Package ‘httk’

September 22, 2022

Version 2.2.1
Date 2022-09-16
Title High-Throughput Toxicokinetics

Description Pre-made models that can be rapidly tailored to various chemicals and species using chemical-specific in vitro data and physiological information. These tools allow incorporation of chemical toxicokinetics (‘TK’) and in vitro-in vivo extrapolation (‘IVIVE’) into bioinformatics, as described by Pearce et al. (2017) (<doi:10.18637/jss.v079.i04>). Chemical-specific in vitro data characterizing toxicokinetics can be been obtained from relatively high-throughput experiments. The chemical-independent (‘generic’) physiologically-based (‘PBTK’) and empirical (for example, one compartment) ‘TK’ models included here can be parameterized with in vitro data or in silico predictions which are provided for thousands of chemicals, multiple exposure routes, and various species. The models are systems of ordinary differential equations that are solved using compiled (C-based) code for speed. A Monte Carlo sampler is included for simulating human biological variability (Ring et al., 2017 <doi:10.1016/j.envint.2017.06.004>) and propagating parameter uncertainty (Wambaugh et al., 2019 <doi:10.1093/toxsci/kfz205>). Empirically calibrated methods are included for predicting tissue:plasma partition coefficients and volume of distribution (Pearce et al., 2017 <doi:10.1007/s10928-017-9548-7>). These functions and data provide a set of tools for using IVIVE to convert concentrations from high-throughput screening experiments (for example, Tox21, ToxCast) to real-world exposures via reverse dosimetry (also known as ‘RTK’) (Wetmore et al., 2015 <doi:10.1093/toxsci/kfv171>).

Depends R (>= 2.10)
Imports deSolve, msm, data.table, survey, mvtnorm, truncnorm, stats, graphics, utils, magrittr, purrr, methods, Rdpack

RdMacros Rdpack

Suggests ggplot2, knitr, rmarkdown, R.rsp, GGally, gplots, scales, EnvStats, MASS, RColorBrewer, TeachingDemos, classInt, ks, stringr, reshape, reshape2, viridis, gmodels,
License GPL-3
LazyData true
LazyDataCompression xz
Encoding UTF-8
VignetteBuilder knitr, R.rsp
RoxygenNote 7.2.1
URL https://www.epa.gov/chemical-research/rapid-chemical-exposure-and-dose-research
BugReports https://github.com/USEPA/CompTox-ExpoCast-httk
NeedsCompilation yes
Author John Wambaugh [aut, cre] (<https://orcid.org/0000-0002-4024-534X>),
Sarah Davidson [aut] (<https://orcid.org/0000-0002-2891-9380>),
Robert Pearce [aut] (<https://orcid.org/0000-0003-3168-4049>),
Caroline Ring [aut] (<https://orcid.org/0000-0002-0463-1251>),
Greg Honda [aut] (<https://orcid.org/0000-0001-7713-9850>),
Mark Sfeir [aut],
Matt Linakis [aut] (<https://orcid.org/0000-0003-0526-2395>),
Dustin Kapraun [aut] (<https://orcid.org/0000-0001-5570-6383>),
Miyuki Breen [ctb] (<https://orcid.org/0000-0001-8511-4653>),
Shannon Bell [ctb] (<https://orcid.org/0000-0002-5754-6085>),
Xiaoqing Chang [ctb] (<https://orcid.org/0000-0003-0752-1848>),
Todor Antonijevic [ctb] (<https://orcid.org/0000-0002-0248-8412>),
Jimena Davis [ctb],
James Sluka [ctb] (<https://orcid.org/0000-0002-5901-1404>),
Nisha Sipes [ctb] (<https://orcid.org/0000-0003-4203-6426>),
Barbara Wetmore [ctb] (<https://orcid.org/0000-0002-6878-5348>),
Woodrow Setzer [ctb] (<https://orcid.org/0000-0002-6709-9186>)
Maintainer John Wambaugh <wambaugh.john@epa.gov>
Copyright This package is primarily developed by employees of the U.S.
Federal government as part of their official duties and is
therefore public domain.

\[R\] topics documented:

httk-package .................................................. 6
add_chemtable .................................................. 7
age_draw_smooth ............................................... 9
armitage_estimate_sarea ....................................... 10
armitage_eval .................................................. 11
armitage_input ................................................ 16
augment.table .................................................. 17
available_rblood2plasma ...................................... 18
aylward2014 .................................................... 19
blood_mass_correct .......................................... 20
blood_weight ................................................... 20
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>bmiage</td>
<td>21</td>
</tr>
<tr>
<td>body_surface_area</td>
<td>22</td>
</tr>
<tr>
<td>bone_mass_age</td>
<td>23</td>
</tr>
<tr>
<td>brain_mass</td>
<td>24</td>
</tr>
<tr>
<td>calc_analytic_css</td>
<td>24</td>
</tr>
<tr>
<td>calc_analytic_css_1comp</td>
<td>27</td>
</tr>
<tr>
<td>calc_analytic_css_3comp</td>
<td>28</td>
</tr>
<tr>
<td>calc_analytic_css_3comps</td>
<td>30</td>
</tr>
<tr>
<td>calc_analytic_css_pbtk</td>
<td>31</td>
</tr>
<tr>
<td>calc_css</td>
<td>32</td>
</tr>
<tr>
<td>calc_elimination_rate</td>
<td>34</td>
</tr>
<tr>
<td>calc_fetal_phys</td>
<td>36</td>
</tr>
<tr>
<td>calc_half_life</td>
<td>41</td>
</tr>
<tr>
<td>calc_hep_clearance</td>
<td>43</td>
</tr>
<tr>
<td>calc_hep_bioavailability</td>
<td>44</td>
</tr>
<tr>
<td>calc_hep clearance</td>
<td>45</td>
</tr>
<tr>
<td>calc_hep_fu</td>
<td>47</td>
</tr>
<tr>
<td>calc_ionization</td>
<td>48</td>
</tr>
<tr>
<td>calc_krbc2pu</td>
<td>50</td>
</tr>
<tr>
<td>calc_maternal_bw</td>
<td>51</td>
</tr>
<tr>
<td>calc_mc_css</td>
<td>52</td>
</tr>
<tr>
<td>calc_mc_oral_equiv</td>
<td>57</td>
</tr>
<tr>
<td>calc_mc_tk</td>
<td>62</td>
</tr>
<tr>
<td>calc_rblood2plasma</td>
<td>65</td>
</tr>
<tr>
<td>calc_stats</td>
<td>67</td>
</tr>
<tr>
<td>calc_tkstats</td>
<td>69</td>
</tr>
<tr>
<td>calc_total_clearance</td>
<td>71</td>
</tr>
<tr>
<td>calc_vdist</td>
<td>72</td>
</tr>
<tr>
<td>CAS.checksum</td>
<td>73</td>
</tr>
<tr>
<td>chem.invivo.PK.aggregate.data</td>
<td>74</td>
</tr>
<tr>
<td>chem.invivo.PK.data</td>
<td>75</td>
</tr>
<tr>
<td>chem.invivo.PK.summary.data</td>
<td>78</td>
</tr>
<tr>
<td>chem.physical_and_invitro.data</td>
<td>81</td>
</tr>
<tr>
<td>ckd_epi_eq</td>
<td>84</td>
</tr>
<tr>
<td>concentration_data_Linakis2020</td>
<td>85</td>
</tr>
<tr>
<td>convert_httkpop_1comp</td>
<td>85</td>
</tr>
<tr>
<td>convert_solve_x</td>
<td>86</td>
</tr>
<tr>
<td>convert_units</td>
<td>88</td>
</tr>
<tr>
<td>create_mc_samples</td>
<td>90</td>
</tr>
<tr>
<td>dawson2021</td>
<td>93</td>
</tr>
<tr>
<td>EPA.ref</td>
<td>93</td>
</tr>
<tr>
<td>estimate_gfr</td>
<td>94</td>
</tr>
<tr>
<td>estimate_gfr_ped</td>
<td>95</td>
</tr>
<tr>
<td>estimate_hematocrit</td>
<td>95</td>
</tr>
<tr>
<td>export_pbtk_jarnac</td>
<td>96</td>
</tr>
<tr>
<td>export_pbtk_sbml</td>
<td>97</td>
</tr>
<tr>
<td>fetalpcs</td>
<td>98</td>
</tr>
<tr>
<td>Frank2018invivo</td>
<td>99</td>
</tr>
<tr>
<td>gen_age_height_weight</td>
<td>100</td>
</tr>
<tr>
<td>gen_height_weight</td>
<td>101</td>
</tr>
<tr>
<td>gen_serum_creatinine</td>
<td>102</td>
</tr>
<tr>
<td>get_cheminfo</td>
<td>103</td>
</tr>
</tbody>
</table>
### R topics documented:

- `get_chem_id` .......................... 106
- `get_gfr_category` ..................... 107
- `get_invitroPK_param` ................. 107
- `get_lit_cheminfo` .................... 109
- `get_lit_oral_equiv` .................. 111
- `get_physchem_param` ................. 112
- `get_rblood2plasma` ................... 113
- `get_weight_class` .................... 114
- `get_wetmore_cheminfo` ............... 115
- `get_wetmore_css` .................... 116
- `get_wetmore_oral_equiv` ............. 118
- `hct_h` .................................. 119
- `hematocrit_infants` .................. 120
- `honda.ivive` .......................... 121
- `howgate` ................................ 122
- `httkpop` ................................ 123
- `httkpop_biotophys_default` .......... 127
- `httkpop_direct_resample` ............. 128
- `httkpop_direct_resample_inner` ...... 129
- `httkpop_generate` ..................... 131
- `httkpop.mc` ........................... 135
- `httkpop_virtual_indiv` ............... 136
- `hw_H` .................................. 137
- `in.list` ................................ 138
- `invitro.mc` ............................ 140
- `is.httk` ................................ 142
- `is_in_inclusive` ...................... 144
- `johnson` ................................ 145
- `kapraun2019` ......................... 145
- `kidney_mass_children` ............... 146
- `liver_mass_children` .................. 147
- `load_dawson2021` ...................... 147
- `load_pradeep2020` ..................... 148
- `load_sipes2017` ....................... 149
- `lump_tissues` ......................... 150
- `lung_mass_children` .................. 152
- `mcnally_dt` ............................ 153
- `mecdt` ................................ 154
- `metabolism_data_Linakis2020` ........ 155
- `monte_carlo` ........................... 155
- `Obach2008` ............................ 157
- `onlyp` ................................ 157
- `pancreas_mass_children` ............. 158
- `parameterize_1comp` ................. 158
- `parameterize_3comp` ................. 161
- `parameterize_fetal_pbtk` ............. 163
- `parameterize_gas_pbtk` ............... 166
- `parameterize_pbtk` ................. 170
- `parameterize_schmitt` ............... 173
- `parameterize_steadystate` .......... 174
- `pc.data` ............................... 177
R topics documented:

pearce2017regression .............................................. 178
pharma ................................................................. 178
physiology.data ....................................................... 179
pksim.pcs .............................................................. 180
pradeep2020 ........................................................... 180
predict_partitioning_schmitt ....................................... 181
pregnonpregaucs ..................................................... 183
propagate_invitrouv_1comp ........................................ 183
propagate_invitrouv_3comp ........................................ 184
propagate_invitrouv_pbtk .......................................... 184
reset_httk ............................................................. 185
rfun ................................................................. 186
rmed0non0u95 ........................................................ 186
r_left_censored_norm ............................................. 187
scale_dosing .......................................................... 188
scr_h ................................................................. 189
set_httk_precision ................................................ 190
skeletal_muscle_mass ............................................ 190
skeletal_muscle_mass_children .................................. 191
skin_mass_bosgra .................................................. 192
solve_1comp .......................................................... 192
solve_3comp .......................................................... 196
solve_fetal_pbtk ................................................... 200
solve_gas_pbtk ...................................................... 203
solve_model .......................................................... 208
solve_pbtk ............................................................ 211
spleen_mass_children ............................................ 216
supptab1_Linakis2020 ........................................... 217
supptab2_Linakis2020 ........................................... 217
Tables.Rdata.stamp .............................................. 218
tissue.data .......................................................... 218
tissue_masses_flows ................................................ 219
tissue_scale .......................................................... 220
wambaugh2019 ....................................................... 220
wambaugh2019.nhanes ............................................ 222
wambaugh2019.raw ............................................... 223
wambaugh2019.seem3 ............................................. 225
wambaugh2019.tox21 ............................................... 225
wang2018 ............................................................. 226
well_param .......................................................... 227
Wetmore2012 ........................................................ 228
wfl ................................................................. 228

Index 230
**Description**

Pre-made models that can be rapidly tailored to various chemicals and species using chemical-specific in vitro data and physiological information. These tools allow incorporation of chemical toxicokinetics ("TK") and in vitro-in vivo extrapolation ("IVIVE") into bioinformatics, as described by Pearce et al. (2017) (<doi:10.18637/jss.v079.i04>). Chemical-specific in vitro data characterizing toxicokinetics can be obtained from relatively high-throughput experiments. The chemical-independent ("generic") physiologically-based ("PBTK") and empirical (for example, one compartment) "TK" models included here can be parameterized with in vitro data or in silico predictions which are provided for thousands of chemicals, multiple exposure routes, and various species. The models are systems of ordinary differential equations that are solved using compiled (C-based) code for speed. A Monte Carlo sampler is included for simulating human biological variability (Ring et al., 2017 <doi:10.1016/j.envint.2017.06.004>) and propagating parameter uncertainty (Wambaugh et al., 2019 <doi:10.1093/toxsci/kfv205>). Empirically calibrated methods are included for predicting tissue:plasma partition coefficients and volume of distribution (Pearce et al., 2017 <doi:10.1007/s10928-017-9548-7>). These functions and data provide a set of tools for using IVIVE to convert concentrations from high-throughput screening experiments (for example, Tox21, ToxCast) to real-world exposures via reverse dosimetry (also known as "RTK") (Wetmore et al., 2015 <doi:10.1093/toxsci/kfz171>.

**Author(s)**

John Wambaugh, Robert Pearce, Caroline Ring, Gregory Honda, Nisha Sipes, Jimena Davis, Barbara Wetmore, Woodrow Setzer, Mark Sfeir

**See Also**

PowerPoint Presentation: High-Throughput Toxicokinetics (HTTK) R package


Breen et al. (2021): High-throughput PBTK models for in vitro to in vivo extrapolation

doi:10.18637/jss.v079.i04

Pearce et al. (2017): httk: R Package for High-Throughput Toxicokinetics

doi:10.1021/es501955g

Armitage et al. (2014): Application of mass balance models and the chemical activity concept to facilitate the use of in vitro toxicity data for risk assessment

doi:10.1093/toxsci/kfv171

Incorporating High-Throughput Exposure Predictions with Dosimetry-Adjusted In Vitro Bioactivity to Inform Chemical Toxicity Testing

doi:10.1093/toxsci/kfv118

Wambaugh et al. (2015): Toxicokinetic Triage for Environmental Chemicals

doi:10.1007/s10928-017-9548-7

Pearce et al. (2017): Evaluation and calibration of high-throughput predictions of chemical distribution to tissues

doi:10.1016/j.envint.2017.06.004

Ring et al. (2017): Identifying populations sensitive to environmental chemicals by simulating toxicokinetic variability

doi:10.1021/acs.est.7b00650

Sipes et al. (2017): An Intuitive Approach for Predicting Potential Human Health Risk with the Tox21 10k Library
add_chemtable

Add a table of chemical information for use in making htk predictions.

Description

This function adds chemical-specific information to the table chem.physical_and_invitro.data. This table is queried by the model parameterization functions when attempting to parameterize a model, so adding sufficient data to this table allows additional chemicals to be modeled.

Usage

add_chemtable(
  new.table,  
data.list,  
current.table = NULL,  
reference = NULL,  
species = NULL,  
overwrite = FALSE,  
sig.fig = 4,  
clint.pvalue.overwrite = TRUE,  
allow.na = FALSE
)

Arguments

new.table Object of class data.frame containing one row per chemical, with each chemical minimally described by a CAS number.

data.list This list identifies which properties are to be read from the table. Each item in the list should point to a column in the table new.table. Valid names in the list are: 'Compound', 'CAS', 'DSSTox.GSID' 'SMILES.desalt', 'Reference', 'Species', 'MW', 'logP', 'pKa_Donor', 'pKa_Accept', 'logMA', 'Clint', 'Clint.pValue', 'Funbound.plasma', 'Fgutabs', 'Rblood2plasma'.
current.table This is the table to which data are being added.
reference This is the reference for the data in the new table. This may be omitted if a column in data.list gives the reference value for each chemical.
species This is the species for the data in the new table. This may be omitted if a column in data.list gives the species value for each chemical or if the data are not species-specific (e.g., MW).
add_chemtable

overwrite If overwrite=TRUE then data in current.table will be replaced by any data in new.table that is for the same chemical and property. If overwrite=FALSE (DEFAULT) then new data for the same chemical and property are ignored. Fun-bound.plasma values of 0 (below limit of detection) are overwritten either way.

sig.fig Sets the number of significant figures stored (defaults to 4)

clint.pvalue.overwrite If TRUE then the Cl_int p-value is set to NA when the Cl_int value is changed unless a new p-value is provided. (defaults to TRUE)

allow.na If TRUE (default is FALSE) then NA values are written to the table, otherwise they are ignored.

Value

data.frame A new data.frame containing the data in current.table augmented by new.table

Author(s)

John Wambaugh

Examples

library(httk)
my.new.data <- as.data.frame(c("A","B","C"),stringsAsFactors=FALSE)
my.new.data <- cbind(my.new.data,as.data.frame(c("111-11-2","222-22-0","333-33-5"),
stringsAsFactors=FALSE))
my.new.data <- cbind(my.new.data,as.data.frame(c("DTX1","DTX2","DTX3"),
stringsAsFactors=FALSE))
my.new.data <- cbind(my.new.data,as.data.frame(c(200,200,200)))
my.new.data <- cbind(my.new.data,as.data.frame(c(2,3,4)))
my.new.data <- cbind(my.new.data,as.data.frame(c(0.01,0.02,0.3)))
my.new.data <- cbind(my.new.data,as.data.frame(c(0,10,100)))
colnames(my.new.data) <- c("Name","CASRN","DTXSID","MW","LogP","Fup","CLint")

chem.physical_and_invitro.data <- add_chemtable(my.new.data,
current.table=chem.physical_and_invitro.data,
data.list=list(
Compound="Name",
CAS="CASRN",
DTXSID="DTXSID",
MW="MW",
logP="LogP",
Funbound.plasma="Fup",
Clint="CLint"),
species="Human",
reference="MyPaper 2015")

parameterize_steadystate(chem.name="C")
calc_css(chem.name="B")

# Initialize a column describing proton donors ("acids")
my.new.data$pka.a <- NA
# set chemical C to an acid (pKa_donor = 5):
my.new.data[my.new.data$Name=="C","pka.a"] <- "5"
chem.physical_and_invitro.data <- add_chemtable(my.new.data,
current.table=
   chem.physical_and_invitro.data,
data.list=list(
   Compound="Name",
   CAS="CASRN",
   DTXSID="DTXSID",
   pKa_Donor="pka.a"),
   species="Human",
   reference="MyPaper 2015")

# Note Rblood2plasma and hepatic bioavailability change (relative to above):
parameterize steadystate(chem.name="C")

# Initialize a column describing proton acceptors ("bases")
my.new.data$pka.b <- NA
# set chemical B to a base with multiple pka's (pKa_accept = 7 and 8):
my.new.data[my.new.data$Name=="B","pka.b"] <- "7;8"
chem.physical_and_invitro.data <- add_chemtable(my.new.data,
current.table=
   chem.physical_and_invitro.data,
data.list=list(
   Compound="Name",
   CAS="CASRN",
   DTXSID="DTXSID",
   pKa_Accept="pka.b"),
   species="Human",
   reference="MyPaper 2015")

# Note that average and max change (relative to above):
calc_css(chem.name="B")

---

**age_draw_smooth**

*Draws ages from a smoothed distribution for a given gender/race combination*

**Description**

This function should usually not be called directly by the user. It is used by httkpop_generate() in "virtual-individuals" mode.

**Usage**

`age_draw_smooth(gender, reth, nsamp, agelim_months, nhanes_mec_svy)`

**Arguments**

- `gender` Gender. Either ‘Male’ or ‘Female’.
- `nsamp` Number of ages to draw.
- `agelim_months` Two-element numeric vector giving the minimum and maximum ages in months to include.
- `nhanes_mec_svy` surveydesign object created from mecdt using svydesign (this is done in httkpop_generate)
Value

A named list with members 'ages_months' and 'ages_years', each numeric of length nsamp, giving the sampled ages in months and years.

Author(s)

Caroline Ring

References


Description

Estimate geometry surface area of plastic in well plate based on well plate format suggested values from Corning. option.plastic == TRUE (default) give nonzero surface area (sarea, m²) option.bottom == TRUE (default) includes surface area of the bottom of the well in determining sarea. Optionally include user values for working volume (v_working, m³) and surface area.

Usage

armitage_estimate_sarea(
  tcdata = NA,
  this.well_number = 384,
  this.cell_yield = NA,
  this.v_working = NA
)

Arguments

tcdata A data table with well_number corresponding to plate format, optionally include v_working, sarea, option.bottom, and option.plastic

this.well_number For single value, plate format default is 384, used if is.na(tcdata)==TRUE

this.cell_yield For single value, optionally supply cell_yield, otherwise estimated based on well number

this.v_working For single value, optionally supply working volume, otherwise estimated based on well number (m³)

Value

A data table composed of any input data.table tcdata with only the following columns either created or altered by this function:
<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>well_number</td>
<td>number of wells on plate</td>
<td></td>
</tr>
<tr>
<td>sarea</td>
<td>surface area</td>
<td>m²</td>
</tr>
<tr>
<td>cell_yield</td>
<td>number of cells</td>
<td>cells</td>
</tr>
<tr>
<td>v_working</td>
<td>working (filled) volume of each well</td>
<td>uL</td>
</tr>
<tr>
<td>v_total</td>
<td>total volume of each well</td>
<td>uL</td>
</tr>
</tbody>
</table>

**Author(s)**

Greg Honda

**References**


---

**armitage_eval**

*Evaluate the updated Armitage model*

**Description**

Evaluate the Armitage model for chemical distribution in vitro. Takes input as data table or vectors of values. Outputs a data table. Updates over the model published in Armitage et al. 2014 include binding to plastic walls and lipid and protein compartments in cells.

**Usage**

```
armitage_eval(  
casrn.vector = NA_character_,  
nomconc.vector = 1,  
this.well_number = 384,  
this.FBSf = NA_real_,  
tcdata = NA,  
this.sarea = NA_real_,  
this.v_total = NA_real_,  
this.v_working = NA_real_,  
this.cell_yield = NA_real_,  
this.Tsys = 37,  
this.Tref = 298.15,  
this.option.kbsa2 = FALSE,  
this.option.swat2 = FALSE,  
this.pseudooct = 0.01,  
this.memblip = 0.04,  
this.nlom = 0.2,  
this.P_nlom = 0.035,  
this.P_dom = 0.05,  
this.P_cells = 1,  
this.csalt = 0.15,  
this.celldensity = 1,  
)```
this.cellmass = 3,
this.f_oc = 1,
this.conc_ser_alb = 24,
this.conc_ser_lip = 1.9,
this.Vdom = 0
)

Arguments

casrn.vector For vector or single value, CAS number
nomconc.vector For vector or single value, micromolar nominal concentration (e.g. AC50 value)
this.well_number For single value, plate format default is 384, used if is.na(tcdata)==TRUE
this.FBSf Fraction fetal bovine serum, must be entered by user.
tcdata A data.table with casrn, nomconc, MP, gkow, gkaw, gswat, sarea, v_total, v_working. Otherwise supply single values to this.params.
this.sarea Surface area per well (m^2)
this.v_total Total volume per well (m^3)
this.v_working Working volume per well (m^3)
this.cell_yield Number of cells per well
this.Tsys System temperature (degrees C)
this.Tref Reference temperature (degrees K)
this.option.kbsa2 Use alternative bovine-serum-albumin partitioning model
this.option.swat2 Use alternative water solubility correction
this.pseudooct Pseudo-octanol cell storage lipid content
this.memblip Membrane lipid content of cells
this.nlom Structural protein content of cells
this.P_nlom Proportionality constant to octanol structural protein
this.P_dom Proportionality constant to dissolve organic material
this.P_cells Proportionality constant to octanol storage lipid
this.csalt Ionic strength of buffer, mol/L
this.celldensity Cell density kg/L, g/mL
this.cellmass Mass per cell, ng/cell
this.f_oc 1, everything assumed to be like proteins
this.conc_ser_alb 24 g/L, mass concentration of albumin in serum.
this.conc_ser_lip 1.9 g/L, mass concentration of lipids in serum.
this.Vdom 0 ml, the volume of dissolved organic matter (DOM)
### Value

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
<th>units</th>
</tr>
</thead>
<tbody>
<tr>
<td>casrn</td>
<td>Chemical Abstracts Service Registry Number</td>
<td>mol/L</td>
</tr>
<tr>
<td>nomconc</td>
<td>Nominal Concentration</td>
<td>unitless</td>
</tr>
<tr>
<td>well_number</td>
<td>Number of wells in plate</td>
<td></td>
</tr>
<tr>
<td>sarea</td>
<td>Surface area of well</td>
<td>m^2</td>
</tr>
<tr>
<td>v_total</td>
<td>Total volume of well</td>
<td>m^3</td>
</tr>
<tr>
<td>v_working</td>
<td>Filled volume of well</td>
<td>m^3</td>
</tr>
<tr>
<td>cell_yield</td>
<td>Number of cells</td>
<td>cells</td>
</tr>
<tr>
<td>gkow</td>
<td>log10 octanol to water partition coefficient (PC)</td>
<td>log10</td>
</tr>
<tr>
<td>logHenry</td>
<td>log10 Henry’s law constant ‘</td>
<td>log10 atm-m3/mol</td>
</tr>
<tr>
<td>gswat</td>
<td>log10 Water solubility</td>
<td>log10 mol/L</td>
</tr>
<tr>
<td>MP</td>
<td>Melting Point</td>
<td>degrees Celsius</td>
</tr>
<tr>
<td>MW</td>
<td>Molecular Weight</td>
<td>g/mol</td>
</tr>
<tr>
<td>gkaw</td>
<td>air-water partition coefficient</td>
<td>(mol/m3)/(mol/m3)</td>
</tr>
<tr>
<td>dsm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>duow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>duaw</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dumw</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gkmw</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gkcw</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gkpl</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gksba</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ksalt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tsys</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tref</td>
<td></td>
<td></td>
</tr>
<tr>
<td>option.kbsa2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>option.swat2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FBSf</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pseudoocct</td>
<td></td>
<td></td>
</tr>
<tr>
<td>memblip</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nlon</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P_nlon</td>
<td></td>
<td></td>
</tr>
<tr>
<td>P_dom</td>
<td>dissolved organic matter to water PC</td>
<td>Dimensionless</td>
</tr>
<tr>
<td>P_cells</td>
<td></td>
<td></td>
</tr>
<tr>
<td>csalt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cell_density</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cell_mass</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f_oc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cell_wat</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tcor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vm</td>
<td>Volume of media</td>
<td>L</td>
</tr>
<tr>
<td>Vwell</td>
<td>volume of medium (aqueous phase only)</td>
<td>L</td>
</tr>
<tr>
<td>Vair</td>
<td>volume of head space</td>
<td>L</td>
</tr>
<tr>
<td>Vcells</td>
<td>volume of cells/tissue</td>
<td></td>
</tr>
<tr>
<td>Valb</td>
<td>volume of serum albumin</td>
<td></td>
</tr>
<tr>
<td>Vslip</td>
<td>volume of serum lipids</td>
<td></td>
</tr>
<tr>
<td>Vdom</td>
<td>volume of dissolved organic matter</td>
<td></td>
</tr>
<tr>
<td>F_ratio</td>
<td></td>
<td></td>
</tr>
<tr>
<td>gs1.GSE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>s1.GSE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
gss.GSE
ss.GSE
kmw
kow octanol to water PC
ciair mol/L
calb mol/L
cslip mol/L
cdom concentration of/in dissolved organic matter mol/L
cells mol/L
cplastic mol/L
mwat_s Mass dissolved in water mols
mair Mass in air mols
mbsa Mass bound to bovine serum albumin mols
mslip Mass bound to serum lipids mols
mdom Mass bound to dissolved organic matter mols
mcells Mass in cells mols
mplastic Mass bond to plastic mols
mprecip Mass precipitated out of solution mols
xwat_s Fraction dissolved in water fraction
xair Fraction in the air fraction
xbba Fraction bound to bovine serum albumin fraction
xslip Fraction bound to serum lipids fraction
xdom Fraction bound to dissolved organic matter fraction
xceLLs Fraction within cells fraction
xplastic Fraction bound to plastic fraction
xprecip Fraction precipitated out of solution fraction
eta_free effective availability ratio fraction

c_free.invitro Free concentration in the in vitro media (use for Honda1 and Honda2) micromolar

Author(s)
Greg Honda

References
Examples

```r
corporate_armitage <- library(httk)

# Check to see if we have info on the chemical:
"80-05-7" %in% get_cheminfo()

# We do:
temp <- armitage_eval(casrn.vector = c("80-05-7", "81-81-2"), this.FBSf = 0.1,
this.well_number = 384, nomconc = 10)
print(temp$cfree.invitro)

# Check to see if we have info on the chemical:
"793-24-8" %in% get_cheminfo()

# Since we don't have any info, let's look up phys-chem from dashboard:
cheminfo <- data.frame(
  Compound="6-PPD",
  CASRN="793-24-8",
  DTXSID="DTXSID9025114",
  logP=4.27,
  logHenry=log10(7.69e-8),
  logWSol=log10(1.58e-4),
  MP=99.4,
  MW=268.404
)

# Add the information to HTTK's database:
chem.physical_and_invitro.data <- add_chemtable(
  cheminfo,
  current.table=chem.physical_and_invitro.data,
  data.list=list(
    Compound="Compound",
    CAS="CASRN",
    DTXSID="DTXSID",
    MW="MW",
    logP="logP",
    logHenry="logHenry",
    logWSol="logWSol",
    MP="MP"),
    species="Human",
    reference="CompTox Dashboard 31921")

# Run the Armitage et al. (2014) model:
out <- armitage_eval(
  casrn.vector = "793-24-8",
  this.FBSf = 0.1,
  this.well_number = 384,
  nomconc = 10)
print(out)
```
Armitage et al. (2014) Model Inputs from Honda et al. (2019)

Description

Armitage et al. (2014) Model Inputs from Honda et al. (2019)

Usage

armitage_input

Format

A data frame with 53940 rows and 10 variables:

MP
MW
casrn
compound_name
gkaw
gkow
gswat

Author(s)

Greg Honda

Source

https://www.diamondse.info/

References


augment.table  

Add a parameter value to the chem.physical_and_invitro.data table

Description

This internal function is used by add_chemtable to add a single new parameter to the table of chemical parameters. It should not be typically used from the command line.

Usage

```r
augment.table(
  this.table,
  this.CAS,
  compound.name = NULL,
  this.property,
  value,
  species = NULL,
  reference,
  overwrite = FALSE,
  sig.fig = 4,
  clint.pvalue.overwrite = TRUE,
  allow.na = FALSE
)
```

Arguments

- **this.table**: Object of class data.frame containing one row per chemical.
- **this.CAS**: The Chemical Abstracts Service registry number (CAS-RN) corresponding to the parameter value.
- **compound.name**: A name associated with the chemical (defaults to NULL).
- **this.property**: The property being added/modified.
- **value**: The value being assigned to this.property.
- **species**: This is the species for the data in the new table. This may be omitted if a column in data.list gives the species value for each chemical or if the data are not species-specific (e.g., MW).
- **reference**: This is the reference for the data in the new table. This may be omitted if a column in data.list gives the reference value for each chemical.
- **overwrite**: If overwrite=TRUE then data in current.table will be replaced by any data in new.table that is for the same chemical and property. If overwrite=FALSE (DEFAULT) then new data for the same chemical and property are ignored. Fun-bound.plasma values of 0 (below limit of detection) are overwritten either way.
- **sig.fig**: Sets the number of significant figures stored (defaults to 4)
- **clint.pvalue.overwrite**: If TRUE then the Cl_int p-value is set to NA when the Cl_int value is changed unless a new p-value is provided. (defaults to TRUE)
- **allow.na**: If TRUE (default is FALSE) then NA values are written to the table, otherwise they are ignored.
available_rblood2plasma

Value
data.frame A new data.frame containing the data in current.table augmented by new.table

Author(s)
John Wambaugh

available_rblood2plasma

Find the best available ratio of the blood to plasma concentration constant.

Description
This function finds the best available constant ratio of the blood concentration to the plasma concentration, using get_rblood2plasma and calc_rblood2plasma.

Usage
available_rblood2plasma(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  species = "Human",
  adjusted.Funbound.plasma = TRUE,
  suppress.messages = FALSE
)

Arguments
chem.cas Either the CAS number or the chemical name must be specified.
chem.name Either the chemical name or the CAS number must be specified.
dtxsid EPA's 'DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
adjusted.Funbound.plasma Whether or not to use Funbound.plasma adjustment if calculating Rblood2plasma.
suppress.messages Whether or not to display relevant warning messages to user.

Details
Either retrieves a measured blood:plasma concentration ratio from the chem.physical_and_invitro.data table or calculates it using the red blood cell partition coefficient predicted with Schmitt’s method. If available, in vivo data (from chem.physical_and_invitro.data) for the given species is returned, substituting the human in vivo value when missing for other species. In the absence of in vivo data, the value is calculated with calc_rblood2plasma for the given species. If Funbound.plasma is unavailable for the given species, the human Funbound.plasma is substituted. If none of these are available, the mean human Rblood2plasma from chem.physical_and_invitro.data is returned. details than the description above ~~~
Value

The blood to plasma chemical concentration ratio – measured if available, calculated if not.

Author(s)

Robert Pearce

Examples

available_rblood2plasma(chem.name="Bisphenol A",adjusted.Funbound.plasma=FALSE)
available_rblood2plasma(chem.name="Bisphenol A",species="Rat")

Description

Aylward et al. (2014) compiled measurements of the ratio of maternal to fetal cord blood chemical concentrations at birth for a range of chemicals with environmental routes of exposure, including bromodiphenyl ethers, fluorinated compounds, organochlorine pesticides, polyaromatic hydrocarbons, tobacco smoke components, and vitamins.

Usage

aylward2014

Format

data.frame

Source

Kapraun et al. 2021 (submitted)

References

blood_mass_correct  

Find average blood masses by age.

Description

If blood mass from `blood_weight` is negative or very small, then just default to the mean blood mass by age. (Geigy Scientific Tables, 7th ed.)

Usage

`blood_mass_correct(blood_mass, age_months, age_years, gender, weight)`

Arguments

- `blood_mass`: A vector of blood masses in kg to be replaced with averages.
- `age_months`: A vector of ages in months.
- `age_years`: A vector of ages in years.
- `gender`: A vector of genders (either 'Male' or 'Female').
- `weight`: A vector of body weights in kg.

Value

A vector of blood masses in kg.

Author(s)

Caroline Ring

References


blood_weight  

Predict blood mass.

Description

Predict blood mass based on body surface area and gender, using equations from Bosgra et al. 2012

Usage

`blood_weight(BSA, gender)`

Arguments

- `BSA`: Body surface area in m^2. May be a vector.
- `gender`: Either 'Male' or 'Female'. May be a vector.
**Value**

A vector of blood masses in kg the same length as BSA and gender.

**Author(s)**

Caroline Ring

**References**


---

**bmiage**

A data.table with 434 rows and 5 variables:

- **Sex** Female or Male
- **Agemos** Age in months
- **P5** The 5th percentile BMI for the corresponding sex and age
- **P85** The 85th percentile BMI for the corresponding sex and age
- **P95** The 95th percentile BMI for the corresponding sex and age

**Details**

For children ages 2 to 18, weight class depends on the BMI-for-age percentile.

- **Underweight** <5th percentile
- **Normal weight** 5th-85th percentile
- **Overweight** 85th-95th percentile
- **Obese** >=95th percentile

**Author(s)**

Caroline Ring
body_surface_area

Source
https://www.cdc.gov/growthcharts/data/zscore/bmiagerev.csv

References

Description
Predict body surface area from weight, height, and age, using Mosteller’s formula for age>18 and Haycock’s formula for age<18

Usage
body_surface_area(BW, H, age_years)

Arguments

- BW A vector of body weights in kg.
- H A vector of heights in cm.
- age_years A vector of ages in years.

Value
A vector of body surface areas in cm^2.

Author(s)
Caroline Ring

References
bone_mass_age

Predict bone mass

Description
Predict bone mass from age_years, height, weight, gender, using logistic equations fit to data from Baxter-Jones et al. 2011, or for infants < 1 year, using equation from Koo et al. 2000 (See Price et al. 2003)

Usage
bone_mass_age(age_years, age_months, height, weight, gender)

Arguments
- age_years: Vector of ages in years.
- age_months: Vector of ages in months.
- height: Vector of heights in cm.
- weight: Vector of body weights in kg.
- gender: Vector of genders, either 'Male' or 'Female'.

Value
Vector of bone masses.

Author(s)
Caroline Ring

References
brain_mass Predict brain mass.

Description
Predict brain mass from gender and age.

Usage
brain_mass(gender, age_years)

Arguments
- gender: Vector of genders, either 'Male' or 'Female'
- age_years: Vector of ages in years.

Value
A vector of brain masses in kg.

Author(s)
Caroline Ring

References

calc_analytic_css Calculate the analytic steady state plasma concentration.

Description
This function calculates the analytic steady state plasma or venous blood concentrations as a result of infusion dosing for the three compartment and multiple compartment PBTK models.

Usage
calc_analytic_css(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  parameters = NULL,
  species = "human",
  daily.dose = 1,
  route = "oral",
  exp.conc = 1,
  period = 24,
  exp.duration = 24,
calc_analytic_css

output.units = "uM",
model = "pbtk",
concentration = "plasma",
suppress.messages = FALSE,
tissue = NULL,
restrictive.clearance = TRUE,
bioactive.free.invivo = FALSE,
IVIVE = NULL,
parameterize.args = list(),
...
)

Arguments

chem.name
Either the chemical name, CAS number, or the parameters must be specified.

chem.cas
Either the chemical name, CAS number, or the parameters must be specified.

dtxsid
EPA's DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs

parameters
Chemical parameters from parameterize_pbtk (for model = 'pbtk'), parameterize_3comp (for model = '3compartment), parameterize_1comp(for model = '1compartment') or parameterize_steadystate (for model = '3compartmentss'), overrides chem.name and chem.cas.

species
Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").

daily.dose
Total daily dose, mg/kg BW.

route
Route of exposure (either "oral", "iv", or "inhalation" default "oral").

exp.conc
Specified inhalation exposure concentration for use in assembling 'forcings' data series argument for integrator. Defaults to uM/L

period
For use in assembling forcing function data series 'forcings' argument, specified in hours

exp.duration
For use in assembling forcing function data series 'forcings' argument, specified in hours

output.units
Units for returned concentrations, defaults to uM (specify units = "uM") but can also be mg/L.

model
Model used in calculation,"gas_pbtk" for the gas pbtk model, 'pbtk' for the multiple compartment model, '3compartment' for the three compartment model, '3compartmentss' for the three compartment steady state model, and '1compartment' for one compartment model.

concentration
Desired concentration type: 'blood', 'tissue', or default 'plasma'. In the case that the concentration is for plasma, selecting "blood" will use the blood:plasma ratio to estimate blood concentration. In the case that the argument 'tissue' specifies a particular tissue of the body, concentration defaults to 'tissue' — that is, the concentration in the If cocentration is set to 'blood' or 'plasma' and 'tissue' specifies a specific tissue then the value returned is for the plasma or blood in that specific tissue.

suppress.messages
Whether or not the output message is suppressed.

tissue
Desired steady state tissue concentration. Default is of NULL typically gives whole body plasma concentration.
If TRUE (default), then only the fraction of chemical not bound to protein is available for metabolism in the liver. If FALSE, then all chemical in the liver is metabolized (faster metabolism due to rapid off-binding).

If FALSE (default), then the total concentration is treated as bioactive in vivo. If TRUE, the the unbound (free) plasma concentration is treated as bioactive in vivo. Only works with tissue = NULL in current implementation.

Honda et al. (2019) identified four plausible sets of assumptions for \textit{in vitro-in vivo} extrapolation (IVIVE) assumptions. Argument may be set to "Honda1" through "Honda4". If used, this function overwrites the tissue, restrictive.clearance, and bioactive.free.invivo arguments. See Details below for more information.

List of arguments passed to model’s associated parameterization function, including default.to.human, adjusted.Funbound.plasma, regression, and minimum.Funbound.plasma. The default.to.human argument substitutes missing animal values with human values if true, adjusted.Funbound.plasma returns adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value, regression indicates whether or not to use the regressions in calculating partition coefficients, and minimum.Funbound.plasma is the value to which Monte Carlo draws less than this value are set (default is 0.0001 – half the lowest measured Fup in our dataset).

Additional parameters passed to parameterize function if parameters is NULL.

Concentrations are calculated for the specified model with constant oral infusion dosing. All tissues other than gut, liver, and lung are the product of the steady state plasma concentration and the tissue to plasma partition coefficient.

<table>
<thead>
<tr>
<th>IVIVE</th>
<th>Conc.</th>
<th>Metabolic Clearance</th>
<th>Bioactive Chemical Conc.</th>
<th>TK Statistic Used*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honda1</td>
<td>Veinous</td>
<td>Restrictive</td>
<td>Free</td>
<td>Mean Conc.</td>
</tr>
<tr>
<td>Honda2</td>
<td>Veinous</td>
<td>Restrictive</td>
<td>Free</td>
<td>Max Conc.</td>
</tr>
<tr>
<td>Honda3</td>
<td>Veinous</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Mean Conc.</td>
</tr>
<tr>
<td>Honda4</td>
<td>Veinous</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Max Conc.</td>
</tr>
<tr>
<td>Honda5</td>
<td>Target Tissue</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Mean Conc.</td>
</tr>
<tr>
<td>Honda6</td>
<td>Target Tissue</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Max Conc.</td>
</tr>
</tbody>
</table>

*Assumption is currently ignored because analytical steady-state solutions are currently used by this function.

Steady state plasma concentration in specified units

Robert Pearce, John Wambaugh, Greg Honda, Miyuki Breen

calc_analytic_css_1comp

Examples

calc_analytic_css(chem.name='Bisphenol-A', output.units='mg/L',
                 model='3compartment', concentration='blood')

calc_analytic_css(chem.name='Bisphenol-A', tissue='liver', species='rabbit',
                 parameterize.args = list(
                     default.to.human=TRUE,
                     adjusted.Funbound.plasma=TRUE,
                     regression=TRUE,
                     minimum.Funbound.plasma=1e-4), daily.dose=2)

calc_analytic_css(chem.name="bisphenol a", model="1compartment")

calc_analytic_css(chem.cas="80-05-7", model="3compartmentss")

params <- parameterize_pbtk(chem.cas="80-05-7")

calc_analytic_css(parameters=params, model="pbtk")

calc_analytic_css_1comp

Calculate the analytic steady state concentration for the one compartment model.

Description

This function calculates the analytic steady state plasma or venous blood concentrations as a result of infusion dosing.

Usage

calc_analytic_css_1comp(
    chem.name = NULL,
    chem.cas = NULL,
    dtxsid = NULL,
    parameters = NULL,
    hourly.dose = 1/24,
    concentration = "plasma",
    suppress.messages = FALSE,
    recalc.blood2plasma = FALSE,
    tissue = NULL,
    restrictive.clearance = TRUE,
    bioactive.free.invivo = FALSE,
    ...
)

Arguments

chem.name Either the chemical name, CAS number, or the parameters must be specified.
chem.cas Either the chemical name, CAS number, or the parameters must be specified.
calc_analytic_css_3comp

This function calculates the analytic steady state plasma or venous blood concentrations as a result of infusion dosing.

Usage

```r
calc_analytic_css_3comp(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  parameters = NULL,
)```

Value

Steady state plasma concentration in mg/L units

Author(s)

Robert Pearce and John Wambaugh
calc_analytic_css_3comp

hourly.dose = 1/24,
concentration = "plasma",
suppress.messages = FALSE,
recalc.blood2plasma = FALSE,
tissue = NULL,
restrictive.clearance = TRUE,
bioactive.free.invivo = FALSE,
...)

