Appendix A

EEMlab v026 tutorial

Abstract

This tutorial illustrates the use of the EEMlab GUI toghether with the Decomposition Routines for Excitation Emission Matrices (drEEM) toolbox for correcting, preprocessing and the final analysis and validation of a fluorescence experiment. In this case, the example dataset is the same used in the reference document [Murphy et al., 2013, *J. Mar. Syst.* 111-112, 157-166]. This document is referred to the EEMlab v026. The example presented next has been processed with EEMlab run on MATLAB Version 9.8.0 (R2020a) without any reported problem

Index

1 Foreword

This document contains a tutorial on using the EEMlab GUI for implementing spectral correction, preprocessing and PARAFAC analysis of fluorescence Excitation Emission Matrices (EEMs). It is a version of the appendix published in the journal Analytical Methods [Murphy et al., 2013, *Anal Methods*, 5, 6557-6566] (from now, the reference document) for illustrating the use of the drEEM toolbox. This version is adapted to the use of EEMlab on the mentioned dataset.

This tutorial acompanies EEMlab v011. It is expected that periodic revisions will uptade the EEMlab software. Visit the EEMlab Project website to check for updates and bug fixes [Micó, 2017, online].

2 Setting up MATLAB to run with EEMlab

MATLAB files with '.m' extension are termed mfiles. In this document, data files, mfiles and folder names are shown in *italics*. Any sequence of instructions to be selected by the user from the menus in the GUI is written in `courier`. To install the application and set up the MATLAB environment for running it, do the following steps:

- download the last version of the EEMlab bundle (GUI + drEEM processing core + interface functions) available in the EEMlab's project site pabmitor.webs.upv.es/eemlab. Remember that a minimized version of the drEEM toolbox (EEMlab's processing core) is ALREADY included in this EEMlab bundle. The full drEEM toolbox is publicly available in http://www.models.life.ku.dk/drEEM
- ask for the original drEEM example dataset files to the authors (in http://www.models.life.ku.dk/drEEM) or download them from the EEMlab's project site
- in addition, the user has the possibility of download the EEMlab's output results (raw, corrected, preprocessed and splitted dataset and also the correspondent models, all available in the EEMlab's project site). This is so because, as you will check throughout this tutorial, to model the dataset can be a very long processing, and not always getting the best results. In this point be aware of the big size of this file (2.2 GB)
- decompress the EEMlab's bundle ZIP file into a user's directory (we recommend *.\Documents\MATLAB*). This path will be referred as your EEMlab's installation path (*EEMLABAPP*)
- decompress the example dataset ZIP file into a user data directory (we recommend *.\Documents\EEMlab\dset*) This path will be referred as your EEMlab's dataset path (*EEMLABDSET*). Then, our example dataset is now located in *.\Documents\EEMlab\dset\dreem_full*
- (if downloaded) decompress the dataset results ZIP file into your *EEMLABDSET* directory. Then, the results for the example dataset are now located in *.\Documents\EEMlab\dset\dreem_full\OutputFiles*
- start MATLAB and include your EEMlab's installation directory (and subdirectories) into the MATLAB list of paths
- type `EEMlab` in the MATLAB's console and the GUI starts working

3 EEMlab overview

The EEMlab is a GUI developed for the processing and analysis of fluorescence experiments. The menus in EEMlab are organized as follows:

- EEMlab: with all the references and the GUI help

- Load: in this section the user define all the files that are used for the experiment. Once the experiment is configured, the user can load all the files
- Dataset: this section includes the dataset correction and also the preprocessing stage (resizing, removal of scattering, outlier detection and removal, sample correction and normalization)
- Models: in this section the user can do a fast exploratory PARAFAC analysis, a model refinement and the final split half analysis and model validation
- Plots: in were are included all the tools for plotting the samples of the dataset, the calculated PARAFAC models and all re results referred to the models
- Experiment: to load and save full experiments in MATLAB format and also manage the log file

4 Creating a corrected EEM dataset from Raw data files

In this section we load and correct an EEM dataset according to the standarized procedures described in [Murphy et al., 2010, *Environ Sci Technol*, 44, 9405-9412]. As the number and significance of the files can be a little bit messy, in here you are a standard proposal about how to organize experimental files in folders, in order to facilitate the further processing:

- */dsXX/*: this is the HOME directory of the `XX` dataset, including the sample log file
- */dsXX/Abs*: absorbance files for IFE correction
- */dsXX/BlankEEMs*: blank EEMs for blank subtraction
- */dsXX/BlankEEMs/RamanYY*: water Raman scans for the blank files acquired at an excitation wavelength of `YY` nm
- */dsXX/CorrectionFiles/Emcorr*: emission correction scans. This directory remains empty for devices with automatic spectral correction
- */dsXX/CorrectionFiles/Excorr*: excitation correction scans. This directory remains empty for devices with automatic spectral correction
- */dsXX/EEMs*: excitation-emission matrices of acquired samples
- */dsXX/EEMs/RamanYY*: water Raman scans for EEMs acquired at an excitation wavelength of `YY` nm
- */dsXX/QS/RamanYY*: water Raman scans for blank (acid) in QS series acquired at an excitation wavelength of `YY` nm
- */dsXX/OutputFiles*: this directory is EEMlab's made by default. Here the user can store EEM experiments, at any processing stage, in matlab format

4.1 About the dataset

The dataset that we take to illustrate the use of the EEMlab GUI is the same that illustrates how to use the drEEM's toolbox for fluorimetry experiments. Users can ask for the dataset source files directly to the authors in [Murphy et al., 2014, online]. This demo dataset contains measurements made during four surveys of San Francisco Bay that took place in spring, summer, autumn and winter 2006 [Murphy et al., 2013, *J. Mar. Syst.* 111-112, 157-166]. The list of all the dataset components, factors and relationships is fully described in the correspondent sample log metadata file (*SampleLog_PortSurveyDemo*). The dataset includes the following files:

- Sample log: 2 files (CSV and XLS formats, just one of them is needed)
- Fluorescence EEMs for samples: 224 files
- Water Raman scans for the EEMs: 31 files
- Corrections for the excitation spectrum: 8 files

- Corrections for the emission spectrum: 2 files
- Blanks for EEMs: 20 files
- Water Raman scans for blanks: 0 files
- Water Raman scans for quinine sulfate (QS) dilution series: 3 files
- Absorbances: 171 files

Other information as the slope for the QS dilution series, the dilution factor or the different samples' labels is included in the sample log. The origin, acquisition procedure, and descriptions of the samples and files are detailed in the original referred paper.

4.2 Getting started

As it is described with the use of the drEEM toolbox, to create a fully corrected EEM dataset from raw data files we will have to: (i) import the rawdata files, (ii) reconcile the various datasets and (iii) correct the EEMs. Next, we describe the use of the EEMlab GUI in order to do all this processing, avoiding reserchers to directly use the MATLAB programming language for coding.

4.3 Import the raw data files

The EEMlab GUI is not fault tolerant in reference to missing or extra files defined in the experiment. Then and in order to ensure the correct processing, the first step consists on fixing all the information between the sample log metadata file (that describes the experiment) and the GUI itself. From the EEMlab GUI, select:

Load → Configuration

The experiment configuration assistant is displayed in order to define which are the files and parameters to include in the experiment. This oblies the researcher to check all the data included for the experiment in the EEMlab's dataset directory. The minimum information fields required for one fluorimetry experiment to run are displayed in **bold**. According to the files, the excitation wavelength and the acquisition devices used in the example dataset, we configure:

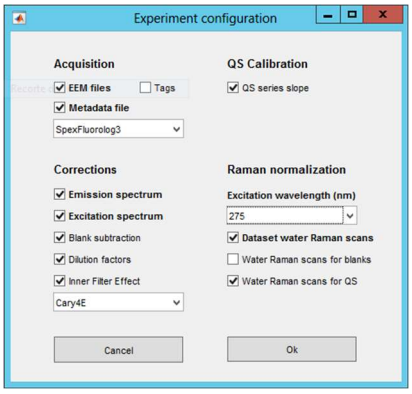

Figure 1. *EEMlab's configuration for the example dataset*

4.3.1 Import a sample log

The sample log is the metadata file in where the fluorimetry experiment files, the factors and the interactions between all of them are descibed. This is compulsory for the experiment correction. The original sample log includes 229 records to describe the 229 samples that are part of the experiment. To import the sample log do

Load → Dataset → Metadata

and use the loader assistant to select the *SampleLog_PortSurveyDemo* sample log file. The GUI supports both *.xlsx* and *.csv* formats. To check the success in the load you can:

- execute the assistant and see how the correspondent checkbox is now disabled
- edit the log file (Experiment → Log) and see what happened with the load

Both checkings can be repeated for each one of the steps described next.

