BCI HW2 0816077 陳柏安

Part 1. Multiple Choice

Problem 1. Imagine that we are looking for a 5 μV ERP effect, and the noise is 10 μV in the single-trial EEG, giving us a 5:10 (or 1:2) signal-to-noise ratio on single trials. How many trials would we need to average together to get a 2:1 signal-to-noise ratio in the averaged ERP waveform?

We assume that the signal is uncorrelated to the noise and noise in all trials are not correlated. According to the hint, after averaging N trials of recorded signals, we have the resulting variance which is 1/N of the original, getting 1/sqrt(N) of the original noise level. To enhance signal-to-noise ratio from 1:2 to 2:1, we must reduce noise to 1/4 = 1/sqrt(N), N = 16. We need to average 16 trials together to make the ratio 2:1.

Problem 2. The following are techniques that are commonly applied to EEG data. Which ones are unsupervised? (there may be more than one correct answer)

(A)PCA: Principal component analysis is an **unsupervised** method applied on data to reduce dimensions. It doesn't consider about labels.

(B)LDA: Linear discriminant analysis is a supervised method of dimension reduction since it needs labels to make sure data with the same label stay close to each other and keep data with different label far away.

(C)CSP: Common spatial pattern is a supervised method to separate a signal into subcomponents which have maximum differences in variance.

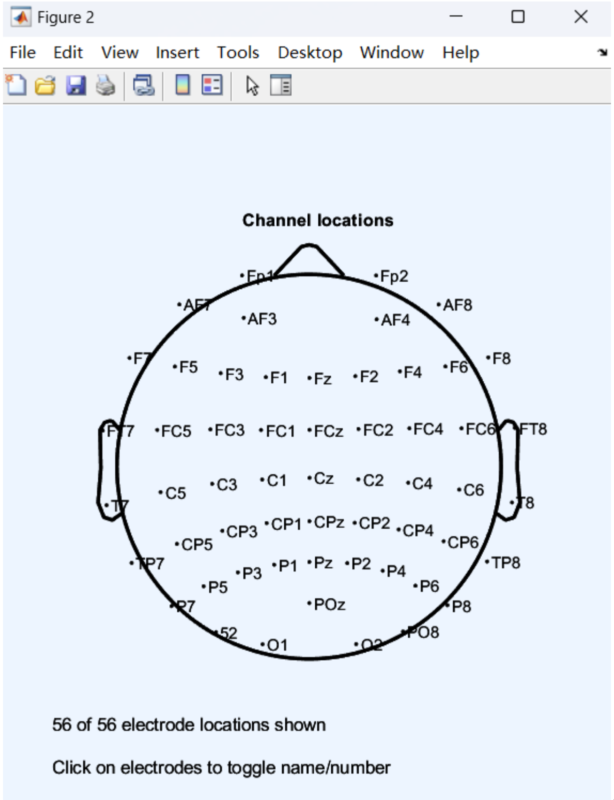
(D)ICA: Independent component analysis is an unsupervised method for extracting some components from the original signal data.

(E)K-means clustering: This is an unsupervised method of partitioning data into K different clusters where each data point belongs to the closest mean center.

Part 2. Programming Problem

Problem 1

1. Plot 2D channel location map



2. Run ICA and record computational time of ICA by code.

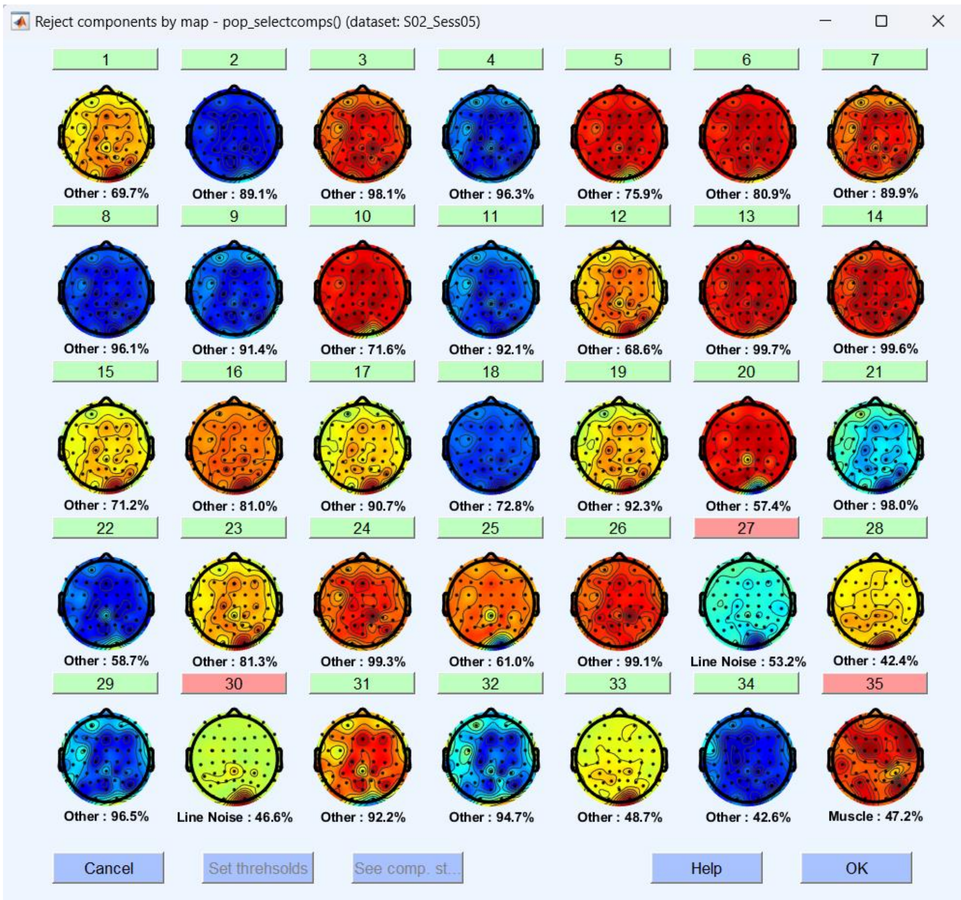
```
tic
EEG = pop_runica(EEG, 'icatype', 'runica', 'extended',1,'interrupt','on');
[ALLEEG EEG] = eeg_store(ALLEEG, EEG, CURRENTSET);
toc
```

