SAFT - Semi-automatic fluorescence trace analysis

The purpose of SAFT is to automate some steps in analysis of recordings of synaptic activity, namely fluorescence time series corresponding to the emission of a neurotransmitter indicator. The steps that can be automated are background subtraction and peak finding. The user can tune both of these algorithms, and check the consequences interactively. The main workflow is directed towards recordings that come from different ‘regions of interest’ (‘ROIs’) over the same neuron that is stimulated in a regular fashion. The stimulus protocol is not known, but is inferred from the statistics of the mean responses. This makes analysis possible in a coordinated and efficient way over multiple conditions using the same parameters.

Installing

SAFT is written in Python, but basic Python is not enough. SAFT has quite a few dependencies, meaning external Python libraries that must be downloaded. These are for calculations (`numpy` and `scipy`), data management (`pandas`) and the GUI (`PySide2` and `pyqtgraph`). You need the versions below at time of writing. Unfortunately, some unclear incompatibilities mean that some more modern versions of the relevant packages do not work.

Getting all these packages aligned is not easy. You can use anaconda, pip or compile from source. It is best to start with a fresh environment.

Conda currently provides Qt 5.9.7 and this makes it convenient to use conda to set up the package environment. To make getting the right environment in place easy, we include a conda environment .yml file in the distribution (SAFTenv.yml). To use this, you must have Anaconda or Miniconda. If you don't have Anaconda already, just get [Miniconda](# https://docs.conda.io/projects/continuumio-conda/en/latest/user-guide/install/macos.html).

To install the packages needed for SAFT into a fresh environment with Miniconda, go to the SAFT directory and enter the following in the Terminal,

```
    conda env create --file SAFTenv.yml
```

This will create an environment called 'SAFT' with a working set of packages.
As conda will tell you, to switch to this environment, enter the following,

```
    conda activate SAFT
```

then navigate to the SAFT directory and execute the commands as above to launch SAFT.

If you happen to already have an environment named SAFT, you would need to remove it first:

```
    conda env remove --name SAFT
```

Very important : Qt versions after 5.9 are not open source (breaking the conditions of our license), and coincidentally have serious bugs on macOS. We need Qt for the high-performance scientific plotting library pyqtgraph. PySide2 versions up to 5.15 seem to run happily with Qt 5.9.7 but it may be difficult to avoid a concurrent Qt update, unless you have a separate version of Qt compiled from open source and know how to point PySide2 at it. The latter solution is not stable.