4.3.2 Sample EEMs

The complete list of EEMs that compose the experiment is detailed in the sample log metadata file. To do the load of the samples go to the EEMlab's menu toolbar and select:

Load → Dataset → EEM files

The loader assistant shows you the list of fields included in the sample log file. Select the `EEMfile` field that stands for the list of EEM files included in the experiment. Now, you use the assistant to navigate and select the directory in where the original EEM files are stored (typically in *EEMLABDSET\dreem\EEMs*). The load fails as the sample log presents five records with EEM files that are not stored in the directory. Then an error message is displayed and the list of wrong files is reported in the EEMlab's log (Experiment → Log). In order to solve the problem:

- edit the sample log and delete the records that include the wrong EEM files (records 56, 57, 73, 74 and 75). Then the new sample log includes 224 records (instead of 229) that is consistent with the number of EEM files located in the directory
- repeat the sample log's load
- repeat the sample EEMs' load

Now the load is completed and the GUI displays the plot and information for the first sample in the series of 224 that compose the experiment.

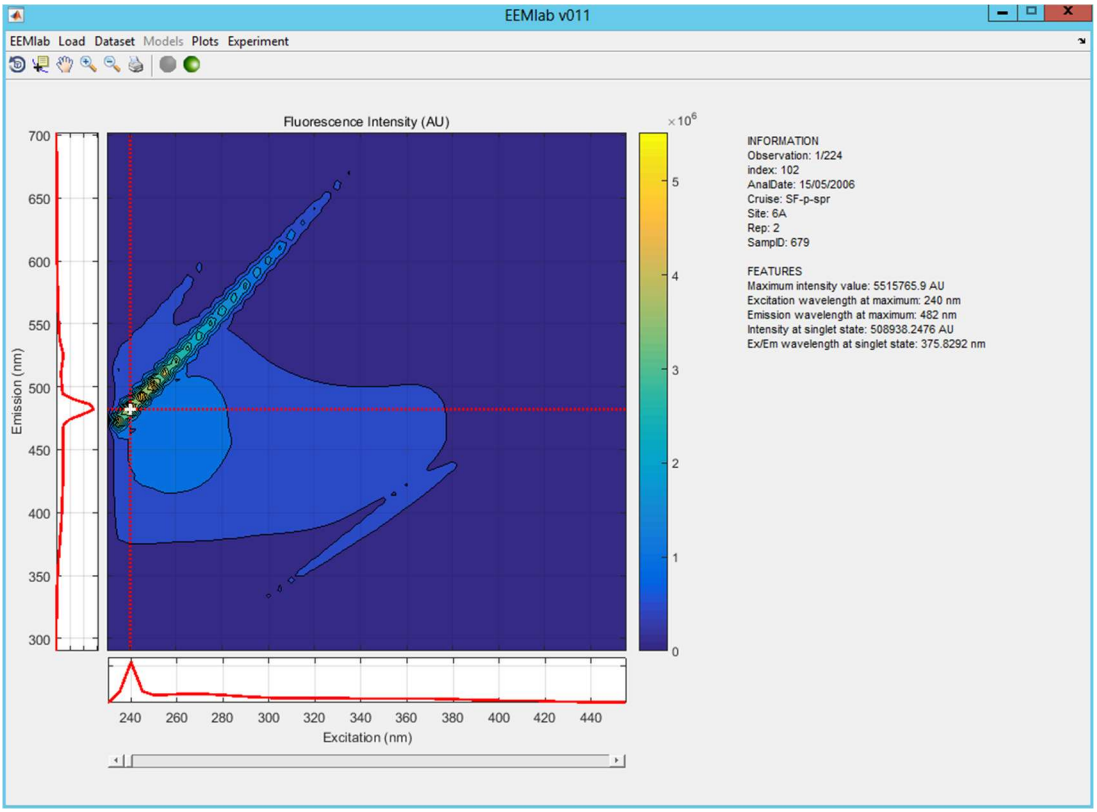

Figure 2. *Load of the EEM files. (left) The EEM for selected sample, the maximum intensity value and the Ex/Em scans for the maximum. (right) The information referred to the labels (if loaded) and saple features*

4.3.3 Labels

Depending on the sample log file, a multiple series of labels can be included for each sample. Labels are important for the sample identification and grouping (and specially for the outlier identification stage). To the load and reconciliation of the labels do:

Load → Dataset → Tags

and the use the assitant to select the tags for `index`, `AnalDate`, `Cruise`, `Site`, `Rep` and `SampId`. Then, the labels for the sample index (related to the sample log file), the sample identifier, the analysis date, the cruise and site and the number of sample replicates for each one of the samples are automatically included into the experiment. In addition, the GUI is updated the sample displayed is also updated with the correspondent labels.

4.3.4 Spectral correction files

As it was marked in the experiment configuration, the dreem dataset samples need to be spectrally corrected. To load the excitation correction scans do the following:

Load → Corrections → Spectrum → Excitation

The assistant shows you the list of fields included in the sample log file. Select the `ExcorFile` field that stands for the list of excitation spectrum correction files defined by the sample log. Now, you select the directory in where the 8 excitation correction files are stored (typically in *EEMLABDSET\dreem\CorrectionFiles\Excor*). In this case the EEMlab performs both, the load of the files and the automatic reconciliation of the correction files with their correspondent EEM files according to the information included in the sample log file.

For the load of the correction files for the emission spectrum just repeat the processing described above (Load → Corrections → Spectrum → Emission) and change the selection on the EEMlab's menu and the emission correction files' directory.

4.3.5 Absorbance Scans

The next step is to load the files to correct the fluorimeter inner filter effect (IFE). These files are absorbance scans acquired with a spectrophotometer. To load and reconcile the absorbance scans according to the sample log do:

Load → Corrections → Inner filter effect → Absorbance scans

Then check the *ABSfile* field and select the directory withn the scans (*EEMLABDSET\dreem\Abs*). Now, the EEMlab displays an error message with the list of 6 extra absorbance files that are not referenced in the sample log. The lis of wrong files is updated in the EEMlab's log. In order to solve this problem you have to:

- look at the log to get the list of wrong files and delete them (or move them into a new directory in *EEMLABDSET\dreem\Abs\extra*, as the load script is not recursive)
- repeat the load. If no errors, the reconciliation between the scans and their correspondent EEM files is automatically done according to the sample log

4.3.6 Blank (miliQ) EEMs

Now we repeat the procedure in order to load the blanks.

Load → Corrections → Dataset dilution → Blanks

Select the *BlankFile* field and load blanks from *EEMLABDSET\dreem\BlankEEMs*. Again, a list of fourteen blank files with no sample log reference is displayed. Then, repeat the procedure described in the previous paragraph. If both, the sample log and the information

match, the load and reconciliation are performed. In addition, water Raman scans for blanks are extracted at the defined excitation wavelength (see the experiment configuration) and reconciled with EEM files accordinag to the sample log. This is automatically made by EEMlab and the scans are used for normalize blanks only in the case of need.
Once blanks are loaded we also load the dilution factors directly included in the sample log.

Load → Correction files → Dataset dilution → Factors

Then, select from the assitant the `dilutionfactor` tag. No other action is required.

3.3.6 Quinine sulfate dilution series

The example dataset is calibrated to QSE units for interlaboratory comparison [Murphy et al., 2010, Environ Sci Technol, 44, 9405-9412]. So, we load the slope for the QS dilution series. This information relies on the sample log. Just do:

Load → Calibrations → QS series slope

and select the *QS_slope* tag. The load and the reconciliation are performed and the results, once the dataset corrected, will be displayed in QSE units.

4.3.7 Water Raman Scans

To normalize fluorescence intensities to RU275 we have to load the water Raman scans acquired at 275 nm excitation wavelength (see experiment configuration). Also accordingly with the experiment configuration no water Raman scans for blanks were produced. Then, scans for blank are directly taken from the balnks themselves at the referred excitation wavelength (see the paragraph describing the *Blank (miliQ) EEMs* load). To load the scans for the dataset do:

Load → Normalizations → Water Raman scans for dataset

and select the *RamanFile* tag and the *EEMLABDSET\dreem\EEMs\Raman275* directory. There is, again, a list of three raman scans not included in the sample log. Remove them from the folder and repeat the load.
Finally, and in order to load the scans for the QS dilution series do:

Load → Normalizations → Water Raman scans for QS

and select the *Qw* tag and the *EEMLABDSET\dreem\QS\Raman275* directory. If there is no error in the processing, the load of files and reconciliation is automatically done.

4.4 Align the various files and datasets

As it is described in the section above, the sorting of the files and data to match the reference in the sample log file (reconcile data) is synchronized and done accordingly with the load of files. On this sense, no other actions are required by the EEMlab's user.

