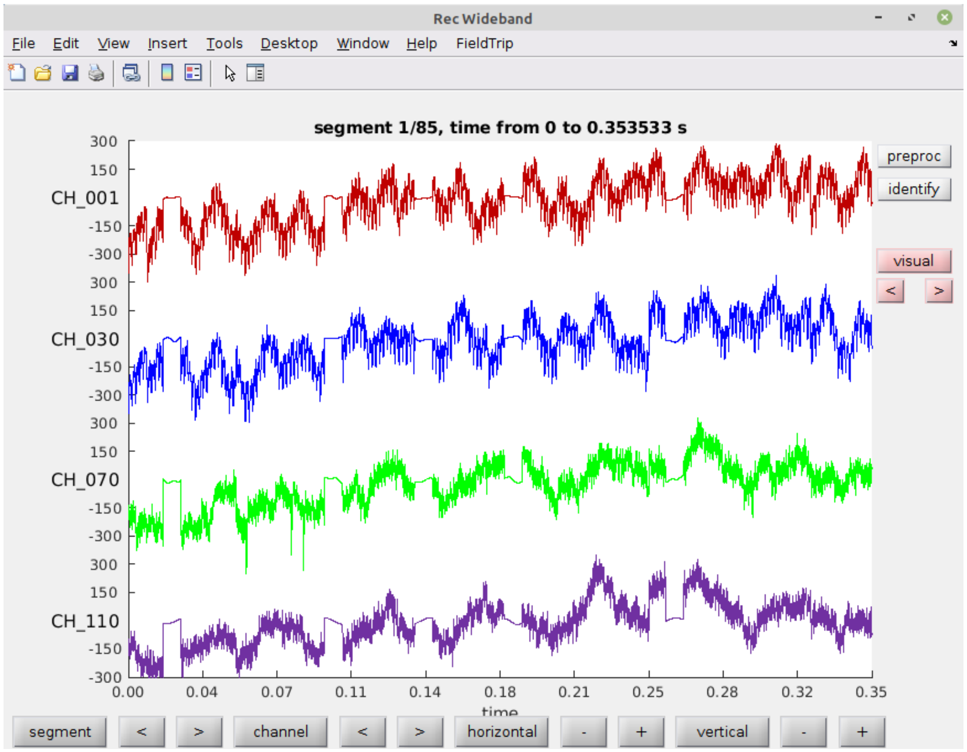
Dataset Sanity Check Script – Code Reference

Written by Christopher Thomas – March 4, 2022.



Contents

Chapter 1

Overview

1.1 README.md

```
# Chris’s Ephys Dataset Checking Script

## Overview

This tool walks through a set of folders looking for ephys data. If it finds
ephys data, it loads a small portion of it and tries to determine which
analog ephys channels contain valid data.

In addition to Field Trip, this uses the "‘exp-utils-cjt‘" library and the
"‘LoopUtil‘" library. This script is a wrapper for the
"‘euTools_sanityCheckTree()‘" function.

To run the tool in GUI mode (which prompts you for a folder to search), use
"‘make rungui‘", or type "‘do_sanity_gui‘" in Matlab.

## Getting Field Trip

To get Field Trip:
* Check that you have at least the following Matlab toolboxes:
    * Signal Processing Toolbox
    * Statistics Toolbox
* Go to [fieldtriptoolbox.org](https://www.fieldtriptoolbox.org).
* Click "latest release" in the sidebar on the top right
(or click [here](https://www.fieldtriptoolbox.org/#latest-release)).
* Look for "FieldTrip version (link) has been released". Follow that
GitHub link (example:
[Nov. 2021 link](http://github.com/fieldtrip/fieldtrip/releases/tag/20211118)).
* Unpack the archive somewhere appropriate, and add that directory to
Matlab’s search path.

## Other Libraries Needed
```

You’re also going to need the following libraries. Download the relevant
GitHub projects and make sure the appropriate folders from them are on path:

* [Open Ephys analysis tools](https://github.com/open-ephys/analysis-tools)
(Needed for reading Open Ephys files; the root folder needs to be on path.)
* [NumPy Matlab](https://github.com/kwikteam/npy-matlab)
(Needed for reading Open Ephys files; the "npy-matlab" subfolder needs to be
on path.)
* My [LoopUtil libraries](https://github.com/att-circ-contrl/LoopUtil)
(Needed for reading Intan files and for integrating with Field Trip; the
"libraries" subfolder needs to be on path.)
* My [experiment utility libraries](https://github.com/att-circ-contrl/exp-utils-cjt)
(Needed for processing steps that are specific to our lab, and more Field
Trip integration; the "libraries" subfolder needs to be on path.)

