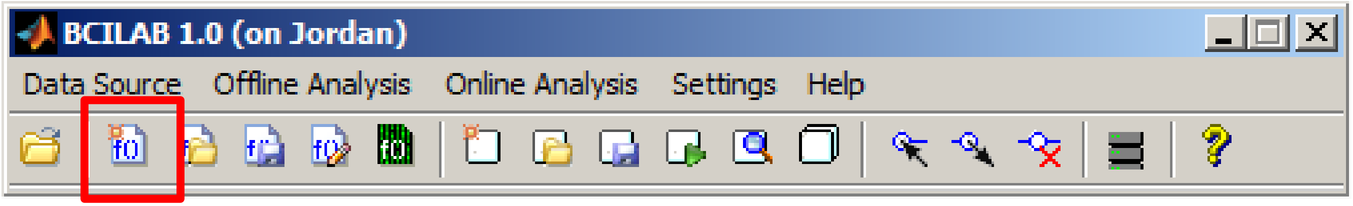


Name It

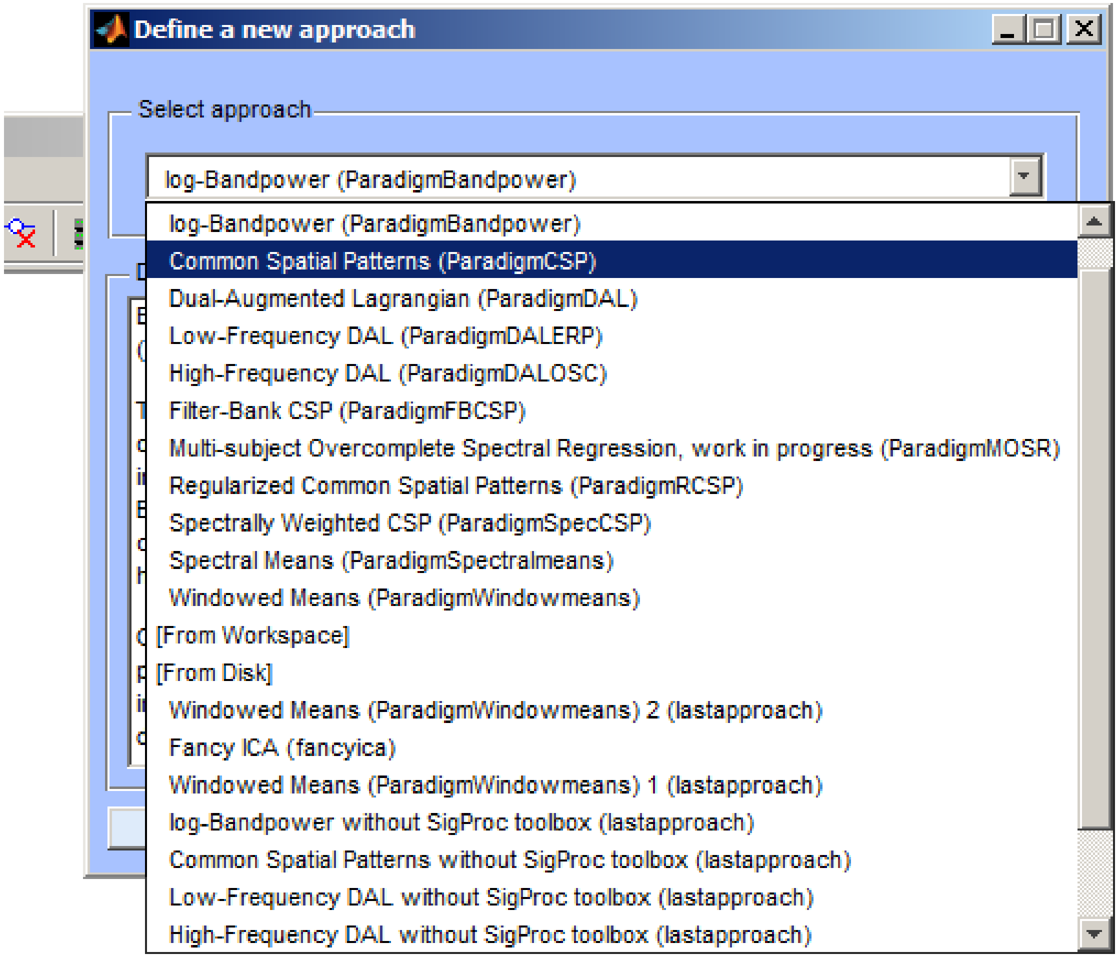


Load source data
Optionally load data in reduced form
Channel index subset
Sample range subset
Time range subset
Channel type subset
Misc options for io_loadset
Save data in workspace as
traindata
Check data
OK
Help
Trigger channel...
Add markers...
Cancel
OK

New Approach

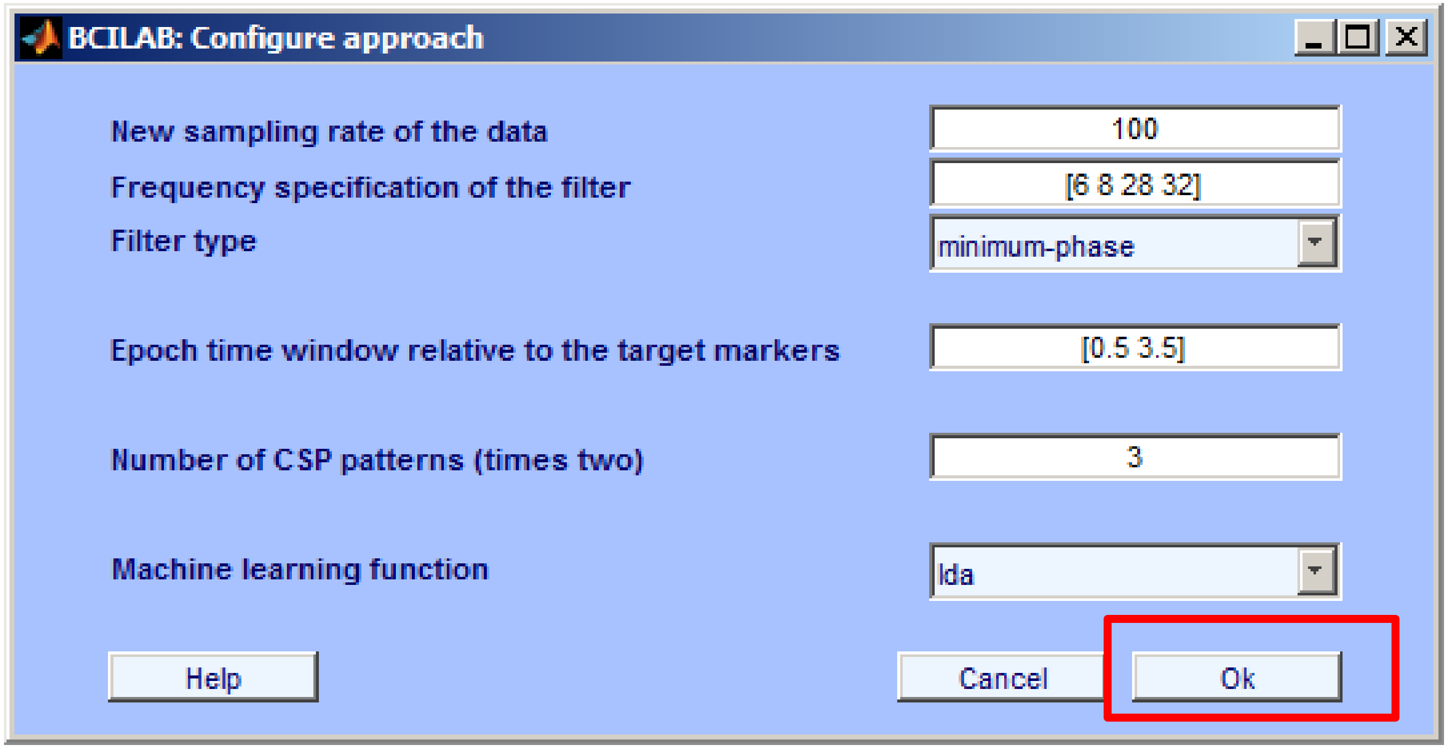


Selecting the Common Spatial Patterns Paradigm

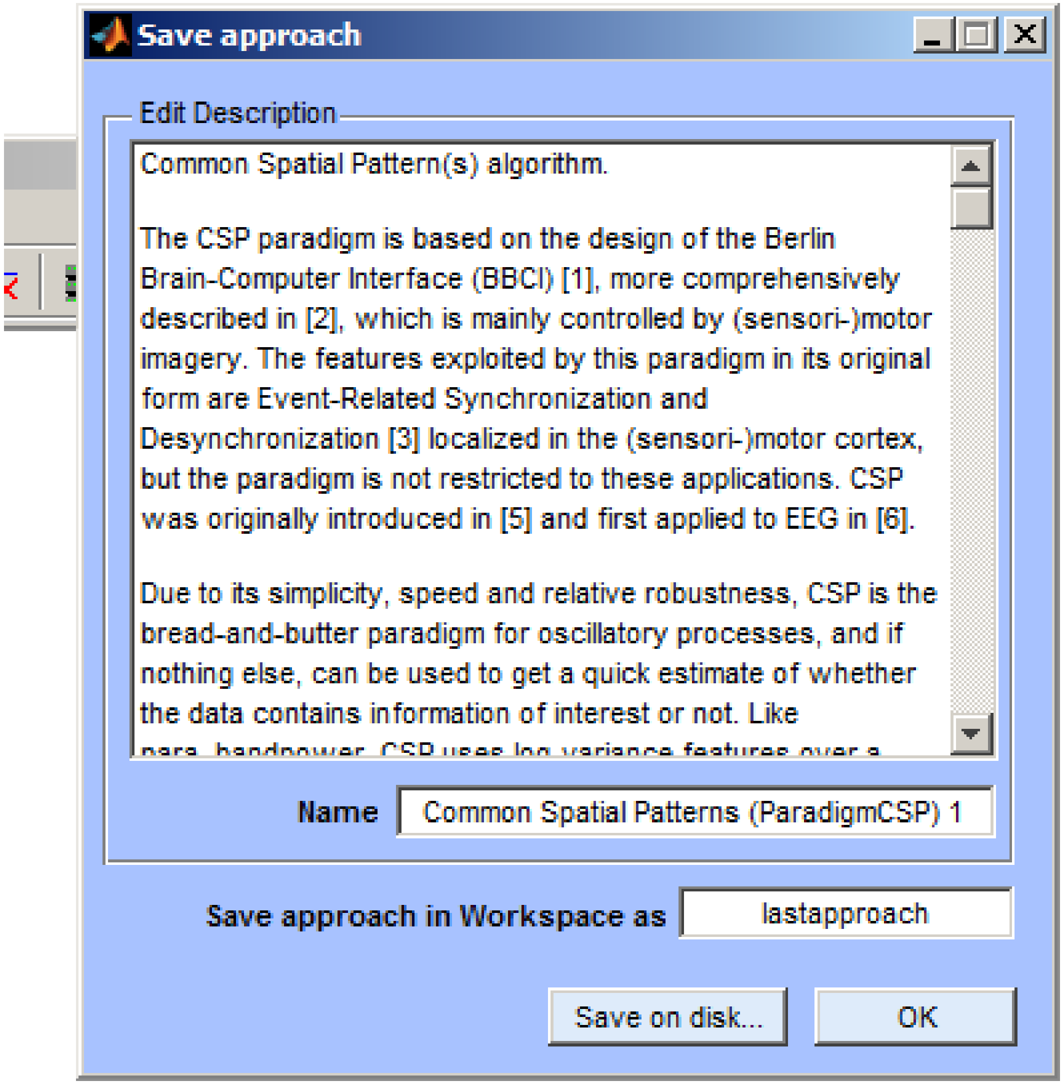


Note: If you don't have a signal processing license, try instead the method "Common Spatial Patterns without SigProc toolbox" at the bottom.

