Package: kwb.fcr (via r-universe)

November 2, 2024

Title Fertilizer chemical risk assessment

Version 0.0.0.9000

Description This risk assessment is based on the TGD on risk assessment by the European Comission. Possible Endpoints are soil organisms, groundwater organisms and human health via food consumption.Every variable can be entered as probability distribution to include uncertainties or site specific variations.

License MIT + file LICENSE

URL https://github.com/KWB-R/kwb.fcr

BugReports https://github.com/KWB-R/kwb.fcr/issues

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Suggests covr, knitr, rmarkdown

VignetteBuilder knitr

Imports readxl, magrittr

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

RemoteUrl https://github.com/KWB-R/kwb.fcr

RemoteRef HEAD

RemoteSha d275c0ba804b49c4a5afc9726a10315913c2496a

Contents

`add_AirWater` *Estimate Air-Water partition coefficient*

Description

Estimation is based on the FCR prepared Monte Carlo table

Usage

```
add_AirWater(p)
```

Arguments

`p`	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the partition coefficient. Unit is m³/m³.

`add_bcf` *Estimate bio concentration factor*

Description

Estimation is based on the FCR prepared Monte Carlo table

Usage

```
add_bcf(p, sub_info)
```

Arguments

`p`	Parameter table created with `oneYear_matrix()`
`sub_info`	The table containing additional substance information loaded with `read_fcr_input()`

Value

The Parameter table extended by a column for the bio concentration factor.

`add_deposition` *Calculate soil mass related daily atmospheric deposition*

Description

Calculation is based on the FCR prepared Monte Carlo table

Usage

```
add_deposition(p)
```

Arguments

`p`	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the atmospheric Deposition. Unit is mg/(kg * d).

add_DT50 *Estimate biological half-life*

Description

Estimation is based on the FCR prepared Monte Carlo table

Usage

```
add_DT50(p, sub_info)
```

Arguments

p	Parameter table created with oneYear_matrix()
sub_info	The table containing additional substance information loaded with read_fcr_input()

Details

The estimation is based on the K_d value of a substance according to the technical guidance document on risk assessment Part II (table 8). If there is a distribution of K_d values, the median K_d is used for the estimation. In this function there is no distinction between biodegradbility classes. All substances are assumend to be inherently biodegradable to encourage the user of further half-life research.

Value

The Parameter table extended by a column for the bio concentration factor.

add_Henry *Estimate Henry Coefficient*

Description

Estimation is based on the FCR prepared Monte Carlo table

Usage

```
add_Henry(p)
```

Arguments

p	Parameter table created with oneYear_matrix()

Value

The Parameter table extended by a column for Henry coefficient. Unit is (Pa*m3)/mol

add_kbio *Calculate biodegredation rate*

Description

Calculation is based on the FCR prepared Monte Carlo table

Usage

```
add_kbio(p)
```

Arguments

p	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the biodegredation rate. Unit is 1/d.

add_Kd *Estimate Sorption Coefficient*

Description

Estimation is based on the FCR prepared Monte Carlo table

Usage

```
add_Kd(p, sub_info)
```

Arguments

p	Parameter table created with `oneYear_matrix()`. Note: D_air variable must be available in p if K_d is estimated with linear regression.
sub_info	The table containing additional substance information loaded with `read_fcr_input()`

Details

Theoretically the initial concentration in soil can become negative if the pollutant concentration in fertilizes is negative. A negative concentration makes obviously no sense in a real situation. However, long-term impact is assessed, a broad concentration range of the pollutant also leads to more uncertain results. Negative concentrations in one year are averaged out. For the regression of the sorption coefficient, the concentration must be positive. Thus, negative values are increased to the concentration in top soil that would appear after one day of deposition.