Arguments

chem.name Either the chemical name, CAS number, or the parameters must be specified.
chem.cas Either the chemical name, CAS number, or the parameters must be specified.
dtxsid EPA’s DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
parameters Chemical parameters from parameterize_pbtk (for model = 'pbtk'), parameterize_3comp (for model = '3compartment'), parameterize_1comp(for model = '1compartment') or parameterize_steadystate (for model = '3compartmentss'), overrides chem.name and chem.cas.
hourly.dose Hourly dose rate mg/kg BW/h.
concentration Desired concentration type, 'blood' or default 'plasma'.
suppress.messages Whether or not the output message is suppressed.
recalc.blood2plasma Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters. Use this if you have altered hematocrit, Funbound.plasma, or Krbc2pu.
tissue Desired tissue concentration (defaults to whole body concentration.)
restrictive.clearance If TRUE (default), then only the fraction of chemical not bound to protein is available for metabolism in the liver. If FALSE, then all chemical in the liver is metabolized (faster metabolism due to rapid off-binding).
bioactive.free.invivo If FALSE (default), then the total concentration is treated as bioactive in vivo. If TRUE, the the unbound (free) plasma concentration is treated as bioactive in vivo. Only works with tissue = NULL in current implementation.
...

Value

Steady state plasma concentration in mg/L units

Author(s)

Robert Pearce and John Wambaugh
calc_analytic_css_3compss

Calculate the analytic steady state concentration for the three compartment steady-state model

Description

This function calculates the analytic steady state plasma or venous blood concentrations as a result of infusion dosing.

Usage

```r
calc_analytic_css_3compss(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  parameters = NULL,
  hourly.dose = 1/24,
  concentration = "plasma",
  suppress.messages = FALSE,
  recalc.blood2plasma = FALSE,
  tissue = NULL,
  restrictive.clearance = TRUE,
  bioactive.free.invivo = FALSE,
  ...)
```

Arguments

- **chem.name**: Either the chemical name, CAS number, or the parameters must be specified.
- **chem.cas**: Either the chemical name, CAS number, or the parameters must be specified.
- **dtxsid**: EPA’s ’DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
- **parameters**: Chemical parameters from parameterize_pbtk (for model = 'pbtk'), parameterize_3comp (for model = '3compartment'), parameterize_1comp(for model = '1compartment') or parameterize_steadystate (for model = '3compartmentss'), overrides chem.name and chem.cas.
- **hourly.dose**: Hourly dose rate mg/kg BW/h.
- **concentration**: Desired concentration type, ‘blood’ or default ’plasma’.
- **suppress.messages**: Whether or not the output message is suppressed.
- **recalc.blood2plasma**: Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters. Use this if you have ’altered hematocrit, Funbound.plasma, or Krbc2pu.
- **tissue**: Desired tissue concentration (defaults to whole body concentration.)
- **restrictive.clearance**: If TRUE (default), then only the fraction of chemical not bound to protein is available for metabolism in the liver. If FALSE, then all chemical in the liver is metabolized (faster metabolism due to rapid off-binding).
bioactive.free.invivo

If FALSE (default), then the total concentration is treated as bioactive in vivo.
If TRUE, the the unbound (free) plasma concentration is treated as bioactive in vivo. Only works with tissue = NULL in current implementation.

Additional parameters passed to parameterize function if parameters is NULL.

Value
Steady state plasma concentration in mg/L units

Author(s)
Robert Pearce and John Wambaugh

calc_analytic_css_pbtk

Calculate the analytic steady state plasma concentration for model pbtk.

Description
This function calculates the analytic steady state plasma or venous blood concentrations as a result of infusion dosing.

Usage

\[
calc\_analytic\_css\_pbtk( \\
chem.name = NULL, \\
chem.cas = NULL, \\
dtxsid = NULL, \\
parameters = NULL, \\
hourly.dose = 1/24, \\
concentration = "plasma", \\
suppress.messages = FALSE, \\
recalc.blood2plasma = FALSE, \\
tissue = NULL, \\
restrictive.clearance = TRUE, \\
bioactive.free.invivo = FALSE, \\
\ldots \\
) \\
\]

Arguments
chem.name Either the chemical name, CAS number, or the parameters must be specified.
chem.cas Either the chemical name, CAS number, or the parameters must be specified.
dtxsid EPA’s ’DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
parameters Chemical parameters from parameterize_pbtk (for model = ’pbtk’), parameterize_3comp (for model = ’3compartment’), parameterize_1comp(for model = ’1compartment’) or parameterize_steadystate (for model = ’3compartmentss’), overrides chem.name and chem.cas.
hourly.dose  Hourly dose rate mg/kg BW/h.
concentration Desired concentration type, 'blood', 'tissue', or default 'plasma'.
suppress.messages Whether or not the output message is suppressed.
recalc.blood2plasma Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters. Use this if you have 'altered hematocrit, Funbound.plasma, or Krbc2pu.
tissue Desired tissue concentration (defaults to whole body concentration.)
restrictive.clearance If TRUE (default), then only the fraction of chemical not bound to protein is available for metabolism in the liver. If FALSE, then all chemical in the liver is metabolized (faster metabolism due to rapid off-binding).
bioactive.free.invivo If FALSE (default), then the total concentration is treated as bioactive in vivo. If TRUE, the the unbound (free) plasma concentration is treated as bioactive in vivo. Only works with tissue = NULL in current implementation.

Value

Steady state plasma concentration in mg/L units

Author(s)

Robert Pearce and John Wambaugh

Description

This function finds the day a chemical comes within the specified range of the analytical steady state venous blood or plasma concentration(from calc_analytic_css) for the multiple compartment, three compartment, and one compartment models, the fraction of the true steady state value reached on that day, the maximum concentration, and the average concentration at the end of the simulation.

Usage

calc_css(  
  chem.name = NULL,  
  chem.cas = NULL,  
  dtxsid = NULL,  
  parameters = NULL,  
  species = "Human",  
  f = 0.01,  
  daily.dose = 1,  
  doses.per.day = 3,  
  days = 21,  
  output.units = "uM",  
)
calc_css

    suppress.messages = FALSE,
tissue = NULL,
model = "pbtk",
default.to.human = FALSE,
f.change = 1e-05,
adjusted.Funbound.plasma = TRUE,
regression = TRUE,
well.stirred.correction = TRUE,
restrictive.clearance = TRUE,
dosing = NULL,

)

Arguments

chem.name           Either the chemical name, CAS number, or parameters must be specified.
chem.cas            Either the chemical name, CAS number, or parameters must be specified.
dtxsid             EPA’s DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
parameters          Chemical parameters from parameterize_pbtk function, overrides chem.name and chem.cas.
species            Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
f                Fractional distance from the final steady state concentration that the average concentration must come within to be considered at steady state.
daily.dose       Total daily dose, mg/kg BW.
doses.per.day    Number of doses per day.
days             Initial number of days to run simulation that is multiplied on each iteration.
output.units     Units for returned concentrations, defaults to uM (specify units = "uM") but can also be mg/L.
suppress.messages Whether or not to suppress messages.
tissue            Desired tissue concentration (default value is NULL, will depend on model – see steady.state.compartment in model.info file for further details.)
model             Model used in calculation, ’pbtk’ for the multiple compartment model,’3compartment’ for the three compartment model, and ’1compartment’ for the one compartment model.
default.to.human Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).
f.change          Fractional change of daily steady state concentration reached to stop calculating.
adjusted.Funbound.plasma    Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.
regression        Whether or not to use the regressions in calculating partition coefficients.
well.stirred.correction Uses correction in calculation of hepatic clearance for well-stirred model if TRUE for model 1compartment elimination rate. This assumes clearance relative to amount unbound in whole blood instead of plasma, but converted to use with plasma concentration.
calc_elimination_rate

Calculate the elimination rate for a one compartment model

Description

This function calculates an elimination rate from the three compartment steady state model where elimination is entirely due to metabolism by the liver and glomerular filtration in the kidneys.
Usage

calc_elimination_rate(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  species = "Human",
  suppress.messages = FALSE,
  default.to.human = FALSE,
  restrictive.clearance = TRUE,
  adjusted.Funbound.plasma = TRUE,
  adjusted.Clint = TRUE,
  regression = TRUE,
  well.stirred.correction = TRUE,
  clint.pvalue.threshold = 0.05,
  minimum.Funbound.plasma = 1e-04
)

Arguments

chem.cas Either the cas number or the chemical name must be specified.
chem.name Either the chemical name or the cas number must be specified.
dtxsid EPA's DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
parameters Chemical parameters from parameterize_steadystate or 1compartment function, overrides chem.name and chem.cas.
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
suppress.messages Whether or not the output message is suppressed.
default.to.human Substitutes missing animal values with human values if true.
restrictive.clearance In calculating elimination rate, protein binding is not taken into account (set to 1) in liver clearance if FALSE.
adjusted.Funbound.plasma Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.
adjusted.Clint Uses Kilford et al. (2008) hepatocyte incubation binding adjustment for Clint when set to TRUE (Default).
regression Whether or not to use the regressions in calculating partition coefficients.
well.stirred.correction Uses correction in calculation of hepatic clearance for -stirred model if TRUE. This assumes clearance relative to amount unbound in whole blood instead of plasma, but converted to use with plasma concentration.
clint.pvalue.threshold Hepatic clearance for chemicals where the in vitro clearance assay result has a p-values greater than the threshold are set to zero.
minimum.Funbound.plasma Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).
Details

Elimination rate calculated by dividing the total clearance (using the default -stirred hepatic model) by the volume of distribution. When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data(volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value

Elimination rate
Units of 1/h.

Author(s)

John Wambaugh

References


Examples

calc_elimination_rate(chem.name="Bisphenol A")
calc_elimination_rate(chem.name="Bisphenol A",species="Rat")
calc_elimination_rate(chem.cas="80-05-7")


calc_fetal_phys Calculate maternal-fetal physiological parameters

Description

This function uses the equations from Kapraun (2019) to calculate chemical-independent physiological parameters as a function of gestational age in weeks.

Usage

calc_fetal_phys(week = 12, ...)

Arguments

  week     Gestational week
  ...     Additional arguments to parameterize_fetal_pbtk
Details

\[ BW = \text{pregnant}_{BW} + \text{BW}_{\text{cubic}}h_{eta1} \times tw + \text{BW}_{\text{cubic}}h_{eta2} \times tw^2 + \text{BW}_{\text{cubic}}h_{eta3} \times tw^3 \]

\[ W_{\text{adipose}} = W_{\text{adipose linear}}h_{eta0} + W_{\text{adipose linear}}h_{eta1} \times tw; \]

\[ W_{\text{kidney}} = 0.001 \times W_{\text{kidney}}ompertz_{heta0} \times \exp(W_{\text{kidney}}ompertz_{heta1} / W_{\text{kidney}}ompertz_{heta2} - 1) \]

\[ W_{\text{thyroid}} = 0.001 \times W_{\text{thyroid}}ompertz_{heta0} \times \exp(W_{\text{thyroid}}ompertz_{heta1} / W_{\text{thyroid}}ompertz_{heta2} - 1) \]

\[ W_{\text{liver}} = 0.001 \times W_{\text{liver}}ompertz_{heta0} \times \exp(W_{\text{liver}}ompertz_{heta1} / W_{\text{liver}}ompertz_{heta2} - 1) \]

\[ W_{\text{gut}} = 0.001 \times W_{\text{gut}}ompertz_{heta0} \times \exp(W_{\text{gut}}ompertz_{heta1} / W_{\text{gut}}ompertz_{heta2} - 1) \]

\[ W_{\text{lung}} = 0.001 \times W_{\text{lung}}ompertz_{heta0} \times \exp(W_{\text{lung}}ompertz_{heta1} / W_{\text{lung}}ompertz_{heta2} - 1) \]

\[ \text{hematocrit} = (\text{hematocrit}_{\text{quad}}h_{eta0} + \text{hematocrit}_{\text{quad}}h_{eta1} \times tw + \text{hematocrit}_{\text{quad}}h_{eta2} \times pow(tw)) \]

\[ R_{\text{blood2plasma}} = 1 - \text{hematocrit} + \text{hematocrit} \times K_{\text{rbc2pu}} \times \text{Fraction}_{\text{unbound}}plasma; \]

\[ f_{\text{hematocrit}} = (f_{\text{hematocrit}}_{\text{ubic}}h_{eta1} \times tw + f_{\text{hematocrit}}_{\text{ubic}}h_{eta2} \times pow(tw, 2) + f_{\text{hematocrit}}_{\text{ubic}}h_{eta3} \times pow(tw, 3)) \]

\[ R_{\text{blood2plasma}} = 1 - f_{\text{hematocrit}} + f_{\text{hematocrit}} \times K_{\text{rbc2pu}} \times \text{Fraction}_{\text{unbound}}plasma; \]

\[ f_{BW} = 0.001 \times f_{BW}_{ompertz}_{heta0} \times \exp(f_{BW}_{ompertz}_{heta1} / f_{BW}_{ompertz}_{heta2}) \times (1 - \exp(-f_{BW}_{ompertz}_{heta0})) \]

\[ V_{\text{placenta}} = 0.001 \times V_{\text{placenta}}_{\text{ubic}}h_{eta1} \times tw + V_{\text{placenta}}_{\text{ubic}}h_{eta2} \times pow(tw, 2) + V_{\text{placenta}}_{\text{ubic}}h_{eta3} \times pow(tw, 3) \]

\[ V_{\text{amn}} = 0.001 \times V_{\text{amn}}_{\text{logistic}}h_{eta0} / (1 + \exp(-V_{\text{amn}}_{\text{logistic}}h_{eta1} \times (tw - V_{\text{amn}}_{\text{logistic}}h_{eta2}))) \]

\[ V_{\text{plasma}} = V_{\text{plasma}}_{\text{m}}\text{logistic}_{heta0} / (1 + \exp(-V_{\text{plasma}}_{\text{m}}\text{logistic}_{heta1} \times (tw - V_{\text{plasma}}_{\text{m}}\text{logistic}_{heta2}))) \]

\[ V_{\text{rbc}} = \text{hematocrit} / (1 - \text{hematocrit}) \times V_{\text{plasma}}; \]
V_{ven} = \text{venous blood fraction} \times (V_{rbc} + V_{plasma});

V_{art} = \text{arterial blood fraction} \times (V_{rbc} + V_{plasma});

V_{adipose} = 1/adipose\text{density} \times \text{Wadipose};

V_{ffmx} = 1/ffmx\text{density} \times (BW - \text{Wadipose} - (fBW + \text{placenta density} \times V_{placenta} + \text{amn\text{density}} \times V_{amn}));

V_{allx} = V_{art} + V_{ven} + V_{thyroid} + V_{kidney} + V_{gut} + V_{liver} + V_{lung};

V_{rest} = V_{ffmx} - V_{allx};

V_{fart} = 0.001 \times \text{arterial blood fraction} \times f_{blood\text{weight ratio}} \times f_{BW};

V_{fven} = 0.001 \times \text{venous blood fraction} \times f_{blood\text{weight ratio}} \times f_{BW};

V_{fkidney} = 1/kidney\text{density} \times W_{fkidney};

V_{fthyroid} = 1/thyroid\text{density} \times W_{fthyroid};

V_{fliver} = 1/liver\text{density} \times W_{fliver};

V_{fbrain} = 1/brain\text{density} \times W_{fbrain};

V_{fgut} = 1/gut\text{density} \times W_{fgut};

V_{flung} = 1/lung\text{density} \times W_{flung};

V_{frest} = f_{BW} - (V_{fart} + V_{fven} + V_{fbrain} + V_{fkidney} + V_{fthyroid} + V_{fliver} + V_{fgut} + V_{flung});

Q_{\text{cardiac}} = 24 \times (Q_{\text{cardiac,ubic},\text{heta0}} + Q_{\text{cardiac,ubic},\text{heta1} \times tw} + Q_{\text{cardiac,ubic},\text{heta2} \times pow(tw, 2)} + Q_{\text{cardiac,ubic},\text{heta3} \times pow(tw, 3)});

Q_{\text{gut}} = 0.01 \times (Q_{\text{gut,percent,initial}} + (Q_{\text{gut,percent,terminal}} - Q_{\text{gut,percent,initial}}) / \text{term} \times tw) \times Q_{\text{cardiac}};
\[
Q_{\text{kidney}} = 24 \times (Q_{\text{kidney,ubic,theta}0} + Q_{\text{kidney,ubic,theta}1} \times tw + Q_{\text{kidney,ubic,theta}2} \times \text{pow}(tw, 2)) + Q_{\text{kidney,ubic,theta}3}
\]
\[
Q_{\text{liver}} = 0.01 \times (Q_{\text{liver,percent,initial}} + (Q_{\text{liver,percent,terminal}} - Q_{\text{liver,percent,initial}}) / \text{term} \times tw) + Q_{\text{cardiac}};
\]
\[
Q_{\text{thyroid}} = 0.01 \times (Q_{\text{thyroid,percent,initial}} + (Q_{\text{thyroid,percent,terminal}} - Q_{\text{thyroid,percent,initial}}) / \text{term} \times tw) + Q_{\text{cardiac}};
\]
\[
Q_{\text{placenta}} = 24 \times Q_{\text{placenta,linear,theta}1} \times 1000 \times V_{\text{placenta}};
\]
\[
Q_{\text{dipose}} = 0.01 \times (Q_{\text{dipose,percent,initial}} + (Q_{\text{dipose,percent,terminal}} - Q_{\text{dipose,percent,initial}}) / \text{term} \times tw) + Q_{\text{cardiac}};
\]
\[
Q_{\text{rest}} = Q_{\text{cardiac}} - (Q_{\text{gut,percent}} + Q_{\text{kidney}} + Q_{\text{liver}} + Q_{\text{thyroid}} + Q_{\text{placenta}} + Q_{\text{dipose}});
\]
\[
Q_{\text{gfr}} = 60 \times 24 \times 0.001 \times (Q_{\text{gfr,quad,theta}0} + Q_{\text{gfr,quad,theta}1} \times tw + Q_{\text{gfr,quad,theta}2} \times \text{pow}(tw, 2));
\]
\[
Q_{\text{fretl}} = 60 \times 24 \times 0.001 \times Q_{\text{fretl,logistic,theta}0} / (1 + \exp(-Q_{\text{fretl,logistic,theta}1} \times (tw - Q_{\text{fretl,logistic,theta}2})));
\]
\[
Q_{\text{fletl}} = 60 \times 24 \times 0.001 \times Q_{\text{fletl,logistic,theta0}} / (1 + \exp(-Q_{\text{fletl,logistic,theta1}} \times (tw - Q_{\text{fletl,logistic,theta2}})));
\]
\[
Q_{\text{fda}} = 60 \times 24 \times 0.001 \times Q_{\text{fda,logistic,theta1}} / (1 + \exp(-Q_{\text{fda,logistic,theta1}} \times (tw - Q_{\text{fda,logistic,theta2}}));
\]
\[
Q_{\text{ftarb}} = Q_{\text{fletl}} + Q_{\text{fda}};
\]
\[
Q_{\text{fcardiac}} = Q_{\text{ftarb}};
\]
\[
Q_{\text{flung}} = Q_{\text{fretl}} - Q_{\text{fda}};
\]
\[
Q_{\text{fplacenta}} = 60 \times 24 \times 0.001 \times Q_{\text{fplacenta,logistic,theta0}} / (1 + \exp(-Q_{\text{fplacenta,logistic,theta1}} \times (tw - Q_{\text{fplacenta,logistic,theta2}}));
\]
\[
Q_{\text{fdv}} = 60 \times 24 \times 0.001 \times Q_{\text{fdv,ompertz,theta0}} \times \exp(Q_{\text{fdv,ompertz,theta1}} / Q_{\text{fdv,ompertz,theta2}}) \times (1 - \exp(-Q_{\text{fdv,ompertz,theta2}}) + Q_{\text{fght}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{fplacenta}} = (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{fplacenta}} = (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{placenta}} = Q_{\text{fplacenta,percent}} \times Q_{\text{flung}} / Q_{\text{fplacenta,percent}} * (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{placenta}} = Q_{\text{fplacenta,percent}} \times Q_{\text{flung}} / Q_{\text{fplacenta,percent}} * (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{fplacenta}} = Q_{\text{fplacenta,percent}} \times Q_{\text{flung}} / Q_{\text{fplacenta,percent}} * (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{placenta}} = Q_{\text{fplacenta,percent}} \times Q_{\text{flung}} / Q_{\text{fplacenta,percent}} * (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{placenta}} = Q_{\text{fplacenta,percent}} \times Q_{\text{flung}} / Q_{\text{fplacenta,percent}} * (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{placenta}} = Q_{\text{fplacenta,percent}} \times Q_{\text{flung}} / Q_{\text{fplacenta,percent}} * (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{placenta}} = Q_{\text{fplacenta,percent}} \times Q_{\text{flung}} / Q_{\text{fplacenta,percent}} * (1 - Q_{\text{placenta}} / Q_{\text{ftarb}}) \times Q_{\text{ftarb}};
\]
\[
Q_{\text{bypass}} = Q_{\text{fcardiac}} - Q_{\text{flung}};
### Value

A list containing:

- **BW**  
  Maternal body weight, kg

- **Wadipose**  
  Maternal adipose fraction of total weight

- **Wfkidney**  
  Fetal kidney fraction of total weight

- **Wfthyroid**  
  Fetal thyroid fraction of total weight

- **Wfliver**  
  Fetal liver fraction of total weight

- **Wfbrain**  
  Fetal brain fraction of total weight

- **Wfgut**  
  Fetal gut fraction of total weight

- **Wflung**  
  Fetal lung fraction of total weight

- **hematocrit**  
  Maternal hematocrit fraction of blood

- **Rblood2plasma**  
  Maternal Rblood2plasma

- **fhematocrit**  
  Fetal hematocrit fraction of blood

- **Rfblood2plasma**  
  Fetal Rfblood2plasma

- **fBW**  
  Fetal body weight, kg

- **Vplacenta**  
  Volume of Vplacenta, L

- **Vamnf**  
  Volume of amniotic fluid, L

- **Vplasma**  
  Maternal volume of plasma, L

- **Vrbc**  
  Maternal volume of red blood cells, L

- **Vven**  
  Maternal volume of venous blood, L

- **Vart**  
  Maternal volume of arterial blood, L

- **Vadipose**  
  Maternal volume of adipose, L

- **Vffmx**  
  Fetal volume of Vffmx, L

- **Vallx**  
  Vallx, L

- **Vrest**  
  Maternal volume of rest of body, L

- **Vfart**  
  Fetal volume of arterial blood, L

- **Vfven**  
  Fetal volume of venous blood, L

- **Vfkidney**  
  Fetal volume of kidney, L

- **Vfthyroid**  
  Fetal volume of thyroid, L

- **Vfliver**  
  Fetal volume of liver, L

- **Vfbrain**  
  Fetal volume of brain, L

- **Vfrest**  
  Fetal volume of rest of body, L

- **Qcardiac**  
  Maternal cardiac output blood flow, L/day

- **Qgut**  
  Maternal blood flow to gut, L/day

- **Qkidney**  
  Maternal blood flow to kidney, L/day

- **Qliver**  
  Maternal blood flow to liver, L/day

- **Qthyroid**  
  Maternal blood flow to thyroid, L/day

- **Qplacenta**  
  Maternal blood flow to placenta, L/day
calc_half_life

Qadipose  Maternal blood flow to adipose, L/day
Qrest    Maternal blood flow to rest, L/day
Qgfr     Maternal glomerular filtration rate in kidney, L/day
Qfrvtl   Fetal blood flow to right ventricle, L/day
Qf1vtl   Fetal blood flow to left ventricle, L/day
Qfda     Fetal blood flow to Qfda, L/day
Qfarth   Fetal blood flow to Qfarth, L/day
Qfcardiac Fetal cardiac output blood flow, L/day
Qflung   Fetal blood flow to lung, L/day
Qfplacenta Fetal blood flow to placenta, L/day
Qfdv     Fetal blood flow to Qfdv, L/day
Qfgut    Fetal blood flow to gut, L/day
Qfkidney Fetal blood flow to kidney, L/day
Qfbrain  Fetal blood flow to brain, L/day
Qf liver  Fetal blood flow to liver, L/day
Qfthyroid Fetal blood flow to thyroid, L/day
Qfrest   Fetal blood flow to rest, L/day
Qfbypass Fetal blood flow to Qfbypass, L/day

Author(s)
John Wambaugh

References

calc_half_life

Calculates the half-life for a one compartment model.

Description
This function calculates the half life from the three compartment steady state model where elimination is entirely due to metabolism by the liver and glomerular filtration in the kidneys.

Usage
calc_half_life(chem.cas = NULL,
               chem.name = NULL,
               dtxsid = NULL,
               parameters = NULL,
               species = "Human",
               suppress.messages = FALSE,
               default.to.human = FALSE,
calc_half_life

restrictive.clearance = TRUE,
adjusted.Funbound.plasma = TRUE,
regression = TRUE,
well.stirred.correction = TRUE,
clint.pvalue.threshold = 0.05,
minimum.Funbound.plasma = 1e-04
)

Arguments

chem.cas Either the cas number or the chemical name must be specified.
chem.name Either the chemical name or the cas number must be specified.
dtxsid EPA's 'DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
parameters Chemical parameters from parameterize_steadystate or 1compartment function, overrides chem.name and chem.cas.
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
suppress.messages Whether or not the output message is suppressed.
default.to.human Substitutes missing animal values with human values if true.
restrictive.clearance In calculating elimination rate, protein binding is not taken into account (set to 1) in liver clearance if FALSE.
adjusted.Funbound.plasma Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.
regression Whether or not to use the regressions in calculating partition coefficients.
well.stirred.correction Uses correction in calculation of hepatic clearance for -stirred model if TRUE. This assumes clearance relative to amount unbound in whole blood instead of plasma, but converted to use with plasma concentration.
clint.pvalue.threshold Hepatic clearance for chemicals where the in vitro clearance assay result has a p-values greater than the threshold are set to zero.
minimum.Funbound.plasma Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

Details

Half life is calculated by dividing the natural-log of 2 by the elimination rate from the one compartment model.

Value

Half life Units of h.

Author(s)

Sarah E. Davidson
calc_hepatic_clearance

**Calculate the hepatic clearance (deprecated).**

**Description**

This function is included for backward compatibility. It calls `calc_hep_clearance` which calculates the hepatic clearance in plasma for a well-stirred model or other type if specified. Based on Ito and Houston (2004)

**Usage**

```r
calc_hepatic_clearance(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  parameters = NULL,
  species = "Human",
  default.to.human = FALSE,
  hepatic.model = "well-stirred",
  suppress.messages = FALSE,
  well.stirred.correction = TRUE,
  restrictive.clearance = TRUE,
  adjusted.Funbound.plasma = TRUE,
  ...
)
```

**Arguments**

- `chem.name` (Either the chemical name, CAS number, or the parameters must be specified.
- `chem.cas` (Either the chemical name, CAS number, or the parameters must be specified.
- `dtxsid` (EPA's DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
- `parameters` (Chemical parameters from parameterize_steadystate function, overrides chem.name and chem.cas.
- `species` (Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- `default.to.human` (Substitutes missing animal values with human values if true.
- `hepatic.model` (Model used in calculating hepatic clearance, unscaled, parallel tube, dispersion, or default well-stirred.

**See Also**

`[calc_elimination_rate()]` for the elimination rate calculation

**Examples**

```r
calc_half_life(chem.name="Bisphenol A")
calc_half_life(chem.name="Bisphenol A",species="Rat")
calc_half_life(chem.cas="80-05-7")
```
suppress.messages
Whether or not to suppress the output message.

well.stirred.correction
Uses correction in calculation of hepatic clearance for well-stirred model if TRUE for hepatic.model well-stirred. This assumes clearance relative to amount unbound in whole blood instead of plasma, but converted to use with plasma concentration.

restrictive.clearance
Protein binding not taken into account (set to 1) in liver clearance if FALSE.

adjusted.Funbound.plasma
Uses adjusted Funbound.plasma when set to TRUE.

Value
Hepatic Clearance
Units of L/h/kg BW.

Author(s)
John Wambaugh and Robert Pearce

References

Examples
calc_hep_clearance(chem.name="Ibuprofen",hepatic.model='unscaled')
calc_hep_clearance(chem.name="Ibuprofen",well.stirred.correction=FALSE)

calc_hep_bioavailability
Calculate first pass metabolism

Description
For models that don’t described first pass blood flow from the gut, need to calculate a hepatic bioavailability, that is, the fraction of chemical systemically available after metabolism during the first pass through the liver (Rowland, 1973 Equation 29, where k21 is blood flow through the liver and k23 is clearance from the liver in Figure 1).
calc_hep_clearance

Usage

calc_hep_bioavailability(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  restrictive.clearance = TRUE,
  flow.34 = TRUE
)

Arguments

chem.cas Chemical Abstract Services Registry Number (CAS-RN) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
chem.name Chemical name (spaces and capitalization ignored) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
dtxsid EPA's 'DSSTox Structure ID (https://comptox.epa.gov/dashboard) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISDs
parameters Parameters from the appropriate parameterization function for the model indicated by argument model
restrictive.clearance Protein binding not taken into account (set to 1) in liver clearance if FALSE.
flow.34 A logical constraint

Value

A data.table whose columns are the parameters of the HTTK model specified in model.

Author(s)

John Wambaugh

References


calc_hep_clearance Calculate the hepatic clearance.

Description

This function calculates the hepatic clearance in plasma for using the Houston (2004) are also available. In vitro measured hepatic clearace is corrected for the free fraction in the assay using the model of Kilford et al. (2008).
Usage

calc_hep_clearance(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  parameters = NULL,
  species = "Human",
  default.to.human = FALSE,
  hepatic.model = "well-stirred",
  suppress.messages = FALSE,
  well.stirred.correction = TRUE,
  restrictive.clearance = TRUE,
  adjusted.Funbound.plasma = TRUE,
  ...
)

Arguments

chem.name Either the chemical name, CAS number, or the parameters must be specified.
chem.cas Either the chemical name, CAS number, or the parameters must be specified.
dtxsid EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs
parameters Chemical parameters from parameterize_steadystate function, overrides chem.name and chem.cas.
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
default.to.human Substitutes missing animal values with human values if true.
hepatic.model Model used in calculating hepatic clearance, unscaled, parallel tube, dispersion, or default well-stirred.
suppress.messages Whether or not to suppress the output message.
well.stirred.correction Uses correction in calculation of hepatic clearance for well-stirred model if TRUE for hepatic.model well-stirred. This assumes clearance relative to amount unbound in whole blood instead of plasma, but converted to use with plasma concentration.
restrictive.clearance Protein binding not taken into account (set to 1) in liver clearance if FALSE.
adjusted.Funbound.plasma Uses adjusted Funbound.plasma when set to TRUE.
...

Value

Hepatic Clearance
Units of L/h/kg BW.

Author(s)

John Wambaugh and Robert Pearce
**References**


**Examples**

```
calc_hep_clearance(chem.name="Ibuprofen",hepatic.model='unscaled')
calc_hep_clearance(chem.name="Ibuprofen",well.stirred.correction=FALSE)
```

---

**calc_hep_fu**

*Calculate the free chemical in the hepatic clearance assay*

**Description**

Method from Kilford et al. (2008) for fraction of unbound chemical in the hepatocyte intrinsic clearance assay

**Usage**

```
calc_hep_fu(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  Vr = 0.005,
  pH = 7.4
)
```

**Arguments**

- **chem.cas** Chemical Abstract Services Registry Number (CAS-RN) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
- **chem.name** Chemical name (spaces and capitalization ignored) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
- **dtxsid** EPA’s DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISDs
- **parameters** Parameters from the appropriate parameterization function for the model indicated by argument model
- **Vr** Rratio of cell volume to incubation volume. Default is taken from
- **pH** pH of the incubation medium.
**calc_ionization**

**Value**

A numeric fraction between zero and one

**Author(s)**

John Wambaugh and Robert Pearce

**References**


---

**Description**

This function calculates the ionization of a compound at a given pH. The pKa’s are either entered as parameters or taken from a specific compound in the package.

**Usage**

```r
calc_ionization(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  pH = NULL,
  pKa_Donor = NULL,
  pKa_Accept = NULL
)
```

**Arguments**

- **chem.cas**: Either the chemical name or the CAS number must be specified.
- **chem.name**: Either the chemical name or the CAS number must be specified.
- **dtxsid**: EPA’s 'DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
- **parameters**: Chemical parameters from a parameterize_MODEL function, overrides chem.name and chem.cas.
- **pH**: pH where ionization is evaluated.
- **pKa_Donor**: Compound H dissociation equilibrium constant(s). Overwrites chem.name and chem.cas.
- **pKa_Accept**: Compound H association equilibrium constant(s). Overwrites chem.name and chem.cas.
Details

The arguments pKa_Donor and pKa_Accept may be single numbers, characters, or vectors. We support characters because there are many instances with multiple predicted values and all those values can be included by concatenating with commas (for example, pKa_Donor = "8.1,8.6"). Finally, pka_Donor and pKa_Accept may be vectors of characters representing different chemicals or instances of chemical parameters to allow for uncertainty analysis.

The fractions are calculated by determining the coefficients for each species and dividing the particular species by the sum of all three. The positive, negative and zwitterionic/neutral coefficients are given by:

\[
\text{fraction neutral} = \frac{1}{\text{sum}}
\]
\[
\text{fraction positive} = \frac{\sum_{i} pK_a + \cdots + pK_a - i \times pH}{\text{sum}}
\]
\[
\text{fraction negative} = \frac{\sum_{i} i \times pH - pK_a + \cdots - pK_a}{\text{sum}}
\]

where \( i \) begins at 1 and ends at the number of points above (for negative) or below (for positive) the neutral/zwitterionic range. The neutral/zwitterionic range is either the pH range between 2 pKa's where the number of acceptors above is equal to the number of donors below, everything above the pKa acceptors if there are no donors, or everything below the pKa donors if there are no acceptors. Each of the terms in the sums represent a different ionization.

Value

- **fraction_neutral**: fraction of compound neutral
- **fraction_charged**: fraction of compound charged
- **fraction_negative**: fraction of compound negative
- **fraction_positive**: fraction of compound positive
- **fraction_zwitter**: fraction of compound zwitterionic

Author(s)

Robert Pearce and John Wambaugh

References


Examples

```r
# Donor pKa's 9.78,10.39 -- Should be almost all neutral at plasma pH:
out <- calc_ionization(chem.name='bisphenola',pH=7.4)
print(out)
out[['fraction_neutral']]==max(unlist(out))

# Donor pKa's 9.78,10.39 -- Should be almost all negative (anion) at higher pH:
out <- calc_ionization(chem.name='bisphenola',pH=11)
```
calc_krbc2pu

Back-calculates the Red Blood Cell to Unbound Plasma Partition Coefficient

Description

Given an observed ratio of chemical concentration in blood to plasma, this function calculates a Red Blood Cell to unbound plasma (Krbc2pu) partition coefficient that would be consistent with that observation.