Dependencies current as of 2020-11-12

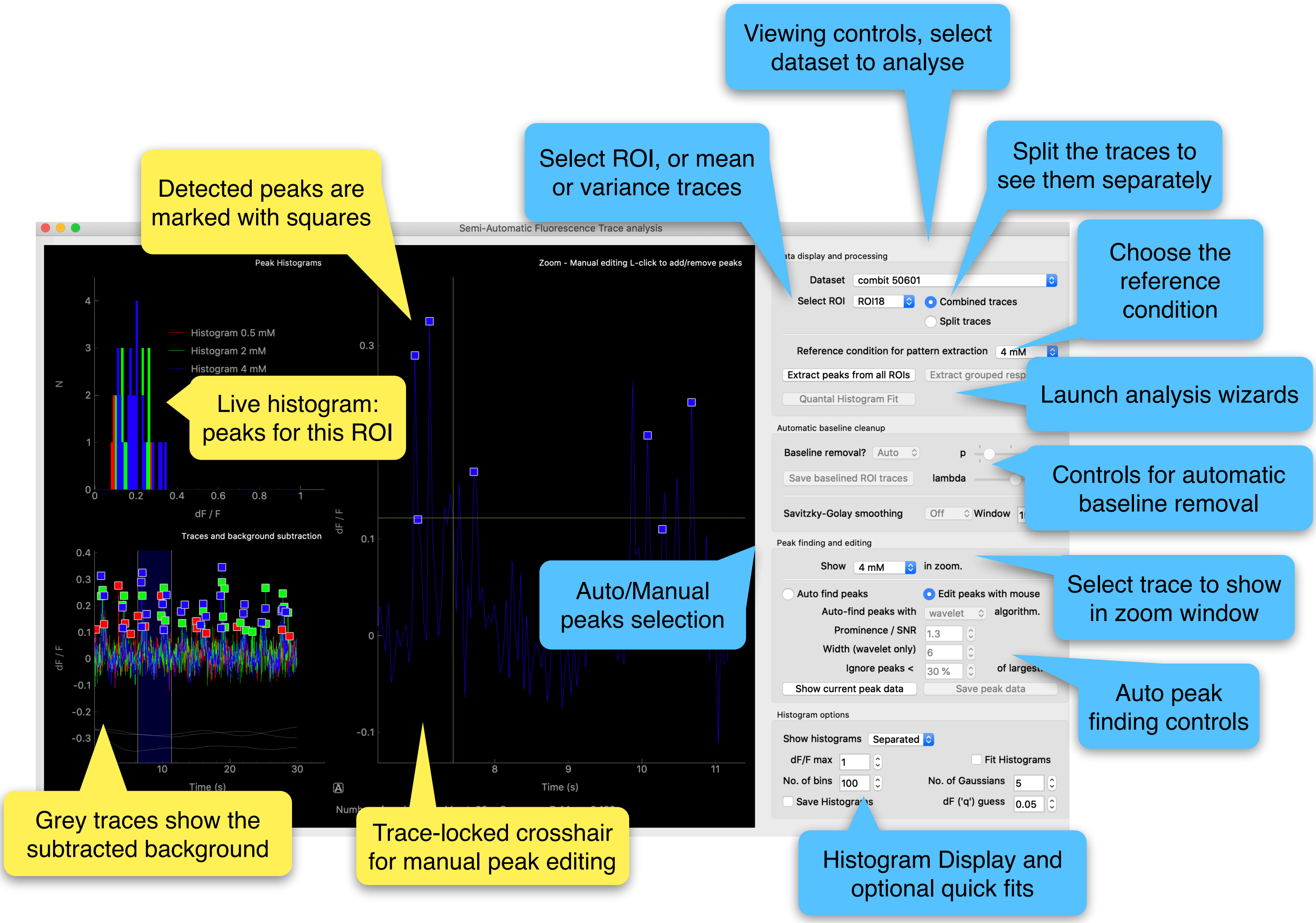
```
Python 3.7.9
qt==5.9.7
PySide2>=5.9.0a1
numpy==1.19.2 scipy==1.5.2
pyqtgraph==0.11.0
pandas==0.24.0
matplotlib==3.3.2
xlrd==1.2.0
```

```
openpyxl==3.0.5
pyobjc-framework-Cocoa==6.2.2
python.app 2
```

To get SAFT as intended, launch it with the command from the directory where you unpacked it:

```
pythonw SAFT.py
```

After launching SAFT, you are confronted with the main window, ‘Semi-automatic Fluorescence Trace analysis’. A few commands are in the menu bar, but the advanced controls are presented directly in the interface.


Semi-Automatic Fluorescence Trace analysis
Peak Histograms
Histogram 0.5 mM
Histogram 2 mM
Histogram 4 mM
N
dF / F
Traces and background subtraction
Time (s)
Zoom - Manual editing L-click to add/remove peaks
Time (s)
dF / F

ata display and processing

Dataset combit 50601

Select ROI ROI18 ● Combined traces ○ Split traces

Reference condition for pattern extraction 4 mM

Extract peaks from all ROIs | Extract grouped resp

Quantal Histogram Fit

Automatic baseline cleanup

Baseline removal? Auto p

Save baselined ROI traces lambda

Savitzky-Golay smoothing Off Window 1

Peak finding and editing

Show 4 mM in zoom.

○ Auto find peaks ● Edit peaks with mouse

Auto-find peaks with wavelet algorithm.

