SPA Toolbox Manual 1.0

Guang Ouyang, 2022 Feb

I. General

SPA is an EEG toolbox for removing EEG or ERP based on variance of components decomposed by PSC. The theoretical ground is that variance of PC components from raw EEG data form bimodal distribution, suggesting that artifacts can be removed based on simple thresholding of variance. Such algorithm enjoys high computational efficiency while largely preserving the neural components and effects. Detailed description and validation of SPA can be found from:

Ouyang, G., Dien, J., & Lorenz, R. (2021). Handling EEG artifacts and searching individually optimal experimental parameter in real time: a system development and demonstration. *Journal of Neural Engineering*.

Originally, SPA was developed to quickly remove artifacts from single trial ERP data. SPA originally stands for "Single-trial PCA-based Artifact removal". In this toolbox, we extended SPA to processing resting state (or continuous EEG data). Therefore, the name becomes "Segment-by-segment PCA-based Artifact removal", which can also be abbreviated as SPA. The extended algorithm for resting-state data is the same of on ERP data except that we applied a smoothing algorithm to smooth adjacent segments of resting state data (see below for details).

II. Installation

SPA is an extension (plugin) toolbox under EEGLab toolbox. You can directly download SPA from within EEGLab (File → Manage Extensions) or from the GitHub website (https://github.com/guangouyang/SPA). If you download the zip from GitHub, upzip the folder and simply put the SPA.0 folder under the 'plugins' folder in EEGLab directories. Make sure that it is the folder that contains all the matlab scripts (named 'ReSync1.0') is put under 'plugins'. After this is set, launch EEGLab and you will see SPA under Tools after you load some EEG dataset into EEGLab.

III. Apply SPA one continuous EEG data using scripts

It is extremely simple to apply SPA on your EEG data. I suggest you apply some basic preprocessing first before applying SPA (e.g., band-pass filtering, referencing), but no need to apply sophisticated artifact extraction and identification methods, as the whole point of SPA is to replace those sophisticated and time-consuming artifact handling methods in some circumstances.

In the toolbox folder we have a sample EEG data. This is a dataset from a single participant doing a visual oddball task ('S 22', frequent ball; 'S 21', odd ball). This data has been band-pass filtered at 1-40Hz and average referenced. Load the data into EEGLab using UI or script.

```
EEG = pop_loadset('filename','sample_data.set','filepath','sample_data\');
```

If you plot the EEG data it looks like this:

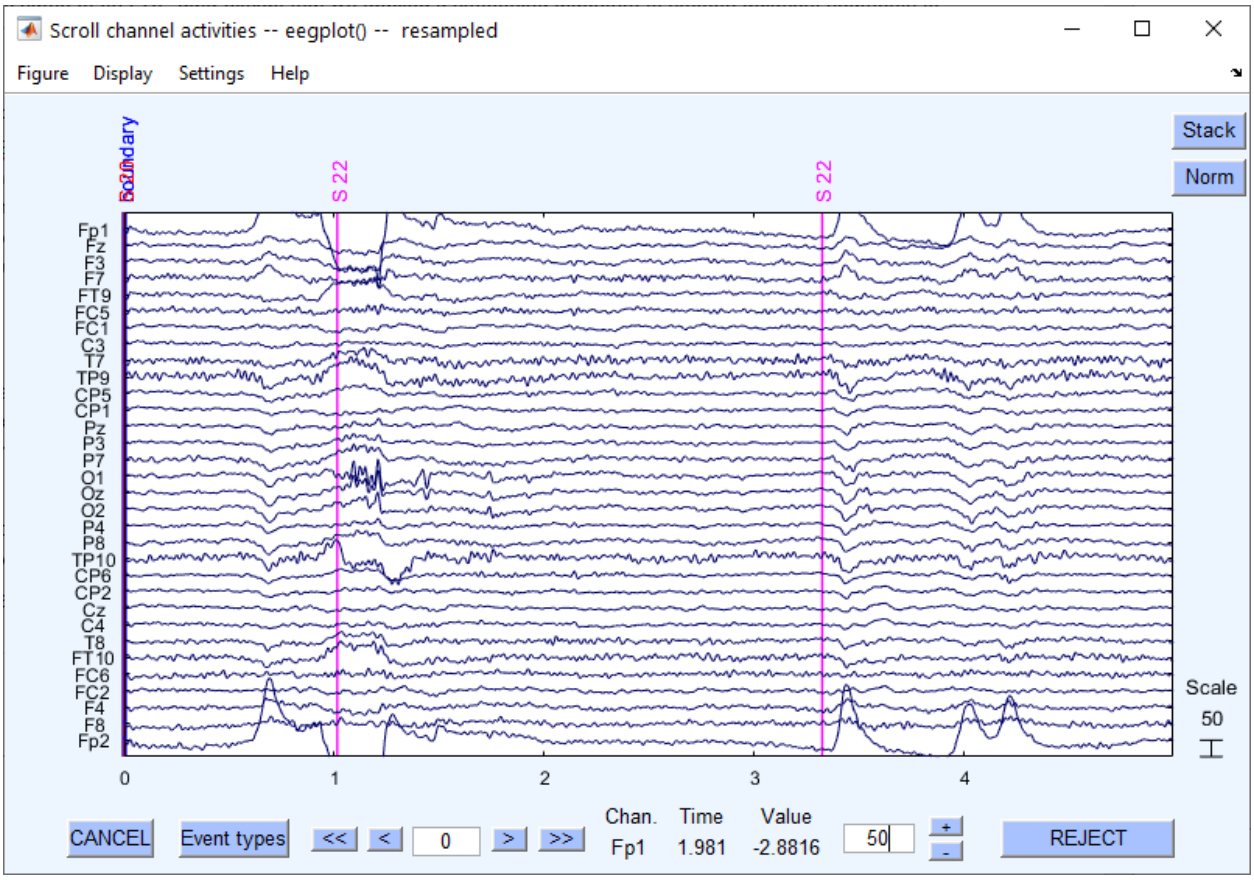

Fig 1.

Now let's treat this dataset as resting state EEG data and directly apply SPA to remove the artifact:

```
EEG = SPA_EEG(EEG,30,2,2);
```

And plot the data again, it looks like this (remember to set the vertical scale to 50):

