Package ‘clinPK’

May 9, 2022

Type Package

Title Clinical Pharmacokinetics Toolkit

Version 0.11.1

Date 2022-05-06

Description Calculates equations commonly used in clinical pharmacokinetics and clinical pharmacology, such as equations for dose individualization, compartmental pharmacokinetics, drug exposure, anthropomorphic calculations, clinical chemistry, and conversion of common clinical parameters. Where possible and relevant, it provides multiple published and peer-reviewed equations within the respective R function.

URL `https://github.com/InsightRX/clinPK`

License MIT + file LICENSE

RoxygenNote 7.1.2

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Ron Keizer [aut],
Jasmine Hughes [aut],
Dominic Tong [aut],
Kara Woo [aut, cre],
InsightRX [cph, fnd]

Maintainer Kara Woo `<kara@insight-rx.com>`

Repository CRAN

Date/Publication 2022-05-09 08:10:05 UTC

R topics documented:

clinPK-package 4
absolute2relative_bsa 4
accumulation_ratio 5

add_ruv 5
as.num 6
auc2dose 6
calc_abw 7
calc_aki_stage 7
calc_amts_for_conc 9
calc_baseline_scr 10
calc_bmi 11
calc_bsa 11
calc_creat 12
calc_creat_neo 13
calc_dosing_weight 13
calc_egfr 14
calc_egfr_cystatin 18
calc_ffm 18
calc_ibw 20
calc_kel_double_tdm 21
calc_kel_single_tdm 22
calc_kgfr 23
calc_lbw 24
calc_t12 25
check_covs_available 26
cm2inch 26
conc2mol 27
convert_creat_assay 27
convert_creat_unit 28
convert_flow_unit 28
dose2auc 29
egfr_cov_reqs 30
find_factor 30
find_nearest_dose 31
find_nearest_interval 31
fraction_of_ss 32
ibw_devine 32
ibw_standard 33
inch2cm 33
is.nil 34
kdigo_stage 34
kg2lbs 35
kg2oz 35
lbs2kg 36
mol2conc 36
nca 37
nca_trapezoid 38
oz2kg 39
pct_bmi_for_age 39
pct_for_age_generic 40
pct_height_for_age 40

pct_weight_for_age 41
pk_1cmt_bolus 42
pk_1cmt_bolus_cmax_ss 42
pk_1cmt_bolus_cmin_ss 43
pk_1cmt_bolus_dose_from_cmax 44
pk_1cmt_bolus_dose_from_cmin 44
pk_1cmt_bolus_ss 45
pk_1cmt_inf 46
pk_1cmt_inf_cmax_ss 47
pk_1cmt_inf_cmin_ss 47
pk_1cmt_inf_dose_for_range 48
pk_1cmt_inf_dose_from_cmax 49
pk_1cmt_inf_dose_from_cmin 49
pk_1cmt_inf_ss 50
pk_1cmt_oral 51
pk_1cmt_t12 52
pk_2cmt_bolus 52
pk_2cmt_bolus_cmax_ss 53
pk_2cmt_bolus_cmin_ss 54
pk_2cmt_bolus_dose_from_cmax 55
pk_2cmt_bolus_dose_from_cmin 56
pk_2cmt_bolus_ss 57
pk_2cmt_inf 58
pk_2cmt_inf_cmax_ss 58
pk_2cmt_inf_cmin_ss 59
pk_2cmt_inf_dose_from_cmax 60
pk_2cmt_inf_dose_from_cmin 61
pk_2cmt_inf_ss 62
pk_2cmt_t12 63
pk_2cmt_t12_interval 63
print.nca_output 64
read_who_table 64
relative2absolute_bsa 65
remove_lt_gt 66
time_to_ss 66
valid_units 67
weight2kg 67
%>=% 68
%<=% 68

Index **69**

`clinPK-package` *clinPK*

Description

Equations and tool for clinical pharmacokinetics

Author(s)

Ron Keizer `<ronkeizer@gmail.com>`

`absolute2relative_bsa` *Convert quantity expressed in absolute units relative to normalized BSA*

Description

Often used for eGFR estimates

Usage

```
absolute2relative_bsa(quantity, bsa = NULL, ...)
```

Arguments

`quantity`	quantity expressed in absolute units
`bsa`	ideal body weight in kg
`...`	arguments passed on to ‘calc_bsa‘, if bsa is NULL

Value

quantity expressed relative to /1.73m2

Examples

```
absolute2relative_bsa(quantity = 60, bsa = 1.6)
absolute2relative_bsa(quantity = 60, weight = 14, height = 90, method = "dubois")
```

`accumulation_ratio` *Calculate accumulation ratio This is the ratio of drug concentration or AUC at steady state over concentrations after single dose*

Description

Calculate accumulation ratio This is the ratio of drug concentration or AUC at steady state over concentrations after single dose

Usage

```
accumulation_ratio(kel = NULL, halflife = NULL, tau = 24)
```

Arguments

`kel`	drug elimination rate
`halflife`	halflife. Either ‘kel‘ or ‘halflife‘ is required.
`tau`	dosing interval

Examples

```
accumulation_ratio(halflife = 24, tau = 24)
accumulation_ratio(kel = 0.08, tau = 12)
```

`add_ruv` *Add residual variability to data*

Description

Add residual variability to data

Usage

```
add_ruv(x, ruv = list())
```

Arguments

`x`	data
`ruv`	list with arguments prop, add, exp

Examples

```
y <- pk_1cmt_inf()$y
y + add_ruv(y, list(prop = 0.1, add = 0.05))
```

`as.num` *factors or characters to numeric*

Description

factors or characters to numeric

Usage

```
as.num(x)
```

Arguments

`x`	value

`auc2dose` *Convert AUCtau or AUCt to dose (for 1-compartment linear PK model)*

Description

Convert AUCtau or AUCt to dose (for 1-compartment linear PK model)

Usage

```
auc2dose(auc, CL, V, t_auc = NA)
```

Arguments

`auc`	AUCtau
`CL`	Clearance
`V`	Volume of distribution
`t_auc`	if AUCtau is not known but only AUCt, ‘t_auc‘ specifies time until which AUC_t is calculated to be able to calculate dose

Examples

```
auc2dose(450, CL = 5, V = 50)
```

`calc_abw` *Calculate adjusted body weight (ABW)*

Description

Often used for chemotherapy calculations when actual weight > 120 Adjusted body weight is returned in units of kg.

Usage

```
calc_abw(weight = NULL, ibw = NULL, factor = 0.4, verbose = TRUE, ...)
```

Arguments

`weight`	actual body weight in kg
`ibw`	ideal body weight in kg
`factor`	weighting factor, commonly 0.4 or 0.3
`verbose`	show output?
`...`	parameters passed to ibw function (if ‘ibw‘ not specified)

Value

adjusted body weight in kg

Examples

```
calc_abw(weight = 80, ibw = 60)
calc_abw(weight = 80, height = 160, sex = "male", age = 60)
```

`calc_aki_stage` *Calculate AKI stage*

Description

Calculate AKI class based on serum creatinine values over time, using various methods for children (pRIFLE) and adults (RIFLE, kDIGO)

Usage

```
calc_aki_stage(
  scr = NULL,
  times = NULL,
  method = "kdigo",
  baseline_scr = "median",
  baseline_egfr = NULL,
  first_dose_time = NULL,
  age = NULL,
  egfr = NULL,
  egfr_method = NULL,
  force_numeric = FALSE,
  override_prifle_baseline = FALSE,
  verbose = TRUE,
  return_object = TRUE,
  ...
)
```

Arguments

`scr`	serum creatinine in mg/dL. Use ‘convert_creat()‘ to convert from mmol/L. Values below the detection limit ("<0.2") will be converted to numeric (0.2)
`times`	creatinine sample times in hours
`method`	classification method, one of ‘KDIGO‘, ‘RIFLE‘, ‘pRIFLE‘ (case insensitive)
`baseline_scr`	baseline serum creatinine, required for ‘RIFLE‘ classifation. Will use value if numeric. If ‘character‘, can be either ‘median‘, ‘median_before_treatment‘, ‘lowest‘, or ‘first‘.
`baseline_egfr`	baseline eGFR, required for ‘RIFLE‘ classifations. Will take median of ‘egfr‘ values if ‘NULL‘.
`first_dose_time`	time in hours of first dose relative to sCr value, used for calculate baseline serum creatinine in ‘median_before_treatment‘ approach.
`age`	age in years, needed when eGFR is used in the classification method
`egfr`	eGFR in ml/min/1.73m^2. Optional, can also be calcualted if ‘age‘, ‘weight‘, ‘height‘, ‘sex‘, ‘egfr_method‘ are specified as arguments.
`egfr_method`	eGFR calculation method, used by ‘calc_egfr()‘. If NULL, will pick default based on classification system (‘cockroft_gault‘ for RIFLE / kDIGO, ‘revised_schwartz‘ for pRIFLE).
`force_numeric`	keep stage numeric (1, 2, or 3), instead of e.g. "R", "I", "F" as in RIFLE. Default ‘FALSE‘.
`override_prifle_baseline`	by default, ‘pRIFLE‘ compares eGFR to 120 ml/min. Override by setting to TRUE.
`verbose`	verbose (‘TRUE‘ or ‘FALSE‘)
`return_object`	return object with detailed data (default ‘TRUE‘). If ‘FALSE‘, will just return maximum stage.