4.5 Saving and retrieving the dataset

Anytime, the EEMlab gives the user the option of saving the dataset in MATLAB format. This is convinient if we want to stop processing the experiment to recover it later. The datased is saved according to the drEEM's structure (see the drEEM's toolbox `assembledataset` script). In addition the EEMlab gives the chance to include in the experiment log file any annotations done by the user. In this point we can insert a comment (clarifying at which point the experiment is saved) and save the dataset.
For including a comment in the log file do:

Experiment → Insert comment

To save the dataset, do:

Experiment → Save as MAT file

the *EEMLABDSET\dreem\OutputFiles* directory is created automatically with the experiment mfile and experiment log are stored in it. The dowloadable dataset already provides this mfile in *EEMLABDSET\dreem\OutputFiles\dreem_raw.mat.*
In order to recover a previously stored experiment yo can do:

Experiment → Load MAT file

and EEMlab will automatically proceed with the load of the requested file. Be aware of the long loading time depending on the data size. Once loaded, the user can access the log file to consult all the processing followed to get the current data.

4.6 Correct the EEMs

Once the correction files are loaded we proceed to the correction of the EEM samples. EEMlab offers two options to apply corrections:

Option 1→ all the correction in one step, for that in EEMlab, select:

Dataset → Correct→ Step 4 (when QS normalization is not needed) or Step 5 (in case QS normalization is needed). EEMlab will automatically apply all the previous steps

Option 2 → apply the correction one by one, for that in EEMlab select:

Dataset → Correct → then press step by step the corrections that are of your interest

then a modified version of the drEEM's `fdomcorrect` script is invoked an all the corrections are done. Feedback information is displayed thrtough the MATLAB console. The processing ends and the application updates the new information for the sample

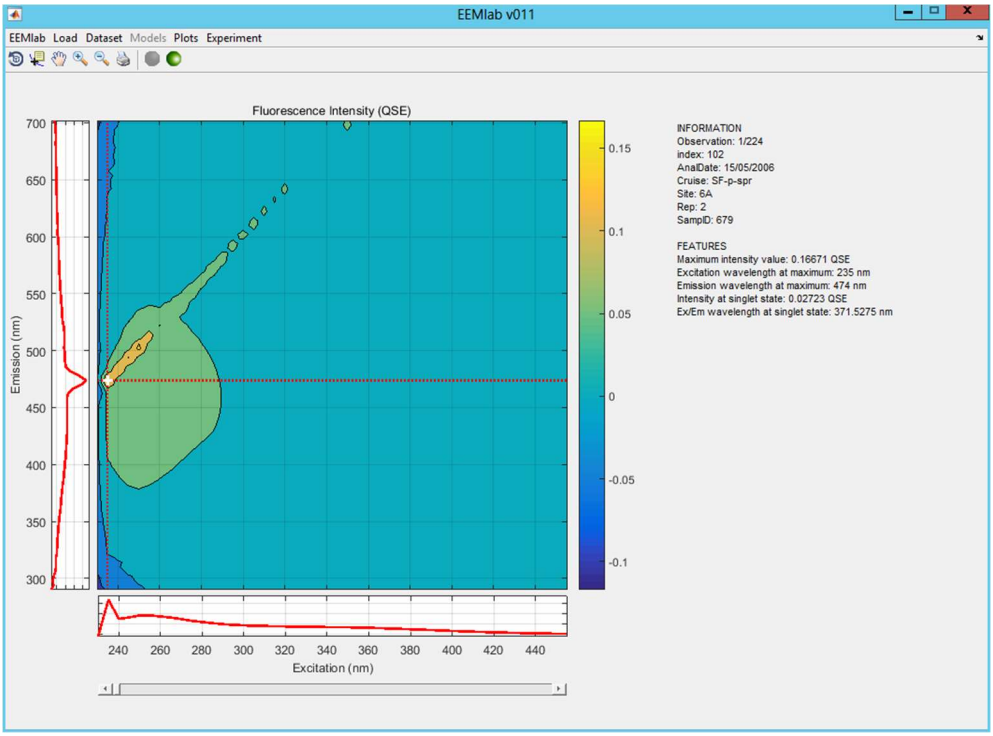

Figure 3. *Corrected EEM samples*

Now, you can repeat the steps above to insert a comment explaining the work done in this point and to save the corrected dataset. At the end of this processing we have loaded, corrected and saved a dataset of EEMs without the need of coding in MATLAB. Anyway, still remain several preprocessing steps before getting a valid dataset, suitable for the PARAFAC analysis. At this point we recommend the user to save the corrected dataset as a waypoint to go back if the further processing fails (Experiment → Save as MAT file). If you have downloaded and installed the dataset results ZIP file, the mfile correspondent to this waypoint is stored in *EEMLABDSET\dreem\OutputFiles\dreem_corrected.mat.*

5. PARAFAC analysis using EEMlab

5.1 Getting started

The EEMlab GUI uses the drEEM toolbox as the core processing for (i) the preprocessing, (ii) the exploratory PARAFAC modeling, (iii) the refinement of the exploratory models, (iv) the split half analysis and (v) the final model validation.

5.2 Data Import

At this point and if you have downloaded and installed the dataset results ZIP filethe experiment data can be imported into EEMlab taking it from the mfile stored in *EEMLABDSET\dreem\OutputFiles\dreem_corrected.mat*. This file includes the previously loaded and corrected dataset. Another option is to continue with the EEMlab tutorial at the point where you finished the previous section. In any case the GUI loads and displays the samples for the corrected example dataset.

5.3 Preprocessing

The EEMlab implements the preprocessing of samples inside the *Dataset* menu, including:

- the resize of the dataset to exlude noisy or contaminated excitation or emission wavelengths
- the remove of scatter peaks and contaminated parts of EEMs
- the identification and removal of outlier samples
- the correction of contaminated areas in samples
- the dataset statistical normalisation

All the proposed preprocessing stages can be iteratively repeated.

5.3.1 Resize the dataset

Sometimes the excitation/emission range used in acquisition is too wide or it contains contaminated areas that have to be eliminated by resizing the dataset. Non-valid areas are identified by manual inspection. To do that the user can use the EEMlab's slider to inspect sample by sample. Another option is to select:

Plots → Dataset → Collection

and to define the layout in where the collection of samples is displayed. Once the collection of samples is inspected by the user (and from the eference document):

> *"The plots… show non-trilinear variation (appearing as diagonal peaks) due to primary and secondary Raman and Rayleigh-Tyndall scatter… get rid of parts of the EEM that have more scatter that signal (Em > 600) or that are noisy and/or likely to exert disproportionate leverage on the model (Ex < 250)"*

Then select:

Dataset → Resize

to define the wen ROI accordingly to the cited paragraph (250<Ex<455 nm and 290<Em<600 nm). In fact, we can remove scattering effects before resizing, as is easier to define a ROI on the corrected samples.

5.3.2 Remove scatter peaks

In this regard just use EEMlab to select:

Dataset → Remove scattering

and define the correction bandwidth for both, Rayleigh (1st and 2nd order) and Raman (1st and 2nd order) scattering effects. By default the bandwidths (in nm) are:

- Rayleigh 1st order: 18 - 15
- Raman 1st order: 15 - 15
- Rayleigh 2nd order: 17 - 18
- Raman 2nd order: 19 - 18

now you can select the following options to best define the removing scatter peaks processment:

- Interpolación on/of: turn it on/off by writing 1 or 0, for each scatter band [Ray1 Ram1 Ray2 Ram2] by writting 1=on, 0=off. For example:
  - [] or [1 1 1 1]: interpolation of all peaks
  - [1 1 0 0]: interpolation for primary peaks and NaNs for secondary peaks
- Distance below 1st Rayleigh before zeros (nm): zeros may be placed at specified distance below the line Em=Ex. If zerowith=[], no zeros or NaNs are placed. The default value is 40 nm

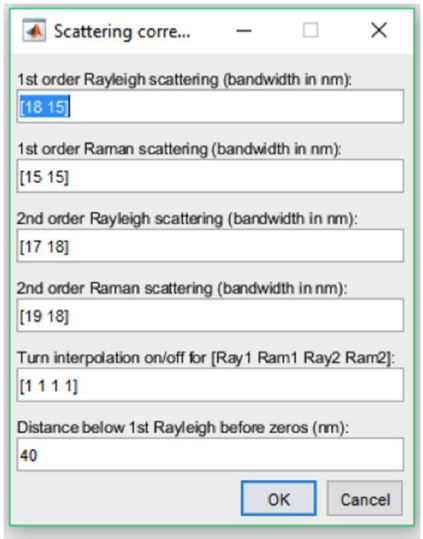

Figure 4. *Detail of options to remove scatter peaks*

At the end of the processing the smooting is done and the scattering bands are corrected. Correction on the full collection of samples can be manually inspected by the Plots → Dataset → Collection option.