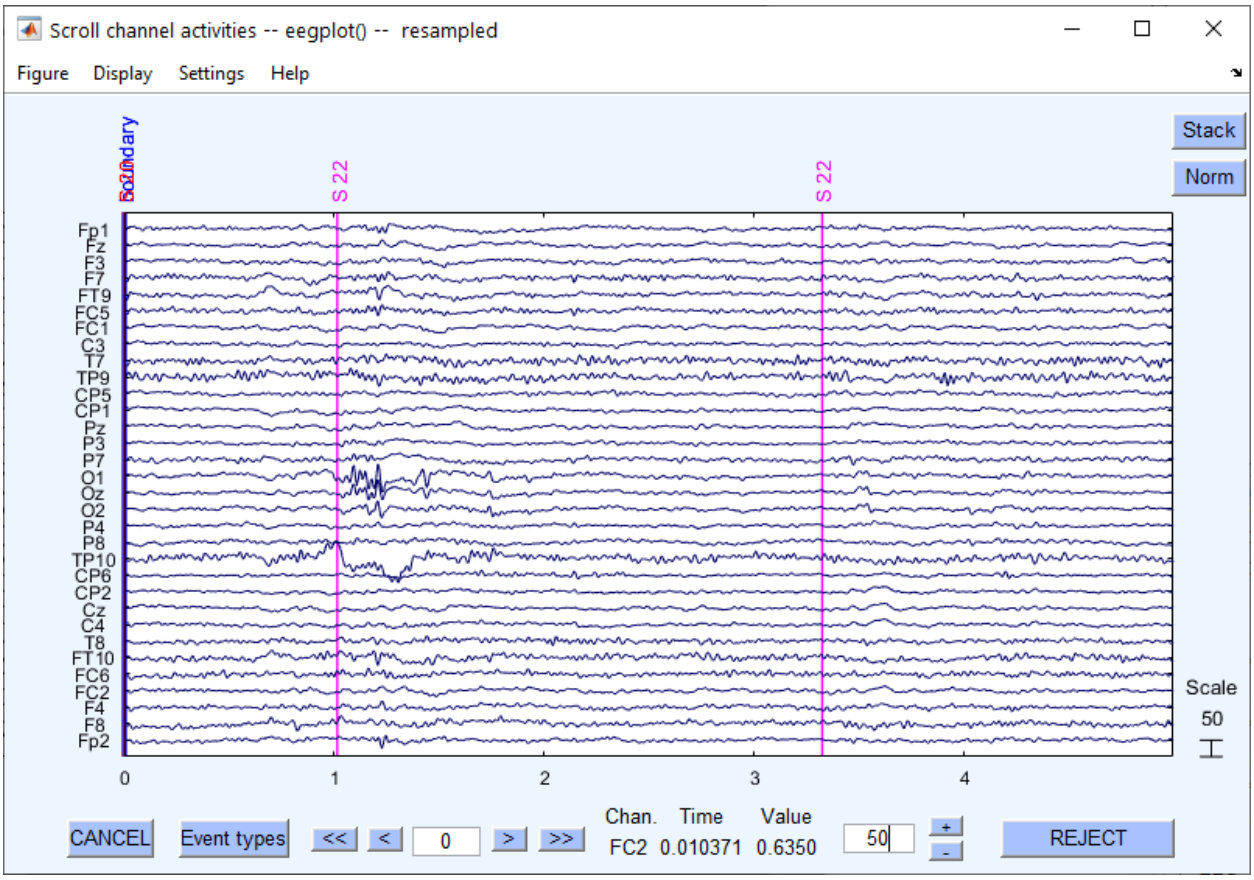

Fig 2.

The major artifact components (ocular) appear to have been removed without substantially affecting the ongoing activities. This is how to apply SPA on resting state (continuous) EEG data using only a single line. The meaning of the SPA parameters, 30, 2, 2 will be explained later. You are recommended to use these default values.

IV. Apply SPA on ERP data using scripts

While you can apply SPA to your EEG data treating it as a continuous segment, and extract the ERP data afterwards, you can also directly apply SPA on single trial ERP data using the following script:

```
markers = {'S 22','S 21'};
epoch_twd = [-200,1000];
baseline_twd = [-200,0];
EEG = SPA_ERP(EEG,30,markers,epoch_twd,baseline_twd);
```

Again, here 30 is the threshold. ‘markers’ are the time markers from which you want to extract your ERP. ‘epoch_twd’ and ‘baseline_twd’ refers to the ERP window and baseline window. After this, you can see several new variables generated under ‘EEG’. They are: ‘ERPs’,’ERPs_SPA’,’ST’,’ST_SPA’,’t_axis’. They refer to the original (artifact uncorrected) and SPAed ERPS, original and SPAed single trials, and the time points. To understand them, you can plot them our using the following script:

While you can apply SPA to your EEG data treating it as a continuous segment, and extract the ERP data afterwards, you can also directly apply SPA on single trial ERP data using the following script:

```
figure;
subplot(2,2,1);plot(EEG.t_axis,EEG.ERPs(:,:,1));
title('original ERP (S 22)');xlabel('time (ms)');xlim([-200,1000]);ylim([-
20,20]);
subplot(2,2,2);plot(EEG.t_axis,EEG.ERPs(:,:,2));
title('original ERP (S 21)');xlabel('time (ms)');xlim([-200,1000]);ylim([-
20,20]);
subplot(2,2,3);plot(EEG.t_axis,EEG.ERPs_SPA(:,:,1));
title('SPA ERP (S 22)');xlabel('time (ms)');xlim([-200,1000]);ylim([-20,20]);
subplot(2,2,4);plot(EEG.t_axis,EEG.ERPs_SPA(:,:,2));
title('SPA ERP (S 21)');xlabel('time (ms)');xlim([-200,1000]);ylim([-20,20]);
```