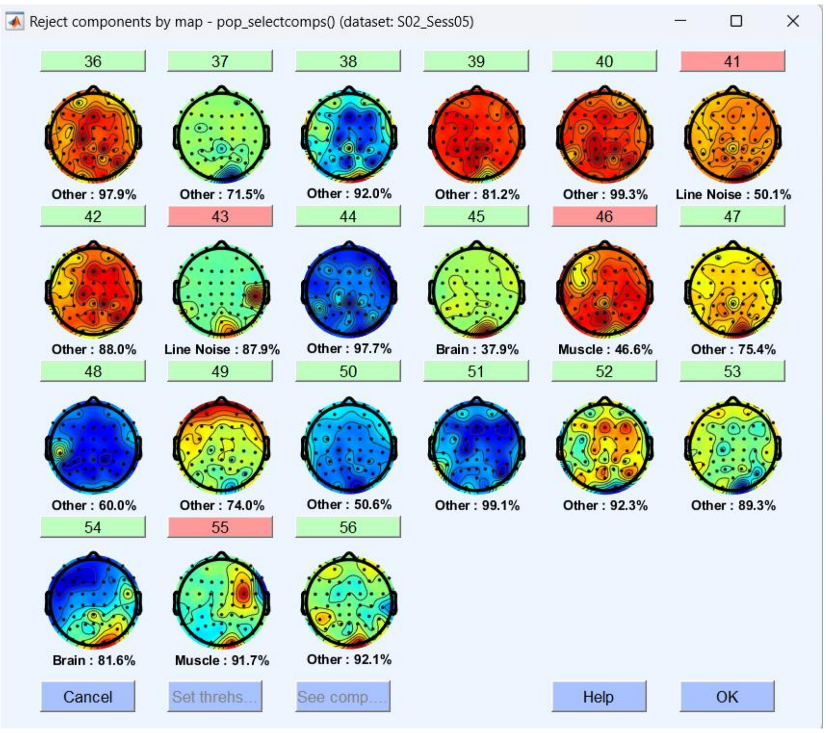
```
step 395 - lrate 0.000000, wchange 0.00000010, angledelta 93.0 deg
Sorting components in descending order of mean projected variance ...
Scaling components to RMS microvolt
Scaling components to RMS microvolt
Scaling components to RMS microvolt
Elapsed time is 659.861810 seconds.
```

It took about 11 mins to finish the process.

3. Plot component maps in 2D.

4. Indicate noise component(s) if they exist and explain the reason why you identify this component as noise or artifacts

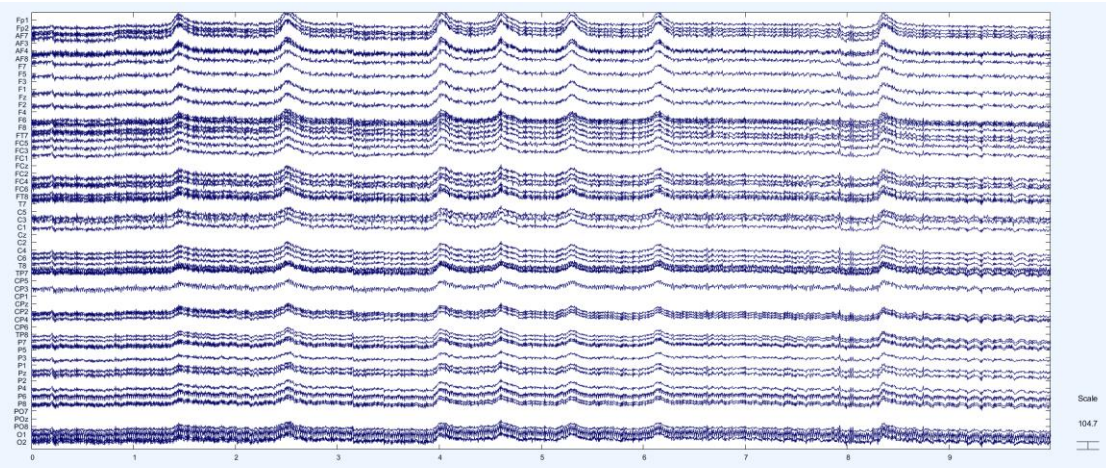




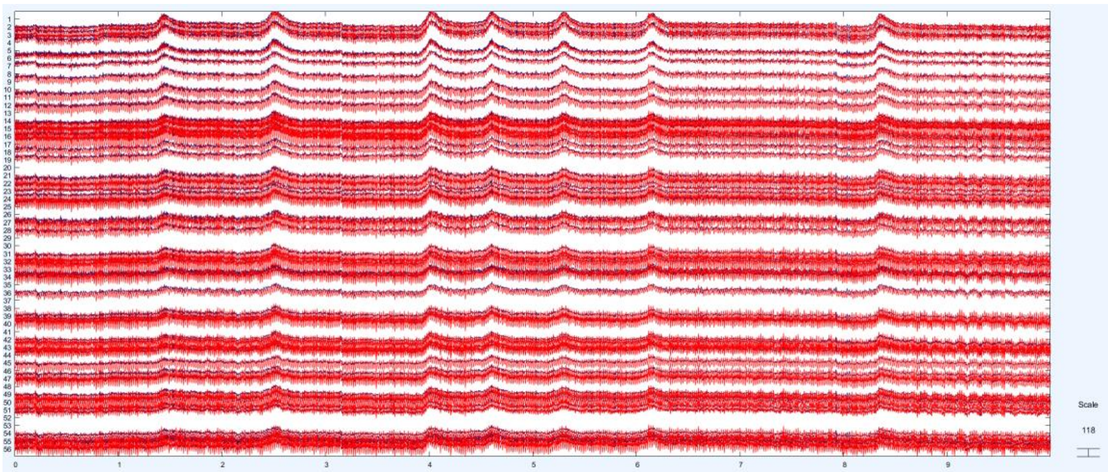
I used the built-in tool to label the components. Since the data was so noisy and most of the components were classified as Others, I only selected the obvious ones like line noise and muscle movement.