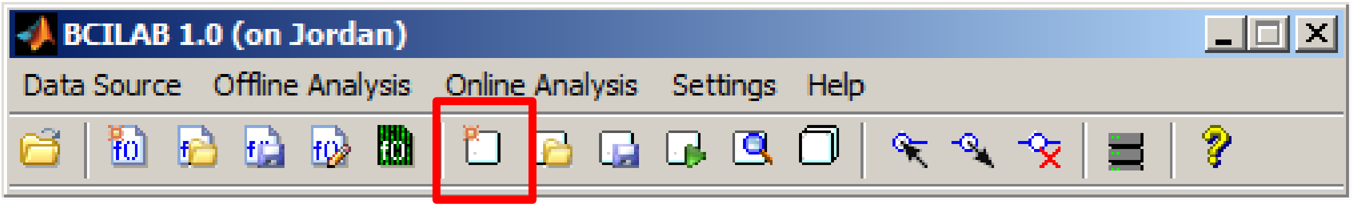
Configuring the Approach



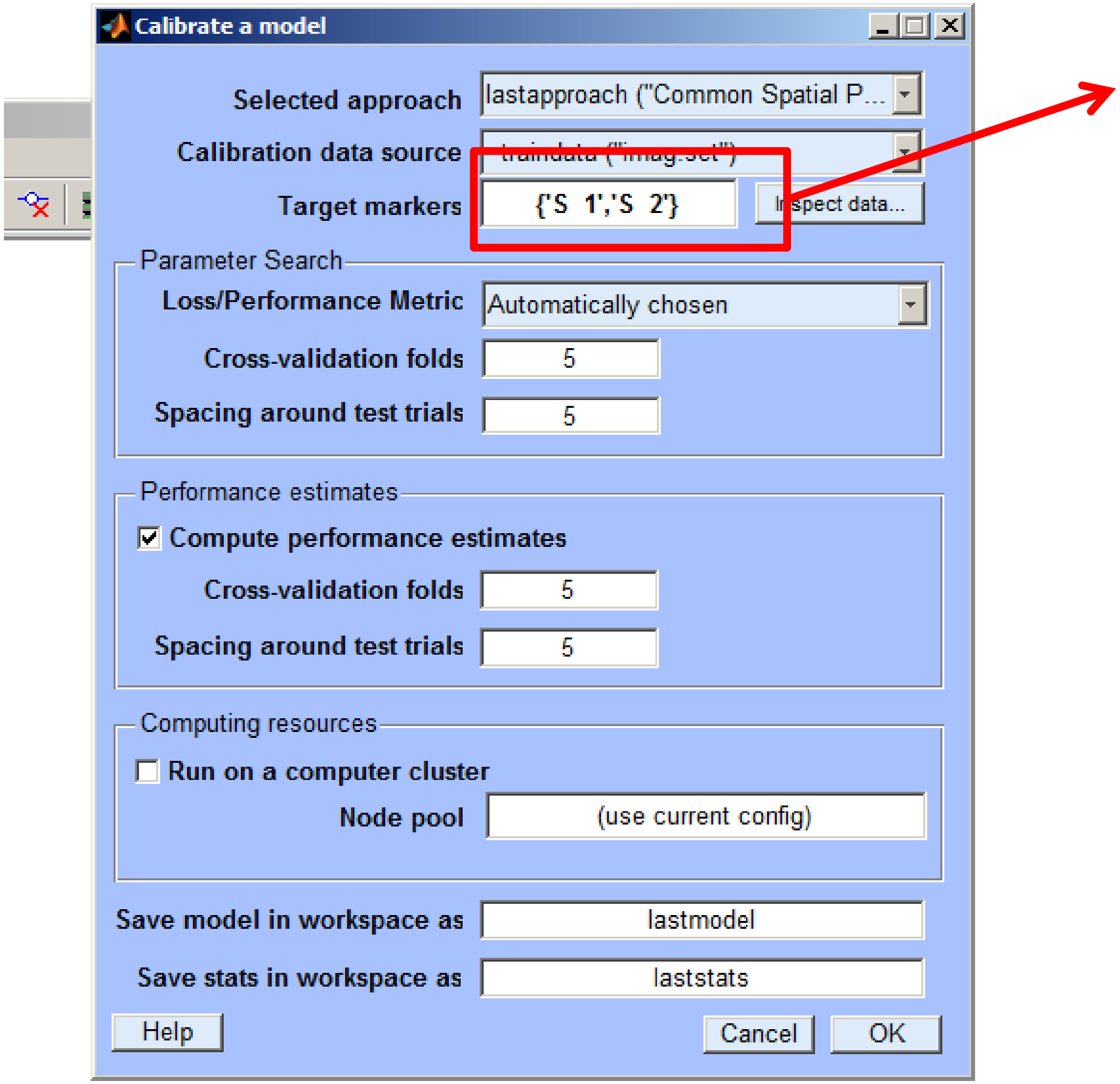
Save Approach to Workspace



Calibrating a New Model



Calibrating a New Model

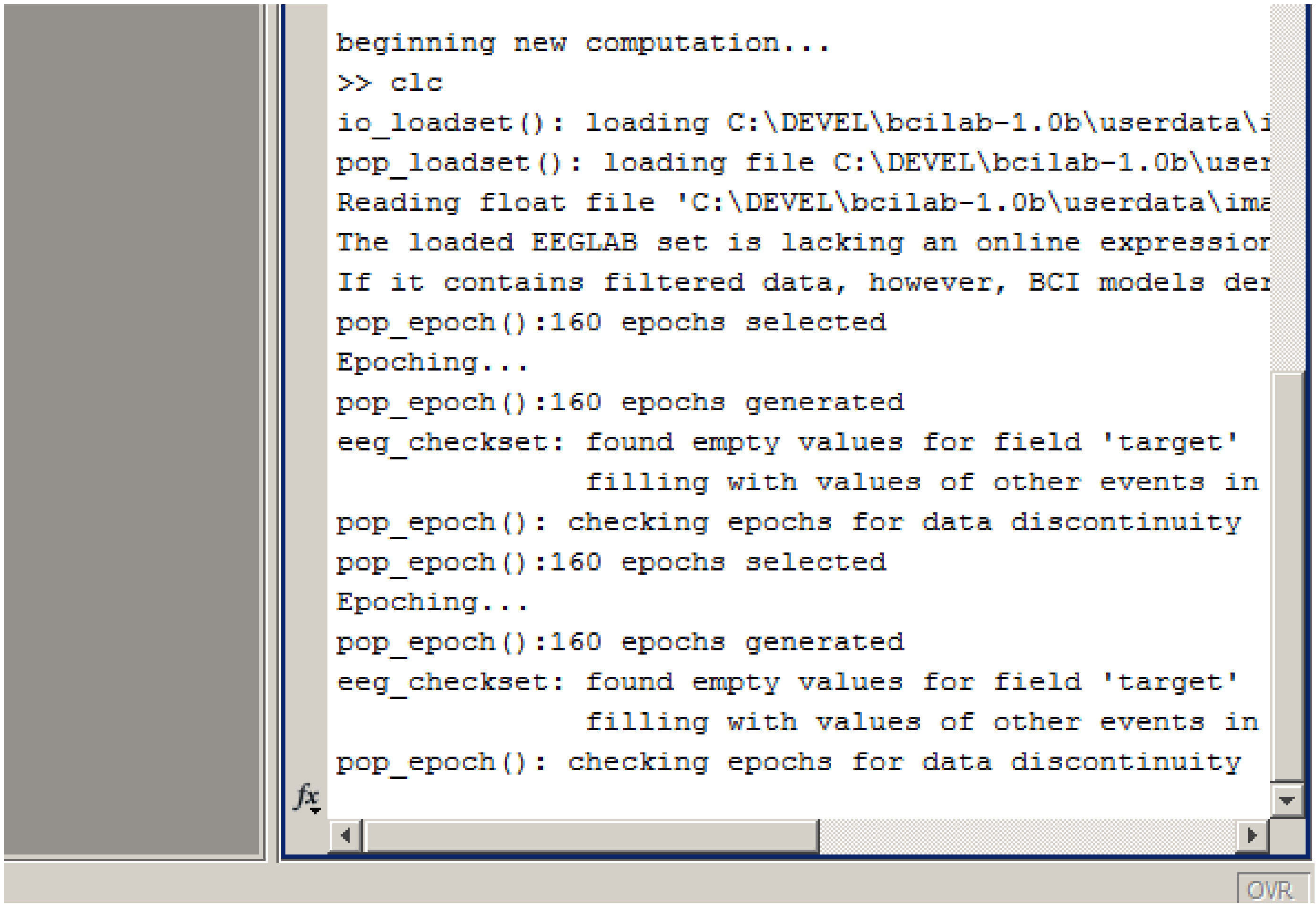


There are only two markers here, one per condition (no sub-groups). Type the following here: **{'S 1','S 2'}**

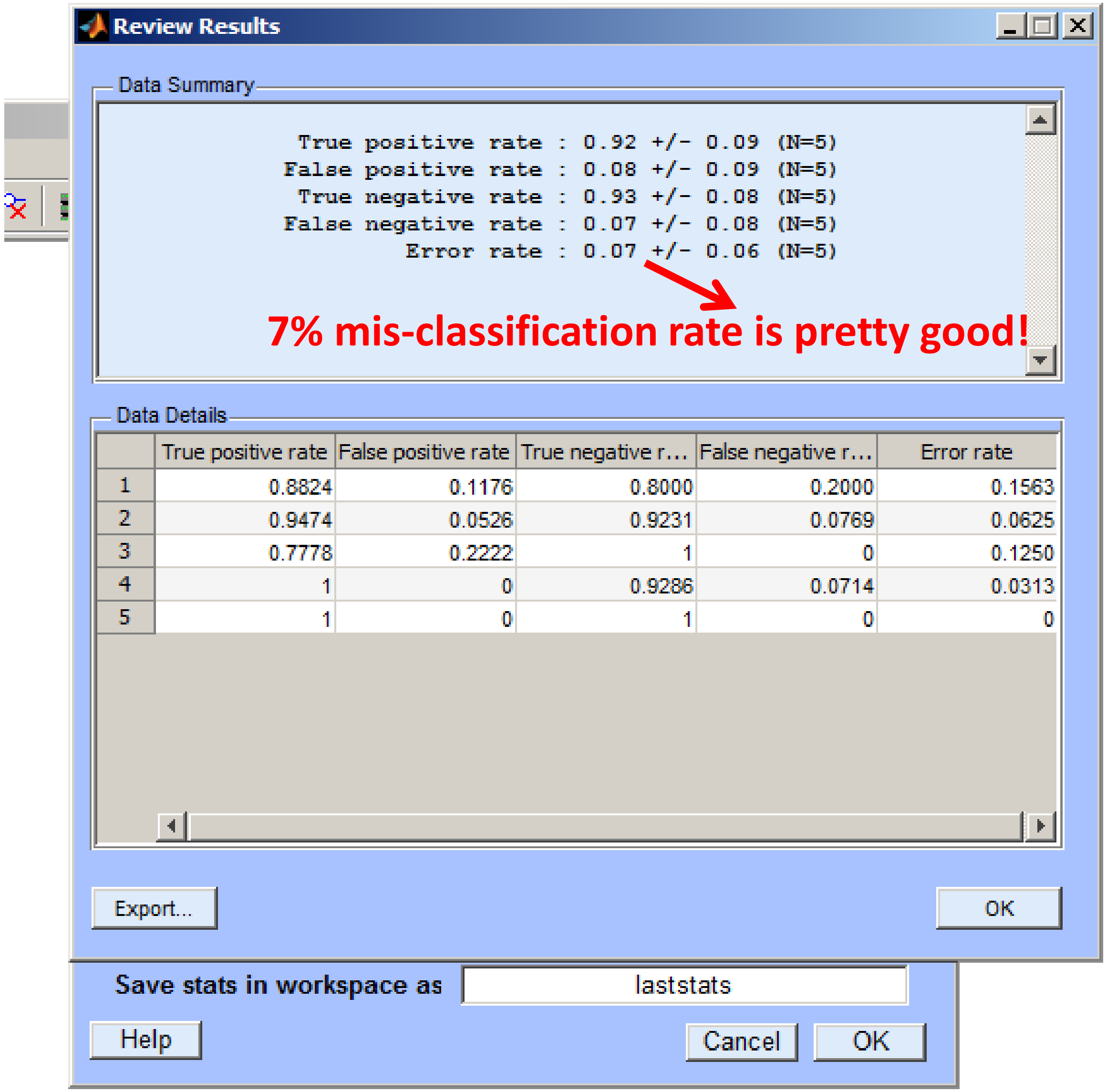
IMPORTANT: There are 2 space characters between the S and the number!

The marker "S 1" indicates the moment when the subject is instructed to imagine a left-hand movement, and the marker "S 2" indicates the moment when a right-hand movement should be imagined.

Watching the Computation…

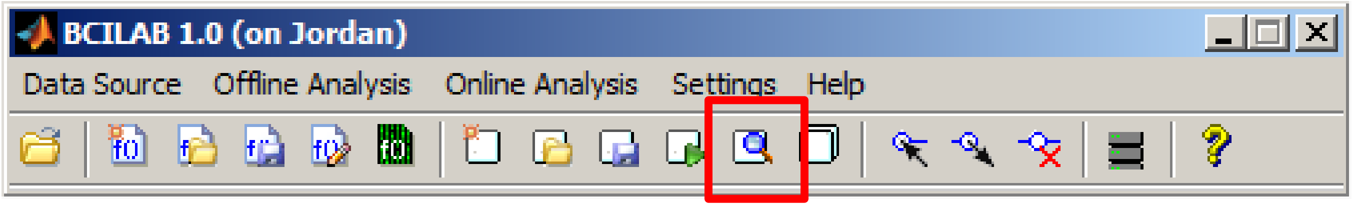


Reviewing Results

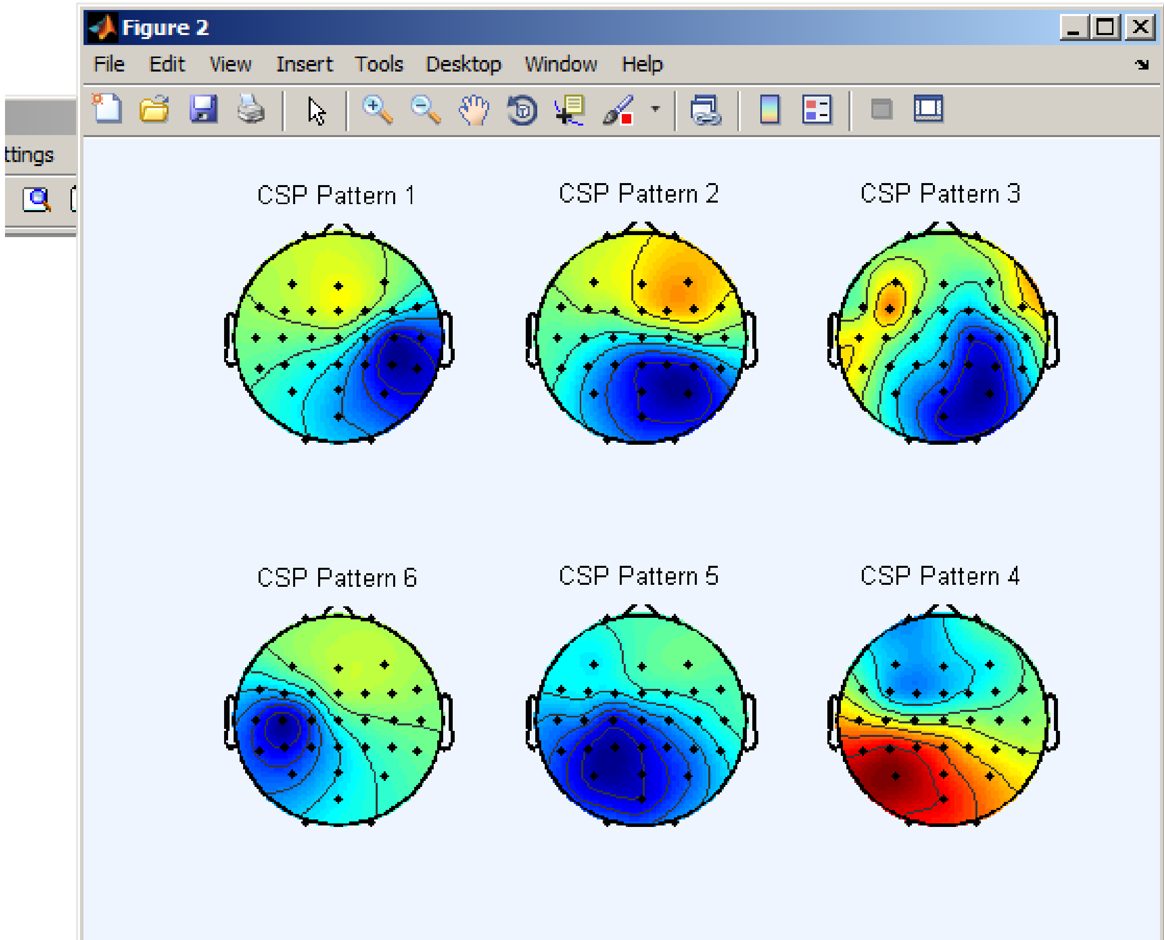


	True positive rate	False positive rate	True negative r...	False negative r...	Error rate
1	0.8824	0.1176	0.8000	0.2000	0.1563
2	0.9474	0.0526	0.9231	0.0769	0.0625
3	0.7778	0.2222	1	0	0.1250
4	1	0	0.9286	0.0714	0.0313
5	1	0	1	0	0