Value

The Parameter table extended by a column for the sorption coefficient

add_kleach *Calculate infiltration rate*

Description

Calculation is based on the FCR prepared Monte Carlo table

Usage

```
add_kleach(p)
```

Arguments

p	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the infiltration rate. Unit is 1/d.

add_kplant *Calculate plant uptake rate*

Description

Calculation is based on the FCR prepared Monte Carlo table

Usage

```
add_kplant(p)
```

Arguments

p	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the plant uptake rate. Unit is 1/d.

add_kvolat *Calculate volatilization rate*

Description

Calculation is based on the FCR prepared Monte Carlo table

Usage

```
add_kvolat(p)
```

Arguments

p	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the volatilization rate. Unit is 1/d.

add_PNEC_soil *Estimate Predicted no-effect concentration for soil organisms*

Description

Calculation is based on the FCR prepared Monte Carlo table

Usage

```
add_PNEC_soil(p)
```

Arguments

p	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the Predicted no-effect concentration for soil organisms. Unit is mg/(kg Dry Matter).

add_SoilWater *Estimate Soil-Water partition coefficient*

Description

Estimation is based on the FCR prepared Monte Carlo table

Usage

```
add_SoilWater(p)
```

Arguments

p	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the partition coefficient. Unit is m³/m³.

add_SoilWater_reverse *Estimate Soil-Water partition coefficient starting from infiltration rate*

Description

Estimation is based on the FCR prepared Monte Carlo table

Usage

```
add_SoilWater_reverse(p)
```

Arguments

p	Parameter table created with `oneYear_matrix()`

Value

The Parameter table extended by a column for the partition coefficient. Unit is m³/m³.

add_variables *Query and complementation of input variables*

Description

Query and complementation of input variables

Usage

```
add_variables(p, info)
```

Arguments

p	The input data table created with read_fcr_input()
info	Additional input data information, created with read_fcr_input()

Value

The table p exetended by new variable columns needed for the risk assessment

BCF_regression *Estimate the bio concentration factor in two different ways*

Description

Estimation via direct or indirect logarithmic (log 10?) linear regression including pH, organic carbon and soil concentration

Usage

```
BCF_regression(
  constant,
  beta_ph,
  beta_org,
  beta_conc,
  regType = "direct",
  pH,
  org_c,
  conc
)
```

Arguments

constant	Log constant in linear Kd regression
beta_ph, beta_org, beta_conc	Log coefficient for pH, organic carbon and soil concentration in log-linear Kd regression
regType	Chacter specifying the way the K_d value is calculated. Either "direct" which means that the regression directly leads to the sorption coefficient or"indirect" which means the regression estimates a concentration in pore water and the sorption coefficient is subsequently be calculated by the quotient of soil and pore water concentration.
pH	pH value
org_c	Organic carbon content in percent.
conc	Soil concentration in mg/kg

Details

The constant value, the regression parameters (beta) and the soil characteristics can be single numeric values or vectors of the same length.

Value

A vector of bio concentration factors without unit. It is defined as concentration in plant dry matter divided by concentration in soil.

---

create_mcs_input *List of Variables into Monte Carlo simulation input table*

Description

List of Variables into Monte Carlo simulation input table

Usage

```
create_mcs_input(data_list, nFields)
```

Arguments

data_list	A list of input variable tables. Each table must have columns "value_1", "value_2", "distribution", "site_specific", "pID" and "shift"
nFields	Number of Monte Carlo Cycles

Value

A matrix with all input variables from the list

CumSumSoil — *Plot cumulative sum of RQ after application and compare it with initial state*

Description

Plot cumulative sum of RQ after application and compare it with initial state

Usage

```
CumSumSoil(v0, v, year_x, xmin = 0.001, xmax = 10)
```

Arguments

v0	Vector of initial risk quotient distribution
v	Vector of final risk quotient distribution of x years
year_x	Numarical value of the last year of the risk assessment
xmin	Minimum value of x-Axis
xmax	Maximumg value of x-Axis

Value

Cumulative sums are plotted and the underlying table of quantiles of the initial and final risk quotients

get_c_i_from_PNEC_porewater — *Calculate the initial concentration in soil for RQ porewater = 1*

Description

The initial soil concentration is calculated so that the average concetrantion of the first 30 days leads to a PEC_water that equals the PNEC water

Usage

```
get_c_i_from_PNEC_porewater(dat, info, nFields)
```

Arguments

dat	List with all the input variables. This list is produced by function read_fcr_input() from the Excel sheets.
info	he table containing additional substance information loaded with read_fcr_input()
nFields	Number of Monte Carlo cycles

Details

The equation is a combination of the definition for PEC soil and PEC porewater the from TGD, back-calculating to c0.