5. Plot first 10-second channel data before and after deleting noise/artifact component(s).

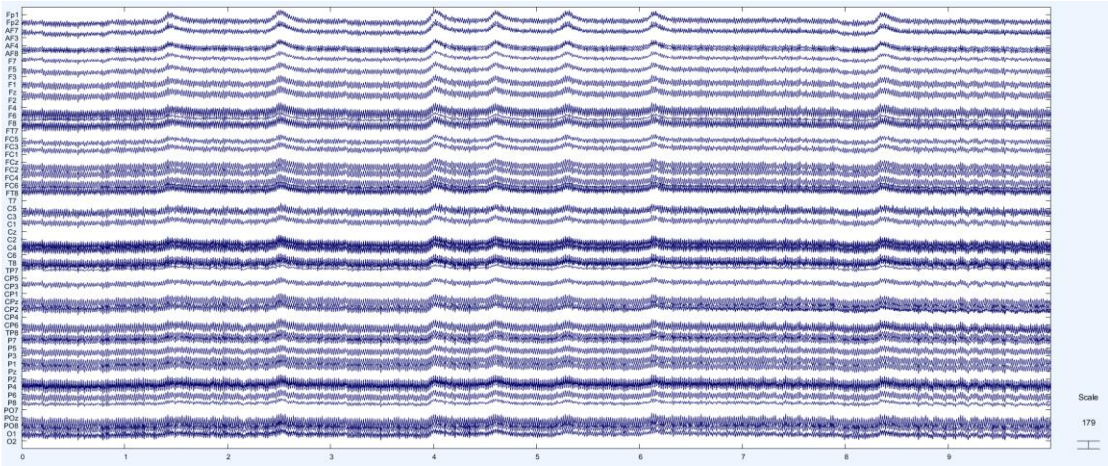
Before deletion



Red lines are the signal that will be deleted



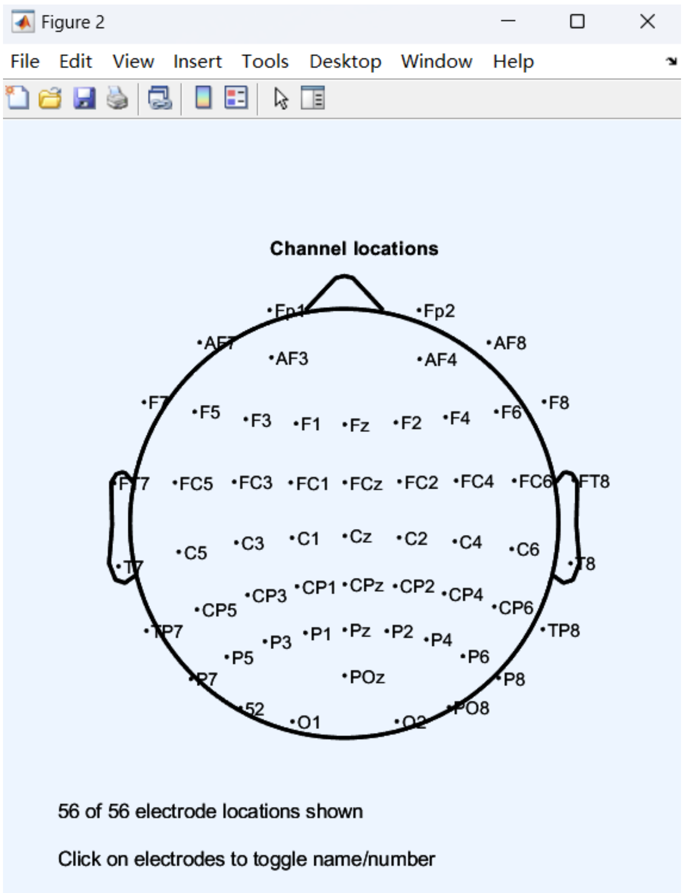
After deletion



There were no significant changes.