Visualizing the Model

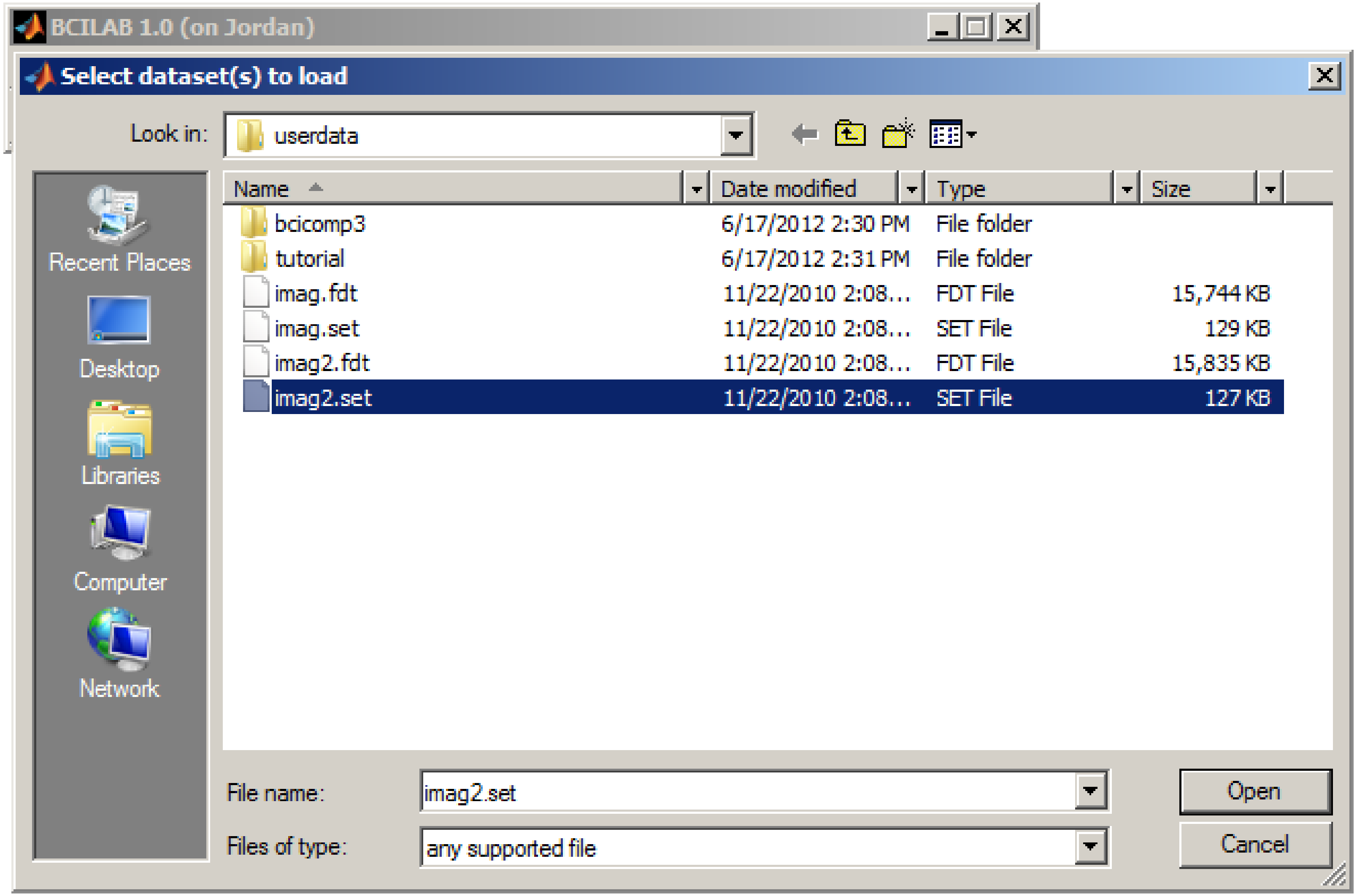


Visualizing the Model

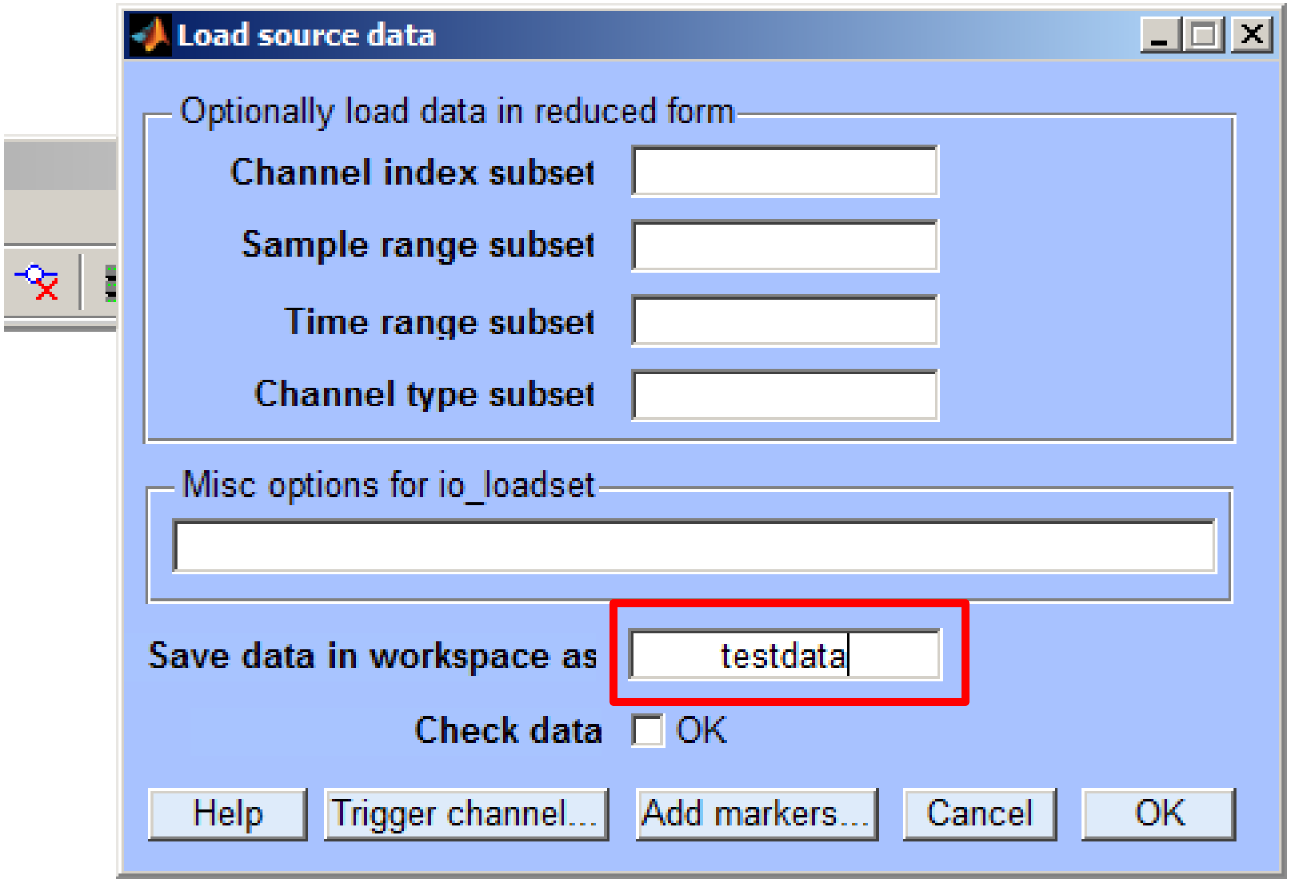


Note: these are not the spatial filters but the source forward projections .