`...`	arguments passed on to ‘calc_egfr()‘

References

- pRIFLE: Ackan-Arikan et al. "Modified RIFLE criteria in critically ill children with acute kidney injury." Kidney Int. (2007)
- RIFLE: Bellomo et al. "Acute renal failure - definition, outcome measures, animal models, fluid therapy and information technology needs: the Second International Consensus Conference of the Acute Dialysis Quality Initiative (ADQI) Group." Critical Care. (2004)
- KDIGO: Khwaja. "KDIGO clinical practice guidelines for acute kidney injury." Nephron Clinical Practice. (2012)
- pRIFLE baseline eGFR: Soler et al. "pRIFLE (Pediatric Risk, Injury, Failure, Loss, End Stage Renal Disease) score identifies Acute Kidney Injury and predicts mortality in critically ill children : a prospective study." Pediatric Critical Care Medicine. (2014)

Examples

```
calc_aki_stage(
  scr = c(0.7, 0.9, 1.8, 1.5),
  t = c(0, 40, 100, 130),
  age = 50, weight = 60,
  height = 170, sex = "female")
```

`calc_amts_for_conc` *Calculate the amounts in all compartments in a compartmental PK system based on a given concentration in the central compartment, and assuming steady state.*

Description

Calculate the amounts in all compartments in a compartmental PK system based on a given concentration in the central compartment, and assuming steady state.

Usage

```
calc_amts_for_conc(conc = 10, parameters = NULL, n_cmt = 1)
```

Arguments

`conc`	concentration in central compartment
`parameters`	for PK model
`n_cmt`	number of compartments

Examples

```
calc_amts_for_conc(conc = 10, parameters = list(CL = 5, V = 50), n_cmt = 1)
calc_amts_for_conc(
  conc = 10,
  parameters = list(CL = 5, V = 50, Q = 20, V2 = 100),
  n_cmt = 2)
calc_amts_for_conc(
  conc = 10,
  parameters = list(CL = 5, V = 50, Q = 20, V2 = 100, Q2 = 30, V3 = 200),
  n_cmt = 3)
```

`calc_baseline_scr` *Calculate baseline sCr*

Description

Calculate baseline sCr

Usage

```
calc_baseline_scr(
  baseline_scr,
  scr,
  times,
  method,
  first_dose_time = NULL,
  verbose
)
```

Arguments

`baseline_scr`	baseline serum creatinine method (character). See calc_aki_stage() for availabloptions.
`scr`	serum creatinine in mg/dL. Use ‘convert_creat()‘ to convert from mmol/L. Values below the detection limit ("<0.2") will be converted to numeric (0.2)
`times`	creatinine sample times in hours
`method`	classification method, one of ‘KDIGO‘, ‘RIFLE‘, ‘pRIFLE‘ (case insensitive)
`first_dose_time`	time in hours of first dose relative to sCr value, used for calculate baseline serum creatinine in ‘median_before_treatment‘ approach.
`verbose`	verbose (‘TRUE‘ or ‘FALSE‘)

`calc_bmi` *Calculate BMI*

Description

Calculate BMI

Usage

```
calc_bmi(weight, height)
```

Arguments

`weight`	weight in kg
`height`	height in cm

Value

value of BMI in kg/m2

Examples

```
calc_bmi(weight = 70, height = 160)
```

`calc_bsa` *Calculate body surface area*

Description

Get an estimate of body-surface area (in m2) based on weight and height

Usage

```
calc_bsa(
  weight = NULL,
  height = NULL,
  method = c("dubois", "mosteller", "haycock", "gehan_george", "boyd")
)
```

Arguments

`weight`	weight
`height`	height
`method`	estimation method, choose from ‘dubois‘, ‘mosteller‘, ‘haycock‘, ‘gehan_george‘, ‘boyd‘

Value

Returns a list of the following elements:

`value`	Body Surface Area (BSA) in units of m2
`unit`	Unit describing BSA, (m2)

Examples

```
calc_bsa(weight = 70, height = 170)
calc_bsa(weight = 70, height = 170, method = "gehan_george")
```

`calc_creat` *Estimate serum creatinine*

Description

Calculate an estimated serum creatinine. Function takes vectorized input as well.

Usage

```
calc_creat(sex = NULL, age = NULL, digits = 1)
```

Arguments

`sex`	sex, either ‘male‘ or ‘female‘
`age`	age in years
`digits`	number of digits to round to

Details

Uses equations described in Ceriotti et al. Clin Chem. 2008, and Junge W et al. Clin Chim Acta. 2004. For age 15-18, a linear interpolation is used between equations for <15 and >18 years as described in Johanssen A et al. Ther Drug Monit 2011.

Examples

```
calc_creat(sex = "male", age = 40)
calc_creat(sex = "male", age = c(10, 17, 60))
```

`calc_creat_neo` *Estimate serum creatinine in neonates*

Description

Calculate an estimated serum creatinine. Function takes vectorized input as well.

Usage

```
calc_creat_neo(pma = NULL, digits = 1)
```

Arguments

`pma`	post-natal age in weeks
`digits`	number of digits to round to

Details

Uses equations described in Germovsek E et al. (http://www.ncbi.nlm.nih.gov/pubmed/27270281) based on data from Cuzzolin et al. (http://www.ncbi.nlm.nih.gov/pubmed/16773403) and Rudd et al. (http://www.ncbi.nlm.nih.gov/pubmed/6838252)

Examples

```
calc_creat_neo(pma = 36)
convert_creat_unit(calc_creat_neo(pma = 36))
```

`calc_dosing_weight` *Calculate commonly used "dosing weight"*

Description

Dosing weight is determined based on total (TBW), ideal (IBW), or adjusted (ABW) body weight in kg.

Usage

```
calc_dosing_weight(weight, height, age, sex, verbose = TRUE, ...)
```

Arguments

`weight`	weight
`height`	height
`age`	age
`sex`	sex
`verbose`	verbosity (‘TRUE‘ or ‘FALSE‘)
`...`	pased to ‘calc_abw()‘ function

Details

This is derived using following: - In principle, use IBW - If total body weight (TBW) > 1.2*IBW, then use ABW - If TBW < IBW, use TBW

Value

Returns a list of the following elements:

`value`	Dosing weight, in units of kg
`unit`	Units of dosing weight (kg)
`type`	Type of dosing weight selected, e.g., total body weight, ideal body weight.

Examples

```
calc_dosing_weight(weight = 50, height = 170, sex = "female", age = 50)
```

`calc_egfr` *Calculate eGFR*

Description

Calculate the estimated glomerular filtration rate (an indicator of renal function) based on measured serum creatinine using one of the following approaches:

- Cockcroft-Gault (using weight, ideal body weight, or adjusted body weight)
- C-G spinal cord injury (using correction factor of 0.7, representing median correction point reported in the original publication (parapalegic patients: 0.8; tetrapalegic patients: 0.6))
- Revised Lund-Malmo
- Modification of Diet in Renal Disease study (MDRD; with or without consideration of race, using either the original equation (published 2001) or the equation updated to reflect serum creatinine assay standardization (2006))
- CKD-EPI (with or without consideration of race, or 2021 re-fit without race)
- Schwartz
- Schwartz revised / bedside
- Jelliffe
- Jelliffe for unstable renal function. Note that the 15 P_adj recommended for hemodialysis patients is not included in this implementation.
- Wright equation for eGFR in cancer patients, with creatinine measured using the Jaffe assay.

Equations for estimation of eGFR from Cystatin C concentrations are available from the ‘calc_egfr_cystatin()‘ function.

Usage

```
calc_egfr(
  method = "cockcroft_gault",
  sex = NULL,
  age = NULL,
  scr = NULL,
  scr_unit = NULL,
  race = "other",
  weight = NULL,
  height = NULL,
  bsa = NULL,
  preterm = FALSE,
  ckd = FALSE,
  times = NULL,
  bsa_method = "dubois",
  relative = NULL,
  unit_out = "mL/min",
  verbose = TRUE,
  min_value = NULL,
  max_value = NULL,
  fail = TRUE,
  ...
)

egfr_wright(age, bsa, sex, scr)

egfr_jelliffe(age, sex, bsa, scr)

egfr_jelliffe_unstable(weight, times, scr, age, sex)

egfr_mdrd(sex, race, scr, age, use_race, original_expression)

egfr_ckd_epi(sex, race, scr, age, use_race)

egfr_ckd_epi_as_2021(sex, scr, age)

egfr_cockcroft_gault_sci(sex, age, scr, weight)

egfr_cockcroft_gault(sex, age, scr, weight)

egfr_malmo_lund(sex, age, scr)

egfr_bedside_schwartz(age, height, scr, verbose)

egfr_schwartz(age, preterm, sex, height, scr)
```

Arguments

`method`	eGFR estimation method, choose from ‘cockcroft_gault‘, ‘cockcroft_gault_ideal‘, ‘cockcroft_gault_adjusted‘, ‘cockcroft_gault_adaptive‘, ‘mdrd‘, ‘mdrd_ignore_race‘, ‘mdrd_original‘, ‘mdrd_original_ignore_race‘, ‘ckd_epi‘, ‘ckd_epi_ignore_race‘, ‘ckd_epi_as_2021‘, ‘malmo_lund_revised‘, ‘schwartz‘, ‘jelliffe‘, ‘jellife_unstable‘, ‘wright‘.
`sex`	sex
`age`	age, in years
`scr`	serum creatinine (mg/dL)
`scr_unit,`	‘mg/dL‘ or ‘micromol/L‘ (==‘umol/L‘)
`race`	‘black‘ or ‘other‘, Required for CKD-EPI and MDRD methods for estimating GFR. To use these methods without race, use ‘method = "ckd_epi_ignore_race"‘, ‘method = "ckd_epi_as_2021"‘, ‘method = "mdrd_ignore_race"‘ or ‘method = "mdrd_original_ignore_race"‘. See Note section below for important considerations when using race as a predictive factor in eGFR.
`weight`	weight, in ‘kg‘
`height`	height, in ‘cm‘, used for converting to/from BSA-normalized units.
`bsa`	body surface area
`preterm`	is patient preterm? Used for Schwartz method.
`ckd`	chronic kidney disease? Used for Schwartz method.
`times`	vector of sampling times (in days!) for creatinine (only used in Jelliffe equation for unstable patients)
`bsa_method`	BSA estimation method, see ‘calc_bsa()‘ for details
`relative`	‘TRUE‘/‘FALSE‘. Report eGFR as per 1.73 m2? Requires BSA if re-calculation required. If ‘NULL‘ (=default), will choose value typical for ‘method‘.
`unit_out`	‘ml/min‘ (default), ‘L/hr‘, or ‘mL/hr‘
`verbose`	verbosity, show guidance and warnings. ‘TRUE‘ by default
`min_value`	minimum value (‘NULL‘ by default). The cap is applied in the same unit as the ‘unit_out‘.
`max_value`	maximum value (‘NULL‘ by default). The cap is applied in the same unit as the ‘unit_out‘.
`fail`	invoke ‘stop()‘ if not all covariates available?
`...`	arguments passed on to ‘calc_abw‘ or ‘calc_dosing_weight‘
`use_race`	whether to include race as a factor in the calculation (TRUE or FALSE); see note
`original_expression`	whether the MDRD equation should use the 2001 coefficient (TRUE) or the 2006 coefficient (FALSE), which was updated for standardization of the creatinine assay.

