

Package: kwb.ogre.model (via r-universe)

October 31, 2024

Title Load Model Developed in OGRE Project

Version 0.0.0.9000

Description This package contains the functions written by Andreas Matzinger that were originally located in the script ``OgRe_LoadModel_functions.R". Since this script was copied by some students and then adapted there are many duplicated functions now. I (Hauke) will replace those functions in the copied scripts that did not change at all with ``links" to the original functions in this package.

License MIT + file LICENSE

URL <https://github.com/KWB-R/kwb.ogre.model>

BugReports <https://github.com/KWB-R/kwb.ogre.model/issues>

Imports kwb.odm, kwb.ogre, kwb.utils, dplyr, rlang

Remotes github::kwb-r/kwb.odm, github::kwb-r/kwb.ogre,
github::kwb-r/kwb.utils

Encoding UTF-8

LazyData TRUE

RoxygenNote 7.1.1

Suggests covr

Repository <https://kwb-r.r-universe.dev>

RemoteUrl <https://github.com/KWB-R/kwb.ogre.model>

RemoteRef HEAD

RemoteSha c2aac15f4751a25dd6ad600d5c14809e50ead75d

Contents

adapt_nondetect	2
annual_load_rain	3
annual_load_sewage	3
annual_mean_conc	4

annual_stats	4
default_statistics	5
detect	5
geom_mean	5
getNewDetectionLimits	6
get_lab_values	6
non_detect	6
no_Panke	7
only_composite	7
only_new_dl_metals	7
only_representative_subst	8
Panke	8
quant25	8
quant75	9
quant95	9
reduce_codes	9
remove_group	10

Index	11
--------------	-----------

adapt_nondetect	<i>adapts values of single results < detection limit.</i>
-----------------	--

Description

requires list of these substances as data.frame for each site (one column per site). If substance is always < dl at one site, results are set to zero. If substance is sometimes > dl, results are set to a factor*dl

Usage

```
adapt_nondetect(x_in, x_nd, factor = 0.5)
```

Arguments

x_in	name of input data.frame
x_nd	vector with substances < detection (one column per site)
factor	multiplier of detection limit if smaller dl (e.g., 0, 0.5 or 1)

annual_load_rain *calculates the load for each substance*

Description

separates pathways (rain runoff, CSO and WWTP)

Usage

```
annual_load_rain(data.dir, error_removal_rate = 0.3)
```

Arguments

data.dir path of model data (annual mean concentrations "annual_mean_conc.csv", rain runoff volumes "Vol_rain.csv", removal at WWTP "substance_info.csv")

error_removal_rate relative error in removal at WWTP

Value

Function returns list with loads and standard deviations, by entry path (sep, cso, wwtp) and by surface water catchment. Concentration in units "mg/L" and "ug/L" is automatically transformed to loads in "kg/yr". Other (unknown) units are left unchanged, resulting in "unit * m3/yr".

annual_load_sewage *calculates the load for each substance*

Description

separates pathways (CSO and WWTP)

Usage

```
annual_load_sewage(data.dir, error_removal_rate = 0.3, error_conc = 0.5)
```

Arguments

data.dir path of model data (annual mean concentrations "substance_info.csv", WWTP runoff volumes "Vol_sewage.csv", removal at WWTP "substance_info.csv", optional: relative error by substance can be indicated as additional column "error_conc" in "substance_info.csv")

error_removal_rate relative error in removal at WWTP

error_conc constant relative error in concentrations at WWTP outflow (default = 0.5) or "individual" if relative error by substance is included in "substance_info.csv"

Value

Function returns list with loads and standard deviations, by entry path (cso, wwtp) and by surface water catchment. Concentration in units "mg/L" and "ug/L" is automatically transformed to loads in "kg/yr". Other (unknown) units are left unchanged, resulting in "unit * m3/yr".

annual_mean_conc	<i>estimates annual mean concentrations per site.</i>
------------------	---

Description

Apart from a matrix with mean concentrations for each substance and site (= 0 if always below detection limit), matrices with N, standard error, standard deviation, RMSE, as well as measured min and max are calculated. Result is given as a list, of these matrices. Different methods can be chosen: Method 1: arithmetic mean Method 2: functions and RMSE from file, arithmetic mean for substances without functions

Usage

```
annual_mean_conc(x_in, method, data.dir)
```

Arguments

x_in	name of input data.frame
method	estimation method: 1 = arithmetic mean; 2 = functions and RMSE from file, arithmetic mean for substances without functions
data.dir	file directory where correlation data and rain series for method 2 are located

annual_stats	<i>estimates annual mean concentrations per site.</i>
--------------	---

Description

Apart from a matrix with mean concentrations for each substance and site (= 0 if always below detection limit), matrices with N, 95 measured min and max are calculated. Result is given as a list, of these matrices.