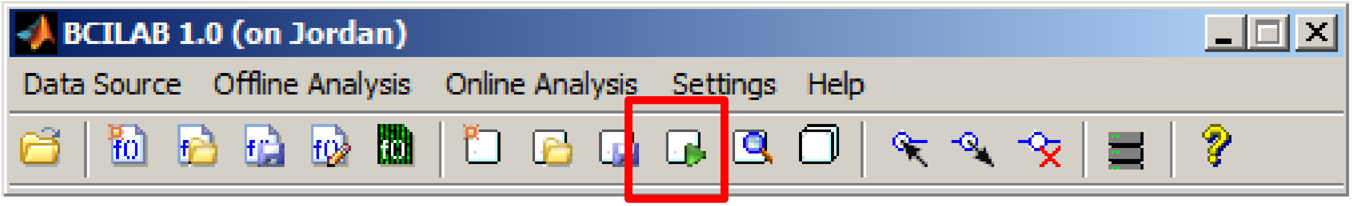
Loading A Separate Test Set



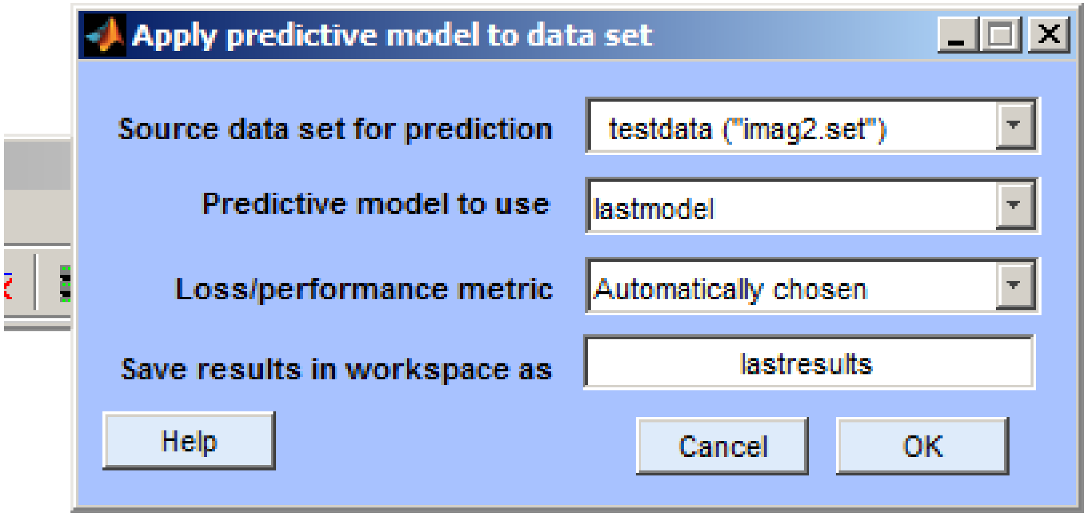
Naming the Test Set



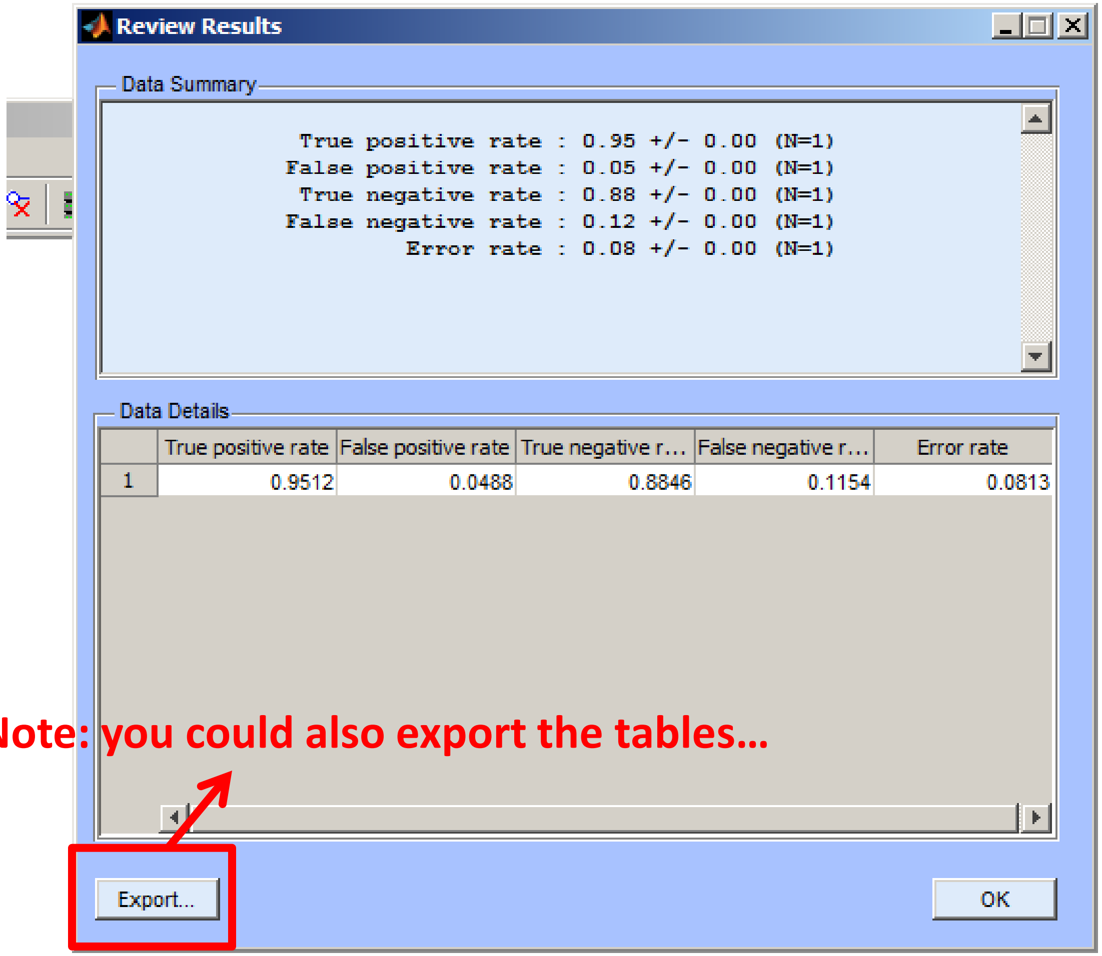
Applying the Model to the Test Data



Applying the Model to the Test Data

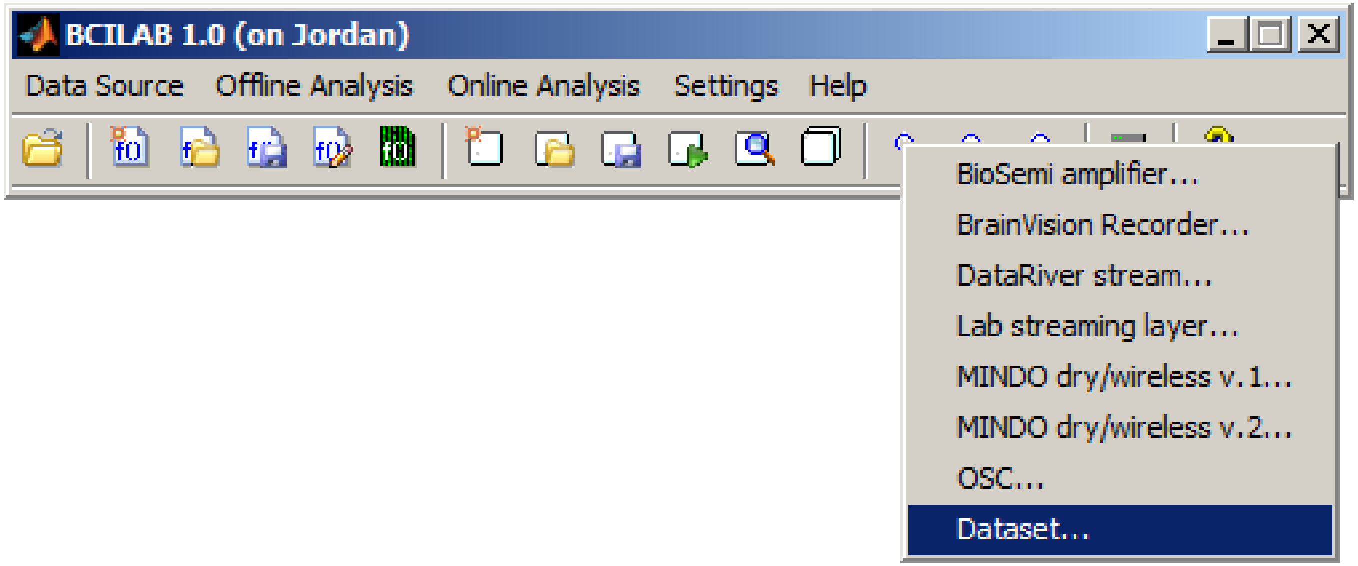


Reviewing Results



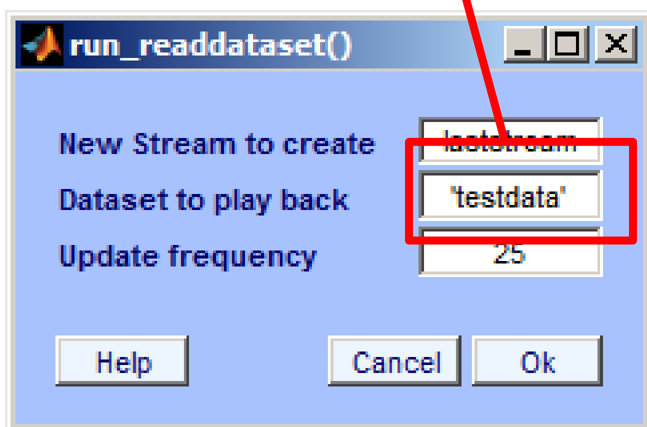
Note: you could also export the tables…

Starting Online Analysis

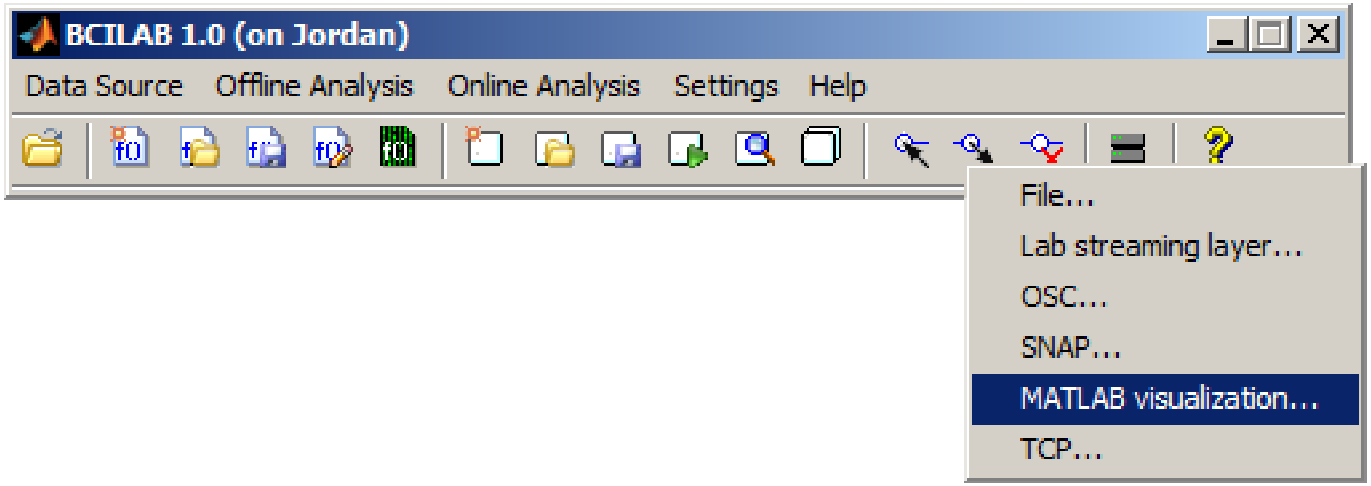


Starting Online Analysis

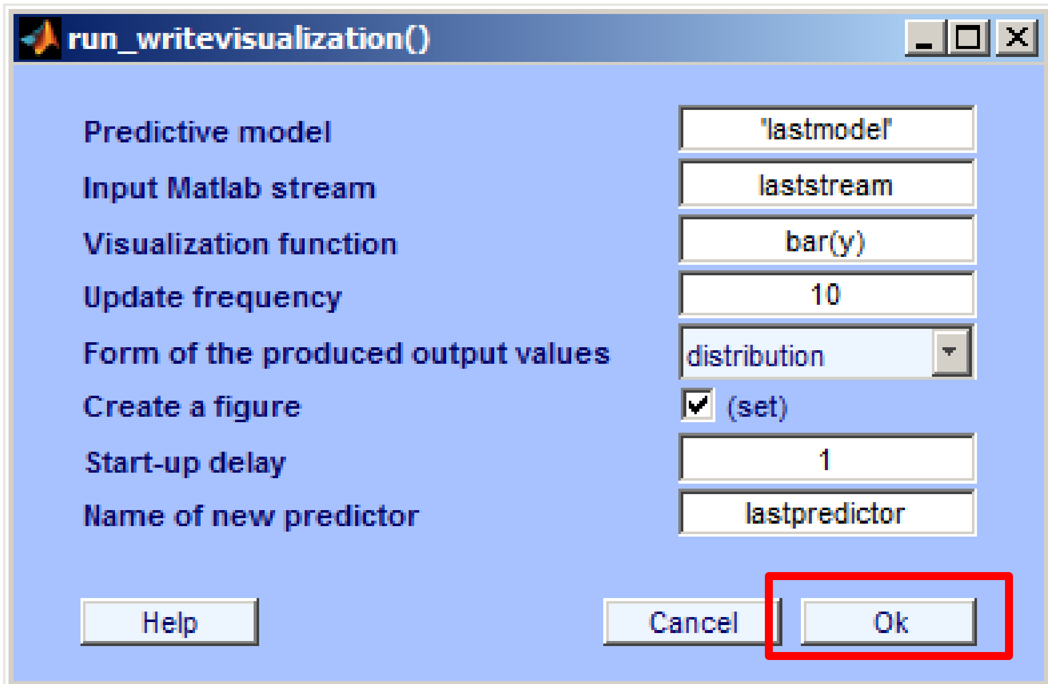
IMPORTANT: Put the name of your test data set here!