5.3.3 Remove outlier samples

The removal of the outlier samples is a key stage for getting an easy-to-model dataset. Indeed, the outliers (high-leverage samples) become undesirable elements which could cause a significant loss of useful information and corrupt the dataset. To facilitate the outiler identification directly on the corrected dataset, the EEMlab includes a scatter plot to associate each representation of the sample with a label (see the section about loading the labels). This plot is based on a series of features, that are automatically extracted from each sample:

- the maximum fluorescence intensity value
- the single state, defined by the wavelength in where the excitation and emission scans for the maximum fluorescence intensity value crosses themselves
- the stokes distance, that is the distance among excitation and emission wavelengths for the maximum fluorescence intensity value

To start with the outiler identification do:

Plots → Dataset → Scatter

and select the two features to plot for each sample (i.e. the maximum intensity and the singlet state). Next, select the labels to display in the scatter plot (let the default value if you do not want to label the samples). In this case we select the label for the `Site`. From the reference document, the samples identified as outliers are the 9 samples tagged with `bl` (blank samples, with `SampID` 739, 740, 741, 742, 744, 1344, 1345, 1346 and 1348) and the 7 samples tagged with `0A` (tributary measurements, with `SampID` 1025, 1026, 1027, 1034, 1036, 1431 and 1433). On that document, the method followed for the outlier identification was the manual inspection of the full collection reinforced with, maybe, the previous knowledge about the origin of the samples. We are supposed not to have any prior about the origin of the samples and so, the scatter plot is used to assess the outlier identification.

Once the graph is displayed, you can use the *Data Cursor* and *Zoom* tools (in the figure toolbar) for selecting and checking the values in each sample. Be careful with your inspection as the *Data Cursor* shows the *observation* number referred to in the main *EEMlab GUI*. Then we easely identify the `bl` (blank) samples as outliers. This is no so clear with the `0A` (tributary) samples. Anyway we remove both tagged `0A` and `bl` samples. Just do:

Dataset → Remove outliers → by Tag

the we select the category label used for the scatter plot (`Site`) and next, the conctrete tags (`A0`, `bl`) whose samples we want to remove. Now, the dataset is reduced to 208 valid samples. Now, the scatter plot does not include the `A0` and `bl` tagged samples and the cloud of points seems to be better defined than before. This outlier removal stage can be repeated anytime before the dataset normalization.

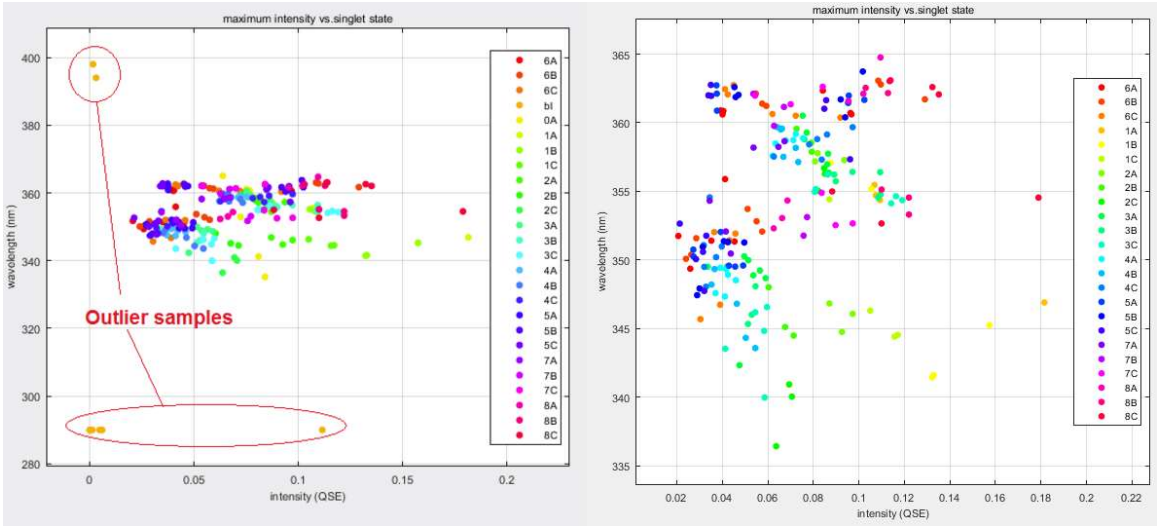

Figure 5. *Scatter plot of the labelled samples ('Site' label). Original dataset (left) where'Ao' and 'bl' samples lare identified as outiler samples (left). Dataset when outliers are removed (right)*

In addition, EEMlab offers another option to remove samples individually. Just do

Dataset → Remove outliers → by Component

and use the *ctrl* key for the individual selection of samples from the list.

5.3.4 Correct samples

As an alternative to identify a sample as an outlier and delete it from the database, it is also possible to remove parts of samples containinig faulty data. In this way, we can recover the sample just marking the corrupted areas as not valid areas. The identification of these valid but not correct samples is made by manual inspection. To this respect we can use Plots → Dataset → Collection to quicky inspect all the samples in the database. The observed sample 176/208 (with `SampID 667` and tagged with `Site 4C`) is the only candidate (from the samples collected at the same time and place) that shows a sharp peak in emission scans collected at excitation wavelengths of 345 and 350 nm. This is characteristics of a fluorometer error and then, we can correct the sample with:

Dataset → Correct samples

select the observation 176/208 and correct the 345 - 350 nm excitation wavelength band.

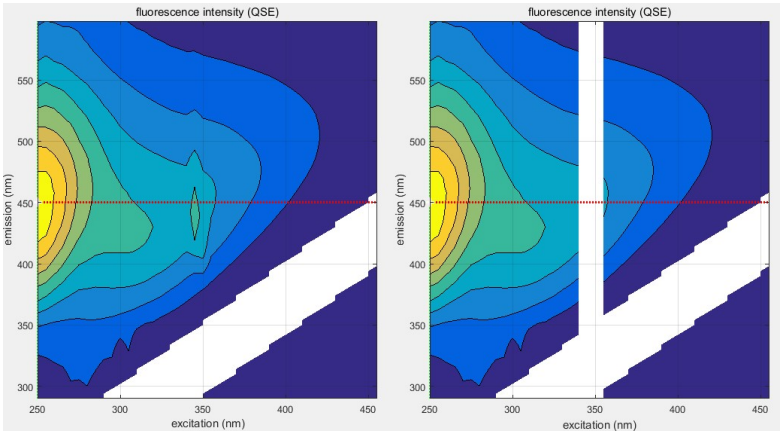

Figure 6. *Sample before (left) and after correction (right). Corrupted area was due to fluorometer errors*

5.3.5 Dataset statistical normalization

This preprocessing option can be used together with the exploratory data analysis in order to reduce the colinearity between components. As we will see next, its use depends on the results in the preliminary analisys. Both options, normalization and its reversal are avaliable for the dataset (Dataset → Normalize) and also for all the exploratory and refined models (if they are included in the dataset as a result of the PARAFAC modeling). If you have downloaded and installed the dataset results ZIP file the data in this point i stored in *EEMLABDSET\dreem\OutputFiles\dreem_preprocessed.mat.*

6 Dataset modeling

Once we have the example dataset corrected and preprocessed we can start using PARAFAC for getting the model that best fits with the dataset. In this sense EEMlab can manage different models depending on the drEEM tools used to develop each stage of the modeling process. There are three possible models and the final validation of the overall model depends on how models have been created and tested:

- Exploratory models: that are used for a fast dataset modeling in order to explore ther dataset and look for outlier samples and/or leveraged Ex/Em wavelengths.The processing is fast due to these do not use all the information available in the dataset
- Refined models: once the dataset is explored, the user can refine exploratory models. The (exploratory model) is refined by using all the data available in the dataset and running PARAFAC a certain number of iterations. Each iteration is randomly initializaed. The best least squares regression model is selected as the refined model. This processing is slower than the exploratory analysis
- Splitted models: that are calculated over a certain split of the dataset in order to make the model independent from sample groups. Splitted models are then calculated on the spitted dataset and nested into the `Split` struct. To calculate splitted models EEMlab runs an independent PARAFAC for each half of the dataset. This processing plus the model validation is implemeted and known as the extended $S_4C_6T_3$ split half analysis
- Model validation: only splitted or refined models can be validated. The processing starts with the validation of the n-component splitted models with the goal of the final validation of the corresponding n-component refined (overall) model. If splitted models are valid (we get similar models with independence of the split of the dataset used) then we can compare the refined model with the splitted ones. The refined model is also valid if their Ex/Em splectrums are similar (component by component) to the correspondent Ex/Em splitted models spectrums for each one of the compared splits. If the model is not valid the user can try a more restrictive refinement. This processing (refinement/split half validation) iterates until the user finds a valid model or decides to repeat previous processings (preprocessing, outlier test...) to improve the dataset

As EEMlab implements error checking (this means that user can not do a processing if all the previous requirements are not met), the EEMlab's dataset modeling lifecycle is the following:

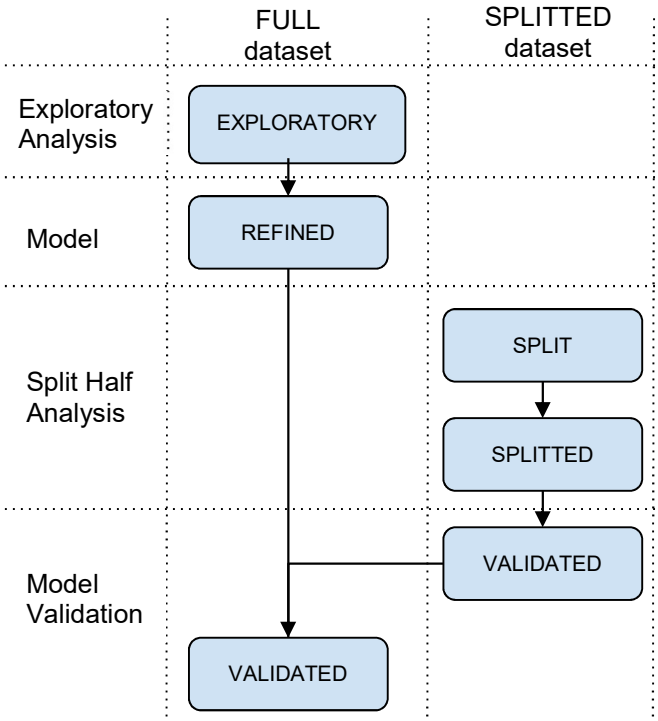

Figure 7. *EEMlab's modeling lifecycle*

6.1 Exploratory data analysis

At the end of the preprocessing stage and to our understanding, a coherent dataset, reviewed by manual inspection, is ready to be modeled. The results in a further exploratory data analysis can change this perception, and maybe some of the stages in the preprocessing will be repeated. From the reference document:

> *"The goal of exploratory data analysis is to get a feel for a dataset without investing a great deal of time an effort in obtaining the best possible solutions"*

In this way EEMlab enables a configurable preliminary test using the drEEM's `outliertest` script. All this is just to check, in a fast way, that the EEMs are ready for modeling with PARAFAC. From now, just select

Models → Explore dataset

then, the assistant opens and the user can configure all the parameters for generating several test models. These models will be used for checking the best conditions for the later PARAFAC analysis. In this case we use all the data to test models from 3 to 7 components with a non-negativity constraint. The simulation runs 'at once' (no intermediate plots are produced). If you want to check the final Ex/Em spectra for the simulated models, do:

Plots → Models → Compare loadings between components

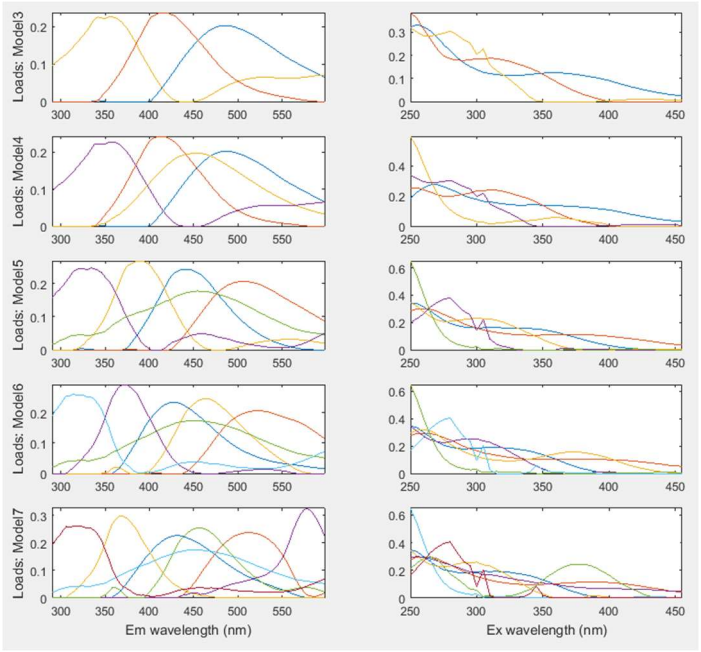

Figure 8. *Comparing Em and Ex spectra for models with 3 to 7 components*

The following step is to display scatter plots between components for a selected model. In ths sense the EEMlab uses the `compcorrplot` script for plotting the scores for the estimated concentration for each one of the components in the model. For example, if you want to inspect the correlations between components in the 5-component model do:

Plots → Models → Correlations between components

and use the assitant to select one of the models (and tags, if desired tagged scatter plots) previously created for the exploratory analysis. If the plot shows a strong correlation between two or more components (as it happens) this suggest that dilution is a dominant mechanism in the dataset.

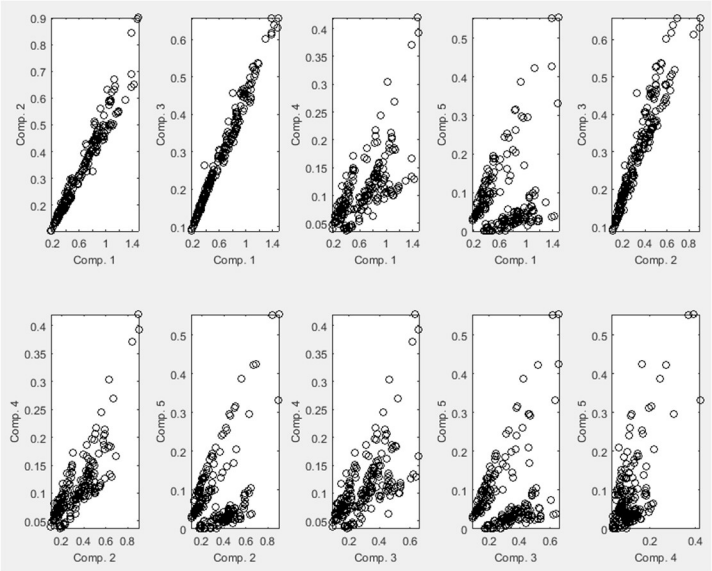

Figure 9. *5-component Model. Strong linear correlation between components 1, 2 and 3 due to dilution factors*

In order to reduce the concentration-related colinearity and give the low-concentration samples the chance to enter the model just apply each sample a statistical normalization to unit norm. The normalization consists on reducing data variance to unit and it can be reversed after the PARAFAC analysis in order to get the true data. For normalization do:

Dataset → Normalize → Forward

Once the dataset is normalized, we repeat the exploratory analysis for the outlier testing of models from 3 to 7 components. We get the following scatter plots where colinearity has been drastically reduced.

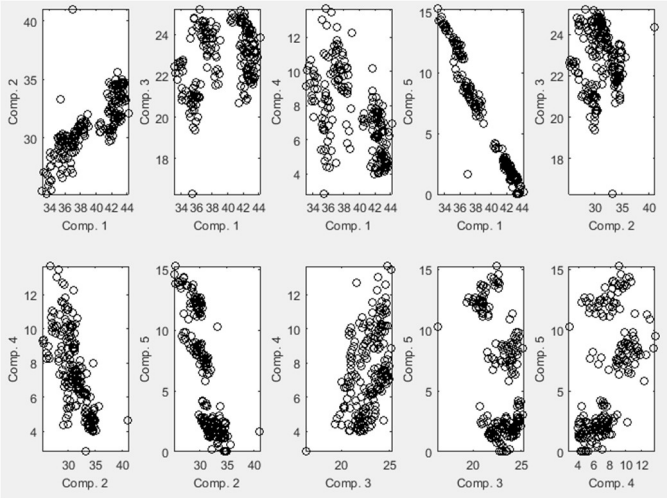

Figure 10. *Colinearity related to concentrations is reduced due to normalization*

We can also check the waveforms of the loadings for excitation/emission spectrums. This checking is useful to compare the spectra for all the n-component models and identify spectra that does not look like fluorophores (and maybe they are being used to model noise). Just, do:

Plots → Models → Spectral loadings per component

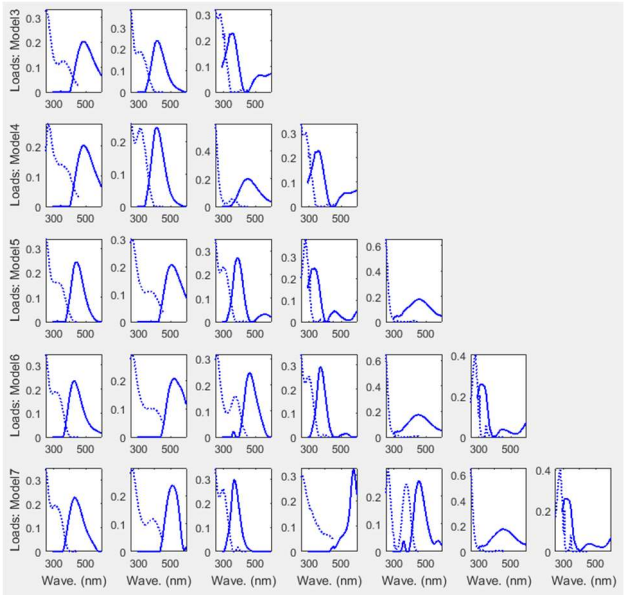

Figure 11. *Spectral loadings for all the n-component models*

And, if we want, it is also possible to enrich the analysis plotting additional information as the leverages detailed for sample and spectrum.