Usage

calc_krbc2pu(
  Rb2p,
  Funbound.plasma,
  hematocrit = NULL,
  default.to.human = FALSE,
  species = "Human",
  suppress.messages = TRUE
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rb2p</td>
<td>The chemical blood:plasma concentration ratio</td>
</tr>
<tr>
<td>Funbound.plasma</td>
<td>The free fraction of chemical in the presence of plasma protein Rblood2plasma.</td>
</tr>
<tr>
<td>hematocrit</td>
<td>Overwrites default hematocrit value in calculating Rblood2plasma.</td>
</tr>
<tr>
<td>default.to.human</td>
<td>Substitutes missing animal values with human values if true.</td>
</tr>
</tbody>
</table>
calc_maternal_bw

Description
This function initializes the parameters needed in the functions solve_fetal_pbtk by calling solve_pbtk and adding additional parameters.

Usage
calc_maternal_bw(week = 12)

Arguments
week
Gestational week

Details
BW <- params$pre_pregnant_BW + params$BW_cubic_theta1 * tw + params$BW_cubic_theta2 * tw^2 + params$BW_cubic_theta3 * tw^3

Value
BW
Maternal Body Weight, kg.

Author(s)
John Wambaugh

References
calc_mc_css

Distribution of chemical steady state concentration with uncertainty and variability

Description

For a given chemical and fixed dose rate this function determines a distribution of steady-state concentrations reflecting measurement uncertainty and population variability. Uncertainty and variability are simulated via the Monte Carlo method—many sets of model parameters are drawn according to probability distributions described in Ring et al. (2017) (doi:10.1016/j.envint.2017.06.004) for human variability and Wambaugh et al. (2019) (doi:10.1093/toxsci/kfz205) for measurement uncertainty. Monte Carlo samples are generated by the function create_mc_samples. To allow rapid application of the Monte Carlo method we make use of analytical solutions for the steady-state concentration for a particular model via a given route (when available) as opposed to solving the model numerically (that is, using differential equations). For each sample of the Monte Carlo method (as specified by argument samples) the parameters for the analytical solution are varied. An ensemble of steady-state predictions are produced, though by default only the quantiles specified by argument which.quantile are provided. If the full set of predicted values are desired use set the argument return.samples to TRUE.

Usage

```
calc_mc_css(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  samples = 1000,
  which.quantile = 0.95,
  species = "Human",
  suppress.messages = FALSE,
  model = "3compartmentss",
  httkpop = TRUE,
  invitrouv = TRUE,
  calcrb2p = TRUE,
  censored.params = list(),
  vary.params = list(),
  return.samples = FALSE,
  tissue = NULL,
  concentration = "plasma",
  output.units = "mg/L",
  invitro.mc.arg.list = list(adjusted.Funbound.plasma = TRUE, poormetab = TRUE,
    fup.censored.dist = FALSE, fup.lod = 0.01, fup.meas.cv = 0.4, clint.meas.cv = 0.3,
    fup.pop.cv = 0.3, clint.pop.cv = 0.3),
  httkpop.generate.arg.list = list(method = "direct resampling", gendernum = NULL,
    agelim_years = NULL, agelim_months = NULL, weight_category = c("Underweight",
      "Normal", "Overweight", "Obese"), gfr_category = c("Normal", "Kidney Disease",
      "Kidney Failure"), reths = c("Mexican American", "Other Hispanic",
      "Non-Hispanic White", "Non-Hispanic Black", "Other")),
  convert.httkpop.arg.list = list(),
  parameterize.arg.list = list(default.to.human = FALSE, clint.pvalue.threshold = 0.05,
```
Argument

- `chem.cas` Chemical Abstract Services Registry Number (CAS-RN) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
- `chem.name` Chemical name (spaces and capitalization ignored) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
- `dtxsid` EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISDs
- `parameters` Parameters from the appropriate parameterization function for the model indicated by argument model
- `samples` Number of samples generated in calculating quantiles.
- `which.quantile` Which quantile from Monte Carlo simulation is requested. Can be a vector.
- `species` Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human"). Species must be set to "Human" to run httkpop model.
- `suppress.messages` Whether or not to suppress output message.
- `model` Model used in calculation. 'gas_pbtk' for the gas pbtk model, 'pbtk' for the multiple compartment model, '3compartment' for the three compartment model, '3compartmentss' for the three compartment steady state model, and '1compartment' for one compartment model. This only applies when httkpop=TRUE and species="Human", otherwise '3compartmentss' is used.
- `httkpop` Whether or not to use population generator and sampler from httkpop. This is overwrites censored.params and vary.params and is only for human physiology. Species must also be set to 'Human'.
- `invitrouv` Logical to indicate whether to include in vitro parameters in uncertainty and variability analysis
- `calcrb2p` Logical determining whether or not to recalculate the chemical ratio of blood to plasma
- `censored.params` The parameters listed in censored.params are sampled from a normal distribution that is censored for values less than the limit of detection (specified separately for each parameter). This argument should be a list of sublists. Each sublist is named for a parameter in "parameters" and contains two elements: "CV" (coefficient of variation) and "LOD" (limit of detection, below which parameter values are censored). New values are sampled with mean equal to the value in "parameters" and standard deviation equal to the mean times the CV. Censored values are sampled on a uniform distribution between 0 and the limit of detection. Not used with httkpop model.
- `vary.params` The parameters listed in vary.params are sampled from a normal distribution that is truncated at zero. This argument should be a list of coefficients of variation (CV) for the normal distribution. Each entry in the list is named for a parameter
calc_mc_css

in "parameters". New values are sampled with mean equal to the value in "parameters" and standard deviation equal to the mean times the CV. Not used with httkpop model.

return.samples Whether or not to return the vector containing the samples from the simulation instead of the selected quantile.

tissue Desired steady state tissue concentration. Default is of NULL typically gives whole body plasma concentration.

concentration Desired concentration type: 'blood', 'tissue', or default 'plasma'. In the case that the concentration is for plasma, selecting "blood" will use the blood:plasma ratio to estimate blood concentration. In the case that the argument 'tissue' specifies a particular tissue of the body, concentration defaults to 'tissue' – that is, the concentration in the If concentration is set to 'blood' or 'plasma' and 'tissue' specifies a specific tissue then the value returned is for the plasma or blood in that specific tissue.

output.units Plasma concentration units, either uM or default mg/L.

invitro.mc.arg.list
List of additional parameters passed to invitro_mc

httkpop.generate.arg.list
Additional parameters passed to httkpop_generate.

convert.httkpop.arg.list
Additional parameters passed to the convert_httkpop_* function for the model.

parameterize.arg.list
Additional parameters passed to the parameterize_* function for the model.

calc.analytic.css.arg.list
Additional parameters passed to calc_analytic_css.

parameterize.args
A list of arguments to be passed to the model parameterization function (that is, parameterize_MODEL) corresponding to argument "model". (Defaults to NULL.)

Details

The chemical-specific steady-state concentration for a dose rate of 1 mg/kg body weight/day can be used for in in vitro-in vivo extrapolation (IVIVE) of a bioactive in vitro concentration by dividing the in vitro concentration by the steady-state concentration to predict a dose rate (mg/kg/day) that would produce that concentration in plasma. Using quantiles of the distribution (such as the upper 95th percentile) allow incorporation of uncertainty and variability into IVIVE.

Reverse Dosimetry Toxicodynamic IVIVE
Figure from Breen et al. (2021) (doi:10.1080/17425255.2021.1935867) shows reverse dosimetry toxicodynamic IVIVE. Equivalent external dose is determined by solving the TK model in reverse by deriving the external dose (that is, TK model input) that produces a specified internal concentration (that is, TK model output). Reverse dosimetry and IVIVE using HTTK relies on the linearity of the models. We calculate a scaling factor to relate \( in vitro \) concentrations (\( \mu M \)) to administered equivalent doses (AED). The scaling factor is the inverse of the steady state plasma concentration (C_{ss}) predicted for a 1 mg/kg/day exposure dose rate. We use Monte Carlo to simulate variability and propagate uncertainty; for example, to calculate an upper 95th percentile C_{ss,95} for individuals who get higher plasma concentrations from the same exposure.

The Monte Carlo methods used here were recently updated and described by Breen et al. (submitted).

httk-pop is used only for humans. For non-human species biological variability is simulated by drawing parameters from uncorrelated log-normal distributions.

Chemical-specific httk data are available primarily for human and for a few hundred chemicals in rat. All in silico predictions are for human. Thus, when species is specified as rabbit, dog, or mouse, the user can choose to set the argument \texttt{default.to.human} to \texttt{TRUE} so that this function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

If the argument \texttt{tissue} is used, the steady-state concentration in that tissue, if available, is provided. If that tissue is included in the model used (specified by argument \texttt{model}) then the actual tissue concentration is provided. Otherwise, the tissue-specific partition coefficient is used to estimate the concentration from the plasma.

The six sets of plausible IVIVE assumptions identified by Honda et al. (2019) (doi:10.1371/journal.pone.0217564) are:

<table>
<thead>
<tr>
<th>Honda1</th>
<th>Veinous (Plasma)</th>
<th>Restrictive</th>
<th>Free</th>
<th>Mean Conc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honda2</td>
<td>Veinous</td>
<td>Restrictive</td>
<td>Free</td>
<td>Max Conc.</td>
</tr>
</tbody>
</table>

\[
\text{AED}_{C_{ss,95}} = \frac{[X]}{C_{ss,95}}
\]
<table>
<thead>
<tr>
<th>Honda3</th>
<th>Veinous</th>
<th>Non-restrictive</th>
<th>Total</th>
<th>Mean Conc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Honda4</td>
<td>Veinous</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Max Conc.</td>
</tr>
<tr>
<td>Honda5</td>
<td>Target Tissue</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Mean Conc.</td>
</tr>
<tr>
<td>Honda6</td>
<td>Target Tissue</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Max Conc.</td>
</tr>
</tbody>
</table>

*Assumption is currently ignored because analytical steady-state solutions are currently used by this function.

**Value**

Quantiles (specified by which.quantile) of the distribution of plasma steady-state concentration (Css) from the Monte Carlo simulation

**Author(s)**

Caroline Ring, Robert Pearce, John Wambaugh, Miyuki Breen

**References**


**Examples**

# Basic in vitro - in vivo extrapolation with httk, convert 3 uM in vitro
# concentration of chemical with CAS 2451-62-9 to mg/kg/day:
set.seed(1234)
3/calc_mc_css(chem.cas="2451-62-9",samples=10,output.units="uM")

# The significant digits should give the same answer as:
set.seed(1234)
calc_mc_oral_equiv(chem.cas="2451-62-9",conc=3,samples=10)

set.seed(1234)
calc_mc_css(chem.name='Bisphenol A',output.units='uM',
            samples=100,return.samples=TRUE)

set.seed(1234)
calc_mc_css(chem.name='Bisphenol A',output.units='uM',httkpop.generate.arg.list=list(method='vi'))

# The following example should result in an error since we do not
# estimate tissue partitioning with '3compartmentss'.
set.seed(1234)
try(calc_mc_css(chem.name='2,4-d',which.quantile=.9,httkpop=FALSE,tissue='heart'))

set.seed(1234)
calc_mc_css(chem.name='2,4-d',model='pbtk',which.quantile=.9,httkpop=FALSE,tissue='heart')
calc_mc_oral_equiv

Calculate Monte Carlo Oral Equivalent Dose

```
set.seed(1234)
calc.mc.css(chem.cas = "80-05-7", which.quantile = 0.5,
output.units = "uM", samples = 2000,
httkpop.generate.arg.list=list(method='vi', gendernum=NULL,
agelim_years=NULL, agelim_months=NULL, weight_category =
c("Underweight", "Normal", "Overweight", "Obese")))

params <- parameterize_pbtk(chem.cas="80-05-7")
set.seed(1234)
calc.mc.css(parameters=params,model="pbtk")

set.seed(1234)
# Standard HTTK Monte Carlo:
NSAMP = 500
calc.mc.css(chem.cas="90-43-7",model="pbtk",samples=NSAMP)
set.seed(1234)
calc.mc.css(chem.cas="90-43-7",
model="pbtk",
samples=NSAMP,
invitro.mc.arg.list = list(
  adjusted.Funbound.plasma = TRUE,
  poormetab = TRUE,
  fup.censored.dist = FALSE,
  fup.lod = 0.01,
  fup.meas.cv = 0.0,
  clint.meas.cv = 0.0,
  clint.pop.cv = 0.3))
set.seed(1234)
# HTTK Monte Carlo with no HTTK-Pop physiological variability):
calc.mc.css(chem.cas="90-43-7",model="pbtk",samples=NSAMP,httkpop=FALSE)
set.seed(1234)
# HTTK Monte Carlo with no in vitro uncertainty and variability):
calc.mc.css(chem.cas="90-43-7",model="pbtk",samples=NSAMP,invitrouv=FALSE)
set.seed(1234)
# HTTK Monte Carlo with no HTTK-Pop and no in vitro uncertainty and variability):
calc.mc.css(chem.cas="90-43-7",model="pbtk",samples=NSAMP,httkpop=FALSE,invitrouv=FALSE)
# Should be the same as the mean result:
calc.analytic.css(chem.cas="90-43-7",model="pbtk",output.units="mg/L")
set.seed(1234)
# HTTK Monte Carlo using basic Monte Carlo sampler:
calc.mc.css(chem.cas="90-43-7",
model="pbtk",
samples=NSAMP,
httkpop=FALSE,
invitrouv=FALSE,
vary.params=list(Pow=0.3))
```
**Description**

This function converts a chemical plasma concentration to an oral administered equivalent dose (AED) using a concentration obtained from `calc_mc_css`. This function uses reverse dosimetry-based *in vitro-in vivo* extrapolation (IVIVE) for high throughput risk screening. The user can input the chemical and *in vitro* bioactive concentration, select the TK model, and then automatically predict the *in vivo* AED which would produce a body concentration equal to the *in vitro* bioactive concentration. This function relies on the Monte Carlo method (via function `create_mc_samples`) to simulate both uncertainty and variability so that the result is a distribution of equivalent doses, from which we provide specific quantiles (specified by `which.quantile`), though the full set of predictions can be obtained by setting `return.samples` to TRUE.

**Usage**

```r
calc_mc_oral_equiv(
  conc, 
  chem.name = NULL, 
  chem.cas = NULL, 
  dtxsid = NULL, 
  which.quantile = 0.95, 
  species = "Human", 
  input.units = "uM", 
  output.units = "mgpkgpday", 
  suppress.messages = FALSE, 
  return.samples = FALSE, 
  restrictive.clearance = TRUE, 
  bioactive.free.invivo = FALSE, 
  tissue = NULL, 
  concentration = "plasma", 
  IVIVE = NULL, 
  model = "3compartmentss", 
  ...)
```

**Arguments**

- **conc**: Bioactive in vitro concentration in units of uM.
- **chem.name**: Either the chemical name or the CAS number must be specified.
- **chem.cas**: Either the CAS number or the chemical name must be specified.
- **dtxsid**: EPA's 'DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs
- **which.quantile**: Which quantile from Monte Carlo steady-state simulation (`calc_mc_css`) is requested. Can be a vector. Note that 95th concentration quantile is the same population as the 5th dose quantile.
- **species**: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- **input.units**: Units of given concentration, default of uM but can also be mg/L.
- **output.units**: Units of dose, default of ’mgpkgpday’ for mg/kg BW/ day or ’umolpkgpday’ for umol/ kg BW/ day.
- **suppress.messages**: Suppress text messages.
return.samples  Whether or not to return the vector containing the samples from the simulation instead of the selected quantile.

restrictive.clearance
Protein binding not taken into account (set to 1) in liver clearance if FALSE.

bioactive.free.invivo
If FALSE (default), then the total concentration is treated as bioactive in vivo.
If TRUE, the the unbound (free) plasma concentration is treated as bioactive in vivo. Only works with tissue = NULL in current implementation.

tissue
Desired steady state tissue concentration. Default is of NULL typically gives whole body plasma concentration.

concentration
Desired concentration type: 'blood','tissue', or default 'plasma'. In the case that the concentration is for plasma, selecting ”blood” will use the blood/plasma ratio to estimate blood concentration. In the case that the argument 'tissue' specifies a particular tissue of the body, concentration defaults to ‘tissue’ – that is, the concentration in the If coecentration is set to 'blood' or 'plasma' and 'tissue' specifies a specific tissue then the value returned is for the plasma or blood in that specific tissue.

IVIVE
Honda et al. (2019) identified six plausible sets of assumptions for in vitro-in vivo extrapolation (IVIVE) assumptions. Argument may be set to ”Honda1” through ”Honda6”. If used, this function overwrites the tissue, restrictive.clearance, and plasma.binding arguments. See Details below for more information.

model
Model used in calculation,”gas_pbtk” for the gas pbtk model, ’pbtk’ for the multiple compartment model, ’3compartment’ for the three compartment model, ’3compartmentss’ for the three compartment steady state model, and ’1compartment’ for one compartment model. This only applies when httkpop=TRUE and species=”Human”, otherwise ’3compartmentss’ is used.

Additional parameters passed to calc_mc_css for httkpop and variance of parameters.

Details
The chemical-specific steady-state concentration for a dose rate of 1 mg/kg body weight/day can be used for in IVIVE of a bioactive in vitro concentration by dividing the in vitro concentration by the steady-state concentration to predict a dose rate (mg/kg/day) that would produce that concentration in plasma. Using quantiles of the distribution (such as the upper 95th percentile) allow incorporation of uncertainty and variability into IVIVE.

This approach relies on the linearity of the models to calculate a scaling factor to relate in vitro concentrations (uM) with AED. The scaling factor is the inverse of the steady-state plasma concentration (Css) predicted for a 1 mg/kg/day exposure dose rate where in vitro concentration [X] and Css must be in the same units. Note that it is typical for in vitro concentrations to be reported in units of uM and Css in units of mg/L, in which case one must be converted to the other.

Reverse Dosimetry Toxicodynamic IVIVE
Figure from Breen et al. (2021) (doi:10.1080/17425255.2021.1935867) shows reverse dosimetry toxicodynamic IVIVE. Equivalent external dose is determined by solving the TK model in reverse by deriving the external dose (that is, TK model input) that produces a specified internal concentration (that is, TK model output). Reverse dosimetry and IVIVE using HTTK relies on the linearity of the models. We calculate a scaling factor to relate in vitro concentrations (uM) to AEDs. The scaling factor is the inverse of theCss predicted for a 1 mg/kg/day exposure dose rate. We use Monte Carlo to simulate variability and propagate uncertainty; for example, to calculate an upper 95th percentileCss,95 for individuals who get higher plasma concentrations from the same exposure.

The Monte Carlo methods used here were recently updated and described by Breen et al. (submitted).

All arguments after httkpop only apply if httkpop is set to TRUE and species to "Human".

When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Tissue concentrations are calculated for the pbtk model with oral infusion dosing. All tissues other than gut, liver, and lung are the product of the steady state plasma concentration and the tissue to plasma partition coefficient.

The six sets of plausible IVIVE assumptions identified by Honda et al. (2019) (doi:10.1371/journal.pone.0217564) are:

| Honda1 | Veinous (Plasma) | Restrictive | Free | Mean Conc. |
| Honda2 | Veinous          | Restrictive | Free | Max Conc.  |
| Honda3 | Veinous          | Non-restrictive | Total | Mean Conc. |
| Honda4 | Veinous          | Non-restrictive | Total | Max Conc.  |
| Honda5 | Target Tissue    | Non-restrictive | Total | Mean Conc. |
| Honda6 | Target Tissue    | Non-restrictive | Total | Max Conc.  |
calc_mc_oral_equiv

*Assumption is currently ignored because analytical steady-state solutions are currently used by this function.

**Value**

Equivalent dose in specified units, default of mg/kg BW/day.

**Author(s)**

John Wambaugh

**References**


**Examples**

# Basic in vitro - in vivo extrapolation with httk, convert 0.5 uM in vitro concentration of chemical Surinabant to mg/kg/day:
set.seed(1234)
0.5/calc_mc_css(chem.name="Surinabant",samples=10,output.units="uM")
# The significant digits should give the same answer as:
set.seed(1234)
calc_mc_oral_equiv(chem.name="Surinabant",conc=0.5,samples=10)
# Note that we use set.seed to get the same sequence of random numbers for the two different function calls (calc_mc_css and calc_mc_oral_equiv)

# The following example should result in an error since we do not estimate tissue partitioning with '3compartmentss'.
set.seed(1234)
try(calc_mc_oral_equiv(0.1,chem.cas="34256-82-1",which.quantile=c(0.05,0.5,0.95),tissue='brain'))

set.seed(1234)
calc_mc_oral_equiv(0.1,chem.cas="34256-82-1",model='pbtk',which.quantile=c(0.05,0.5,0.95),tissue='brain')
**calc_mc_tk**

**Conduct multiple TK simulations using Monte Carlo**

**Description**

This function finds the analytical steady state plasma concentration (from `calc_analytic_css`) using a Monte Carlo simulation (`monte_carlo`).

**Usage**

```r
calc_MC_tk(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  samples = 1000,
  which.quantile = 0.95,
  species = "Human",
  suppress.messages = FALSE,
  model = "pbtk",
  httkpop = TRUE,
  invitrouv = TRUE,
  calcrb2p = TRUE,
  censored.params = list(),
  vary.params = list(),
  return.samples = FALSE,
  tissue = NULL,
  output.units = "mg/L",
  solvemodel.arg.list = list(times = c(0, 0.25, 0.5, 0.75, 1, 1.5, 2, 2.5, 3, 4, 5)),
  invitro.mc.arg.list = list(adjusted.Funbound.plasma = TRUE, poormetab = TRUE,
                     fup.censored.dist = FALSE, fup.lod = 0.01, fup.meas.cv = 0.4, clint.meas.cv = 0.3,
                     fup.pop.cv = 0.3, clint.pop.cv = 0.3),
  httkpop.generate.arg.list = list(method = "direct resampling", gendernum = NULL,
                      agelim_years = NULL, agelim_months = NULL, weight_category = c("Underweight",
                      "Normal", "Overweight", "Obese"), gfr_category = c("Normal", "Kidney Disease",
                      "Kidney Failure"), reths = c("Mexican American", "Other Hispanic",
                      "Non-Hispanic White", "Non-Hispanic Black", "Other")),
  convert.httkpop.arg.list = list(),
  parameterize.arg.list = list(default.to.human = FALSE, clint.pvalue.threshold = 0.05,
                   restrictive.clearance = TRUE, regression = TRUE),
  return.all.sims = FALSE
)
```

**Arguments**

- `chem.cas` Either the CAS number, parameters, or the chemical name must be specified.
- `chem.name` Either the chemical parameters, name, or the CAS number must be specified.
- `dtxsid` EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs.
- `parameters` Parameters from parameterize_steady whole. Not used with httkpop model.
samples Number of samples generated in calculating quantiles.
which.quantile Which quantile from Monte Carlo simulation is requested. Can be a vector.
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human"). Species must be set to "Human" to run httkpop model.
suppress.messages Whether or not to suppress output message.
model Model used in calculation: 'pbtk' for the multiple compartment model, '3compartment' for the three compartment model, '3compartmentss' for the three compartment steady state model, and '1compartment' for one compartment model. This only applies when httkpop=TRUE and species="Human", otherwise '3compartmentss' is used.
httkpop Whether or not to use population generator and sampler from httkpop. This is overwrites censored.params and vary.params and is only for human physiology. Species must also be set to 'Human'.
invitrouv Logical to indicate whether to include in vitro parameters in uncertainty and variability analysis
calcrb2p Logical determining whether or not to recalculate the chemical ratio of blood to plasma
censored.params The parameters listed in censored.params are sampled from a normal distribution that is censored for values less than the limit of detection (specified separately for each parameter). This argument should be a list of sub-lists. Each sublist is named for a parameter in "parameters" and contains two elements: “CV” (coefficient of variation) and “LOD” (limit of detection, below which parameter values are censored. New values are sampled with mean equal to the value in "parameters" and standard deviation equal to the mean times the CV. Censored values are sampled on a uniform distribution between 0 and the limit of detection. Not used with httkpop model.
vary.params The parameters listed in vary.params are sampled from a normal distribution that is truncated at zero. This argument should be a list of coefficients of variation (CV) for the normal distribution. Each entry in the list is named for a parameter in "parameters". New values are sampled with mean equal to the value in "parameters" and standard deviation equal to the mean times the CV. Not used with httkpop model.
return.samples Whether or not to return the vector containing the samples from the simulation instead of the selected quantile.
tissue Desired steady state tissue concentration.
output.units Plasma concentration units, either uM or default mg/L.
solvemodel.arg.list Additional arguments ultimately passed to solve_model
invitro.mc.arg.list List of additional parameters passed to invitro_mc
httkpop.generate.arg.list Additional parameters passed to httkpop_generate.
convert.httkpop.arg.list Additional parameters passed to the convert_httkpop_* function for the model.
parameterize.arg.list Additional parameters passed to the parameterize_* function for the model.
return.all.sims

   Logical indicating whether to return the results of all simulations, in addition to
   the default toxicokinetic statistics

Details

The Monte Carlo methods used here were recently updated and described by Breen et al. (submitted).

All arguments after httkpop only apply if httkpop is set to TRUE and species to "Human".

When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological
data(volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic
hepatic clearance.

Tissue concentrations are calculated for the pbtk model with oral infusion dosing. All tissues other
than gut, liver, and lung are the product of the steady state plasma concentration and the tissue to
plasma partition coefficient.

The six sets of plausible in vitro-in vivo extrapolation (IVIVE) assumptions identified by Honda et
al. (2019) (doi:10.1371/journal.pone.0217564) are:

<table>
<thead>
<tr>
<th>Honda1</th>
<th>in vivo Conc.</th>
<th>Metabolic Clearance</th>
<th>Bioactive Chemical Conc.</th>
<th>TK Statistic Used*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Veinous</td>
<td>Restrictive</td>
<td>Free</td>
<td>Mean Conc.</td>
<td></td>
</tr>
<tr>
<td>Honda2</td>
<td>Veinous</td>
<td>Restrictive</td>
<td>Free</td>
<td>Max Conc.</td>
</tr>
<tr>
<td>Honda3</td>
<td>Veinous</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Mean Conc.</td>
</tr>
<tr>
<td>Honda4</td>
<td>Veinous</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Max Conc.</td>
</tr>
<tr>
<td>Honda5</td>
<td>Target Tissue</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Mean Conc.</td>
</tr>
<tr>
<td>Honda6</td>
<td>Target Tissue</td>
<td>Non-restrictive</td>
<td>Total</td>
<td>Max Conc.</td>
</tr>
</tbody>
</table>

* Assumption is currently ignored because analytical steady-state solutions are currently used by
this function.

Value

If return.all.sims == FALSE (default) a list with:

   means   The mean concentration for each model compartment as a function of time
            across the Monte Carlo simulation

   sds     The standard deviation for each model compartment as a function of time across
            the Monte Carlo simulation

If return.all.sums == TRUE then a list is returned with:

   stats   The list of means and sds from return.all.sums=FALSE

   sims    The concentration vs. time results for each compartment for every (samples) set
            of parameters in the Monte Carlo simulation

Author(s)

John Wambaugh
calc_rblood2plasma

Calculate the constant ratio of the blood concentration to the plasma concentration.

Description

This function calculates the constant ratio of the blood concentration to the plasma concentration.

Usage

calc_rblood2plasma(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  hematocrit = NULL,
  Krbc2pu = NULL,
  Funbound.plasma = NULL,
  default.to.human = FALSE,
  species = "Human",
  adjusted.Funbound.plasma = TRUE,
  suppress.messages = TRUE
)

Arguments

c.chem.cas 
  Either the CAS number or the chemical name must be specified.

c.chem.name 
  Either the chemical name or the CAS number must be specified.
**calc_rblood2plasma**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dtxsid</td>
<td>EPA’s DSSTox Structure ID (<a href="https://comptox.epa.gov/dashboard">https://comptox.epa.gov/dashboard</a>) the chemical must be identified by either CAS, name, or DTXSIDs</td>
</tr>
<tr>
<td>parameters</td>
<td>Parameters from <a href="#">parameterize_schmitt</a></td>
</tr>
<tr>
<td>hematocrit</td>
<td>Overwrites default hematocrit value in calculating Rblood2plasma.</td>
</tr>
<tr>
<td>Krbc2pu</td>
<td>The red blood cell to unbound plasma chemical partition coefficient, typically from <a href="#">predict_partitioning_schmitt</a></td>
</tr>
<tr>
<td>Funbound.plasma</td>
<td>The fraction of chemical unbound (free) in the presence of plasma protein</td>
</tr>
<tr>
<td>default.to.human</td>
<td>Substitutes missing animal values with human values if true.</td>
</tr>
<tr>
<td>species</td>
<td>Species desired (either &quot;Rat&quot;, &quot;Rabbit&quot;, &quot;Dog&quot;, &quot;Mouse&quot;, or default &quot;Human&quot;).</td>
</tr>
<tr>
<td>adjusted.Funbound.plasma</td>
<td>Whether or not to use Funbound.plasma adjustment.</td>
</tr>
<tr>
<td>suppress.messages</td>
<td>Determine whether to display certain usage feedback.</td>
</tr>
</tbody>
</table>

**Details**

The red blood cell (RBC) partition coefficient as predicted by the Schmitt (2008) method is used in the calculation. The value is calculated with the equation: 1 - hematocrit + hematocrit * Krbc2pu * Funbound.plasma, summing the red blood cell to plasma and plasma:plasma (equal to 1) partition coefficients multiplied by their respective fractional volumes. When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (hematocrit and temperature), but substitutes human fraction unbound and tissue volumes.

**Value**

The blood to plasma chemical concentration ratio

**Author(s)**

John Wambaugh and Robert Pearce

**References**

Schmitt W. "General approach for the calculation of tissue to plasma partition coefficients." Toxicology In Vitro, 22, 457-467 (2008).


**Examples**

```r
calc_rblood2plasma(chem.name="Bisphenol A")
calc_rblood2plasma(chem.name="Bisphenol A", species="Rat")
```
calc_stats

Calculate toxicokinetic summary statistics (deprecated).

**Description**

This function is included for backward compatibility. It calls `calc_tkstats` which calculates the area under the curve, the mean, and the peak values for the venous blood or plasma concentration of a specified chemical or all chemicals if none is specified for the multiple compartment model with a given number of days, dose, and number of doses per day.

**Usage**

```r
calc_stats(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  parameters = NULL,
  route = "oral",
  stats = c("AUC", "peak", "mean"),
  species = "Human",
  days = 28,
  daily.dose = 1,
  dose = NULL,
  doses.per.day = 1,
  output.units = "uM",
  concentration = "plasma",
  tissue = "plasma",
  model = "pbtk",
  default.to.human = FALSE,
  adjusted.Funbound.plasma = TRUE,
  regression = TRUE,
  restrictive.clearance = TRUE,
  suppress.messages = FALSE,
  ...
)
```

**Arguments**

- `chem.name`: Name of desired chemical.
- `chem.cas`: CAS number of desired chemical.
- `dtxsid`: EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs.
- `parameters`: Chemical parameters from `parameterize_pbtk` function, overrides `chem.name` and `chem.cas`.
- `route`: String specification of route of exposure for simulation: "oral", "iv", "inhalation", ...
- `stats`: Desired values (either "AUC", "mean", "peak", or a vector containing any combination).
- `species`: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
days Length of the simulation.
daily.dose Total daily dose, mg/kg BW.
dose Amount of a single dose at time zero, mg/kg BW.
doses.per.day Number of doses per day.
output.units Desired units (either "mg/L", "mg", "umol", or default "uM").
concentration Desired concentration type, 'blood' or default 'plasma'.
tissue Desired steady state tissue concentration.
model Model used in calculation, 'pbtk' for the multiple compartment model, '3compartment' for the three compartment model, '3compartmentss' for the three compartment steady state model, and '1compartment' for one compartment model.
default.to.human Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).
adjusted.Funbound.plasma Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.
regression Whether or not to use the regressions in calculating partition coefficients.
restrictive.clearance Protein binding not taken into account (set to 1) in liver clearance if FALSE.
suppress.messages Whether to suppress output message.
... Arguments passed to solve function.

Details

Default value of 0 for doses.per.day solves for a single dose.

When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value

AUC Area under the plasma concentration curve.
mean.conc The area under the curve divided by the number of days.
peak.conc The highest concentration.

Author(s)

Robert Pearce and John Wambaugh
**calc_tkstats**

*Calculate toxicokinetic summary statistics.*

**Description**

This function calculates the area under the curve, the mean, and the peak values for the venous blood or plasma concentration of a specified chemical or all chemicals if none is specified for the multiple compartment model with a given number of days, dose, and number of doses per day.

**Usage**

```r
calc_tkstats(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  parameters = NULL,
  route = "oral",
  stats = c("AUC", "peak", "mean"),
  species = "Human",
  days = 28,
  daily.dose = 1,
  dose = NULL,
  doses.per.day = 1,
  output.units = "uM",
  concentration = "plasma",
  tissue = "plasma",
  model = "pbtk",
  default.to.human = FALSE,
  adjusted.Fbound.plasma = TRUE,
  regression = TRUE,
  restrictive.clearance = TRUE,
  suppress.messages = FALSE,
  ...
)
```

**Arguments**

- `chem.name`: Name of desired chemical.
- `chem.cas`: CAS number of desired chemical.
- `dtxsid`: EPA’s DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs.
- `parameters`: Chemical parameters from parameterize_pbtk function, overrides chem.name and chem.cas.
- `route`: String specification of route of exposure for simulation: "oral", "iv", "inhalation", ...
- `stats`: Desired values (either 'AUC', 'mean', 'peak', or a vector containing any combination).
- `species`: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- `days`: Length of the simulation.
calc_tkstats

daily.dose  Total daily dose, mg/kg BW.
dose        Amount of a single dose at time zero, mg/kg BW.
doses.per.day Number of doses per day.
output.units Desired units (either "mg/L", "mg", "umol", or default "uM").
concentration Desired concentration type, 'blood' or default 'plasma'.
tissue      Desired steady state tissue concentration.
model       Model used in calculation, 'pbtk' for the multiple compartment model,'3compartment' for the three compartment model, '3compartmentss' for the three compartment steady state model, and '1compartment' for one compartment model.
default.to.human
adjusted.Funbound.plasma
regression
restrictive.clearance
suppress.messages

Details

Default value of 0 for doses.per.day solves for a single dose.

When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data(volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value

AUC          Area under the plasma concentration curve.
mean.conc    The area under the curve divided by the number of days.
peak.conc    The highest concentration.

Author(s)

Robert Pearce and John Wambaugh

Examples

calc_tkstats(chem.name="Bisphenol-A", days=100, stats='mean', model='3compartment')

calc_tkstats(chem.name="Bisphenol-A", days=100, stats=c('peak','mean'), species='Rat')

triclosan.stats <- calc_tkstats(days=10, chem.name = "triclosan")
**calc_total_clearance**  
*Calculate the total plasma clearance.*

**Description**

This function calculates the total clearance rate for a one compartment model for plasma where clearance is entirely due to metabolism by the liver and glomerular filtration in the kidneys, identical to clearance of three compartment steady state model.

**Usage**

```r
calc_total_clearance(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  species = "Human",
  suppress.messages = FALSE,
  default.to.human = FALSE,
  well.stirred.correction = TRUE,
  restrictive.clearance = TRUE,
  adjusted.Funbound.plasma = TRUE,
  ...
)
```

**Arguments**

- `chem.cas`: Either the chemical name, CAS number, or the parameters must be specified.
- `chem.name`: Either the chemical name, CAS number, or the parameters must be specified.
- `dtxsid`: EPA's 'DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs.
- `parameters`: Chemical parameters from parameterize_steadystate function, overrides chem.name and chem.cas.
- `species`: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- `suppress.messages`: Whether or not the output message is suppressed.
- `default.to.human`: Substitutes missing animal values with human values if true.
- `well.stirred.correction`: Uses correction in calculation of hepatic clearance for well-stirred model if TRUE. This assumes clearance relative to amount unbound in whole blood instead of plasma, but converted to use with plasma concentration.
- `restrictive.clearance`: Protein binding is not taken into account (set to 1) in liver clearance if FALSE.
- `adjusted.Funbound.plasma`: Uses adjusted Funbound.plasma when set to TRUE.
- `...`: Additional parameters passed to parameterize_steadystate if parameters is NULL.
**calc_vdist**

**Value**

Total Clearance

Units of L/h/kg BW.

**Author(s)**

John Wambaugh

**Examples**

```r
calc_total_clearance(chem.name="Ibuprofen")
```

---

**calc_vdist**  
*Calculate the volume of distribution for a one compartment model.*

**Description**

This function predicts partition coefficients for all tissues, then lumps them into a single compartment.

**Usage**

```r
calc_vdist(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  default.to.human = FALSE,
  species = "Human",
  suppress.messages = FALSE,
  adjusted.Funbound.plasma = TRUE,
  regression = TRUE,
  minimum.Funbound.plasma = 1e-04
)
```

**Arguments**

- **chem.cas**  
  Either the CAS number or the chemical name must be specified when Funbound.plasma is not given in parameter list.

- **chem.name**  
  Either the chemical name or the CAS number must be specified when Funbound.plasma is not given in parameter list.

- **dtxsid**  
  EPA’s DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSID.

- **parameters**  
  Parameters from parameterize_3comp, parameterize_pbtk or predict_partitioning_schmitt.

- **default.to.human**  
  Substitutes missing animal values with human values if true.

- **species**  
  Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
suppress.messages
Whether or not the output message is suppressed.

adjusted.Funbound.plasma
Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.

regression
Whether or not to use the regressions in calculating partition coefficients.

minimum.Funbound.plasma
Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

Details
The effective volume of distribution is calculated by summing each tissues volume times it’s partition coefficient relative to plasma. Plasma, and the partitioning into RBCs are also added to get the total volume of distribution in L/KG BW. Partition coefficients are calculated using Schmitt’s (2008) method. When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value
Volume of distribution
Units of L/ kg BW.

Author(s)
John Wambaugh and Robert Pearce

References

Examples
calc_vdist(chem.cas="80-05-7")
calc_vdist(chem.name="Bisphenol A")
calc_vdist(chem.name="Bisphenol A",species="Rat")

---

CAS.checksum
Test the check digit of a CAS number to confirm validity

Description
Chemical abstracts services registry numbers (CAS-RN) include a final digit as a "checksum" to test for validity (that is, that the number has not been corrupted).

Usage
CAS.checksum(CAS.string)
Arguments

CAS.string  A character string of three numbers separated by two dashes

Details

The check digit (final number) is calculated by working from right to left, starting with the second to last digit of the CAS-RN. We multiply each digit by an increasing digit (1, 2, 3...) and sum as we work from right to left. The check digit should equal the final digit of the sum.

Value

logical (TRUE if final digit of CAS is consistent with other digits)

Author(s)

John Wambaugh

chem.invivo.PK.aggregate.data

Parameter Estimates from Wambaugh et al. (2018)

Description

This table includes 1 and 2 compartment fits of plasma concentration vs time data aggregated from chem.invivo.PK.data, performed in Wambaugh et al. 2018. Data includes volume of distribution (Vdist, L/kg), elimination rate (kelim, 1/h), gut absorption rate (kgutabs, 1/h), fraction absorbed (Fgutabs), and steady state concentration (Css, mg/L).

Usage

chem.invivo.PK.aggregate.data

Format

data.frame

Author(s)

John Wambaugh

Source

Wambaugh et al. 2018 Toxicological Sciences, in press
Published toxicokinetic time course measurements

Description

This data set includes time and dose specific measurements of chemical concentration in tissues taken from animals administered control doses of the chemicals either orally or intravenously. This plasma concentration-time data is from rat experiments reported in public sources. Toxicokinetic data were retrieved from those studies by the Netherlands Organisation for Applied Scientific Research (TNO) using curve stripping (TechDig v2). This data is provided for statistical analysis as in Wambaugh et al. 2018.

Usage

chem.invivo.PK.data

Format

A data.frame containing 597 rows and 13 columns.

Author(s)

Sieto Bosgra

Source

Wambaugh et al. 2018 Toxicological Sciences, in press

References


chem.invivo.PK.summary.data

Summary of published toxicokinetic time course experiments

Description
This data set summarizes the time course data in the chem.invivo.PK.data table. Maximum concentration (Cmax), time integrated plasma concentration for the duration of treatment (AUC.treatment) and extrapolated to zero concentration (AUC.infinity) as well as half-life are calculated. Summary values are given for each study and dosage. These data can be used to evaluate toxicokinetic model predictions.