Prominence / SNR 1.3

Width (wavelet only) 6

Ignore peaks < 30 % of larges

Show current peak data | Save peak data

Histogram options

Show histograms Separated

dF/F max 1 □ Fit Histograms

No. of bins 100 No. of Gaussians 5

□ Save Histograms dF ('q') guess 0.05

Viewing controls, select dataset to analyse

Select ROI, or mean or variance traces

Split the traces to see them separately

Detected peaks are marked with squares

Choose the reference condition

Launch analysis wizards

Live histogram: peaks for this ROI

Controls for automatic baseline removal

Auto/Manual peaks selection

Select trace to show in zoom window

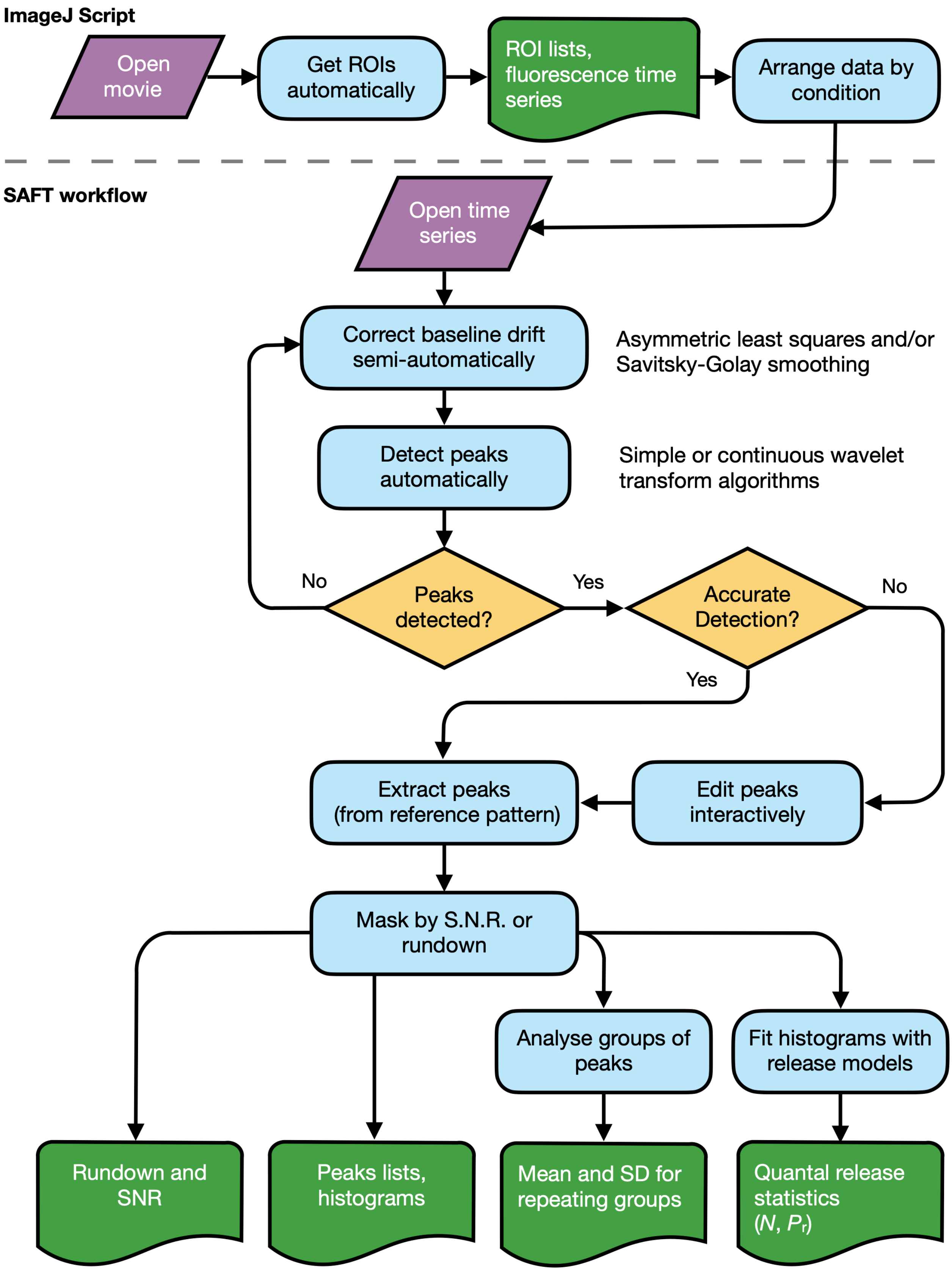
Auto peak finding controls

Grey traces show the subtracted background

Trace-locked crosshair for manual peak editing

Histogram Display and optional quick fits

SAFT main window with data file loaded.


ImageJ Script
Open movie
Get ROIs automatically
ROI lists, fluorescence time series
Arrange data by condition
SAFT workflow
Open time series
Correct baseline drift semi-automatically
Asymmetric least squares and/or Savitsky-Golay smoothing
Detect peaks automatically
Simple or continuous wavelet transform algorithms
No
Peaks detected?
Yes
Accurate Detection?
No
Yes
Extract peaks (from reference pattern)
Edit peaks interactively
Mask by S.N.R. or rundown
Analyse groups of peaks
Fit histograms with release models
Rundown and SNR
Peaks lists, histograms
Mean and SD for repeating groups
Quantal release statistics (N, P_r)