This will generate the following ERP figure for your information. It shows how the artifacts are being removed:

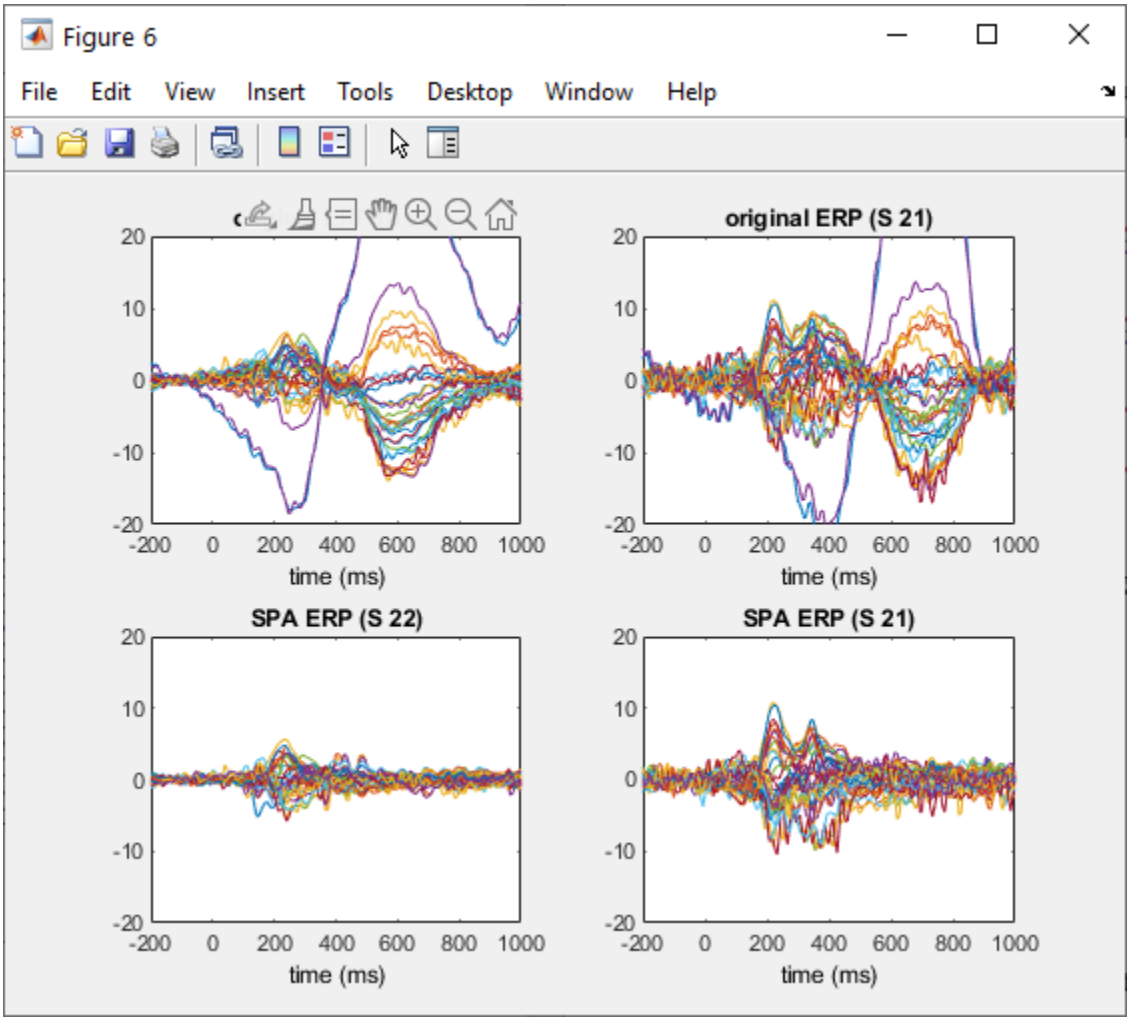

Fig 3.