Note

The MDRD and CKD-EPI equations use race as a factor in estimation of GFR. Racism has historically been and continues to be a problem in medicine, with racialized patients experiencing poorer outcomes. Given this context, the use of race in clinical algorithms should be considered carefully (Vyas et al., NEJM (2020)). Provided here are versions of the CKD-EPI and MDRD equations that do not consider the race of the patient. Removing race from GFR estimation may lead to worse outcomes for Black patients in some contexts (Casal et al., The Lancet (2021)). On the other hand, including race in GFR estimation may also prevent Black patients from obtaining procedures like kidney transplants (Zelnick, et al. JAMA Netw Open. (2021)). In 2021, the NKF/ASN Task Force on Reassessing the Inclusion of Race in Diagnosing Kidney Diseases published revised versions of the CKD-EPI equations refit on the original data but with race excluded, which may produce less biased estimates (Inker, et al., NEJM (2021)).

References

- Cockcroft-Gault: Cockcroft & Gault, Nephron (1976)
- Cockcroft-Gault for spinal cord injury: Mirahmadi et al., Paraplegia (1983)
- Revised Lund-Malmo: Nyman et al., Clinical Chemistry and Laboratory Medicine (2014)
- MDRD: Manjunath et al., Curr. Opin. Nephrol. Hypertens. (2001) and Levey et al., Clinical Chemistry (2007). (See Note.)
- CKD-EPI: Levey et al., Annals of Internal Medicine (2009). (See Note.)
- CKD-EPI (2021): Inker, et al., NEJM (2021).
- Schwartz: Schwartz et al., Pediatrics (1976)
- Schwartz revised / bedside: Schwartz et al., Journal of the American Society of Nephrology (2009)
- Jelliffe: Jelliffe, Annals of Internal Medicine (1973)
- Jelliffe for unstable renal function: Jelliffe, American Journal of Nephrology (2002)
- Wright: Wright et al., British Journal of Cancer (2001)

Examples

```
calc_egfr(sex = "male", age = 50, scr = 1.1, weight = 70)
calc_egfr(sex = "male", age = 50, scr = 1.1, weight = 70, unit_out = "L/hr")
calc_egfr(sex = "male", age = 50, scr = 1.1, weight = 70, bsa = 1.8, method = "ckd_epi")
calc_egfr(sex = "male", age = 50, scr = c(1.1, 0.8),
  weight = 70, height = 170, method = "jelliffe")
calc_egfr(sex = "male", age = 50, scr = c(1.1, 0.8),
  weight = 70, height = 170, method = "jelliffe_unstable")
calc_egfr(sex = "male", age = 50, scr = 1.1,
  weight = 70, bsa = 1.6, method = "malmo_lund_revised", relative = FALSE)
```

`calc_egfr_cystatin` *Calculate eGFR based on Cystatin C measurements*

Description

Calculate eGFR based on Cystatin C measurements

Usage

```
calc_egfr_cystatin(
  cystatin = NULL,
  cystatin_unit = "mg/L",
  method = "grubb",
  unit_out = "mL/min"
)
```

Arguments

`cystatin`	serum cystatin concentration (mg/L)
`cystatin_unit,`	only ‘mg/L‘ available
`method`	eGFR estimation method, choose from ‘grubb‘, ‘larsson‘
`unit_out`	eGFR output unit, choose from ‘ml/min‘, ‘ml/hr‘, ‘l/min‘, ‘l/hr‘

Examples

```
calc_egfr_cystatin(1.0)
calc_egfr_cystatin(1.0, method = "larsson")
calc_egfr_cystatin(1.0, unit_out = "l/hr")
```

`calc_ffm` *Calculate fat-free mass*

Description

Get an estimate of fat-free mass (FFM, in kg) based on weight, height, and sex (and age for Storset equation).

Usage

```
calc_ffm(
  weight = NULL,
  bmi = NULL,
  sex = NULL,
  height = NULL,
  age = NULL,
 method = c("janmahasatian", "green", "al-sallami", "storset", "bucaloiu", "hume",
    "james", "garrow_webster"),
  digits = 1
)

ffm_janmahasatian_green(weight, sex, height = NULL, bmi = NULL)

ffm_al_sallami(weight, sex, age, height = NULL, bmi = NULL)

ffm_storset(weight, sex, height, age)

ffm_bucaloiu(weight, sex, height, age)

ffm_hume(weight, sex, height)

ffm_james(weight, sex, height)

ffm_garrow_webster(weight, sex, height)
```

Arguments

`weight`	total body weight in kg
`bmi`	BMI, only used in 'green' method. If 'weight' and 'height' are both specified, 'bmi' will be calculated on-the-fly.
`sex`	sex, either 'male' of 'female'
`height`	height in cm, only required for 'holford' method, can be used instead of 'bmi' for 'green' method
`age`	age, only used for Storset equation
`method`	estimation method, one of 'janmahasatian' (default), 'green', 'al-sallami', 'storset', 'bucaloiu', 'hume', 'james', or 'garrow_webster'.
`digits`	round to number of digits

Details

References: 'janmahasatian', 'green': Janmahasatian et al. Clin Pharmacokinet. 2005;44(10):1051-65) 'al-sallami': Al-Sallami et al. Clin Pharmacokinet 2015 'storset': Storset E et al. TDM 2016 'bucaloiu': Bucaloiu ID et al. Int J of Nephrol Renovascular Dis. 2011 (Morbidly obese females) 'hume': Hume R. J Clin Pathol 1966 'james': James WPT et al. Research on obesity: a report of the DHSS/MRC Group 1976 'garrow_webster': Garrow JS, Webster J. Quetelet's index (W/H2) as a measure of fatness. Int J Obesity 1984

Overview: - Sinha J, Duffull1 SB, Al-Sallami HS. Clin Pharmacokinet 2018. https://doi.org/10.1007/s40262-017-0622-5

Value

Returns a list of the following elements:

`value`	Fat-free Mass (FFM) in units of kg
`unit`	Unit describing FFM, (kg)
`method`	Method used to calculate FFF

Examples

```
calc_ffm(weight = 70, bmi = 25, sex = "male")
calc_ffm(weight = 70, height = 180, age = 40, sex = "female", method = "storset")
```

`calc_ibw` *Calculate ideal body weight in kg for children and adults*

Description

Get an estimate of ideal body weight. This function allows several commonly used equations

Usage

```
calc_ibw(
  weight = NULL,
  height = NULL,
  age = NULL,
  sex = "male",
  method_children = "standard",
  method_adults = "devine"
)
```

Arguments

`weight`	weight in kg
`height`	height in cm
`age`	age in years
`sex`	sex
`method_children`	method to use for children >1 and <18 years. Currently ‘"standard"‘ is the only method that is supported.
`method_adults`	method to use for >=18 years. Currently ‘"devine"‘ is the only method that is supported (Devine BJ. Drug Intell Clin Pharm. 1974;8:650-655).

Details

Equations:

<1yo Use actual body weight

1-17 years old ('standard'): if height < 5ft: IBW= (height in cm2 x 1.65)/1000 if height > 5ft: IBW (male) = 39 + (2.27 x height in inches over 5 feet) IBW (female) = 42.2 + (2.27 x height in inches over 5 feet)

Methods not implemented yet: McLaren: IBW = - step1: x = 50th percentile height for given age - step2: IBW = 50th percentile weight for x on weight-for-height scale Moore: IBW = weight at percentile x for given age, where x is percentile of height for given age BMI: IBW = 50th percentile of BMI for given age x (height in m)^2 ADA: IBW = 50th percentile of WT for given age

>= 18 years old (Devine equation) IBW (male) = 50 + (2.3 x height in inches over 5 feet) IBW (female) = 45.5 + (2.3 x height in inches over 5 feet)

Examples

```
calc_ibw(weight = 70, height = 170, age = 40, sex = "female")
calc_ibw(weight = 30, height = 140, age = 10, sex = "female")
```

`calc_kel_double_tdm` *Calculate elimination rate when given two TDM samples*

Description

Calculate elimination rate when given two TDM samples

Usage

```
calc_kel_double_tdm(
  dose = 1000,
  t = c(2, 11.5),
  dv = c(30, 10),
  tau = 12,
  t_inf = 1,
  V = NULL,
  steady_state = TRUE,
  return_parameters = FALSE
)
```

Arguments

`dose`	dose amount
`t`	time or time after dose, vector of size 2
`dv`	observed value, vector of size 2
`tau`	dosing interval

`t_inf`	infusion time
`V`	if specified, use that (empiric) value and don't estimate from data. Default 'NULL'.
`steady_state`	samples taken at steady state? Only influences AUCtau.
`return_parameters`	return all parameters instead of only kel?

