EEG Datafile Restructure Toolbox（DRT）

Manual

Yufan Zhang: zyf15816794709@163.com
Li Dong: lidong@uestc.edu.cn
Version: 1.0
Date: 09/06/2020

The Key Laboratory for NeuroInformation of Ministry of Education,
School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu, 610054, China

Catalog

1.Introduction

DRT (EEG Datafile Structure Toolbox) is an EEG file format standardization tool based on MATLAB programming language. Currently, DRT provides MATLAB and Docker versions, supports Windows, Ubuntu and other operating systems. The latest version can be available from the WeBrain website.

2.Preparation

2.1 Installation

Before importing the DRT toolkit into MATLAB, it is necessary to ensure that the MATLAB software is installed correctly. For details of the installation of MATLAB software, please refer to: https://mathworks.com/help/install/ug/install-mathworks-software.html。

Each of steps of importing DRT into MATLAB is described in detail below:

1) Unzip the DRT package and place it in the toolbox directory of MATLAB.
2) In the "HOME" option of MATLAB, click the “Set Path” button (Fig.1) to enter the “Set Path” interface (Fig.2).

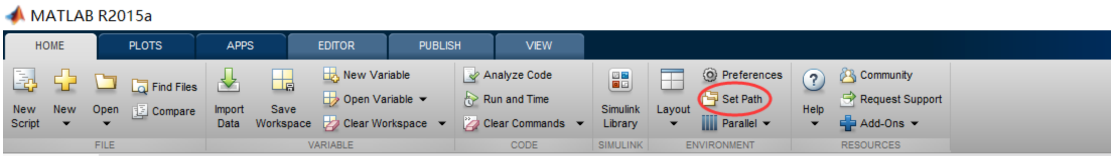

Fig.1 click the “Set Path” button

3) In the “Set Path” interface (Figure 2), click the “Add Folder” button to find the path where the DRT toolkit is located, and then click the “Save” button.

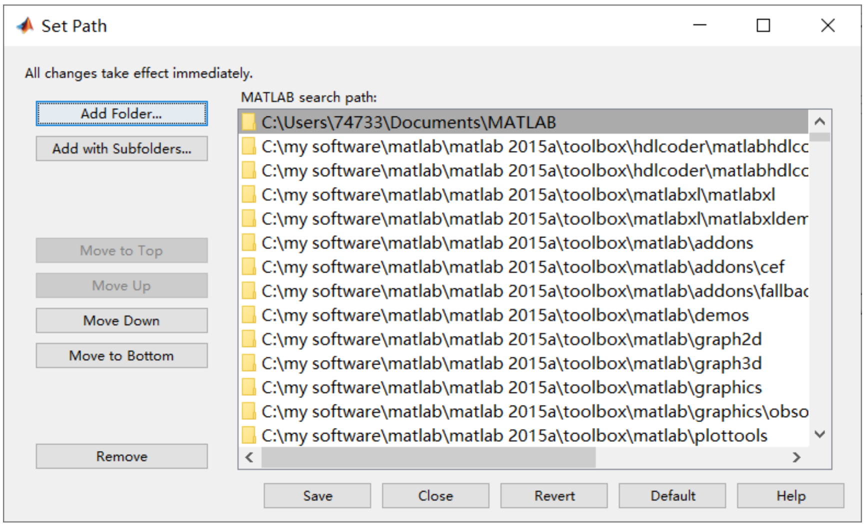

Fig.2 “Set Path” interface

4) In the command line interface of MATLAB, enter the “DRT” command to pop up the DRT interface (Fig.3) , indicating that the DRT is successfully installed.