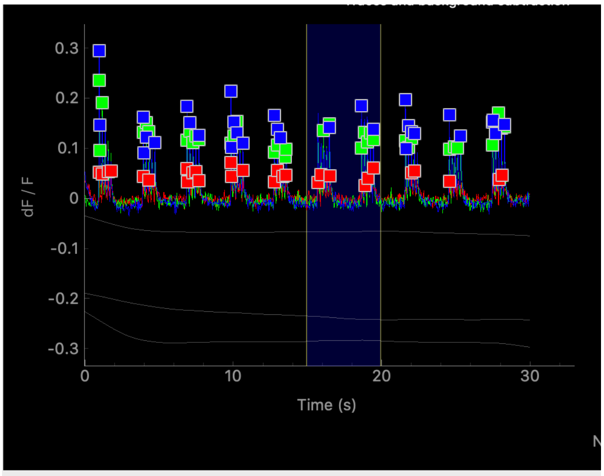
Here is the workflow of SAFT. The program expects a set of time series as input.

SAFT has four windows for analysing the peak responses.

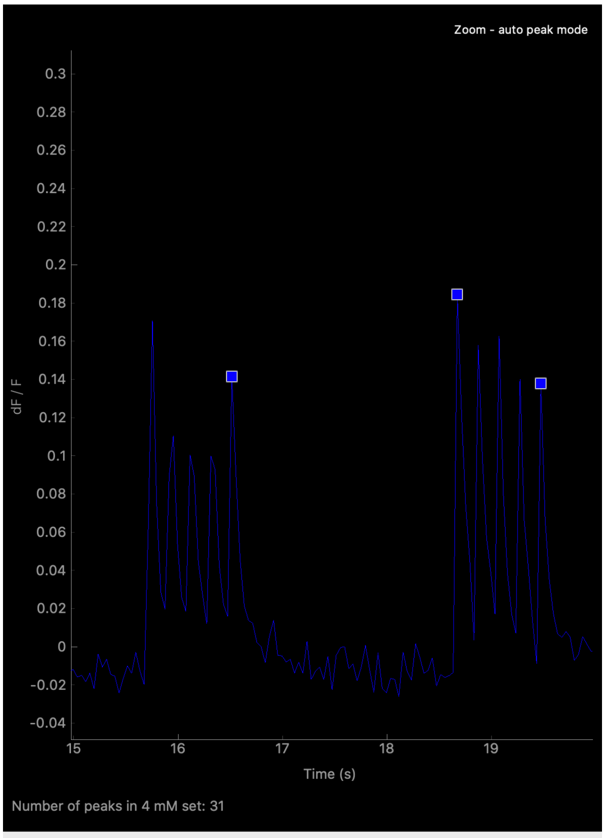
1. Main window (`SAFT.py`) for loading trace data and identifying the position of peaks, removing baseline. This window also shows histograms of the peak data.
2. Peak extraction dialog (`extractPeakResponses.py`, opened by ‘*Extract peaks from all ROIs*’ button in the main window) for extracting peak data from known timepoints across a large number of ROI and sets. Automatic baseline correction is done. On completion and return to the main window, the peaks can be seen (as well as failures), edited and compared with the baseline subtracted traces.
3. Group peaks dialog (`processGroupedPeaks.py` called by the “*Extract group responses*” button in the main window) where you can get statistics for repetitive responses, for example, if you repetitively stimulate with trains, you can get the average and SD of the 2nd, 3rd, nth response in the train. Data, including automatically generated graphs, can be saved here. This kind of analysis is quite easy to do in Excel for example, but takes a lot of cutting and pasting, which is time consuming and leads to errors.
4. Histogram fitting (`histogramFitDialog.py` which is opened by the ‘*Launch Histogram Fit*’ button) in which you can fit histograms with binomial and poisson models of vesicle release, if they represent quantal responses. Global fitting of models across conditions can also be done. No allowance is made for reporter characteristics, such as non-linearity.

To get started, load the file SAFT_Data.xlsx using the `File > Open File` command. These traces were automatically extracted from movies in imageJ using the SNFR_ROI_Finder.ijm script (written by Benjamin König). This script is also in the repository.

Traces load and are automatically colour-coded. Three panels show the entire traces for each condition with the baseline correction plotted separately (‘Traces and background subtraction’, bottom left), a zoom into a chosen trace (‘zoom’, the scope is indicated by the blue translucent region in the bottom left panel) and histograms of automatically found peak responses (upper right).



On the left hand side of the main window, a small graph shows all the traces overlaid and the automatically subtracted baselines. You can see here how “wobbly” the subtraction is. If you don’t want to subtract the steady-state response of the indicator, it should be smooth.