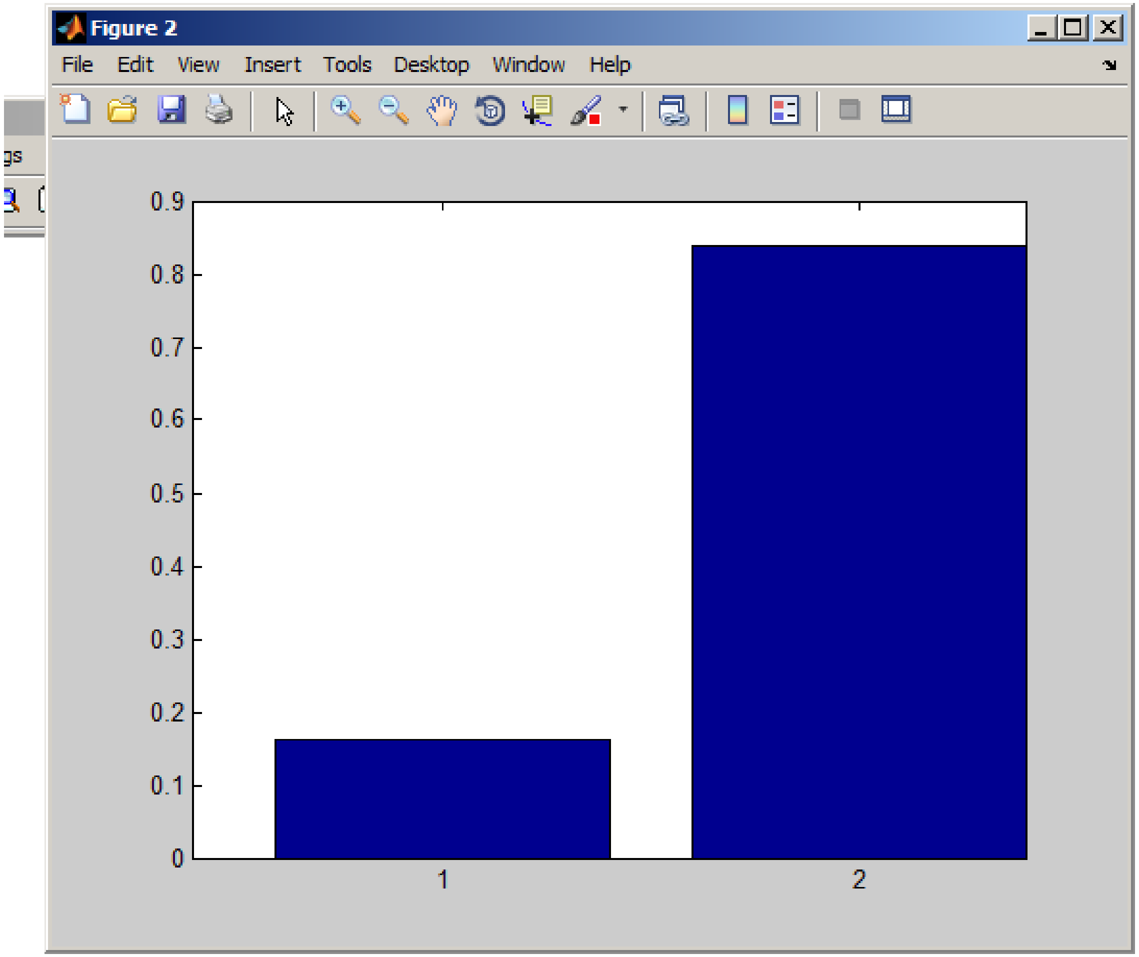
Select Output Destination



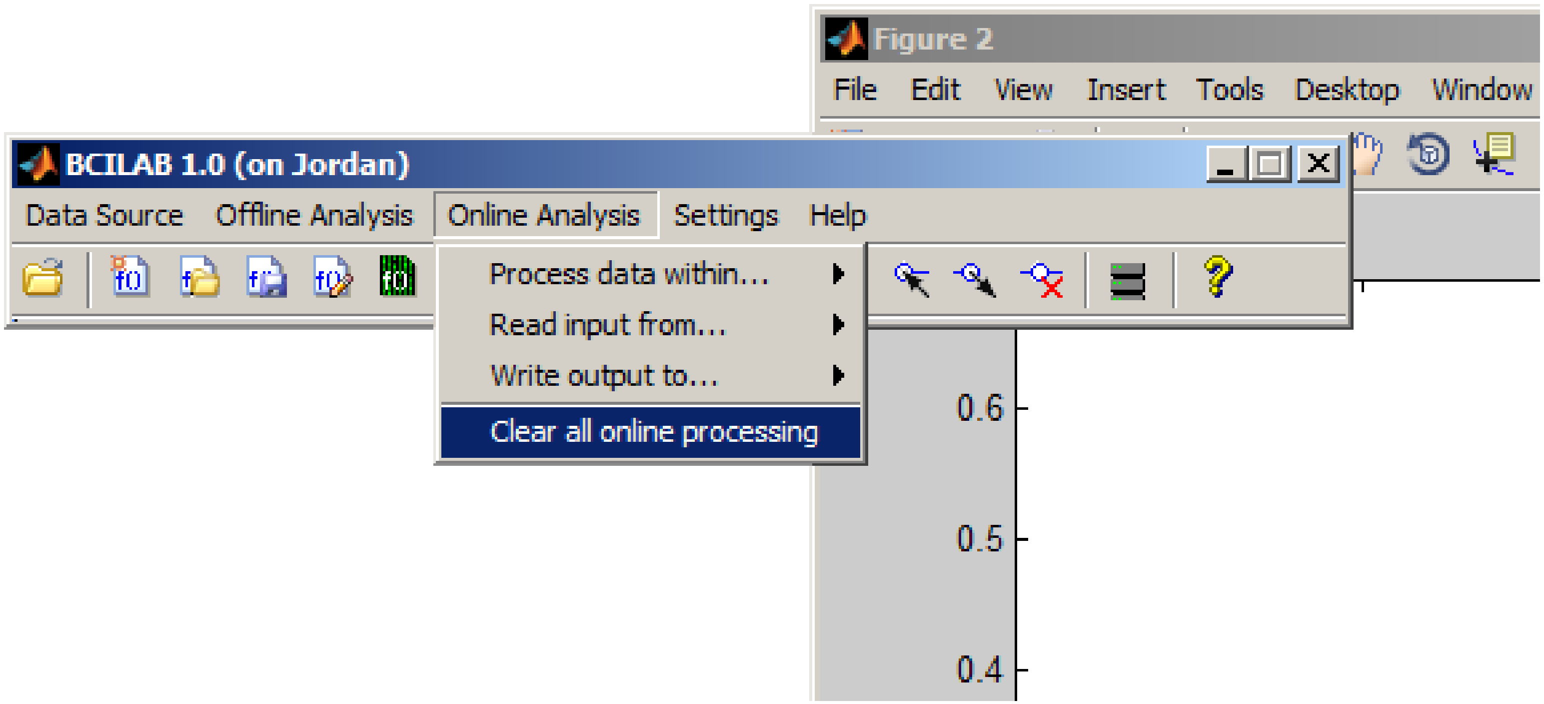
Configure/Confirm



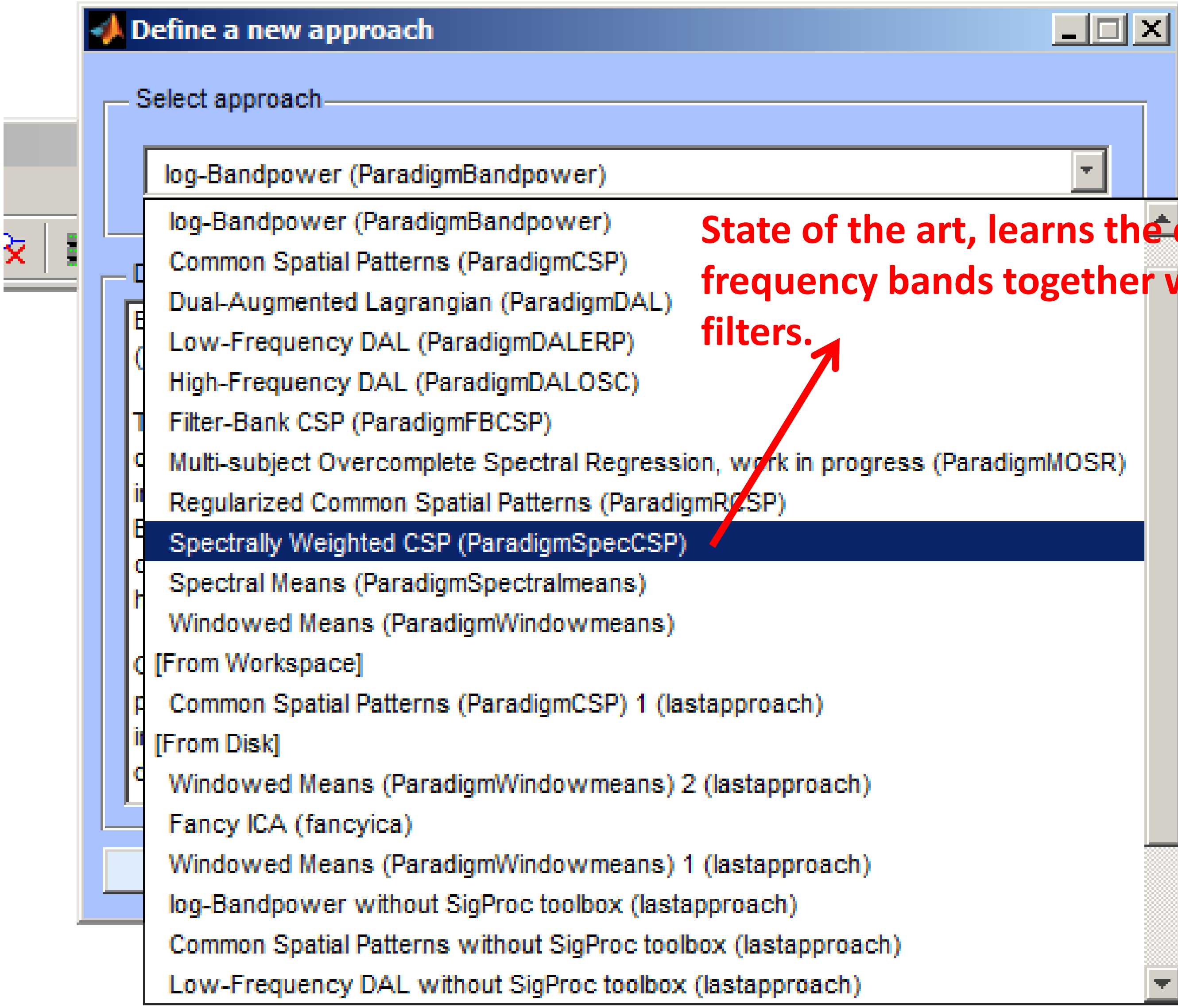
Online Output



Clear Online Processing

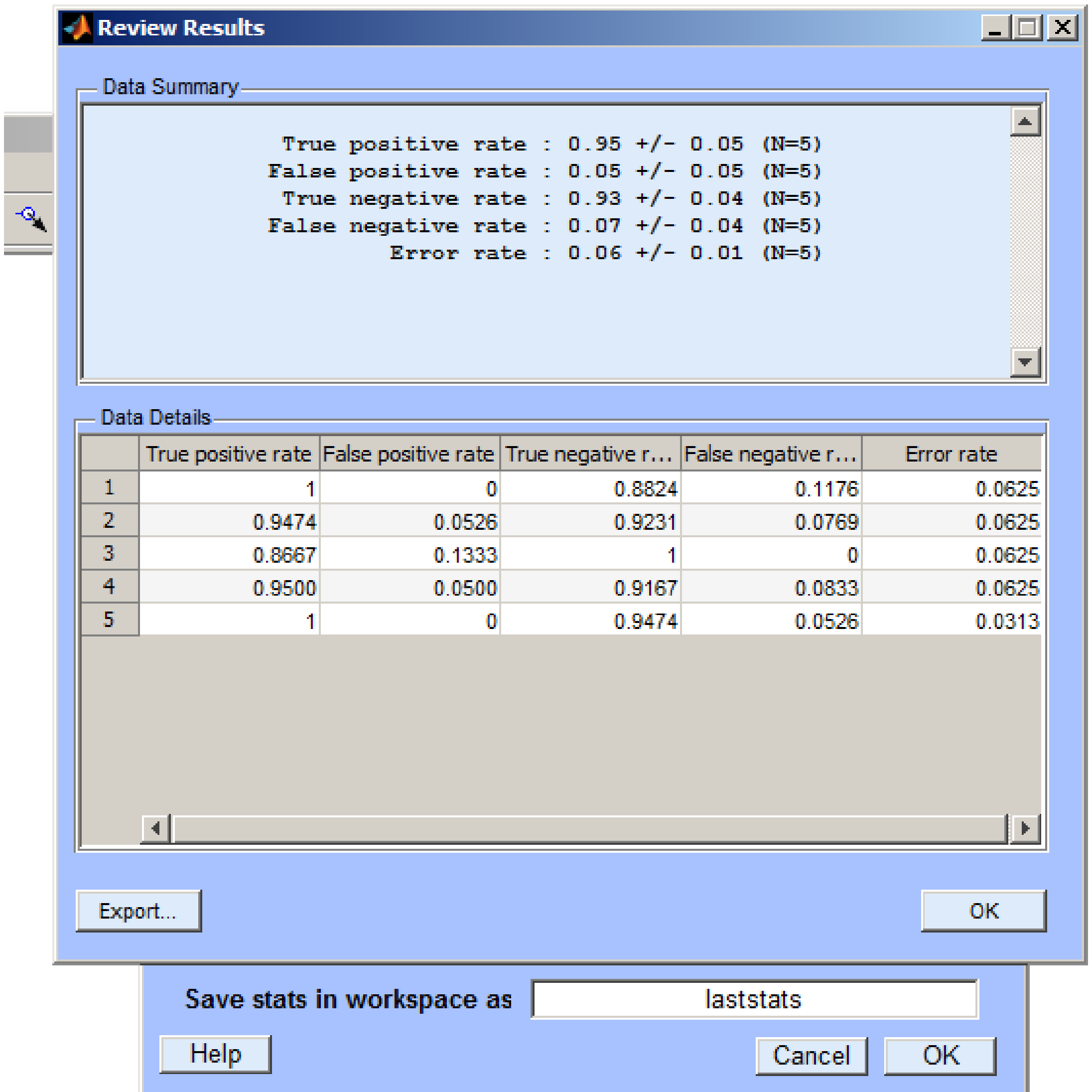


Alternative Paradigms: Spec-CSP



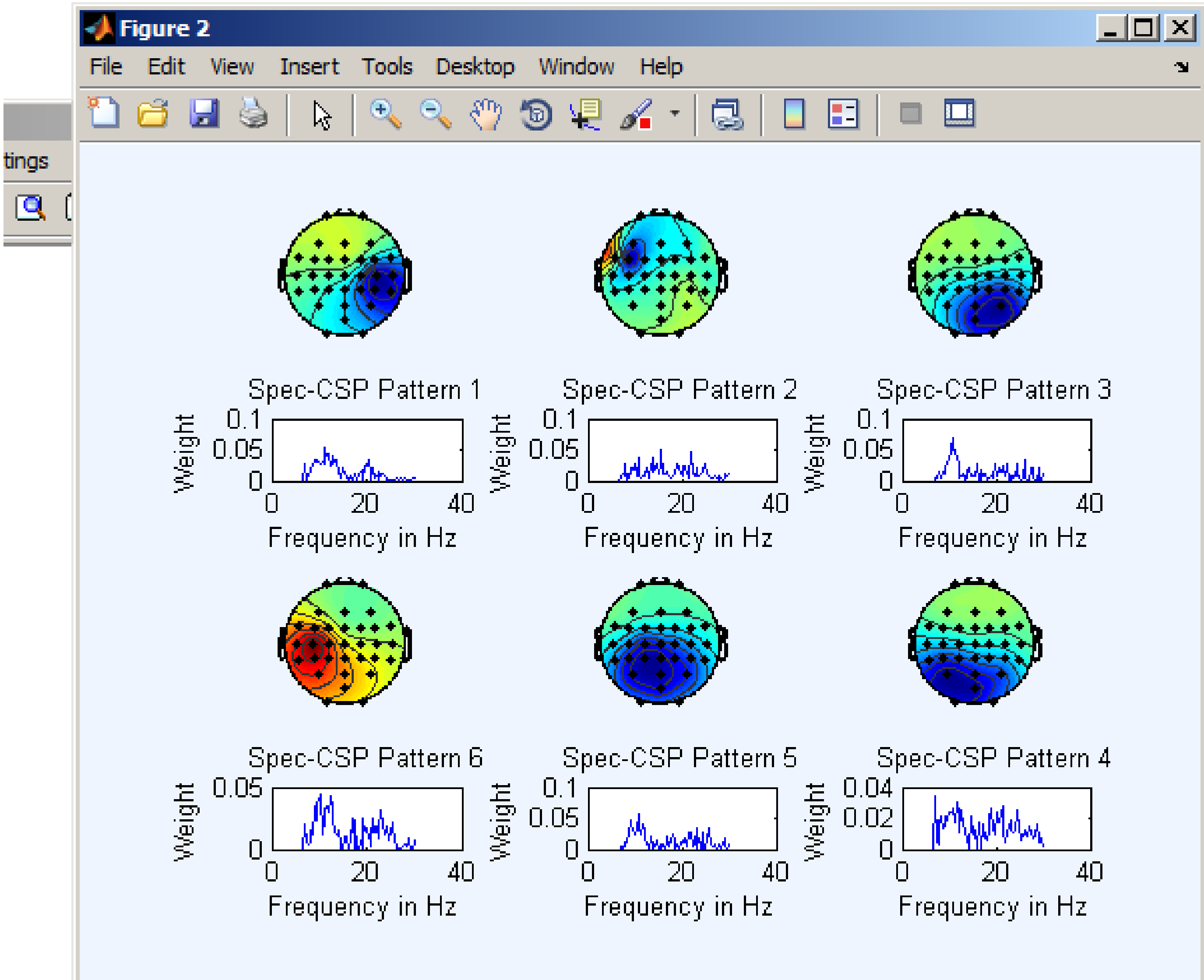
State of the art, learns the correct frequency bands together with spatial filters.

Alternative Paradigms: Spec-CSP



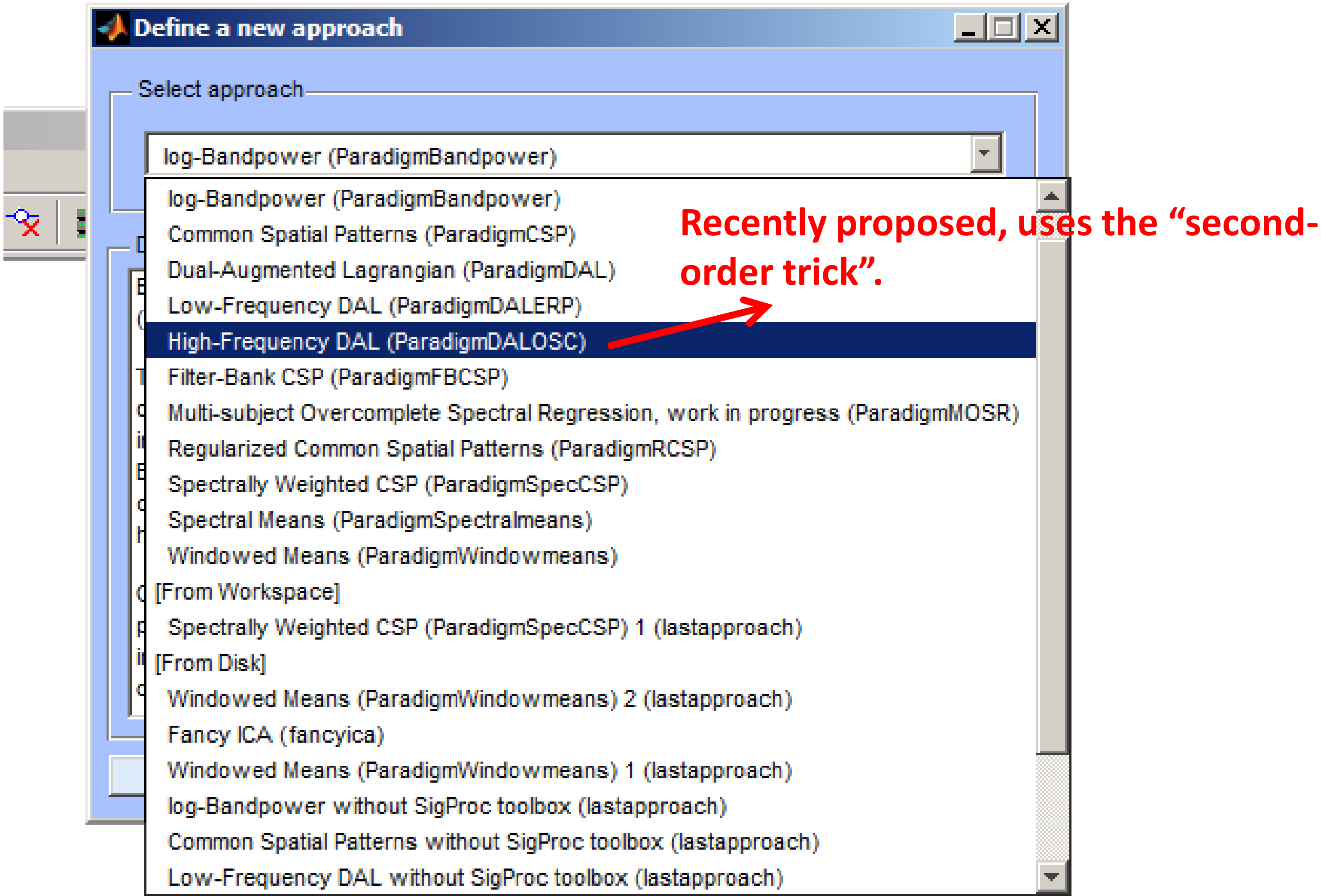
	True positive rate	False positive rate	True negative r...	False negative r...	Error rate
1	1	0	0.8824	0.1176	0.0625
2	0.9474	0.0526	0.9231	0.0769	0.0625
3	0.8667	0.1333	1	0	0.0625
4	0.9500	0.0500	0.9167	0.0833	0.0625
5	1	0	0.9474	0.0526	0.0313

Alternative Paradigms: Spec-CSP



Alternative Paradigms: DAL-OSC

Note: this takes several minutes of computation – only for fast computers!



Alternative Paradigms: DAL-OSC

- Watching the computation…

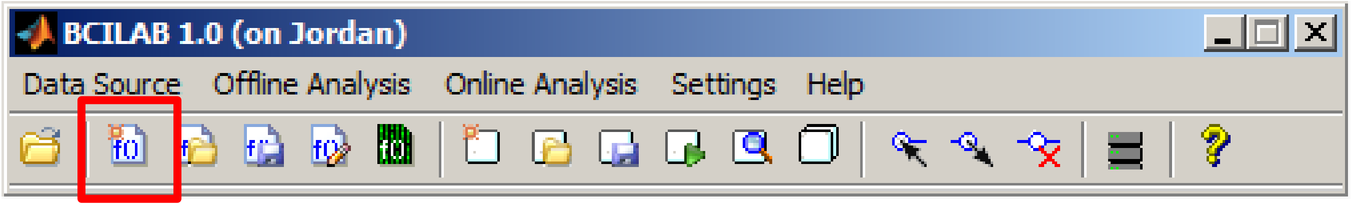
```
scanning lambda = 861.077929... model rank = 0
scanning lambda = 724.077344... model rank = 0
scanning lambda = 608.874043... model rank = 0
scanning lambda = 512.000000... model rank = 0
scanning lambda = 430.538965... model rank = 0
scanning lambda = 362.038672... model rank = 0
scanning lambda = 304.437021... model rank = 0
scanning lambda = 256.000000... model rank = 1
scanning lambda = 215.269482... model rank = 2
scanning lambda = 181.019336... model rank = 2
scanning lambda = 152.218511... model rank = 2
scanning lambda = 128.000000... model rank = 2
scanning lambda = 107.634741... model rank = 2
scanning lambda = 90.509668... model rank = 2
scanning lambda = 76.109255... model rank = 2
scanning lambda = 64.000000... model rank = 2
scanning lambda = 53.817371... model rank = 2
scanning lambda = 45.254834... model rank = 2
scanning lambda = 38.054628... model rank = 3
scanning lambda = 32.000000... model rank = 3
scanning lambda = 26.908685... model rank = 4
scanning lambda = 22.627417... model rank = 4
scanning lambda = 19.027314... model rank = 5
scanning lambda = 16.000000... model rank = 5
scanning lambda = 13.454343...>>
```

Alternative Paradigms: DAL-OSC

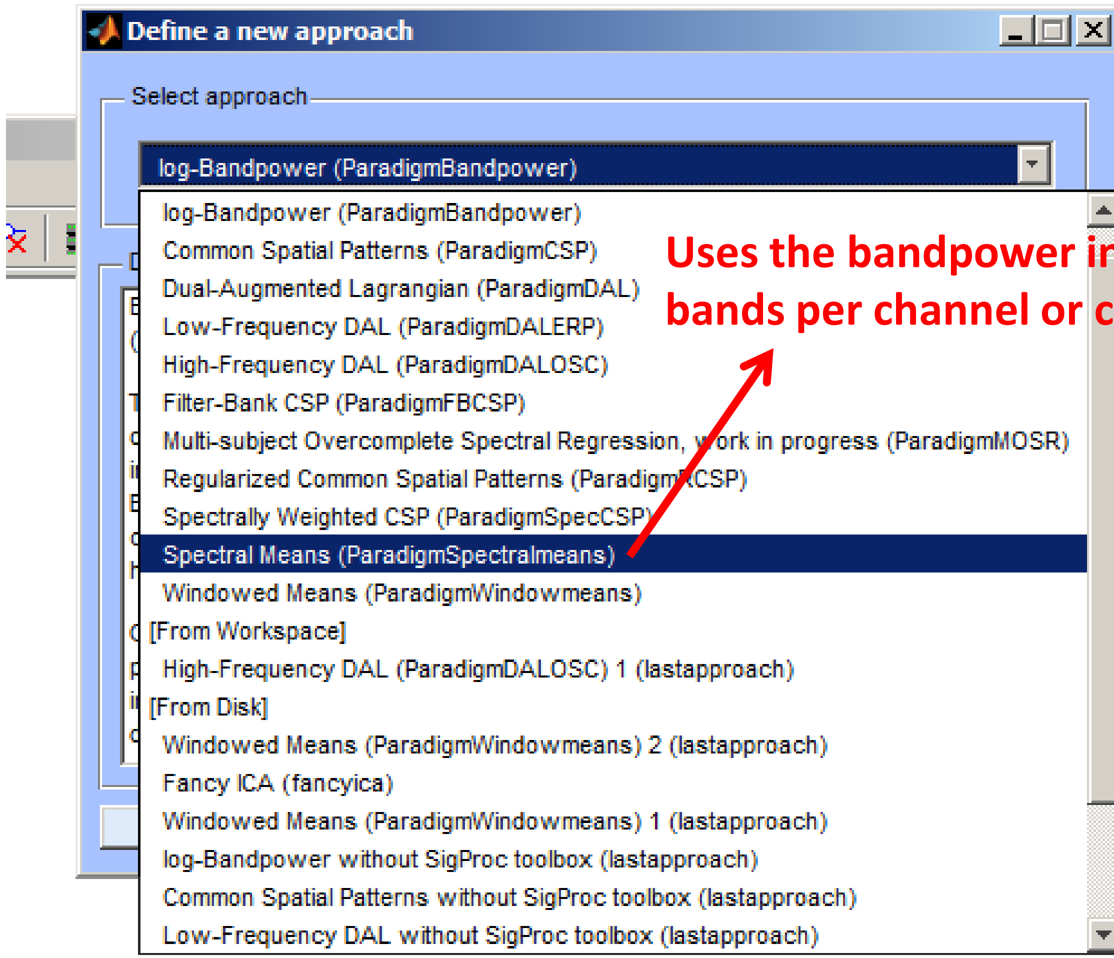
- No visualization yet, but can apply to test-set data or run online