Plots → Models → Loadings & leverages

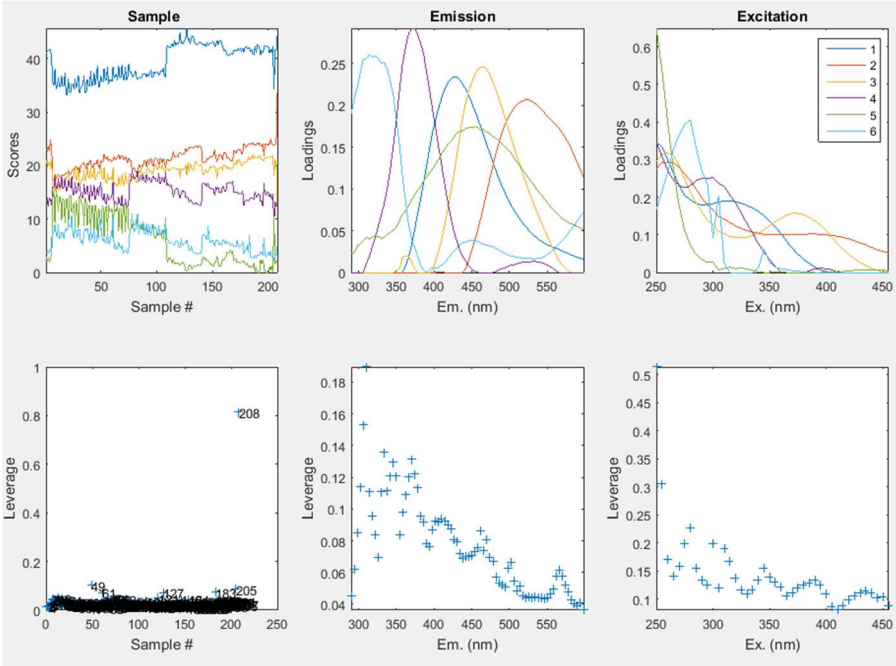

Figure 12. *Loadings & leverages for the 6-component model*

Having a look on the samples' leverages it seems very clear that the sample 208 is an outlier. We can not take other conclusion about outlier excitation or emission wavelengths. Consequently and now we have to:

- recover ALL the models as they were before the normalization (Dataset → Normalize → `Reverse`)
- delete the outlier 208 (Dataset → Remove outliers → by Component)
- normalize the dataset (Dataset → Normalize)
- inspect the distribution of the samples just displaying the scatter plot for the singlet state with the `Cruise` tag ()
- repeat the fast exploratory analysis with the new 207 samples dataset (Models → Explore dataset)

When repeating the manual inspection on the loadings and leverages, there are still some high leverage samples but none appear to be severe. A possibility to improve the dataset is deleting the range of wavelengths that add more leverage to the dataset (Dataset → Correct samples) but, again, if inspecting the lleverages for Ex/Em ranges none of the wavelengths seem be significant enough to be corrected.

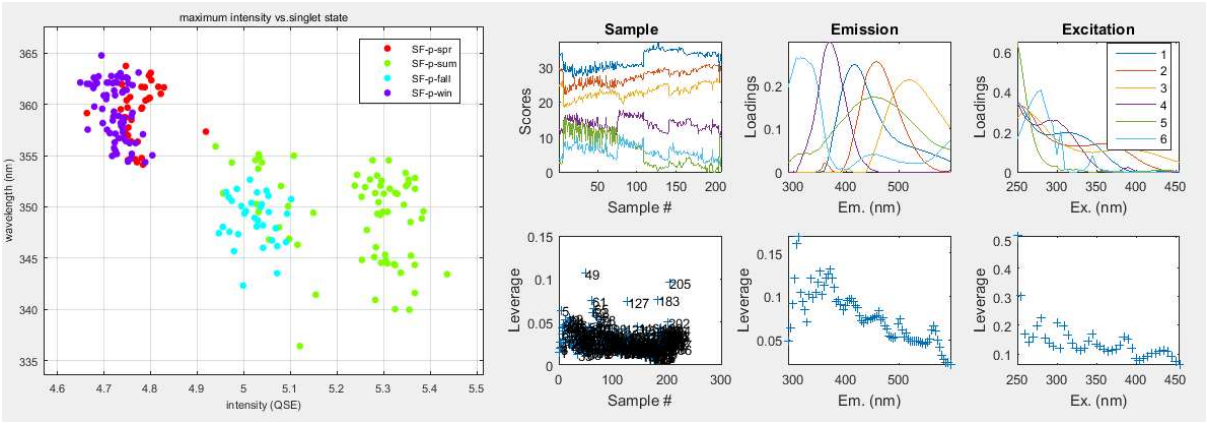

Figure 13. *Scatter plot of the samples tagged with the 'Site' label (left). Loadings and leverages for the new 207 sample dataset (right)*

In addition, EEMlab implements a series of plots that can help the user to decide which is the PARAFAC that best models the EEM dataset. To look at the error residual, do:

Plots → Models → Error residuals for one model

and use the assistant to select the model. On the other hand, you also have the possibility to compare error residuals among two models with;

Plots → Models → Compare error residuals (two models)

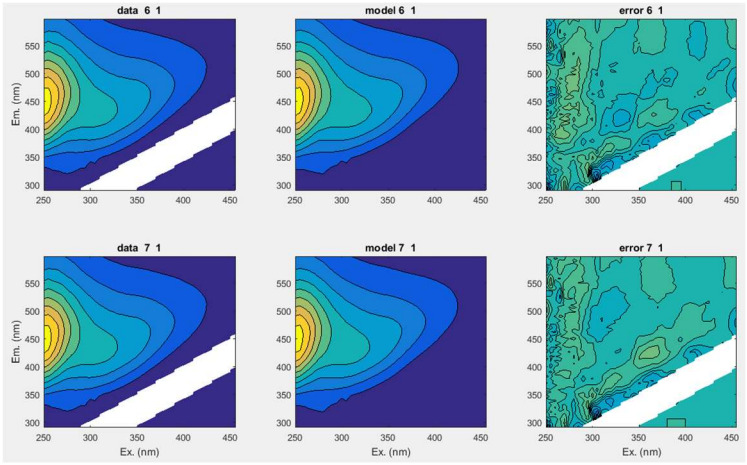

Figure 14. *Error residuals for sample 1 in models 6 (row up) and 7 (row down)*

Next, we can compare the Sum of Squared Error for all the models in the Ex/Em spectrums (Plots → Models → Spectral Sum of Squared Error (SSE)). From the reference document:

> *"A useful indication of the number of components that should be in the model can be obtained by SPECSSE, which shows the effect on model fit of adding more components, expressed as the sum of squared error (SSE) for each model plotted as a function of wavelength"*

Then, the SSE plot can help to deduce the number of components that underly in the dataset. In this case, the 6-component model is better than the 5-component (less SSE in all the spectrum) and the 7-component model does not seem to improve in terms of SSE the 6-component model.

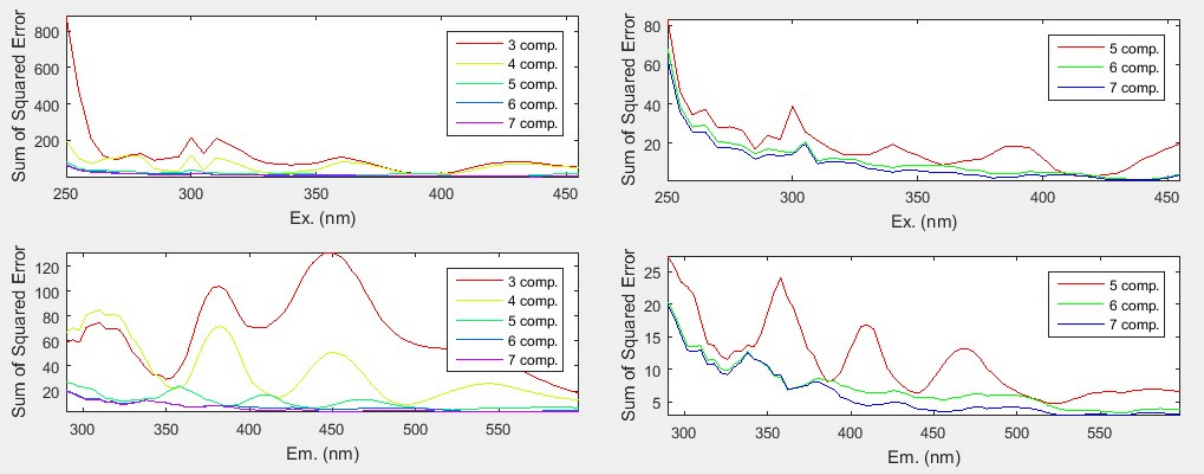

Figure 15. *SSE for the spectrums of all the models (left) and the SSE detailed for the 5, 6 and 7-component models (right)*

After this discussion we can take as the best supposition the assumption of DOM samples composed by a mixture of 6 fluorophores.
If you have downloaded and installed the dataset results ZIP file the exploratory models corresponding to this waypoint are provided in the *EEMLABDSET\dreem\OutputFiles\dreem_models.mat* MATLAB file.