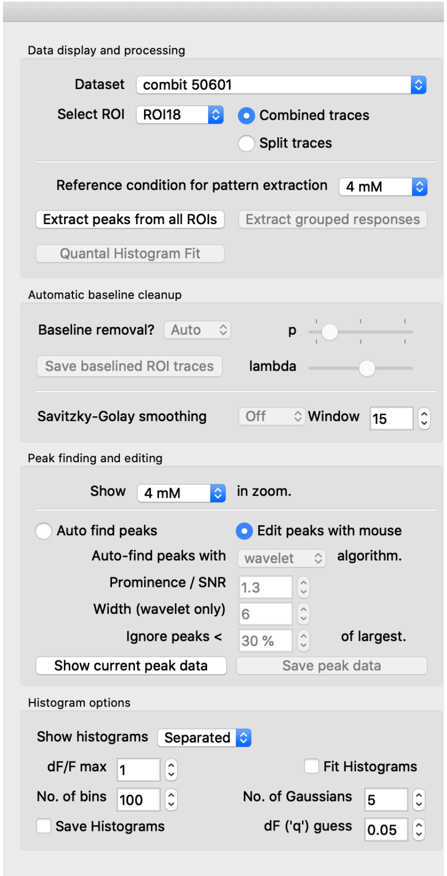


Number of peaks in 4 mM set: 31

In the centre, the zoom graph displays the trace (select which one in the “Peak finding and editing panel”). You can adjust the peak finding controls to collect the most peaks automatically - these are overlaid as square points on the trace, then switch to manual mode. A left click will add a peak, or remove it if you click on an existing one. You can look at any ROI but SAFT will use the “Mean” ROI for a reference (you can choose which condition in “Data display and processing”) when extracting peaks from all records.

When you are happy that all the peak locations are accurately specified, you can launch “Extract peaks from all ROIs”.

On the right side of the main window, a set of panels gives you control over analysis and data display. These controls are divided into four sections:



Data display and processing

Dataset: combit 50601

Select ROI: ROI18 ● Combined traces ○ Split traces

Reference condition for pattern extraction: 4 mM

Extract peaks from all ROIs
Extract grouped responses
Quantal Histogram Fit

Automatic baseline cleanup

Baseline removal?: Auto

p
Save baselined ROI traces
lambda

Savitzky-Golay smoothing: Off Window: 15

Peak finding and editing

Show 4 mM in zoom.

○ Auto find peaks ● Edit peaks with mouse

Auto-find peaks with wavelet algorithm.

Prominence / SNR: 1.3

Width (wavelet only): 6

Ignore peaks < 30 % of largest.

Histogram options

Show histograms: Separated

dF/F max: 1 □ Fit Histograms

No. of bins: 100 No. of Gaussians: 5

□ Save Histograms dF ('q') guess: 0.05

SAFT main window control panel

'Data display and processing'
The dataset takes the name of the file, and you can select which ROI to view. Two automatically generated virtual ROIs, the mean and the variance are also viewable.

Here, a panel of some buttons give easy access to downstream analysis windows, saving data and you can choose which trace from the virtual ROI 'Mean' you want to use as the reference

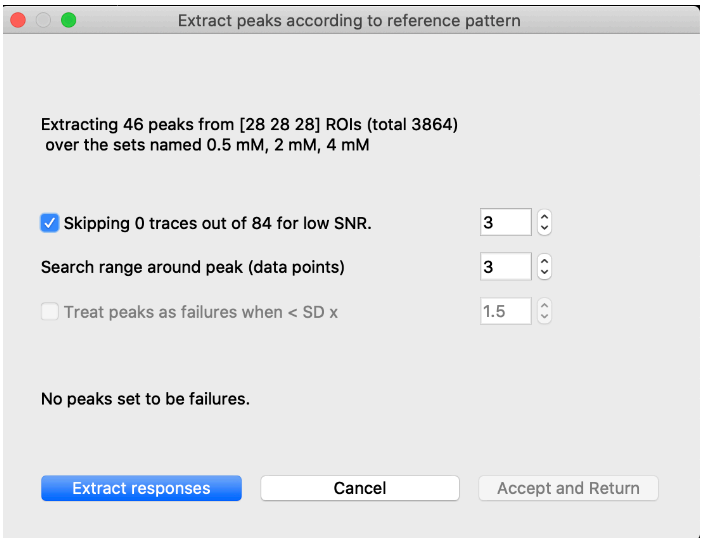
'Automatic baseline cleanup'
Baseline removal can be switched on ('Auto'), off ('None') or locked with fixed parameters ('Lock'). Datasets generated following analysis will automatically turn off baseline removal. The baseline removal can be tuned with the p and lambda (length) scale controls. The lower the value of lambda, the shorter the region of a steady state response that will be removed. The baseline that is subtracted is drawn in the bottom left graph panel.
Savitsky-Golay smoothing is also implemented but did not yet work well in out hands.