Problem 2

1. Plot 2D channel location map

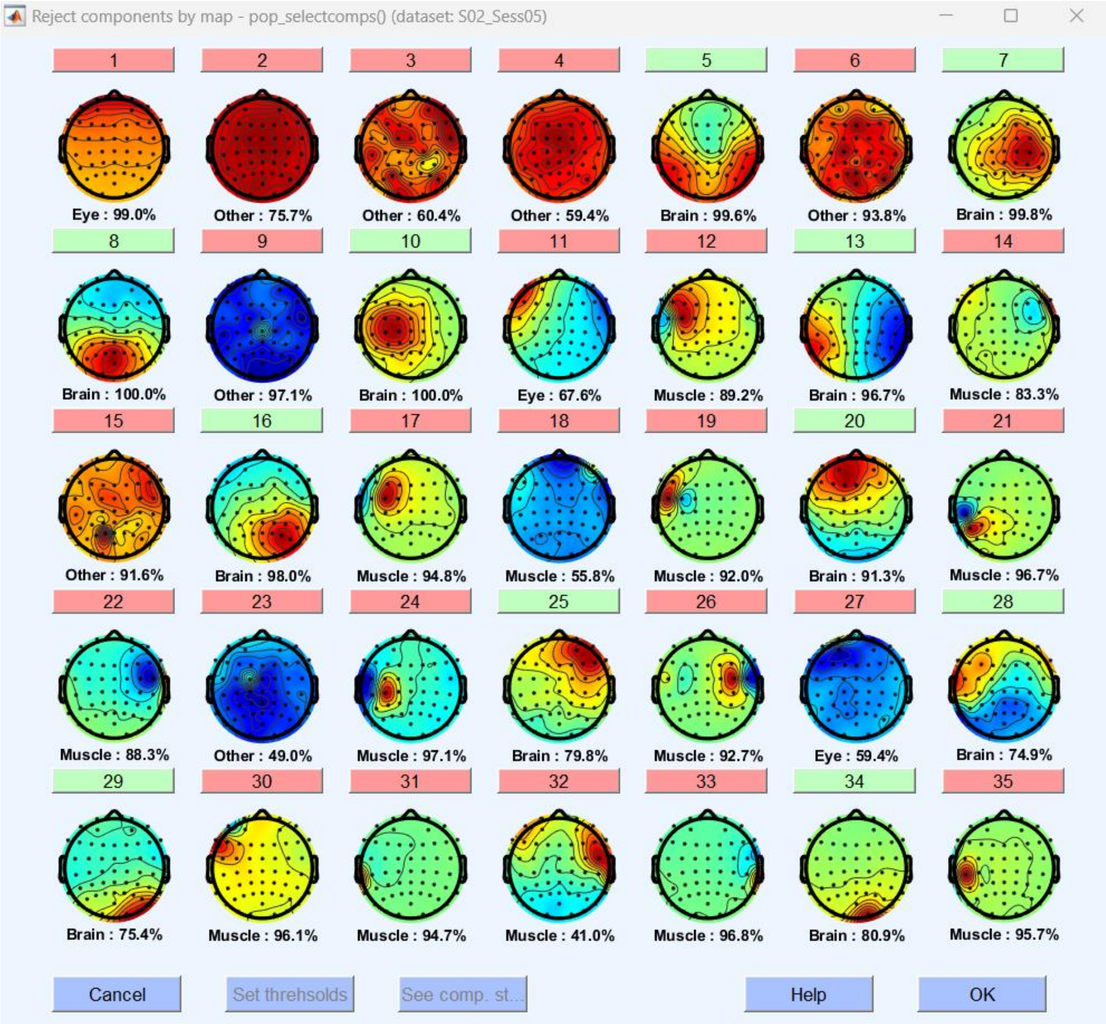


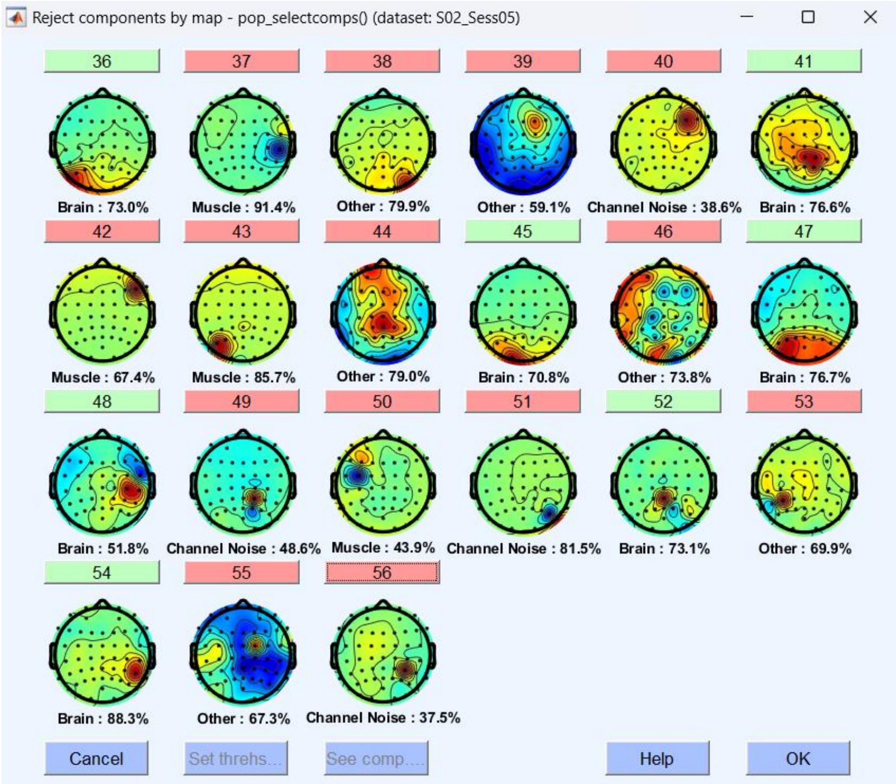
2. Bandpass filtering [1, 48] Hz.
3. Run ICA and record computational time of ICA by code.

```
step 401 - lrate 0.000000, wchange 0.00000009, angledelta 96.7 deg
Sorting components in descending order of mean projected variance ...
Scaling components to RMS microvolt
Scaling components to RMS microvolt
Scaling components to RMS microvolt
Elapsed time is 767.387879 seconds.
```

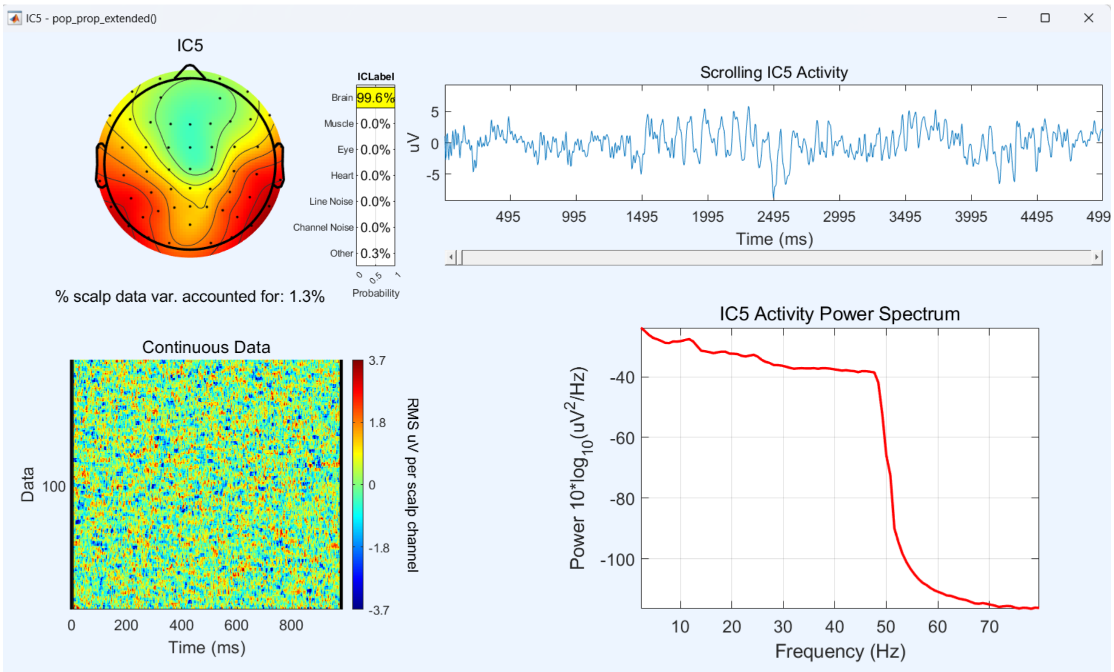
It took a little bit longer to finish the computation, about 13 mins.