Examples

```
calc_kel_double_tdm(dose = 1000, t = c(3, 18), dv = c(30, 10))
```

`calc_kel_single_tdm` *Calculate elimination rate when given a single TDM sample*

Description

Using iterative k_el calculation, and based on given Volume

Usage

```
calc_kel_single_tdm(
  dose = 1000,
  V = 50,
  t = 10,
  dv = 10,
  tau = 12,
  t_inf = 1,
  kel_init = 0.1,
  n_iter = 25,
  learn_rate = 0.2
)
```

Arguments

`dose`	dose amount
`V`	volume of distribution
`t`	time or time after dose
`dv`	observed value
`tau`	dosing interval
`t_inf`	infusion time
`kel_init`	estimate of elimination rate
`n_iter`	number of iterations to improve estimate of elimination rate
`learn_rate`	default is 0.2

Examples

```
calc_kel_single_tdm(dose = 1000, t = 18)
```

---

calc_kgfr *Calculate kinetic GFR*

Description

Calculate the kinetic GFR based on a patients first two serum creatinine measurements. Kinetic GFR may be more predictive of future AKI for patients whose serum creatinine is changing quickly. Briefly, an increase in SCr over the course of a day indicates an effective GFR lower than the most recent SCr measurement may indicate if steadystate is assumed, while a decrease in SCr over a short time indicates a higher effective GFR than the most recent SCr would indicate. There are several ways of approximating maximum theoretical creatinine accumulation rate; here the method used by Pianta et al., (PLoS ONE, 2015) has been implemented.

Usage

```
calc_kgfr(
  scr1 = NULL,
  scr2 = NULL,
  scr_unit = "mg/dl",
  time_delay = NULL,
  weight = NULL,
  vd = NULL,
  egfr = NULL,
  egfr_method = NULL,
  sex = NULL,
  age = NULL,
  height = NULL,
  ...
)
```

Arguments

`scr1`	baseline scr
`scr2`	second scr measurement
`scr_unit`	scr unit, defaults to mg/dl
`time_delay`	time between scr1 and scr2 in hours
`weight`	patient weight in kg
`vd`	volume of distribution in L, defaults to 0.6 * weight
`egfr`	eGFR in ml/min at the time of scr1, or leave blank to call calc_egfr
`egfr_method`	string, only necessary if egfr is not specified.
`sex`	string (male or female), only necessary if egfr is not specified.

`age`	age in years, only necessary if egfr is not specified.
`height`	in m, necessary only for some egfr calculation methods.
`...`	further arguments (optional) to be passed to calc_egfr.

Value

kGFR, in ml/min

References

Pianta et al., PLoS ONE (2015)

Examples

```
calc_kgfr(weight = 100, scr1 = 150, scr2 = 200, scr_unit = 'umol/l',
         time_delay = 24, egfr = 30)
calc_kgfr(weight = 70, scr1 = 350, scr2 = 300, scr_unit = 'umol/l',
          time_delay = 24, egfr_method = 'mdrd', age = 70, sex = 'male')
```

`calc_lbw` *Calculate lean body weight*

Description

Get an estimate of lean body weight (LBW, in kg) based on weight, height, and sex.

Usage

```
calc_lbw(
  weight = NULL,
  bmi = NULL,
  sex = NULL,
  height = NULL,
  method = "green",
  digits = 1
)
```

Arguments

`weight`	total body weight in kg
`bmi`	bmi
`sex`	sex, either ‘male‘ of ‘female‘
`height`	height in cm
`method`	estimation method, either ‘green‘ (default), ‘boer‘, ‘james‘, ‘hume‘
`digits`	round to number of digits

Details

Note: technically not the same as fat-free mass, although difference is small.

References: ‘green‘: Green and Duffull. Clin Pharmacol Ther 2002; ‘james‘: Absalom AR et al. Br J Anaesth 2009; 103:26-37. James W. Research on obesity. London: Her Majesty’s Stationary Office, 1976. ‘hume‘ : Hume R et al. J Clin Pathol. 1966 Jul; 19(4):389-91. ‘boer‘ : Boer P et al. Am J Physiol 1984; 247: F632-5

Value

Returns a list of the following elements:

`value`	Lean Body Weight (LBW) in units of kg
`unit`	Unit describing LBW, (kg)

Examples

```
calc_lbw(weight = 80, height = 170, sex = "male")
calc_lbw(weight = 80, height = 170, sex = "male", method = "james")
```

`calc_t12` *Calculate half-life based on two points*

Description

based on two sampling points (in same interval)

Usage

```
calc_t12(t1, t2, y1, y2)
```

Arguments

`t1`	first sampling timepoint
`t2`	second sampling timepoint
`y1`	first sample value
`y2`	second sample value

Examples

```
calc_t12(3, 24, 30, 10)
```

`check_covs_available` *Returns parameters needed to calculate eGFR according to the method specified.*

Description

returns true if all patient covs specified in required covs are non-null, non-NA and not a 0-character string. See ‘is.nil‘ for missing data types checked. Returns TRUE if no covariates are required.

Usage

```
check_covs_available(
  cov_reqs = NULL,
  patient_covs = NULL,
  verbose = TRUE,
  fail = TRUE
)
```

Arguments

`cov_reqs`	vector of covariates required for calculating derived covariatiate
`patient_covs`	named list of covariates
`verbose`	stop and describe missing covariate(s)?
`fail`	invoke ‘stop()‘ if not all covariates available?

Examples

```
check_covs_available(
  egfr_cov_reqs('cockcroft_gault_ideal')[[1]],
  list(creat = 1, weight = 100, height = 160, sex = 'female', age = 90))
```

`cm2inch` *Convert cm to inch*

Description

Convert cm to inch

Usage

```
cm2inch(cm)
```

Arguments

`cm`	vector

Examples

```
cm2inch(2.54)
```

`conc2mol` *Convert concentration to molar*

Description

Convert concentration to molar

Usage

```
conc2mol(conc = NULL, unit_conc = NULL, mol_weight = NULL, unit_mol = NULL)
```

Arguments

`conc`	concentration in e.g. g/L
`unit_conc,`	one of ‘g/l‘, ‘mg/l‘, ‘microg/l‘, ‘mcg/l", ‘ng/l‘, ‘mg/ml‘, ‘microg/ml‘, ‘mcg/ml‘, ‘ng/ml‘
`mol_weight`	concentration in g/mol
`unit_mol`	one of ‘mol/L‘, ‘mmol/mL‘, ‘mmol/L‘

Examples

```
conc2mol(100, unit_conc = "g/l", mol_weight = 180.15588)
```

`convert_creat_assay` *Convert serum creatinine from various assays to Jaffe*

Description

Based on equations as reported in Srivastava et al. 2009 (Pediatr Res. 2009 Jan;65(1):113-6. doi: 10.1203/PDR.0b013e318189a6e8)

Usage

```
convert_creat_assay(scr, from = "idms", to = "jaffe")
```

Arguments

`scr`	vector of serum creatinine values
`from`	assay type, either ‘jaffe‘, ‘enzymatic‘ or ‘idms‘
`to`	assay type, either ‘jaffe‘, ‘enzymatic‘ or ‘idms‘

Examples

```
convert_creat_assay(scr = c(1.1, 0.8, 0.7), from = "enzymatic", to = "jaffe")
```

`convert_creat_unit` *Convert creatinine to different unit*

Description

Convert creatinine to different unit

Usage

```
convert_creat_unit(value = NULL, unit_in = "mg/dL")
```

Arguments

`value`	serum creatinine in either mg/dL or micromol/L
`unit_in`	unit, either ‘mg/dL‘ or ‘micromol/L‘

Examples

```
convert_creat_unit(1)
convert_creat_unit(88.42, unit_in = "micromol/l")
```

`convert_flow_unit` *Convert flow (e.g. clearance) from / to units*

Description

Flow units are expected to be specified as a combination of volume per time units, potentially specified per kg body weight, e.g. "mL/min", or "L/hr/kg".

Usage

```
convert_flow_unit(value = NULL, from = "l", to = "ml", weight = NULL)
```

Arguments

`value`	flow value
`from`	from flow unit, e.g. ‘L/hr‘.
`to`	to flow unit, e.g. ‘mL/min‘
`weight`	for performing per weight (kg) conversion

Details

Accepted volume units are "L", "dL", and "mL". Accepted time units are "min", "hr", and "day". The only accepted weight unit is "kg".

The function is not case-sensitive.

Examples

```
## single values
convert_flow_unit(60, "L/hr", "ml/min")
convert_flow_unit(1, "L/hr/kg", "ml/min", weight = 80)

## vectorized
convert_flow_unit(
  c(10, 20, 30),
  from = c("L/hr", "mL/min", "L/hr"),
  to = c("ml/min/kg", "L/hr", "L/hr/kg"),
  weight = c(70, 80, 90))
```

`dose2auc` *Convert dose to expected AUCinf or AUCt for 1 compartment linear PK model*

Description

Convert dose to expected AUCinf or AUCt for 1 compartment linear PK model

Usage

```
dose2auc(dose, CL, V, t_auc = NULL)
```

Arguments

`dose`	dose amount
`CL`	Clearance
`V`	Volume of distribution
`t_auc`	if AUC_t is desired, ‘t_auc‘ specifies time until which AUC_t is calculated

Examples

```
dose2auc(dose = 1000, CL = 5, V = 50)
dose2auc(dose = 1000, CL = 5, V = 50, t_auc = c(12, 24, 48, 72))
```

`egfr_cov_reqs` *Returns parameters needed to calculate eGFR according to the method specified.*

Description

returns a named list, with the name being the eGFR method after being checked for certain typos or misspecifications, and the values being the required covariates.