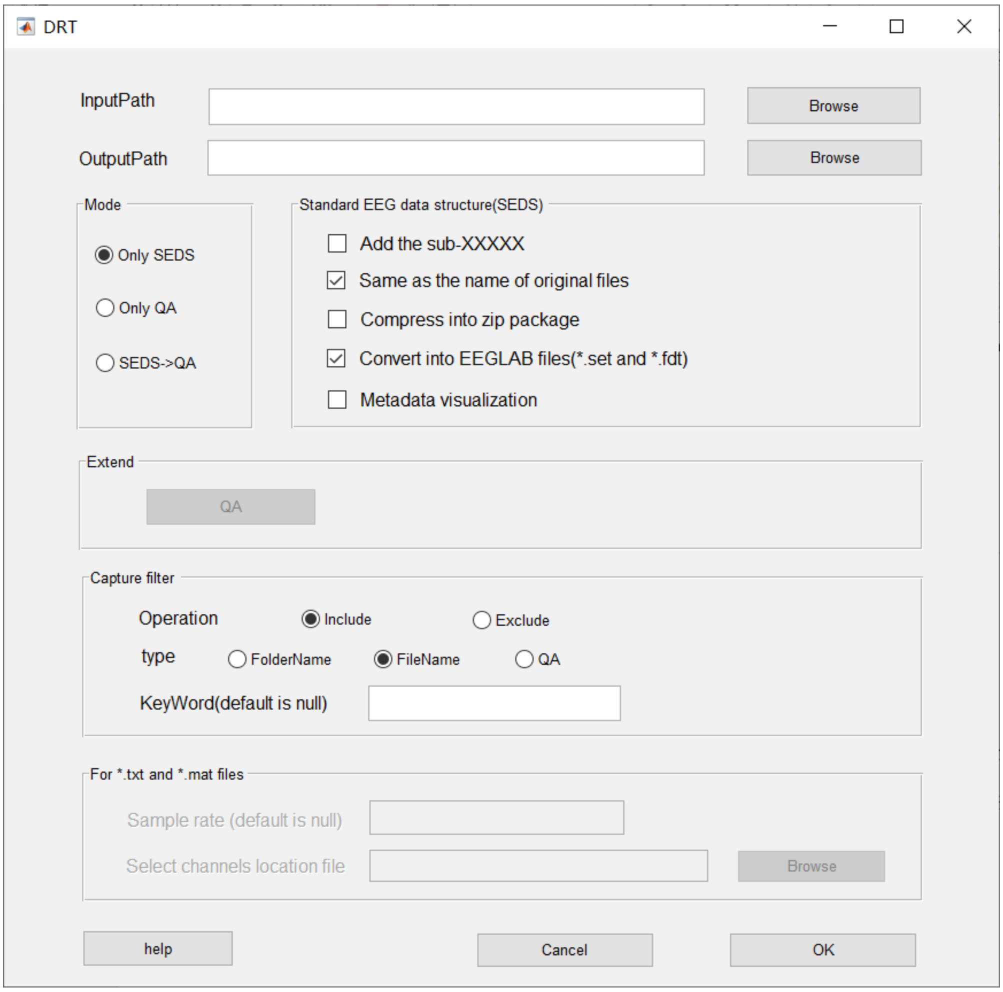

Fig.3 DRT main interface

2.2 Basic operation

In order to facilitate users to use the tool better, a GUI-based usage process is provided, so that users do not need to manually type parameters under the command line interface. The following will demonstrate how to operate:

1） In the main interface of DRT (Fig. 3), click the "Browse" button in the InputPath line to open the input path selection interface (Fig. 4) and select the folder path for storing EEG data files.

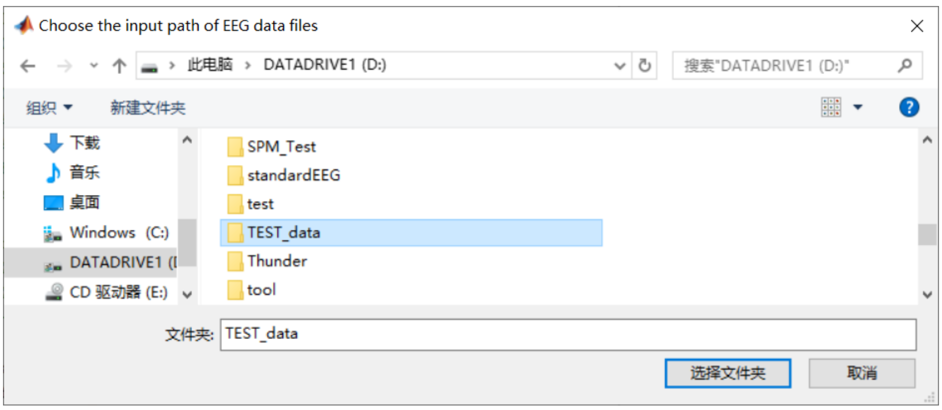

Fig.4 Input path selection interface

2） Similarly, in the main interface of DRT (Fig.3), click the "Browse" button on the OutputPath line, and the output path selection interface will pop up to select the folder path where the result file is stored.

3） Select the function mode. There are three functional modes: “SEDS”, “QA” and “SEDS->QA” in DRT, different implementation modes have different options, see “3.Instruction”.

4） After the function mode is selected, the corresponding parameters need to be selected.

5） After the parameter selection is completed, click the “OK” button, and the program execution log can be seen in the MATLAB command line window. When the words “Success” or “Failed” appear, the program execution is completed.

2.3 Help

The user can place the mouse over any option, and the detailed description of the option will automatically emerge, which is a simple way to help users use this tool better. Figure 5 show we place the mouse over "Convert into EEGLAB files (*. set && *. fdt) options and the description of this option is shown.

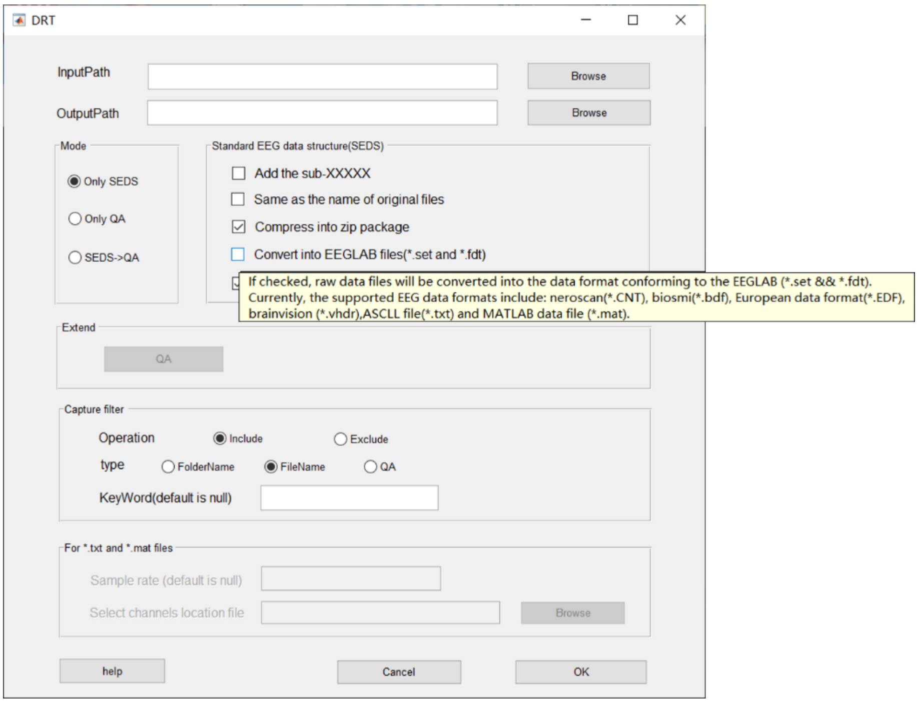

Fig.5 Description of “Convert into EEGLAB files (*.set && *.fdt)” option

Moreover，click "Help" Button in DRT main interface (Figure 3) to get the usage of DRT (Fig.6).