Note: the second hump in Fig 3 (from ~500 ms to ~800 ms) is an eye-blink related artifact component. This hump is also removed by advanced artifact algorithm such as ICA.

V. Apply SPA using UI.

You can also play around with SPA using the UI item of "SPA" under tools. But the simple script can already do pretty much everything.

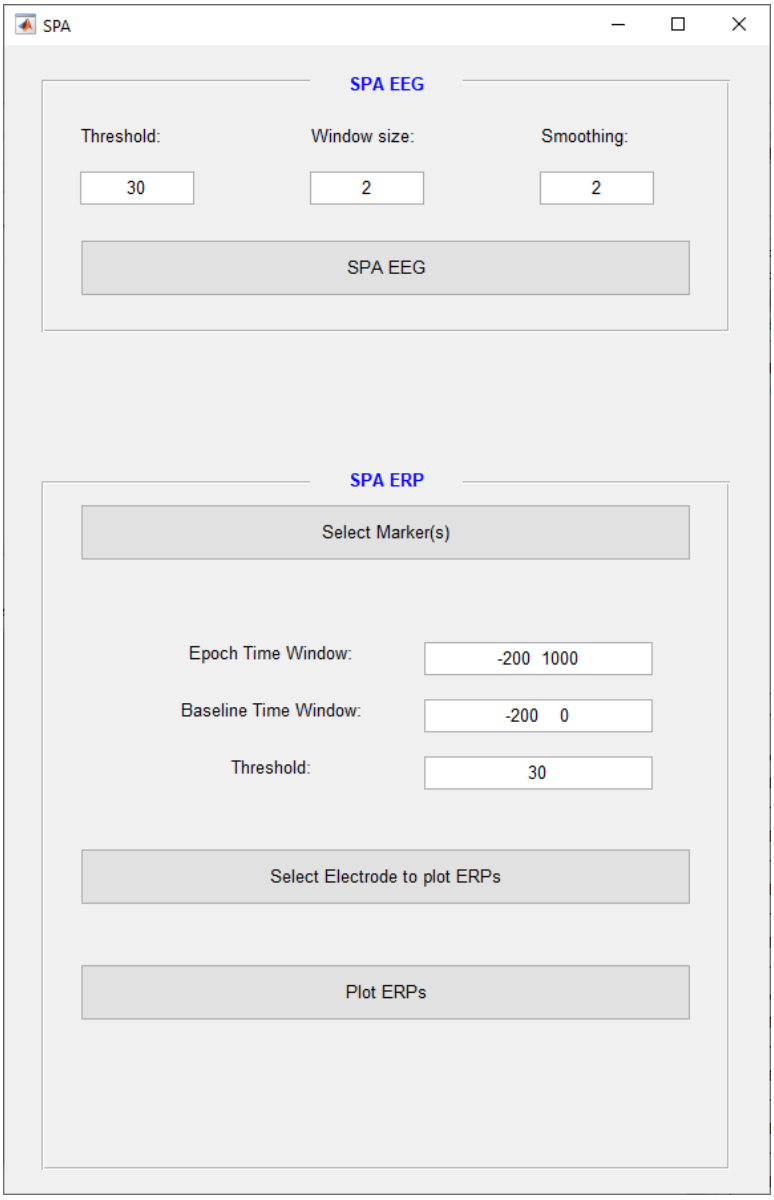

Fig 4.

Appendix: meaning of parameter.

```
SPA_EEG(EEG,threshold,window_size,smoothing);
```

Threshold: the level of variance magnitude above which the PC will be removed (see the paper). Default: 30.

Window size: the size of window of EEG segments that SPA applies on continuous EEG data. Default: 2 seconds.

Smoothing: The parameter for smoothing adjacent segments (must be larger than 1). Higher value, more abrupt transition. Default: 2.