Usage

```
egfr_cov_reqs(method, relative = NULL)
```

Arguments

`method`	egfr calculation method
`relative`	if egfr calculations should be relative or not

Examples

```
egfr_cov_reqs('schwartz_revised')
```

`find_factor` *Helper function to grab the conversion factor from an input unit and given list*

Description

Helper function to grab the conversion factor from an input unit and given list

Usage

```
find_factor(full_unit, units = NULL, prefix = "^")
```

Arguments

`full_unit`	full unit, e.g. "mL/min/kg"
`units`	unit specification list, e.g. ‘list("ml" = 1/1000, "dl" = 1/10, "l" = 1)‘
`prefix`	prefix used in matching units, e.g. "^" only matches at start of string while "_" matches units specified as "/"

`find_nearest_dose` *Generic function to calculate the dose nearest to a specific dose unit increment*

Description

Generic function to calculate the dose nearest to a specific dose unit increment

Usage

```
find_nearest_dose(dose = NULL, increment = 250, type = "round")
```

Arguments

`dose`	dose value
`increment`	available increments of dose
`type`	how to round, one of ‘round‘, ‘floor‘, or ‘ceiling‘

Examples

```
find_nearest_dose(573)
find_nearest_dose(573, increment = 50)
```

`find_nearest_interval` *Generic function to calculate the interval nearest to a possible dosing interval*

Description

Generic function to calculate the interval nearest to a possible dosing interval

Usage

```
find_nearest_interval(
  interval = NULL,
  possible = c(4, 6, 8, 12, 24, 36, 48),
  type = "absolute"
)
```

Arguments

`interval`	dose value
`possible`	available increments of dose
`type`	pick either ‘nearest‘ absolute interval, or nearest ‘lower‘, or nearest ‘higher‘ interval.

Examples

```
find_nearest_interval(19.7)
find_nearest_interval(19.7, c(6, 8, 12))
```

`fraction_of_ss` *Calculate fraction of steady state at particular time after start of dosing*

Description

Calculate fraction of steady state at particular time after start of dosing

Usage

```
fraction_of_ss(kel = NULL, halflife = NULL, t = NULL, n = NULL, tau = NULL)
```

Arguments

`kel`	drug elimination rate
`halflife`	halflife. Either ‘kel‘ or ‘halflife‘ is required.
`t`	time at which to calculate fraction of steady state
`n`	number of dosing intervals after which to calculate fraction of steady state. Requires ‘tau‘ as well, cannot be used together with ‘t‘ argument.
`tau`	dosing interval

Examples

```
fraction_of_ss(halflife = 24, t = 72)
fraction_of_ss(halflife = 36, n = 3, tau = 24)
```

`ibw_devine` *Calculate IBW using "devine" equation*

Description

Calculate IBW using "devine" equation

Usage

```
ibw_devine(age, height = NULL, sex = NULL)
```

Arguments

`age`	age in years
`height`	height in cm
`sex`	sex

`ibw_standard` *Calculate IBW using "standard" equation*

Description

Calculate IBW using "standard" equation

Usage

```
ibw_standard(age, height = NULL, sex = NULL)
```

Arguments

`age`	age in years
`height`	height in cm
`sex`	sex

`inch2cm` *Convert inch to cm*

Description

Convert inch to cm

Usage

```
inch2cm(inch)
```

Arguments

`inch`	vector

Examples

```
inch2cm(1)
```

`is.nil` *Check if values in vector are empty*

Description

Check if values in vector are empty

Usage

```
is.nil(x = NULL)
```

Arguments

`x`	vector

`kdigo_stage` *Calculate AKI stage according to KDIGO criteria*

Description

Calculate AKI stage according to KDIGO criteria

Usage

```
kdigo_stage(dat, baseline_scr, age)
```

Arguments

`dat`	Data frame containing at least the following columns: • `scr`: serum creatinine • `t`: creatinine sample times in hours • `baseline_scr_diff`: difference between baseline scr and scr at current timepoint • `egfr`: eGFR at timepoint
`baseline_scr`	Baseline serum creatinine value (numeric)
`age`	Patient age

`kg2lbs` *Convert kg to lbs*

Description

Convert kg to lbs

Usage

```
kg2lbs(kg)
```

Arguments

kg	vector

Examples

```
kg2lbs(1)
```

`kg2oz` *Convert kg to oz*

Description

Convert kg to oz

Usage

```
kg2oz(kg)
```

Arguments

kg	vector

Examples

```
kg2oz(1)
```

`lbs2kg` *Convert lbs to kg*

Description

Convert lbs to kg

Usage

```
lbs2kg(lbs)
```

Arguments

`lbs`	vector

Examples

```
lbs2kg(2.20462)
```

`mol2conc` *Convert molar to concentration*

Description

Convert molar to concentration

Usage

```
mol2conc(mol = NULL, unit_mol = NULL, unit_conc = NULL, mol_weight = NULL)
```

Arguments

`mol`	concentration in molars
`unit_mol`	unit of input concentration (molar), one of ‘mol/L‘, ‘mmol/mL‘, ‘mmol/L‘
`unit_conc,`	output unit, one of ‘g/l‘, ‘mg/l‘, ‘microg/l‘, ‘mcg/l", ‘ng/l‘, ‘mg/ml‘, ‘microg/ml‘, ‘mcg/ml‘, ‘ng/ml‘
`mol_weight`	concentration in g/mol

Examples

```
mol2conc(1, unit_mol = "mmol/l", mol_weight = 180)
```

`nca` *Perform an NCA based on a NONMEM-style dataset*

Description

Perform an NCA based on a NONMEM-style dataset

Usage

```
nca(
  data = NULL,
  dose = 100,
  tau = 24,
  method = c("log_linear", "log_log", "linear"),
  scale = list(auc = 1, conc = 1),
  dv_min = 0.001,
  t_inf = NULL,
  fit_samples = NULL,
  weights = NULL,
  extend = TRUE,
  has_baseline = TRUE,
  route = c("iv", "oral", "im", "sc")
)
```

Arguments

`data`	data.frame with time and dv columns
`dose`	dose amount
`tau`	dosing frequency, default is 24.
`method`	‘linear‘, ‘log_linear‘ (default), or ‘log_log‘
`scale`	list with scaling for auc and concentration (‘conc‘)
`dv_min`	minimum concentrations, lower observations will be set to this value
`t_inf`	infusion time, defaults to 0
`fit_samples`	vector of sample indexes used in fit to calculate elimination rate, e.g. ‘c(3,4,5)‘. If not specified (default), it will evaluate which of the last n samples shows the largest adjusted R^2 when log-transformed data is fitted using linear regression, and use those samples in the estimation of the elimination rate.
`weights`	vector of weights to be used in linear regression (same size as specified concentration data), or function with concentration as argument.
`extend`	perform an ’extended’ NCA, i.e. for the calculation of the AUCs, back-extend to the expected true Cmax to also include that area.
`has_baseline`	does the included data include a baseline? If ‘FALSE‘, baseline is set to zero.
`route`	administration route, ‘iv‘ (intravenous, default), ‘oral‘, ‘sc‘ (sub-cutaneous), or ‘im‘ (intra-muscular).

Value

Returns a list of three lists:

`pk` Lists pk parameters.

- `kel`: elimination constant
- `t_12`: half-life
- `v`: distribution volume
- `cl`: clearance

`descriptive` Lists exposure parameters.

- `cav_t`: the average concentration between the first observation and the last observation without extrapolating to tau
- `cav_tau`: the average concentration from 0 to tau
- `cmin`: the extrapolated concentration at `time` = `tau`
- `c_max_true`: only available if `extend` = `TRUE`, the extrapolated peak concentration
- `c_max`: only available if `extend` = `FALSE`, the observed maximum concentration
- `auc_inf`: the extrapolated AUC as time goes to infinity
- `auc_24`: the extrapolated AUC after 24 hours, provided no further doses are administered
- `auc_tau`: the extrapolated AUC at the end of the dosing interval
- `auc_t`: the AUC at the time of the last observation

`settings` Lists dosing information.

- `dose`: dose quantity
- `tau`: dosing interval

Examples

```
data <- data.frame(time = c(0, 2, 4, 6, 8, 12, 16),
                   dv   = c(0, 10, 14, 11, 9, 5, 1.5))
nca(data, t_inf = 2)
```

`nca_trapezoid` *NCA core function to perform log-linear trapezoid calculations for post-infusion*

Description

NCA core function to perform log-linear trapezoid calculations for post-infusion

Usage

```
nca_trapezoid(data)
```

Arguments

`data`	dataset passed from nca() with ‘time‘ and ‘dv‘ columns

`oz2kg` *Convert oz to kg*

Description

Convert oz to kg

Usage

```
oz2kg(oz)
```

Arguments

`oz`	vector

Examples

```
oz2kg(2.20462)
```

`pct_bmi_for_age` *Percentile BMI for age for children*

Description

Based on tables from WHO: http://www.who.int/growthref/who2007_bmi_for_age/en/

Usage

```
pct_bmi_for_age(
  age = NULL,
  bmi = NULL,
  sex = NULL,
  height = NULL,
  return_median = FALSE,
  ...
)
```

Arguments

`age`	age in years
`bmi`	bmi Optional, if specified, will calculate closest percentile and return in list as ‘percentile‘
`sex`	either ‘male‘ or ‘female‘
`height`	height
`return_median`	just return the median expected value
`...`	parameters passed to ‘read_who_table()‘

Examples

```
pct_bmi_for_age(age = 8, sex = "male")
pct_bmi_for_age(age = 8, bmi = 15, sex = "male")
```

`pct_for_age_generic` *Percentile height or weight for age for children*

Description

This is the underlying function, the exposed functions are pct_weight_for_age() and pct_height_for_age() Based on tables from WHO: http://www.who.int/childgrowth/standards/height_for_age/en/

Usage

```
pct_for_age_generic(
  age = NULL,
  value = NULL,
  sex = NULL,
  variable = "weight",
  ...
)
```

Arguments

`age`	age in years
`value`	height in kg. Optional, if specified, will calculate closest percentile and return in list as 'percentile'
`sex`	either 'male' or 'female'
`variable`	weight or height?
`...`	parameters passed to 'read_who_table()'

`pct_height_for_age` *Percentile height for age for children*

Description

Based on tables from WHO: http://www.who.int/childgrowth/standards/height_for_age/en/