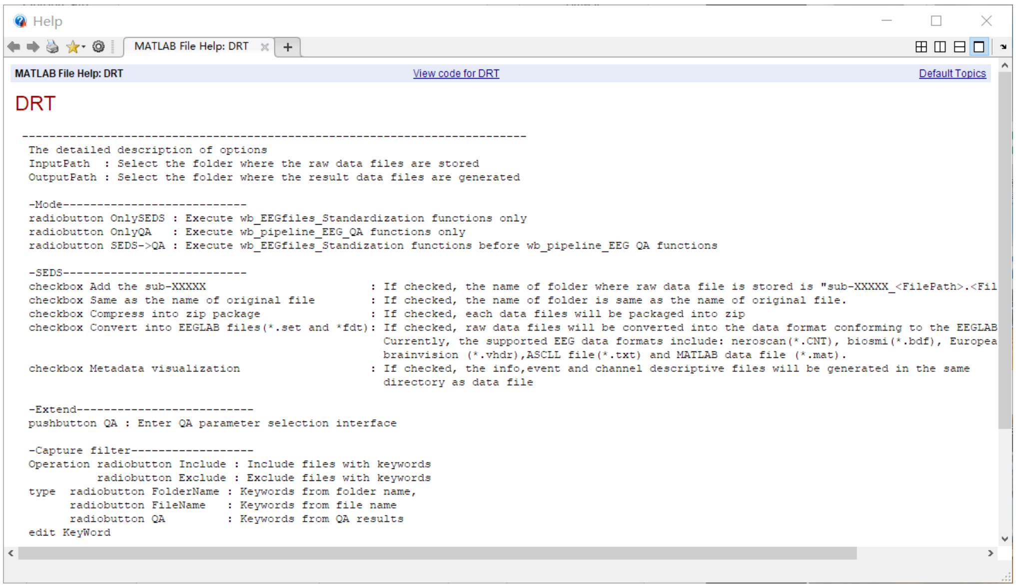

Fig.6 Help of DRT

3. Instructions

There are currently three functional modes: “SEDS”、 “QA” and “SEDS->QA”. Only one mode can be executed at a time.

3.1 SEDS

In “SEDS” mode, the input of EEG data files will be reorganized into standard EEG data structure (SESD). For details of SEDS, please refer to “5.1.1 SEDS”.

Usage of SEDS function are described in the following paragraphs:

1） Select the “SEDS” option in the Mode box of the DRT main interface (Fig.3), and select the corresponding input and output file path. It should be noted that the input path can contain multiple folders, and the currently supported EEG data formats include: Neroscan (*.cnt), Biosmi (*.bdf), European data format (*.edf), and general data format (* .gdf), BrainVision (*.vhdr), EEGLAB (*.set), ASCLL file (*.txt) and MATLAB data file (*.mat).

2） To configure the parameters in the SEDS options (Figure 7), the specific input parameters were as follows:

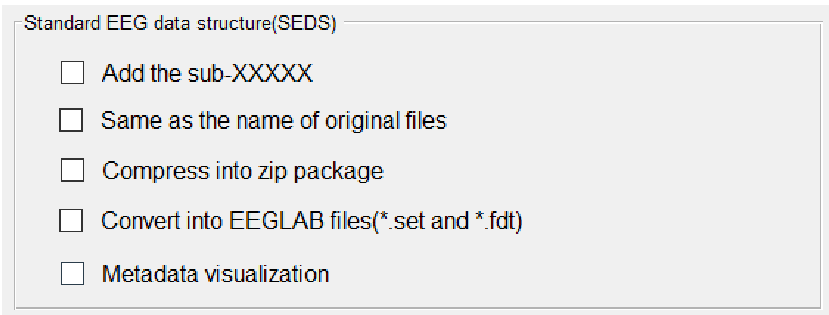

Figure 7 SEDS options box

- “Add the sub-XXXXX” options

If this option is checked, the output folders name will be generated according to the pattern “sub-XXXXX_<FilePath>.<FileName>” and the “sub-XXXXX” will be numbered sequentially from “sub-00001~sub-99999”. If this option is not checked, the output folder name will be generated according to the pattern “<FilePath>.<FileName>”.

- “Same as the name of original files” options

If this option is checked, output folders name is same as the input EEG data file name. The output folder name will be generated according to the pattern “<FilePath>.<FileName>”.

- “Compress into zip package” options

If this option is checked, every output folder will be compressed into a zip package. If it is not checked, no operation is performed.

- “Convert into EEGLAB files(*.set && *.fdt)” options

If this option is checked, different EEG file formats will be converted into EEGLAB file format(*.set && *.fdt) that can be used as the input file of the EEGLAB.

The format of the raw EEG data files currently supported for conversion are Neroscan (*.cnt), Biosmi (*.bdf), European data format (*.edf), general data format (*.gdf), BrainVision (*.vhdr) and EEGLAB(*.set).

- “Metadata visualization” options

If this option is checked, the metadata field of the raw data will be extracted without changing the underlying data, and visualize the metadata information in the form of text and list files. Specifically, a “*info.json” file describing basic data information, a "*channels.csv" file describing electrode information, and a "*event.csv" file describing event information will be generated. If it is not checked, no operation is performed.