Usage
chem.invivo.PK.summary.data

Format
A data.frame containing 100 rows and 25 columns.

Author(s)
John Wambaugh

Source
Wambaugh et al. 2018 Toxicological Sciences, in press

References


Description

This data set contains the necessary information to make basic, high-throughput toxicokinetic (HTTK) predictions for compounds, including Funbound.plasma, molecular weight (g/mol), logP, logMA (membrane affinity), intrinsic clearance(uL/min/10^6 cells), and pKa. These data have been compiled from multiple sources, and can be used to parameterize a variety of toxicokinetic models. See variable EPA.ref for information on the reference EPA.

Usage

chem.physical_and_invitro.data

Format

A data.frame containing 9411 rows and 54 columns.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compound</td>
<td>The preferred name of the chemical compound</td>
<td>none</td>
</tr>
<tr>
<td>CAS</td>
<td>The preferred Chemical Abstracts Service Registry Number</td>
<td>none</td>
</tr>
<tr>
<td>CAS.Checksum</td>
<td>A logical indicating whether the CAS number is valid</td>
<td>none</td>
</tr>
<tr>
<td>DTXSID</td>
<td>DSSTox Structure ID (<a href="http://comptox.epa.gov/dashboard">http://comptox.epa.gov/dashboard</a>)</td>
<td>none</td>
</tr>
<tr>
<td>Formula</td>
<td>The proportions of atoms within the chemical compound</td>
<td>none</td>
</tr>
<tr>
<td>SMILES.desalt</td>
<td>The simplified molecular-input line-entry system structure</td>
<td>none</td>
</tr>
<tr>
<td>All.Compound.Names</td>
<td>All names of the chemical as they occurred in the data</td>
<td>none</td>
</tr>
<tr>
<td>logHenry</td>
<td>The log10 Henry’s law constant</td>
<td>log10(atmosphers*m^3/mole)</td>
</tr>
<tr>
<td>logHenry.Reference</td>
<td>Reference for Henry’s law constant</td>
<td></td>
</tr>
<tr>
<td>logP</td>
<td>The log10 octanol:water partition coefficient (PC)</td>
<td>log10 unitless ratio</td>
</tr>
<tr>
<td>logP.Reference</td>
<td>Reference for logP</td>
<td></td>
</tr>
<tr>
<td>logPwa</td>
<td>The log10 water:air PC</td>
<td>log10 unitless ratio</td>
</tr>
<tr>
<td>logPwa.Reference</td>
<td>Reference for logPwa</td>
<td></td>
</tr>
<tr>
<td>logMA</td>
<td>The log10 phospholipid:water PC or &quot;Membrane affinity&quot;</td>
<td>unitless ratio</td>
</tr>
<tr>
<td>logMA.Reference</td>
<td>Reference for membrane affinity</td>
<td></td>
</tr>
<tr>
<td># &quot; logWSol</td>
<td>The log10 water solubility</td>
<td>log10(mole/L)</td>
</tr>
<tr>
<td>logWSol.Reference</td>
<td>Reference for logWsol</td>
<td></td>
</tr>
<tr>
<td>MP</td>
<td>The chemical compound melting point</td>
<td>degrees Celsius</td>
</tr>
<tr>
<td>MP-reference</td>
<td>Reference for melting point</td>
<td></td>
</tr>
<tr>
<td>MW</td>
<td>The chemical compound molecular weight</td>
<td>g/mol</td>
</tr>
<tr>
<td>MW.Reference</td>
<td>Reference for molecular weight</td>
<td></td>
</tr>
<tr>
<td>pKa_Accept</td>
<td>The hydrogen acceptor equilibria concentrations</td>
<td>logarithm</td>
</tr>
<tr>
<td>pKa_Accept.Reference</td>
<td>Reference for pKa_Accept</td>
<td></td>
</tr>
<tr>
<td>pKa_Donor</td>
<td>The hydrogen acceptor equilibria concentrations</td>
<td>logarithm</td>
</tr>
<tr>
<td>pKa_Donor.Reference</td>
<td>Reference for pKa_Donor</td>
<td></td>
</tr>
<tr>
<td>All.Species</td>
<td>All species for which data were available</td>
<td>none</td>
</tr>
<tr>
<td>DTXSID.Reference</td>
<td>Reference for DTXSID</td>
<td></td>
</tr>
<tr>
<td>Formula.Reference</td>
<td>Reference for chemical formulat</td>
<td></td>
</tr>
<tr>
<td>[SPECIES].Clint</td>
<td>(Primary hepatocyte suspension) intrinsic hepatic clearance</td>
<td>uL/min/10^6 hepatocytes</td>
</tr>
</tbody>
</table>
Details

In some cases the rapid equilibrium dialysis method (Waters et al., 2008) fails to yield detectable concentrations for the free fraction of chemical. In those cases we assume the compound is highly bound (that is, Fup approaches zero). For some calculations (for example, steady-state plasma concentration) there is precedent (Rotroff et al., 2010) for using half the average limit of detection, that is 0.005. We do not recommend using other models where quantities like partition coefficients must be predicted using Fup. We also do not recommend including the value 0.005 in training sets for Fup predictive models.

Note that in some cases the Funbound.plasma and the intrinsic clearance are provided as a series of numbers separated by commas. These values are the result of Bayesian analysis and characterize a distribution: the first value is the median of the distribution, while the second and third values are the lower and upper 95th percentile (that is quantile 2.5 and 97.5) respectively. For intrinsic clearance a fourth value indicating a p-value for a decrease is provided. Typically 4000 samples were used for the Bayesian analysis, such that a p-value of "0" is equivalent to "<0.00025". See Wambaugh et al. (2019) for more details.

Any one chemical compound may have multiple ionization equilibria (see Strope et al., 2018) may both for donating or accepting a proton (and therefore changing charge state). If there are multiple equilibria of the same type (donor/acceptor) the are concatenated by commas.

All species-specific information is initially from experimental measurements. The functions load_sipes2017, load_pradeep2020, and load_dawson2021 may be used to add in silico, structure-based predictions for many thousands of additional compounds to this table.

Author(s)

John Wambaugh

Source


References

CompTox Chemicals Dashboard (http://comptox.epa.gov/dashboard)


Paini, Alicia; Cole, Thomas; Meinero, Maria; Carpi, Donatella; Decuvinck, Pierre; Macko, Peter; Palosaari, Taiana; Sund, Jukka; Worth, Andrew; Whelan, Maurice (2020): EURL ECVAM in vitro hepatocyte clearance and blood plasma protein binding dataset for 77 chemicals. European Commission, Joint Research Centre (JRC) [Dataset] PID: https://data.europa.eu/89h/a2ff867f-db80-4acf-8e5c-e45502713bee


ckd_epi_eq

*CKD-EPI equation for GFR.*

**Description**

Predict GFR from serum creatinine, gender, and age.

**Usage**

```r
ckd_epi_eq(scr, gender, reth, age_years, ckd_epi_race_coeff = FALSE)
```

**Arguments**

- `scr` Vector of serum creatinine values in mg/dL.
- `gender` Vector of genders (either ‘Male’ or ‘Female’).
- `reth` Vector of races/ethnicities. Not used unless `ckd_epi_race_coeff` is TRUE.
- `age_years` Vector of ages in years.
- `ckd_epi_race_coeff` Whether to use the "race coefficient" in the CKD-EPI equation. Default is FALSE.

**Details**


**Value**

Vector of GFR values in mL/min/1.73m^2.
**concentration_data_Linakis2020**

**Author(s)**
Caroline Ring

**References**

---

**concentration_data_Linakis2020**

*Concentration data involved in Linakis 2020 vignette analysis.*

---

**Description**
Concentration data involved in Linakis 2020 vignette analysis.

**Usage**

```r
concentration_data_Linakis2020
```

**Format**
A data.frame containing x rows and y columns.

**Author(s)**
Matt Linakis

**Source**
Matt Linakis

**References**
DSStox database (https://www.epa.gov/ncct/dsstox

---

**convert_httkpop_1comp**

*Converts HTTK-Pop physiology into parameters relevant to the one compartment model*

---

**Description**
Converts HTTK-Pop physiology into parameters relevant to the one compartment model

**Usage**

```r
convert_httkpop_1comp(parameters.dt, httkpop.dt, ...)
```
convert_solve_x

Arguments

- parameters.dt: Data table returned by `create_mc_samples`.
- htttkpop.dt: Data table returned by `httkpop_generate`.
- Additional arguments passed to `propagate_invitrouv_1comp`.

Value

A data.table whose columns are the parameters of the HTTK model specified in `model`.

Author(s)

Caroline Ring, John Wambaugh, and Greg Honda

References


convert_solve_x

Description

This function is designed to convert compartment values estimated from one of the HTTK models (e.g. "1compartment") using the `solve_model` function. It takes the HTTK model output matrix, model name, desired output units, and compound information to perform the conversion default model units to user specified units.

Usage

```r
convert_solve_x(
  model.output.mat,
  model = NULL,
  output.units = NULL,
  MW = NULL,
  vol = NULL,
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  monitor.vars = NULL,
  suppress.messages = FALSE,
  verbose = FALSE,
  ...
)
```
Arguments

model.output.mat
Matrix of results from HTTK solve_model function.

model
Specified model to use in simulation: "pbtk", "3compartment", "3compartmentss", "1compartment", "schmitt", ...

output.units
Output units of interest for the compiled components. Defaults to NULL, and will provide values in model units if unspecified.

MW
Molecular weight of substance of interest in g/mole

v01
Volume for the target tissue of interest in liters (L). NOTE: Volume should not be in units of per BW, i.e. "kg".

chem.cas
Either the chemical name, CAS number, or the parameters must be specified.

chem.name
Either the chemical name, CAS number, or the parameters must be specified.

dtxsid
EPA's DSSTox Structure ID. (http://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs.

parameters
A set of model parameters, especially a set that includes MW (molecular weight) for our conversions.

monitor.vars
A vector of character strings indicating the model component variables to retain in the conversion factor table (assuming suppress.messages == FALSE). It should also be noted this option does NOT exclude columns from the input matrix provided in the 'model.output.mat' parameter. (Default is NULL, i.e. conversion factors for all model components are included in the reporting matrix.)

suppress.messages
Whether or not the output messages are suppressed. (Default is FALSE, i.e. show messages.)

verbose
Whether or not to display the full conversion factor table. (Default is FALSE, i.e. only include rows where the conversion factor is 1.)

...
Other parameters that can be passed to convert_units, e.g. temperature and compound state. See details in convert_units.

Details

The function can be used to convert all compartments to a single unit, only units for a single model compartment, or units for a set of model compartments.

More details on the unit conversion can be found in the documentation for convert_units.

Value

'new.output.matrix' A matrix with a column for time (in days), each compartment, and the area under the curve (AUC) and a row for each time point. The compartment and AUC columns are converted from model specified units to user specified units.

'output.units.vector' A vector of character strings providing the model compartments and their corresponding units after convert_solve_x.

Author(s)

Sarah E. Davidson
See Also

convert_units

Examples

```r
output.mat <- solve_1comp(dtxsid = "DTXSID0020573")
new.output.mat <- convert_solve_x(output.units = "mg",
                                  model.output.mat = output.mat,
                                  model = "1compartment",
                                  dtxsid = "DTXSID0020573")
```

Description

This function is designed to accept input units, output units, and the molecular weight (MW) of a substance of interest to then use a table lookup to return a scaling factor that can be readily applied for the intended conversion. It can also take chemical identifiers in the place of a specified molecular weight value to retrieve that value for its own use.

Usage

```r
convert_units(
    input.units = NULL,
    output.units = NULL,
    MW = NULL,
    vol = NULL,
    chem.cas = NULL,
    chem.name = NULL,
    dtxsid = NULL,
    parameters = NULL,
    temp = 25,
    state = "liquid"
)
```

Arguments

- `input.units`: Assigned input units of interest
- `output.units`: Desired output units
- `MW`: Molecular weight of substance of interest in g/mole
- `vol`: Volume for the target tissue of interest in liters (L). NOTE: Volume should not be in units of per BW, i.e. "kg".
- `chem.cas`: Either the chemical name, CAS number, or the parameters must be specified.
- `chem.name`: Either the chemical name, CAS number, or the parameters must be specified.
- `dtxsid`: EPA’s DSSTox Structure ID ([http://comptox.epa.gov/dashboard](http://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs
- `parameters`: A set of model parameters, especially a set that includes MW (molecular weight) for our conversions
\textit{convert\_units}

\begin{itemize}
\item \textbf{temp} \quad Temperature for conversions (default = 25 degrees C)
\item \textbf{state} \quad Chemical state (gas or default liquid)
\end{itemize}

Details

If input or output units not contained in the table are queried, it gives a corresponding error message. It gives a warning message about the handling of ‘ppmv,’ as the function is only set up to convert between ppmv and mass-based units (like mg/m$^3$ or umol/L) in the context of ideal gases.

\texttt{convert\_units} is not directly configured to accept and convert units based on BW, like mg/kg. For this purpose, see \texttt{scale\_dosing}.

The function supports a limited set of most relevant units across toxicological models, currently including umol, uM, mg, mg/L, mg/m$^3$ or umol/L), and in the context of gases assumed to be ideal, ppmv.

\textit{Andersen and Clewell's Rules of PBPK Modeling}:

\begin{itemize}
\item 1Check Your Units
\item 2Check Your Units
\item 3Check Mass Balance
\end{itemize}

Author(s)

Mark Sfeir, John Wambaugh, and Sarah E. Davidson

Examples

\begin{verbatim}
# MW BPA is 228.29 g/mol
# 1 mg/L -> 1/228.29*1000 = 4.38 uM
convert_units("mg/L","uM",chem.cas="80-05-7")

# MW Diclofenac is 296.148 g/mol
# 1 uM -> 296.148/1000 = 0.296
convert_units("uM","mg/L",chem.name="diclofenac")

# 1 ppmv Toluene -> 0.263 ppmv
convert_units("ug/L","ppmv",chem.name="toluene")

# Compare with https://www3.epa.gov/ceampubl/learn2model/part-two/onsite/ia_unit_conversion.html
# 1 pppmv Toluene, 0.0038 mg/L
convert_units("ppmv","mg/L",chem.name="toluene")

MW_pyrene <- get_physchem_param(param='MW', chem.name='pyrene')
conversion_factor <- convert_units(input.units='mg/L', output.units = 'uM', MW=MW_pyrene)
\end{verbatim}
create_mc_samples

Create a table of parameter values for Monte Carlo

Description

This is the HTTK master function for creating a data table for use with Monte Carlo methods to simulate parameter uncertainty and variability. Each column of the output table corresponds to an HTTK model parameter and each row corresponds to a different random draw (for example, different individuals when considering biological variability). This function call three different key functions to simulate parameter parameter uncertainty and/or variability in one of three ways. First parameters can be varied in an uncorrelated manner using truncated normal distributions by the function `monte_carlo`. Then, physiological parameters can be varied in a correlated manner according to the Ring et al. (2017) (doi:10.1016/j.envint.2017.06.004) `httk-pop` approach by the function `httkpop_mc`. Next, both uncertainty and variability of in vitro HTTK parameters can be simulated by the function `invitro_mc` as described by Wambaugh et al. (2019) (doi:10.1093/toxsci/kfkz205). Finally, tissue-specific partition coefficients are predicted for each draw using the Schmitt (2008) (doi:10.1016/j.tiv.2007.09.010) method as calibrated to in vivo data by Pearce et al. (2017) (doi:10.1007/s1092801795487) and implemented in `predict_partitioning_schmitt`.

Usage

```r
create_mc_samples(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  samples = 1000,
  species = "Human",
  suppress.messages = FALSE,
  model = "3compartmentss",
  httkpop = TRUE,
  invitrouv = TRUE,
  calcrb2p = TRUE,
  censored.params = list(),
  vary.params = list(),
  return.samples = FALSE,
  tissue = NULL,
  httkpop.dt = NULL,
  invitrotmc.arg.list = list(adjusted.Funbound.plasma = TRUE, poormetab = TRUE,
    fup.censored.dist = FALSE, fup.lod = 0.01, fup.meas.cv = 0.4, clint.meas.cv = 0.3,
    fup.pop.cv = 0.3, clint.pop.cv = 0.3),
  httkpop.generate.arg.list = list(method = "direct resampling", gendernum = NULL,
    agelim.years = NULL, agelim.months = NULL, weight.category = c("Underweight",
      "Normal", "Overweight", "Obese"), gfr.category = c("Normal", "Kidney Disease",
      "Kidney Failure"), reths = c("Mexican American", "Other Hispanic",
      "Non-Hispanic White", "Non-Hispanic Black", "Other")),
  convert.httkpop.arg.list = list(),
  propagate.invitrouv.arg.list = list(),
  parameterize.arg.list = list(restrictive.clearance = TRUE, default.to.human = FALSE,
    clint.pvalue.threshold = 0.05, regression = TRUE)
)
```
Arguments

chem.cas
Chemical Abstract Services Registry Number (CAS-RN) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD.

chem.name
Chemical name (spaces and capitalization ignored) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD.

dtxsid
EPA’s DSSTox Structure ID (https://comptox.epa.gov/dashboard) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD.

parameters
Parameters from the appropriate parameterization function for the model indicated by argument model.

samples
Number of samples generated in calculating quantiles.

species
Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human"). Species must be set to "Human" to run httkpop model.

suppress.messages
Whether or not to suppress output message.

model
Model used in calculation: ‘pbtk’ for the multiple compartment model, ‘3compartment’ for the three compartment model, ‘3compartmentss’ for the three compartment steady state model, and ‘1compartment’ for one compartment model. This only applies when httkpop=TRUE and species="Human", otherwise ‘3compartmentss’ is used.

httkpop
Whether or not to use the Ring et al. (2017) "httkpop" population generator. Species must be "Human".

invitrouv
Logical to indicate whether to include in vitro parameters such as intrinsic hepatic clearance rate and fraction unbound in plasma in uncertainty and variability analysis.

calcrb2p
Logical determining whether or not to recalculate the chemical ratio of blood to plasma.

censored.params
The parameters listed in censored.params are sampled from a normal distribution that is censored for values less than the limit of detection (specified separately for each parameter). This argument should be a list of sub-lists. Each sublist is named for a parameter in "parameters" and contains two elements: "CV" (coefficient of variation) and "LOD" (limit of detection, below which parameter values are censored. New values are sampled with mean equal to the value in "parameters" and standard deviation equal to the mean times the CV. Censored values are sampled on a uniform distribution between 0 and the limit of detection. Not used with httkpop model.

vary.params
The parameters listed in vary.params are sampled from a normal distribution that is truncated at zero. This argument should be a list of coefficients of variation (CV) for the normal distribution. Each entry in the list is named for a parameter in "parameters". New values are sampled with mean equal to the value in "parameters" and standard deviation equal to the mean times the CV. Not used with httkpop model.

return.samples
Whether or not to return the vector containing the samples from the simulation instead of the selected quantile.

tissue
Desired steady state tissue concentration.

httkpop.dt
A data table generated by httkpop_generate. This defaults to NULL, in which case httkpop_generate is called to generate this table.
Details

The Monte Carlo methods used here were recently updated and described by Breen et al. (submitted).

We aim to make any function that uses chemical identifiers (name, CAS, DTXSID) also work if passed a complete vector of parameters (that is, a row from the table generated by this function). This allows the use of Monte Carlo to vary the parameters and therefore vary the function output. Depending on the type of parameters (for example, physiological vs. in vitro measurements) we vary the parameters in different ways with different functions.

Value

A data table where each column corresponds to parameters needed for the specified model and each row represents a different Monte Carlo sample of parameter values.

Author(s)

Caroline Ring, Robert Pearce, and John Wambaugh

References


Examples

```python
sample_set = create_mc_samples(chem.name = 'bisphenol a')
```
Description

This table includes QSAR (Random Forest) model predicted values for unbound fraction plasma protein (fup) and intrinsic hepatic clearance (clint) for a subset of chemicals in the Tox21 library (see https://www.epa.gov/chemical-research/toxicology-testing-21st-century-tox21).

Usage
dawson2021

Format
data.frame

Details

Predictions were made with a set of Random Forest QSAR models, as reported in Dawson et al. (2021).

Author(s)
Daniel E. Dawson

Source
Dawson et al. 2021 Random Forest QSAR Model

References
Dawson, Daniel E. et al. "Designing QSARs for parameters of high-throughput toxicokinetic models using open-source descriptors." Environmental Science & Technology____. (2021):______.
**Author(s)**

John Wambaugh

**Source**

https://comptox.epa.gov/dashboard

---

**estimate_gfr**

*Predict GFR.*

---

**Description**

Predict GFR using CKD-EPI equation (for adults) or BSA-based equation (for children).

**Usage**

`estimate_gfr(gfrtmp.dt, gfr_resid_var = TRUE, ckd_epi_race_coeff = FALSE)`

**Arguments**

- `gfrtmp.dt`: A data.table with columns `gender`, `reth`, `age_years`, `age_months`, `BSA_adj`, `serum_creat`.
- `gfr_resid_var`: Logical value indicating whether or not to include residual variability when generating GFR values. (Default is TRUE.)
- `ckd_epi_race_coeff`: Logical value indicating whether or not to use the "race coefficient" from the CKD-EPI equation when estimating GFR values. (Default is FALSE.)

**Details**

Add residual variability based on reported residuals for each equation.

**Value**

The same data.table with a `gfr_est` column added, containing estimated GFR values.

**Author(s)**

Caroline Ring

**References**

### estimate_gfr_ped

*Predict GFR in children.*

**Description**


**Usage**

```r
estimate_gfr_ped(BSA)
```

**Arguments**

- **BSA**: Vector of body surface areas in m².

**Value**

Vector of GFRs in mL/min/1.73m².

**Author(s)**

Caroline Ring

**References**


---

### estimate_hematocrit

*Generate hematocrit values for a virtual population*

**Description**

Predict hematocrit from age using smoothing splines and kernel density estimates of residual variability fitted to NHANES data, for a given combination of gender and NHANES race/ethnicity category.

**Usage**

```r
estimate_hematocrit(gender, reth, age_years, age_months, nhanes_mec_svy)
```
Arguments

- **gender**: Gender for which to generate hematocrit values ("Male" or "Female")
- **reth**: NHANES race/ethnicity category for which to generate serum creatinine values ("Mexican American", "Non-Hispanic Black", "Non-Hispanic White", "Other", or "Other Hispanic")
- **age_years**: Vector of ages in years for individuals for whom to generate hematocrit values (corresponding to age_months)
- **age_months**: Vector of ages in months for individuals for whom to generate hematocrit values (between 0-959 months)
- **nhanes_mec_svy**: Survey design object created from `mecdt` using `svydesign` (this is done in `httkpop_generate`)

Details

This function should usually not be called directly by the user. It is used by `httkpop_generate()` in "virtual-individuals" mode, after drawing gender, NHANES race/ethnicity category, and age from their NHANES proportions/distributions.

Value

A vector of numeric generated hematocrit values (blood percentage red blood cells by volume).

Author(s)

Caroline Ring

References


Description

This function exports the multiple compartment PBTK model to a jarnac file.

Usage

```r
export_pbtk_jarnac(
  chem.cas = NULL,
  chem.name = NULL,
  species = "Human",
  initial.amounts = list(Agutlumen = 0),
  filename = "default.jan",
  digits = 4
)
```
export_pbtk_sbml

Arguments

chem.cas Either the chemical name or CAS number must be specified.
chem.name Either the chemical name or CAS number must be specified.
species Species desired (either "Rat", "Rabbit", "Dog", or default "Human").
initial.amounts Must specify initial amounts in units of choice.
filename The name of the jarnac file containing the model.
digits Desired number of decimal places to round the parameters.

Details

Compartments to enter into the initial.amounts list includes Agutlumen, Aart, Aven, Alung, Agut, Aliver, Akidney, and Arest.
When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value

Text containing a Jarnac language version of the PBTK model.

Author(s)

Robert Pearce

Examples

export_pbtk_jarnac(chem.name='Nicotine', initial.amounts=list(Agutlumen=1), filename='PBTKmodel.jan')

Description

This function exports the multiple compartment PBTK model to an sbml file.

Usage

export_pbtk_sbml(
  chem.cas = NULL,
  chem.name = NULL,
  species = "Human",
  initial.amounts = list(Agutlumen = 0),
  filename = "default.xml",
  digits = 4
)
Arguments

- **chem.cas**: Either the chemical name or CAS number must be specified.
- **chem.name**: Either the chemical name or CAS number must be specified.
- **species**: Species desired (either "Rat", "Rabbit", "Dog", or default "Human").
- **initial.amounts**: Must specify initial amounts in units of choice.
- **filename**: The name of the jarnac file containing the model.
- **digits**: Desired number of decimal places to round the parameters.

Details

Compartments to enter into the initial.amounts list includes Agutlumen, Aart, Aven, Alung, Agut, Aliver, Akidney, and Arest.

When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value

Text describing the PBTK model in SBML.

Author(s)

Robert Pearce

Examples

```r
export_pbtk_sbml(chem.name='Nicotine', initial.amounts=list(Agutlumen=1), filename='PBTKmodel.xml')
```

---

### fetalpcs

**Fetal Partition Coefficients**

Description

Partition coefficients were measured for tissues, including placenta, in vitro by Csanady et al. (2002) for Bisphenol A and Diadzen. Curley et al. (1969) measured the concentration of a variety of pesticides in the cord blood of newborns and in the tissues of infants that were stillborn.

Usage

```r
fetalpcs
```

Format

- `data.frame`
Details

Three of the chemicals studied by Curley et al. (1969) were modeled by Weijs et al. (2013) using the same partition coefficients for mother and fetus. The values used represented "prior knowledge" summarizing the available literature.

Source

Kapraun et al. 2021 (submitted)

References


Frank2018invivo

Literature In Vivo Data on Doses Causing Neurological Effects

Description

Studies were selected from Table 1 in Mundy et al., 2015, as the studies in that publication were cited as examples of compounds with evidence for developmental neurotoxicity. There were sufficient in vitro toxicokinetic data available for this package for only 6 of the 42 chemicals.

Usage

Frank2018invivo

Format

A data.frame containing 14 rows and 16 columns.

Author(s)

Timothy J. Shafer

References


**gen_age_height_weight**  Generate demographic parameters for a virtual population

**Description**
Generate gender, NHANES race/ethnicity category, ages, heights, and weights for a virtual population, based on NHANES data.

**Usage**
```
gen_age_height_weight(
  nsamp = NULL,
  gendernum = NULL,
  reths,
  weight_category,
  agelim_years,
  agelim_months,
  nhanes_mec_svy
)
```

**Arguments**
- **nsamp**
  The desired number of individuals in the virtual population. `nsamp` need not be provided if `gendernum` is provided.
- **gendernum**
  Optional: A named list giving the numbers of male and female individuals to include in the population, e.g. `list(Male=100,Female=100)`. Default is NULL, meaning both males and females are included, in their proportions in the NHANES data. If both `nsamp` and `gendernum` are provided, they must agree (i.e., `nsamp` must be the sum of `gendernum`).
- **reths**
  Optional: A character vector giving the races/ethnicities to include in the population. Default is `c('Mexican American','Other Hispanic','Non-Hispanic White','Non-Hispanic Black','Other')`, to include all races and ethnicities in their proportions in the NHANES data. User-supplied vector must contain one or more of these strings.
- **weight_category**
  Optional: The weight categories to include in the population. Default is `c('Underweight','Normal','Overweight','Obese')`. User-supplied vector must contain one or more of these strings.
- **agelim_years**
  Optional: A two-element numeric vector giving the minimum and maximum ages (in years) to include in the population. Default is `c(0,79)`. If `agelim_years` is provided and `agelim_months` is not, `agelim_years` will override the default value of `agelim_months`.
- **agelim_months**
  Optional: A two-element numeric vector giving the minimum and maximum ages (in months) to include in the population. Default is `c(0, 959)`, equivalent to the default `agelim_years`. If `agelim_months` is provided and `agelim_years` is not, `agelim_months` will override the default values of `agelim_years`.
- **nhanes_mec_svy**
  Surveydesign object created from `mecdt` using `svydesign` (this is done in `httkpop_generate`
Details

This function should usually not be called directly by the user. It is used by `httkpop_generate()` in "virtual-individuals" mode.

Value

A data.table containing variables

- `gender`: Gender of each virtual individual
- `reth`: Race/ethnicity of each virtual individual
- `age_months`: Age in months of each virtual individual
- `age_years`: Age in years of each virtual individual
- `weight`: Body weight in kg of each virtual individual
- `height`: Height in cm of each virtual individual

Author(s)

Caroline Ring

References


```r
importFrom survey svymean
```

---

`gen_height_weight`      
*Generate heights and weights for a virtual population.*

Description

Predict height and weight from age using smoothing splines, and then add residual variability from a 2-D KDE, both fitted to NHANES data, for a given combination of gender and NHANES race/ethnicity category.

Usage

```r
gen_height_weight(gender, reth, age_months, nhanes_mec_svy)
```

Arguments

- `gender`: Gender for which to calculate height/weight ("Male" or "Female")
- `reth`: NHANES race/ethnicity category for which to calculate height/weight ("Mexican American", "Non-Hispanic Black", "Non-Hispanic White", "Other", or "Other Hispanic")
- `age_months`: vector of ages in months for individuals for whom to calculate height/weight (between 0-959 months)
- `nhanes_mec_svy`: surveydesign object created from `mecdt` using `svydesign` (this is done in `httkpop_generate`)

```r
importFrom survey svymean
```
Details

This function should usually not be called directly by the user. It is used by `httkpop_generate()` in "virtual-individuals" mode, after drawing gender, NHANES race/ethnicity category, and age from their NHANES proportions/distributions.

Value

A list containing two named elements, `weight` and `height`, each of which is a numeric vector. `weight` gives individual body weights in kg, and `height` gives individual heights in cm, corresponding to each item in the input `age_months`.

Author(s)

Caroline Ring

References


---

**gen_serum_creatinine**

*Generate serum creatinine values for a virtual population.*

**Description**

Predict serum creatinine from age using smoothing splines and kernel density estimates of residual variability fitted to NHANES data, for a given combination of gender and NHANES race/ethnicity category.

**Usage**

`gen_serum_creatinine(gender, reth, age_years, age_months, nhanes_mec_svy)`

**Arguments**

- `gender` Gender for which to generate serum creatinine values ("Male" or "Female")
- `reth` NHANES race/ethnicity category for which to generate serum creatinine values ("Mexican American", "Non-Hispanic Black", "Non-Hispanic White", "Other", or "Other Hispanic")
- `age_years` Vector of ages in years for individuals for whom to generate serum creatinine values (corresponding to `age_months`)
- `age_months` Vector of ages in months for individuals for whom to generate serum creatinine values (between 0-959 months)
- `nhanes_mec_svy` surveydesign object created from `mecdt` using `svydesign` (this is done in `httkpop_generate()`)
Value
A vector of numeric generated serum creatinine values (mg/dL).

Author(s)
Caroline Ring

References
Ring, Caroline L., et al. "Identifying populations sensitive to environmental chemicals by simulat-
ing toxicokinetic variability." Environment International 106 (2017): 105-118

get_cheminfo
Retrieve chemical information from HTTK package

Description
This function provides the information specified in "info=" (can be single entry or vector) for all
chemicals for which a toxicokinetic model can be parameterized for a given species. Since different
models have different requirements and not all chemicals have complete data, this function will
return different number of chemicals depending on the model specified.

Usage
get_cheminfo(
  info = "CAS",
  species = "Human",
  fup.lod.default = 0.005,
  model = "3compartmentss",
  default.to.human = FALSE,
  median.only = FALSE,
  fup.ci.cutoff = TRUE,
  clint.pvalue.threshold = 0.05,
  suppress.messages = FALSE
)

Arguments
info A single character vector (or collection of character vectors) from "Compound", "CAS", "DTXSID", "logP", "pKa_Donor", "pKa_Accept", "MW", "Clint", "Clint.pValue", "Funbound.plasma", "Structure_Formula", or "Substance_Type". info="all" gives all information for the model and species.
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
fup.lod.default Default value used for fraction of unbound plasma for chemicals where mea-
sured value was below the limit of detection. Default value is 0.005.
model Model used in calculation, ‘pbtk’ for the multiple compartment model, ‘1com-
partment’ for the one compartment model, ‘3compartment’ for three compart-
ment model, ‘3compartmentss’ for the three compartment model without par-
tition coefficients, or ‘schmitt’ for chemicals with logP and fraction unbound
(used in predict_partitioning_schmitt).
default.to.human
Substitutes missing values with human values if true.

median.only
Use median values only for fup and clint. Default is FALSE.

fup.ci.cutoff
Cutoff for the level of uncertainty in fup estimates. This value should be between (0,1). Default is ‘NULL’ specifying no filtering.

clint.pvalue.threshold
Hepatic clearance for chemicals where the in vitro clearance assay result has a p-values greater than the threshold are set to zero.

suppress.messages
Whether or not the output messages are suppressed.

Details
When default.to.human is set to TRUE, and the species-specific data, Funbound.plasma and Clint, are missing from chem.physical_and_invitro.data, human values are given instead.

In some cases the rapid equilibrium dailysis method (Waters et al., 2008) fails to yield detectable concentrations for the free fraction of chemical. In those cases we assume the compound is highly bound (that is, Fup approaches zero). For some calculations (for example, steady-state plasma concentration) there is precedent (Rotroff et al., 2010) for using half the average limit of detection, that is, 0.005 (this value is configurable via the argument fup.lod.default). We do not recomend using other models where quantities like partition coefficients must be predicted using Fup. We also do not recomend including the value 0.005 in training sets for Fup predictive models.

Note that in some cases the Funbound.plasma and the intrinsic clearance are provided as a series of numbers separated by commas. These values are the result of Bayesian analysis and characterize a distribution: the first value is the median of the distribution, while the second and third values are the lower and upper 95th percentile (that is qunatile 2.5 and 97.5) respectively. For intrinsic clearance a fourth value indicating a p-value for a decrease is provided. Typically 4000 samples were used for the Bayesian analysis, such that a p-value of ’0’ is equivale to ”<0.00025”. See Wambaugh et al. (2019) for more details. If argument median.only == TRUE then only the median is reported for parameters with Bayesian analysis distributions. If the 95 credible interval is larger than fup.ci.cutoff (defaults to NULL) then the Fup is treated as too uncertain and the value NA is given.

Value

vector/data.table
Table (if info has multiple entries) or vector containing a column for each valid entry specified in the argument ”info” and a row for each chemical with sufficient data for the model specified by argument ”model”:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
<th>units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compound</td>
<td>The preferred name of the chemical compound</td>
<td>none</td>
</tr>
<tr>
<td>CAS</td>
<td>The preferred Chemical Abstracts Service Registry Number</td>
<td>none</td>
</tr>
<tr>
<td>DTXSID</td>
<td>DSSTox Structure ID (<a href="http://comptox.epa.gov/dashboard">http://comptox.epa.gov/dashboard</a>)</td>
<td>none</td>
</tr>
<tr>
<td>logP</td>
<td>The log10 octanol:water partition coefficient</td>
<td>log10 unitless ratio</td>
</tr>
<tr>
<td>MW</td>
<td>The chemical compound molecular weight</td>
<td>g/mol</td>
</tr>
<tr>
<td>pKa_Accept</td>
<td>The hydrogen acceptor equilibria concentrations</td>
<td>logarithm</td>
</tr>
<tr>
<td>pKa_Donor</td>
<td>The hydrogen donor equilibria concentrations</td>
<td>logarithm</td>
</tr>
<tr>
<td>[SPECIES].Clint</td>
<td>(Primary hepatocyte suspension) intrinsic hepatic clearance</td>
<td>uL/min/10^6 hepatocytes</td>
</tr>
<tr>
<td>[SPECIES].Clint.pValue</td>
<td>Probability that there is no clearance observed</td>
<td>none</td>
</tr>
<tr>
<td>[SPECIES].Funbound.plasma</td>
<td>Chemical fraction unbound in presence of plasma proteins</td>
<td>unitless fraction</td>
</tr>
<tr>
<td>[SPECIES].Rblood2plasma</td>
<td>Chemical concentration blood to plasma ratio</td>
<td>unitless ratio</td>
</tr>
</tbody>
</table>
Author(s)

John Wambaugh, Robert Pearce, and Sarah E. Davidson

References


Examples

# List all CAS numbers for which the 3compartmentss model can be run in humans:
get_cheminfo()

get_cheminfo(info=c('compound','funbound.plasma','logP'),model='pbtk')

# See all the data for humans:
get_cheminfo(info="all")


httk.TPO.rat.table <- subset(get_cheminfo(info="all",species="rat"), 
CAS %in% TPO.cas)

httk.TPO.human.table <- subset(get_cheminfo(info=c('compound','funbound.plasma','logP'),model='pbtk', 
"median.only") 
CAS %in% TPO.cas)

# create a data.frame with all the Fup values, we ask for model="schmitt" since 
# that model only needs fup, we ask for "median.only" because we don't care 
# about uncertainty intervals here:
fup.tab <- get_cheminfo(info="all",median.only=TRUE,model="schmitt")
# calculate the median, making sure to convert to numeric values:
median(as.numeric(fup.tab$Human.Funbound.plasma),na.rm=TRUE)
# calculate the mean:
mean(as.numeric(fup.tab$Human.Funbound.plasma),na.rm=TRUE)
# count how many non-NA values we have (should be the same as the number of
# rows in the table but just in case we ask for non NA values:
sum(!is.na(fup.tab$Human.Funbound.plasma))

get_chem_id

Retrieve chemical identity from HTTK package

Description

Given one of chem.name, chem.cas (Chemical Abstract Service Registry Number), or DTXSID (DSSTox Substance Identifier https://comptox.epa.gov/dashboard) this function checks if the chemical is available and, if so, returns all three pieces of information.

Usage

get_chem_id(chem.cas = NULL, chem.name = NULL, dtxsid = NULL)

Arguments

chem.cas  CAS registry number
chem.name  Chemical name
dtxsid  DSSTox Substance identifier

Value

A list containing the following chemical identifiers:

chem.cas  CAS registry number
chem.name  Name
dtxsid  DTXSID

Author(s)

John Wambaugh and Robert Pearce
**get_gfr_category**

*Categorize kidney function by GFR.*

**Description**

For adults: In general GFR > 60 is considered normal 15 < GFR < 60 is considered kidney disease GFR < 15 is considered kidney failure

**Usage**

```
get_gfr_category(age_years, age_months, gfr_est)
```

**Arguments**

- `age_years` Vector of ages in years.
- `age_months` Vector of ages in months.
- `gfr_est` Vector of estimated GFR values in mL/min/1.73m^2.

**Details**

These values can also be used for children 2 years old and greater (see PEDIATRICS IN REVIEW Vol. 29 No. 10 October 1, 2008 pp. 335-341 (doi: 10.1542/pir.29-10-335))

**Value**

Vector of GFR categories: ’Normal’, ’Kidney Disease’, ’Kidney Failure’.

**Author(s)**

Caroline Ring

**References**


---

**get_invitroPK_param**

*Retrieve data from chem.physical_and_invitro.data table*

**Description**

or fraction unbound in plasma) from the main HTTK data. This function looks for species-specific values.
Usage

get_invitroPK_param(
  param,
  species,
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL
)

Arguments

param     The in vitro pharmacokinetic parameter needed.

species   Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").

chem.name Either the chemical name, CAS number, or the parameters must be specified.

chem.cas Either the chemical name, CAS number, or the parameters must be specified.

dtxsid    EPA's DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs

Value

The value of the parameter, if found

Author(s)

John Wambaugh and Robert Pearce

gget_lit_cheminfo

Get literature Chemical Information.

Description

This function provides the information specified in "info=" for all chemicals with data from the Wetmore et al. (2012) and (2013) publications and other literature.