'Peak finding and editing'
Select which condition to show in the zoom graph panel (here it's the '4 mM' condition). You can switch between auto peak finding, and manual editing of peaks with the mouse. If you switch to manual editing, a crosshair appears and you can click to mark peaks (squares). Click any square to remove it.
If *Auto find peaks* is enabled, you can choose between a simple threshold method and the wavelet algorithms.
Parameters like *Prominence* (for the simple algorithm) or *SNR* and *width* for the wavelet algorithm can be adjusted. You can optionally ignore any small peaks too.
There is a button ('*Show current peak data*') to give a quick popup of the underlying values.

'Histogram options'
There is an option to sum the histograms, which makes sense if the quantal size is consistent across conditions. The scale of the histogram display can be controlled here. There is also the option to fit the histograms with simple functions.

Extract Peaks according to a reference pattern

The number of peaks in each trace, the number of traces (ROIs) and the theoretical maximum total of peaks to find over the entire set of conditions is shown. You can choose to skip individual traces on the basis of the signal to noise. You can adjust the range in each trace to search for a maximum around the peak location.



Extracting 46 peaks from [28 28 28] ROIs (total 3864)
over the sets named 0.5 mM, 2 mM, 4 mM

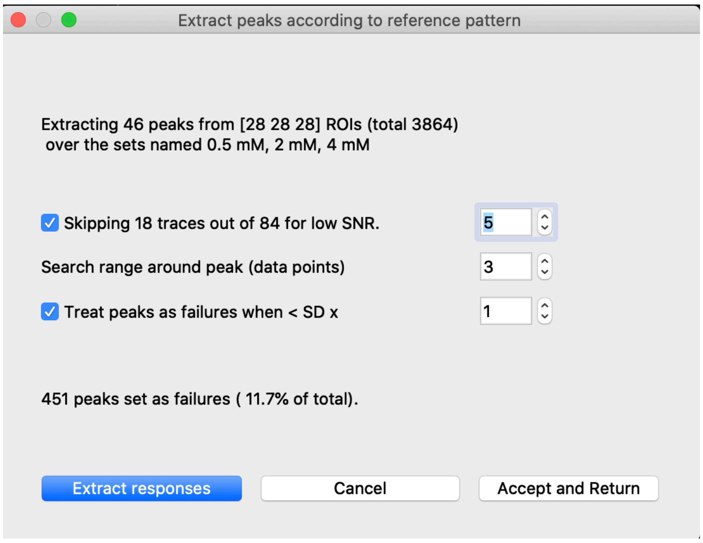
☑ Skipping 0 traces out of 84 for low SNR. 3

Search range around peak (data points) 3

☐ Treat peaks as failures when < SD x 1.5

No peaks set to be failures.

Once you have clicked “Extract responses”, you have the option to treat detected peaks as “failures”, that is, set their amplitude to zero, when they are less than the SD x factor. The lower this value, the more peaks you will discard. The dialog tells you the number that were set to failures and the fraction.



Extracting 46 peaks from [28 28 28] ROIs (total 3864)
over the sets named 0.5 mM, 2 mM, 4 mM

☑ Skipping 18 traces out of 84 for low SNR. 5

Search range around peak (data points) 3

☑ Treat peaks as failures when < SD x 1

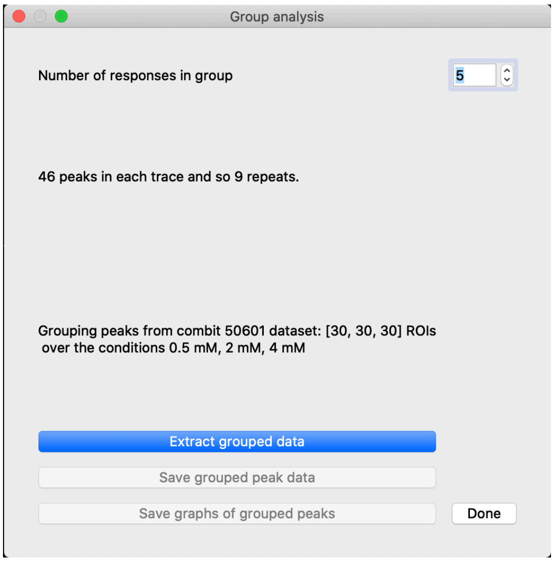
451 peaks set as failures (11.7% of total).