In this case, check the “Metadata visualization” option, “Convert into EEGLAB files(*.set && *.fdt) ” option, “Compress into zip package” option, and other options are not checked.

3） After setting the parameters, click “OK” Button, waiting for the program to finish.

4） The word “Success” appears in the MATLAB command line interface, indicating successful execution. Fig.8 shows that the raw EEG data file structure changes. Among them, the left of Fig.8 is the structure of the raw data file, and the right of Fig. 8 is the data file structure after execution.

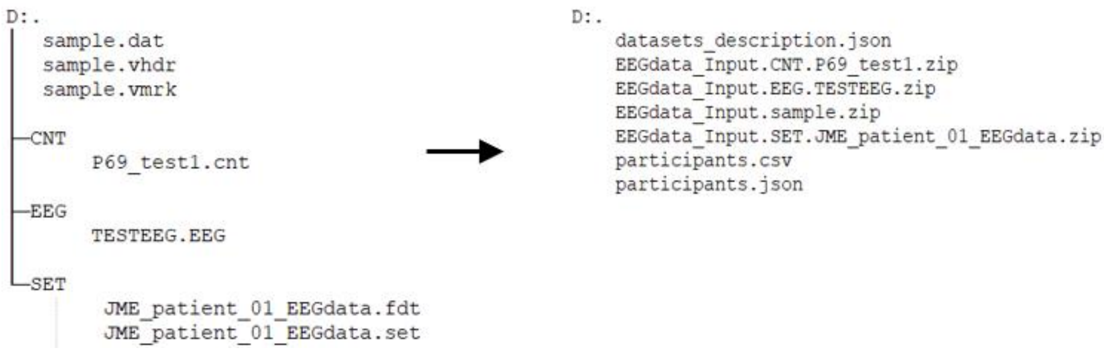

Fig.8 files structure changes

Next, select “EEGdata_Input.P69_test1.zip” to decompress it. Fig.9 show that the raw EEG file in the “*.cnt” data format has been converted into EEGLAB file (“*.set” && “*.fdt”), and three descriptive files are generated.

Fig.9 “EEGdata_Input.P69_test1.zip” document content

3.2 QA

QA is used to evaluate the quality of continuous EEG raw data. The QA integrated in the DRT comes from the WeBrain tool. For specific function details and implementation process refer to 5.1.2. The following will introduce the steps of using QA function in DRT tool:

1） Select the “QA” option in the “Mode” box of the DRT main interface (Fig.3),and then select the corresponding input and output file path. It should be noted that the input EEG data file format must be EEGLAB file format ("*.set" && "*.fdt") and must be packaged as a zip.

2） In the “Extend” box of the DRT main interface (Fig.3), click the “QA” button to enter the parameter setting interface of QA (Fig.10).

3） Set the parameters in the parameter setting interface of QA(Fig.10). The parameters are as follows:

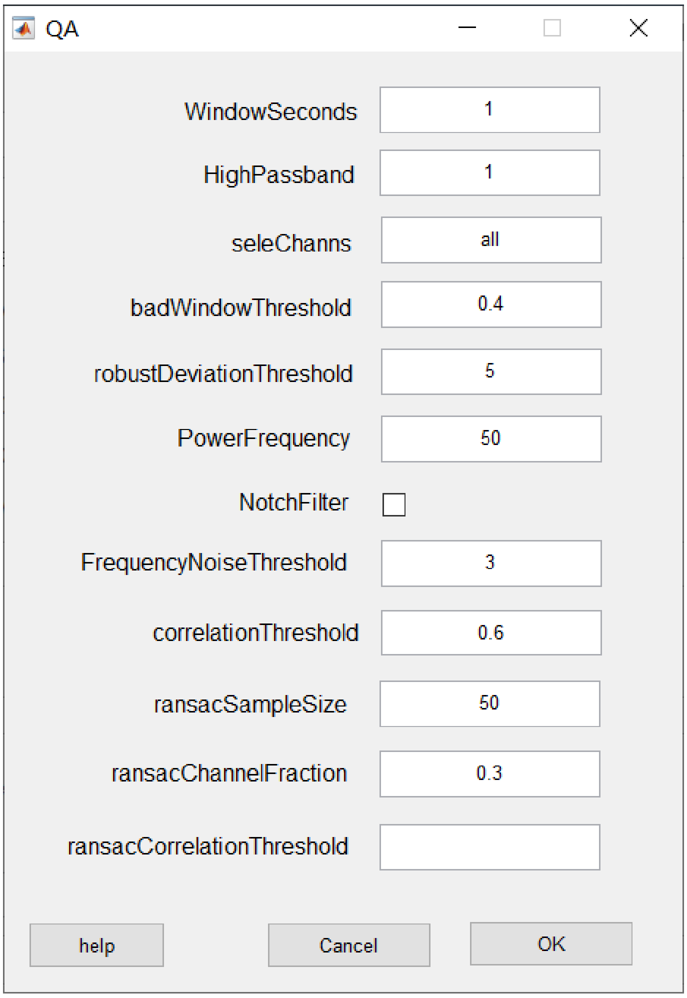

Fig.10 parameter setting interface of QA

WindowSeconds: the window size (in seconds, default = 1 sec.) over which the above methods are conducted.

HighPassband: lower edge of the frequency for high pass filtering. Default is 1 Hz.

seleChanns: number with indices of the selected channels (e.g. ‘[1:4,7:30]’ or ‘all’). Default is ‘all’.

badWindowThreshold: cutoff fraction of bad windows (default = 0.4) for detecting bad channels.

robustDeviationThreshold: Z-score cutoff for robust time deviation in each window (default = 5).