Value

Numeric vector of lenght nFields, containig the initial concentrations in µg/L

---

get_PEC_human *Predicted human consumption*

Description

Predicted human consumption via crop consumption. The calculation is based on the average predicted soil concentration within the growing period.

Usage

```
get_PEC_human(p, d, food_only)
```

Arguments

p	Parameter table created with `oneYear_matrix()` and extended with `add_variables()`
d	The number of days of the growing period.
food_only	If TRUE, the predicted human consumption via food is multiplied by 2 to compensate the disregard of water consumption

Value

Numeric vector of human consumption via crops in µg/d

---

get_PEC_porewater *Predicted pore water concentration*

Description

Average predicted environmental concentration (PEC) over d days after fertilizer application. The Calculation is based on the FCR prepared Monte Carlo table

Usage

```
get_PEC_porewater(p, d)
```

Arguments

p	Parameter table created with oneYear_matrix() and extended with add_variables()
d	The number of days after fertilizer application. Somewhere between 1 and the number of days of the growing period, since plant uptake is considered

Value

Numeric vector of concentrations per field in µg/L

---

get_PEC_soil *Predicted soil concentration*

Description

Average predicted environmental concentration (PEC) over d days after fertilizer application. The Calculation is based on the FCR prepared Monte Carlo table

Usage

```
get_PEC_soil(p, d)
```

Arguments

p	Parameter table created with oneYear_matrix()
d	The number of days after fertilizer application. Somewhere between 1 and the number of days of the growing period, since plant uptake is considered

Value

Numeric vector of concentrations per field in mg/kg soil

---

get_risk *Risk quotients and high risk scenarios with and without fertilization*

Description

Risk quotients and high risk scenarios with and without fertilization

Usage

```
get_risk(fertPEC, noFertPEC, PNEC)
```

Arguments

fertPEC	A vector of predicted environmental concentrations at the end of simulation with fertilization
noFertPEC	A vector of predicted environmental concentrations at the end of simulation without fertilization. Both vectors must be based on the same environmental conditions during simulation.
PNEC	A numeric value, defining the predicted no-effect concentration

Value

Data frame with PECs, Risk Quotient and identification of high-risk scenarios for both simulations, with and without fertilization, as well as the risk quotient difference due to fertilization

---

Kd_regression *Estimate the soil sorption coefficient in two different ways*

Description

Estimation via direct or indirect logarithmic linear regression including pH, organic carbon and soil concentration. (log10 is used for all variables)

Usage

```
Kd_regression(
  constant,
  beta_ph,
  beta_org,
  beta_conc,
  regType = "direct",
  pH,
  org_c,
  conc
)
```

Arguments

constant	Log constant in linear Kd regression
beta_ph, beta_org, beta_conc	Log coefficient for pH, organic carbon and soil concentration in log-linear Kd regression
regType	Chacter specifying the way the K_d value is calculated. Either "direct" which means that the regression directly leads to the sorption coefficient or"indirect" which means the regression estimates a concentration in pore water and the sorption coefficient is subsequently be calculated by the quotient of soil and pore water concentration.

pH	pH value
org_c	Organic carbon content in percent.
conc	Soil concentration in mg/kg

Details

The constant value, the regression parameter (beta) and the soil characteristics can be single numeric values or vectors of the same length.

Value

A vector of sorption coefficients in L/kg

---

longterm_PEC *PECs for soil, porewater and human consumption over a long period of time*

Description

Based on the input data provided in Excel sheets, 3 different PECs are calculated. The number of years and number of agricultural fields with different properties (Number of Monte Carlo cycles) can be chosen.