Creating an Entirely Customized Approach

Create a New Approach

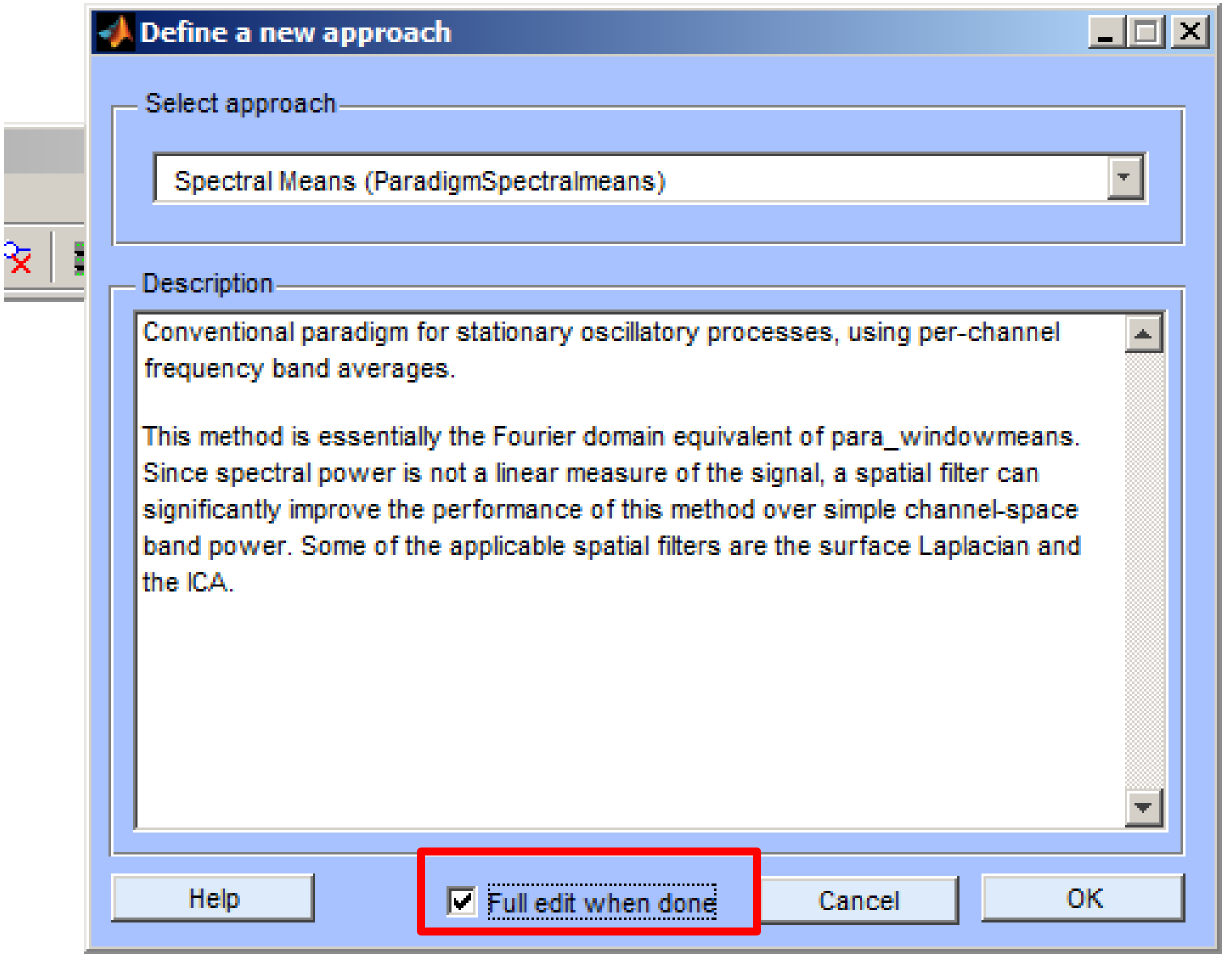


Custom ICA-based Approach

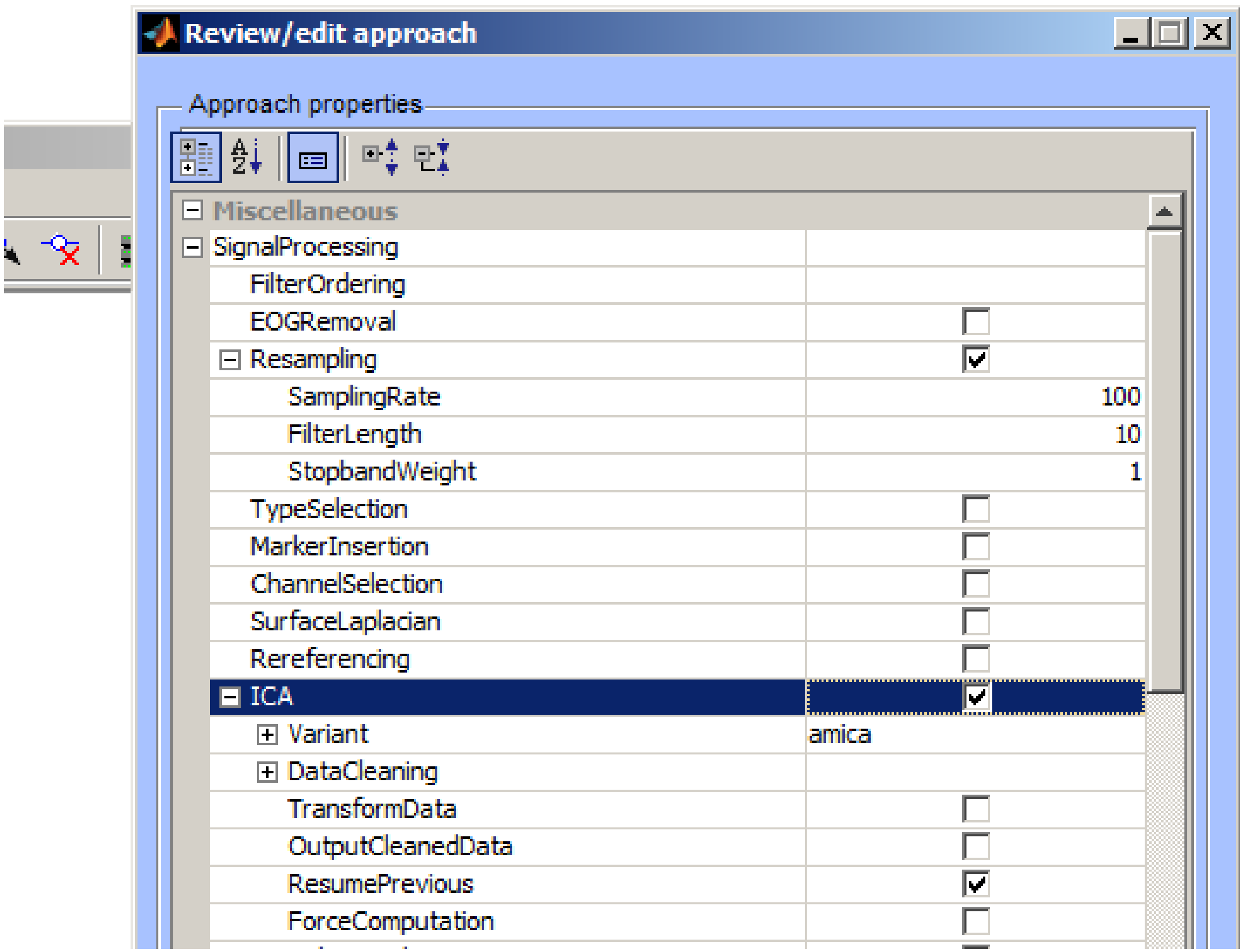


Uses the bandpower in several frequency bands per channel or component.

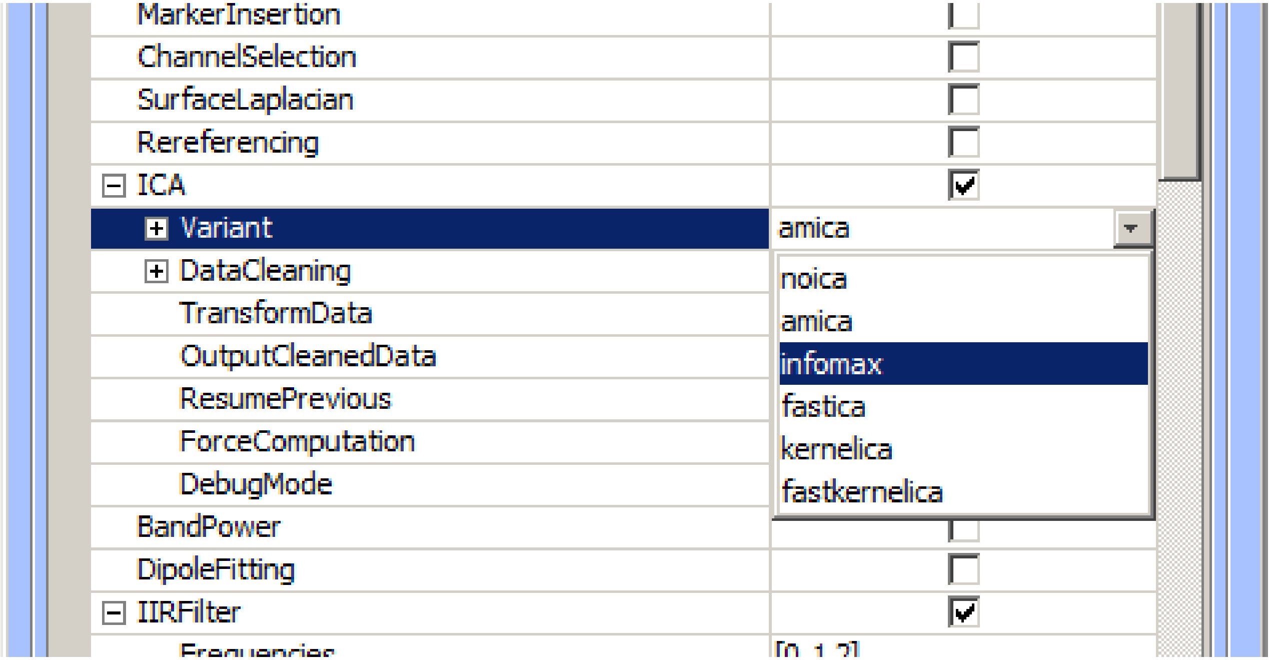
Custom ICA-based Approach



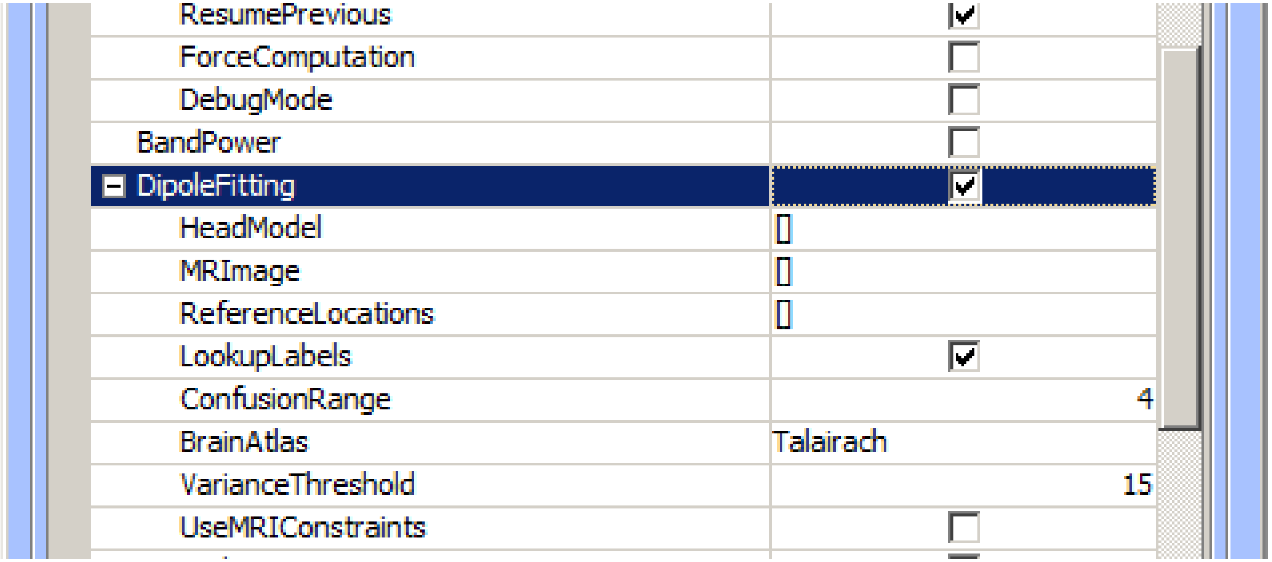
Enable ICA Filter Stage



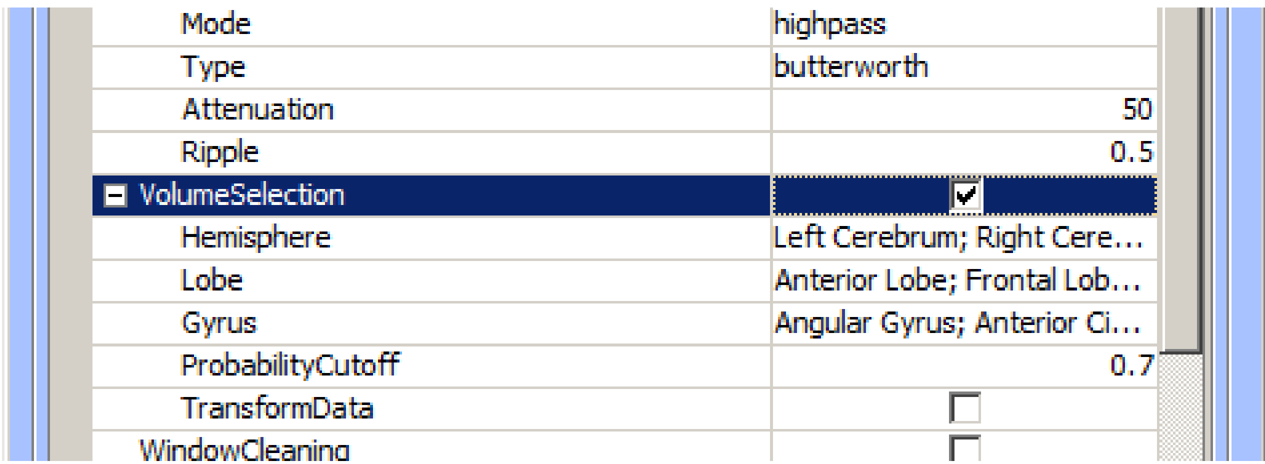
Select the Infomax Variant



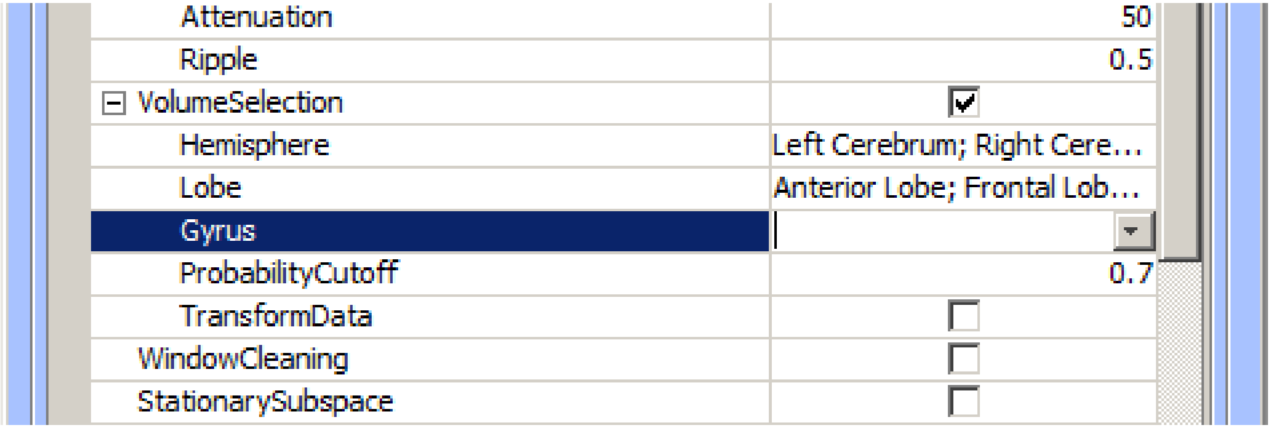
Enable Dipole Fitting



Enable Volume-Based Selection (Region of Interest)