PowerFrequenc: power frequency. Default is 50 Hz (in Chinese). Noting that in USA, power frequency is 60Hz flagNotchFilter: flagNotchFilter = 1: remove 0.5×power frequency noise using notch filtering. Default is off (flagNotchFilter = 0).

FrequencyNoiseThreshold: Z-score cutoff for NSR (signal above power frequency -10Hz). Default is 3. If the z score of estimate of signal above 40 Hz (power frequency -10Hz) to that below 40 Hz above ‘highFrequencyNoiseThreshold’ or absolute NSR exceeds 0.5, the small window is considered to be bad.

correlationThreshold: maximal correlation below which window is bad (range is (0,1), default = 0.6). If the maximum correlation of the window of a channel to the other channels falls below ‘correlationThreshold’, the window is considered bad.

ransacSampleSize: samples for computing RANSAC (default = 50).

ransacChannelFraction: fraction of channels for robust reconstruction (default = 0.3).

ransacCorrelationThreshold: cutoff correlation for abnormal wrt neighbors(default = [] | --> not performed). Default is 0.6.

srate: sampling rate of EEG data. It can be automatically detected in EEG data. But for ASCII/Float .txt File or MATLAB .mat File, user should fill the sampling rate by hand. Default is ‘[]’.

In this case, the default parameters are used.

4）After completing the parameters, click the “OK” button to return to the DRT main interface (Fig.3).

5）In the DRT main interface (Fig.3), click the "OK" button and wait for the execution to complete.

6）After successful execution, in the output path folder, each subject will generate a mat file, which contains all quality assessment results and parameters (saved as results_QA_*.mat file) and will also generate a file named QA_table. The mat file contains the quality assessment indicators of all the subjects (including the successful and skipped subjects) as shown in Fig.11.

results_QA_erp-epoched.mat	Microsoft Acces...	5 KB
results_QA_JME_patient_01_EEGdata....	Microsoft Acces...	14 KB
TaskID-1_QA_table.mat	Microsoft Acces...	2 KB

Fig.11 Result files

In the MATLAB development environment, load the “TaskID-1_QA_table.mat” file, the quality assessment results of all files are shown as Fig.12.

QA_table

2x16 table

	1 SubNumber	2 filename	3 ONS	4 OHA	5 OFN	6 OLC	7 OLRC	8 badChannels	9 NBC	10 OBC	11 OBClus	12 allMAV	13 badMAV	14 goodMAV	15 ODQ	16 DataQualityRating
	1	'JME_pa...	0.0056	0.0418	0.0158	0.0917	[]	[28,57,63,64...	6	0.0909	0.0943	7.9356	31.4679	5.2336	89.6813	'B'
	2	'erp-ep...	0	0	0	0.0317	[]	[47,48]	2	0.0317	Inf	4.3465	4.5441	4.3401	96.8254	'A'

Fig.12 “TaskID-1_QA_table.mat” document content

The following is the parameter description of the “results_QA_*.mat” and “QA_table.mat files:

results_QA.ONS: Overall ratio of No Signal windows;

results_QA.OHA: Overall ratio of windows of High Amplitudes;

results_QA.OFN: Overall ratio of windows of high Frequency and power frequency Noise;

results_QA.OLC: Overall ratio of windows of Low Correlation;

results_QA.OLRC: Overall ratio of windows of Low RANSAC Correlation (optional);

results_QA.badChannels: Bad channels detected by the QA tool;

results_QA.NBC: No. of Bad Channels;

results_QA.OBC: Overall ratio of Bad Channels;
results_QA.OBClus: Overall ratio of Bad Clusters;
results_QA.ODQ: Overall Data Quality: overall ratio of windows of good data;
results_QA.DataQualityRating: Overall Data Quality Rating

Level A: ODQ >= 90;
Level B: ODQ >= 80 && ODQ < 90;
Level C: ODQ >= 60 && ODQ < 80;
Level D: ODQ < 60;

results_QA.allMAV: mean absolute value of all windows;
results_QA.badMAV: mean absolute value of bad windows;
results_QA.goodMAV: mean absolute value of good windows;
results_QA.NoSignalMask: a mask of windows with no signals (with dimension channels × windows);
results_QA.AmpliChannelMask: a mask of windows with high amplitudes (with dimension channels ×windows);
results_QA.FrequencyNoiseMask: a mask of windows with high frequency (and power frequency, if applicable) noise (with dimension channels ×windows);
results_QA.LowCorrelationMask: a mask of windows with low correlations (with dimension channels ×windows);
results_QA.RansacBadWindowMask: a mask of windows with RANSAC low correlations (with dimension channels ×windows);
results_QA.OverallBadMask: a mask of windows with overall bad signals (with dimension channels ×windows);
results_QA.fractionBadWindows: fractions of bad windows for each channel (with dimension channels ×1);
results_QA.badChannelsFromAll: logical value of bad channels from all methods (with dimension channels ×1).

