TUTORIAL R-SCRIPT FOR DATA ANALYSIS – PRODOM project

GENERAL INFORMATION

Project: PRODOM - Proactive Optical Monitoring of Catchment Dissolved Organic Matter for Drinking Water Source Protection. University College Cork.

Version: 21 February 2023 – v1.0

Authors: Boris Droz, Elena Fernández-Pascual, Jean O'Dwyer, Emma H. Goslan, Xie Quishi, Simon Harrison, Connie O'Driscoll, John Weatherill

Corresponding author: John Weatherill (PI)

School of Biological, Earth and Environmental Sciences, University College Cork, Cork, Ireland

e-mail: john.weatherill@ucc.ie

Content: The document briefly describes the script present on GitHub (https://github.com/Boris-Droz/PRODOM). The source code is primarily written in the R language version 4.2.0.

Associated paper:

Boris Droz, Elena Fernández-Pascual, Jean O'Dwyer, Emma H. Goslan, Xie Quishi, Simon Harrison, Connie O'Driscoll, John Weatherill. Predicting Disinfection Byproduct Formation in Chlorinated Drinking Water using Fluorescence Spectroscopy and Machine Learning, submit in ES&T.

Associated dataset and workflow:

Boris Droz, Elena Fernández-Pascual, Jean O'Dwyer, Emma H. Goslan, Xie Quishi, Simon Harrison, Connie O'Driscoll, John Weatherill Data of the EPA - PRODOM project v1.0. https://doi.org/10.5281/zenodo.7244913.

Boris Droz, Elena Fernández-Pascual, Jean O'Dwyer, Emma H. Goslan, Xie Quishi, Simon Harrison, Connie O'Driscoll, John Weatherill Calibrate machine learning model workflow for disinfection byproduct formation prediction for R. v1.0. https://doi.org/10.5281/zenodo.7886046.

Funding: Irish Environmental Protection Agency (Grant No. 2019-W-MS43) as part of the EPA Research Program 2021-2030

License: GNU General Public License v3.0

SUMMARY OF THE WORKFLOW

The analytical workflow is separated into two main steps divided into two respective folders:

1. excitation−emission matrix (EEM) - parallel factor analysis (PARAFAC): `EEM-PARAFAC` folder
2. Machine learning (ML) tool: `ML-script` folder

The workflow refers to the method used in the associated paper. Please refer to it for more details.

The analytical workflow presented here also contains additional options not used in the context of the paper but are described in the details of each script.

INPUT DATA

A full description of the input data is presented at https://doi.org/10.5281/zenodo.7244913 on the metadata file.

OUTPUT

The calibrated model used in the associated paper is presented at https://doi.org/10.5281/zenodo.7886046. The repository contains the calibrated model used in the paper as well as a function (script) to perform the direct predictions using the calibrated set of models. For each disinfection byproduct species, the two models presented in the associated paper are available in the repository. ML_opt3 and ML_UV refer to the optimal and spectral model, respectively.

TUTORIAL

Whenever you start a new project, it is not necessary to start this from a new working directory. The script automatically produces a new output folder containing the ongoing analysis with the date of the year. If two analyses are performed on the same date an increment is just added at the end of the folder name (e.g, 2022-11-04_mod.proj_1, 2022-11-04_mod.proj_2,..).

Starting and developing a new project typically consists of:

1. Create your working directory containing a folder named `input` and `output`.
2. Copy the script from GitHub and put it in your working directory (preferentially in a separate folder named script).
3. The file structure of the input is presented in the metadata file at https://doi.org/10.5281/zenodo.7244913 should be followed.
4. Scripts are organized in chronological order 1, 2, …

All scripts are organized similarly and contain a header with the script name and basic information (running R version, script version, coder, date). This is followed by a short description of what the script does, the important information about the script, and the reference for codes taken from earlier works. Finally, a PARAMETER section provides the parameters that the user should or can change to correctly run the script. In the parameter section of each script, the working directory (`workdir`), i.e, the folder path the project folder is located, should be set similarly for a given project. In each script, the code is provided below the declaration:

---- SCRIPT STARTS HERE ---.

FOLDER & FILE OVERVIEW

- EEM-PARAFAC

1.EEM_data_processing_PRODOM_v2.0: perform preprocessing before PARAFAC analysis.

2.PARAFAC_MODEL_PRODOM_v3.0: perform PARAFAC analysis.

- ML-script

1. Transform.norm_v3.1: test transformation of the dataset to obtain normally distributed data. Transformation is evaluated on the quantile-quantile plot, Shapiro, and kurtosis test.

2. nnet_cancel_proc_v3: Perform Neuronal Network input cancellation FUNCTION

2. var.explo.cor.pca_v6.1: Perform diverse analysis to select the best variable set for the further ML model. Correlation plot, principal component analysis (PCA) and cluster analysis are performed.

3. Model.proj_v14.6: Perform (calibrate) separately several machine learning techniques (chosen by user):

    - Neural Networks (nnet -- nnet package v.7.3-17)

    - Stuttgart Neural Network Simulator (SNNS -- RSNNS package v.0.4-14)

    - Extreme Learning Machine (ELM -- elmNNRcpp package v. 1.0.4)

    - Random Forest (RF -- randomForest package v.4.7-1.1)

    - Case-specific Random Forest (CSRF --- ranger package v 0.14.1)

    - Bagging tree method (BAG -- ipred package v.0.9-13)

    - Extreme Gradient Boosting (EGB --- xgboost v.1.6.0.1)

    - Generalized Boosted Regression Models (GBM -- gbm package v.2.1.8)

    - Generalized Linear Models (GLM)

    - GLM step-wise (GLM_sw)

    - Generalized Additive Models (GAM --package gam v. 1.20.2)

4. Model.eval_v2.3: Evaluated the performance of an ensemble model performed by combining calibrated prediction of step 3.

- EEM-tools: Tools to evaluate the EEM spectra. Not necessary for the analysis but useful to compare or validate some aspects of the EEMs.

Check_uv-absv1: check the values of Absorbance for all samples within the range of interest.

Compare_two EEM: compare the direct value of two EEMs.

RMSE_blank: perform root mean square error of all blank data.

Similarity_EEM_replicate: perform similarity test between EEM replicates.

- input example: contain several files necessary for performing the analysis or explaining how the input data are organized. **! Should be renamed `input` to be used in the working directory folder by the script.**

SPECTRA folder: contains raw EEM and UV-vis spectra (see description in https://doi.org/10.5281/zenodo.7244913).

PRODOM_spectra.DBPv1.csv: see metadata file on https://doi.org/10.5281/zenodo.7244913 for full explanation.

exclude.list.txt: list of sample names to exclude in the analysis separately by a carriage return/line feed (CR/LF).

var.list.txt: list of the input and output variables for the ML techniques. Contains the following header:

n: unique number from 1 to...

var.type: "x" or "y" for independent and dependent variables, respectively.

var.name: variables names explained in https://doi.org/10.5281/zenodo.7244913

var.transf: transformation function. The number reported corresponds to the following transformation:

1. no transformation
2. log
3. log(x+1)
4. x^2
5. sqrt(x)
6. 1/x
7. 1/(logx)
8. 1/x^2
9. 1/(sqrt(x))

The following header refers to the options for variables selected in the ML technique as a predictor (x) and to be predicted (y). The name of the header should be defined as similar to the parameter `list.var.sele` in the ML script. Several headers may be named. 0 and 1 are written in the line corresponding to unused and used variables respectively.