

Package: reasonabletools (via r-universe)

August 20, 2024

Title Clean Water Quality Data for NPDES Reasonable Potential Analyses

Version 0.1

Date 2020-10-22

Depends R (>= 4.0), stats

Maintainer Matthew Reusswig <matt.reusswig@gmail.com>

BugReports <https://github.com/mattreusswig/reasonabletools/issues>

URL <https://github.com/mattreusswig/reasonabletools>

Description Functions for cleaning and summarising water quality data for use in National Pollutant Discharge Elimination Service (NPDES) permit reasonable potential analyses and water quality-based effluent limitation calculations. Procedures are based on those contained in the ``Technical Support Document for Water Quality-based Toxics Control'', United States Environmental Protection Agency (1991).

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

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

RemoteUrl <https://github.com/mattreusswig/reasonabletools>

RemoteRef HEAD

RemoteSha f3d8a53732ef9d49f9e9960842b8a959098fd371

Contents

calc_AML	2
calc_MDL	3
calc_WLA	4
cen_sum	5

cv_adj	5
find_mec	6
fuse_samples	7
mrg_cen_cols	8
project_mec	9

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

calc_AML	<i>Compute AML</i>
----------	--------------------

Description

Compute average monthly effluent limitation (AML) according to methods in EPA's Technical Support Document for Water Quality-based Toxics Control.

Usage

```
calc_AML(
  WLAa,
  WLAc,
  WLAhh,
  cv,
  n_samples = 4,
  prob_LTA = 0.99,
  percentile_AML = 0.95
)
```

Arguments

WLAa	Numeric. Acute wasteload allocation (WLA). All WLA units should be identical.
WLAc	Numeric. Chronic wasteload allocation (WLA). All WLA units should be identical.
WLAhh	Numeric. Human health wasteload allocation (WLA). All WLA units should be identical.
cv	Numeric. Coefficient of variation (CV) of effluent data. See cv_adj function.
n_samples	Numeric. Number of sample observations.
prob_LTA	Numeric (fraction). Allowable exceedance probability of the WLA used to estimate long-term average (LTA).
percentile_AML	Numeric (fraction). Lognormal distribution location for AML.

Value

Numeric value in same units as the WLAs.

Examples

```
calc_AML(WLAa=4, WLAc=1, WLAhh=10, cv=0.6)
```

 calc_MDL

Compute MDL

Description

Compute Maximum Daily Effluent Limitation (MDL) according to methods in EPA's Technical Support Document for Water Quality-based Toxics Control.

Usage

```
calc_MDL(
  WLAa,
  WLAc,
  WLAhh,
  cv,
  n_samples = 4,
  prob_LTA = 0.99,
  percentile_MDL = 0.99,
  percentile_AML = 0.95
)
```

Arguments

WLAa	Numeric. Acute wasteload allocation (WLA). All WLA units should be identical.
WLAc	Numeric. Chronic wasteload allocation (WLA). All WLA units should be identical.
WLAhh	Numeric. Human health wasteload allocation (WLA). All WLA units should be identical.
cv	Numeric. Coefficient of variation (CV) of effluent data. See <code>cv_adj</code> function.
n_samples	Numeric. Number of sample observations.
prob_LTA	Numeric (fraction). Allowable exceedance probability of the WLA used to estimate long-term average (LTA).
percentile_MDL	Numeric (fraction). Lognormal distribution location for MDL.
percentile_AML	Numeric (fraction). Lognormal distribution location for AML.

Value

Numeric value in same units as the WLAs.

Examples

```
calc_MDL(WLAa=4, WLAc=1, WLAhh=10, cv=0.6)
```

 calc_WLA

Compute Wasteload Allocation

Description

Compute the wasteload allocation (WLA) used for effluent limit calculation.

Usage

```
calc_WLA(criteria, background, Qrsw, Qeff)
```

Arguments

criteria	Limiting water quality criterion. Must be in same units as background argument.
background	Background pollutant concentration. Must be in same units as criteria argument.
Qrsw	Upstream limiting/design receiving water flowrate. Must be in same units as Qeff argument. Flow arguments may be entered in ratio form.
Qeff	Effluent limiting/design flowrate. Must be in same units as Qrsw argument. Flow arguments may be entered in ratio form.

Value

WLA as numeric value in same units as criteria and background.