results_QA.parameters.srate: sampling rate;
results_QA.parameters.WindowSeconds: window size in seconds (default = 1 sec);
results_QA.parameters.HighPassband: lower edge of the frequency for high pass filtering, Hz;
results_QA.parameters.selechanns: number with indices of the selected channels (e.g. [1:4,7:30] or 'all').Default is 'all';
results_QA.parameters.badWindowThreshold: cutoff fraction of bad windows;
results_QA.parameters.PowerFrequency: power frequency. Default is 50 Hz (in Chinese). Noting that in USA, power frequency is 60Hz;
results_QA.parameters.robustDeviationThreshold: Z-score cutoff for robust channel deviation;
results_QA.parameters.FrequencyNoiseThreshold: Z-score cutoff for nosie-to-signal ratio (signal above 40 Hz);
results_QA.parameters.correlationThreshold: maximal correlation below which window is bad (range is (0,1));
results_QA.parameters.chanlocsflag: flag of channel locations. if chanlocsflag = 1: have channel locations;
results_QA.parameters.chanlocsXYZ: xyz coordinates of selected channels;

results_QA.parameters.chanlocs: channel locations of selected channels;
results_QA.parameters.ransacSampleSize: samples for computing RANSAC (default = 50);
results_QA.parameters.ransacChannelFraction: fraction of channels for robust reconstruction (default = 0.3);
results_QA.parameters.ransacCorrelationThreshold: cutoff correlation for abnormal wrt neighbors(default = [] | --> not performed).

Note:

- Assumptions of QA tool:
  -The signal is a structure of continuous data with data and sampling rate at least.
  -No segments of the EEG data have been removed.
- Noting that quality assessing EEG raw data would NOT change the raw data.
- If channel locations are not contained in EEG data or selected channels do not contain locations, the RANSAC correlation is invalid.
- Noting that if the sampling rate is below 2×power frequency, the step of detecting high or power frequency noises will be skipped.

3.3 SEDS->QA

In "SEDS->QA" mode, the raw EEG data files will first be reorganized into a standard EEG data structure, and then data quality of these files will be assessed.

The following will introduce the use steps of "SEDS->QA" function in detail:

1） Select the "SEDS->QA" option in the Mode box of the DRT main interface (Fig.3), and select the input and output file path. It should be noted that if the format of input data files are Neroscan ("*.CNT"), Biosmi ("*.bdf"), European data format ("*.EDF"), BrainVision (*.vhdr), ASCLL file ("*. txt") and MATLAB data files ("*.mat"), the "Convert into EEGLAB files(*.set && *.fdt)" option need to be checked in the SEDS box (Fig.7). If unchecked, the format of the input data file must be the EEGLAB file format ("*.set" && "*.fdt").
2） Set the parameters in the SEDS box (Fig. 7).
3） In the Extend box, click the "QA" button to pop up the "QA" parameter setting interface (Fig. 10).
4） In the "QA" parameter setting interface (Fig.10), complete the parameter modification, and then click the "OK" button.
5） In the DRT main interface (Fig.3), click the "OK" button and wait for the execution to complete.
6） After successful execution, folders named "SEDS" and "QA" will be generated in the output path folder, and the execution results of SEDS and QA will be stored in the corresponding folders.

3.4 Capture Filter

Capture Filter was integrated in DRT, which can filter out specified EEG files based on keywords. It is mainly used to filter the specific files needed from a large number of EEG files to facilitate subsequent processing and Analysis, saving time for manual screening.

Two operations for file filtering are provided in Capture Filter. When the "Include" option is selected, the data files in line with keyword will be filtered out, and when the "Exclude" option is selected, the files that do not meet keyword will be filtered out.

Capture Filter filters EEG data files based on one of the following three types of keyword:

(1) Folder Name

If “Folder Name” is checked, it is necessary to fill in the keywords contained in the folder name data name of the first-level subdirectory under the input file directory. At present, Capture Filter can only filter out the folders of the first-level sub-directory under the input file directory.

(2) File Name

If “File Name” is checked, the keyword contained in the raw data file name need to be filled.

(3) QA

Only in “SEDS->QA” mode can “QA” be checked. If “QA” is checked, QA results (A, B, C, D) need to be filled in the keyword edit box. EEG data files in line with keyword will be filtered out by Capture Filter according to the QA evaluation results of the raw data. These files will be stored in the folder named "CaptureSEDS" under the output path.

It is worth noting that the keywords are not case sensitive.

4.Attention

In “SEDS” and “SEDS->QA” modes, if the raw EEG data in “*.txt” and “*.mat” formats need to converted into EEGLAB format (*set && *fdt), you need In the interface shown in Figure 13, the sampling rate need to fill and the channel file requires to be import into the interface shown in Fig.13.

Fig.13:

In “QA” mode, if input EEG data files are “*.txt” and “*.mat” formats, only the sampling rate need to be fill in the interface shown in Fig.13.

5.Appendix

5.1 Principle of main methods

5.1.1 SEDS

Fig.14 shows the standard EEG data structure (SEDS) using the EEGLAB file (*.set && *.fdt) as an example. Each zip package represents one experimental data for each subject. The following files are added to facilitate data management and improve data reusability and readability.

1) “datasets_info.json” is used to fill in the description information of some data sets, including the data source (whether the data is downloaded from a public database or collected from experiments). If it is downloaded from a public database, it is recommended to fill in the address link for downloading. In addition, it is recommended to fill in relevant information such as experimental tasks, data acquisition configuration parameters and so on.
2) “subjects_info.csv” is a list file for filling in the information of the subjects. It can contain the relevant information such as the gender and age of the subject, which needs to be filled in manually by the user.