When you click “Accept and Return”, you go back to the main window and a new dataset corresponding to the detected peaks (with the baseline-subtracted traces) is created. Some controls that don’t make sense any more are disabled for this dataset. You can use these data for analysing the responses according to repetitive groups in *“Extract grouped responses”* and the statistics of the peak amplitudes in “*Histogram Fitting with Quantal parameters”*

Extract grouped responses

If each trace corresponds to repetitive grouped responses, for example trains of 5 action potentials, you can analyse these groups and get the mean of the first response in the group, second response in the group etc as well as their standard deviation. This can be useful for comparing to synaptic transmission measured with electrophysiology.

You select how many responses are in each group and the dialog will then indicate how many repeats are found. You can only save these calculated statistics directly from this dialog.



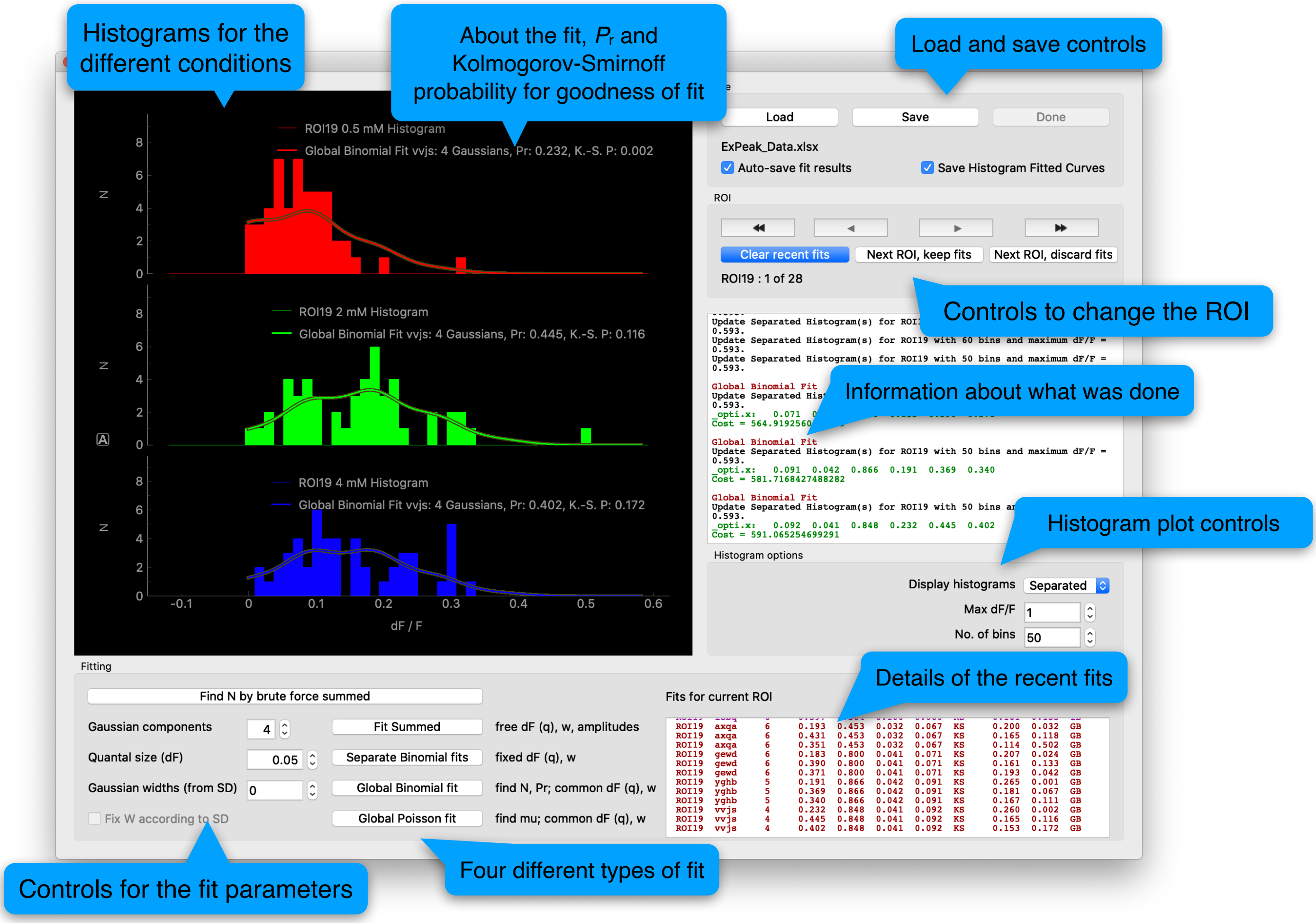
Number of responses in group 5

46 peaks in each trace and so 9 repeats.