6.2 Model refinement

Once we have the exploratory analisys, now we want to refine the PARAFAC preliminary model to ensure that it is a global model independently of how it has been initialized. EEMlab uses the drEEM's `randinitanal` script to refine a model. This script calculates the n-component model a number of times defined by the user. The priors of the model are randomly initialized for each iteration. Experiment is repeated just to check tif the SSE solution becomes global or not. The model with the less SSE is taken as the overall refined model for that number of components.

> *"When modelling does not produce a stable solution, it is an indicator that the model may have too many components and/or that more stringent (smaller) convergence criteria are needed"*

To calcuate the 5, 6 and 7-component refined models, for each model define the number of runs and the restrictions (as convergence criterion) and do the following:

Models → Refine model

Considerations to this processing:

- exploratory models are transformed into refined . This can be checked listing all the models in the GUI (Models → List)
- time consuming: be aware of the elevated computing time due to computation burden. Yo can check the processing time having a look on the log file (Experiment → Read log)
- Global SSE & Core Consistency plots: that help us to check which one of the iterations has produced the best model. See [Murphy et al., 2013, Anal Methods, 5, 6557-6566] for discussion of how to interpret core consistencies (Plots → Models → Global SSE & Core Consistency)

- Fingerprints plots: to see the EEM of PARAFAC components for each model (Plots → Models → Fingerprints)
- Mapping of components: that takes the maximum intensity value of each PARAFAC component and maps it against a region-map of fluorophores' families [Chen et al., 2003, Environ. Sci. Technol. 2003, 37, 5701-5710] (Plots → Models → Map components)

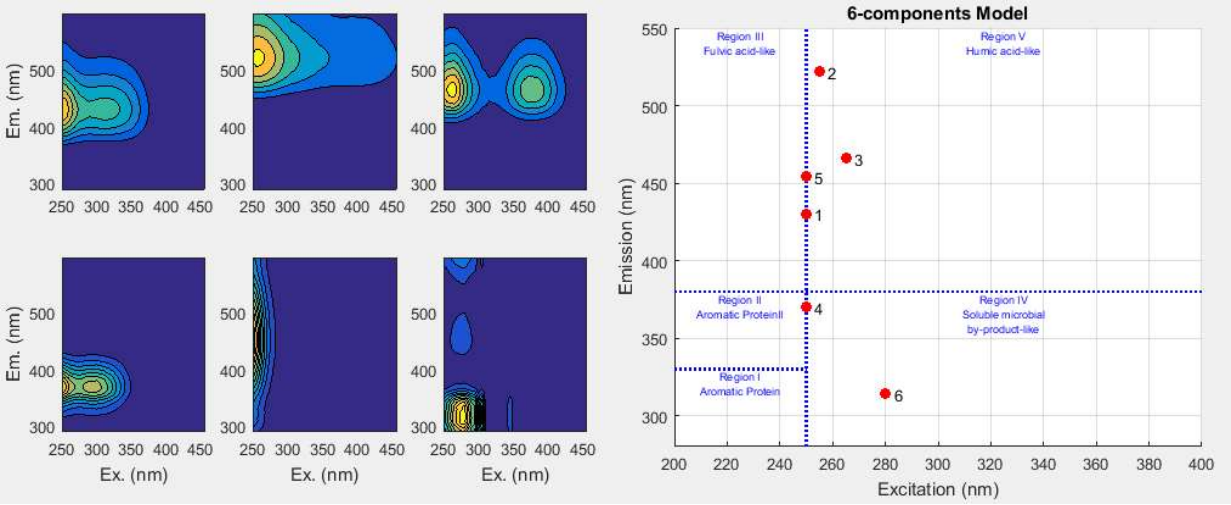

Figure 16. *Fingerprints (left) and mapping of components (right) for the 6-component refined PARAFAC model*

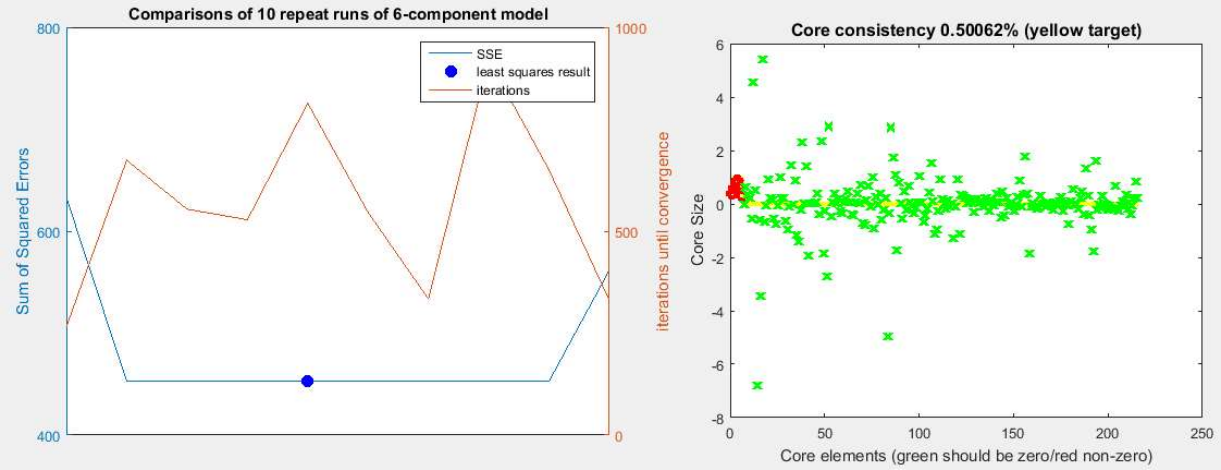

Figure 17. *Global SSE (left) and core consistency (right) for the normalized 6-component refined model*

if all the plots are inspected and when we are satisfied with the results (pay special attention to leverages and residual error plots), next we can try to validate the model with the split half analysis. If you have downloaded and installed the dataset results ZIP file the refined models corresponding to this waypoint are provided in the *EEMLABDSET\dreem\OutputFiles\dreem_models.mat* MATLAB file.

In respect to the computational burden of the refinement stage, we can check the computation time for each one of the refiniements inspecting the log file (Experiment → Read log). For the hardware used in this example[1] the estimations for the model refinement processing time (10 runs, non-negativity constraint and convergence criterion of 10e-6) are:

- 5-component model: 18 min 43 s
- 6-components model: 26 min 22 s
- 7-components model: 41 min 57 s

[1] Mac mini with 2,5 GHz Intel core i5 processor and 16 GB of RAM and macOS Sierra

6.3 Model validation

Once the models have been refined we want to know if they are valid or not. Then we want to be sure that one model does not depend on the number and groups of samples used for the PARAFAC modeling process. This can be done with the split half analysis [Harshman, 1984, Praeger].

6.3.1 Split half analysis

EEMlab implements the extended $S_4C_6T_3$ (4 splits, 6 halves, 3 test datasets) SHA [Stedmon et al., 2008, Limnol. Oceanogr. Methods, 6, 572-579] for the validation of PARAFAC models. The first step is to split the dataset in 4 parts, to combine these parts into 6 halves and, finally, to recombine the halves into 3 new test datasets ($S_4C_6T_3$) (Models → Split half analysis → Split the dataset). The new 6 splits are stored directly in the `Split` field of the `data` struct.

Once the dataset is splitted into 6 halves and, in order to validate it, all the halves are modeled and compared to check if they are similar between them without depending of the split used. Be aware of the heavy computational burden that results on a very long processing time and a very big amount of data. An splitted model can be computed if there has been a previous refined model. Then, the sequence of actions to get the 6-component and 7-component splitted models is:

- calculate the 6-component splitted model (Models → Split half analysis → Model the splits). For the hardware used in this example[1] this processing lasts 31 min 34 s
- repeat the processing to calculate the 7-component splitted model (Models → Split half analysis → Model the splits). For the hardware used in this example[1] this processing lasts 45 min 17 s
- save both, the 6 and 7-component splitted models (Experiment → Save). This action can spend a lot of time due to the big amount of data to save. For the hardware used in this example[1] this processing lasts 2 min 43 s. All the models generated on the datatset splits are again provided in the *EEMLABDSET\dreem\OutputFiles\dreem_models.mat* MATLAB file.