Usage

get_lit_cheminfo(info = "CAS", species = "Human")

Arguments


species Species desired (either "Rat" or default "Human").

Value

info Table/vector containing values specified in "info" for valid chemicals.
**get_lit_css**

**Author(s)**

John Wambaugh

**References**


**Examples**

```r
get_lit_cheminfo()
get_lit_cheminfo(info=c('CAS', 'MW'))
```

---

**get_lit_css**

**Get literature Css**

**Description**

This function retrieves a steady-state plasma concentration as a result of infusion dosing from the Wetmore et al. (2012) and (2013) publications and other literature.

**Usage**

```r
get_lit_css(
  chem.cas = NULL,
  chem.name = NULL,
  daily.dose = 1,
  which.quantile = 0.95,
  species = "Human",
  clearance.assay.conc = NULL,
  output.units = "mg/L",
  suppress.messages = FALSE
)
```
Arguments

chem.cas Either the cas number or the chemical name must be specified.
chem.name Either the chemical name or the CAS number must be specified.
daily.dose Total daily dose infused in units of mg/kg BW/day. Defaults to 1 mg/kg/day.
which.quantile Which quantile from the SimCYP Monte Carlo simulation is requested. Can be a vector.
species Species desired (either "Rat" or default "Human").
clearance.assay.conc Concentration of chemical used in measuring intrinsic clearance data, 1 or 10 uM.
output.units Returned units for function, defaults to mg/L but can also be uM (specify units = "uM").
suppress.messages Whether or not the output message is suppressed.

Value

A numeric vector with the literature steady-state plasma concentration (1 mg/kg/day) for the requested quantiles

Author(s)

John Wambaugh

References


Examples

get_lit_css(chem.cas="34256-82-1")

get_lit_css(chem.cas="34256-82-1",species="Rat",which.quantile=0.5)

get_lit_css(chem.cas="80-05-7", daily.dose = 1,which.quantile = 0.5, output.units = "uM")
**get_lit_oral_equiv**

*Get Literature Oral Equivalent Dose*

**Description**

This function converts a chemical plasma concentration to an oral equivalent dose using the values from the Wetmore et al. (2012) and (2013) publications and other literature.

**Usage**

```r
get_lit_oral_equiv(
  conc, 
  chem.name = NULL, 
  chem.cas = NULL, 
  dtxsid = NULL, 
  suppress.messages = FALSE, 
  which.quantile = 0.95, 
  species = "Human", 
  input.units = "uM", 
  output.units = "mg", 
  clearance.assay.conc = NULL, 
  ... 
)
```

**Arguments**

- **conc**  
  Bioactive in vitro concentration in units of specified input.units, default of uM.

- **chem.name**  
  Either the chemical name or the CAS number must be specified.

- **chem.cas**  
  Either the CAS number or the chemical name must be specified.

- **dtxsid**  
  EPA’s ‘DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs

- **suppress.messages**  
  Suppress output messages.

- **which.quantile**  
  Which quantile from the SimCYP Monte Carlo simulation is requested. Can be a vector. Papers include 0.05, 0.5, and 0.95 for humans and 0.5 for rats.

- **species**  
  Species desired (either "Rat" or default "Human").

- **input.units**  
  Units of given concentration, default of uM but can also be mg/L.

- **output.units**  
  Units of dose, default of ‘mg’ for mg/kg BW/ day or ’mol’ for mol/kg BW/day.

- **clearance.assay.conc**  
  Concentration of chemical used in measuring intrinsic clearance data, 1 or 10 uM.

- **...**  
  Additional parameters passed to get_lit_css.

**Value**

Equivalent dose in specified units, default of mg/kg BW/day.
**get_physchem_param**

**Author(s)**

John Wambaugh

**References**


**Examples**

```r
table <- NULL
for(this.cas in sample(get_lit_cheminfo(),50)) table <- rbind(table,cbind(
as.data.frame(this.cas),as.data.frame(get_lit_oral_equiv(conc=1,chem.cas=this.cas)))))

get_lit_oral_equiv(0.1,chem.cas="34256-82-1")
get_lit_oral_equiv(0.1,chem.cas="34256-82-1",which.quantile=c(0.05,0.5,0.95))
```

### Description

This function retrieves physico-chemical properties ("param") for the chemical specified by chem.name or chem.cas from the vLiver tables.

### Usage

```r
get_physchem_param(param, chem.name = NULL, chem.cas = NULL, dtxsid = NULL)
```
get_rblood2plasma

Arguments

param The desired parameters, a vector or single value.
chem.name The chemical names that you want parameters for, a vector or single value.
chem.cas The chemical CAS numbers that you want parameters for, a vector or single value.
dtxsid EPA's `DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs.

Value

The parameters, either a single value, a named list for a single chemical, or a list of lists.

Author(s)

John Wambaugh and Robert Pearce

Examples

get_physchem_param(param = 'logP', chem.cas = '80-05-7')
get_physchem_param(param = c('logP', 'MW'), chem.cas = c('80-05-7', '81-81-2'))

description

This function attempts to retrieve a measured species- and chemical-specific blood:plasma concentration ratio.

Usage

get_rblood2plasma(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  species = "Human",
  default.to.human = FALSE
)

Arguments

chem.name Either the chemical name or the CAS number must be specified.
chem.cas Either the CAS number or the chemical name must be specified.
dtxsid EPA’s `DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs.
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
default.to.human Substitutes missing animal values with human values if true.
Details

A value of NA is returned when the requested value is unavailable. Values are retrieved from chem.physical_and_invitro.data. details than the description above.

Value

A numeric value for the steady-state ratio of chemical concentration in blood to plasma.

Author(s)

Robert Pearce

Examples

get_rblood2plasma(chem.name="Bisphenol A")
get_rblood2plasma(chem.name="Bisphenol A", species="Rat")

get_weight_class

Assign weight class (underweight, normal, overweight, obese)

Description

Given vectors of age, BMI, recumbent length, weight, and gender, categorizes weight classes using CDC and WHO categories.

Usage

get_weight_class(age_years, age_months, bmi, recumlen, weight, gender)

Arguments

age_years A vector of ages in years.
age_months A vector of ages in months.
bmi A vector of BMIs.
recumlen A vector of heights or recumbent lengths in cm.
weight A vector of body weights in kg.
gender A vector of genders (as 'Male' or 'Female').

Details

According to the CDC (https://www.cdc.gov/obesity/basics/adult-defining.html), adult weight classes are defined using BMI as follows:

Underweight BMI less than 18.5
Normal BMI between 18.5 and 25
Overweight BMI between 25 and 30
Obese BMI greater than 30
For children ages 2 years and older, weight classes are defined using percentiles of sex-specific BMI for age, as follows (Barlow et al., 2007):

- **Underweight**: Below 5th percentile BMI for age
- **Normal**: 5th-85th percentile BMI for age
- **Overweight**: 85th-95th percentile BMI for age
- **Obese**: Above 95th percentile BMI for age

For children birth to age 2, weight classes are defined using percentiles of sex-specific weight-for-length (Grummer-Strawn et al., 2009). Weight above the 97.7th percentile, or below the 2.3rd percentile, of weight-for-length is considered potentially indicative of adverse health conditions. Here, weight below the 2.3rd percentile is categorized as "Underweight" and weight above the 97.7th percentile is categorized as "Obese."

**Value**

A character vector of weight classes. Each element will be one of 'Underweight', 'Normal', 'Overweight', or 'Obese'.

**Author(s)**

Caroline Ring

**References**


**get_wetmore_css**

### Arguments

**info**  

**species**  
Species desired (either "Rat" or default "Human").

### Value

**info**  
Table/vector containing values specified in "info" for valid chemicals.

### Author(s)

John Wambaugh

### References


### Examples

```r
get_lit_cheminfo()
get_lit_cheminfo(info=c('CAS', 'MW'))
```

---

**Description**

This function is included for backward compatibility. It calls `get_lit_css` which retrieves a steady-state plasma concentration as a result of infusion dosing from the Wetmore et al. (2012) and (2013) publications and other literature.
get_wetmore_css

Usage

get_wetmore_css(
  chem.cas = NULL,
  chem.name = NULL,
  daily.dose = 1,
  which.quantile = 0.95,
  species = "Human",
  clearance.assay.conc = NULL,
  output.units = "mg/L",
  suppress.messages = FALSE
)

Arguments

chem.cas Either the cas number or the chemical name must be specified.
chem.name Either the chemical name or the CAS number must be specified.
daily.dose Total daily dose infused in units of mg/kg BW/day. Defaults to 1 mg/kg/day.
which.quantile Which quantile from the SimCYP Monte Carlo simulation is requested. Can be a vector.
species Species desired (either "Rat" or default "Human").
clearance.assay.conc Concentration of chemical used in measuring intrinsic clearance data, 1 or 10 uM.
output.units Returned units for function, defaults to mg/L but can also be uM (specify units = "uM").
suppress.messages Whether or not the output message is suppressed.

Value

A numeric vector with the literature steady-state plasma concentration (1 mg/kg/day) for the requested quantiles

Author(s)

John Wambaugh

References

get_wetmore_oral_equiv

Examples

```r
get_lit_css(chem.cas="34256-82-1")

get_lit_css(chem.cas="34256-82-1", species="Rat", which.quantile=0.5)

get_lit_css(chem.cas="80-05-7", daily.dose = 1,which.quantile = 0.5, output.units = "uM")
```

get_wetmore_oral_equiv

*Get Literature Oral Equivalent Dose (deprecated).*

Description

This function is included for backward compatibility. It calls `get_lit_oral_equiv` which converts a chemical plasma concentration to an oral equivalent dose using the values from the Wetmore et al. (2012) and (2013) publications and other literature.

Usage

```r
get_wetmore_oral_equiv(
  conc,
  chem.name = NULL,
  chem.cas = NULL,
  suppress.messages = FALSE,
  which.quantile = 0.95,
  species = "Human",
  input.units = "uM",
  output.units = "mg",
  clearance.assay.conc = NULL,
  ...
)
```

Arguments

- **conc** Bioactive in vitro concentration in units of specified `input.units`, default of `uM`.
- **chem.name** Either the chemical name or the CAS number must be specified.
- **chem.cas** Either the CAS number or the chemical name must be specified.
- **suppress.messages** Suppress output messages.
- **which.quantile** Which quantile from the SimCYP Monte Carlo simulation is requested. Can be a vector. Papers include 0.05, 0.5, and 0.95 for humans and 0.5 for rats.
- **species** Species desired (either "Rat" or default "Human").
- **input.units** Units of given concentration, default of `uM` but can also be `mg/L`.
- **output.units** Units of dose, default of `mg` for `mg/kg BW/ day` or `mol` for `mol/ kg BW/ day`.
- **clearance.assay.conc** Concentration of chemical used in measuring intrinsic clearance data, 1 or 10 uM.
- **...** Additional parameters passed to `get_lit_css`.
Value

Equivalent dose in specified units, default of mg/kg BW/day.

Author(s)

John Wambaugh

References


Examples

table <- NULL
for(this.cas in sample(get_lit_cheminfo(),50)) table <- rbind(table,cbind(
as.data.frame(this.cas),as.data.frame(get_lit_oral_equiv(conc=1,chem.cas=this.cas)))))

get_lit_oral_equiv(0.1,chem.cas="34256-82-1")

get_lit_oral_equiv(0.1,chem.cas="34256-82-1",which.quantile=c(0.05,0.5,0.95))

hct_h

KDE bandwidths for residual variability in hematocrit

Description

Bandwidths used for a one-dimensional kernel density estimation of the distribution of residual errors around smoothing spline fits of hematocrit vs. age for NHANES respondents in each of ten combinations of sex and race/ethnicity categories.

Usage

hct_h
Format

A named list with 10 elements, each a numeric value. Each list element corresponds to, and is named for, one combination of NHANES sex categories (Male and Female) and NHANES race/ethnicity categories (Mexican American, Other Hispanic, Non-Hispanic White, Non-Hispanic Black, and Other).

Details

Each matrix is the standard deviation for a normal distribution: this is the bandwidth to be used for a kernel density estimation (KDE) (using a normal kernel) of the distribution of residual errors around smoothing spline fits of hematocrit vs. age for NHANES respondents in the specified sex and race/ethnicity category. Optimal bandwidths were pre-calculated by doing the smoothing spline fits, getting the residuals, then calling \texttt{kde} on the residuals (which calls \texttt{hpi} to compute the plug-in bandwidth).

Used by HTTK-Pop only in "virtual individuals" mode (i.e. \texttt{httkpop_generate} with \texttt{method = "v"}), in \texttt{estimate_hematocrit}.

Author(s)

Caroline Ring

References


\begin{verbatim}
hematocrit_infants

Predict hematocrit in infants under 1 year old.

Usage

\texttt{hematocrit\_infants(age\_months)}

Arguments

\begin{itemize}
  \item \texttt{age\_months} Vector of ages in months; all must be <= 12.
\end{itemize}

Details

<table>
<thead>
<tr>
<th>Age</th>
<th>Reference range</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;1 month</td>
<td>31-49</td>
</tr>
<tr>
<td>1-6 months</td>
<td>29-42</td>
</tr>
<tr>
<td>7-12 months</td>
<td>33-38</td>
</tr>
</tbody>
</table>
\end{verbatim}
Value

Vector of hematocrit percentages corresponding to the input vector of ages.

Author(s)

Caroline Ring

References


honda.ivive

Return the assumptions used in Honda et al. 2019

Description

This function returns four of the better performing sets of assumptions evaluated in Honda et al. 2019 (https://doi.org/10.1371/journal.pone.0217564). These include four different combinations of hepatic clearance assumption, in vivo bioactivity assumption, and relevant tissue assumption. Generally, this function is not called directly by the user, but instead called by setting the IVIVE option in calc_mc_oral_equiv, calc_mc_css, and calc_analytic functions. Currently, these IVIVE option is not implemented the solve_1comp etc. functions.

Usage

honda.ivive(method = "Honda1", tissue = "liver")

Arguments

method

This is set to one of "Honda1", "Honda2", "Honda3", or "Honda4".

tissue

This is only relevant to "Honda4" and indicates the relevant tissue compartment.

Details

"Honda1" - tissue = NULL, restrictive.clearance = TRUE, bioactive.free.invivo = TRUE This assumption assumes restrictive hepatic clearance, and treats the free concentration in plasma as the bioactive concentration in vivo. This option must be used in combination with the concentration in vitro predicted by armitage_eval(), otherwise the result will be the same as "Honda2". This option corresponds to the result in Figure 8 panel c) restrictive, mean free plasma conc., Armitage in Honda et al. 2019. "Honda2" - tissue = NULL, restrictive.clearance = TRUE, bioactive.free.invivo = TRUE This assumption assumes restrictive hepatic clearance, and treats the free concentration in plasma as the bioactive concentration in vivo. This option corresponds to the result in Figure 8 panel b) restrictive, mean free plasma conc. in Honda et al. 2019. "Honda3" - tissue = NULL, restrictive.clearance = TRUE, bioactive.free.invivo = TRUE This assumption assumes restrictive hepatic clearance, and treats the free concentration in plasma as the bioactive concentration in vivo. This option corresponds to the result in Figure 8 panel a) restrictive, mean total plasma conc. in Honda et al. 2019. "Honda4" - tissue = tissue, restrictive.clearance = FALSE, bioactive.free.invivo = TRUE This assumption assumes restrictive hepatic clearance, and treats the free concentration in plasma as the bioactive concentration in vivo. The input tissue should be relevant to the in vitro assay endpoint used as input or that the result is being compared to. This option corresponds to the result in Figure 8 panel d) nonrestrictive, mean tissue conc. in Honda et al. 2019.
Value

A list of tissue, bioactive.free-invivo, and restrictive.clearance assumptions.

Author(s)

Greg Honda and John Wambaugh

References


Examples

honda.ivive(method = "Honda1", tissue = NULL)

Description

This data set is only used in Vignette 5.

Usage

howgate

Format

A data.table containing 24 rows and 11 columns.

Author(s)

Caroline Ring

References

The httkpop package generates virtual population physiologies for use in population TK.

Details

To simulate inter-individual variability in the TK model, a MC approach is used: the model parameters are sampled from known or assumed distributions, and the model is evaluated for each sampled set of parameters. To simulate variability across subpopulations, the MC approach needs to capture the parameter correlation structure. For example, kidney function changes with age (Levey et al., 2009), thus the distribution of GFR is likely different in 6-year-olds than in 65-year-olds. To directly measure the parameter correlation structure, all parameters need to be measured in each individual in a representative sample population. Such direct measurements are extremely limited. However, the correlation structure of the physiological parameters can be inferred from their known individual correlations with demographic and anthropometric quantities for which direct population measurements do exist. These quantities are sex, race/ethnicity, age, height, and weight (Howgate et al., 2006; Jamei et al., 2009a; Johnson et al., 2006; McNally et al., 2014; Price et al., 2003). Direct measurements of these quantities in a large, representative sample of the U.S. population are publicly available from NHANES. NHANES also includes laboratory measurements, including both serum creatinine, which can be used to estimate GFR (Levey et al., 2009), and hematocrit. For conciseness, sex, race/ethnicity, age, height, weight, serum creatinine, and hematocrit will be called the NHANES quantities.

HTTK-Pop’s correlated MC approach begins by sampling from the joint distribution of the NHANES quantities to simulate a population. Then, for each individual in the simulated population, HTTK-Pop predicts the physiological parameters from the NHANES quantities using regression equations from the literature (Barter et al., 2007; Baxter-Jones et al., 2011; Bosgra et al., 2012; Koo et al., 2000; Levey et al., 2009; Looker et al., 2013; McNally et al., 2014; Ogiu et al., 1997; Price et al., 2003; Schwartz and Work, 2009; Webber and Barr 2012). Correlations among the physiological parameters are induced by their mutual dependence on the correlated NHANES quantities. Finally, residual variability is added to the predicted physiological parameters using estimates of residual marginal variance (i.e., variance not explained by the regressions on the NHANES quantities) (McNally et al., 2014).

Data were combined from the three most recent publicly-available NHANES cycles: 2007-2008, 2009-2010, and 2011-2012. For each cycle, some NHANES quantities - height, weight, serum creatinine, and hematocrit - were measured only in a subset of respondents. Only these subsets were included in HTTK-Pop. The pooled subsets from the three cycles contained 29,353 unique respondents. Some respondents were excluded from analysis: those with age recorded as 80 years (because all NHANES respondents 80 years and older were marked as "80"); those with missing height, weight or hematocrit data; and those aged 12 years or older with missing serum creatinine data. These criteria excluded 4807 respondents, leaving 24,546 unique respondents. Each NHANES respondent was assigned a cycle-specific sample weight, which can be interpreted as the number of individuals in the total U.S. population represented by each NHANES respondent in each cycle (Johnson et al., 2013). Because data from three cycles were combined, the sample weights were rescaled (divided by the number of cycles being combined, as recommended in NHANES data analysis documentation) (Johnson et al., 2013). To handle the complex NHANES sampling structure, the R survey package was used to analyze the NHANES data (Lumley, 2004).
To allow generation of virtual populations specified by weight class, we coded a categorical variable for each NHANES respondent. The categories Underweight, Normal, Overweight, or Obese were assigned based on weight, age, and height/length (Grummer-Strawn et al., 2010; Kuczmarski et al., 2002; Ogden et al., 2014; WHO, 2006, 2010). We implemented two population simulation methods within HTTK-Pop: the direct-resampling method and the virtual-individuals method. The direct-resampling method simulated a population by sampling NHANES respondents with replacement, with probabilities proportional to the sample weights. Each individual in the resulting simulated population was an NHANES respondent, identified by a unique NHANES sequence number. By contrast, the second method generates "virtual individuals" - sets of NHANES quantities that obey the approximate joint distribution of the NHANES quantities (calculated using weighted smoothing functions and kernel density estimators), but do not necessarily correspond to any particular NHANES respondent. The direct-resampling method removed the possibility of generating unrealistic combinations of the NHANES quantities; the virtual-individuals method allowed the use of interpolation to simulate subpopulations represented by only a small number of NHANES respondents.

For either method, HTTK-Pop takes optional specifications about the population to be simulated and then samples from the appropriate conditional joint distribution of the NHANES quantities.

Once HTTK-Pop has simulated a population characterized by the NHANES quantities, the physiological parameters of the TK model are predicted from the NHANES quantities using regression equations from the literature. Liver mass was predicted for individuals over age 18 using allometric scaling with height from Reference Man (Valentin, 2002), and for individuals under 18 using regression relationships with height and weight published by Ogiu et al. (1997). Residual marginal variability was added for each individual as in PopGen (McNally et al., 2014). Similarly, hepatic portal vein blood flows (in L/h) are predicted as fixed fractions of a cardiac output allometrically scaled with height from Reference Man (Valentin, 2002), and residual marginal variability is added for each individual (McNally et al., 2014). Glomerular filtration rate (GFR) (in L/h/1.73 m2 body surface area) is predicted from age, race, sex, and serum creatinine using the CKD-EPI equation, for individuals over age 18 (Levey et al., 2009). For individuals under age 18, GFR is estimated from body surface area (BSA) (Johnson et al., 2006); BSA is predicted using Mosteller’s formula (Verbraecken et al., 2006) for adults and Haycock’s formula (Haycock et al., 1978) for children. Hepatocellularity (in millions of cells per gram of liver tissue) is predicted from age using an equation developed by Barter et al. (2007). Hematocrit is estimated from NHANES data for individuals 1 year and older. For individuals younger than 1 year, for whom NHANES did not measure hematocrit directly, hematocrit was predicted from age in months, using published reference ranges (Lubin, 1987).

In addition to the HTTK physiological parameters, the HTTK models include chemical-specific parameters representing the fraction of chemical unbound in plasma (Fup) and intrinsic clearance (CLint). Because these parameters represent interactions of the chemical with the body, their values will vary between individuals. To simulate this variability, Fub and CLint were included in MC simulations, by sampling from estimated or assumed distributions for the parameters defining them. Variability in hematocrit was simulated either using NHANES data (for individuals ages 1 and older) or using age-based reference ranges (for individuals under age 1). Fup was treated as a random variable obeying a distribution censored below the average limit of quantification (LOQ) of the in vitro assay. Specifically, Fup was assumed to obey a normal distribution truncated below at 0 and above at 1, centered at the Fup value measured in vitro, with a 30 the average LOQ (0.01), Fup was instead drawn from a uniform distribution between 0 and 0.01. Fup was assumed to be independent of all other parameters. This censored normal distribution was chosen to match that used in Wambaugh et al. (2015).

Variability in hepatocellularity (106 cells/g liver) and Mliver (kg) were simulated. The remaining source of variability in CLint,h is variability in CLint, which was simulated using a Gaussian mixture distribution to represent the population proportions of poor metabolizers (PMs) and non-PMs
of each substance. The true prevalence of PMs is isozyme-specific (Ma et al., 2002; Yasuda et al., 2008); however, isozyme-specific metabolism data were not available for the majority of chemicals considered. We therefore made a simplifying assumption that 5 slower than average. With 95 a normal distribution truncated below at zero, centered at the value measured in vitro, with a 
30 CLint was drawn from a PM distribution: a truncated normal distribution centered on one-tenth of the in vitro value with 30 Both CLint itself and the probability of being a PM were assumed to be independent of all other parameters. The truncated normal nonePM distribution was chosen because it has been used (with 100 in previous work (Rotroff et al., 2010; Wambaugh et al., 2015; Wetmore et al., 2014; Wetmore et al., 2015; Wetmore et al., 2012); the PM distribution was chosen to comport with the nonePM distribution.

**Main function to generate a population**

If you just want to generate a table of (chemical-independent) population physiology parameters, use `httkpop_generate`

**Using HTTK-Pop with HTTK**

To generate a population and then run an HTTK model for that population, the workflow is as follows:

1. Generate a population using `httkpop_generate`
2. For a given HTTK chemical and general model, convert the population data to corresponding sets of HTTK model parameters using `httkpop_mc`

**Author(s)**

Caroline Ring

**References**


httkpop_biotophys_default

Convert HTTK-Pop-generated parameters to HTTK physiological parameters

Description

Convert HTTK-Pop-generated parameters to HTTK physiological parameters

Usage

httkpop_biotophys_default(indiv_dt)

Arguments

indiv_dt The data.table object returned by httkpop_generate()

Value

A data.table with the physiological parameters expected by any HTTK model, including body weight (BW), hematocrit, tissue volumes per kg body weight, tissue flows as fraction of CO, CO per (kg BW)^3/4, GFR per (kg BW)^3/4, portal vein flow per (kg BW)^3/4, and liver density.

Author(s)

Caroline Ring
httkpop_direct_resample

Generate a virtual population by directly resampling the NHANES data.

Description

Generate a virtual population by directly resampling the NHANES data.

Usage

httkpop_direct_resample(
  nsamp = NULL,
  gendernum = NULL,
  agelim_years = NULL,
  agelim_months = NULL,
  weight_category = c("Underweight", "Normal", "Overweight", "Obese"),
  gfr_category = c("Normal", "Kidney Disease", "Kidney Failure"),
  reths = c("Mexican American", "Other Hispanic", "Non-Hispanic White", "Non-Hispanic Black", "Other"),
  gfr_resid_var = TRUE,
  ckd_epi_race_coeff = FALSE,
  nhanes_mec_svy
)

Arguments

nsamp The desired number of individuals in the virtual population. nsamp need not be provided if gendernum is provided.
gendernum Optional: A named list giving the numbers of male and female individuals to include in the population. e.g. list(Male=100,Female=100). Default is NULL, meaning both males and females are included, in their proportions in the NHANES data. If both nsamp and gendernum are provided, they must agree (i.e., nsamp must be the sum of gendernum).
agelim_years Optional: A two-element numeric vector giving the minimum and maximum ages (in years) to include in the population. Default is c(0,79). If agelim_years is provided and agelim_months is not, agelim_years will override the default value of agelim_months.
agelim_months Optional: A two-element numeric vector giving the minimum and maximum ages (in months) to include in the population. Default is c(0, 959), equivalent to the default agelim_years. If agelim_months is provided and agelim_years is not, agelim_months will override the default values of agelim_years.
weight_category Optional: The weight categories to include in the population. Default is c("Underweight", "Normal", "Overweight", "Obese"). User-supplied vector must contain one or more of these strings.

References

httkpop_direct_resample_inner

Inner loop function called by httkpop_direct_resample.

Description

Inner loop function called by httkpop_direct_resample.

Usage

httkpop_direct_resample_inner(
    nsamp,
    gendernum,
    agelim_months,
    agelim_years,
    reths,
    weight_category,
    gfr_resid_var,
    ckd_epi_race_coeff,
    nhanes_mec_svy
)
Arguments

nsamp  The desired number of individuals in the virtual population. nsamp need not be provided if gendernum is provided.

gendernum  Optional: A named list giving the numbers of male and female individuals to include in the population, e.g. list(Male=100,Female=100). Default is NULL, meaning both males and females are included, in their proportions in the NHANES data. If both nsamp and gendernum are provided, they must agree (i.e., nsamp must be the sum of gendernum).

agelim_months  Optional: A two-element numeric vector giving the minimum and maximum ages (in months) to include in the population. Default is c(0, 959), equivalent to the default agelim_years. If agelim_months is provided and agelim_years is not, agelim_months will override the default values of agelim_years.

agelim_years  Optional: A two-element numeric vector giving the minimum and maximum ages (in years) to include in the population. Default is c(0,79). If agelim_years is provided and agelim_months is not, agelim_years will override the default value of agelim_months.

reths  Optional: a character vector giving the races/ethnicities to include in the population. Default is c("Mexican American","Other Hispanic","Non-Hispanic White","Non-Hispanic Black","Other"), to include all races and ethnicities in their proportions in the NHANES data. User-supplied vector must contain one or more of these strings.

weight_category  Optional: The weight categories to include in the population. Default is c("Underweight","Normal","Overweight","Obese"). User-supplied vector must contain one or more of these strings.

gfr_resid_var  Logical value indicating whether or not to include residual variability when generating GFR values. (Default is TRUE, passed from 'httkpop_direct_resample'.)

ckd_epi_race_coeff  Logical value indicating whether or not to use the "race coefficient" from the CKD-EPI equation when estimating GFR values. (Default is FALSE, passed from 'httkpop_direct_resample'.)

nhanes_mec_svy  surveydesign object created from mecdt using svydesign (this is done in httkpop_generate)

Value

A data.table where each row represents an individual, and each column represents a demographic, anthropometric, or physiological parameter.

Author(s)

Caroline Ring

References

Generate a virtual population for PBTK

Description

Generate a virtual population characterized by demographic, anthropometric, and physiological parameters relevant to PBTK.

Usage

```
httkpop_generate(
  method,  # The population-generation method to use. Either "virtual individuals" or "direct resampling." Short names may be used: "d" or "dr" for "direct resampling", and "v" or "vi" for "virtual individuals".
  nsamp = NULL,  # The desired number of individuals in the virtual population. nsamp need not be provided if gendernum is provided.
  gendernum = NULL,  # Optional: A named list giving the numbers of male and female individuals to include in the population, e.g. list(Male=100,Female=100). Default is NULL, meaning both males and females are included, in their proportions in the NHANES data. If both nsamp and gendernum are provided, they must agree (i.e., nsamp must be the sum of gendernum).
  agelim_years = NULL,  # Optional: A two-element numeric vector giving the minimum and maximum ages (in years) to include in the population. Default is c(0,79). If only a single value is provided, both minimum and maximum ages will be set to that value; e.g. agelim_years=3 is equivalent to agelim_years=c(3,3). If agelim_years is provided and agelim_months is not, agelim_years will override the default value of agelim_months.
  agelim_months = NULL,  # Optional: A two-element numeric vector giving the minimum and maximum ages (in months) to include in the population. Default is c(0,959), equivalent to the default agelim_years. If only a single value is provided, both minimum and maximum ages will be set to that value; e.g. agelim_months=36 is equivalent to agelim_months=c(36,36). If agelim_months is provided and agelim_years is not, agelim_months will override the default values of agelim_years.
  weight_category = c("Underweight", "Normal", "Overweight", "Obese"),  # Optional: A vector of weight categories. Default is "Underweight".
  gfr_category = c("Normal", "Kidney Disease", "Kidney Failure"),  # Optional: A vector of GFR categories. Default is "Normal".
  reths = c("Mexican American", "Other Hispanic", "Non-Hispanic White", "Non-Hispanic Black", "Other"),  # Optional: A vector of racial/ethnic categories. Default is "Mexican American".
  gfr_resid_var = TRUE,  # Optional: A logical vector indicating whether to include random variation in GFR. Default is TRUE.
  ckd_epi_race_coeff = FALSE)  # Optional: A logical vector indicating whether to include race-specific coefficients for Ckd Epi. Default is FALSE.
```

Arguments

- **method**: The population-generation method to use. Either "virtual individuals" or "direct resampling." Short names may be used: "d" or "dr" for "direct resampling", and "v" or "vi" for "virtual individuals".
- **nsamp**: The desired number of individuals in the virtual population. nsamp need not be provided if gendernum is provided.
- **gendernum**: Optional: A named list giving the numbers of male and female individuals to include in the population, e.g. list(Male=100,Female=100). Default is NULL, meaning both males and females are included, in their proportions in the NHANES data. If both nsamp and gendernum are provided, they must agree (i.e., nsamp must be the sum of gendernum).
- **agelim_years**: Optional: A two-element numeric vector giving the minimum and maximum ages (in years) to include in the population. Default is c(0,79). If only a single value is provided, both minimum and maximum ages will be set to that value; e.g. agelim_years=3 is equivalent to agelim_years=c(3,3). If agelim_years is provided and agelim_months is not, agelim_years will override the default value of agelim_months.
- **agelim_months**: Optional: A two-element numeric vector giving the minimum and maximum ages (in months) to include in the population. Default is c(0,959), equivalent to the default agelim_years. If only a single value is provided, both minimum and maximum ages will be set to that value; e.g. agelim_months=36 is equivalent to agelim_months=c(36,36). If agelim_months is provided and agelim_years is not, agelim_months will override the default values of agelim_years.
weight_category
Optional: The weight categories to include in the population. Default is c('Underweight', 'Normal', 'Overweight', 'Obese'). User-supplied vector must contain one or more of these strings.

gfr_category
The kidney function categories to include in the population. Default is c('Normal', 'Kidney Disease', 'Kidney Failure') to include all kidney function levels.

reths
Optional: a character vector giving the races/ethnicities to include in the population. Default is c('Mexican American', 'Other Hispanic', 'Non-Hispanic White', 'Non-Hispanic Black', 'Other'), to include all races and ethnicities in their proportions in the NHANES data. User-supplied vector must contain one or more of these strings.

gfr_resid_var
TRUE to add residual variability to GFR predicted from serum creatinine; FALSE to not add residual variability

ckd_epi_race_coeff
TRUE to use the CKD-EPI equation as originally published (with a coefficient changing predicted GFR for individuals identified as "Non-Hispanic Black"); FALSE to set this coefficient to 1.

Details
Demographic and anthropometric (body measures) variables, along with serum creatinine and hematocrit, are generated from survey data from the Centers for Disease Control’s National Health and Nutrition Examination Survey (NHANES). Those data are stored in the object nhanes_mec_svy (a survey.design object, see package survey). With method = "d", these variables will be sampled with replacement directly from NHANES data. Each NHANES respondent’s likelihood of being sampled is given by their sample weight. With method = "v", these variables will be sampled from distributions fitted to NHANES data. Tissue masses and flows are generated based on demographic, body measures, and serum creatinine values, using regression equations from the literature and/or allometric scaling based on height. Extensive details about how each of these parameters are generated are available in the supplemental material of Ring et al. (2017) (see References for full citation).

Value
A data.table where each row represents an individual, and each column represents a demographic, anthropometric, or physiological parameter. Details of the parameters returned and their units are in the following tables.

Demographic variables

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>seqn</td>
<td>NHANES unique identifier (only included if method = &quot;direct resampling&quot;)</td>
<td></td>
</tr>
<tr>
<td>gender</td>
<td>Sex: &quot;Male&quot; or &quot;Female&quot;</td>
<td></td>
</tr>
<tr>
<td>reth</td>
<td>Race/ethnicity: &quot;Non-Hispanic Black&quot;, &quot;Non-Hispanic white&quot;, &quot;Mexican American&quot;, &quot;Other Hispanic&quot;, or &quot;Other&quot;</td>
<td></td>
</tr>
<tr>
<td>age_years</td>
<td>Age (0-79 years)</td>
<td>years</td>
</tr>
<tr>
<td>age_months</td>
<td>Age (0-959 months)</td>
<td>months</td>
</tr>
</tbody>
</table>
Body measures and laboratory measurements

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>height</td>
<td>Height</td>
<td>cm</td>
</tr>
<tr>
<td>weight</td>
<td>Body weight</td>
<td>kg</td>
</tr>
<tr>
<td>serum_creat</td>
<td>Serum creatinine</td>
<td>mg/dL</td>
</tr>
<tr>
<td>hematocrit</td>
<td>Hematocrit (percentage by volume of red blood cells in blood)</td>
<td>%</td>
</tr>
</tbody>
</table>

Tissue masses

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blood_mass</td>
<td>Mass of blood</td>
</tr>
<tr>
<td>Brain_mass</td>
<td>Mass of brain</td>
</tr>
<tr>
<td>Gonads_mass</td>
<td>Mass of gonads</td>
</tr>
<tr>
<td>Heart_mass</td>
<td>Mass of heart</td>
</tr>
<tr>
<td>Kidneys_mass</td>
<td>Mass of kidneys</td>
</tr>
<tr>
<td>Large_intestine_mass</td>
<td>Mass of large intestine</td>
</tr>
<tr>
<td>Liver_mass</td>
<td>Mass of liver</td>
</tr>
<tr>
<td>Lung_mass</td>
<td>Mass of lungs</td>
</tr>
<tr>
<td>Muscle_mass</td>
<td>Mass of skeletal muscle</td>
</tr>
<tr>
<td>Pancreas_mass</td>
<td>Mass of pancreas</td>
</tr>
<tr>
<td>Skeleton_mass</td>
<td>Mass of skeleton (including bone, red and yellow marrow, cartilage, periarticular tissue)</td>
</tr>
<tr>
<td>Skin_mass</td>
<td>Mass of skin</td>
</tr>
<tr>
<td>Small_intestine_mass</td>
<td>Mass of small intestine</td>
</tr>
<tr>
<td>Spleen_mass</td>
<td>Mass of spleen</td>
</tr>
<tr>
<td>Stomach_mass</td>
<td>Mass of stomach tissue</td>
</tr>
<tr>
<td>Other_mass</td>
<td>Mass of GI tract contents (1.4% of body weight) and tissues not otherwise enumerated (3.3% of body weight)</td>
</tr>
<tr>
<td>org_mass_sum</td>
<td>Sum of the above tissue masses. A check to ensure this is less than body weight.</td>
</tr>
<tr>
<td>Adipose_mass</td>
<td>Mass of adipose tissue. Assigned as weight - org_mass_sum.</td>
</tr>
</tbody>
</table>

Tissue flows

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adipose_flow</td>
<td>Blood flow to adipose tissue</td>
</tr>
<tr>
<td>Brain_flow</td>
<td>Blood flow to brain tissue</td>
</tr>
<tr>
<td>CO</td>
<td>Cardiac output</td>
</tr>
<tr>
<td>Gonads_flow</td>
<td>Blood flow to gonads tissue</td>
</tr>
<tr>
<td>Heart_flow</td>
<td>Blood flow to heart tissue</td>
</tr>
<tr>
<td>Kidneys_flow</td>
<td>Blood flow to kidneys tissue (not for glomerular filtration)</td>
</tr>
<tr>
<td>Large_intestine_flow</td>
<td>Blood flow to large intestine</td>
</tr>
<tr>
<td>Liver_flow</td>
<td>Blood flow to liver tissue</td>
</tr>
<tr>
<td>Lung_flow</td>
<td>Blood flow to lung tissue</td>
</tr>
<tr>
<td>Muscle_flow</td>
<td>Blood flow to skeletal muscle tissu</td>
</tr>
<tr>
<td>Pancreas_flow</td>
<td>Blood flow to pancreas tissue</td>
</tr>
<tr>
<td>Skeleton_flow</td>
<td>Blood flow to skeleton</td>
</tr>
<tr>
<td>Skin_flow</td>
<td>Blood flow to skin</td>
</tr>
<tr>
<td>Small_intestine_flow</td>
<td>Blood flow to small intestine</td>
</tr>
</tbody>
</table>
Spleen_flow  Blood flow to spleen L/h
Stomach_flow  Blood flow to stomach L/h
org_flow_check  Sum of blood flows as a fraction of cardiac output (CO). A check to make sure this is less than 1. Unitless fraction

Adjusted variables

<table>
<thead>
<tr>
<th>Name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight_adj</td>
<td>Adjusted body weight: Sum of all tissue masses. kg</td>
</tr>
<tr>
<td>BSA_adj</td>
<td>Adjusted body surface area, based on height and weight_adj. cm^2</td>
</tr>
<tr>
<td>million.cells.per.g.liver</td>
<td>Glomerular filtration rate (GFR) estimated using either the CKD-EPI equation</td>
</tr>
<tr>
<td>gfr_est</td>
<td></td>
</tr>
<tr>
<td>bmi_adj</td>
<td>Body mass index (BMI), adjusted to match weight_adj and height. kg/m^2</td>
</tr>
<tr>
<td>weight_class</td>
<td>Weight category based on bmi_adj: &quot;Underweight&quot; (BMI &lt; 18.5), &quot;Normal&quot; (18.5 &lt; BMI &lt; 24.9), &quot;Overweight&quot; (BMI &gt;= 25.0), &quot;Obese&quot; (BMI &gt;= 30)</td>
</tr>
<tr>
<td>gfr_class</td>
<td>Kidney function category based on GFR: &quot;Normal&quot; (GFR &gt;=60 mL/min/1.73 m^2), &quot;Kidney Disease&quot; (15 &lt;= GFR &lt;= 60), &quot;Kidney Failure&quot; (GFR &lt; 15)</td>
</tr>
</tbody>
</table>

Author(s)

Caroline Ring

References


Examples

#Simply generate a virtual population of 100 individuals, #using the direct-resampling method set.seed(42) httkpop_generate(method='direct resampling', nsamp=100) #Generate a population using the virtual-individuals method, #including 80 females and 20 males, #including only ages 20-65, #including only Mexican American and #Non-Hispanic Black individuals, #including only non-obese individuals httkpop_generate(method = 'virtual individuals', gendernum=list(Female=80, Male=20), agelim_years=c(20,65), reths=c('Mexican American', 'Non-Hispanic Black'), weight_category=c('Underweight', 'Normal', 'Overweight'))
**httkpop_mc**

**httk-pop: Correlated human physiological parameter Monte Carlo**

**Description**

This is the core function for httk-pop correlated human physiological variability simulation as described by Ring et al. (2017) ([doi:10.1016/j.envint.2017.06.004](https://doi.org/10.1016/j.envint.2017.06.004)). This function takes the data table of population biometrics (one individual per row) generated by `httkpop_generate`, and converts it to the corresponding table of HTTK model parameters for a specified HTTK model.