Examples

```
# WLA for pollutant with 2 ug/L acute criteria and upstream receiving water concentration
# of 0.1 ug/L. The critical flows are 3 MGD (1Q10) and 0.5 MGD (max daily flow).
calc_WLA(2, 0.1, 3, 0.5)
```

```
# When using dilution credits, put Qrsw and Qeff in terms of the dilution ratio (D).
D = 7 # Assume a jurisdiction that uses D = (Qrsw + Qeff) / Qeff
Qeff = 1 # Equal to 1 since its the denominator of the ratio, or you can use the critical flow
Qrsw = D - 1 # Same as the expression (D * Qeff) - Qeff
calc_WLA(2, 0.1, Qrsw = D - 1, Qeff = 1)
```

cen_sum	<i>Sum detected and non-detect concentrations.</i>
---------	--

Description

Sum up congener sample concentrations to create a composite parameter value. Primarily intended as a helper function in fuse_samples.

Usage

```
cen_sum(qual, result, nd = c("<", "nd", "ND"))
```

Arguments

qual	A character vector containing non-detect indicator strings, e.g., "<" or "ND". The strings used to indicate censored status can be edited in the "nd" argument.
result	A numeric vector of concentration measurements.
nd	A list indicating all the censoring flags included in the dataset. Defaults to "<", "nd", and "ND".

Value

dataframe with two columns, qual (character) and result (numeric)

cv_adj	<i>Find the adjusted coefficient of variation</i>
--------	---

Description

Calculates adjusted coefficient of variation (CV) according to methods described in EPA's Technical Support Document for Water Quality-based Toxics Control.

Usage

```
cv_adj(qual, result, nd = c("<", "nd", "ND"), nd_adjustment = 0.5)
```

Arguments

qual	A character vector containing non-detect indicator strings, e.g., "<" or "ND". The strings used to indicate censored status can be edited in the "nd" argument.
result	A numeric vector of concentration measurements.
nd	A list indicating all the censoring flags included in the dataset. Defaults to "<", "nd", and "ND".
nd_adjustment	Adjustment factor for non-detect values. Non-detect values (as indicated in qual vector) are multiplied by nd_adjustment factor; i.e. result * nd_adjustment. Typically, method detection limits or reporting limits are used as result values for non-detects.

Value

A numeric coefficient of variation (CV) value

Examples

```
# CV for all detected values
cen_result <- rep("", 10)
result      <- c(1:10)
cv_adj(cen_result, result)

# CV for all non-detected values
cen_result <- rep("<", 10)
cv_adj(cen_result, result)

# CV for fewer than 10 measurements
cen_result <- rep("", 5)
result <-      c(1:5)
cv_adj(cen_result, result)

# Change the default substitution value
cen_result <- c(rep("<", 5), rep("", 15))
result      <- c(101:120)
cv_adj(cen_result, result) # Use default 0.5 multiplier
# Use 1.0 multiplier (equivalent to using MDL)
cv_adj(cen_result, result, nd_adjustment = 1.0)
# Use 0.0 multiplier (equivalent to zero substitution)
cv_adj(cen_result, result, nd_adjustment = 0)
```

find_mec

Find the Maximum Observed Effluent Concentration (MEC)

Description

Find the MEC (no projection) from the observed dataset using methods described in EPA's Technical Support Document for Water Quality-based Toxics Control.

Usage

```
find_mec(qual, result, nd = c("<", "nd", "ND"), simple_output = FALSE)
```

Arguments

qual	A character vector containing non-detect indicator strings, e.g., "<" or "ND". The strings used to indicate censored status can be edited in the "nd" argument.
result	A numeric vector of concentration measurements.
nd	A list indicating all the censoring flags included in the dataset. Defaults to "<", "nd", and "ND".
simple_output	Logical value. If TRUE, the output columns are concatenated into a single character string (e.g., "<0.2") which can be useful for constructing summary tables.

Value

Dataframe with a qualifier column (character) and a MEC column (numeric).

Examples

```
# Find observed MEC
cen_result <- c(rep("", 10), rep("<", 10))
result <- 1:20
find_mec(cen_result, result)

cen_result <- rep("<", 20)
find_mec(cen_result, result)

cen_result <- rep("", 20)
find_mec(cen_result, result)

# Demonstrate simplified output
find_mec(cen_result, result, simple_output = TRUE)

# Define a set of custom non-detect flags
cen_result <- c(rep("non-detect", 5), rep("<", 10), rep("mdl", 5))
find_mec(cen_result, result, nd = c("non-detect", "<", "mdl"))
```

fuse_samples

Combine values for a composite parameter

Description

Calculate composite parameter concentrations using congener concentrations grouped by a sampling date vector. An example would application would be summing PCB congeners collected on a specific sampling date to produce a total PCBs concentration.