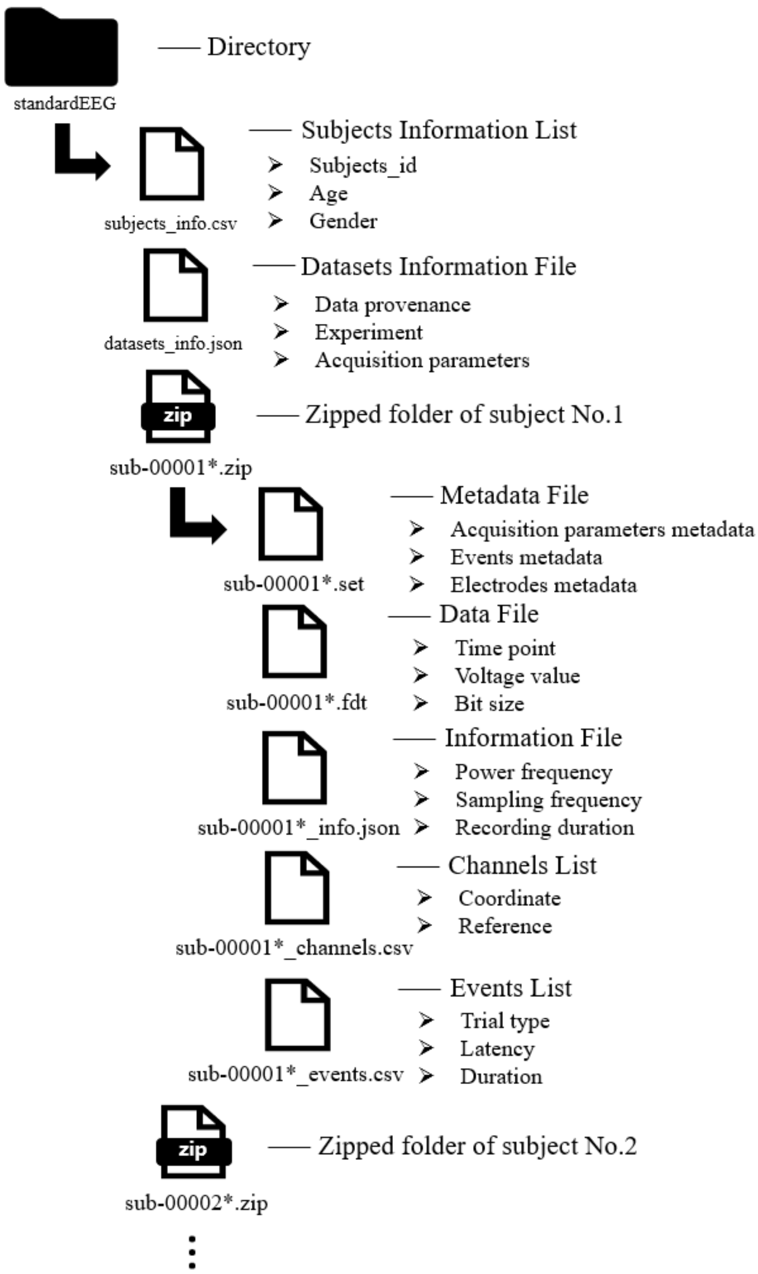

Fig.14: An example for SEDS. The information of all subjects (“subject.csv”), datasets description file (“description.json”) and data files for each subject (“*.zip”) are in the same directory. Data file, metadata file and three descriptive files (“*_info.json”, “*_electrodes.csv” and “*_events.csv”) in each zip package. It should be noted that the data file and the metadata file are not necessarily only two files. It depends on the raw data format.

Although the EEG data files on open database are shared, it is difficult for other researchers to understand them directly because of lacking of explanation. If researchers want to access to information in data files, they must download from website and load into analysis tools to view metadata. To simplify these steps, here we advise to use the descriptive files and lists in each zip package to describe underlying metadata. The following files should be included in each zip package:

1) “sub-XXXXX*_info.json” contains information related to EEG acquisition equipment.
2) “sub-XXXXX*_channels.csv” records spatial coordinates of all electrodes，reference

and so on.

3) “sub-XXXXX*_events.csv” records all stimuli, event type and so on.

The detailed properties contained in these files can be found in the specification document of SEDS. Users are free to choose from the pre-defined properties and only include the properties they need to describe their datasets. The only properties that are required are the title properties. It should be noted that the above description files are not necessary files. They are just for people other than the data uploader to better understand the data structure, configure parameters and provide details, making the data more readable and advantageous for data sharing. This is also one of the purposes of SEDS. In order to simplify the generation steps of these description files, we implement the function that can automatically generate the above description files, which will be described below.

One of the difficulties of data standardization is that the original file may have multiple file directories, and the final result files may come from multiple folders. After large-scale EEG data processing and analysis, sheer number of files make it difficult for researchers to trace source file. Therefore, for the “sub-XXXXX*” above, we recommend using “sub-XXXXX_< Path >.< filename >” instead. <Path> indicates the storage path of the original file. Replacing the slash (/) in path with dot ‘.’ symbol and <filename> indicates the original file name. The advantage of such naming method is that the original file can be traced according to the name of folder.

It is also worth noting, that a participant may have done multiple experiments corresponding to multiple data files. The digital (e.g. sub-00001) on file name only represents the number of data files, not the number of participants.

5.1.2 QA

WB_EEG_QA is a stable tool to realize quality assessment (QA) of a continuous EEG raw data (e.g, resting-state EEG data). The bad data in small windows of each channel could be detected by kinds of 4 methods, and a number of indices related to the data quality will be calculated. Meanwhile, the overall data quality rating will be also provided, including levels of A, B, C, D (corresponding to perfect, good, poor, bad). The QA consists of (Fig. 15):

（1）A continuous EEG data of each channel will be high pass filtered and then segmented as small windows;

（2）Detecting constant or NaN/Inf signals in each window (Method 1);

（3）Detecting unusually high or low amplitude using robust standard deviation across time points in each window (Method 2);

（4）Detecting high or power frequency noises in each window by calculating the noise-to-signal ratio (NSR) based on Christian Kothe's method (Method 3);

（5）Detecting low correlations with other channels in each window using Pearson correlation (default) or RANSAC correlation (Method 4);

（6）Calculating a number of indices relative to the data quality and rating the EEG raw data.