**Usage**

`httkpop_mc(model, samples = 1000, httkpop.dt = NULL, ...)`

**Arguments**

- `model` One of the HTTK models: "1compartment", "3compartmentss", "3compartment", or "pbtk".
- `samples` The number of Monte Carlo samples to use (can often think of these as separate individuals).
- `httkpop.dt` A data table generated by `httkpop_generate`. This defaults to NULL, in which case `httkpop_generate` is called to generate this table.
- `...` Additional arguments passed on to `httkpop_generate`.

**Details**

The Monte Carlo methods used here were recently updated and described by Breen et al. (submitted).

**Value**

A data.table with a row for each individual in the sample and a column for each parameter in the model.

**Author(s)**

Caroline Ring and John Wambaugh

**References**


**Examples**

```r
set.seed(42)
indiv_examp <- httkpop_generate(method="d", nsamp=100)
httk_param <- httkpop_mc(httkpop.dt=indiv_examp, model="1compartment")
```
httkpop_virtual_indiv  Generate a virtual population by the virtual individuals method.

Description
Generate a virtual population by the virtual individuals method.

Usage
httkpop_virtual_indiv(
  nsamp = NULL,
  gendernum = NULL,
  agelim_years = NULL,
  agelim_months = NULL,
  weight_category = c("Underweight", "Normal", "Overweight", "Obese"),
  gfr_category = c("Normal", "Kidney Disease", "Kidney Failure"),
  reths = c("Mexican American", "Other Hispanic", "Non-Hispanic White", "Non-Hispanic Black", "Other"),
  gfr_resid_var = TRUE,
  ckd_epi_race_coeff = FALSE,
  nhanes_mec_svy
)

Arguments
nsamp  The desired number of individuals in the virtual population. nsamp need not be provided if gendernum is provided.
gendernum  Optional: A named list giving the numbers of male and female individuals to include in the population, e.g. list(Male=100,Female=100). Default is NULL, meaning both males and females are included, in their proportions in the NHANES data. If both nsamp and gendernum are provided, they must agree (i.e., nsamp must be the sum of gendernum).
agelim_years  Optional: A two-element numeric vector giving the minimum and maximum ages (in years) to include in the population. Default is c(0,79). If agelim_years is provided and agelim_months is not, agelim_years will override the default value of agelim_months.
agelim_months  Optional: A two-element numeric vector giving the minimum and maximum ages (in months) to include in the population. Default is c(0, 959), equivalent to the default agelim_years. If agelim_months is provided and agelim_years is not, agelim_months will override the default values of agelim_years.
weight_category  Optional: The weight categories to include in the population. Default is c('Underweight', 'Normal', 'Overweight', 'Obese'). User-supplied vector must contain one or more of these strings.
gfr_category  The kidney function categories to include in the population. Default is c('Normal', 'Kidney Disease', 'Kidney Failure') to include all kidney function levels.
reths  Optional: a character vector giving the races/ethnicities to include in the population. Default is c('Mexican American', 'Other Hispanic', 'Non-Hispanic White', 'Non-Hispanic Black', 'Other'), to include all races and ethnicities.
in their proportions in the NHANES data. User-supplied vector must contain one or more of these strings.

gfr_resid_var Logical value indicating whether or not to include residual variability when generating GFR values. (Default is TRUE.)

ckd_epi_race_coeff Logical value indicating whether or not to use the "race coefficient" from the CKD-EPI equation when estimating GFR values. (Default is FALSE.)

nhanes_mec_svy surveydesign object created from mecdt using svydesign (this is done in httkpop_generate, which calls this function)

Value

A data.table where each row represents an individual, and each column represents a demographic, anthropometric, or physiological parameter.

Author(s)

Caroline Ring

References


---

**hw_H**

*KDE bandwidth for residual variability in height/weight*

Description

Bandwidths used for a two-dimensional kernel density estimation of the joint distribution of residual errors around smoothing spline fits of height vs. age and weight vs. age for NHANES respondents in each of ten combinations of sex and race/ethnicity categories.

Usage

```r
hw_H
```

Format

A named list with 10 elements, each a matrix with 2 rows and 2 columns. Each list element corresponds to, and is named for, one combination of NHANES sex categories (Male and Female) and NHANES race/ethnicity categories (Mexican American, Other Hispanic, Non-Hispanic White, Non-Hispanic Black, and Other).
Details

Each matrix is a variance-covariance matrix for a two-dimensional normal distribution: this is the bandwidth to be used for a two-dimensional kernel density estimation (KDE) (using a two-dimensional normal kernel) of the joint distribution of residual errors around smoothing spline fits of height vs. age and weight vs. age for NHANES respondents in the specified sex and race/ethnicity category. Optimal bandwidths were pre-calculated by doing the smoothing spline fits, getting the residuals, then calling `kde` on the residuals (which calls `hp1` to compute the plug-in bandwidth). Used by HTTK-Pop only in "virtual individuals" mode (i.e. `httkpop_generate` with `method = "v"”), in `gen_height_weight`.

Author(s)

Caroline Ring

References


in.list

Convenience Boolean (yes/no) functions to identify chemical membership in several key lists.

Description

These functions allow easy identification of whether or not a chemical CAS is included in various research projects. While it is our intent to keep these lists up-to-date, the information here is only for convenience and should not be considered to be definitive.

Usage

`in.list(chem.cas = NULL, which.list = "ToxCast")`

Arguments

chem.cas  The Chemical Abstracts Service Resgistry Number (CAS-RN) corresponding to the chemical of interest.

which.list  A character string that can take the following values: "ToxCast", "Tox21", "ExpoCast", "NHANES", "NHANES.serum.parent", "NHANES.serum.analyte","NHANES.blood.parent","NHANES.blood.analyte","NHANES.urine.parent","NHANES.urine.analyte"

Details

Tox21: Toxicology in the 21st Century (Tox21) is a U.S. federal High Throughput Screening (HTS) collaboration among EPA, NIH, including National Center for Advancing Translational Sciences and the National Toxicology Program at the National Institute of Environmental Health Sciences, and the Food and Drug Administration. (Bucher et al., 2008)

ToxCast: The Toxicity Forecaster (ToxCast) is a HTS screening project led by the U.S. EPA to perform additional testing of a subset of Tox21 chemicals. (Judson et al. 2010)
ExpoCast: ExpoCast (Exposure Forecaster) is an U.S. EPA research project to generate tentative exposure estimates (e.g., mg/kg BW/day) for thousands of chemicals that have little other information using models and informatics. (Wambaugh et al. 2014)

NHANES: The U.S. Centers for Disease Control (CDC) National Health and Nutrition Examination Survey (NHANES) is an on-going survey to characterize the health and biometrics (e.g., weight, height) of the U.S. population. One set of measurements includes the quantification of xenobiotic chemicals in various samples (blood, serum, urine) of the thousands of surveyed individuals. (CDC, 2014)

Value

logical A Boolean (1/0) value that is TRUE if the chemical is in the list.

Author(s)

John Wambaugh

References


See Also

is.httk for determining inclusion in httk project

Examples

```r
httk.table <- get_cheminfo(info=c("CAS","Compound"))
httk.table[,"Rat"] <- ""
httk.table[,"NHANES"] <- ""
httk.table[,"Tox21"] <- ""
httk.table[,"ToxCast"] <- ""
httk.table[,"ExpoCast"] <- ""
httk.table[,"PBTK"] <- ""

# To make this example run quickly, this loop is only over the first five # chemicals. To build a table with all available chemicals use:
# for (this.cas in httk.table$CAS)
for (this.cas in httk.table$CAS[1:5])
{
  this.index <- httk.table$CAS==this.cas
  if (is.nhanes(this.cas)) httk.table[this.index,"NHANES"] <- "Y" 
  if (is.tox21(this.cas)) httk.table[this.index,"Tox21"] <- "Y" 
  if (is.toxcast(this.cas)) httk.table[this.index,"ToxCast"] <- "Y" 
  if (is.expocast(this.cas)) httk.table[this.index,"ExpoCast"] <- "Y"
```
invitro_mc

Monte Carlo for in vitro toxicokinetic parameters including uncertainty and variability.

Description

Given a CAS in the HTTK data set, a virtual population from HTTK-Pop, some user specifications on the assumed distributions of \texttt{Funbound.plasma} and \texttt{Clint}, draw "individual" values of \texttt{Funbound.plasma} and \texttt{Clint} from those distributions. The methodology for this function was developed and described by Wambaugh et al. (2019) (doi:10.1093/toxsci/kfz205).

Usage

```r
invitro_mc(
  parameters.dt = NULL,
  samples,
  fup.meas.mc = TRUE,
  fup.pop.mc = TRUE,
  clint.meas.mc = TRUE,
  clint.pop.mc = TRUE,
  fup.meas.cv = 0.4,
  clint.meas.cv = 0.3,
  fup.pop.cv = 0.3,
  clint.pop.cv = 0.3,
  poormetab = TRUE,
  fup.lod = 0.01,
  fup.censored.dist = FALSE,
  adjusted.Funbound.plasma = TRUE,
  adjusted.Clint = TRUE,
  clint.pvalue.threshold = 0.05,
  minimum.Funbound.plasma = 1e-04
)
```

Arguments

- `parameters.dt`: A data table of physiological and chemical-specific parameters
- `samples`: The number of samples to draw.
- `fup.meas.mc`: Logical - should we perform measurement (uncertainty) Monte Carlo for \texttt{Funbound.plasma} values (Default TRUE). If FALSE, the user may choose to provide columns for "unadjusted.Funbound.plasma" or "fup.mean" from their own methods.
- `fup.pop.mc`: Logical - should we perform population (variability) Monte Carlo for \texttt{Funbound.plasma} values (Default TRUE)
- `clint.meas.mc`: Logical - should we perform measurement (uncertainty) Monte Carlo for \texttt{Clint} values (Default TRUE)
invitro_mc

clint.pop.mc Logical – should we perform population (variability) Monte Carlo for Clint values (Default TRUE)
fup.meas.cv Coefficient of variation of distribution of measured Funbound.plasma values.
clint.meas.cv Coefficient of variation of distribution of measured Clint values.
fup.pop.cv Coefficient of variation of distribution of population Funbound.plasma values.
clint.pop.cv Coefficient of variation of distribution of population Clint values.
poormetab Logical. Whether to include poor metabolizers in the Clint distribution or not.
fup.lod The average limit of detection for Funbound.plasma, below which distribution will be censored if fup.censored.dist is TRUE. Default 0.01.
fup.censored.dist Logical. Whether to draw Funbound.plasma from a censored distribution or not.
adjusted.Funbound.plasma Uses Pearce et al. (2017) lipid binding adjustment for Funbound.plasma when set to TRUE (Default).
adjusted.Clint Uses Kilford et al. (2008) hepatocyte incubation binding adjustment for Clint when set to TRUE (Default).
clint.pvalue.threshold Hepatic clearance for chemicals where the in vitro clearance assay result has a p-values greater than the threshold are set to zero.
minimum.Funbound.plasma Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).
parameters A list of chemical-specific model parameters containing at least Funbound.plasma, Clint, and Fhep.assay.correction.

Details

The Monte Carlo methods used here were recently updated and described by Breen et al. (submitted).

Value

A data.table with three columns: Funbound.plasma and Clint, containing the sampled values, and Fhep.assay.correction, containing the value for fraction unbound in hepatocyte assay.

Author(s)

Caroline Ring and John Wambaugh

References


Examples

```r
# Simply generate a virtual population of 100 individuals,
# using the direct-resampling method
set.seed(42)
# Pull mean vchemical=specific values:
chem.props <- parameterize_pbtk(chem.name="bisphenolaf")
# Convert to data.table with one row per sample:
parameters.dt <- monte_carlo(chem.props,samples=100)
# Use httk-pop to generate a population:
pop <- httkpop_generate(method='direct resampling', nsamp=100)
# Overwrite parameters specified by httk-pop:
parameters.dt[,names(pop):=pop]
# Vary in vitro parameters:
parameters.dt <- invitro_mc(parameters.dt,samples=100)
```

is.httk  

Convenience Boolean (yes/no) function to identify chemical membership and treatment within the httk project.

Description

Allows easy identification of whether or not a chemical CAS is included in various aspects of the httk research project (by model type and species of interest). While it is our intent to keep these lists up-to-date, the information here is only for convenience and should not be considered definitive.

Usage

```r
is.httk(chem.cas, species = "Human", model = "3compartmentss")
```

Arguments

- `chem.cas`: The Chemical Abstracts Service Registry Number (CAS-RN) corresponding to the chemical of interest.
- `species`: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- `model`: Model used in calculation, 'pbtk' for the multiple compartment model, '1compartment' for the one compartment model, '3compartment' for three compartment model, '3compartmentss' for the three compartment model without partition coefficients, or 'schmitt' for chemicals with logP and fraction unbound (used in predict_partitioning_schmitt).

Details

Tox21: Toxicology in the 21st Century (Tox21) is a U.S. federal High Throughput Screening (HTS) collaboration among EPA, NIH, including National Center for Advancing Translational Sciences and the National Toxicology Program at the National Institute of Environmental Health Sciences, and the Food and Drug Administration. (Bucher et al., 2008)

ToxCast: The Toxicity Forecaster (ToxCast) is a HTS screening project led by the U.S. EPA to perform additional testing of a subset of Tox21 chemicals. (Judson et al. 2010)
ExpoCast: ExpoCast (Exposure Forecaster) is an U.S. EPA research project to generate tentative exposure estimates (e.g., mg/kg BW/day) for thousands of chemicals that have little other information using models and informatics. (Wambaugh et al. 2014)

NHANES: The U.S. Centers for Disease Control (CDC) National Health and Nutrition Examination Survey (NHANES) is an on-going survey to characterize the health and biometrics (e.g., weight, height) of the U.S. population. One set of measurements includes the quantification of xenobiotic chemicals in various samples (blood, serum, urine) of the thousands of surveyed individuals. (CDC, 2014)

Value

A Boolean (1/0) value that is TRUE if the chemical is included in the httk project with a given modeling scheme (PBTK) and a given species

Author(s)

John Wambaugh

References


See Also

in.list for determining chemical membership in several other key lists

Examples

httk.table <- get_cheminfo(info=c("CAS","Compound"))
httk.table[,"Rat"] <- ""
httk.table[,"NHANES"] <- ""
httk.table[,"Tox21"] <- ""
httk.table[,"ToxCast"] <- ""
httk.table[,"ExpoCast"] <- ""
httk.table[,"PBTK"] <- ""

# To make this example run quickly, this loop is only over the first five chemicals. To build a table with all available chemicals use:
# for (this.cas in httk.table$CAS)
for (this.cas in httk.table$CAS[1:5])
{
this.index <- httk.table$CAS==this.cas
if (is.nhanes(this.cas)) httk.table[this.index,"NHANES"] <- "Y"
if (is.tox21(this.cas)) httk.table[this.index,"Tox21"] <- "Y"
if (is.toxcast(this.cas)) httk.table[this.index,"ToxCast"] <- "Y"
is_in_inclusive

Checks whether a value, or all values in a vector, is within inclusive limits

Description

Checks whether a value, or all values in a vector, is within inclusive limits

Usage

is_in_inclusive(x, lims)

Arguments

x
A numeric value, or vector of values.

lims
A two-element vector of (min, max) values for the inclusive limits. If x is a vector, lims may also be a two-column matrix with nrow=length(x) where the first column is lower limits and the second column is upper limits. If x is a vector and lims is a two-element vector, then each element of x will be checked against the same limits. If x is a vector and lims is a matrix, then each element of x will be checked against the limits given by the corresponding row of lims.

Value

A logical vector the same length as x, indicating whether each element of x is within the inclusive limits given by lims.

Author(s)

Caroline Ring

References

**johnson  Johnson 2006**

**Description**
This data set is only used in Vignette 5.

**Usage**
johnson

**Format**
A data.table containing 60 rows and 11 columns.

**Author(s)**
Caroline Ring

**References**

---

**kapraun2019  Kapraun et al. 2019 data**

**Description**
A list object containing time-varying parameters for the human maternal-fetal HTTK model. List elements contain scalar coefficients for the polynomial, logistic, Gompertz, and other functions of time describing blood flow rates, tissue volumes, hematocrits, and other anatomical/physiological quantities that change in the human mother and her fetus during pregnancy and gestation.

**Usage**
kapraun2019

**Format**
list

**Author(s)**
Dustin F. Kapraun

**Source**
Kapraun et al. 2019 Fetal PBTK Model
kidney_mass_children

References


kidney_mass_children  Predict kidney mass for children

Description

For individuals under age 18, predict kidney mass from weight, height, and gender. using equations from Ogiu et al. 1997

Usage

kidney_mass_children(weight, height, gender)

Arguments

weight  Vector of weights in kg.
height  Vector of heights in cm.
gender  Vector of genders (either 'Male' or 'Female').

Value

A vector of kidney masses in kg.

Author(s)

Caroline Ring

References


liver_mass_children  Predict liver mass for children

Description
For individuals under 18, predict the liver mass from height, weight, and gender, using equations from Ogiu et al. 1997

Usage
liver_mass_children(height, weight, gender)

Arguments
height  Vector of heights in cm.
weight  Vector of weights in kg.
gender  Vector of genders (either 'Male' or 'Female').

Value
A vector of liver masses in kg.

Author(s)
Caroline Ring

References


Description
This function returns an updated version of chem.physical_and_invitro.data that includes data predicted with Random Forest QSAR models developed and presented in Dawson et al. 2021, included in dawson2021.

Usage
load_dawson2021(overwrite = FALSE, exclude_oad = TRUE, target.env = .GlobalEnv)
Arguments

overwrite
Only matters if load.image=FALSE. If overwrite=TRUE then existing data in chem.physical_and_invitro.data will be replaced by any data/predictions in Dawson et al. (2021) that is for the same chemical and property. If overwrite=FALSE (DEFAULT) then new data for the same chemical and property are ignored. Fun-bound.plasma values of 0 (below limit of detection) are overwritten either way.

exclude_oad
Include the chemicals only within the applicability domain. If exclude_oad=TRUE (DEFAULT) chemicals outside the applicability domain do not have their predicted values loaded.

target.env
The environment where the new chem.physical_and_invitro.data is loaded. Defaults to global environment.

Value
data.frame
An updated version of chem.physical_and_invitro.data.

Author(s)
Sarah E. Davidson

References

Examples

```r
## Not run:
chem.physical_and_invitro.data <- load_dawson2021()
chem.physical_and_invitro.data <- load_dawson2021(overwrite=TRUE)

## End(Not run)
```

---

load_pradeep2020

**Load data from Pradeep et al. 2020.**

**Description**
This function returns an updated version of chem.physical_and_invitro.data that includes data predicted with Support Vector Machine and Random Forest models developed and presented in Pradeep et al. 2020, included in pradeep2020.

**Usage**

`load_pradeep2020(overwrite = FALSE, target.env = .GlobalEnv)`
load_sipes2017

Arguments

overwrite

Only matters if load.image=FALSE. If overwrite=TRUE then existing data in chem.physical_and_invitro.data will be replaced by any data/predictions in Pradeep et al. (2020) that is for the same chemical and property. If overwrite=FALSE (DEFAULT) then new data for the same chemical and property are ignored. Funbound.plasma values of 0 (below limit of detection) are overwritten either way.

target.env

The environment where the new chem.physical_and_invitro.data is loaded. Defaults to global environment.

Value

data.frame

An updated version of chem.physical_and_invitro.data.

Author(s)

Sarah E. Davidson

References


Examples

```r
## Not run:
chem.physical_and_invitro.data <- load_pradeep2020()
chem.physical_and_invitro.data <- load_pradeep2020(overwrite=TRUE)
## End(Not run)
```

load_sipes2017

Load data from Sipes et al 2017.

Description

This function returns an updated version of chem.physical_and_invitro.data that includes data predicted with Simulations Plus’ ADMET predictor that was used in Sipes et al. 2017, included in admet.data.

Usage

```r
load_sipes2017(overwrite = FALSE, target.env = .GlobalEnv)
```
Arguments

overwrite
Only matters if load.image=FALSE. If overwrite=TRUE then existing data in chem.physical_and_invitro.data will be replaced by any data/predictions in Sipes et al. (2017) that is for the same chemical and property. If overwrite=FALSE (DEFAULT) then new data for the same chemical and property are ignored. Funbound.plasma values of 0 (below limit of detection) are overwritten either way.

target.env
The environment where the new chem.physical_and_invitro.data is loaded. Defaults to global environment.

Value

data.frame
An updated version of chem.physical_and_invitro.data.

Author(s)
Robert Pearce and John Wambaugh

References


Examples

num.chems <- length(get_cheminfo())
load_sipes2017()
#We should have the ADMet Predicted chemicals from Sipes et al. (2017), #this one is a good test since the logP is nearly 10
calc_css(chem.cas="26040-51-7")
#Let's see how many chemicals we have now with the Sipes (2017) data loaded:
length(get_cheminfo())
#Now let us reset
reset_httk()
# We should be back to our original number:
num.chems == length(get_cheminfo())

---

**lump_tissues**  
**Lump tissue parameters**

Description

This function takes the parameters from predict_partitioning_schmitt and lumps the partition coefficients along with the volumes and flows based on the given tissue list. It is useful in Monte Carlo simulation of individual partition coefficients when calculating the rest of body partition coefficient.
lump_tissues

Usage

lump_tissues(
  Ktissue2pu.in,
  parameters = NULL,
  tissuelist = NULL,
  species = "Human",
  tissue.vols = NULL,
  tissue.flows = NULL,
  model = "pbtk",
  suppress.messages = FALSE
)

Arguments

Ktissue2pu.in List of partition coefficients from predict_partitioning_schmitt.
parameters A list of physiological parameters including flows and volumes for tissues in tissuelist
Tissuelist Manually specifies compartment names and tissues, which override the standard compartment names and tissues that are usually specified in a model's associated modelinfo file. Remaining tissues in the model's associated alltissues listing are lumped in the rest of the body.
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
tissue.vols A list of volumes for tissues in tissuelist
tissue.flows A list of flows for tissues in tissuelist
model Specify which model (and therefore which tissues) are being considered
suppress.messages Whether or not the output message is suppressed.

Details

This function returns the flows, volumes, and partition coefficients for the lumped tissues specified in tissue list Ktissue2plasma – tissue to free plasma concentration partition coefficients for every tissue specified by Schmitt (2008) (the tissue.data table) tissuelist – a list of character vectors, the name of each entry in the list is its own compartment. The tissues in the alltissues vector are the Schmitt (2008) tissues that are to be considered in the lumping process. The tissuelist can also be manually specified for alternate lumping schemes: for example, tissuelist<--list(Rapid=c("Brain","Kidney")) specifies the flow.col and vol.col in the tissuedata.table.

Value

Krbc2pu Ratio of concentration of chemical in red blood cells to unbound concentration in plasma.
Krest2pu Ratio of concentration of chemical in rest of body tissue to unbound concentration in plasma.
Vrestc Volume of the rest of the body per kg body weight, L/kg BW.
Vliverc Volume of the liver per kg body weight, L/kg BW.
Qtotal.liverf Fraction of cardiac output flowing to the gut and liver, i.e. out of the liver.
Qgutf Fraction of cardiac output flowing to the gut.
Qkidneyf Fraction of cardiac output flowing to the kidneys.
Author(s)

John Wambaugh and Robert Pearce

References


Examples

```r
pcs <- predict_partitioning_schmitt(chem.name='bisphenola')
tissuelist <- list(liver=c("liver"),kidney=c("kidney"),lung=c("lung"),gut=c("gut"),muscle.bone=c('muscle','bone'))

lump_tissues(pcs,tissuelist=tissuelist)
```

**lung_mass_children**

**Predict lung mass for children**

Description

For individuals under 18, predict the liver mass from height, weight, and gender, using equations from Ogiu et al. 1997

Usage

```
lung_mass_children(height, weight, gender)
```

Arguments

- height: Vector of heights in cm.
- weight: Vector of weights in kg.
- gender: Vector of genders (either 'Male' or 'Female').

Value

A vector of lung masses in kg.

Author(s)

Caroline Ring

References


mcnally_dt

Description
Reference tissue masses, flows, and residual variance distributions from Tables 1, 4, and 5 of McNally et al. 2014.

Usage
mcnally_dt

Format
A data.table with variables:

tissue  Body tissue
gender  Gender: Male or Female
mass_ref Reference mass in kg, from Reference Man
mass_cv Coefficient of variation for mass
mass_dist Distribution for mass: Normal or Log-normal
flow_ref Reference flow in L/h, from Reference Man
flow_cv Coefficient of variation for flow (all normally distributed)
height_ref Reference heights (by gender)
CO_ref Reference cardiac output by gender
flow_frac Fraction of CO flowing to each tissue: flow_ref/CO_ref

Author(s)
Caroline Ring

Source

References
**Description**

NHANES data on demographics, anthropometrics, and some laboratory measures, cleaned and combined into a single data set.

**Usage**

mecd

**Format**

A data.table with 23620 rows and 12 variables.

- **seqn** NHANES unique identifier for individual respondents.
- **riagendr** Gender: "Male" or "Female"
- **ridreth1** Race/ethnicity category: one of "Mexican American", "Non-Hispanic White", "Non-Hispanic Black", "Other", "Other Hispanic".
- **ridexagm** Age in months at the time of examination (if not recorded by NHANES, it was imputed based on age at the time of screening)
- **ridexagy** Age in years at the time of examination (if not recorded by NHANES, it was imputed based on age at the time of screening)
- **bmxwt** Weight in kg
- **lbxscr** Serum creatinine, mg/dL
- **lbxhct** Hematocrit, percent by volume of blood composed of red blood cells
- **wtmec6yr** 6-year sample weights for combining 3 cycles, computed by dividing 2-year sample weights by 3.
- **bmxhtlenavg** Average of height and recumbent length if both were measured; if only one was measured, takes value of the one that was measured.
- **weight_class** One of Underweight, Normal, Overweight, or Obese. Assigned using methods in `get_weight_class`.

**Author(s)**

Caroline Ring

**Source**


**References**

metabolism_data_Linakis2020

Metabolism data involved in Linakis 2020 vignette analysis.

Description
Metabolism data involved in Linakis 2020 vignette analysis.

Usage
metabolism_data_Linakis2020

Format
A data.frame containing x rows and y columns.

Author(s)
Matt Linakis

Source
Matt Linakis

References
DSStox database (https://www.epa.gov/nct/dsstox

monte_carlo

Monte Carlo for toxicokinetic model parameters

Description
This function performs basic, uncorrelated Monte Carlo to simulate uncertainty and/or variability for parameters of toxicokinetic models. Parameters can be varied according to either a normal distribution that is truncated at zero (using argument cv.params) or from a normal distribution that is censored for values less than the limit of detection (censored.params). Coefficient of variation (cv) and limit of of detectin can be specified separately for each parameter.

Usage
monte_carlo(
  parameters,
  cv.params = NULL,
  censored.params = NULL,
  samples = 1000
)
Arguments

- **parameters**: These parameters that are also listed in either cv.params or censored.params are sampled using Monte Carlo.
- **cv.params**: The parameters listed in cv.params are sampled from a normal distribution that is truncated at zero. This argument should be a list of coefficients of variation (cv) for the normal distribution. Each entry in the list is named for a parameter in "parameters". New values are sampled with mean equal to the value in "parameters" and standard deviation equal to the mean times the cv.
- **censored.params**: The parameters listed in censored.params are sampled from a normal distribution that is censored for values less than the limit of detection (specified separately for each parameter). This argument should be a list of sub-lists. Each sublist is named for a parameter in "params" and contains two elements: "cv" (coefficient of variation) and "LOD" (limit of detection), below which parameter values are censored. New values are sampled with mean equal to the value in "params" and standard deviation equal to the mean times the cv. Censored values are sampled on a uniform distribution between 0 and the limit of detection.
- **samples**: This argument is the number of samples to be generated for calculating quantiles.

Value

A data.table with a row for each individual in the sample and a column for each parameter in the model.

Author(s)

John Wambaugh

References


Examples

#Example based on Pearce et al. (2017):

# Set up means:
params <- parameterize_pbtk(chem.name="zoxamide")
# Nothing changes:
monte_carlo(params)

vary.params <- NULL
for (this.param in names(params)[!(names(params) %in% 
c("Funbound.plasma", "pKa_Donor", "pKa_Accept" )) & 
!is.na(as.numeric(params))]) vary.params[this.param] <- 0.2
# Most everything varies with CV of 0.2:
monte_carlo(
parameters=params, 
cv.params = vary.params)

censored.params <- list(Funbound.plasma = list(cv = 0.2, lod = 0.01))
# Fup is censored below 0.01:
monte_carlo(
parameters=params,
cv.params = vary.params,
censored.params = censored.params)

Obach2008  Published Pharmacokinetic Parameters from Obach et al. 2008

Description
This data set is used in Vignette 4 for steady state concentration.

Usage
Obach2008

Format
A data.frame containing 670 rows and 8 columns.

References

onlyp  NHANES Exposure Data

Description
This data set is only used in Vignette 6.

Usage
onlyp

Format
A data.table containing 1060 rows and 5 columns.

Author(s)
Caroline Ring

References
pancreas_mass_children

*Predict pancreas mass for children*

**Description**

For individuals under 18, predict the pancreas mass from height, weight, and gender, using equations from Ogiu et al.

**Usage**

```r
pancreas_mass_children(height, weight, gender)
```

**Arguments**

- **height**: Vector of heights in cm.
- **weight**: Vector of weights in kg.
- **gender**: Vector of genders (either 'Male' or 'Female').

**Value**

A vector of pancreas masses in kg.

**Author(s)**

Caroline Ring

**References**


---

parameterize_1comp

*Parameters for a one compartment (empirical) toxicokinetic model*

**Description**

This function initializes the parameters needed in the function solve_1comp. Volume of distribution is estimated by using a modified Schmitt (2008) method to predict tissue partition coefficients (Pearce et al., 2017) and then lumping the compartments weighted by tissue volume:
usage

```r
parameterize_1comp(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  species = "Human",
  default.to.human = FALSE,
  adjusted.Funbound.plasma = TRUE,
  adjusted.Clint = TRUE,
  regression = TRUE,
  restrictive.clearance = TRUE,
  well.stirred.correction = TRUE,
  suppress.messages = FALSE,
  clint.pvalue.threshold = 0.05,
  minimum.Funbound.plasma = 1e-04
)
```

arguments

c.chem.cas
  Chemical Abstract Services Registry Number (CAS-RN) – the chemical must be identified by either CAS, name, or DTXISD

c.chem.name
  Chemical name (spaces and capitalization ignored) – the chemical must be identified by either CAS, name, or DTXISD

dtxsid
  EPA’s DSSTox Structure ID (https://comptox.epa.gov/dashboard) – the chemical must be identified by either CAS, name, or DTXISDs

species
  Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
default.to.human
  Substitutes missing rat values with human values if true.

adjusted.Funbound.plasma
  Uses Pearce et al. (2017) lipid binding adjustment for Funbound.plasma (which impacts volume of distribution) when set to TRUE (Default).

adjusted.Clint
  Uses Kilford et al. (2008) hepatocyte incubation binding adjustment for Clint when set to TRUE (Default).

regression
  Whether or not to use the regressions in calculating partition coefficients in volume of distribution calculation.

restrictive.clearance
  In calculating elimination rate and hepatic bioavailability, protein binding is not taken into account (set to 1) in liver clearance if FALSE.

well.stirred.correction
  Uses correction in calculation of hepatic clearance for well-stirred model if TRUE. This assumes clearance relative to amount unbound in whole blood instead of plasma, but converted to use with plasma concentration.

suppress.messages
  Whether or not to suppress messages.

clint.pvalue.threshold
  Hepatic clearance for chemicals where the in vitro clearance assay result has a p-value greater than the threshold are set to zero.

minimum.Funbound.plasma
  Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).
Details

\[ V_{d, steady-state} = \sum_{i \in tissues} K_i V_i + V_{plasma} \]

where \( K_i \) is the tissue:unbound plasma concentration partition coefficient for tissue \( i \).

Value

- **Vdist**: Volume of distribution, units of \( \text{L/kg BW} \).
- **Fgutabs**: Fraction of the oral dose absorbed, i.e. the fraction of the dose that enters the gut lumen.
- **Fhep.assay.correction**: The fraction of chemical unbound in hepatocyte assay using the method of Kilford et al. (2008)
- **kelim**: Elimination rate, units of \( \text{1/h} \).
- **hematocrit**: Percent volume of red blood cells in the blood.
- **kgutabs**: Rate chemical is absorbed, \( \text{1/h} \).
- **million.cells.per.gliver**: Millions cells per gram of liver tissue.
- **MW**: Molecular Weight, \( \text{g/mol} \).
- **Rblood2plasma**: The ratio of the concentration of the chemical in the blood to the concentration in the plasma. Not used in calculations but included for the conversion of plasma outputs.
- **hepatic.bioavailability**: Fraction of dose remaining after first pass clearance, calculated from the corrected well-stirred model.
- **BW**: Body Weight, \( \text{kg} \).

Author(s)

John Wambaugh and Robert Pearce

References


Examples

```r
parameters <- parameterize_1comp(chem.name='Bisphenol-A', species='Rat')
parameters <- parameterize_1comp(chem.cas='80-05-7',
                                 restrictive.clearance=FALSE,
                                 species='rabbit',
                                 default.to.human=TRUE)
out <- solve_1comp(parameters=parameters)
```
Description

This function generates the chemical- and species-specific parameters needed for model ‘3compartment’, for example solve_3comp. A call is made to parameterize_pbtk to use Schmitt (2008)’s method as modified by Pearce et al. (2017) to predict partition coefficients based on descriptions in tissue.data. Organ volumes and flows are retrieved from table physiology.data.

Usage

parameterize_3comp(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  species = "Human",
  default.to.human = FALSE,
  force.human.clint.fup = FALSE,
  clint.pvalue.threshold = 0.05,
  adjusted.Funbound.plasma = TRUE,
  adjusted.Clint = TRUE,
  regression = TRUE,
  suppress.messages = FALSE,
  restrictive.clearance = TRUE,
  minimum.Funbound.plasma = 1e-04
)

Arguments

chem.cas Chemical Abstract Services Registry Number (CAS-RN) – the chemical must be identified by either CAS, name, or DTXISD
chem.name Chemical name (spaces and capitalization ignored) – the chemical must be identified by either CAS, name, or DTXISD
dtxsid EPA’s ’DSTox Structure ID (https://comptox.epa.gov/dashboard) – the chemical must be identified by either CAS, name, or DTXISD
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
default.to.human Substitutes missing animal values with human values if true.
force.human.clint.fup Forces use of human values for hepatic intrinsic clearance and fraction of unbound plasma if true.
clint.pvalue.threshold Hepatic clearances with clearance assays having p-values greater than the threshold are set to zero.
adjusted.Funbound.plasma Uses Pearce et al. (2017) lipid binding adjustment for Funbound.plasma (which impacts partition coefficients) when set to TRUE (Default).
adjusted.Clint  Uses Kilford et al. (2008) hepatocyte incubation binding adjustment for Clint when set to TRUE (Default).
regression  Whether or not to use the regressions in calculating partition coefficients.
suppress.messages  Whether or not the output message is suppressed.
restrictive.clearance  In calculating hepatic.bioavailability, protein binding is not taken into account (set to 1) in liver clearance if FALSE.
minimum.Funbound.plasma  Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

Value

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BW</td>
<td>Body Weight, kg.</td>
</tr>
<tr>
<td>Clmetabolismc</td>
<td>Hepatic Clearance, L/h/kg BW.</td>
</tr>
<tr>
<td>Fgutabs</td>
<td>Fraction of the oral dose absorbed, i.e. the fraction of the dose that enters the gutlumen.</td>
</tr>
<tr>
<td>Funbound.plasma</td>
<td>Fraction of plasma that is not bound.</td>
</tr>
<tr>
<td>Fhep.assay.correction</td>
<td>The fraction of chemical unbound in hepatocyte assay using the method of Kilford et al. (2008)</td>
</tr>
<tr>
<td>hematocrit</td>
<td>Percent volume of red blood cells in the blood.</td>
</tr>
<tr>
<td>Kgut2pu</td>
<td>Ratio of concentration of chemical in gut tissue to unbound concentration in plasma.</td>
</tr>
<tr>
<td>Kliver2pu</td>
<td>Ratio of concentration of chemical in liver tissue to unbound concentration in plasma.</td>
</tr>
<tr>
<td>Krbc2pu</td>
<td>Ratio of concentration of chemical in red blood cells to unbound concentration in plasma.</td>
</tr>
<tr>
<td>Krest2pu</td>
<td>Ratio of concentration of chemical in rest of body tissue to unbound concentration in plasma.</td>
</tr>
<tr>
<td>million.cells.per.gliver</td>
<td>Millions cells per gram of liver tissue.</td>
</tr>
<tr>
<td>MW</td>
<td>Molecular Weight, g/mol.</td>
</tr>
<tr>
<td>Qcardiacc</td>
<td>Cardiac Output, L/h/kg BW^3/4.</td>
</tr>
<tr>
<td>Qgfrc</td>
<td>Glomerular Filtration Rate, L/h/kg BW^3/4, volume of fluid filtered from kidney and excreted.</td>
</tr>
<tr>
<td>Qgutf</td>
<td>Fraction of cardiac output flowing to the gut.</td>
</tr>
<tr>
<td>Qliverf</td>
<td>Fraction of cardiac output flowing to the liver.</td>
</tr>
<tr>
<td>Rblood2plasma</td>
<td>The ratio of the concentration of the chemical in the blood to the concentration in the plasma.</td>
</tr>
<tr>
<td>Vgutc</td>
<td>Volume of the gut per kg body weight, L/kg BW.</td>
</tr>
<tr>
<td>Vliverc</td>
<td>Volume of the liver per kg body weight, L/kg BW.</td>
</tr>
<tr>
<td>Vrestc</td>
<td>Volume of the rest of the body per kg body weight, L/kg BW.</td>
</tr>
</tbody>
</table>
Author(s)

Robert Pearce and John Wambaugh

References


Examples

```r
parameters <- parameterize_3comp(chem.name='Bisphenol-A',species='Rat')
parameters <- parameterize_3comp(chem.cas='80-05-7',species='rabbit',default.to.human=TRUE)
out <- solve_3comp(parameters=parameters,plots=TRUE)
```

---

**parameterize_fetal_pbtk**

*Parameterize_fetal_PBTK*

**Description**

This function initializes the parameters needed in the functions solve_fetal_pbtk by calling solve_pbtk and adding additional parameters.