Usage

```
fuse_samples(date_grp, qual, result, nd = c("<", "nd", "ND"))
```

Arguments

date_grp	A date vector to group the dataset.
qual	A character vector containing non-detect indicator strings, e.g., "<" or "ND". The strings used to indicate censored status can be edited in the "nd" argument.
result	A numeric vector of concentration measurements.
nd	A list indicating all the censoring flags included in the dataset. Defaults to "<", "nd", and "ND".

Value

A dataframe containing a column of sampling dates used to group data, a qualifier column (character), and a MEC result column (numeric).

Examples

```
df <- data.frame(dates = rep(seq.Date(from = as.Date("1982-03-10"),
                                     to = as.Date("1982-03-15"),
                                     by = 1), 5),
                congeners = sort(rep(LETTERS[1:6], 5)),
                qualifier = sample(c("<", ""), size = 30,
                                  replace = TRUE, prob = c(0.8, 0.2)),
                result = sample(seq(0.1, 0.5, 0.1), 30, replace = TRUE))

fuse_samples(df$dates, df$qualifier, df$result)
```

mrg_cen_cols

Merge together observed concentrations and detection limits

Description

Merge a results column with a detection limits column by overwriting the censored results values with the corresponding detection limit.

Usage

```
mrg_cen_cols(qual, result, limit, nd = c("<", "nd", "ND"))
```

Arguments

qual	A character vector containing non-detect indicator strings, e.g., "<" or "ND". The strings used to indicate censored status can be edited in the "nd" argument.
result	A numeric vector of concentration measurements.
limit	A numeric vector of method detection limit/reporting limit values.
nd	A list indicating all the censoring flags included in the dataset. Defaults to "<", "nd", and "ND".

Value

A numeric vector.

Examples

```

result <- runif(5, 1, 10)
cen_result <- sample(c("<", ""), length(result), replace = TRUE)
method_limit <- rep(0.1, length(result))
result <- mrg_cen_cols(cen_result, result, method_limit)
print(data.frame(cen_result, result, stringsAsFactors = FALSE))

```

project_mec

Find the projected Maximum Observed Effluent Concentration (MEC)

Description

Find the MEC projected from a lognormal distribution using methods described in EPA's Technical Support Document for Water Quality-based Toxics Control.

Usage

```

project_mec(
  qual,
  result,
  nd = c("<", "nd", "ND"),
  percentile = 0.95,
  conf_level = 0.99,
  nd_adjustment = 0.5,
  simple_output = FALSE
)

```

Arguments

qual	A character vector containing non-detect indicator strings, e.g., "<" or "ND". The strings used to indicate censored status can be edited in the "nd" argument.
result	A numeric vector of concentration measurements.
nd	A list indicating all the censoring flags included in the dataset. Defaults to "<", "nd", and "ND".
percentile	Location on lognormal distribution to estimate the projected MEC.
conf_level	Confidence level of projected estimate.
nd_adjustment	Adjustment factor for non-detect values. Non-detect values (as indicated in qual vector) are multiplied by nd_adjustment factor; i.e. result * nd_adjustment. Typically, method detection limits or reporting limits are used as result values for non-detects.
simple_output	Logical value. If TRUE, the output columns are concatenated into a single character string (e.g., "<0.2") which can be useful for constructing summary tables.

Value

Dataframe with a qualifier column (character) and a MEC column (numeric).

Examples

```
# Find observed MEC
cen_result <- c(rep("", 10), rep("<", 10))
result     <- 1:20
project_mec(cen_result, result)

# Demonstrate simplified output
cen_result <- rep("<", 20)
project_mec(cen_result, result, simple_output = TRUE)

# Define a set of custom non-detect flags
cen_result <- c(rep("non-detect", 5), rep("<", 10), rep("mdl", 5))
project_mec(cen_result, result, nd = c("non-detect", "<", "mdl"))

# Change the substitution multiplier used for non-detect values
cen_result <- c(rep("<", 5), rep("", 15))
result     <- c(101:120)
project_mec(cen_result, result) # Use default 0.5 multiplier

# Use 1.0 multiplier (equivalent to using MDL)
project_mec(cen_result, result, nd_adjustment = 1.0)

# Use 0.0 multiplier (equivalent to zero substitution)
project_mec(cen_result, result, nd_adjustment = 0)
```

Index

calc_AML, [2](#)
calc_MDL, [3](#)
calc_WLA, [4](#)
cen_sum, [5](#)
cv_adj, [5](#)

find_mec, [6](#)
fuse_samples, [7](#)

mrg_cen_cols, [8](#)

project_mec, [9](#)