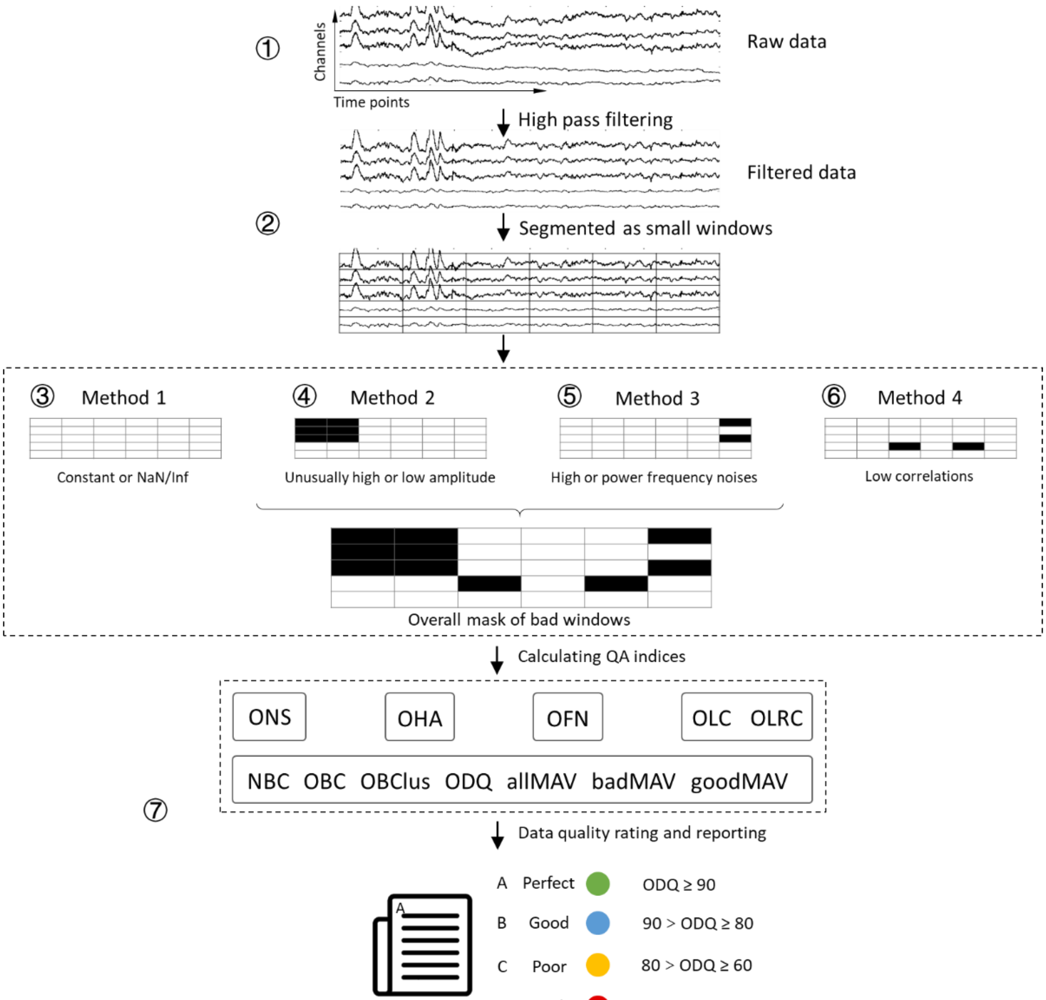

Fig. 15: Pipeline of quality assessment of continuous EEG raw data. (1) Raw EEG data with artifacts such as eye blink, eye movement etc. (2) The continuous EEG data of each channel will be high pass filtered and then segmented as small windows. Here 'WindowSeconds' is the window size (e.g. 1 sec.) over which the following methods are conducted. (3) Detecting constant or NaN/Inf signals in each window (Method 1). (4) Detecting unusually high or low amplitude using robust standard deviation across time points in each window (Method 2). If the z score of robust time deviation falls below 'robustDeviationThreshold' or the absolute amplitude exceeds 200 microvolts (μV), the small window is considered to be bad. (5) Detecting high or power frequency noises in each window by calculating the noise-to-signal ratio based on Christian Kothe's method (Method 3) (clean_rawdata0.32 https://sccn.ucsd.edu/wiki/Artifact_Subspace_Reconstruction_(ASR)). If the z score of estimate of signal above 40 Hz (power frequency -10Hz) to that below 40 Hz above 'highFrequencyNoiseThreshold' or absolute NSR exceeds 0.5, the small window is considered to be bad. Noting that if the sampling rate is below 2×power frequency, this step will be skipped. (6) Detecting low correlations with other channels in each window using Pearson correlation (default) or RANSAC correlation (Method 4). For Pearson correlation, if the maximum correlation of the window of a channel to the other channels falls below 'correlationThreshold', the window is considered bad. For RANSAC correlation (Bigdely-Shamlo et al., 2015), each window of a channel is predicted using RANSAC interpolation based on a RANSAC fraction of the channels. If the correlation of the prediction to the actual behavior falls below 'ransacCorrelationThreshold' or

calculation is too long, the window is marked as bad. The time cost of this method is high, and the channel locations are required. The RANSAC correlation is optional and default is not performed.
(7) Calculating a number of indices relative to the data quality and rating the EEG raw data.

5.2 Copyright