Usage

```
longterm_PEC(
  dat,
  info,
  years,
  nFields,
  use_mixing_factor = FALSE,
  PNECwater_c_i = FALSE,
  food_only = TRUE,
  growing_period = 180,
  t_res = 1,
  traceBackVariables = FALSE,
  keep_c_course = TRUE
)
```

Arguments

dat	A List with all the input variables. This list is produced by function read_fcr_input() from the Excel sheets.
info	A table containing additional substance information loaded with read_fcr_input()
years	Years of fertilizer application
nFields	Number of Monte Carlo cycles

use_mixing_factor
: Not working yet! If TRUE, porewater is diluted by pollutant free groundwater for a more realistic estimation of risks in groundwater. However the TGD approach assumes porewater = groundwater for the assessment.

PNECwater_c_i
: If TRUE, the initial concentration in soil is derived by a RQ = 1 for groundwater risk assessment (PEC_porewater = PNEC_porewater). Thus, the inital concentration is defined by the applied PNEC for groundwater.

food_only
: If TRUE, the predicted human consumption via food is multiplied by 2 to compensate the disregard of water consumption

growing_period
: Numeric value specifiyng the number of days with plant growth

t_res
: Temporal resolution in days to be returned in the output matrix (no effect on the calculation). The default is 1 (every day) which can lead to a very large output matrix. The maximal reslution is 365 days, since every year is calculated seperately.

traceBackVariables
: If TRUE, the variables of every simulation year are returned. Note: this can lead to very large lists.

keep_c_course
: If FALSE, the course of concentration is not part of the output. In that case variabel t_res is meaningless.

Value

List with

1. table for all PEC types per year,
2. table for concentration course in top soil in the predefined temporal resolution,
3. optional: If traceBackVariables TRUE: list of distributed input variables for each year, if traceBackVariables FALSE: Input variables of the first year including initial situations for human consumption and inital concentration in porewater

---

oneYear_matrix *Creates a matrix with all Input Data for one year Concentration calculation*

Description

The information of the Excel input files are used to prepare a Monte Carlo based data input for the calculation of a yearly concentration dynamic.

Usage

```
oneYear_matrix(dat, c_i, nFields, use_mixing_factor = FALSE)
```

Arguments

dat	List with all the input variables. This list is produced by function read_fcr_input() from the Excel sheets.
c_i	The initial concentration before fertilizer application
nFields	The number Monte Carlo Simulations. This can be seen as different agriculatural fields with different charactersitics.
use_mixing_factor	If TRUE, a mixing factor for porewater dilution by unpolluted groundwater is used to get the Risk quotient for Groundwater. The defahutl is FALSE. In that case pore water concentration is identical with groundwater concentration as is the approach in the Technical Guidance Document

Details

The pollutant concentration at t = 0 (c_0) already includes the fertilizer application

Value

A Matrix with all available input data for the Assessment. The columns represent the input variabls, the rows correspond to the definied number of fields aka. number of Monte Carlo cycles.

---

one_year *Calculation of the pollutant concentation course in top soil*

Description

Estimation is based on the FCR prepared Monte Carlo table. The output rate is split into one rate with plant uptake during plant growth and one rate without plant uptake during winter time.

Usage

```
one_year(p, growing_period, t_res)
```

Arguments

p	Parameter table created with oneYear_matrix()
growing_period	Numeric value specifiyng the number of days with plant growth
t_res	Temporal resolution to be returned in the output matrix (no effect on the calculation)

Details

A time period can be considered as growing period if 1) plants grow and 2) plants will be harvested after growing. Otherwise the uptaken pollutants will return to the soil.

Value

Matrix with number of columns equal to the rows of p and number of rows depending on the defined timesteps with t_res

---

`rderived` *Distribution derived by comparable circumstances*

Description

This function combines a uniform distribution from minimum to maximum with the upper and lower tail of a normal distribution.

Usage

```
rderived(n, min, max, a = -Inf)
```

Arguments

`n`	Number of samples to be drawn
`min`	Minimum value of normal comparables
`max`	Maximum value of normal comparables
`a`	Minimum of distribution (Default is -Inf)

Details

95%, 2.5% and 2.5% of all drawn values originate from the uniform distribution, the lower tail, and the upper tail of a normal distribution, respectively. The normal distribution is characterized by the mean of minimum and maximum. The standard deviation is derived from the 2.5% and 97.5% quantile definied by the minimum and maximum values. (sd = (max - mean) / 1.959963)

The number of Fields is multiplied by 38 for the uniform distribution, while it is multiplied by 1 for the tails of the normal distribution (38/40 = 95%). Finally, the defined number of fields is randomly drawn from the vector.