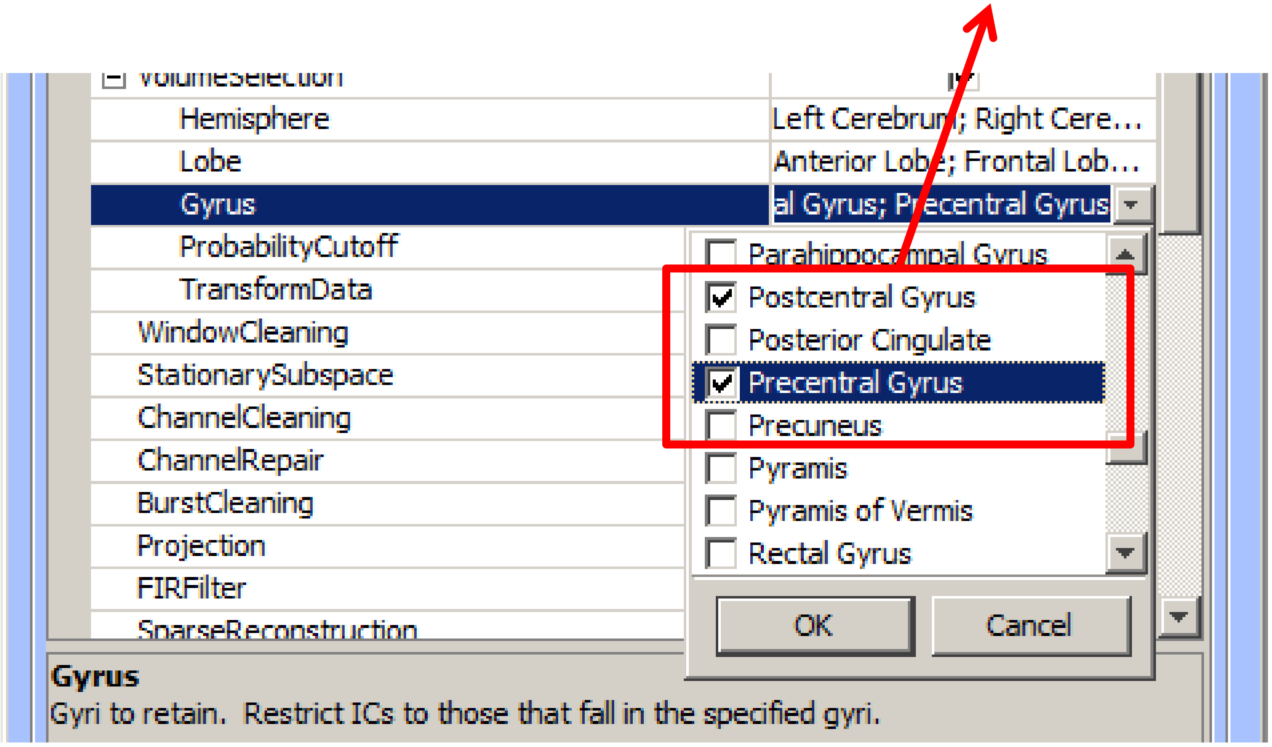


Clear All Selected Gyri

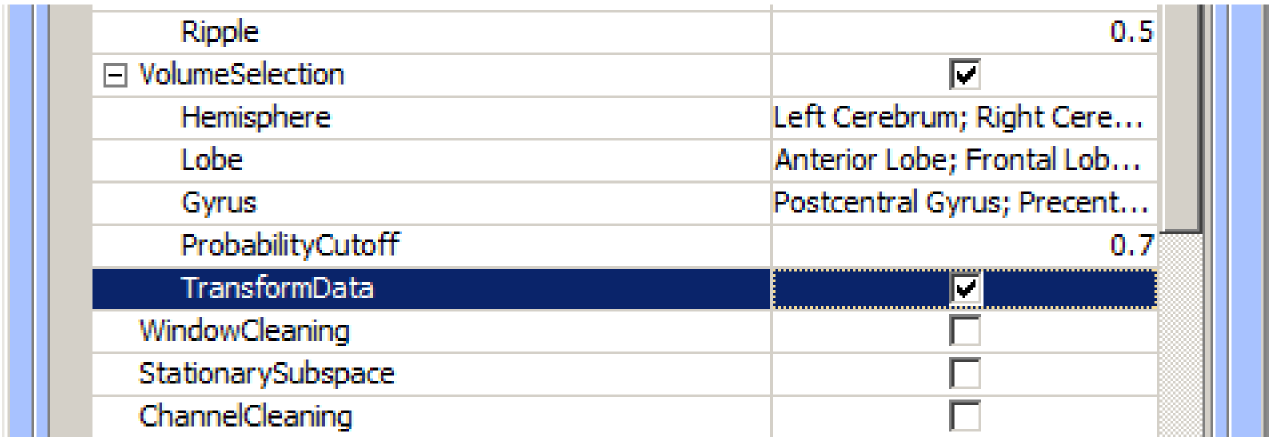


Select Only Motor Cortex

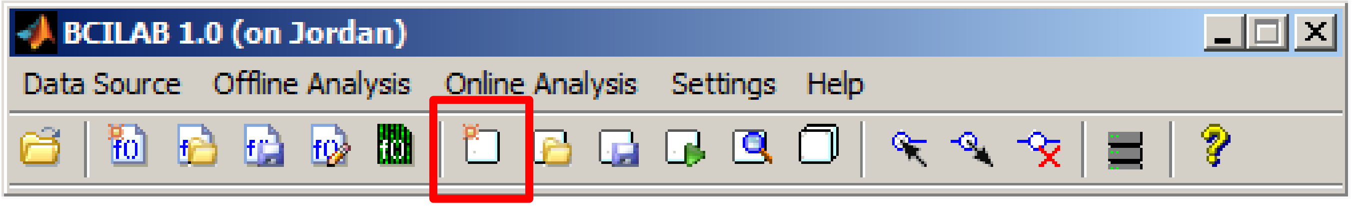
Just pick Precentral Gyrus and Postcentral Gyrus



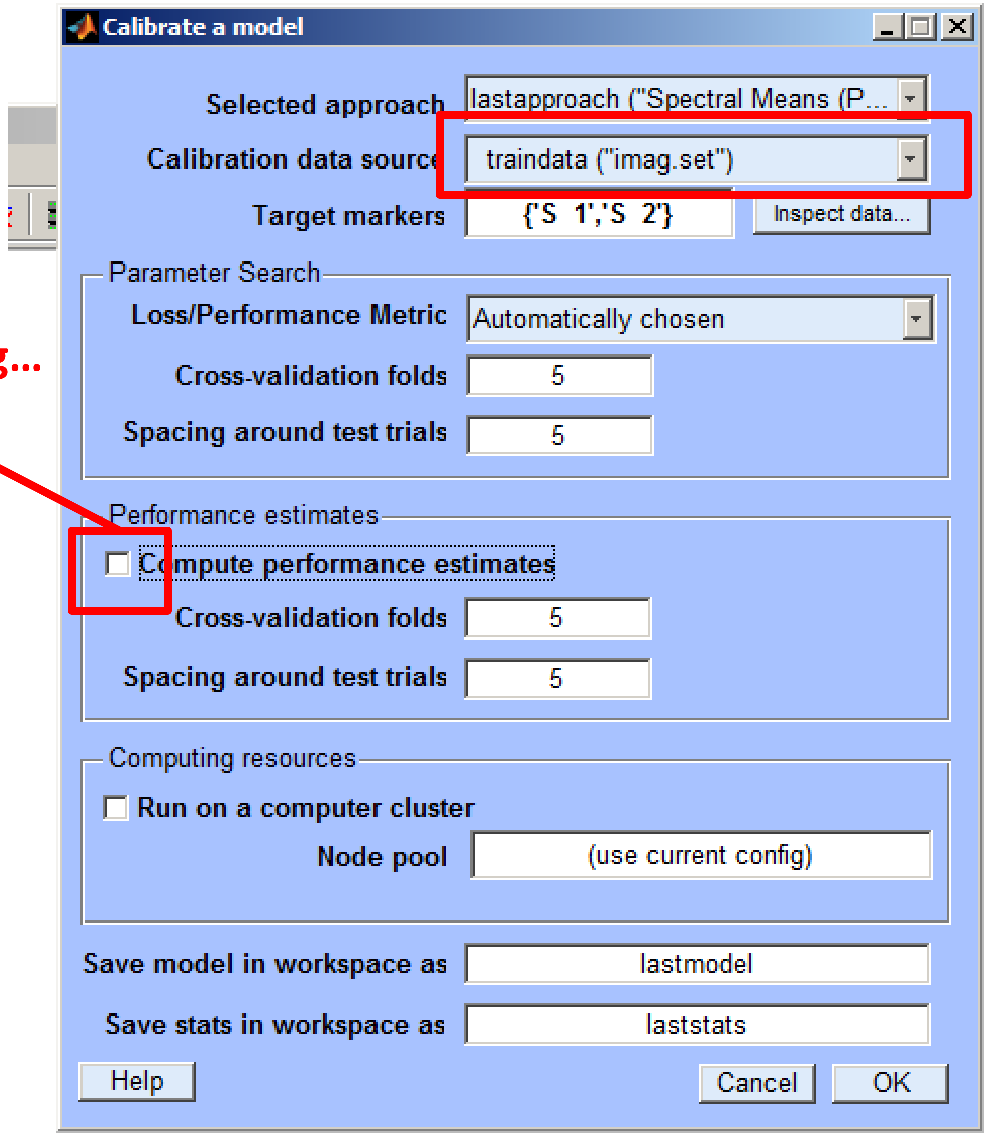
Transform the Data Into Components



Calibrate a New Model



Calibrate a New Model



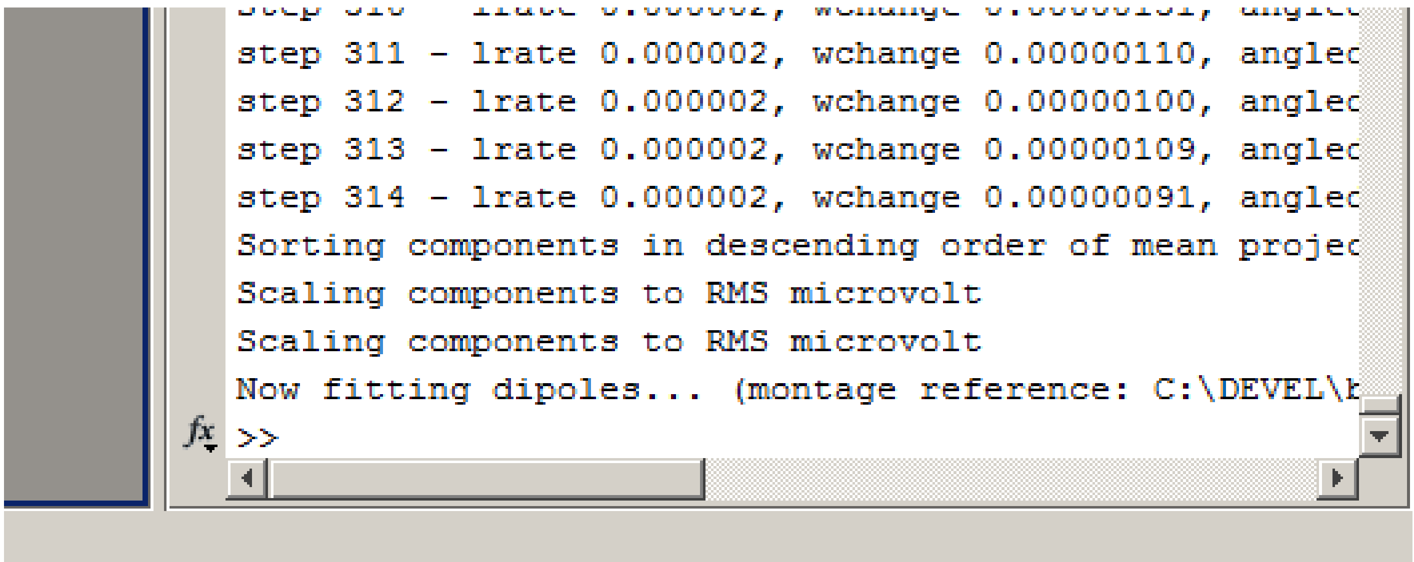
Cross-Validation would take too long...