6.3.2 Validations

After the SHA analysis now we can validate the EEMlab models. The goal of validation is to check that a refined model (caclulated over all the samples, this is the overall model) is a valid model and do not depend on the samples in the dataset.

> *"If a model validates with SPLITVALIDATION it means that all of the components in the split model being compared in each test have found a match with Tucker correlation coefficient > 0.95"*

The first step is to validate the models for the dataset splits (models independently calculated for each dataset half). One model is valid if meets the Tuker correlation coefficient. In this sense, the *EEMlab v012* lets the user to define a value for the correlation coefficient (it set to 0.95 by default). Depending on the strictness of this parameter the splits can be valid or not. When we find that just one of the models is valid for one split [Bro et al., 2011, Chemometrics and Intelligent Laboratory Systems] in the extended $S_4C_6T_3$ validation tests then we check if the correspondent refined (overall) model is also valid. To this respect EEMlab implements the validation for the modeled splits (Models → Validations → Validate the splits) and for the overall refined models (Models → Validations → Validate the model). So, if one of the splits is

valid, the user can then validate the refined model. Just try and select the n-component refined model from the drop down.
When the refined model is valid both, the Ex/Em spectrums for the valid split and the overall model are displayed. The refined model fingerprints are also displayed.
In the case of the refined model is not valid, the user can repeat the previous processing (preprocessing, outlier test...) to improve the dataset and try to get a vaild model.
For the example dataset, when we validate the 6 and 7-component refined models, we obtain a valid 6-component model and a not valid 7-component model.

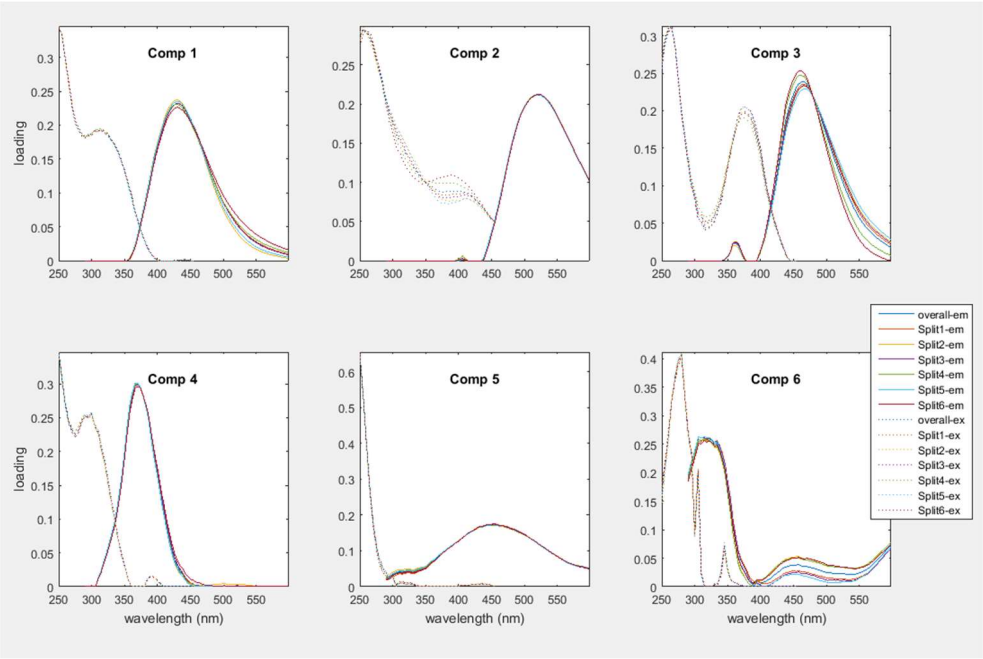

Figure 18. *Ex/Em spectrums for the 6-component splitted and refined models, grouped by component*

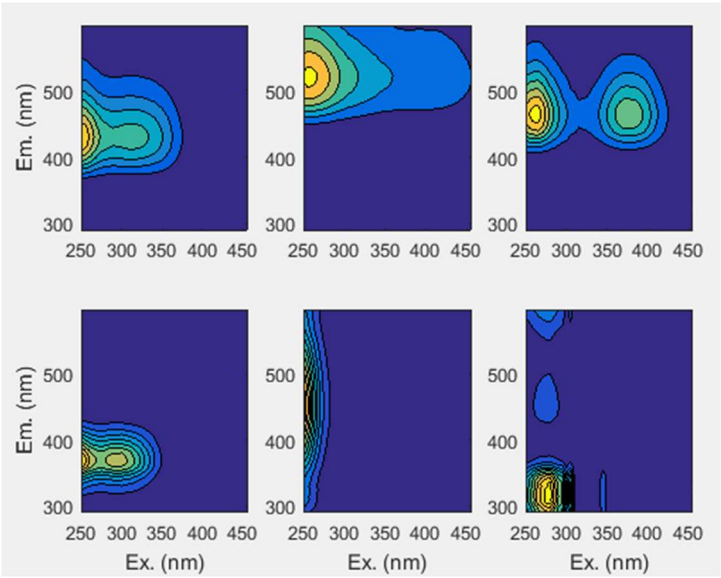

Figure 19. *Fingerprints for the 6-component valid refined model*

6.3.3 True scores

As the models are calculated with the normalized dataset, in order to get the true scores for the valid model the user have to reverse the normalization. Then do Dataset → Normalize →

Reverse. For the example dataset, and the referred MATLAB data structure, the true scores for the valid 6-componet refined model are located in:

- excitation spectrum for the 6 identified components in `data.Model6{3}`
- emission spectrum for the 6 identified components in `data.Model6{2}`
- component concentration per sample in `data.Model6{1}`

Anyway, the user can manually inspect Em/Em waveforms using the tools enabled in the Plots → Models menu.

7 Bibliography

- [Murphy et al., 2013, Anal Methods, 5, 6557-6566] K.R. Murphy, C.A. Stedmon, D. Graeber, R. Bro, *'"Fluorescence spectroscopy and multi-way techniques. PARAFAC"*, Analytical Methods 2013, 5, 6557-6566, online
- [Stedmon et al., 2008, Limnol. Oceanogr. Methods, 6, 572-579] C.A. Stedmon, R. Bro, "Characterizing dissolved organic matter fluorescence with paralel factor analysis: a tutorial", Limnol. Oceanogr. Methods, 6, 2008, 572-579
- [Murphy et al., 2010, Environ Sci Technol, 44, 9405-9412] K.R. Murphy, K.D. Butler, R.G.M. Spencer, C.A. Stedmon, J.R. Boheme and G.R. Aiken, *"Measurement of Dissolved Organic Matter Fluorescence in Aquatic Environments: An Interlaboratory Comparison"*, Environ. Sci. Technol. 2010, 44, 9405-9412
- [Murphy et al., 2013, *J. Mar. Syst.* 111-112, 157-166] K.R. Murphy, J.R. Boheme, C. Brown, M. Noble, G. Smith, D. Sparks, G.M. Ruiz, *"Exploring the limits of dissolved organic matter fluorescence for determining seawater sources and ballast water exchange on the US Pacific coast"*, *J. Mar. Syst.* 2013, 111-112, 157-166
- [Murphy et al., 2014, online] K.R. Murphy, C.A. Stedmon, D. Graeber, R. Bro, *"The drEEM toolbox for MATLAB"*, online
- [Chen et al., 2003, Environ. Sci. Technol. 2003, 37, 5701-5710] W. Chen, P. Westerhoff, J.A. Leenheer, K. Booksh, "Fluorescence Excitation - Emission Matrix Regional Integration to Quantify Spectra for Dissolved Organic Matter", Environ. Sci. Technol. 2003, 37, 5701-5710
- [Harshman, 1984, Praeger] R.A. Harshman, "Research methods for multimode data analysis", ed. H.G. Law, J.C.W. Snyder, J. Hattie and R.P. McDonald, Praeger, New York, 1984, pp. 602-642
- [Bro et al., 2011, Chemometrics and Intelligent Laboratory Systems] R. Bro, M. Vidal, "EEMizer: Automated modeling of fluorescence EEM data", Chemometrics and Intelligent Laboratory Systems, 106 (2011), 86 - 92
- [Micó, 2017, online] P. Micó, *"The EEMlab Project"*, online

8 Glossary of acronyms

- DOM: Dissolved Organic Matter
- drEEM: Decomposition Routines for Excitation Emission Matrices
- EEMlab: Excitation Emission Matrices Laboratory
- IHSS: International Humis Substances Society
- NOM: Natural Organic Matter
- PARAFAC: PARAllel FACtor analysis
- QS: Quinine Sulfate
- QSE: Quinine Sulfate Equivalent units
- AU: Arbitrary Units
- RU: Raman Units (at excitation 350 nm)
- RU275: Raman Units (at excitation 275 nm)
- RU350: Raman Units (at excitation 350 nm)
- SHA: Split Half Analysis
- SSE: Sum of Squared Errors