**Usage**

```r
parameterize_fetal_pbtk(
  chem.cas = NULL, 
  chem.name = NULL, 
  dtxsid = NULL, 
  species = "Human", 
  fetal_fup_adjustment = TRUE, 
  return.kapraun2019 = TRUE, 
  suppress.messages = FALSE, 
  ...
)
```
parameterize_fetal_pbtk

Arguments

c chem.cas  Either the chemical name or the CAS number must be specified.
c chem.name Either the chemical name or the CAS number must be specified.
d dtxsid  EPA's DSSTox Structure ID (http://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
c species  Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human"). Currently only a narrow human model is supported.

c fetal_fup_adjustment Logical indicator of whether to use an adjusted estimate for fetal fup based on the fetal:maternal plasma protein binding ratios presented in McNamara and Alcorn's 2002 study "Protein Binding Predictions in Infants." Defaults to TRUE.
c return.kapraun2019  If TRUE (default) the empirical parameters for the Kapraun et al. (2019) maternal-fetal growth parameters are provided.
c suppress.messages  Whether or not the output message is suppressed.
...  Arguments passed to parameterize_pbtk.

Value

c pre_pregnant_BW  Body Weight before pregnancy, kg.
c Clmetabolism  Hepatic Clearance, L/h/kg BW.
c Fgutabs  Fraction of the oral dose absorbed, i.e. the fraction of the dose that enters the gutlumen.
c Funbound.plasma  Fraction of plasma that is not bound.
c Fhep.assay.correction  The fraction of chemical unbound in hepatocyte assay using the method of Kilford et al. (2008)
c hematocrit  Percent volume of red blood cells in the blood.
c Kgut2pu  Ratio of concentration of chemical in gut tissue to unbound concentration in plasma.
c kgutabs  Rate that chemical enters the gut from gutlumen, 1/h.
c Kkidney2pu  Ratio of concentration of chemical in kidney tissue to unbound concentration in plasma.
c Kliver2pu  Ratio of concentration of chemical in liver tissue to unbound concentration in plasma.
c Klung2pu  Ratio of concentration of chemical in lung tissue to unbound concentration in plasma.
c Krbc2pu  Ratio of concentration of chemical in red blood cells to unbound concentration in plasma.
c Krest2pu  Ratio of concentration of chemical in rest of body tissue to unbound concentration in plasma.
c million.cells.per.gliver  Millions cells per gram of liver tissue.
c MW  Molecular Weight, g/mol.
**Qgfrc**  Glomerular Filtration Rate, L/h/kg BW^{3/4}, volume of fluid filtered from kidney and excreted.

**Rblood2plasma**  The ratio of the concentration of the chemical in the blood to the concentration in the plasma from available_rblood2plasma.

**Vgutc**  Volume of the gut per kg body weight, L/kg BW.

**Vkidneyc**  Volume of the kidneys per kg body weight, L/kg BW.

**Vliverc**  Volume of the liver per kg body weight, L/kg BW.

**Vlungc**  Volume of the lungs per kg body weight, L/kg BW.

**Vthyroidc**  Volume of the thyroid per kg body weight, L/kg BW.

**Kfgut2pu**  Ratio of concentration of chemical in fetal gut tissue to unbound concentration in plasma.

**Kfkidney2pu**  Ratio of concentration of chemical in fetal kidney tissue to unbound concentration in plasma.

**Kfliver2pu**  Ratio of concentration of chemical in fetal liver tissue to unbound concentration in plasma.

**Kflung2pu**  Ratio of concentration of chemical in fetal lung tissue to unbound concentration in plasma.

**Kfrest2pu**  Ratio of concentration of chemical in fetal rest of body tissue to unbound concentration in plasma.

**Kfbrain2pu**  Ratio of concentration of chemical in fetal brain tissue to unbound concentration in plasma.

**Kthyroid2pu**  Ratio of concentration of chemical in fetal thyroid tissue to unbound concentration in plasma.

**Kfthyroid2pu**  Ratio of concentration of chemical in fetal thyroid tissue to unbound concentration in plasma.

**Kplacenta2pu**  Ratio of concentration of chemical in placental tissue to unbound concentration in maternal plasma.

**Kfplacenta2pu**  Ratio of concentration of chemical in placental tissue to unbound concentration in fetal plasma.

**Author(s)**

Robert Pearce, Mark Sfeir, John Wambaugh, and Dustin Kapraun

Mark Sfeir, Dustin Kapraun, John Wambaugh

**References**


Examples

```r
parameters <- parameterize_fetal_pbtk(chem.cas='80-05-7')
parameters <- parameterize_fetal_pbtk(chem.name='Bisphenol-A',species='Rat')
```

Description

This function initializes the parameters needed for the model `gas_pbtk`, for example `solve_gas_pbtk`. Chemical- and species-specific model parameters are generated. These include tissue:plasma partition coefficients via Schmitt (2008)'s method as modified by Pearce et al. (2017). Organ volumes and flows are retrieved from table `physiology.data`. This model was first described by Linakis et al. (2020).

Usage

```r
parameterize_gas_pbtk(  
  chem.cas = NULL,  
  chem.name = NULL,  
  dtxsid = NULL,  
  species = "Human",  
  default.to.human = FALSE,  
  tissuelist = list(liver = c("liver"), kidney = c("kidney"), lung = c("lung"), gut = c("gut")),  
  force.human.clint.fup = FALSE,  
  clint.pvalue.threshold = 0.05,  
  adjusted.Funbound.plasma = TRUE,  
  adjusted.Clint = TRUE,  
  regression = TRUE,  
  vmax = 0,  
  km = 1,  
  exercise = FALSE,  
  fR = 12,  
  VT = 0.75,  
  VD = 0.15,  
  suppress.messages = FALSE,  
  minimum.Funbound.plasma = 1e-04,  
  ...  
)
```

Arguments

- `chem.cas`: Either the chemical name or the CAS number must be specified.
- `chem.name`: Either the chemical name or the CAS number must be specified.
parameterize_gas_pbtk

dtxsid  EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs

species  Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").

default.to.human  Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).

tissuelist  Specifies compartment names and tissues groupings. Remaining tissues in tissue.data are lumped in the rest of the body. However, solve_pbtk only works with the default parameters.

force.human.clinfup  Forces use of human values for hepatic intrinsic clearance and fraction of unbound plasma if true.

clint.pvalue.threshold  Hepatic clearance for chemicals where the in vitro clearance assay result has a p-values greater than the threshold are set to zero.

adjusted.Funbound.plasma  Uses Pearce et al. (2017) lipid binding adjustment for Funbound.plasma (which impacts partition coefficients) when set to TRUE (Default).

adjusted.Clint  Uses Kilford et al. (2008) hepatocyte incubation binding adjustment for Clint when set to TRUE (Default).

regression  Whether or not to use the regressions in calculating partition coefficients.

vmax  Michaelis-Menten vmax value in reactions/min

km  Michaelis-Menten concentration of half-maximal reaction velocity in desired output concentration units.

exercise  Logical indicator of whether to simulate an exercise-induced heightened respiration rate

fR  Respiratory frequency (breaths/minute), used especially to adjust breathing rate in the case of exercise. This parameter, along with VT and VD (below) gives another option for calculating Qalv (Alveolar ventilation) in case pulmonary ventilation rate is not known

VT  Tidal volume (L), to be modulated especially as part of simulating the state of exercise

VD  Anatomical dead space (L), to be modulated especially as part of simulating the state of exercise

suppress.messages  Whether or not the output message is suppressed.

minimum.Funbound.plasma  Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

...  Other parameters

**Value**

BW  Body Weight, kg.

Clint  Hepatic intrinsic clearance, uL/min/10^6 cells

Clint.dist  Distribution of hepatic intrinsic clearance values (median, lower 95th, upper 95th, p value)
Cimetabolismc  Hepatic Clearance, L/h/kg BW.
Fgutabs  Fraction of the oral dose absorbed, i.e. the fraction of the dose that enters the gut lumen.
Fhep.assay.correction  The fraction of chemical unbound in hepatocyte assay using the method of Kilford et al. (2008)
Funbound.plasma  Fraction of chemical unbound to plasma.
Funbound.plasma.adj  Fraction unbound to plasma adjusted as described in Pearce et al. 2017
Funbound.plasma.dist  Distribution of fraction unbound to plasma (median, lower 95th, upper 95th)
hematocrit  Percent volume of red blood cells in the blood.
Kblood2air  Ratio of concentration of chemical in blood to air
Kgut2pu  Ratio of concentration of chemical in gut tissue to unbound concentration in plasma.
kgutabs  Rate that chemical enters the gut from gut lumen, 1/h.
Kkidney2pu  Ratio of concentration of chemical in kidney tissue to unbound concentration in plasma.
Kliver2pu  Ratio of concentration of chemical in liver tissue to unbound concentration in plasma.
Klung2pu  Ratio of concentration of chemical in lung tissue to unbound concentration in plasma.
km  Michaelis-Menten concentration of half-maximal activity
Kmuc2air  Mucus to air partition coefficient
Krbc2pu  Ratio of concentration of chemical in red blood cells to unbound concentration in plasma.
Krest2pu  Ratio of concentration of chemical in rest of body tissue to unbound concentration in plasma.
kUrtc  Unscaled upper respiratory tract uptake parameter (L/h/kg^0.75)
liver.density  Density of liver in g/mL
MA  phospholipid:water distribution coefficient, membrane affinity
million.cells.per.gliver  Millions cells per gram of liver tissue.
MW  Molecular Weight, g/mol.
pKa_Accept  compound H association equilibrium constant(s)
pKa_Donor  compound H dissociation equilibrium constant(s)
Pow  octanol:water partition coefficient (not log transformed)
Qalvc  Unscaled alveolar ventilation rate (L/h/kg^0.75)
Qcardiac  Cardiac Output, L/h/kg BW^3/4.
Qgfrc  Glomerular Filtration Rate, L/h/kg BW^0.75, volume of fluid filtered from kidney and excreted.
Qgutf  Fraction of cardiac output flowing to the gut.
Qkidneyf  Fraction of cardiac output flowing to the kidneys.
**parameterize_gas_pbtk**

- **Qliverf**: Fraction of cardiac output flowing to the liver.
- **Qlungf**: Fraction of cardiac output flowing to lung tissue.
- **Qrestf**: Fraction of blood flow to rest of body.
- **Rblood2plasma**: The ratio of the concentration of the chemical in the blood to the concentration in the plasma from available rblood2plasma.
- **Vartc**: Volume of the arteries per kg body weight, L/kg BW.
- **Vgutc**: Volume of the gut per kg body weight, L/kg BW.
- **VkidneyC**: Volume of the kidneys per kg body weight, L/kg BW.
- **VliverC**: Volume of the liver per kg body weight, L/kg BW.
- **VlungC**: Volume of the lungs per kg body weight, L/kg BW.
- **vmax**: Michaelis-Menten maximum reaction velocity (1/min).
- **Vmucc**: Unscaled mucosal volume (L/kg BW^0.75).
- **VrestC**: Volume of the rest of the body per kg body weight, L/kg BW.
- **Vvenc**: Volume of the veins per kg body weight, L/kg BW.

**Author(s)**

Matt Linakis, Robert Pearce, John Wambaugh

**References**


**Examples**

```r
parameters <- parameterize_gas_pbtk(chem.cas='129-00-0')
parameters <- parameterize_gas_pbtk(chem.name='pyrene', species='Rat')
parameters <- parameterize_gas_pbtk(chem.cas = '56-23-5')
parameters <- parameterize_gas_pbtk(chem.name='Carbon tetrachloride', species='Rat')

# Change the tissue lumping:
compartments <- list(liver=c("liver"), fast=c("heart","brain","muscle","kidney"),
                     lung=c("lung"), gut=c("gut"), slow=c("bone"))
parameterize_gas_pbtk(chem.name="Bisphenol a", species="Rat", default.to.human=TRUE,
                      tissuelist=compartments)
```
parameterize_pbtk Parameters for a generic physiologically-based toxicokinetic model

Description

Generate a chemical- and species-specific set of model parameters, including tissue:plasma partition coefficients (via Schmitt (2008)'s method as modified by Pearce et al. (2017)) and organ volumes and flows (from table physiology.data) for an arbitrary tissue lumping scheme (tissues must be described in table tissue.data).

Usage

parameterize_pbtk(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  species = "Human",
  default.to.human = FALSE,
  tissuelist = list(liver = c("liver"), kidney = c("kidney"), lung = c("lung"), gut = c("gut")),
  force.human.clint.fup = FALSE,
  Clint.pvalue.threshold = 0.05,
  adjusted.Funbound.plasma = TRUE,
  adjusted.Clint = TRUE,
  regression = TRUE,
  suppress.messages = FALSE,
  restrictive.clearance = TRUE,
  minimum.Funbound.plasma = 1e-04
)

Arguments

- chem.cas: Chemical Abstract Services Registry Number (CAS-RN) – the chemical must be identified by either CAS, name, or DTXISD
- chem.name: Chemical name (spaces and capitalization ignored) – the chemical must be identified by either CAS, name, or DTXISD
- dtxsid: EPA's 'DSSTox Structure ID (https://comptox.epa.gov/dashboard) – the chemical must be identified by either CAS, name, or DTXISD
- species: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human")
- default.to.human: Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).
- tissuelist: Specifies compartment names and tissues groupings. Remaining tissues in tissue.data are lumped in the rest of the body. However, solve_pbtk only works with the default parameters.
- force.human.clint.fup: Forces use of human values for hepatic intrinsic clearance and fraction of unbound plasma if true.
**clint.pvalue.threshold**

Hepatic clearance for chemicals where the in vitro clearance assay result has a p-values greater than the threshold are set to zero.

**adjusted.Funbound.plasma**

Uses Pearce et al. (2017) lipid binding adjustment for Funbound.plasma (which impacts partition coefficients) when set to TRUE (Default).

**adjusted.Clint**

Uses Kilford et al. (2008) hepatocyte incubation binding adjustment for Clint when set to TRUE (Default).

**regression**

Whether or not to use the regressions in calculating partition coefficients.

**suppress.messages**

Whether or not the output message is suppressed.

**restrictive.clearance**

In calculating hepatic.bioavailability, protein binding is not taken into account (set to 1) in liver clearance if FALSE.

**minimum.Funbound.plasma**

Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

### Details

By default, this function initializes the parameters needed in the functions `solve_pbtk`, `calc_css`, and others using the httk default generic PBTK model (for oral and intravenous dosing only).

### Value

- **BW**
  
  Body Weight, kg.

- **Clmetabolismc**
  
  Hepatic Clearance, L/h/kg BW.

- **Fgutabs**
  
  Fraction of the oral dose absorbed, i.e. the fraction of the dose that enters the gutlumen.

- **Funbound.plasma**
  
  Fraction of plasma that is not bound.

- **Fhep.assay.correction**
  
  The fraction of chemical unbound in hepatocyte assay using the method of Kilford et al. (2008)

- **hematocrit**
  
  Percent volume of red blood cells in the blood.

- **Kgut2pu**
  
  Ratio of concentration of chemical in gut tissue to unbound concentration in plasma.

- **kgutabs**
  
  Rate that chemical enters the gut from gutlumen, l/h.

- **Kkidney2pu**
  
  Ratio of concentration of chemical in kidney tissue to unbound concentration in plasma.

- **Kliver2pu**
  
  Ratio of concentration of chemical in liver tissue to unbound concentration in plasma.

- **Klung2pu**
  
  Ratio of concentration of chemical in lung tissue to unbound concentration in plasma.

- **Krbc2pu**
  
  Ratio of concentration of chemical in red blood cells to unbound concentration in plasma.

- **Krest2pu**
  
  Ratio of concentration of chemical in rest of body tissue to unbound concentration in plasma.
million.cells.per.gliver

Millions cells per gram of liver tissue.

MW
Molecular Weight, g/mol.

Qcardiacc
Cardiac Output, L/h/kg BW^3/4.

Qgfrc
Glomerular Filtration Rate, L/h/kg BW^3/4, volume of fluid filtered from kidney and excreted.

Qgutf
Fraction of cardiac output flowing to the gut.

Qkidneyf
Fraction of cardiac output flowing to the kidneys.

Qliverf
Fraction of cardiac output flowing to the liver.

Rblood2plasma
The ratio of the concentration of the chemical in the blood to the concentration in the plasma from available_rblood2plasma.

Vartc
Volume of the arteries per kg body weight, L/kg BW.

Vgutc
Volume of the gut per kg body weight, L/kg BW.

Vkidneyc
Volume of the kidneys per kg body weight, L/kg BW.

Vliverc
Volume of the liver per kg body weight, L/kg BW.

Vlungc
Volume of the lungs per kg body weight, L/kg BW.

Vrestc
Volume of the rest of the body per kg body weight, L/kg BW.

Vvenc
Volume of the veins per kg body weight, L/kg BW.

Author(s)

John Wambaugh and Robert Pearce

References


Examples

```r
parameters <- parameterize_pbtk(chem.cas='80-05-7')

parameters <- parameterize_pbtk(chem.name='Bisphenol-A',species='Rat')

# Change the tissue lumping (note, these model parameters will not work with our current solver):
compartments <- list(liver=c("liver"),fast=c("heart","brain","muscle","kidney"),
                      lung=c("lung"),gut=c("gut"),slow=c("bone"))
parameterize_pbtk(chem.name="Bisphenol a",species="Rat",default.to.human=TRUE,
tissuelist=compartments)
```
Description

This function provides the necessary parameters to run `predict_partitioning_schmitt`, excluding the data in table `tissue.data`. The model is based on the Schmitt (2008) method for predicting tissue:plasma partition coefficients as modified by Pearce et al. (2017). The modifications include approaches adapted from Peyret et al. (2010).

Usage

```R
parameterize_schmitt(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  parameters = NULL,
  species = "Human",
  default.to.human = FALSE,
  force.human.fup = FALSE,
  adjusted.Funbound.plasma = TRUE,
  suppress.messages = FALSE,
  minimum.Funbound.plasma = 1e-04
)
```

Arguments

- `chem.cas`: Chemical Abstract Services Registry Number (CAS-RN) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
- `chem.name`: Chemical name (spaces and capitalization ignored) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISD
- `dtxsid`: EPA’s DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) – if parameters is not specified then the chemical must be identified by either CAS, name, or DTXISDs
- `parameters`: Chemical and physiological description parameters needed to run the Schmitt et al. (2008) model
- `species`: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- `default.to.human`: Substitutes missing fraction of unbound plasma with human values if true.
- `force.human.fup`: Returns human fraction of unbound plasma in calculation for rats if true. When species is specified as rabbit, dog, or mouse, the human unbound fraction is substituted.
- `adjusted.Funbound.plasma`: Uses Pearce et al. (2017) lipid binding adjustment for Funbound.plasma (which impacts partition coefficients) when set to TRUE (Default).
- `suppress.messages`: Whether or not the output message is suppressed.
- `minimum.Funbound.plasma`: Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).
**Value**

- `Funbound.plasma` Unbound fraction in plasma, adjusted for lipid binding according to Pearce et al. (2017)
- `unadjusted.Funbound.plasma` measured unbound fraction in plasma (0.005 if below limit of detection)
- `Pow` octanol:water partition coefficient (not log transformed)
- `pKa_Donor` compound H dissociation equilibrium constant(s)
- `pKa_Accept` compound H association equilibrium constant(s)
- `MA` phospholipid:water distribution coefficient, membrane affinity
- `Fprotein.plasma` protein fraction in plasma
- `plasma.pH` pH of the plasma

**Author(s)**

Robert Pearce and John Wambaugh

**References**


**Examples**

```r
parameterize_schmitt(chem.name='bisphenola')
```

**Description**

This function initializes the parameters needed in the functions `calc_mc_css`, `calc_mc_oral_equiv`, and `calc_analytic_css` for the three compartment steady state model ('3compartmentss') as used in Rotroff et al. (2010), Wetmore et al. (2012), Wetmore et al. (2015), and elsewhere. By assuming that enough time has passed to reach steady-state, we eliminate the need for tissue-specific partition coefficients because we assume all tissues have come to equilibrium with the unbound concentration in plasma. However, we still use chemical properties to predict the blood:plasma ratio for estimating first-pass hepatic metabolism for oral exposures.
Usage

```r
parameterize_steadystate(
  chem.cas = NULL,
  chem.name = NULL,
  dtxsid = NULL,
  species = "Human",
  clint.pvalue.threshold = 0.05,
  default.to.human = FALSE,
  human.clint.fup = FALSE,
  adjusted.Funbound.plasma = TRUE,
  adjusted.Clint = TRUE,
  restrictive.clearance = TRUE,
  fup.lod.default = 0.005,
  suppress.messages = FALSE,
  minimum.Funbound.plasma = 1e-04
)
```

Arguments

- **chem.cas**: Chemical Abstract Services Registry Number (CAS-RN) – the chemical must be identified by either CAS, name, or DTXISD
- **chem.name**: Chemical name (spaces and capitalization ignored) – the chemical must be identified by either CAS, name, or DTXISD
- **dtxsid**: EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) – the chemical must be identified by either CAS, name, or DTXISDs
- **species**: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- **clint.pvalue.threshold**: Hepatic clearances with clearance assays having p-values greater than the threshold are set to zero.
- **default.to.human**: Substitutes missing rat values with human values if true.
- **human.clint.fup**: Uses human hepatic intrinsic clearance and fraction of unbound plasma in calculation of partition coefficients for rats if true.
- **adjusted.Funbound.plasma**: Uses Pearce et al. (2017) lipid binding adjustment for Funbound.plasma (which impacts partition coefficients) when set to TRUE (Default).
- **adjusted.Clint**: Uses Kilford et al. (2008) hepatocyte incubation binding adjustment for Clint when set to TRUE (Default).
- **restrictive.clearance**: In calculating hepatic.bioavailability, protein binding is not taken into account (set to 1) in liver clearance if FALSE.
- **fup.lod.default**: Default value used for fraction of unbound plasma for chemicals where measured value was below the limit of detection. Default value is 0.0005.
- **suppress.messages**: Whether or not the output message is suppressed.
- **minimum.Funbound.plasma**: Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).
Value

- **Clint**: Hepatic Intrinsic Clearance, uL/min/10^6 cells.
- **Fgutabs**: Fraction of the oral dose absorbed, i.e. the fraction of the dose that enters the gut lumen.
- **Funbound.plasma**: Fraction of plasma that is not bound.
- **Qtotal.liverc**: Flow rate of blood exiting the liver, L/h/kg BW^{3/4}.
- **Qgfrc**: Glomerular Filtration Rate, L/h/kg BW^{3/4}, volume of fluid filtered from kidney and excreted.
- **BW**: Body Weight, kg
- **MW**: Molecular Weight, g/mol
- **million.cells.per.gliver**: Millions cells per gram of liver tissue.
- **Vliverc**: Volume of the liver per kg body weight, L/kg BW.
- **liver.density**: Liver tissue density, kg/L.
- **Fhep.assay.correction**: The fraction of chemical unbound in hepatocyte assay using the method of Kilford et al. (2008)
- **hepatic.bioavailability**: Fraction of dose remaining after first pass clearance, calculated from the corrected well-stirred model.

Author(s)

John Wambaugh

References


Examples

```r
parameters <- parameterize_steadystate(chem.name='Bisphenol-A', species='Rat')
parameters <- parameterize_steadystate(chem.cas='80-05-7')
```
Partition Coefficient Data

Description
Measured rat in vivo partition coefficients and data for predicting them.

Usage
pc.data

Format
A data.frame.

Author(s)
Jimena Davis and Robert Pearce

References
pearce2017regression  *Pearce et al. 2017 data*

**Description**

This table includes the adjusted and unadjusted regression parameter estimates for the chemical-specific plasma protein unbound fraction (fup) in 12 different tissue types.

**Usage**

`pearce2017regression`

**Format**

data.frame

**Details**

Predictions were made with regression models, as reported in Pearce et al. (2017).

**Author(s)**

Robert G. Pearce

**Source**

Pearce et al. 2017 Regression Models

**References**


---

pharma  *DRUGS|NORMAN: Pharmaceutical List with EU, Swiss, US Consumption Data*

**Description**

SWISSPHARMA is a list of pharmaceuticals with consumption data from Switzerland, France, Germany and the USA, used for a suspect screening/exposure modelling approach described in Singer et al 2016, DOI: 10.1021/acs.est.5b03332. The original data is available on the NORMAN Suspect List Exchange.

**Usage**

`pharma`

**Format**

An object of class `data.frame` with 954 rows and 14 columns.
physiology.data

Source

https://comptox.epa.gov/dashboard/chemical_lists/swisspharma

References


physiology.data  Species-specific physiology parameters

Description

This data set contains values from Davies and Morris (1993) necessary to parameterize a toxicokinetic model for human, mouse, rat, dog, or rabbit. The temperature for each species are taken from Robertshaw et al. (2004), Gordon (1993), and Stammers (1926).

Usage

physiology.data

Format

A data.frame containing 11 rows and 7 columns.

Author(s)

John Wambaugh and Nisha Sipes

Source


References


**pksim.pcs**  
*Partition Coefficients from PK-Sim*

**Description**  

**Usage**  
`pksim.pcs`

**Format**  
data.frame

**Source**  
Kapraun et al. 2021 (submitted)

**References**  

---

**pradeep2020**  
*Pradeep et al. 2020*

**Description**  
This table includes Support Vector Machine and Random Forest model predicted values for unbound fraction plasma protein (fup) and intrinsic hepatic clearance (clint) values for a subset of chemicals in the Tox21 library (see [https://www.epa.gov/chemical-research/toxicology-testing-21st-century](https://www.epa.gov/chemical-research/toxicology-testing-21st-century)).

**Usage**  
`pradeep2020`

**Format**  
data.frame

**Details**  
Prediction were made with Support Vector Machine and Random Forest models, as reported in Pradeep et al. (2020).
predict_partitioning_schmitt

Predict partition coefficients using the method from Schmitt (2008).

Description

This function implements the method from Schmitt (2008) in predicting the tissue to unbound plasma partition coefficients for the tissues contained in the tissue.data table.

Usage

predict_partitioning_schmitt(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  species = "Human",
  model = "pbtk",
  default.to.human = FALSE,
  parameters = NULL,
  alpha = 0.001,
  adjusted.Funbound.plasma = TRUE,
  regression = TRUE,
  regression.list = c("brain", "adipose", "gut", "heart", "kidney", "liver", "lung",
                     "muscle", "skin", "spleen", "bone"),
  tissues = NULL,
  minimum.Funbound.plasma = 1e-04,
  suppress.messages = FALSE
)

Arguments

chem.name Either the chemical name or the CAS number must be specified.
chem.cas Either the chemical name or the CAS number must be specified.
dtxsid EPA's DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
species Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
model Model for which partition coefficients are needed (for example, "pbtk", "3 compartment")

Source

Pradeep et al. 2020 Chemical Structure Predictive Models for HTTK

References

predict_partitioning_schmitt

default.to.human
Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).

parameters
Chemical parameters from parameterize_schmitt overrides chem.name, dtxsid, and chem.cas.

alpha
Ratio of Distribution coefficient D of totally charged species and that of the neutral form

adjusted.Funbound.plasma
Whether or not to use Funbound.plasma adjustment.

regression
Whether or not to use the regressions. Regressions are used by default.

regression.list
Tissues to use regressions on.

tissues
Vector of desired partition coefficients. Returns all by default.

minimum.Funbound.plasma
Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

suppress.messages
Whether or not the output message is suppressed.

Details
A separate regression is used when adjusted.Funbound.plasma is FALSE.
A regression is used for membrane affinity when not provided. The regressions for correcting each tissue are performed on tissue plasma partition coefficients (Ktissue2pu * Funbound.plasma) calculated with the corrected Funbound.plasma value and divided by this value to get Ktissue2pu. Thus the regressions should be used with the corrected Funbound.plasma.
The red blood cell regression can be used but is not by default because of the span of the data used, reducing confidence in the regression for higher and lower predicted values.
Human tissue volumes are used for species other than Rat.

Value
Returns tissue to unbound plasma partition coefficients for each tissue.

Author(s)
Robert Pearce

References
Examples

`predict_partitioning_schmitt(chem.name='ibuprofen',regression=FALSE)`

---

**Description**

Dallmann et al. (2018) includes compiled literature descriptions of toxicokinetic summary statistics, including time-integrated plasma concentrations (area under the curve or AUC) for drugs administered to a sample of subjects including both pregnant and non-pregnant women. The circumstances of the dosing varied slightly between drugs and are summarized in the table.

**Usage**

`pregnonpregaucs`

**Format**

data.frame

**Source**

Kapraun et al. 2021 (submitted)

**References**


---

**propagate_invitrouv_1comp**

*Propagates uncertainty and variability in in vitro HTTK data into one compartment model parameters*

**Description**

Propagates uncertainty and variability in in vitro HTTK data into one compartment model parameters

**Usage**

`propagate_invitrouv_1comp(parameters.dt, ...)`

**Arguments**

- `parameters.dt`: The data table of parameters being used by the Monte Carlo sampler
- `...`: Additional arguments passed to `calc_elimination_rate`
propagate_invitrouv_3comp

Propagates uncertainty and variability in in vitro HTTK data into three compartment model parameters

Description

Propagates uncertainty and variability in in vitro HTTK data into three compartment model parameters

Usage

propagate_invitrouv_3comp(parameters.dt, ...)

Arguments

parameters.dt The data table of parameters being used by the Monte Carlo sampler
...
... Additional arguments passed to calc_hep_clearance

Value

A data.table whose columns are the parameters of the HTTK model specified in model.

Author(s)

John Wambaugh

propagate_invitrouv_pbtk

Propagates uncertainty and variability in in vitro HTTK data into PBPK model parameters

Description

Propagates uncertainty and variability in in vitro HTTK data into PBPK model parameters

Usage

propagate_invitrouv_pbtk(parameters.dt, ...)

Arguments

parameters.dt The data table of parameters being used by the Monte Carlo sampler
...
... Additional arguments passed to calc_hep_clearance

Value

A data.table whose columns are the parameters of the HTTK model specified in model.

Author(s)

John Wambaugh
**reset_httk**

Arguments

- **parameters.dt**  The data table of parameters being used by the Monte Carlo sampler
- ...  Additional arguments passed to `calc_hep_clearance`

Value

A data.table whose columns are the parameters of the HTTK model specified in `model`.

Author(s)

John Wambaugh

---

**reset_httk**  *Reset HTTK to Default Data Tables*

Description

This function returns an updated version of `chem.physical_and_invitro.data` that includes data predicted with Simulations Plus’ ADMET predictor that was used in Sipes et al. 2017, included in `admet.data`.

Usage

```r
reset_httk(target.env = .GlobalEnv)
```

Arguments

- **target.env**  The environment where the new `chem.physical_and_invitro.data` is loaded. Defaults to global environment.

Value

- **data.frame**  The package default version of `chem.physical_and_invitro.data`.

Author(s)

John Wambaugh

Examples

```r
chem.physical_and_invitro.data <- load_sipes2017()
reset_httk()
```
rfun

**Randomly draws from a one-dimensional KDE**

**Description**
Randomly draws from a one-dimensional KDE

**Usage**
```
rfun(n, fhat)
```

**Arguments**
- `n`: Number of samples to draw
- `fhat`: A list with elements `x`, `w`, and `h` (h is the KDE bandwidth).

**Value**
A vector of n samples from the KDE `fhat`

**Author(s)**
Caroline Ring

**References**

rmed0non0u95

**Draw random numbers with LOD median but non-zero upper 95th percentile**

**Description**
This function draws N random numbers from a distribution that approximates a median that is equal to the limit of detection (LOD, value `x.LOD`) but has an upper 95th percentile (`x.u95`) that is above `x.LOD`. We make the assumption that values above `x.u95` are uniformly distributed between `x.u95` and `x.u95 + (x.u95 - x.LOD)`

**Usage**
```
rmed0non0u95(n, x.u95, x.min = 0, x.LOD = 0.005)
```

**Arguments**
- `n`: Number of samples to draw
- `x.u95`: The upper limit on the 95th confidence/credible interval (this is the 97.5 percentile)
- `x.min`: The minimum allowed value (defaults to 0)
- `x.LOD`: The limit of detection (defaults to 0.005)
**r_left_censored_norm**

Returns draws from a normal distribution with a lower censoring limit of lod (limit of detection)

---

**Value**

A vector of samples where the 50th and 97.5th quantiles approximate x.LOD and x.u95 respectively.

**Author(s)**

John Wambaugh

**References**

Breen et al., in preparation

**Examples**

```r
Fup.95 <- 0.02
N <- 1000

set.seed(1235)
Fup.vec <- rmed0non0u95(n=N, x.u95=Fup.95)
quantile(Fup.vec,c(0.5,0.975))

quantile(rmed0non0u95(200,x.u95=0.05,x.min=10^-4,x.LOD=0.01),c(0.5,0.975))
hist(rmed0non0u95(1000,x.u95=0.05,x.min=10^-4,x.LOD=0.01))

quantile(rmed0non0u95(200,x.u95=0.005,x.min=10^-4,x.LOD=0.01),c(0.5,0.975))
hist(rmed0non0u95(1000,x.u95=0.005,x.min=10^-4,x.LOD=0.01))
```

---

**Description**

Returns draws from a normal distribution with a lower censoring limit of lod (limit of detection)

**Usage**

```r
r_left_censored_norm(n, mean = 0, sd = 1, lod = 0.005, lower = 0, upper = 1)
```

**Arguments**

- `n`: Number of samples to take
- `mean`: Mean of censored distribution. Default `0`
- `sd`: Standard deviation of censored distribution. Default `1`
- `lod`: Bound below which to censor. Default `0.005`
- `lower`: Lower bound on censored distribution. Default `0`
- `upper`: Upper bound on censored distribution. Default `1`

**Value**

A vector of samples from the specified censored distribution.
scale_dosing  

Scale mg/kg body weight doses according to body weight and units

Description

This function transforms the dose (in mg/kg) into the appropriate units. It handles single doses, matrices of doses, or daily repeated doses at varying intervals. Gut absorption is also factored in through the parameter Fgutabs, and scaling is currently avoided in the inhalation exposure case with a scale factor of 1.

Usage

scale_dosing(
  dosing,
  parameters,
  route,
  input.units = NULL,
  output.units = "uM",
  vol = NULL
)

Arguments

dosing  List of dosing metrics used in simulation, which must include the general entries with names "initial.dose", "doses.per.day", "daily.dose", and "dosing.matrix". The "dosing.matrix" is used for more precise dose regimen specification, and is a matrix consisting of two columns or rows named "time" and "dose" containing the time and amount, in mg/kg BW, of each dose. The minimal usage case involves all entries but "initial.dose" set to NULL in value.

parameters Chemical parameters from parameterize_pbtk function, overrides chem.name and chem.cas.

route  String specification of route of exposure for simulation: "oral", "iv", "inhalation", ...

input.units Units of the dose values being scaled. (Default is NULL.) Currently supported units "mg/L", "ug/L", "ug/mL", "uM", "umol/L", "ug/dL", "ug/g", "nmol/L", "nM", and "ppmw" (supported input.units subject to change).

output.units Desired units (either "mg/L", "mg", "umol", or default "uM").

vol  Volume for the target tissue of interest. NOTE: Volume should not be in units of per BW, i.e. "kg".

Value

A list of numeric values for doses converted to output.units, potentially (depending on argument dosing) including:

- initial.dose  The first dose given
- dosing.matrix A 2xN matrix where the first column is dose time and the second is dose amount for N doses
- daily.dose  The total cumulative daily dose
**scr_h**

**Author(s)**

John Wambaugh and Sarah E. Davidson

---

**KDE bandwidths for residual variability in serum creatinine**

**Description**

Bandwidths used for a one-dimensional kernel density estimation of the distribution of residual errors around smoothing spline fits of serum creatinine vs. age for NHANES respondents in each of ten combinations of sex and race/ethnicity categories.

**Usage**

```r
scr_h
```

**Format**

A named list with 10 elements, each a numeric value. Each list element corresponds to, and is named for, one combination of NHANES sex categories (Male and Female) and NHANES race/ethnicity categories (Mexican American, Other Hispanic, Non-Hispanic White, Non-Hispanic Black, and Other).

**Details**

Each matrix is the standard deviation for a normal distribution: this is the bandwidth to be used for a kernel density estimation (KDE) (using a normal kernel) of the distribution of residual errors around smoothing spline fits of serum creatinine vs. age for NHANES respondents in the specified sex and race/ethnicity category. Optimal bandwidths were pre-calculated by doing the smoothing spline fits, getting the residuals, then calling `kde` on the residuals (which calls `hpi` to compute the plug-in bandwidth).

Used by HTTK-Pop only in "virtual individuals" mode (i.e. `httkpop_generate` with `method = "v"`), in `gen_serum_creatinine`.

**Author(s)**

Caroline Ring

**References**

set_httk_precision

Description

Although the ODE solver and other functions return very precise numbers, we cannot (or at least
do not spend enough computing time to) be sure of the precision to an arbitrary level. This function
both limits the number of significant figures reported and truncates the numerical precision.

Usage

set_httk_precision(in.num, sig.fig = 4, num.prec = 9)

Arguments

in.num The numeric variable (or assembly of numerics) to be processed.
sig.fig The number of significant figures reported. Defaults to 4.
um.prec The precision maintained, digits below 10^num.prec are dropped. Defaults to 9.

Value

numeric values

Author(s)

John Wambaugh

skeletal_muscle_mass

Description

Predict skeletal muscle mass from age, height, and gender.

Usage

skeletal_muscle_mass(smm, age_years, height, gender)

Arguments

smm Vector of allometrically-scaled skeletal muscle masses.
age_years Vector of ages in years.
height Vector of heights in cm.
gender Vector of genders, either 'Male' or 'Female.'

Details

For individuals over age 18, use allometrically-scaled muscle mass with an age-based scaling factor,
to account for loss of muscle mass with age (Janssen et al. 2000). For individuals under age 18, use
skeletal_muscle_mass_children.
skeletal_muscle_mass_children

Value
Vector of skeletal muscle masses in kg.

Author(s)
Caroline Ring

References

See Also
skeletal_muscle_mass_children

Predict skeletal muscle mass for children

Description
For individuals under age 18, predict skeletal muscle mass from gender and age, using a nonlinear equation from Webber and Barr (2012)

Usage
skeletal_muscle_mass_children(gender, age_years)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gender</td>
<td>Vector of genders (either 'Male' or 'Female').</td>
</tr>
<tr>
<td>age_years</td>
<td>Vector of ages in years.</td>
</tr>
</tbody>
</table>

Value
Vector of skeletal muscle masses in kg.

Author(s)
Caroline Ring

References
### skin_mass_bosgra

**Predict skin mass**

**Description**

Using equation from Bosgra et al. 2012, predict skin mass from body surface area.

**Usage**

```r
skin_mass_bosgra(BSA)
```

**Arguments**

- **BSA**: Vector of body surface areas in cm².

**Value**

Vector of skin masses in kg.

**Author(s)**

Caroline Ring

**References**


---

### solve_1comp

**Solve one compartment TK model**

**Description**

This function solves for the amount or concentration of a chemical in plasma for a one compartment model as a function of time based on the dose and dosing frequency.