Usage

```
pct_height_for_age(
  age = NULL,
  height = NULL,
  sex = NULL,
  return_median = FALSE,
  ...
)
```

Arguments

`age`	age in years
`height`	height in kg. Optional, if specified, will calculate closest percentile and return in list as ‘percentile‘
`sex`	either ‘male‘ or ‘female‘
`return_median`	just return the median expected value
`...`	parameters passed to ‘read_who_table()‘

Examples

```
pct_height_for_age(age = 5, sex = "female")
pct_height_for_age(age = 5, height = 112, sex = "female")
```

`pct_weight_for_age` *Percentile weight for age for children*

Description

Based on tables from WHO: http://www.who.int/childgrowth/standards/weight_for_age/en/

Usage

```
pct_weight_for_age(
  age = NULL,
  weight = NULL,
  sex = NULL,
  return_median = FALSE,
  ...
)
```

Arguments

`age`	age in years
`weight`	weight in kg. Optional, if specified, will calculate closest percentile and return in list as ‘percentile‘
`sex`	either ‘male‘ or ‘female‘
`return_median`	just return the median expected value
`...`	parameters passed to ‘read_who_table()‘

Examples

```
pct_weight_for_age(age = 5, sex = "female")
pct_weight_for_age(age = 5, weight = 20, sex = "female")
```

`pk_1cmt_bolus` *Concentration predictions for 1-compartmental PK model after single or multiple bolus doses*

Description

Concentration predictions for 1-compartmental PK model after single or multiple bolus doses

Usage

```
pk_1cmt_bolus(t = c(0:24), dose = 100, tau = 12, CL = 3, V = 30, ruv = NULL)
```

Arguments

`t`	vector of time
`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution
`ruv`	residual error (list)

Examples

```
pk_1cmt_bolus(dose = 500, tau = 12, CL = 5, V = 50)
pk_1cmt_bolus(dose = 500, tau = 12, CL = 5, V = 50, t = 24)
pk_1cmt_bolus(
  dose = 500, tau = 12, CL = 5, V = 50,
  ruv = list(prop = 0.1, add = 0.1))
```

`pk_1cmt_bolus_cmax_ss` *Cmax for linear 1-compartment PK model at steady state, bolus dosing*

Description

Takes single values for dose or model parameters, or vector of either dose or parameters (but not both).

Usage

```
pk_1cmt_bolus_cmax_ss(dose = 100, tau = 12, CL = 3, V = 30, ruv = NULL)
```

Arguments

`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of distrubition
`ruv`	residual variability, specified as list with optional arguments for proportional, additive, or exponential components, e.g. ‘list(prop=0.1, add=1, exp=0)‘

Examples

```
pk_1cmt_bolus_cmax_ss(
  dose = 500, tau = 12, CL = 5, V = 50)
```

`pk_1cmt_bolus_cmin_ss` *Cmin (trough) for linear 1-compartment PK model at steady state, bolus dosing*

Description

Takes single values for dose or model parameters, or vector of either dose or parameters (but not both).

Usage

```
pk_1cmt_bolus_cmin_ss(dose = 100, tau = 12, CL = 3, V = 30, ruv = NULL)
```

Arguments

`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of distrubition
`ruv`	residual variability, specified as list with optional arguments for proportional, additive, or exponential components, e.g. ‘list(prop=0.1, add=1, exp=0)‘

Examples

```
pk_1cmt_bolus_cmin_ss(
  dose = 500, tau = 12, CL = 5, V = 50)
```

`pk_1cmt_bolus_dose_from_cmax` *Calculate dose to achieve steady state Cmax for 1-compartmental PK model bolus dosing at steady state*

Description

Calculate dose to achieve steady state Cmax for 1-compartmental PK model bolus dosing at steady state

Usage

```
pk_1cmt_bolus_dose_from_cmax(cmax = 1, tau = 12, CL = 3, V = 30)
```

Arguments

`cmax`	desired trough concentration
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution

Examples

```
dos <- pk_1cmt_bolus_dose_from_cmax(
  cmax = 10, tau = 12, CL = 5, V = 50)
find_nearest_dose(dos, 100)
```

`pk_1cmt_bolus_dose_from_cmin` *Calculate dose to achieve steady state trough for 1-compartmental PK model bolus dosing at steady state*

Description

Calculate dose to achieve steady state trough for 1-compartmental PK model bolus dosing at steady state

Usage

```
pk_1cmt_bolus_dose_from_cmin(cmin = 1, tau = 12, CL = 3, V = 30)
```

Arguments

`cmin`	desired trough concentration
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution

Examples

```
dos <- pk_1cmt_bolus_dose_from_cmin(
  cmin = 5, tau = 12, CL = 5, V = 50)
find_nearest_dose(dos, 100)
```

`pk_1cmt_bolus_ss` *Concentration predictions for 1-compartmental PK model with bolus dosing at steady state*

Description

Concentration predictions for 1-compartmental PK model with bolus dosing at steady state

Usage

```
pk_1cmt_bolus_ss(t = c(0:24), dose = 100, tau = 12, CL = 3, V = 30, ruv = NULL)
```

Arguments

`t`	vector of time
`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution
`ruv`	residual variability, specified as list with optional arguments for proportional, additive, or exponential components, e.g. ‘list(prop=0.1, add=1, exp=0)‘

Examples

```
pk_1cmt_bolus_ss(dose = 500, tau = 12, CL = 5, V = 50)
pk_1cmt_bolus_ss(
  dose = 500, tau = 12, CL = 5, V = 50,
  ruv = list(prop = 0.1, add = 0.1))
```

`pk_1cmt_inf` *Concentration predictions for 1-compartmental PK model after single or multiple bolus doses*

Description

Concentration predictions for 1-compartmental PK model after single or multiple bolus doses

Usage

```
pk_1cmt_inf(
  t = c(0:24),
  dose = 100,
  tau = 12,
  t_inf = 2,
  CL = 3,
  V = 30,
  ruv = NULL
)
```

Arguments

`t`	vector of time
`dose`	dose
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance
`V`	volume of distribution
`ruv`	residual error (list)

Examples

```
pk_1cmt_inf(dose = 500, tau = 12, t_inf = 2, CL = 5, V = 50)
pk_1cmt_inf(
  dose = 500, tau = 12, t_inf = 2, CL = 5, V = 50,
  ruv = list(prop = 0.1, add = 0.1))
```

`pk_1cmt_inf_cmax_ss` *Cmax for linear 1-compartment PK model at steady state*

Description

Takes single values for dose or model parameters, or vector of either dose or parameters (but not both).

Usage

```
pk_1cmt_inf_cmax_ss(dose, tau, CL, V, t_inf, ruv = NULL)
```

Arguments

`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of distrubition
`t_inf`	infusion time
`ruv`	residual variability, specified as list with optional arguments for proportional, additive, or exponential components, e.g. ‘list(prop=0.1, add=1, exp=0)‘

Examples

```
pk_1cmt_inf_cmax_ss(dose = 500, tau = 12, t_inf = 2, CL = 5, V = 50)
```

`pk_1cmt_inf_cmin_ss` *Cmin (trough) for linear 1-compartment PK model at steady state*

Description

Takes single values for dose or model parameters, or vector of either dose or parameters (but not both).

Usage

```
pk_1cmt_inf_cmin_ss(
  dose = 100,
  tau = 12,
  CL = 3,
  V = 30,
  t_inf = 2,
  ruv = NULL
)
```

Arguments

`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of distrubition
`t_inf`	infusion time
`ruv`	residual variability, specified as list with optional arguments for proportional, additive, or exponential components, e.g. ‘list(prop=0.1, add=1, exp=0)‘

Examples

```
pk_1cmt_inf_cmin_ss(dose = 500, tau = 12, t_inf = 2, CL = 5, V = 50)
```

`pk_1cmt_inf_dose_for_range` *Calculate dose based on a given AUC24, Cmax, and Cmin, assuming 1-compartment model*

Description

Calculate dose based on a given AUC24, Cmax, and Cmin, assuming 1-compartment model

Usage

```
pk_1cmt_inf_dose_for_range(
  target = 500,
  type = "auc",
  conc_range = c(10, 40),
  parameters = list(),
  interval = 24,
  t_inf = 1,
  optimize_interval = TRUE,
  round_interval = TRUE
)
```

Arguments

`target`	numeric value of target
`type`	target type, one of ‘auc‘, ‘auc24‘, ‘ctrough‘, ‘cmin‘
`conc_range`	concentration range to stay within, vector of length 2
`parameters`	list of ‘CL‘ and ‘V‘, or ‘KEL‘ and ‘CL‘
`interval`	dosing interval
`t_inf`	infusion time
`optimize_interval`	find optimal interval (to stay within ‘conc_range‘?
`round_interval`	round interval to nearest nominal interval?