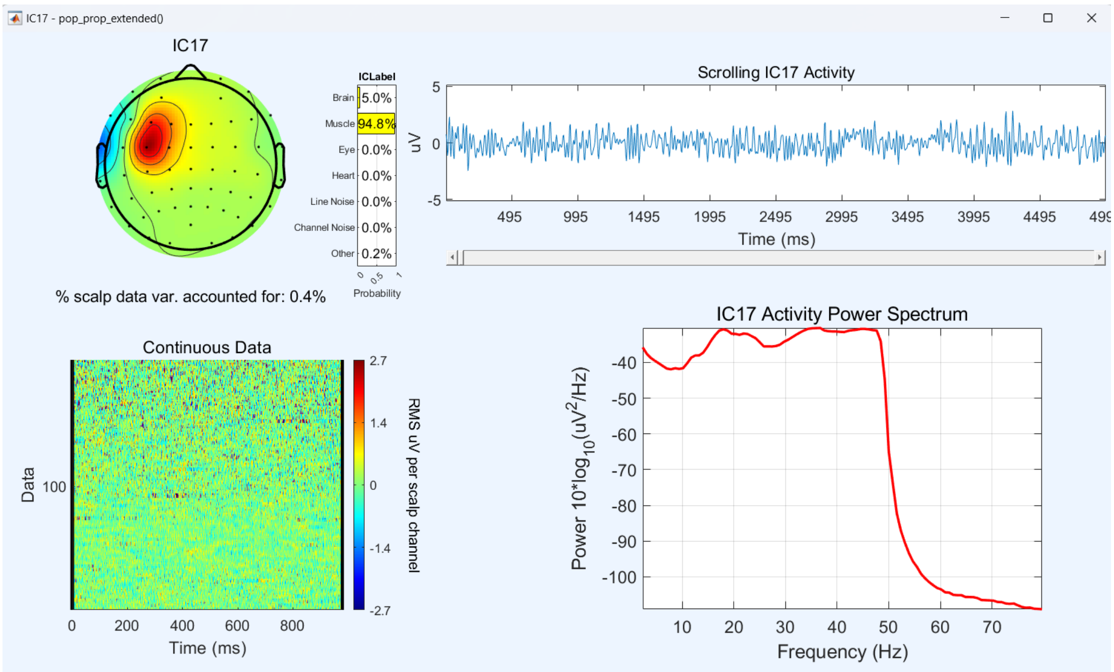
4. Plot component maps in 2D.
5. Indicate noise component(s) if they exist and explain the reason why you identify this component as noise or artifacts.

