
Air Quality Toolkit Documentation

Release 1

Jack McKew

Oct 05, 2018

CONTENTS:

1	Stitcher	1
2	CSV Formatter	2
3	Factorizer	3
	Python Module Index	4

STITCHER

`Stitcher.Stitcher(filepath_list, filename, scale, headers_list, output_filename)`

This function sums multiple dataframes together and outputs a csv of the result.

Parameters

- **filepath_list** (*list[str]*) – This should be a list of filepaths to associated filename list.
- **filename** (*list[str]*) – This should be a list of filenames to associated filepath (filepath_list) list.
- **scale** (*list[float]*) – This should be a list of scalars to scale associated filename dataframe by.
- **headers_list** (*list[int]*) – This should be a list of numbers to exclude number of columns from dataframes.
- **output_filename** (*str.*) – Output filename.

CSV FORMATTER

`CSVFormatter.csvformatter(filename, olm_state, output_filename)`

This function formats dataset outputs from air quality modelling software to CSV format.

Parameters

- **filename** (*str.*) – This should be a string of the filename to convert to CSV.
- **olm_state** (*bool.*) – This should be a boolean value specifying if the file is OLM format (x and y information header).
- **output_filename** (*str.*) – Output filename.

FACTORIZER

`Factorizer.calcrow(row)`

This function is used for multiplying entire pandas dataframe single row by scalar value located in the last column of the dataframe.

Parameters `row (float.)` – This should be the slice from the pandas dataframe.

Returns `pd.Series` – The row multiplied by the scalar in the last column of the row

`Factorizer.factorizer(dataset_filename, factor_filename, output_filename)`

This function multiplies air quality datasets with vectors given.

Parameters

- **dataset_filename (str.)** – This should be a string of the filename containing the dataset.
- **factor_filename (str.)** – This should be a string of the filename containing the vector dataset to multiply with.
- **output_filename (str.)** – Output filename.

NO2PROCESSOR

`NO2Processor.process(header_length, initial, exceedance, background_name, input_data, output_filename)`

This function applies air quality modelling functions and generates statistics.

Parameters

- **header_length** (*int.*) – This should be an integer declaring how many header columns in the dataset
- **initial** (*float.*) – This should be a float declaring initial percentage to work with eg 0.1 = 10%
- **exceedance** (*int.*) – This should be an integer declaring how many exceedances eg compare if any are greater than 246
- **background_name** (*str.*) – This should be a string representing input background NO2 filename (Must be located in same directory as .exe).
- **input_data** (*str.*) – This should be a string representing input filename (Must be located in same directory as .exe).
- **output_filename** (*str.*) – Output filename.

STATISTICS GENERATOR

`Statistics_Generator.Statistics_Generator(settings, header_length, input_data, output_filename)`

This function generates statistics on given input datasets.

Parameters

- **settings** (*dict[str:int]*) – This should be a dictionary of settings with their name as the key and state as value.
- **header_length** (*int.*) – This should be an integer declaring how many header columns in the dataset
- **input_data** (*str.*) – This should be a string representing input filename (Must be located in same directory as .exe).
- **output_filename** (*str.*) – Output filename.

PYTHON MODULE INDEX

C

CSVFormatter, [2](#)

f

Factorizer, [3](#)

n

NO2Processor, [??](#)

s

Statistics_Generator, [??](#)

Stitcher, [1](#)