## Project Files and Folders

These are where the script looks for input and saves output:
* ‘datasets‘ is where the non-GUI script looks for data by default.
* ‘output‘ is where the script saves its analysis. "‘sanityreport.txt‘"
contains a human-readable report (also echoed to the console unless config
was set to suppress it), and "‘sanitydata.mat‘" contains the report and the
structure array returned by "‘euTools_sanityCheckTree()‘".

These are the tool scripts themselves; all of them together would fit on a
single page (they really are mostly wrappers).
* ‘do_paths.m‘ initializes paths using my test environment symlinks.
* ‘do_quiet.m‘ suppresses most output from Field Trip and other libraries.
* ‘do_sanity.m‘ is the non-GUI entrypoint.
* ‘do_sanity_gui.m‘ is the GUI-mode entrypoint.

These are the library symlinks that "‘do_paths.m‘" looks for:
* ‘lib-exp-utils-cjt‘ points to exp-utils-cjt’s "‘libraries‘" folder.
* ‘lib-fieldtrip‘ points to Field Trip’s top-level folder.
* ‘lib-looputil‘ points to LoopUtil’s "‘libraries‘" folder.
* ‘lib-npy-matlab‘ points to NumPy Matlab’s "‘npy-matlab‘" folder.
* ‘lib-openephys‘ points to Open Ephys’s "‘analysis-tools‘" folder.

This is the end of the file.

Chapter 2

Top-Level Scripts

2.1 do_paths.m

```
% This adds the various ACC Lab and external project paths.

addpath(’lib-exp-utils-cjt’);
addpath(’lib-looputil’);
addpath(’lib-fieldtrip’);
addpath(’lib-openephys’);
addpath(’lib-npy-matlab’);

addPathsExpUtilsCjt;
addPathsLoopUtil;

% Wrap this in "evalc" to avoid the annoying banner.
evalc(’ft_defaults’);

% This is the end of the file.
```

2.2 do_quiet.m

```
% This turns off messages from FT and from the NPY library.

% Set FT defaults via "evalc" again, to suppress the banner again.
evalc(’ft_defaults’);

ft_notice(’off’);
ft_info(’off’);

% We’ll sometimes get lots of warnings about deprecated config fields.
ft_warning(’off’);
```

```
% NPy loves to complain about text data.
% Use "warning(warnstate)" to restore warnings.
warnstate = warning('off');

% This is the end of the file.
```

2.3 do_sanity.m

```
% Quick and dirty test script for dataset sanity-checking.
% Written by Christopher Thomas.

do_paths;
do_quiet;

if ~exist('sourcedir', 'var')
  % NOTE - Pick the subfolder for a rapid test.
  %sourcedir = 'datasets';
  sourcedir = 'datasets/*tungsten';
end

% Set up configuration to look at the early part of the data.
% This avoids stimulation artifacts.
config = struct( 'readposition', 0.05 );

[ reporttext folderdata ] = euTools_sanityCheckTree( sourcedir, config );

save( 'output/sanitydata.mat', 'reporttext', 'folderdata', '-v7.3' );

thisfid = fopen('output/sanityreport.txt', 'w');
fwrite(thisfid, reporttext);
fclose(thisfid);

%
% This is the end of the file.
```

2.4 do_sanity_gui.m

```
% Quick and dirty test script for dataset sanity-checking - GUI wrapper.
% Written by Christopher Thomas.

sourcedir = uigetdir;

do_sanity;

%
% This is the end of the file.
```