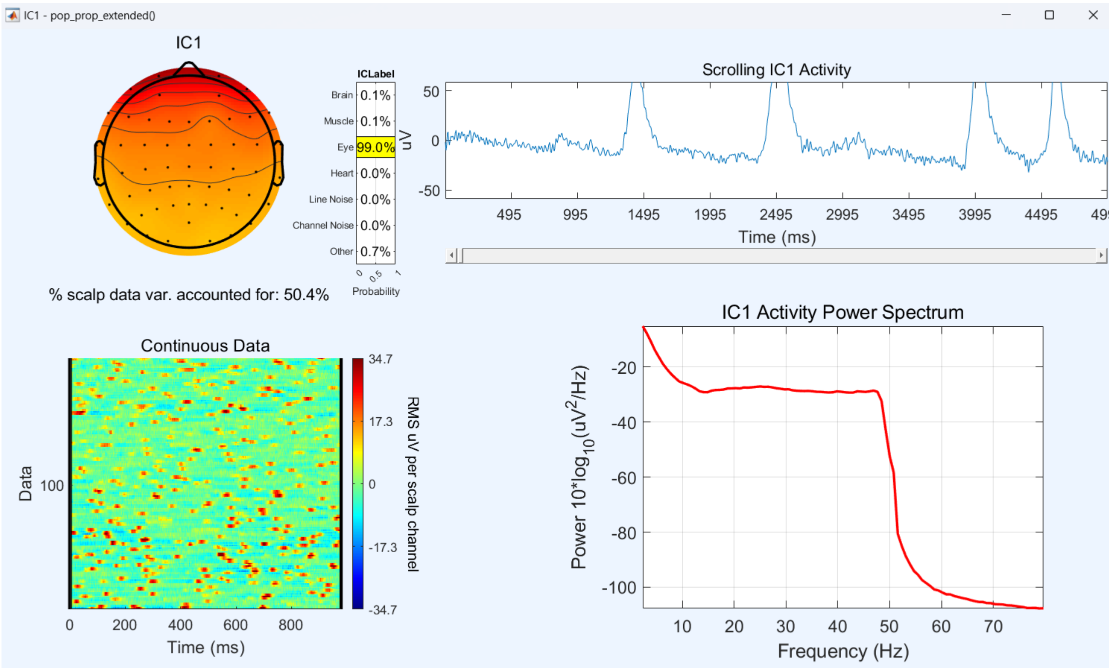


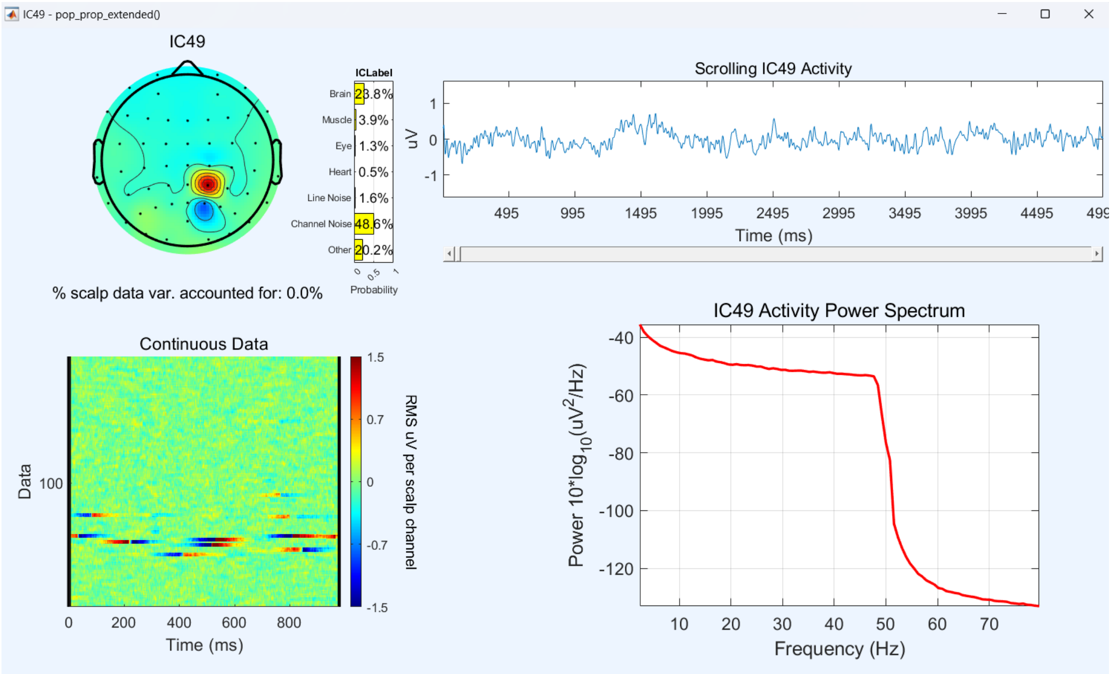


Comparing to the component map in the previous problem, there are much more obvious cases and can be easily removed.




IC17 - pop_prop_extended()
IC17
ICLabel
Brain 5.0%
Muscle 94.8%
Eye 0.0%
Heart 0.0%
Line Noise 0.0%
Channel Noise 0.0%
Other 0.2%
Probability
% scalp data var. accounted for: 0.4%
Scrolling IC17 Activity
Time (ms)
Continuous Data
Data
Time (ms)
RMS uV per scalp channel
IC17 Activity Power Spectrum
Power 10*log10(uV2/Hz)
Frequency (Hz)

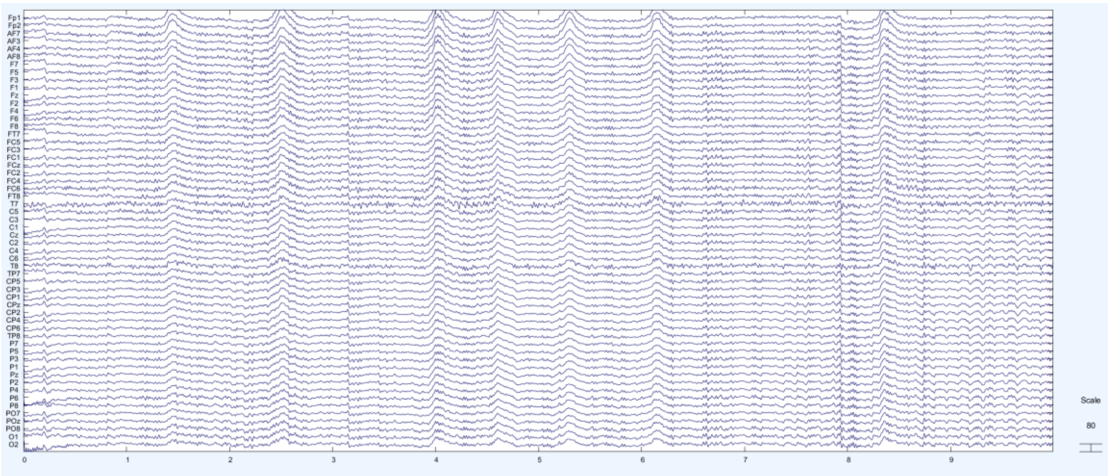

IC1 - pop_prop_extended()
IC1
ICLabel
Brain 0.1%
Muscle 0.1%
Eye 99.0%
Heart 0.0%
Line Noise 0.0%
Channel Noise 0.0%
Other 0.7%
Probability
% scalp data var. accounted for: 50.4%
Scrolling IC1 Activity
Time (ms)
Continuous Data
Data
Time (ms)
RMS uV per scalp channel
IC1 Activity Power Spectrum
Power 10*log10(uV2/Hz)
Frequency (Hz)



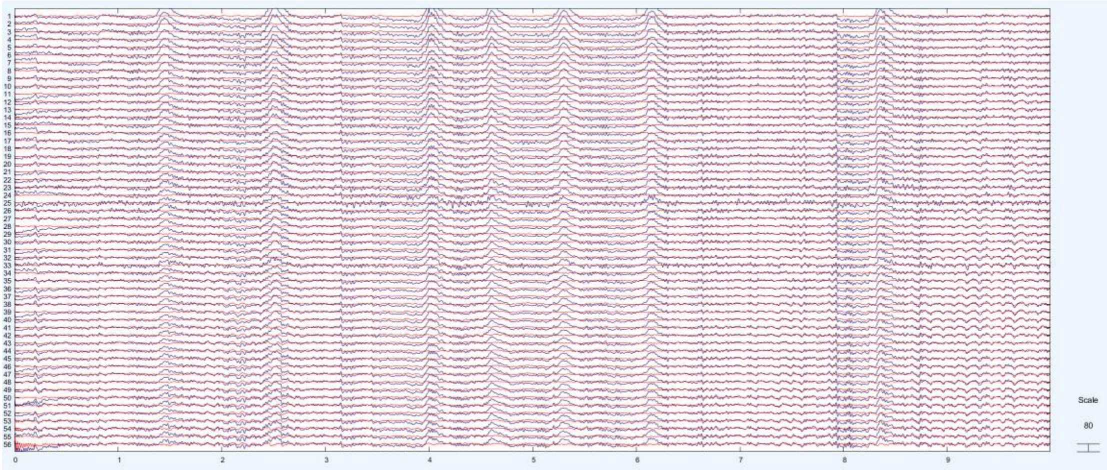
By observing the charts above, we can understand the pattern of these signal or artifact.