**Usage**

```r
solve_1comp(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  times = NULL,
  parameters = NULL,
  days = 10,
  tsteps = 4,
  daily.dose = NULL,
)```

solve_1comp
dose = NULL,
doses.per.day = NULL,
initial.values = NULL,
plots = FALSE,
suppress.messages = FALSE,
species = "Human",
iv.dose = FALSE,
input.units = "mg/kg",
output.units = NULL,
method = "lsoda",
rtol = 1e-08,
atol = 1e-12,
default.to.human = FALSE,
recalc.blood2plasma = FALSE,
recalc.clearance = FALSE,
dosing.matrix = NULL,
adjusted.Funbound.plasma = TRUE,
regression = TRUE,
restrictive.clearance = TRUE,
minimum.Funbound.plasma = 1e-04,
monitor.vars = NULL,
... )

Arguments

chem.name Either the chemical name, CAS number, or the parameters must be specified.
chem.cas Either the chemical name, CAS number, or the parameters must be specified.
dtxsid EPA’s DSSTox Structure ID (https://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
times Optional time sequence for specified number of days.
parameters Chemical parameters from parameterize_1comp function, overrides chem.name and chem.cas.
days Length of the simulation.
tsteps The number time steps per hour.
daily.dose Total daily dose, default is mg/kg BW.
dose Amount of a single dose, default is mg/kg BW.
doses.per.day Number of doses per day.
initial.values Vector containing the initial concentrations or amounts of the chemical in specified tissues with units corresponding to output.units. Defaults are zero.
plots Plots all outputs if true.
suppress.messages Whether or not the output message is suppressed.
species Species desired (either "Rat", "Rabbit", "Dog", or default "Human").
iv.dose Simulates a single i.v. dose if true.
input.units Input units of interest assigned to dosing, defaults to "mg/kg" BW.
output.units A named vector of output units expected for the model results. Default, NULL, returns model results in units specified in the ‘modelinfo’ file. See table below for details.
solve_1comp

method  Method used by integrator (deSolve).
rtol    Argument passed to integrator (deSolve).
atol    Argument passed to integrator (deSolve).
default.to.human
    Substitutes missing rat values with human values if true.
recalc.blood2plasma
    Whether or not to recalculate the blood:plasma chemical concentration ratio
recalc.clearance
    Whether or not to recalculate the elimination rate.
dosing.matrix
    Vector of dosing times or a matrix consisting of two columns or rows named
    "dose" and "time" containing the time and amount, in mg/kg BW by default, of
    each dose.
adjusted.Funbound.plasma
    Uses adjusted Funbound.plasma when set to TRUE along with volume of distri-
    bution calculated with this value.
regression
    Whether or not to use the regressions in calculating partition coefficients in vol-
    ume of distribution calculation.
restrictive.clearance
    In calculating elimination rate, protein binding is not taken into account (set to
    1) in liver clearance if FALSE.
minimum.Funbound.plasma
    Monte Carlo draws less than this value are set equal to this value (default is
    0.0001 – half the lowest measured Fup in our dataset).
monitor.vars
    Which variables are returned as a function of time. Defaults value of NULL
    provides "Agutlumen", "Ccompartent", "Ametabolized", "AUC"
...  Additional arguments passed to the integrator.

Details

Note that the model parameters have units of hours while the model output is in days.

Default value of NULL for doses.per.day solves for a single dose.

When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological
data(volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic
hepatic clearance.

AUC is area under plasma concentration curve.
Value

A matrix with a column for time (in days) and a column for the compartment and the area under the curve (concentration only).

Author(s)

Robert Pearce

References

Examples

solve_1comp(chem.name='Bisphenol-A', days=1)
params <- parameterize_1comp(chem.cas="80-05-7")
solve_1comp(parameters=params)

solve_3comp
Solve_3comp

Description

This function solves for the amounts or concentrations of a chemical in different tissues as functions of time based on the dose and dosing frequency. It uses a three compartment model with partition coefficients.

Usage

solve_3comp(
    chem.name = NULL,
    chem.cas = NULL,
    dtxsid = NULL,
    times = NULL,
    parameters = NULL,
    days = 10,
    tsteps = 4,
    daily.dose = NULL,
    dose = NULL,
    doses.per.day = NULL,
    initial.values = NULL,
    plots = FALSE,
    suppress.messages = FALSE,
    species = "Human",
    iv.dose = FALSE,
    input.units = "mg/kg",
    output.units = NULL,
    method = "lsoda",
    rtol = 1e-08,
    atol = 1e-12,
    default.to.human = FALSE,
    recalc.blood2plasma = FALSE,
    recalc.clearance = FALSE,
    dosing.matrix = NULL,
    adjusted.Funbound.plasma = TRUE,
    regression = TRUE,
    restrictive.clearance = TRUE,
    minimum.Funbound.plasma = 1e-04,
    monitor.vars = NULL,
    ...
)
**solve_3comp**

**Arguments**

- `chem.name`: Either the chemical name, CAS number, or the parameters must be specified.
- `chem.cas`: Either the chemical name, CAS number, or the parameters must be specified.
- `dtxsid`: EPA's 'DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs.
- `times`: Optional time sequence for specified number of days. The dosing sequence begins at the beginning of times.
- `parameters`: Chemical parameters from parameterize_3comp function, overrides `chem.name` and `chem.cas`.
- `days`: Length of the simulation.
- `tsteps`: The number time steps per hour.
- `daily.dose`: Total daily dose, mg/kg BW.
- `dose`: Amount of a single dose, mg/kg BW.
- `doses.per.day`: Number of doses per day.
- `initial.values`: Vector containing the initial concentrations or amounts of the chemical in specified tissues with units corresponding to `output.units`. Defaults are zero.
- `plots`: Plots all outputs if true.
- `suppress.messages`: Whether or not the output message is suppressed.
- `species`: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- `iv.dose`: Simulates a single i.v. dose if true.
- `input.units`: Input units of interest assigned to dosing, defaults to mg/kg BW.
- `output.units`: A named vector of output units expected for the model results. Default, NULL, returns model results in units specified in the 'modelinfo' file. See table below for details.
- `method`: Method used by integrator (deSolve).
- `rtol`: Argument passed to integrator (deSolve).
- `atol`: Argument passed to integrator (deSolve).
- `default.to.human`: Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).
- `recalc.blood2plasma`: Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters, calculated with hematocrit, Funbound.plasma, and Krbc2pu.
- `recalc.clearance`: Recalculates the the hepatic clearance (Clmetabolism) with new `million.cells.per.gliver` parameter.
- `dosing.matrix`: Vector of dosing times or a matrix consisting of two columns or rows named "dose" and "time" containing the time and amount, in mg/kg BW, of each dose.
- `adjusted.Funbound.plasma`: Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.
- `regression`: Whether or not to use the regressions in calculating partition coefficients.
- `restrictive.clearance`: Protein binding not taken into account (set to 1) in liver clearance if FALSE.
minimum.Funbound.plasma

Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

monitor.vars

Which variables are returned as a function of time. Defaults value of NULL provides "Cliver", "Csyscomp", "Atubules", "Ametabolized", "AUC"

Additional arguments passed to the integrator.

Details

Note that the model parameters have units of hours while the model output is in days.

Default of NULL for doses.per.day solves for a single dose.

The compartments used in this model are the gutlumen, gut, liver, and rest-of-body, with the plasma equivalent to the liver plasma.
When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

**Value**

A matrix of class deSolve with a column for time (in days) and each compartment, the plasma concentration, area under the curve, and a row for each time point.

**Author(s)**

John Wambaugh and Robert Pearce
References

Examples
```r
solve_3comp(chem.name='Bisphenol-A',doses.per.day=2,daily.dose=.5,days=1,tsteps=2)
params <-parameterize_3comp(chem.cas="80-05-7")
solve_3comp(parameters=params)
```

Description
This function solves for the amounts or concentrations in uM of a chemical in different tissues of a maternofetal system as functions of time based on the dose and dosing frequency.

Usage
```r
solve_fetal_pbtk(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  times = seq(13 * 7, 40 * 7, 1),
  parameters = NULL,
  days = NULL,
  species = "human",
  tsteps = 4,
  dose = NULL,
  dosing.matrix = NULL,
  daily.dose = NULL,
  doses.per.day = NULL,
  initial.values = NULL,
  plots = FALSE,
  suppress.messages = FALSE,
  iv.dose = FALSE,
  input.units = "mg/kg",
  output.units = NULL,
  method = "lsoda",
  rtol = 1e-08,
  atol = 1e-12,
  default.to.human = FALSE,
  recalc.blood2plasma = FALSE,
  recalc.clearance = FALSE,
  adjusted.Funbound.plasma = TRUE,
  regression = TRUE,
  restrictive.clearance = TRUE,
  minimum.Funbound.plasma = 1e-04,
)
monitor.vars = NULL,
...
)

Arguments

chem.name Either the chemical name, CAS number, or the parameters must be specified.
chem.cas Either the chemical name, CAS number, or the parameters must be specified.
dtxsid EPA's DSSTox Structure ID (http://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
times Optional time sequence in days. Dosing sequence begins at the beginning of times. Default is from 13th week of pregnancy to 40th due to data constraints.
parameters Chemical parameters from parameterize_fetal_pbtk function, overrides chem.name and chem.cas.
days Length of the simulation.
species Included for compatibility with other functions, but the model will not run for non-human species (default "Human").
tsteps The number time steps per hour. Default of 4.
dose Amount of a single dose, mg/kg BW.
dosing.matrix A matrix of either one column (or row) with a set of dosing times or with two columns (or rows) correspondingly named "dose" and "time" containing the time and amount, in mg/kg BW, of each dose.
daily.dose Total daily dose, mg.
doses.per.day Number of doses per day.
initial.values Vector containing the initial concentrations or amounts of the chemical in specified tissues with units corresponding to compartment.units. Defaults are zero.
plots Plots all outputs if true.
suppress.messages Whether or not the output message is suppressed.
iv.dose Simulates a single i.v. dose if true.
input.units Input units of interest assigned to dosing, defaults to mg/kg BW
output.units A named vector of output units expected for the model results. Default, NULL, returns model results in units specified in the 'modelinfo' file. See table below for details.
method Method used by integrator (deSolve).
rtol Argument passed to integrator (deSolve).
atol Argument passed to integrator (deSolve).
default.to.human Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).
recalc.blood2plasma Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters, calculated with hematocrit, Funbound.plasma, and Krbc2pu.
recalc.clearance Recalculates the the hepatic clearance (Clmetabolism) with new million.cells.per.gliver parameter.
adjusted.Funbound.plasma

Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.

regression

Whether or not to use the regressions in calculating partition coefficients.

restrictive.clearance

Protein binding not taken into account (set to 1) in liver clearance if FALSE.

minimum.Funbound.plasma

Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

monitor.vars

Which variables to track by default

Additional arguments passed to the integrator.

Details

The stage of pregnancy simulated here begins by default at the 13th week due to a relative lack of data to support parameterization prior, in line with the recommendations of Kapraun et al. 2019 ("Empirical models for anatomical and physiological..."), and ends at the 40th week of pregnancy. Note that the model parameters have units of hours while the model output is in days. Dose is in mg, not scaled for body weight.

Default NULL value for doses.per.day solves for a single dose.

The maternal compartments used in this model are the gut lumen, gut, liver, venous blood, arterial blood, lung, adipose tissue, kidney, thyroid, and rest of body. A placenta is modeled as a joint organ shared by mother and fetus, through which chemical exchange can occur with the fetus. Fetal compartments include arterial blood, venous blood, kidney, thyroid, liver, lung, gut, brain, and rest of body.

The extra compartments include the amounts or concentrations metabolized by the liver and excreted by the kidneys through the tubules.

AUC is the area under the curve of the plasma concentration.

This gestational model is only parameterized for humans.

Value

A matrix of class deSolve with a column for time(in days), each compartment, the area under the curve, and plasma concentration and a row for each time point.

Author(s)

John Wambaugh, Mark Sfeir, and Dustin Kapraun

Examples

```r
out = solve_fetal_pbtk(chem.name = 'bisphenol a', daily.dose = 1, doses.per.day = 3)
```
Description

This function solves for the amounts or concentrations of a chemical in different tissues as functions of time as a result of inhalation exposure to an ideal gas.

Usage

```r
solve_gas_pbtk(
    chem.name = NULL,
    chem.cas = NULL,
    dtxsid = NULL,
    parameters = NULL,
    times = NULL,
    days = 10,
    tsteps = 4,
    daily.dose = NULL,
    doses.per.day = NULL,
    dose = NULL,
    dosing.matrix = NULL,
    forcings = NULL,
    exp.start.time = 0,
    exp.conc = 1,
    period = 24,
    exp.duration = 12,
    initial.values = NULL,
    plots = FALSE,
    suppress.messages = FALSE,
    species = "Human",
    iv.dose = FALSE,
    input.units = "ppmv",
    output.units = NULL,
    method = "lsoda",
    rtol = 1e-08,
    atol = 1e-12,
    default.to.human = FALSE,
    recalc.blood2plasma = FALSE,
    recalc.clearance = FALSE,
    adjusted.Funbound.plasma = TRUE,
    regression = TRUE,
    restrictive.clearance = TRUE,
    minimum.Funbound.plasma = 1e-04,
    monitor.vars = NULL,
    vmax = 0,
    km = 1,
    exercise = FALSE,
    fR = 12,
    VT = 0.75,
    VD = 0.15,
)```
Arguments

- **chem.name**: Either the chemical name, CAS number, or the parameters must be specified.
- **chem.cas**: Either the chemical name, CAS number, or the parameters must be specified.
- **dtxsid**: EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs.
- **parameters**: Chemical parameters from `parameterize_gas_pbtk` (or other bespoke) function, overrides `chem.name` and `chem.cas`.
- **times**: Optional time sequence for specified number of days. Dosing sequence begins at the beginning of times.
- **days**: Length of the simulation.
- **tsteps**: The number of time steps per hour.
- **daily.dose**: Total daily dose.
- **doses.per.day**: Number of doses per day.
- **dose**: Amount of a single dose.
- **dosing.matrix**: Vector of dosing times or a matrix consisting of two columns or rows named "dose" and "time" containing the time and amount of each dose.
- **forcings**: Manual input of 'forcings' data series argument for ode integrator. If left unspecified, 'forcings' defaults to NULL, and then other input parameters (see `exp.start.time`, `exp.conc`, `exp.duration`, and `period`) provide the necessary information to assemble a forcings data series.
- **exp.start.time**: Start time in specifying forcing exposure series, default 0.
- **exp.conc**: Specified inhalation exposure concentration for use in assembling 'forcings' data series argument for integrator. Defaults to units of ppmv.
- **period**: For use in assembling forcing function data series 'forcings' argument, specified in hours.
- **exp.duration**: For use in assembling forcing function data series 'forcings' argument, specified in hours.
- **initial.values**: Vector containing the initial concentrations or amounts of the chemical in specified tissues with units corresponding to those specified for the model outputs. Default values are zero.
- **plots**: Plots all outputs if true.
- **suppress.messages**: Whether or not the output message is suppressed.
- **species**: Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").
- **iv.dose**: Simulates a single i.v. dose if true.
- **input.units**: Input units of interest assigned to dosing, including forcings. Defaults to "ppmv" as applied to the default forcings scheme.
- **output.units**: A named vector of output units expected for the model results. Default, NULL, returns model results in units specified in the 'modelinfo' file. See table below for details.
- **method**: Method used by integrator (deSolve).
- **rtol**: Argument passed to integrator (deSolve).
atol Argument passed to integrator (deSolve).
default.to.human Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).
recalc.blood2plasma Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters, calculated with hematocrit, Funbound.plasma, and Krbc2pu.
recalc.clearance Recalculates the hepatic clearance (Clmetabolism) with new million.cells.per.gliver parameter.
adjusted.Funbound.plasma Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.
regression Whether or not to use the regressions in calculating partition coefficients.
restrictive.clearance Protein binding not taken into account (set to 1) in liver clearance if FALSE.
minimum.Funbound.plasma Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).
vmax Michaelis-Menten vmax value in reactions/min
km Michaelis-Menten concentration of half-maximal reaction velocity in desired output concentration units.
exercise Logical indicator of whether to simulate an exercise-induced heightened respiration rate
fR Respiratory frequency (breaths/minute), used especially to adjust breathing rate in the case of exercise. This parameter, along with VT and VD (below) gives another option for calculating Qalv (Alveolar ventilation) in case pulmonary ventilation rate is not known
VT Tidal volume (L), to be modulated especially as part of simulating the state of exercise
VD Anatomical dead space (L), to be modulated especially as part of simulating the state of exercise
... Additional arguments passed to the integrator.

**Details**

The default dosing scheme involves a specification of the start time of exposure (exp.start.time), the concentration of gas inhaled (exp.conc), the period of a cycle of exposure and non-exposure (period), the duration of the exposure during that period (exp.duration), and the total days simulated. Together, these arguments determine the "forcings" passed to the ODE integrator. Forcings can also be specified manually, or effectively turned off by setting exposure concentration to zero, if the user prefers to simulate dosing by other means.

The "forcings" object is configured to be passed to the integrator with, at the most, a basic unit conversion among ppmv, mg/L, and uM. No scaling by BW is set to be performed on the forcings series.
Note that the model parameters have units of hours while the model output is in days.
Default NULL value for doses per day solves for a single dose.
The compartments used in this model are the gut lumen, gut, liver, kidneys, veins, arteries, lungs, and the rest of the body.
The extra compartments include the amounts or concentrations metabolized by the liver and excreted by the kidneys through the tubules.
AUC is the area under the curve of the plasma concentration.
Model Figure from (Linakis et al. 2020):

\[
\begin{array}{c}
\text{Inhaled Air} \\
\text{Exhaled Air} \\
\text{Mucus} \\
\text{Alveolar Space} \\
\text{Lung} \\
\text{Gut Lumen} \\
\text{Liver} \\
\text{Body} \\
\text{Kidney} \\
\text{Venous Blood} \\
\text{Arterial Blood}
\end{array}
\]

Model parameters are named according to the following convention:

<table>
<thead>
<tr>
<th>prefix</th>
<th>suffix</th>
<th>Meaning</th>
<th>units</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td></td>
<td>Partition coefficient for tissue to free plasma</td>
<td>unitless</td>
</tr>
<tr>
<td>V</td>
<td></td>
<td>Volume</td>
<td></td>
</tr>
</tbody>
</table>
When species is specified but chemical-specific in vitro data are not available, the function uses the appropriate physiological data (volumes and flows) but default.to.human = TRUE must be used to substitute human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value

A matrix of class deSolve with a column for time (in days), each compartment, the area under the curve, and plasma concentration and a row for each time point.

Author(s)

Matt Linakis, John Wambaugh, Mark Sfeir, Miyuki Breen

References


Examples

```r
solve_gas_pbtk(chem.name = 'pyrene', exp.conc = 1, period = 24, expduration = 24)

out <- solve_gas_pbtk(chem.name='pyrene',exp.conc = 0, doses.per.day = 2,
                        daily.dose = 3, input.units = "umol", plots=TRUE,initial.values=c(Aven=20))

out <- solve_gas_pbtk(chem.name = 'pyrene', exp.conc = 3, period = 24,
                        exp.duration = 6, exercise = TRUE)

params <- parameterize_gas_pbtk(chem.cas="80-05-7")
solve_gas_pbtk(parameters=params)

# Oral dose with exhalation as a route of elimination:
out <- solve_gas_pbtk(chem.name = 'bisphenol a', exp.conc = 0, dose=100,
                        input.units="mg/kg")

# Note that different model compartments for this model have different units
# and that the final units can be controlled with the output.units argument:
head(solve_gas_pbtk(chem.name="lindane"))

# Convert all compartment units to mg/L:
head(solve_gas_pbtk(chem.name="lindane",output.units="mg/L"))

# Convert just the plasma to mg/L:
head(solve_gas_pbtk(chem.name="lindane",output.units=list(Cplasma="mg/L")))
```
solve_model

Description

solve_model is designed to accept systematized metadata (provided by the model.list defined in the
modelinfo files) for a given toxicokinetic model, including names of variables, parameterization
functions, and key units, and use it along with chemical information to prepare an ode system
for numerical solution over time of the amounts or concentrations of chemical in different bodily
compartments of a given species (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").

Usage

```r
solve_model(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  times = NULL,
  parameters = NULL,
  model = NULL,
  route = "oral",
  dosing = NULL,
  days = 10,
  tsteps = 4,
  initial.values = NULL,
  initial.value.units = NULL,
  plots = FALSE,
  monitor.vars = NULL,
  suppress.messages = FALSE,
  species = "Human",
  input.units = "mg/kg",
  output.units = NULL,
  method = "lsoda",
  rtol = 1e-08,
  atol = 1e-12,
  recalc.blood2plasma = FALSE,
  recalc.clearance = FALSE,
  adjusted.Funbound.plasma = TRUE,
  minimum.Funbound.plasma = 1e-04,
  parameterize.arg.list = list(default.to.human = FALSE, clint.pvalue.threshold = 0.05,
                          restrictive.clearance = TRUE, regression = TRUE),
  ...
)
```

Arguments

- **chem.name**: Either the chemical name, CAS number, or the parameters must be specified.
- **chem.cas**: Either the chemical name, CAS number, or the parameters must be specified.
- **dtxsid**: EPA’s DSSTox Structure ID (http://comptox.epa.gov/dashboard) the chemical must be identified by either CAS, name, or DTXSIDs
**solve_model**

**times**  
Optional time sequence for specified number of days. Dosing sequence begins at the beginning of times.

**parameters**  
List of chemical parameters, as output by parameterize_pbtk function. Overrides chem.name and chem.cas.

**model**  
Specified model to use in simulation: "pbtk", "3compartment", "3compartmentss", "1compartment", "schmitt", ...

**route**  
String specification of route of exposure for simulation: "oral", "iv", "inhalation", ...

**dosing**  
List of dosing metrics used in simulation, which includes the namesake entries of a model's associated dosing.params. In the case of most httk models, these should include "initial.dose", "doses.per.day", "daily.dose", and "dosing.matrix". The "dosing.matrix" is used for more precise dose regimen specification, and is a matrix consisting of two columns or rows named "time" and "dose" containing the time and amount of each dose. If none of the namesake entries of the dosing list is set to a non-NULL value, solve_model uses a default dose of 1 mg/kg BW along with the dose type (add/multiply) specified for a given route (e.g. add the dose to gut lumen for oral route)

**days**  
Simulated period. Default 10 days.

**tsteps**  
The number of time steps per hour. Default of 4.

**initial.values**  
Vector of numeric values containing the initial concentrations or amounts of the chemical in specified tissues with units corresponding to those specified for the model outputs. Default values are zero.

**initial.value.units**  
Vector of character strings containing the units corresponding to 'initial.values' specified for the model outputs. Default is assuming the units match expected compartment units for the model.

**plots**  
Plots all outputs if true.

**monitor.vars**  
Which variables are returned as a function of time. Default values of NULL looks up variables specified in modelinfo_MODEL.R

**suppress.messages**  
Whether or not the output messages are suppressed.

**species**  
Species desired (models have been designed to be parameterized for some subset of the following species: "Rat", "Rabbit", "Dog", "Mouse", or default "Human").

**input.units**  
Input units of interest assigned to dosing. Defaults to mg/kg BW, in line with the default dosing scheme of a one-time dose of 1 mg/kg in which no other dosing parameters are specified.

**output.units**  
Output units of interest for the compiled components. Defaults to NULL, and will provide values in model units if unspecified.

**method**  
Method used by integrator (deSolve).

**rtol**  
Argument passed to integrator (deSolve).

**atol**  
Argument passed to integrator (deSolve).

**recalc.blood2plasma**  
Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters, calculated with hematocrit, Funbound.plasma, and Krbc2pu.

**recalc.clearance**  
Recalculates the the hepatic clearance (Clmetabolism) with new million.cells.per.gliver parameter.
solve_model

adjusted.Funbound.plasma
Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.

minimum.Funbound.plasma
Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset)

parameterize.arg.list
Additional parameterized passed to the model parameterization function.

... Additional arguments passed to the integrator.

default.to.human
Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).

regression
Whether or not to use the regressions in calculating partition coefficients.

restrictive.clearance
Protein binding not taken into account (set to 1) in liver clearance if FALSE.

Details

Dosing values with certain acceptable associated input.units (like mg/kg BW) are configured to undergo a unit conversion. All model simulations are intended to run with units as specified by "compartment.units" in the model.list (as defined by the modelinfo files).

The ‘dosing’ argument includes all parameters needed to describe exposure in terms of route of administration, frequency, and quantity short of scenarios that require use of a more precise forcing function. If the dosing argument’s namesake entries are left NULL, solve_model defaults to a single-time dose of 1 mg/kg BW according to the given dosing route and associated type (either add/multiply, for example we typically add a dose to gut lumen when oral route is specified).

AUC is the area under the curve of the plasma concentration.

Model parameters are named according to the following convention:

<table>
<thead>
<tr>
<th>prefix</th>
<th>suffix</th>
<th>Meaning</th>
<th>units</th>
</tr>
</thead>
<tbody>
<tr>
<td>K</td>
<td></td>
<td>Partition coefficient for tissue to free plasma</td>
<td>tab unitless</td>
</tr>
<tr>
<td>V</td>
<td></td>
<td>Volume</td>
<td>L</td>
</tr>
<tr>
<td>Q</td>
<td></td>
<td>Flow</td>
<td>L/h</td>
</tr>
<tr>
<td>k</td>
<td></td>
<td>Rate</td>
<td>1/h</td>
</tr>
<tr>
<td>c</td>
<td></td>
<td>Parameter is proportional to body weight</td>
<td>1 / kg for volumes and 1/kg^(3/4) for flows</td>
</tr>
</tbody>
</table>

When species is specified but chemical-specific in vitro data are not available, the function uses the appropriate physiological data (volumes and flows) but default.to.human = TRUE must be used to substitute human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

Value

A matrix of class deSolve with a column for time(in days), each compartment, the area under the curve, and plasma concentration and a row for each time point.

Author(s)

John Wambaugh, Robert Pearce, Miyuki Breen, Mark Sfeir, and Sarah E. Davidson
References


solve_pbtk

Description

This function solves for the amounts or concentrations in uM of a chemical in different tissues as functions of time based on the dose and dosing frequency.

Usage

solve_pbtk(
  chem.name = NULL,
  chem.cas = NULL,
  dtxsid = NULL,
  times = NULL,
  parameters = NULL,
  days = 10,
  tsteps = 4,
  daily.dose = NULL,
  dose = NULL,
  doses.per.day = NULL,
  initial.values = NULL,
  plots = FALSE,
  suppress.messages = FALSE,
  species = "Human",
  iv.dose = FALSE,
  input.units = "mg/kg",
  output.units = NULL,
  method = "lsoda",
  rtol = 1e-08,
  atol = 1e-12,
  default.to.human = FALSE,
  recalc.blood2plasma = FALSE,
  recalc.clearance = FALSE,
  dosing.matrix = NULL,
  adjusted.Fbound.plasma = TRUE,
  regression = TRUE,
  restrictive.clearance = TRUE,
  minimum.Fbound.plasma = 1e-04,
  monitor.vars = NULL,
  ...
)

Arguments

chem.name Either the chemical name, CAS number, or the parameters must be specified.
Either the chemical name, CAS number, or the parameters must be specified.

EPA's DSSTox Structure ID ([https://comptox.epa.gov/dashboard](https://comptox.epa.gov/dashboard)) the chemical must be identified by either CAS, name, or DTXSIDs

Optional time sequence for specified number of days. Dosing sequence begins at the beginning of times.

Chemical parameters from parameterize_pbtk function, overrides chem.name and chem.cas.

Length of the simulation.

The number of time steps per hour.

Total daily dose, defaults to mg/kg BW.

Amount of a single dose, defaults to mg/kg BW.

Number of doses per day.

Vector containing the initial concentrations or amounts of the chemical in specified tissues with units corresponding to output.units. Defaults are zero.

Plots all outputs if true.

Whether or not the output message is suppressed.

Species desired (either "Rat", "Rabbit", "Dog", "Mouse", or default "Human").

Simulates a single i.v. dose if true.

Input units of interest assigned to dosing, defaults to mg/kg BW

A named vector of output units expected for the model results. Default, NULL, returns model results in units specified in the 'modelinfo' file. See table below for details.

Method used by integrator (deSolve).

Argument passed to integrator (deSolve).

Argument passed to integrator (deSolve).

Substitutes missing animal values with human values if true (hepatic intrinsic clearance or fraction of unbound plasma).

Recalculates the ratio of the amount of chemical in the blood to plasma using the input parameters, calculated with hematocrit, Funbound.plasma, and Krbc2pu.

Recalculates the the hepatic clearance (Clmetabolism) with new million.cells.per.gliver parameter.

Vector of dosing times or a matrix consisting of two columns or rows named "dose" and "time" containing the time and amount, in mg/kg BW, of each dose.

Uses adjusted Funbound.plasma when set to TRUE along with partition coefficients calculated with this value.

Whether or not to use the regressions in calculating partition coefficients.

Protein binding not taken into account (set to 1) in liver clearance if FALSE.

Monte Carlo draws less than this value are set equal to this value (default is 0.0001 – half the lowest measured Fup in our dataset).

Additional arguments passed to the integrator.

Details

Note that the model parameters have units of hours while the model output is in days.

Default NULL value for doses_per_day solves for a single dose.

The compartments used in this model are the gutlumen, gut, liver, kidneys, veins, arteries, lungs, and the rest of the body.

The extra compartments include the amounts or concentrations metabolized by the liver and excreted by the kidneys through the tubules.

AUC is the area under the curve of the plasma concentration.
When species is specified as rabbit, dog, or mouse, the function uses the appropriate physiological data (volumes and flows) but substitutes human fraction unbound, partition coefficients, and intrinsic hepatic clearance.

**Value**

A matrix of class deSolve with a column for time (in days), each compartment, the area under the curve, and plasma concentration and a row for each time point.
solve_pbtk

Author(s)

John Wambaugh and Robert Pearce

References


Examples

# Multiple doses per day:
head(solve_pbtk(
  chem.name='Bisphenol-A',
  daily.dose=.5,
  days=5,
  doses.per.day=2,
  tsteps=2))

# Starting with an initial concentration:
out <- solve_pbtk(
  chem.name='bisphenola',
  dose=0,
  output.units="mg/L",
  initial.values=c(Agut=200))

# Working with parameters (rather than having solve_pbtk retrieve them):
params <- parameterize_pbtk(chem.cas="80-05-7")
head(solve_pbtk(parameters=params))

# We can change the parameters given to us by parameterize_pbtk:
params <- parameterize_pbtk(dtxsid="DTXSID4020406", species = "rat")
params["Funbound.plasma"] <- 0.1
out <- solve_pbtk(parameters=params)

# A fifty day simulation:
out <- solve_pbtk(
  chem.name = "Bisphenol A",
  days = 50,
  daily.dose=1,
  doses.per.day = 3)
plot.data <- as.data.frame(out)
css <- calc_analytic_css(chem.name = "Bisphenol A")

library("ggplot2")
c.vs.t <- ggplot(plot.data, aes(time, Cplasma)) +
  geom_line() +
  geom_hline(yintercept = css) +
  ylab("Plasma Concentration (uM)") +
  xlab("Day") +
  theme(
    axis.text = element_text(size = 16),
    axis.title = element_text(size = 16),
    plot.title = element_text(size = 17)) +
ggtitle("Bisphenol A")
print(c.vs.t)
spleen_mass_children  
*Predict spleen mass for children*

### Description

For individuals under 18, predict the spleen mass from height, weight, and gender, using equations from Ogiu et al. (1997)

### Usage

```
spleen_mass_children(height, weight, gender)
```

### Arguments

- **height**: Vector of heights in cm.
- **weight**: Vector of weights in kg.
- **gender**: Vector of genders (either 'Male' or 'Female').

### Value

A vector of spleen masses in kg.

### Author(s)

Caroline Ring

### References

Supplementary output from Linakis 2020 vignette analysis.

**Description**
Supplementary output from Linakis 2020 vignette analysis.

**Usage**
supptab1_Linakis2020

**Format**
A data.frame containing x rows and y columns.

**Author(s)**
Matt Linakis

**Source**
Matt Linakis

**References**
DSStox database (https://www.epa.gov/ncct/dsstox)

More supplementary output from Linakis 2020 vignette analysis.

**Description**
More supplementary output from Linakis 2020 vignette analysis.

**Usage**
supptab2_Linakis2020

**Format**
A data.frame containing x rows and y columns.

**Author(s)**
Matt Linakis

**Source**
Matt Linakis

**References**
DSStox database (https://www.epa.gov/ncct/dsstox)
Tables.Rdata.stamp  
*A timestamp of table creation*

**Description**

The Tables.RData file is separately created as part of building a new release of HTTK. This timestamp indicates the script used to build the file and when it was run.

**Usage**

Tables.Rdata.stamp

**Format**

An object of class character of length 1.

**Author(s)**

John Wambaugh

---

tissue.data  
*Tissue composition and species-specific physiology parameters*

**Description**

This data set contains values from Schmitt (2008) and Ruark et al. (2014) describing the composition of specific tissues and from Birnbaum et al. (1994) describing volumes of and blood flows to those tissues, allowing parameterization of toxicokinetic models for human, mouse, rat, dog, or rabbit. Tissue volumes were calculated by converting the fractional mass of each tissue with its density (both from ICRP), lumping the remaining tissues into the rest-of-body, excluding the mass of the gastrointestinal contents.

**Usage**

tissue.data

**Format**

A data.frame containing 13 rows and 20 columns.

**Author(s)**

John Wambaugh, Robert Pearce, and Nisha Sipes

**Source**

Pearce et al. (2017), in preparation,

tissue_masses_flows

References


tissue_masses_flows  Given a data.table describing a virtual population by the NHANES quantities, generates HTTK physiological parameters for each individual.

Description

Given a data.table describing a virtual population by the NHANES quantities, generates HTTK physiological parameters for each individual.

Usage

tissue_masses_flows(tmf_dt)

Arguments

- tmf_dt: A data.table generated by gen_age_height_weight(), containing variables gender, reth, age_months, age_years, weight, and height.

Value

The same data.table, with additional variables describing tissue masses and flows.

Author(s)

Caroline Ring

References


tissue_scale  

Description

Allometrically scale a tissue mass or flow based on height^{3/4}.

Usage

tissue_scale(height_ref, height_indiv, tissue_mean_ref)

Arguments

height_ref: Reference height in cm.

height_indiv: Individual height in cm.

tissue_mean_ref: Reference tissue mass or flow.

Value

Allometrically scaled tissue mass or flow, in the same units as tissue_mean_ref.

Author(s)

Caroline Ring

References


wambaugh2019  

in vitro Toxicokinetic Data from Wambaugh et al. (2019)

Description

These data are the new HTTK in vitro data for chemicals reported in Wambaugh et al. (2019). They are the processed values used to make the figures in that manuscript. These data summarize the results of Bayesian analysis of the in vitro toxicokinetic experiments conducted by Cyprotex to characterize fraction unbound in the presence of pooled human plasma protein and the intrinsic hepatic clearance of the chemical by pooled human hepatocytes.

Usage

wambaugh2019
### Format

A data frame with 496 rows and 17 variables:

- **Compound** The name of the chemical
- **CAS** The Chemical Abstracts Service Registry Number
- **Human.Clint** Median of Bayesian credible interval for intrinsic hepatic clearance (uL/min/million hepatocytes)
- **Human.Clint.pValue** Probability that there is no clearance
- **Human.Funbound.plasma** Median of Bayesian credible interval for fraction of chemical free in the presence of plasma
- **pKa_Accept** pH(s) at which hydrogen acceptor sites (if any) are at equilibrium
- **pKa_Donor** pH(s) at which hydrogen donor sites (if any) are at equilibrium
- **DSSTox_Substance_Id** Identifier for CompTox Chemical Dashboard
- **SMILES** Simplified Molecular-Input Line-Entry System structure description
- **Human.Clint.Low95** Lower 95th percentile of Bayesian credible interval for intrinsic hepatic clearance (uL/min/million hepatocytes)
- **Human.Clint.High95** Upper 95th percentile of Bayesian credible interval for intrinsic hepatic clearance (uL/min/million hepatocytes)
- **Human.Clint.Point** Point estimate of intrinsic hepatic clearance (uL/min/million hepatocytes)
- **Human.Funbound.plasma.Low95** Lower 95th percentile of Bayesian credible interval for fraction of chemical free in the presence of plasma
- **Human.Funbound.plasma.High95** Upper 95th percentile of Bayesian credible interval for fraction of chemical free in the presence of plasma
- **Human.Funbound.plasma.Point** Point estimate of the fraction of chemical free in the presence of plasma
- **MW** Molecular weight (Daltons)
- **logP** log base ten of octanol:water partition coefficient

### Author(s)

John Wambaugh

### Source

Wambaugh et al. (2019)

### References

**Description**

These data are a subset of the Bayesian inferences reported by Ring et al. (2017) from the U.S. Centers for Disease Control and Prevention (CDC) National Health and Nutrition Examination Survey (NHANES). They reflect the population median intake rate (mg/kg body weight/day), with uncertainty.

**Usage**

**Format**

A data frame with 20 rows and 4 variables:

- **IP**: The median of the Bayesian credible interval for median population intake rate (mg/kg body-weight/day)
- **IP.min**: The lower 95th percentile of the Bayesian credible interval for median population intake rate (mg/kg bodyweight/day)
- **IP.max**: The upper 95th percentile of the Bayesian credible interval for median population intake rate (mg/kg bodyweight/day)
- **CASRN**: The Chemical Abstracts Service Registry Number

**Author(s)**

John Wambaugh

**Source**

Wambaugh et al. (2019)

**References**


These data are the new HTTK in vitro data for chemicals reported in Wambaugh et al. (2019). They are the output of different Bayesian models evaluated to compare using a single protein concentration vs. the new three concentration titration protocol. These data summarize the results of Bayesian analysis of the in vitro toxicokinetic experiments conducted by Cyprotex to characterize fraction unbound in the presence of pooled human plasma protein and the intrinsic hepatic clearance of the chemical by pooled human hepatocytes. This file includes replicates (different Compound-Name id’s but same chemical).
Decreases.Prob  Probability that the chemical concentration decreased systematically during hepatic clearance assay.

Saturates.Prob  Probability that the rate of chemical concentration decrease varied between the 1 and 10 uM hepatic clearance experiments.

Slope.1uM.Median  Estimated slope for chemical concentration decrease in the 1 uM hepatic clearance assay.

Slope.10uM.Median  Estimated slope for chemical concentration decrease in the 10 uM hepatic clearance assay.

CLint.1uM.Median  Median of Bayesian credible interval for intrinsic hepatic clearance at 1 uM initial chemical concentration (uL/min/million hepatocytes)

CLint.1uM.Low95th  Lower 95th percentile of Bayesian credible interval for intrinsic hepatic clearance at 1 uM initial chemical concentration (uL/min/million hepatocytes)

CLint.1uM.High95th  Upper 95th percentile of Bayesian credible interval for intrinsic hepatic clearance at 1 uM initial chemical concentration (uL/min/million hepatocytes)

CLint.10uM.Median  Median of Bayesian credible interval for intrinsic hepatic clearance at 10 uM initial chemical concentration (uL/min/million hepatocytes)

CLint.10uM.Low95th  Lower 95th percentile of Bayesian credible interval for intrinsic hepatic clearance at 10 uM initial chemical concentration (uL/min/million hepatocytes)

CLint.10uM.High95th  Upper 95th percentile of Bayesian credible interval for intrinsic hepatic clearance at 10 uM initial chemical concentration (uL/min/million hepatocytes)

CLint.1uM.Point  Point estimate of intrinsic hepatic clearance (uL/min/million hepatocytes) for 1 uM initial chemical concentration

CLint.10uM.Point  Point estimate of intrinsic hepatic clearance (uL/min/million hepatocytes) for 10 uM initial chemical concentration

Fit  Classification of clearance observed

SMILES  Simplified Molecular-Input Line-Entry System structure description

Author(s)

John Wambaugh

Source

Wambaugh et al. (2019)

References

**wambaugh2019.seem3**  
*ExpoCast SEEM3 Consensus Exposure Model Predictions for Chemical Intake Rates*

**Description**

These data are a subset of the Bayesian inferences reported by Ring et al. (2019) for a consensus model of twelve exposure predictors. The predictors were calibrated based upon their ability to predict intake rates inferred National Health and Nutrition Examination Survey (NHANES). They reflect the population median intake rate (mg/kg body weight/day), with uncertainty.

**Usage**

`wambaugh2019.seem3`

**Format**

A data frame with 385 rows and 38 variables:

**Author(s)**

John Wambaugh

**Source**

Wambaugh et al. (2019)

**References**


---

**wambaugh2019.tox21**  
*Tox21 2015 Active Hit Calls (EPA)*

**Description**

The ToxCast and Tox21 research programs employ batteries of high-throughput assays to assess chemical bioactivity in vitro. Not every chemical is tested through every assay. Most assays are conducted in concentration response, and each corresponding assay endpoint is analyzed statistically to determine if there is a concentration-dependent response or "hit" using the ToxCast Pipeline. