# Sliding Window Analysis Toolbox (S.W.A.T)

**Developers:** Khaled Talaat (ktalaat@unm.edu)
and Cameron Trapp

**Principal Investigator:** Dr. Stefan Posse

**Please reference**: Vakamudi, K, Trapp, C, Talaat, K., Gao, K., Sa De La Rocque Guimaraes, B, Posse, *S,* Real-time Resting-State fMRI using Averaged Sliding Windows with Partial Correlations and Regression of Confounding Signals, Brain Connectivity, *in revision*

## About S.W.A.T

Sliding Window Analysis Toolbox is a MATLAB toolbox for signal processing and simulation of resting state fMRI data. The toolbox features a graphical user interface with various functions. The user can:

- Generate or import signals.
- Modify signals through mixing, filtering, frequency injection, or confound definition.
- Define multiple detrending vectors and detrend signals.
- Calculate correlations between signals using a formula of choice.
- Carry out sliding window correlation analysis with or without online detrending and obtain the correlation coefficients as function of window width.
- Animate and simulate online detrending and sliding window analysis.
- Plot and compare the online detrending coefficients (gammas) for different window sizes and signal-vector combinations.
- View the frequency spectrum of the original or the detrended signals.
- Record, edit, and play macros for automation and for more flexible and precise control.
- Run the same analysis multiple times and obtain standard deviations through the batch mode functionality built into the macro system.
- Save and load cases and figures.

## Prerequisites

S.W.A.T was tested on MATLAB 2016a and MATLAB 2017a on both Windows and Ubuntu operating systems. The program requires no special packages.

## Using S.W.A.T

### Importing Signals

S.W.A.T allows you to import signals from .csv, .dat, or .txt files. Each signal in the file can be a row or a column. You will be asked to enter a name for each signal imported. Imported signals must all have the same length. They also should have the same length as any existing signals.

### Generating Signals

You can easily generate signals in S.W.A.T. First, you will be asked about the number of points in the signals, then name of the signal, and then the probability distribution function and the parameters that describe the signal. You may select normal/gaussian, rician, rayliegh, non-central chi-square, or a colored noise of your choice.

## Adding Noise

You can add Rician or Gaussian noise to selected signals with control over the signal to noise ratio and Rician noise parameters. When you add noise to multiple signals, different random noise signals will be injected into each.

## Mixing Signals

In S.W.A.T you can mix one signal into another based on the following equation:

$$Signal1 = (1\text{-}beta) \times Signal1 + beta \times Signal2$$

In the above equation, signal 2 is injected into signal 1 with a mixing coefficient (beta). The mixing coefficient is arbitrary (typically from 0 to 1). The closer the mixing coefficient is to 1, the more similar signal 1 will be to signal 2. This feature may be useful if you wish to inject correlations. S.W.A.T also has a partial , stochastic mixing option. In stochastic mixing some points may not be mixed. Beta would determine the probability that a point in signal 2 is mixed into signal 1.

## Filtering Signals

You can filter the signals using conventional filtering, zero phase filtering, or moving average filtering. Both conventional and zero phase filters use built-in filters in MATLAB. For the moving average filter, you'll need to specify the number of hamming points. Filters are only applied to the signals that you select.

## Exporting Signals

Exporting signals is necessary if you wish to reproduce the same signals for another analysis or if you wish to produce higher quality figures with the data. You can do that using the export signals function. The output file will be a .csv file with each signal as a column.

## Confound Definition

You can interactively inject box cars, spikes, drifts, sinusoids, or shift a signal in x or y from the confounds panel as shown in Figure 1. First, you need to select the signal that you would like to modify from the drop down. This can be either a generated or an imported signal. Then select the type of confound you wish to add. You will be then given a cross-hair to define the width of the confound and will then be asked about its amplitude which can be positive or negative. If you make a mistake, you can use the undo option on the top right. If precision in selection is necessary, you may use the macro system to define the confounds.


**Figure 1:** Confounds Panel in S.W.A.T

## Controlling the Display

S.W.A.T allows you to show some selected signals and hide the rest. The selected signals don't have to fall under the same class. To display selected signals, first make sure that "Show selected signals only" is checked. Then click on select signals. Any signals you select will be added to the display without erasing the existing signals. This allows you to display signals from different classes. If you wish to clear the display and select a new set of signals, simply check "Clear previous selections" and make a new selection.

**Defining Repetition Time**

The default repetition time in S.W.A.T is 1 second. If you wish to modify the repetition time, simply use the Define TR function to do that. This will rescale the display. This a necessary step if you wish to perform an FFT to look at the frequency spectrum.

**Viewing the Frequency Spectrum**

You can view the frequency spectrum through the FFT panel. You can easily adjust the axis limits from the x-lim and y-lim controls. Please note that if you make changes to the display such as changing the signals displayed, you'll need to click on FFT again on the menu or the frequency spectrum will not update.

**Injecting Frequency Confounds**

The frequency confounds menu allows you to inject frequencies in an arbitrary range into the data. This feature works though generating a random signal and applying a bandpass filter to it with a 0.05 transition band. This signal is then added to the original signal but the result will be stored temporarily under a separate class. To implement the frequency confounds added in an actual signal, you must click on "Generate Signal(s)" and select the ones to generate. The new signal will be added to the original signals class. It will not overwrite any of the existing signals. Please note that if you don't click on "Generate Signal(s)" and navigate away from the frequency confounds menu, all confounds you add in the frequency domain will be lost.

**Detrending**

To perform any kind of detrending in S.W.A.T you need to define the detrending vectors first which can be one or multiple. To define a detrending vector, simply select it from the list of vectors (the list contains all imported/generated signals) and click on >> to move it to the detrending vectors list as shown in Figure 2. You can remove a detrending vector by doing the opposite. To obtain full-width detrended signals, you should click on "Detrend" and select the signals to be detrended. The detrended signals will then be available under "Detrended" class. To view detrended signals at any time, you can do that from the "Select signals" menu and "Show selected signals only."

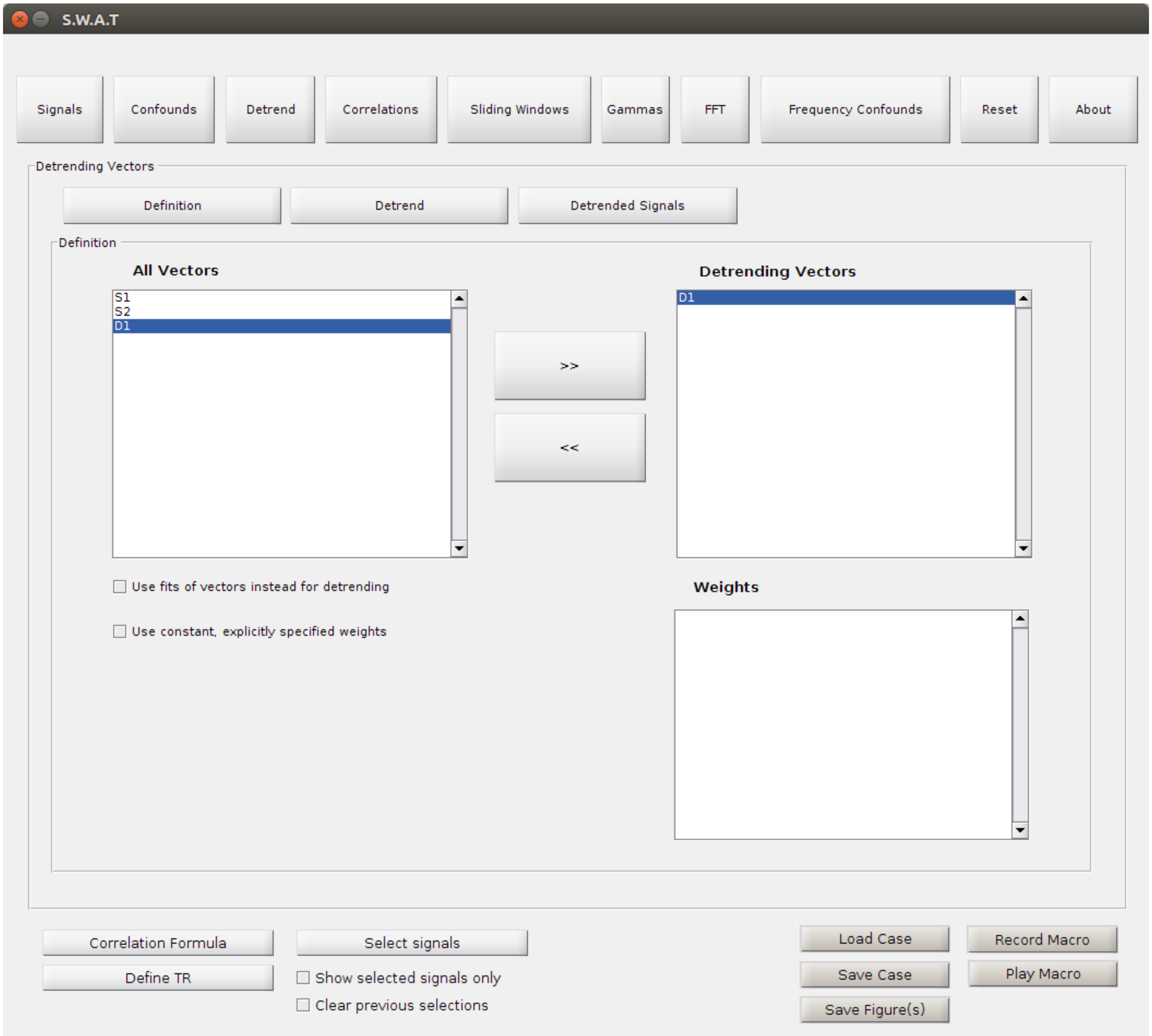


**Figure 2:** Detrending Panel in S.W.A.T

## Correlations Panel

The correlations panel allows you to calculate the correlations and z-scores between different signals of the same class. You can change the correlation formula that S.W.A.T uses in all correlation calculations using the “Correlation Formula” function. You can choose between Pearson correlation and Pearson with no mean subtraction.

## Sliding Window Analysis

The core purpose of S.W.A.T is to simulate sliding window connectivity analysis as a function of window width for different classes of signals. To perform sliding window analysis in S.W.A.T, you first need to define a few parameters. The displacement parameter is simply how many points the window moves each time (default is 1). The initial size parameter is the smallest window width to consider. S.W.A.T will perform the sliding window analysis for windows of different sizes for comparison.The size increment parameter is how much the window width should increase in each analysis. The window count is automatically calculated and is the total number of windows S.W.A.T will use. Please note that this is the number of windows not the number of steps. To start the sliding window analysis, click on “Start Analysis” and select the class of signals. *Current* is the class of signals containing non-detrended signals with confounds and filtering applied, if any. *Detrended* is the class of signals containing the predetrended signals. This will be empty unless you detrend signals from the Detrend panel. The predetrended signals are analyzed just like original signals with no further detrending applied. *Detrend Online* will show you all signals available for online detrending. Online detrending is detrending of the most recent N points using the most recent N points of the detrending vectors. The detrending coefficients are recalculated in each step. Sample sliding window analysis is shown in Figure 3.

To view or compare the results of the sliding window analysis, simply click on Results and select correlations or z-scores. You can select more than one analysis for comparison. The plot shows the average correlation as a function of window width. The batch mode panel is quite similar to the results panel. However, batch mode only works if the macro system is used with the proper card for batch mode specified. Batch mode shows the standard deviation error bars calculated from the results of multiple simulations. A more thorough description of batch mode is provided in the macro section.

**Displaying the Online Detrending Coefficients**

You can display the online detrending coefficients from the Gammas panel. First, select the window size, and then click on Gamma and select the signal. Every time you do this a new curve will be added to the figure without erasing the existing figures. To reset the figure, simply click on Reset Figure. You may also save all the gamma plots as MATLAB .fig files through the “Save plot(s)” function.

**Saving and Loading Cases**

S.W.A.T allows you to save or load cases. Saving a case will save all its associated data in one .case file which you can load at any time.


**Figure 3:** Sliding Window Analysis in S.W.A.T

# The Macro System

The macro system in S.W.A.T is one of the most important features of the program. It can be used to automate all or part of the work, perform batch analysis, or to precisely control some of the variables. The easiest way to write a macro is to record a macro and edit it. The slightly harder way is to learn the syntax and commands. S.W.A.T has its own macro parsing and processing system and is one of the most sophisticated features of the program internally. However, it’s easy to learn and the syntax is very simple.

## Macro Notation

Macros should be saved in a .ktmacro file in ASCII format.

> Denotes a command card. Please refer to the commands table for a list of valid commands.
>> Denotes an input card. A command can have multiple input cards or none at all.
>>> Denotes an input series card. The program will run the macro multiple times using one of the inputs in the series each time.

## Recording a Macro

Recording a macro will record all actions that you do in the program and all inputs that you provide and write out a macro file in the .ktmacro format. You may edit this file and play it in the S.W.A.T at any time. You can make use of this to perform partial or full automation. For instance, if you wish to perform siding window analysis for original, predetrended, and online detrended signals, you can simply record a macro before you do that the first time and play it in the program for other cases with slight modification. Below is a sample macro that does that for signals 1 and 2 as an example.

```
> SlidingWindows
> SlidingWindows_StartAnalysis
>> 1
>> 1,2
>> Original
> SlidingWindows
> SlidingWindows_StartAnalysis
>> 2
>> 1,2
>> Predetrended
> SlidingWindows
> SlidingWindows_StartAnalysis
>> 3
>> 1,2
>> DetrendOnline
```

## Macro Commands

Macro commands are the macro equivalent to clicking on GUI elements. Some commands require inputs and some don't. For instance, if you use a macro command to switch to a different panel, it shouldn't require any input cards. On the other hand, if you use a macro command to mix two signals, you'll need to define what signal to mix into, what signal to mix with, the beta, and the type of mixing as separate input cards. Macro commands are case-sensitive. The first letter of every word is always capitalized. To make the commands more intuitive, the commands were built such that they read the same as the equivalent buttons on the GUI. Because the same button might exist on different panels with different uses, commands for buttons contained on a panel are preceded by the panel name and an underscore. Available macro commands are provided in Table 1.

**Important:** Macro inputs are space and delimiter sensitive. There should always be a space between >> and the input. Don't add spaces between comma or semicolon delimited inputs. Spaces are used as special delimiters for input series cards only. Input cards are also order sensitive.

**Table 1:** List of macro commands and their required input cards

| Command | Use | Required Input Cards | Example |
|---|---|---|---|
| > Signals | Switch display to signals panel | none | > Signals |
| > Confounds | Switch display to confounds panel | none | > Confounds |
| > Detrend | Switch display to detrending panel | none | > Detrend |
| > Correlations | Switch display to correlations panel | none | > Correlations |
| > SlidingWindows | Switch display to sliding windows panel | none | > SlidingWindows |
| > FFT | Switch display to fast fourier transform panel | none | > FFT |
| > FrequencyConfounds | Switch display to frequency confounds panel | none | > FrequencyConfounds |
| > Reset | Reset will delete the generated signals and clear all variables except those associated with the batch mode. Reset | none | > Reset |

| | | | |
|---|---|---|---|
| | command is important in batch analysis. | | |
| > Signals_ImportSignals | Import signals from a file | A card is required for exact path to file and a separate card is required for the name of each signal in the file | > Signals_ImportSignals<br>> /home/signals.csv<br>> S1<br>> S2<br>> S3 |
| > Signals_GenerateRandomSignal | Generate random signals with specific characteristics | If it's the first time that you call the card in the macro file, the first input will be the number of points. If it's not, then you shouldn't specify it again or that will result in an error. You always need to specify the signal name, distribution selection (just the order of the selection in the GUI), and parameters associated with the distribution (single card with distribution parameters delimited by a semicolon) | > Signals_GenerateRandomSignal<br>>> 2000<br>>> S1<br>>> 2<br>>> 1.000000,1.000000,<br>> Signals_GenerateRandomSignal<br>>> S2<br>>> 2<br>>> 1.000000,1.000000,<br><br>The above example generates 2 signals titled S1 and S2 with a Rician distribution with a non-centrality parameter of 1 and sigma of 1. The order of inputs is the same as that in the GUI. The reason 2 indicates Rician here is because Rician is the second option in the GUI. |
| > Signals_AddNoise | Add noise to a signal | Signals to add noise to.<br>Signal to noise ratio.<br>Noise distribution.<br>Rician parameters in case of Rician distribution. | > Signals_AddNoise<br>>> 1,2<br>>> 10<br>>> 1<br><br>The above example adds Gaussian noise to signals 1 and 2 with a signal to noise ratio of 10.<br><br>> Signals_AddNoise<br>>> 2<br>>> 10<br>>> 2<br>>> 1,0.1<br><br>The above example adds Rician noise to signal 2 with an SNR of 10. The Rician noise has a non-centrality parameter of 1 and a sigma of 0.1. |

| | | | |
|---|---|---|---|
| > Signals_Filters | Filter a particular signal | Signal to filter<br>Type of filter<br>Path to filter object (in case of conventional or zero phase filter. For moving average filter, specify the number of hamming points) | > Signals_Filters<br>>> 5<br>>> 2<br>>> /home/filter.d<br><br>This applies a zero phase filter (option 2) to signal 5 from the list using the filter parameters specified by the object at /home/filter.d.<br><br>If you wish to use a conventional filter, specify 1 instead of 2. For a moving average filter, specify 3 and instead of the /home/filter.d entry, specify the hamming points. |
| > Signals_MixSignals | Mix a signal into another signal with a mixing coefficient. | The signal to be modified.<br>The signal that should be mixed with it.<br>The mixing coefficient.<br>The type of mixing (All or Partial) | > Signals_MixSignals<br>>> 1<br>>> 2<br>>> 0.2<br>>> 1<br><br>The above example mixes signal 2 into signal 1 (modified signal) with a mixing coefficient of 0.2 and mixing type of 1 (All) |
| > Signals_ExportSignals | Export all signals to a .csv file | Exact location of .csv file to be created | > Signals_ExportSignals<br>> /home/myexportedsignals.csv |
| > Signals_Undo | Equivalent to clicking on undo button on the signals panel | none | > Signals_Undo |
| > Confounds_SelectSignal | Select a signal to add confounds to. You don't have to select the same signal again to add additional confounds. However, if you wish to modify a different signal you'll need to select it. | Order of the signal from the dropdown list in the confounds panel | > Confounds_SelectSignal<br>>> 2 |
| > Confounds_Boxcar | Add a box car to a signal between two defined points | The two x axis points that define the boxcar width, and the boxcar amplitude. | > Confounds_Boxcar<br>>> 393;543<br>>> 8.000000<br><br>The above example adds a boxcar with amplitude of 8 between the 393rd point and the 543rd point of the selected signal. |
| > Confounds_Spike | Add a spike to a signal at a point | The point at which the spike should be | > Confounds_Spike<br>>> 674 |

| | | | |
|---|---|---|---|
| | | placed in x, and the spike amplitude. | >> 12.000000<br><br>The above example adds a spike with amplitude of 12 at the 674th point of the selected signal. |
| > Confounds_Sinusoid | Add a half sine to a signal between two defined points | The two x axis points that define the sinusoid width, and the peak amplitude. | > Confounds_Sinusoid<br>>> 393;543<br>>> 8.000000 |
| > Confounds_Drift | Add a drift to a signal between two defined points | The two x axis points that define the drift width, and the peak amplitude. | > Confounds_Drift<br>>> 500;2000<br>>> 8.000000 |
| > Confounds_Xshift | Shift the signal cyclically in x | The amplitude of the shift in x | > Confounds_Xshift<br>>> 100 |
| > Confounds_Yshift | Shift the signal in y | The amplitude of the shift in y | > Confounds_Yshift<br>>> 2 |
| > Confounds_Undo | Equivalent to clicking on undo on the confounds panel | none | > Confounds_Undo |
| > DefineTR | Defines the repetition time (TR) | The TR | > DefineTR<br>>> 0.400000 |
| > FrequencyConfounds_SelectSignal | Selects a signal from the signals list in the frequency confounds panel | The order of the signal to be selected | > FrequencyConfounds_SelectSignal<br>>> 1 |
| > FrequencyConfounds_AddConfound | Add noise from a specific frequency band to a signal. | The frequency band and the boost parameter. | > FrequencyConfounds_AddConfound<br>>> 0.2;0.4<br>>> 50 |
| > FrequencyConfounds_GenerateSignal | Generates a signal from the modified signal. | The list of modified signals in the frequency spectrum that should be generated in the time domain and added to original signals. The generated signals will be the last in the original signals list. | > FrequencyConfounds_GenerateSignal<br>>> 1,2 |
| > Detrend_AllVectorsList | Selects a signal for detrending vector definition | The order of the signal from the list on the left on the detrending panel | > Detrend_AllVectorsList<br>>> 3<br><br>Selects signal 3 from the all vectors list. |
| > Detrend_AddDetrendingVectors | Adds the selected signal to the list of detrending vectors | none | > Detrend<br>> Detrend_AllVectorsList<br>>> 3<br>> Detrend_AddDetrendingVectors |

| | | | |
|---|---|---|---|
| > Detrend_DetrendingVectorsList | Selects a detrending vector from the detrending vectors list for removal. | The order of the detrending vector to be removed. | > Detrend_DetrendingVectorsList<br>>> 1 |
| > Detrend_DeleteDetrendingVectors | Removes a selected detrending vector from the list of detrending vectors | none | > Detrend_DetrendingVectorsList<br>>> 1<br>> Detrend_DeleteDetrendingVectors |
| > Detrend_UseFits | Equivalent to checking “Use fits of detrending vectors” on the GUI | Order of the fit curve | > Detrend_UseFits<br>>> 2 |
| > Detrend_Detrend | Detrends selected signals | Signals to be detrended | > Detrend_Detrend<br>>> 1,2<br><br>The above example detrends signals 1 and 2 using the detrending vectors that had been specified. |
| > CorrelationFormula | Selects a correlation formula to be used in correlation calculations | Select the correlation formula from the available options. Check the GUI to see the options. | > CorrelationFormula<br>>> 2<br><br>The above example uses pearson’s correlation without subtraction of mean instead of conventional pearson correlation. |
| > SlidingWindows_InitialSize | Specifies the width of the smallest window to consider | The initial size in seconds | > SlidingWindows_InitialSize<br>>> 20 |
| > SlidingWindows_SizeIncrement | Specifies the value for the window size increment | The size increment in seconds | > SlidingWindows_SizeIncrement<br>>> 20 |
| > SlidingWindows_StartAnalysis | Starts the sliding window analysis. This is equivalent to clicking Start Analysis on the GUI. | The signal set/class.<br>The signals to be correlated.<br>Name of the analysis. | > SlidingWindows<br>> SlidingWindows_StartAnalysis<br>>> 1<br>>> 1,2<br>>> Original<br><br>The above example starts the sliding window analysis for signals 1 and 2 from the Current class. The analysis is named “Original.”<br><br>> SlidingWindows<br>> SlidingWindows_StartAnalysis<br>>> 3<br>>> 4,5<br>>> DetrendOnline<br><br>The above example starts the |

| | | | |
|---|---|---|---|
| | | | sliding window analysis for signals 4 and 5 from the Detrend Online class. As the signals belong to the Detrend Online class, detrending is performed during the sliding window analysis. |
| > SaveCase | Saves the entire case to the macrosaves directory. The case will be saved with the macro iteration concatenated to the name. If you are running in batch mode, it might be necessary to use SaveCase before Reset command to be able to recover the dtaa. | none | > SaveCase |
| > SaveFigures | Saves the figures from the available options in the GUI. | The figures to be saved | > SaveFigures<br>>> 1,2,3,4<br><br>The above example saves the following figures: original signals, predetrended signals, sliding window results, and FFT figure. |
| > SavePlotsGammas | Saves the figures for online detrending coefficients. | Size of the window. The signal you wish to plot the detrending coefficients for. | > SavePlotsGammas<br>>> 1,2,3,4,5<br>>> 1,2<br><br>The above example saves the gammas for the smallest 5 window widths used for signals 1 and 2 as function of window step. |
| **Batch Mode (Important)**<br><br>> NRuns | Defines the number of runs in batch mode analysis. If you are generating random signals, each time different random signals will be generated. You should be able to see the mean correlation and standard deviation as function of window size from Batch Mode in the sliding windows panel. | The number of runs. | > NRuns<br>>> 3<br>..<br>Other macro commands<br>..<br>> SaveCase<br>> Reset |

## Input Series

All inputs in the Table 1 used a >> notation. The >> implies that the input is fixed. However, if you wish to vary the input and re-do the calculations for that input, you may use a >>> notation for the input. You may only do this once in the entire macro.


## Acknowledgments

We thank Arvind Caprihan, PhD for his advice on signal modeling and Kishore Vakamudi for debugging, testing, and evaluation of this software.

This work was funded by the National Institutes of Health (grants 1R21EB022803, 1R41NS090691, 1R21EB018494, 1P30GM122734, and 2P20GM10347).