6. Plot first 10-second channel data before and after deleting noise/artifact component(s).

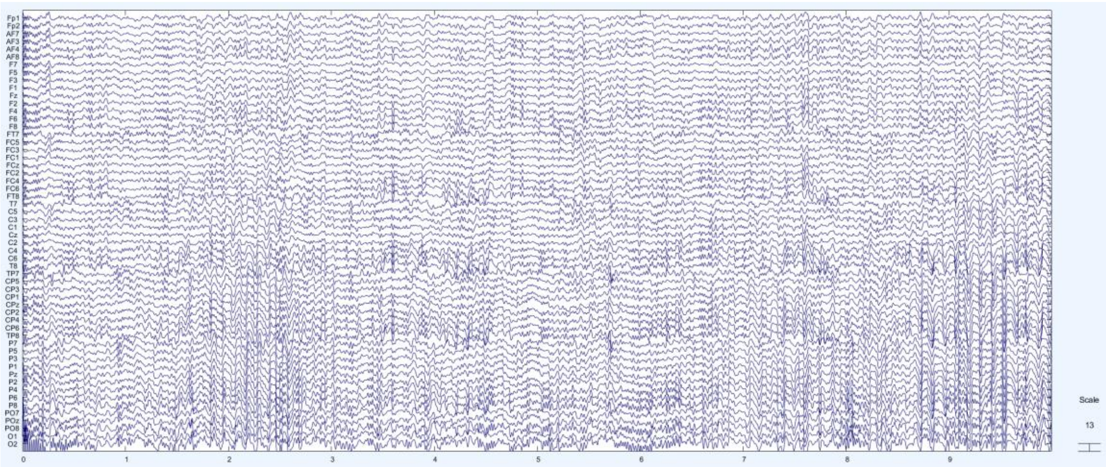
Before deletion



Signal to be removed



After deletion



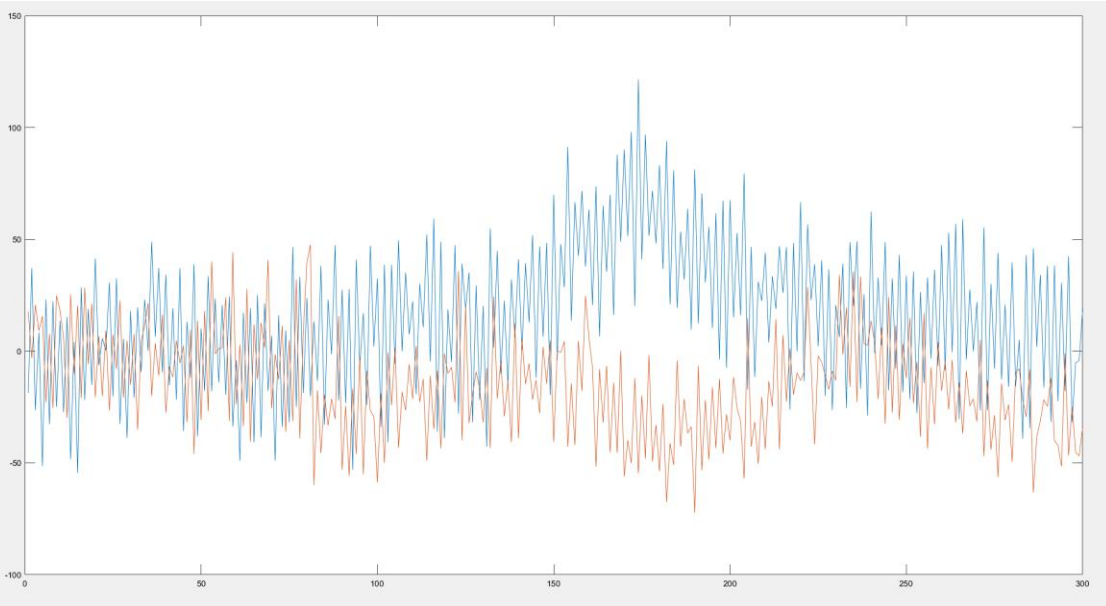
We can now see some pattern we did not see in the chart before deletion.

7. Discuss the effect of band-passing (high-passing) the signal before running ICA.

Obviously, there is a huge difference between using a filter or not. In the first case, since we did not do any preprocessing such as filtering, the data was noisy, and it has low SNR, making it hard to tell what the component was after ICA, so most of the components were categorized as Others. By filtering out the noise, the signal level relatively increase, so that the quality of the signal after ICA will be much better.