Value

numeric vector of length n with randomly drawn values according to the distribution

`rdist` *Random samples from specified distribution*

Description

Draw n samples from uniform, normal, truncated normal (-> positive only), lognormal or gamma distributions based on two input values

Usage

```
rdist(dist_name, value_1, value_2, n, shift = 0, seed = NULL)
```

Arguments

`dist_name`	Character vector specifying the Name of the distribution. Either "none", "uniform", "normal" "tnormal" (for truncated), "lognormal", "gamma", "derived", "tderived" or "logderived"
`value_1, value_2`	Distribution parameters based on dist_name: "none": value_1 is used as constantvalue, value_2 not used "normal": value_1 is mean, value_2 is standard deviation "tnormal": value_1 is mean, value_2 is standard deviation "lognormal": value_1 is log mean, value_2 is standard log deviation (both natural logarithms) "gamma": value_1 is shape, value_2 is rate "uniform": value_1 is minimum, value_2 is maximum "derived": value_1 is minimum of uniform distribution, value_2 is maximum of uniform distribution. Uniform distribution accounts for 95% of all data "tderived": value_1 is minimum of uniform distribution, value_2 is maximum of uniform distribution. Uniform distribution accounts for 95% of all data "logderived": value_1 is minimum of log-uniform distribution, value_2 is maximum of log-uniform distribution (values not log-scaled). Log-Uniform distribution accounts for 95% of all data
`n`	Number of samples to be drawn
`shift`	An numeric value defining a subsequent shift of the distribution The default is 0 (-> no shift)
`seed`	A numeric value to set the seed for random selection. The default is NULL -> no seed

Details

The values specified in value_1 and value_2 represent min and max for "uniform", mean and standard deviation for "normal" and "tnormal", log mean and log standard deviation (both natural logarithm) for "lognormal", shape and rate for"gamma". The shift value is especially interesting for lognormal or gamma distributions.

Value

Numeric vector of length n with randomly drawn samples of the specified distribution

---

`read_fcr_input` *Read environmental and pollutant variables*

Description

Reads Excel files with environmnetal proporties and substance coefficents.

Usage

```
read_fcr_input(input_path, pollutantName, siteName, fertilizerName = "none")
```

Arguments

`input_path`	Path of the folder "pollutants" and "sites" that contain the input data excel sheets
`pollutantName`	Name of Pollutant as in filename of the pollutant (prefix before "_sheet.xslx)
`siteName`	Name of the site as in filename of the environment sheet (prefix before "_sheet.xslx)
`fertilizerName`	Name of the fertilizer as in filename of the fertilizer sheet (prefix before "_sheet.xslx). If fertilizerName = "none", fertilizer application and pollutant concentration will be set to 0.

Details

Both input files must have the same path. The pollutant excel files must be named "'pollutant-Name'_sheet.xlsx". Furthermore, both excel tables must have the same number of columns and column names.

Value

List of two items names "dat" and "info" and containing all the input variable data (information about distribution type, distribution parameters, type of variability and a unique variable ID) and additional information of how to simulate substance behavior.

risk_aggregation *Risk Aggregation Calculation of Delta Risk Quotient and Maximum Risk*

Description

Risk Aggregation Calculation of Delta Risk Quotient and Maximum Risk

Usage

```
risk_aggregation(df_risk)
```

Arguments

df_risk	A dataframe created by get_risk()

Details

The Delta Risk Quotient (RQ) is the average increase of high-risk scenarios caused by fertilization. A high-risk scenarios are defined as scenraios with a final RQ higher than 1. The increase refers to a comparison between two similar scenarios, the only difference is the application of fertilizer.

The maximum risk is the average RQ of the upper 5% of the RQ-Distribution. It is meant to describe the upper tail of the distribution.

