Documentation of the Boron isotope reduction app

**1 Introduction**

A brief explanation what the program is for and what it does

<Boron LA-MC-ICPMS isotopic data processing website>

This program is capable of:

--Read multi-csv data files

--Outlier rejection

--Background correction

--Intra-sequence Instrumental drift correction

--Matrix and spot size dependent B concentration offset correction

--Combine calculation results, laser parameters and trace elements results

--Ready to use final data table

data/data to test/1. data folder20221129-214242/001\_A.exp

**2 Input File Requirements**

A brief description how the input file should look like, how data files should be labelled, etc. Screenshots would be helpful, and we should have a demo dataset in the repo

The raw data file should be a folder composed by .exp or .csv files and the name of these files should in a format of sequence number and sample type, e.g. ‘001-A’, ‘002-B’.

🡪Each file should at least have 4 columns for calculation: ‘9.9’, ‘10B’, ’10.2’, ‘11B’. Titles of data columns are in the 23th row. So, data starts in the 24th row and ends roughly in the 200th row, depends on the number of measuring cycles.

🡪Data in each column is sorted by time or cycle, the former part is noise, then, signal. Data increases rapidly from noise to signal (fig.).



1. **Description of the Python Code**
2. Explain the individual function

-> Include here explanations of what the functions do to the data, e.g., why the regression, why higher orders: or – why the subtraction of the backgrounds, what two backgrounds exist: the ‘normal’ one, and one from an unknown source

---> **def dirFiles(fDir, ending)**: In the raw data folder, there are hundreds of datafiles in different type. dirFiles helps to read all names of datafiles in working folder-fDir, ending is related with the type of datafile which we are gonna work on, can be ‘exp’ or ‘csv’.

--->**def selSmpType(dataFiles)**:getthe number in the name of datafile. The number represents its sequence number, which will be used for Instrumental drift correction.

--->**def outlierCorrection(data, factorSD):** outlier rejection of data, data is out of factorSD times of standard deviation will be taken as outliers.

--->**def bacground\_sub(folder, select\_line, factorSD, factor\_B11):** background subtraction. Folder contains all datafiles that need to be calculated. For 10B data, values above select\_line will be considered as signal, and values below is taken as noise. The same noise and signal interval will apply to ‘9.9’, ’10.2’ and ‘11B’. So, noise in different cups will be subtracted. Then, the average value of ‘9.9’ and ’10.2’ will be used for additional background subtraction for each row of 10B, factor\_B11 times the average value of ‘9.9’ and ’10.2’ will be taken as additional background subtraction for 11B. The last step is calculating the average of 11B/10B, results is out of factorSD times of standard deviation will be taken as outliers. the average of 11B/10B, standard deviation and name of datafile will be returned.

--->**def polynomFit(inp, \*args):** used for regression function.

--->**def regression(x, y, ref\_stand, order, listname):** get the correction of the Intra-sequence Instrumental drift. x is the list of standard sample’s sequence number, and y is the calculated average 11B/10B of standard sample from **bacground\_sub** function. ref\_stand is the reference 11B/10B value of standard. Order is the level of poly regression function. Listname is the sequence number list of all datafiles. O

--->**def prepare\_trace(datafile)**: delete useless information from all columns titles and change the data format to float.

1. Explain the main body of the code

---> 1. Choose the folder which have all datafiles. Use **bacground\_sub** function to correct the background for all datafiles;

---> 2. Choose one sample as standard sample to calculate 11B/10B, and 11B Intra-sequence Instrumental drift with **regression** function. Correction factor from each datafile will be returned.

---> 3. Multiply 11B/10B correction factor and calculate Boron isotopic values and erros.

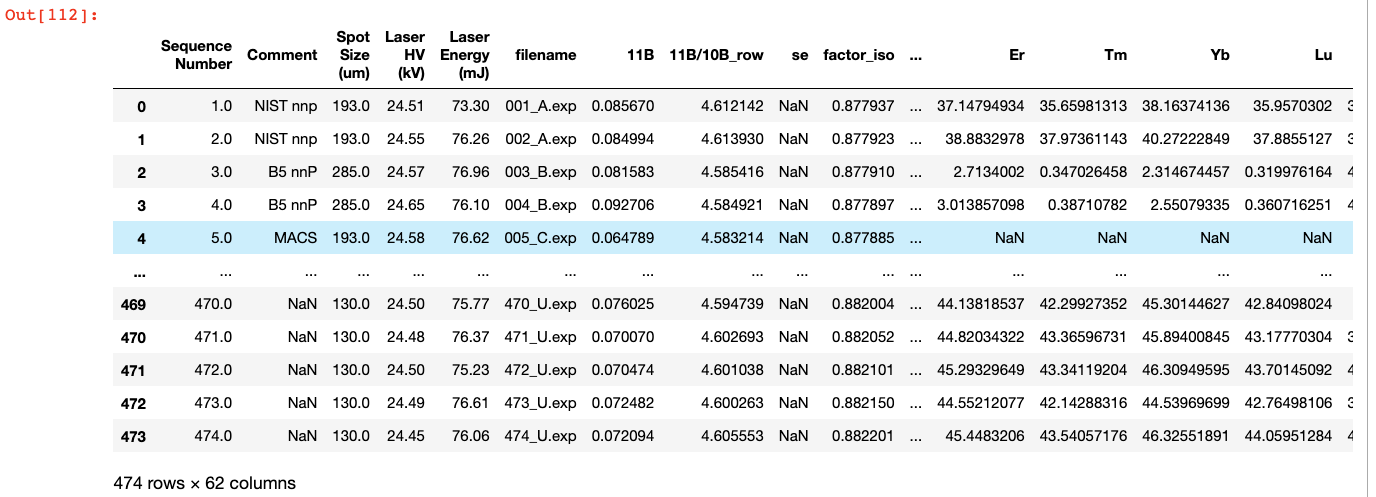
---> 4. Mapping calculate Boron isotopic values and laser parameters.

---> 5. Multiply 11B correction factor and the volume factor to 11B signal for B concentrations.

---> 6. Use **prepare\_trace** function to prepare trace elements data from Element XR and merged with upper results.

**4 Explain the Output**

What exactly is the output, likely best with screen shots.



-->’Sequence Number’ column: the number of datafile in all sequence.

-->The ‘Comment’ column: sample name, labelled by yourself when measuring.

--> ‘Spot size (um)’, ‘Laser HV (kV)’, ‘Laser Energy (mJ)’: useful information selected from laser parameters.

-->The ‘filename’ column: name of datafile.

-->from ‘11B’ to ‘factor\_iso’: all results from Neptune. ‘[B]\_corrected’ is calculated B concentrations from 11B. ‘δ11B’ and ‘δ11B\_se’column are calculated isotope results and erros.

-->from ‘Li’, ‘B’ to ‘U’ are all trace element results from Element XR.

(the following is copied from what was a ‘text’ file.)

1. csv files are changed from original .exp file

2. data automatically from machine can be found in 'data/original data type'.