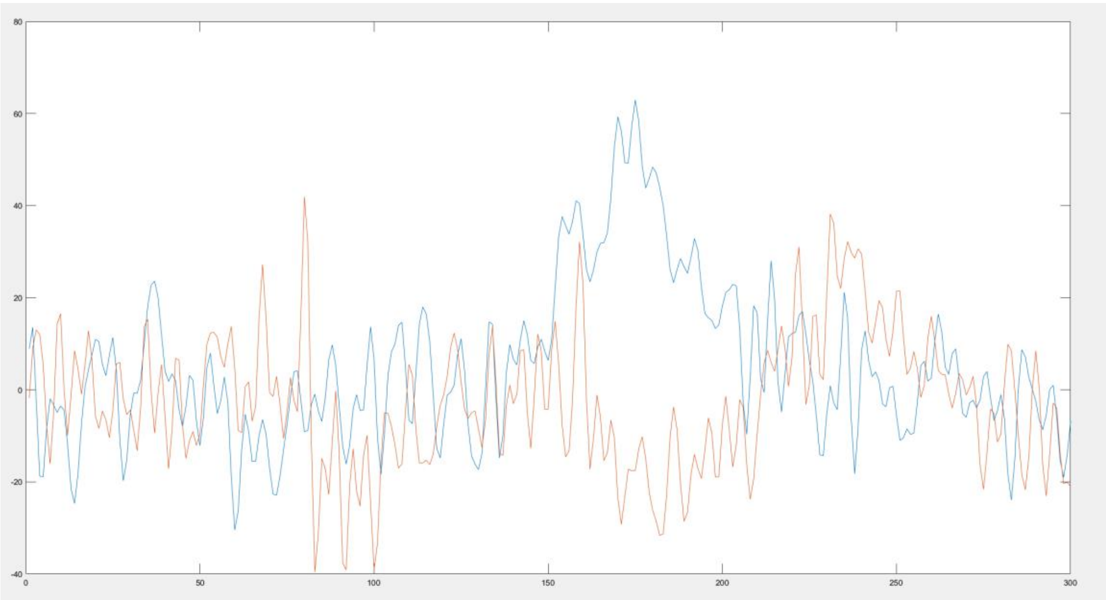
Problem 3

Without any operation



SNR = 2.5577

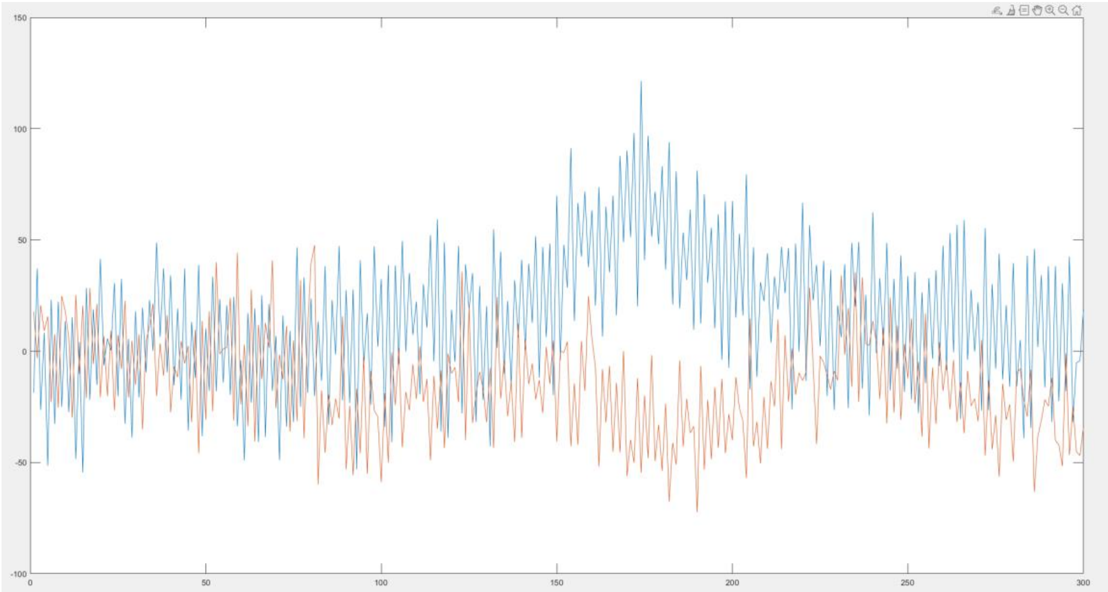
Bandpass only



SNR = 4.8356

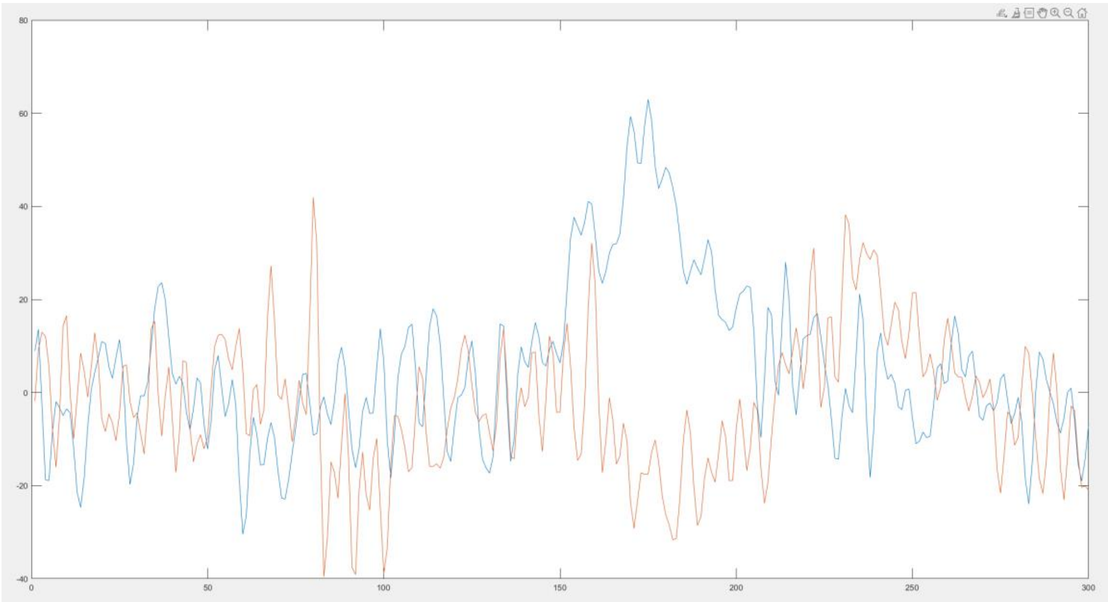
When I was plotting the results by exporting the script and use the EEG.data in it, there seemed to be a problem. The selecting process of rejecting noise in components cannot be exported to the script, and when I typed in the number of the noise components, the result looked very weird. As a result, I can only generate the same result as the two method above.

IC removal only



SNR = 2.5577

Bandpass + IC removal



SNR = 4.8356