`pk_1cmt_inf_dose_from_cmax`

Calculate dose to achieve steady state Cmax for 1-compartmental PK model with infusion dosing at steady state

---

Description

Calculate dose to achieve steady state Cmax for 1-compartmental PK model with infusion dosing at steady state

Usage

```
pk_1cmt_inf_dose_from_cmax(cmax = 1, tau = 12, t_inf = 1, CL = 3, V = 30)
```

Arguments

`cmax`	desired trough concentration
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance
`V`	volume of distribution

Examples

```
pk_1cmt_inf_dose_from_cmax(cmax = 20, tau = 12, t_inf = 2, CL = 5, V = 50)
```

`pk_1cmt_inf_dose_from_cmin`

Calculate dose to achieve steady state trough for 1-compartmental PK model with infusion dosing at steady state

---

Description

Calculate dose to achieve steady state trough for 1-compartmental PK model with infusion dosing at steady state

Usage

```
pk_1cmt_inf_dose_from_cmin(cmin = 1, tau = 12, t_inf = 1, CL = 3, V = 30)
```

Arguments

`cmin`	desired trough concentration
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance
`V`	volume of distribution

Examples

```
dos <- pk_1cmt_inf_dose_from_cmin(
  cmin = 20, tau = 12, t_inf = 2,
  CL = 5, V = 50)
find_nearest_dose(dos, 100)
```

`pk_1cmt_inf_ss` *Concentration predictions for 2-compartmental PK model with infusion dosing at steady state*

Description

Concentration predictions for 2-compartmental PK model with infusion dosing at steady state

Usage

```
pk_1cmt_inf_ss(
  t = c(0:24),
  dose = 100,
  t_inf = 1,
  tau = 12,
  CL = 3,
  V = 30,
  ruv = NULL
)
```

Arguments

`t`	vector of time
`dose`	dose
`t_inf`	infusion time
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution
`ruv`	residual variability, specified as list with optional arguments for proportional, additive, or exponential components, e.g. ‘list(prop=0.1, add=1, exp=0)‘

Examples

```
pk_1cmt_inf_ss(dose = 500, tau = 12, t_inf = 2, CL = 5, V = 50)
pk_1cmt_inf_ss(
  dose = 500, tau = 12, t_inf = 2, CL = 5, V = 50,
  ruv = list(prop = 0.1, add = 0.1))
```

`pk_1cmt_oral` *Concentration predictions for 1-compartmental oral PK model after single or multiple bolus doses*

Description

Concentration predictions for 1-compartmental oral PK model after single or multiple bolus doses

Usage

```
pk_1cmt_oral(
  t = c(0:24),
  dose = 100,
  tau = 12,
  KA = 1,
  CL = 3,
  V = 30,
  F = 1,
  ruv = NULL
)
```

Arguments

`t`	vector of time
`dose`	dose
`tau`	dosing interval
`KA`	absorption rate
`CL`	clearance
`V`	volume of distribution
`F`	bioavailability, commonly between 0 an 1.
`ruv`	residual error (list)

References

Garrett ER. The Bateman function revisited: a critical reevaluation of the quantitative expressions to characterize concentrations in the one compartment body model as a function of time with first-order invasion and first-order elimination. J Pharmacokinet Biopharm (1994) 22(2):103-128.

Bialer M. A simple method for determining whether absorption and elimination rate constants are equal in the one-compartment open model with first-order processes. J Pharmacokinet Biopharm (1980) 8(1):111-113

Nielsen JC, Hutmacher MM et al. J Pharmacokinet Pharmacodyn. 2012 Dec;39(6):619-34. doi: 10.1007/s10928-012-9274-0. Epub 2012 Sep 23.

https://static-content.springer.com/esm/art

Examples

```
pk_1cmt_oral(dose = 500, tau = 12, CL = 5, V = 50, KA = 1)
```

`pk_1cmt_t12` *Calculate terminal half-life for 1-compartment model*

Description

Calculate terminal half-life for 1-compartment model

Usage

```
pk_1cmt_t12(CL = 3, V = 30)
```

Arguments

`CL`	clearance
`V`	volume of central compartment

Examples

```
pk_1cmt_t12(CL = 5, V = 50)
```

`pk_2cmt_bolus` *Concentration predictions for 2-compartmental PK model, single or multiple bolus doses*

Description

Concentration predictions for 2-compartmental PK model, single or multiple bolus doses

Usage

```
pk_2cmt_bolus(
  t = c(0:24),
  dose = 100,
  tau = 12,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`t`	vector of time
`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual error (list)

Examples

```
pk_2cmt_bolus(dose = 1000, tau = 24, CL = 5, V = 50, Q = 15, V2 = 200)
```

`pk_2cmt_bolus_cmax_ss` *Cmax for 2-compartmental PK model, bolus dosing at steady state*

Description

Cmax for 2-compartmental PK model, bolus dosing at steady state

Usage

```
pk_2cmt_bolus_cmax_ss(
  dose = 100,
  tau = 12,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual error (list)

Examples

```
pk_2cmt_bolus_cmax_ss(dose = 1000, tau = 12, CL = 5, V = 50, Q = 20, V2 = 200)
```

`pk_2cmt_bolus_cmin_ss` *Cmin (trough) for 2-compartmental PK model, bolus dosing at steady state*

Description

Cmin (trough) for 2-compartmental PK model, bolus dosing at steady state

Usage

```
pk_2cmt_bolus_cmin_ss(
  dose = 100,
  tau = 12,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual error (list)

Examples

```
pk_2cmt_bolus_cmin_ss(dose = 1000, tau = 12, CL = 5, V = 50, Q = 20, V2 = 200)
```

`pk_2cmt_bolus_dose_from_cmax`

Calculate dose to achieve steady state Cmax for 2-compartmental PK model bolus dosing at steady state

---

Description

Calculate dose to achieve steady state Cmax for 2-compartmental PK model bolus dosing at steady state

Usage

```
pk_2cmt_bolus_dose_from_cmax(
  cmax = 1,
  tau = 12,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20
)
```

Arguments

`cmax`	desired trough concentration
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment

Examples

```
dos <- pk_2cmt_bolus_dose_from_cmax(
  cmax = 10, tau = 12,
  CL = 5, V = 50, Q = 20, V2 = 200)
find_nearest_dose(dos, 100)
```

`pk_2cmt_bolus_dose_from_cmin` *Calculate dose to achieve steady state trough for 2-compartmental PK model bolus dosing at steady state*

Description

Calculate dose to achieve steady state trough for 2-compartmental PK model bolus dosing at steady state

Usage

```
pk_2cmt_bolus_dose_from_cmin(
  cmin = 1,
  tau = 12,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20
)
```

Arguments

`cmin`	desired trough concentration
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment

Examples

```
dos <- pk_2cmt_bolus_dose_from_cmin(
  cmin = 5, tau = 12,
  CL = 5, V = 50, Q = 20, V2 = 200)
find_nearest_dose(dos, 100)
```

`pk_2cmt_bolus_ss` *Concentration predictions for 2-compartmental PK model, bolus dosing at steady state*

Description

Concentration predictions for 2-compartmental PK model, bolus dosing at steady state

Usage

```
pk_2cmt_bolus_ss(
  t = c(0:24),
  dose = 100,
  tau = 12,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`t`	vector of time
`dose`	dose
`tau`	dosing interval
`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual error (list)

Examples

```
pk_2cmt_bolus_ss(dose = 1000, tau = 12, CL = 5, V = 50, Q = 20, V2 = 200)
```

`pk_2cmt_inf` *Concentration predictions for 2-compartmental PK model, single or multiple infusions*

Description

Concentration predictions for 2-compartmental PK model, single or multiple infusions

Usage

```
pk_2cmt_inf(
  t = c(0:24),
  dose = 100,
  tau = 12,
  t_inf = 1,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`t`	vector of time
`dose`	dose
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual error (list)

`pk_2cmt_inf_cmax_ss` *Cmax (trough) for 2-compartmental PK model, bolus dosing at steady state*

Description

Cmax (trough) for 2-compartmental PK model, bolus dosing at steady state

Usage

```
pk_2cmt_inf_cmax_ss(
  dose = 100,
  tau = 12,
  t_inf = 1,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`dose`	dose
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual error (list)

Examples

```
pk_2cmt_inf_cmax_ss(
  dose = 1000, tau = 12, t_inf = 2,
  CL = 5, V = 50, Q = 20, V2 = 200)
```

`pk_2cmt_inf_cmin_ss` *Cmin (trough) for 2-compartmental PK model, bolus dosing at steady state*

Description

Cmin (trough) for 2-compartmental PK model, bolus dosing at steady state

Usage

```
pk_2cmt_inf_cmin_ss(
  dose = 100,
  tau = 12,
  t_inf = 1,
  CL = 3,
  V = 30,
```

```
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`dose`	dose
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual error (list)

Examples

```
pk_2cmt_inf_cmin_ss(
  dose = 1000, tau = 12, t_inf = 2,
  CL = 5, V = 50, Q = 20, V2 = 200)
```

`pk_2cmt_inf_dose_from_cmax` *Calculate dose to achieve steady state Cmax for 2-compartmental PK model with infusion dosing at steady state*

Description

Calculate dose to achieve steady state Cmax for 2-compartmental PK model with infusion dosing at steady state

Usage

```
pk_2cmt_inf_dose_from_cmax(
  cmax = 1,
  tau = 12,
  t_inf = 1,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20
)
```

Arguments

`cmax`	desired trough concentration
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance
`V`	volume of distribution
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment

Examples

```
dos <- pk_2cmt_inf_dose_from_cmax(
  cmax = 25, tau = 12, t_inf = 2,
  CL = 5, V = 50, Q = 20, V2 = 200)
find_nearest_dose(dos, 100)
```

`pk_2cmt_inf_dose_from_cmin` *Calculate dose to achieve steady state trough for 2-compartmental PK model with infusion dosing at steady state*

Description

Calculate dose to achieve steady state trough for 2-compartmental PK model with infusion dosing at steady state

Usage

```
pk_2cmt_inf_dose_from_cmin(
  cmin = 1,
  tau = 12,
  t_inf = 1,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20
)
```

Arguments

`cmin`	desired trough concentration
`tau`	dosing interval
`t_inf`	infusion time
`CL`	clearance

`V`	volume of distribution
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment

Examples

```
dos <- pk_2cmt_inf_dose_from_cmin(
  cmin = 10, tau = 12, t_inf = 2,
  CL = 5, V = 50, Q = 20, V2 = 200)
find_nearest_dose(dos, 100)
```

`pk_2cmt_inf_ss` *Concentration predictions for 2-compartmental PK model with infusion dosing at steady state*

Description

Concentration predictions for 2-compartmental PK model with infusion dosing at steady state

Usage