Grouping peaks from combit 50601 dataset: [30, 30, 30] ROIs
over the conditions 0.5 mM, 2 mM, 4 mM

Histogram Fitting with Quantal parameters

One goal of optical reporting is to gain access to quantal parameters from a large number of synapses at once. To do this, the peak amplitude distributions can be fitted with different models of vesicle release.



Histograms for the different conditions

About the fit, P_r and Kolmogorov-Smirnoff probability for goodness of fit

Load and save controls

Controls to change the ROI

Information about what was done

Histogram plot controls

Details of the recent fits

Controls for the fit parameters

Four different types of fit

```
Update Separated Histogram(s) for ROI1
0.593.
Update Separated Histogram(s) for ROI19 with 60 bins and maximum dF/F =
0.593.
Update Separated Histogram(s) for ROI19 with 50 bins and maximum dF/F =
0.593.

Global Binomial Fit
Update Separated His
0.593.
_opti.x:   0.071  0
Cost = 564.9192560

Global Binomial Fit
Update Separated Histogram(s) for ROI19 with 50 bins and maximum dF/F =
0.593.
_opti.x:   0.091  0.042  0.866  0.191  0.369  0.340
Cost = 581.7168427488282

Global Binomial Fit
Update Separated Histogram(s) for ROI19 with 50 bins ar
0.593.
_opti.x:   0.092  0.041  0.848  0.232  0.445  0.402
Cost = 591.065254699291
```

```
ROI19  axqa   6   0.193  0.453  0.032  0.067  KS   0.200  0.032  GB
ROI19  axqa   6   0.431  0.453  0.032  0.067  KS   0.165  0.118  GB
ROI19  axqa   6   0.351  0.453  0.032  0.067  KS   0.114  0.502  GB
ROI19  gewd   6   0.183  0.800  0.041  0.071  KS   0.207  0.024  GB
ROI19  gewd   6   0.390  0.800  0.041  0.071  KS   0.161  0.133  GB
ROI19  gewd   6   0.371  0.800  0.041  0.071  KS   0.193  0.042  GB
ROI19  yghb   5   0.191  0.866  0.042  0.091  KS   0.265  0.001  GB
ROI19  yghb   5   0.369  0.866  0.042  0.091  KS   0.181  0.067  GB
ROI19  yghb   5   0.340  0.866  0.042  0.091  KS   0.167  0.111  GB
ROI19  vvjs   4   0.232  0.848  0.041  0.092  KS   0.260  0.002  GB
ROI19  vvjs   4   0.445  0.848  0.041  0.092  KS   0.165  0.116  GB
ROI19  vvjs   4   0.402  0.848  0.041  0.092  KS   0.153  0.172  GB
```

The histograms of peak responses for one region of interest are shown. The X- and Y-axis scales are automatically the same for each. You can reset these with the mouse or the histogram controls. You can also choose to display the histograms summed into a single histogram of peaks or separated by condition.

Binomial statistics assume a finite (small) pool of independent vesicles, and from this, you can estimate the release probability (P_r). Poisson statistics assume an infinite pool, but in doing so, one less parameter is needed to describe the shape of the distribution. However, the “rate” of release that is obtained from Poisson fit is harder to interpret than the release probability.

`SAFT` also offers the opportunity to fit the summed histogram, to find the width (*w*) and quantal spacing (called ‘dF(q)’ in the interface). You can also fix the Gaussian width, *w*, to be the same as the SD for the baseline of the trace (peaks are automatically blanked for this calculation. For some as yet incomprehensible reason, using the SD for the baseline seems to give Gaussian functions that are wider than expected.

When you do a fit, you choose the number of Gaussian components - this is not optimised. If you choose a global fit, the quantal spacing and width is taken to be the same for each condition. The width can be optimised. Fit results are autosaved to a file HFtemp.xlsx in the SAFT directory every time you change ROI. If you click “save”, you get to pick a filename. Tick the box to save the fitted curves in the same excel file in a separate sheet.