Computation takes 5-10 Minutes…

```
  In @(hObject,eventdata)gui_calibratemodel('pushbutt
  In uiwait at 82
  In gui_calibratemodel>gui_calibratemodel_OpeningFcn
  In gui_mainfcn at 221
  In gui_calibratemodel at 30
step 1 - lrate 0.001000, wchange 14.48427508, anglede
step 2 - lrate 0.001000, wchange 0.28528851, angledel
step 3 - lrate 0.001000, wchange 0.16171884, angledel
step 4 - lrate 0.000980, wchange 0.15829343, angledel
step 5 - lrate 0.000960, wchange 0.15521940, angledel
step 6 - lrate 0.000941, wchange 0.13290725, angledel
step 7 - lrate 0.000922, wchange 0.12371815, angledel
step 8 - lrate 0.000904, wchange 0.12281045, angledel
step 9 - lrate 0.000886, wchange 0.12478248, angledel
step 10 - lrate 0.000868, wchange 0.13012623, anglede
step 11 - lrate 0.000851, wchange 0.11619709, anglede
step 12 - lrate 0.000834, wchange 0.10145424, anglede
step 13 - lrate 0.000817, wchange 0.13254306, anglede
step 14 - lrate 0.000801, wchange 0.10610921, anglede
step 15 - lrate 0.000785, wchange 0.09375523, anglede
step 16 - lrate 0.000769, wchange 0.09268561, anglede
step 17 - lrate 0.000754, wchange 0.08530438, anglede
step 18 - lrate 0.000739, wchange 0.09399871, anglede
step 19 - lrate 0.000724, wchange 0.09112500, anglede
step 20 - lrate 0.000709, wchange 0.08083924, anglede
step 21 - lrate 0.000695, wchange 0.08624642, anglede
step 22 - lrate 0.000681, wchange 0.07391327, anglede
step 23 - lrate 0.000668, wchange 0.07154284, anglede
step 24 - lrate 0.000654, wchange 0.07210989, anglede
step 25 - lrate 0.000641, wchange 0.06489432, anglede
step 26 - lrate 0.000628, wchange 0.06580106, anglede
>>
```

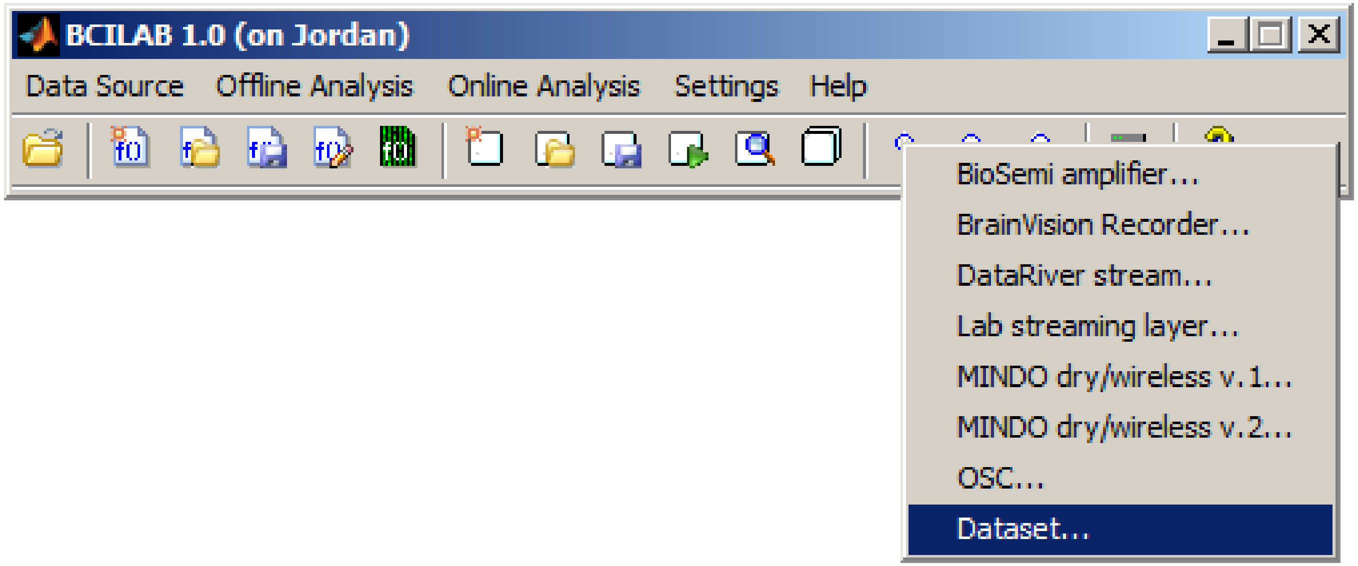
OVR

Computation takes 5-10 Minutes…



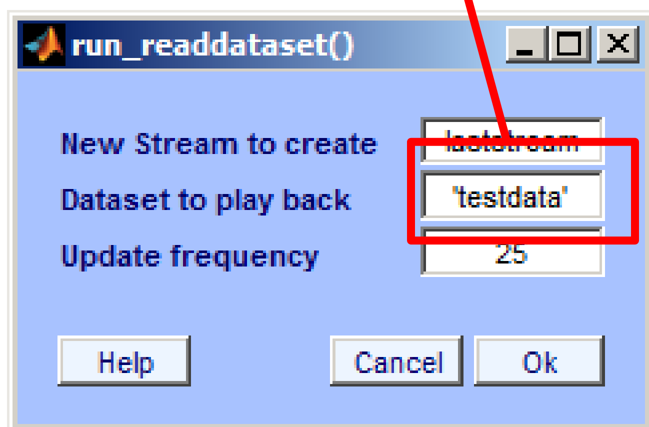
No results statistics dialog will pop up at the end (since statistics were turned off). So just wait until the Calibration Dialog closes itself…

Running it Online…

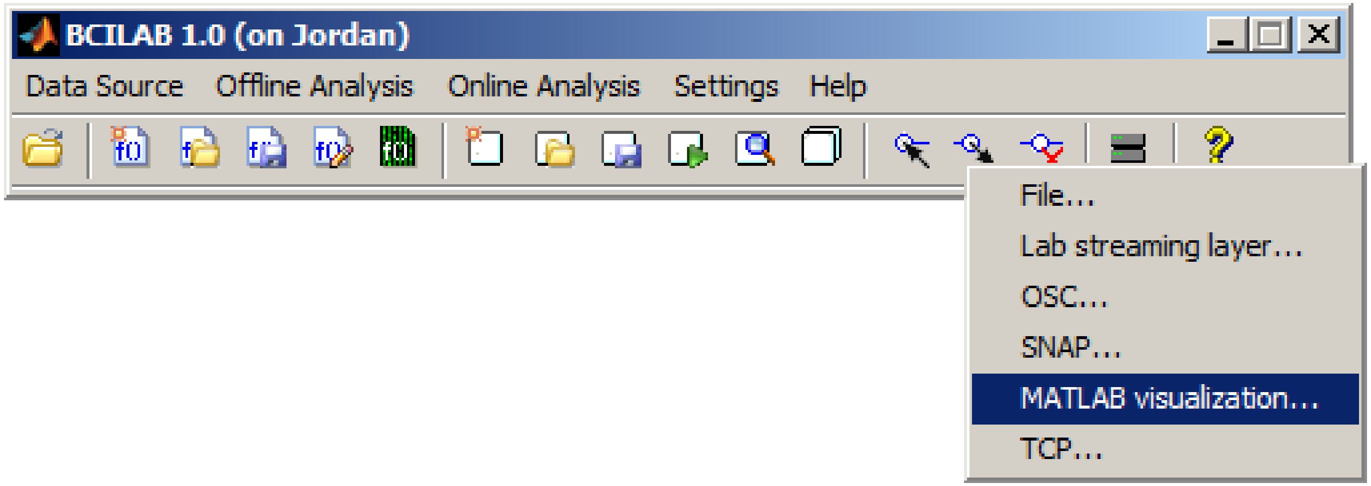


Starting Online Analysis

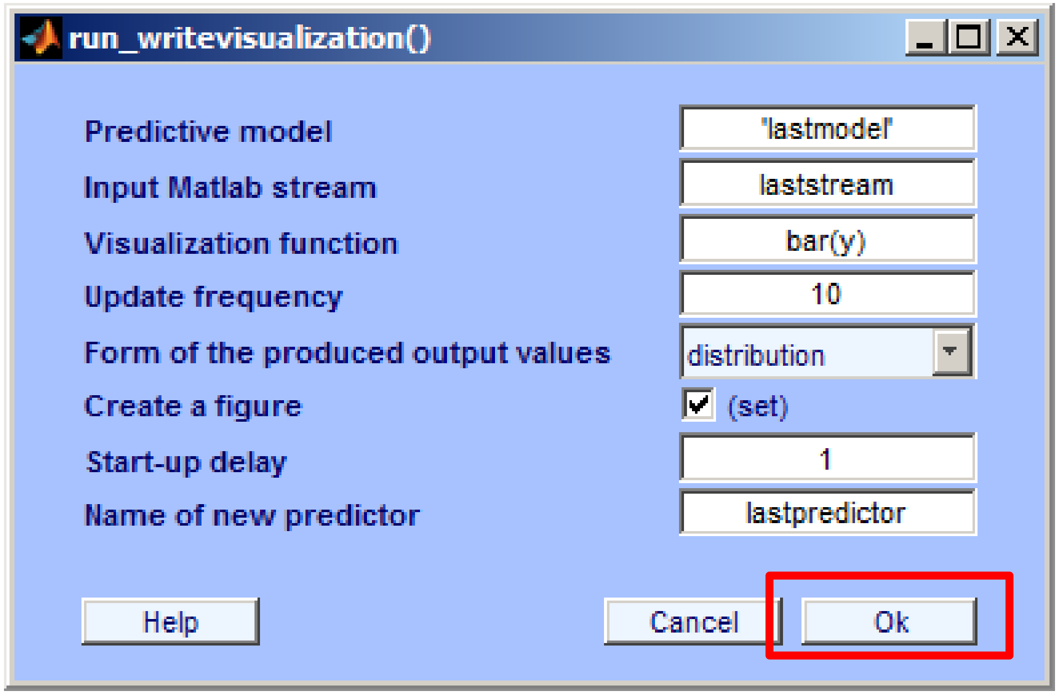
IMPORTANT: Put the name of your test data set here!



Select Output Destination

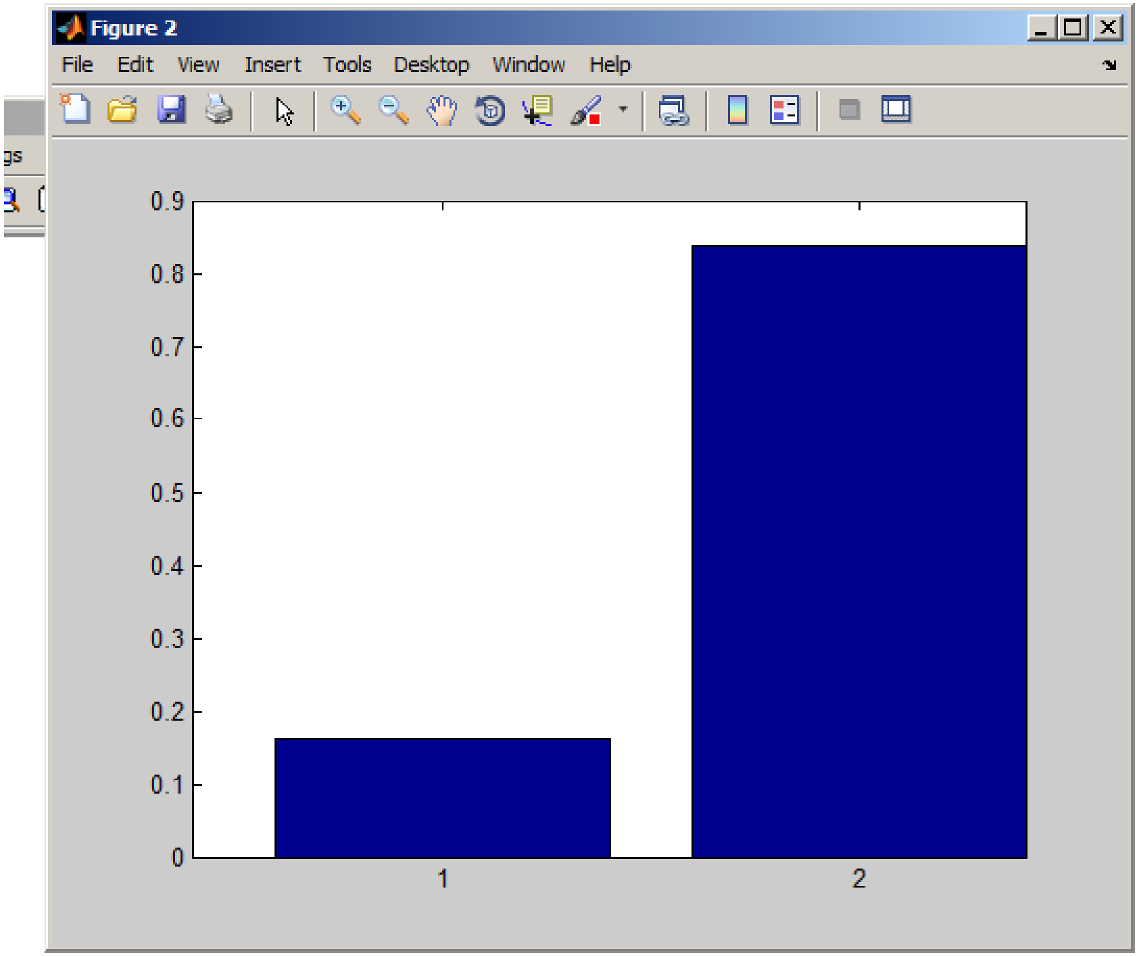


Configure/Confirm

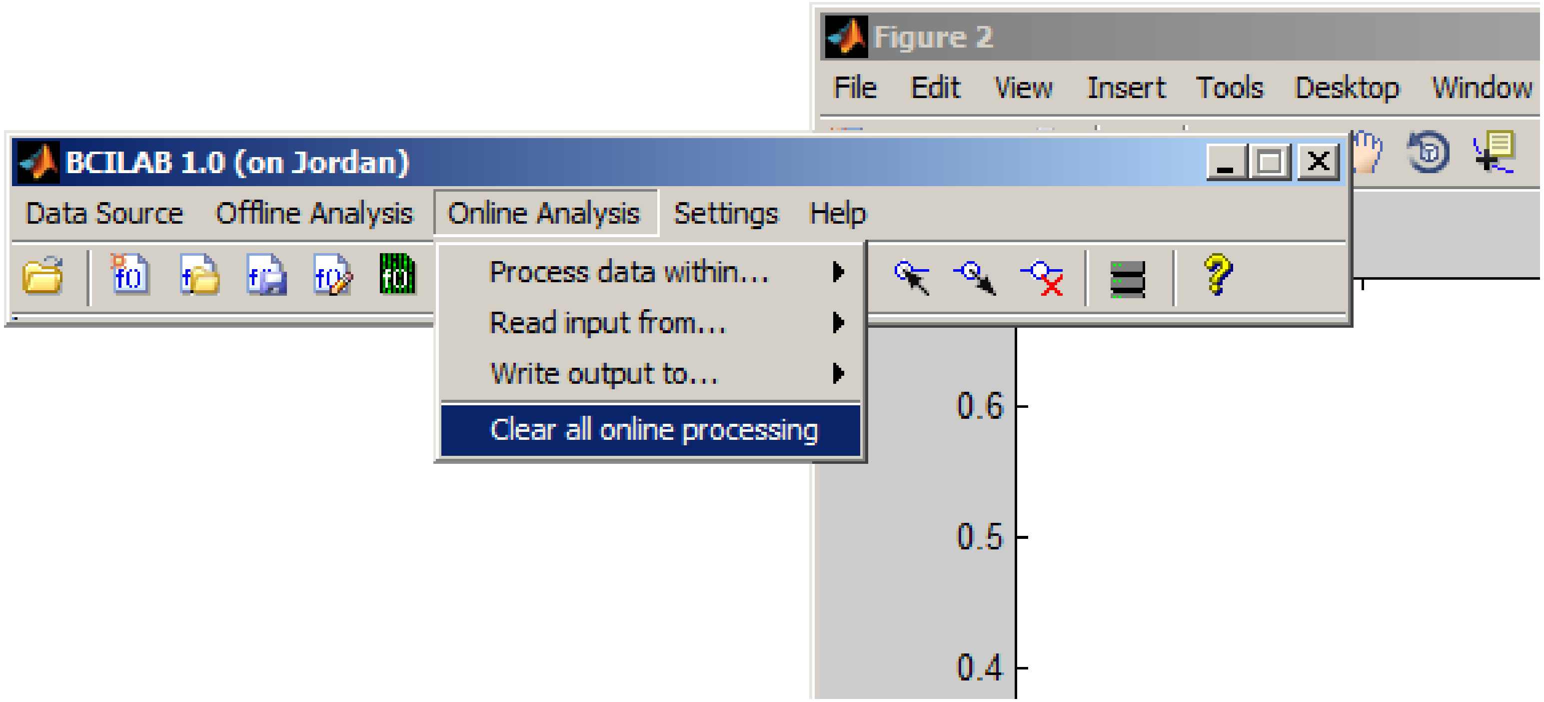


Online Output

You can clearly see the timing of the person's imagination period on these data.



Clear Online Processing



For More Information

- Theory
- Scripting
- Creating Plugins

See Additional Materials

Even More Information

- See the bcilab/userscripts folder for commented sample scripts that perform some of the discussed analyses, and many more (although on other data sets)
- See the bcilab/RELEASE_NOTES.txt file for how to find extra documentation and help

Questions?