```
pk_2cmt_inf_ss(
  t = c(0:24),
  dose = 100,
  t_inf = 1,
  tau = 12,
  CL = 3,
  V = 30,
  Q = 2,
  V2 = 20,
  ruv = NULL
)
```

Arguments

`t`	vector of time
`dose`	dose
`t_inf`	infusion time
`tau`	dosing interval
`CL`	clearance
`V`	volume of distribution
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`ruv`	residual variability, specified as list with optional arguments for proportional, additive, or exponential components, e.g. ‘list(prop=0.1, add=1, exp=0)‘

Examples

```
pk_2cmt_inf_ss(
  dose = 1000, tau = 12, t_inf = 2,
  CL = 5, V = 50, Q = 20, V2 = 200)
```

`pk_2cmt_t12` *Calculate half-life(s) for 2-compartment model*

Description

Calculate half-life(s) for 2-compartment model

Usage

```
pk_2cmt_t12(CL = 3, V = 30, Q = 2, V2 = 20, phase = c("both", "alpha", "beta"))
```

Arguments

`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`phase`	‘alpha‘, ‘beta‘ (default) or ‘both‘ to indicate initial (distribution) or terminal (elimination) phase.

Examples

```
pk_2cmt_t12(CL = 5, V = 50, Q = 20, V2 = 200)
```

`pk_2cmt_t12_interval` *Calculate average half-life for 2-compartment model during a specific interval*

Description

Calculate average half-life for 2-compartment model during a specific interval

Usage

```
pk_2cmt_t12_interval(CL = 3, V = 30, Q = 2, V2 = 20, tau = 12, t_inf = NULL)
```

Arguments

`CL`	clearance
`V`	volume of central compartment
`Q`	inter-compartimental clearance
`V2`	volume of peripheral compartment
`tau`	interval (hours)
`t_inf`	infusion time (hours)

Examples

```
pk_2cmt_t12_interval(CL = 5, V = 50, Q = 20, V2 = 200, tau = 12, t_inf = 2)
```

`print.nca_output` *Print function for output from nca()*

Description

Print function for output from nca()

Usage

```
## S3 method for class 'nca_output'
print(x, ...)
```

Arguments

`x`	output object (list) from nca()
`...`	variables past on to print function

`read_who_table` *Read WHO growth tables*

Description

Provides a data frame of the WHO growth table for a given age, sex, and type of measurement.

Usage

```
read_who_table(sex = NULL, age = NULL, type = "wfa")
```

Arguments

`sex,`	either `male` or `female`
`age`	age in years
`type`	table type, choose from `wfa` (weight for age), `lhfa` (length for age)

Details

This function uses files included in `system.file(package = "clinPK")`. Previously this function also gave the option to download the tables from WHO, but the original URL ("http://www.who.int/entity/childgrowth/standar no longer exists as of 2021-05-19.

`relative2absolute_bsa` *Convert quantity expressed relative to BSA to absolute units*

Description

Often used for eGFR estimates

Usage

```
relative2absolute_bsa(quantity, bsa = NULL, ...)
```

Arguments

`quantity`	quantity expressed in units /1.73m2
`bsa`	ideal body weight in kg
`...`	arguments passed on to ‘calc_bsa‘, if bsa is NULL

Value

quantity expressed in absolute units

Examples

```
relative2absolute_bsa(quantity = 60, bsa = 1.6)
relative2absolute_bsa(quantity = 60, weight = 14, height = 90, method = "dubois")
```

`remove_lt_gt` *Remove less-than/greater-than symbols and convert to numeric*

Description

The following characters will be removed from strings: <, >, =, space. If string contains other characters, the original string will be returned.

Usage

```
remove_lt_gt(x)
```

Arguments

`x`	Vector of numbers possibly containing extraneous strings.

Value

If non-numeric characters were successfully removed, returns a numeric vector. If some elements of ‘x‘ contained other characters, their original value will be returned and the result will be a character vector.

`time_to_ss` *Time to steady state In either time units or number of doses*

Description

Time to steady state In either time units or number of doses

Usage

```
time_to_ss(kel = NULL, halflife = NULL, ss = 0.9, in_doses = FALSE, tau = NULL)
```

Arguments

`kel`	drug elimination rate
`halflife`	halflife. Either ‘kel‘ or ‘halflife‘ is required.
`ss`	level considered "steady state", e.g. ‘0.9‘ is 90% of true steady state.
`in_doses`	return the number of doses instead of time unit? Default ‘FALSE‘. Requires ‘tau‘ as well.
`tau`	dosing interval

Examples

```
time_to_ss(halflife = 12, ss = 0.9)
time_to_ss(halflife = 16, ss = 0.95, in_doses = TRUE, tau = 12)
```

`valid_units` *Valid units*

Description

Return recognized units for height, weight, age, scr.

Usage

```
valid_units(covariate = c("height", "weight", "age", "scr"))
```

Arguments

`covariate`	Covariate (one of "height", "weight", "age", "scr")

Value

Vector of valid units for the given covariate

Examples

```
valid_units("height")
valid_units("weight")
```

`weight2kg` *Convert any weight unit to kg*

Description

Convert any weight unit to kg

Usage

```
weight2kg(value = NULL, unit = NULL)
```

Arguments

`value`	weight in any allowed unit
`unit`	unit of weight, one of "lb", "lbs", "pound", "pounds", "oz", "ounce", "ounces", "g", "gram", "grams"

Examples

```
weight2kg(250, unit = "oz")
weight2kg(250, unit = "pounds")
weight2kg(250, unit = "lbs")
```

`%>=%` *Greater-than-or-equal-to with a little room for floating point precision issues*

Description

Greater-than-or-equal-to with a little room for floating point precision issues

Usage

```
x %>=% y
```

Arguments

`x`	Numeric vector
`y`	Numeric vector

`%<=%` *Less-than-or-equal-to with a little room for floating point precision issues*

Description

Less-than-or-equal-to with a little room for floating point precision issues

Usage

```
x %<=% y
```

Arguments

`x`	Numeric vector
`y`	Numeric vector

Index

%<=%, 68
%>=%, 68

absolute2relative_bsa, 4
accumulation_ratio, 5
add_ruv, 5
as.num, 6
auc2dose, 6

calc_abw, 7
calc_aki_stage, 7
calc_amts_for_conc, 9
calc_baseline_scr, 10
calc_bmi, 11
calc_bsa, 11
calc_creat, 12
calc_creat_neo, 13
calc_dosing_weight, 13
calc_egfr, 14
calc_egfr_cystatin, 18
calc_ffm, 18
calc_ibw, 20
calc_kel_double_tdm, 21
calc_kel_single_tdm, 22
calc_kgfr, 23
calc_lbw, 24
calc_t12, 25
check_covs_available, 26
clinPK-package, 4
cm2inch, 26
conc2mol, 27
convert_creat_assay, 27
convert_creat_unit, 28
convert_flow_unit, 28

dose2auc, 29

egfr_bedside_schwartz *(calc_egfr)*, 14
egfr_ckd_epi *(calc_egfr)*, 14
egfr_ckd_epi_as_2021 *(calc_egfr)*, 14
egfr_cockcroft_gault *(calc_egfr)*, 14
egfr_cockcroft_gault_sci *(calc_egfr)*, 14
egfr_cov_reqs, 30
egfr_jelliffe *(calc_egfr)*, 14
egfr_jelliffe_unstable *(calc_egfr)*, 14
egfr_malmo_lund *(calc_egfr)*, 14
egfr_mdrd *(calc_egfr)*, 14
egfr_schwartz *(calc_egfr)*, 14
egfr_wright *(calc_egfr)*, 14

ffm_al_sallami *(calc_ffm)*, 18
ffm_bucaloiu *(calc_ffm)*, 18
ffm_garrow_webster *(calc_ffm)*, 18
ffm_hume *(calc_ffm)*, 18
ffm_james *(calc_ffm)*, 18
ffm_janmahasatian_green *(calc_ffm)*, 18
ffm_storset *(calc_ffm)*, 18
find_factor, 30
find_nearest_dose, 31
find_nearest_interval, 31
fraction_of_ss, 32

ibw_devine, 32
ibw_standard, 33
inch2cm, 33
is.nil, 34

kdigo_stage, 34
kg2lbs, 35
kg2oz, 35

lbs2kg, 36

mol2conc, 36

nca, 37
nca_trapezoid, 38

oz2kg, 39

pct_bmi_for_age, 39

pct_for_age_generic, 40
pct_height_for_age, 40
pct_weight_for_age, 41
pk_1cmt_bolus, 42
pk_1cmt_bolus_cmax_ss, 42
pk_1cmt_bolus_cmin_ss, 43
pk_1cmt_bolus_dose_from_cmax, 44
pk_1cmt_bolus_dose_from_cmin, 44
pk_1cmt_bolus_ss, 45
pk_1cmt_inf, 46
pk_1cmt_inf_cmax_ss, 47
pk_1cmt_inf_cmin_ss, 47
pk_1cmt_inf_dose_for_range, 48
pk_1cmt_inf_dose_from_cmax, 49
pk_1cmt_inf_dose_from_cmin, 49
pk_1cmt_inf_ss, 50
pk_1cmt_oral, 51
pk_1cmt_t12, 52
pk_2cmt_bolus, 52
pk_2cmt_bolus_cmax_ss, 53
pk_2cmt_bolus_cmin_ss, 54
pk_2cmt_bolus_dose_from_cmax, 55
pk_2cmt_bolus_dose_from_cmin, 56
pk_2cmt_bolus_ss, 57
pk_2cmt_inf, 58
pk_2cmt_inf_cmax_ss, 58
pk_2cmt_inf_cmin_ss, 59
pk_2cmt_inf_dose_from_cmax, 60
pk_2cmt_inf_dose_from_cmin, 61
pk_2cmt_inf_ss, 62
pk_2cmt_t12, 63
pk_2cmt_t12_interval, 63
print.nca_output, 64

read_who_table, 64
relative2absolute_bsa, 65
remove_lt_gt, 66

time_to_ss, 66

valid_units, 67

weight2kg, 67