Further information can be found in the report of the Eurpean Horizon 2020 project Nextgen "Assessment and risk analysis of NextGen demo case solutions" (p.151)

Nextgen: Deliverable 2.1

Value

A list of 2, the delta risk quotient and the maximum risk

risk_interpretation *Risk Interpretation Assigns a risk to the aggregated risk parameters*

Description

Risk Interpretation Assigns a risk to the aggregated risk parameters

Usage

```
risk_interpretation(delta_RQ, risk_max)
```

Arguments

`delta_RQ`	Numeric vector of length 1: Difference of of high-risk simulations between scenarios with and without fertilization
`risk_max`	Numeric vector of length 1: Mean value of the Upper five percent of RQ distribution

Details

Further information can be found in the report of the Eurpean Horizon 2020 project Nextgen "Assessment and risk analysis of NextGen demo case solutions" (p.151)

Nextgen: Deliverable 2.1

Value

A list of 2, the delta risk quotient and the maximum risk

`rlderived` *Log-Distribution derived by comparable circumstances*

Description

This function draws random values from a uniform distribution (minimum to maximum) combined with the upper and lower tail of a log normal distribution. Minimum and maximum values are transformed to a log10 scale first.

Usage

```
rlderived(n, min, max)
```

Arguments

`n`	Number of samples to be drawn
`min`	Minimum value of normal comparables
`max`	Maximum value of normal comparables

Details

95%, 2.5% and 2.5% of all drawn values originate from the log-uniform distribution, the lower tail, and the upper tail of a log-normal distribution, respectively. The log-normal distribution is characterized by the mean of log-minimum and log-maximum. The standard deviation is derived from the 2.5% and 97.5% quantile defined by the minimum and maximum values. sd = (log10(max) - log10(mean)) / 1.959963

The number of Fields is multiplied by 38 for the uniform distribution, while it is multiplied by 1 for the tails of the log-normal distribution (38 / 40 = 95%) Finally, the defined number of fields is randomly drawn from the vector.

Value

numeric vector of length n with randomly drawn values according to the distribution

---

`rtnorm` *Truncate normal distribution*

Description

Truncate normal distribution at a minimum or maximum value

Usage

```
rtnorm(n, mean, sd, a = -Inf, b = Inf)
```

Arguments

`n`	Number of samples to be drawn
`mean`	Mean value of normal distribution
`sd`	Standard deviation of Normal distribution
`a`	Minimum of distribution (Default is -Inf)
`b`	Maximum of distribution (Default is Inf)

Value

numeric vector of length n with randomly drawn numbers of the truncated normal distribution

---

`shadingPlot` *Plot cumulative sum of RQ after application and compare it with initial state*

Description

Plot cumulative sum of RQ after application and compare it with initial state

Usage

```
shadingPlot(
  mat_xRow,
  ymin = 0,
  ymax = max(mat_xRow),
  resolution = 0.01,
  xlab = "",
  ylab = "",
  main = ""
)
```

Arguments

mat_xRow	A matrix where every row is a distribution for one x value
ymin	Minimum value of y-Axis
ymax	Maximumg value of y-Axis
resolution	The widths of quantiles used for shading (default is 1%)
xlab, ylab, main	Óptional definitions of axis and title

Value

A plot with overlapping transperant quantiles

---

temp_c_profile *Calculation of the pollutant concentration in top soil*

Description

Calculation of the pollutant concentration in top soil

Usage

```
temp_c_profile(conti_input, output_rate, c_i, t_max, t_res, t_beg = 0)
```

Arguments

conti_input	Numeric vector of atmospheric depositions in mg/kg topsoil
output_rate	Numeric vector of absolute pollutant decay rates.
c_i	Numeric vector of initial concentrations in top soil
t_max	Numeric value with the last considered timestep (unit depends on the unit of the decay rate)
t_res	Temporal resolution (numeric value) of the timesteps returned (unit depends on the unit of the decay rate)
t_beg	Numeric value of the first considered timestep. This value does not influence the calculation itself, it is only used for row names.

Value

Matrix with columns corresponding to the length of the provided numeric vectors and rows depending on the defined timesteps with t_max and t_res

Index