Usage

```
annual_stats(x_in)
```

Arguments

x_in	name of input data.frame
------	--------------------------

default_statistics	<i>calculate default statistics</i>
--------------------	-------------------------------------

Description

calculate default statistics for a grouped data frame (created with `dplyr::group_by`)

Usage

```
default_statistics(x)
```

Arguments

x	data frame with columns DataValue, CensorCode
---	---

detect	<i>removes substances, without detection from data.frame</i>
--------	--

Description

requires list of these substances as single vector

Usage

```
detect(x_in, x_nd)
```

Arguments

x_in	name of input data.frame
x_nd	vector with substances < detection

geom_mean	<i>function gives geometric mean</i>
-----------	--------------------------------------

Description

function gives geometric mean

Usage

```
geom_mean(x)
```

Arguments

x	vector of numeric values of which to calculate the geometric mean
---	---

getNewDetectionLimits *get detection limits for variables*

Description

that changed (lowered) during the monitoring

Usage

```
getNewDetectionLimits()
```

Value

data frame with columns VariableCode, DetectionLimit

get_lab_values *Reads all lab values from ODBC-source*

Description

Appends also site code (e.g., "NEU") and substance name apart from standard-fields the fields "CensorCode", "QualityControlLevelID", and "UnitsAbbreviation" are included. If one measurements exists twice, only higher QualityControlLevelID is included

Usage

```
get_lab_values(odbc_name)
```

Arguments

odbc_name Name of the odbc source

non_detect *lists substances without detection in any sample*

Description

Apart from entire dataset x_in (first column), lists are given for each site individually (following columns)

Usage

```
non_detect(x_in)
```

Arguments

x_in name of input data.frame

no_Panke	<i>removes rows in data.frame with site code = "PNK"</i>
----------	--

Description

removes rows in data.frame with site code = "PNK"

Usage

```
no_Panke(x_in)
```

Arguments

x_in	name of input data.frame
------	--------------------------

only_composite	<i>removes rows in data.frame with SampleType != "Composite"</i>
----------------	--

Description

removes rows in data.frame with SampleType != "Composite"

Usage

```
only_composite(x_in)
```

Arguments

x_in	name of input data.frame
------	--------------------------

only_new_dl_metals	<i>removes metal samples below detection limit (dl),</i>
--------------------	--

Description

when dl was too high (old analytical method). Works by VariableCode

Usage

```
only_new_dl_metals(x_in)
```

Arguments

x_in	name of input data.frame
------	--------------------------

only_representative_subst

removes Variables, which have not

Description

at least one measurement (can also be below detection limit) per catchment type

Usage

only_representative_subst(x_in)

Arguments

x_in name of input data.frame

Panke

keeps only rows in data.frame with site code = "PNK"

Description

keeps only rows in data.frame with site code = "PNK"

Usage

Panke(x_in)

Arguments

x_in name of input data.frame

quant25

function gives the 25 percent quantile

Description

function gives the 25 percent quantile

Usage

quant25(x)

Arguments

x vector of numeric values of which to calculate the quantile

quant75	<i>function gives the 75 percent quantile</i>
---------	---

Description

function gives the 75 percent quantile

Usage

```
quant75(x)
```

Arguments

x vector of numeric values of which to calculate the quantile

quant95	<i>function gives the 95 percent quantile</i>
---------	---

Description

function gives the 95 percent quantile

Usage

```
quant95(x)
```

Arguments

x vector of numeric values of which to calculate the quantile

reduce_codes	<i>removes lines with censor codes</i>
--------------	--

Description

other than "lt" and "nc" (e.g., "???" are removed)

Usage

```
reduce_codes(x_in)
```

Arguments

x_in name of input data.frame

remove_group	<i>removes measurements of Variables of a specific group</i>
--------------	--

Description

removes measurements of Variables of a specific group

Usage

```
remove_group(x_in, group)
```

Arguments

x_in	name of input data.frame
group	Variable group to be removed (string)

Index

adapt_nondetect, 2
annual_load_rain, 3
annual_load_sewage, 3
annual_mean_conc, 4
annual_stats, 4

default_statistics, 5
detect, 5

geom_mean, 5
get_lab_values, 6
getNewDetectionLimits, 6

no_Panke, 7
non_detect, 6

only_composite, 7
only_new_dl_metals, 7
only_representative_subst, 8

Panke, 8

quant25, 8
quant75, 9
quant95, 9

reduce_codes, 9
remove_group, 10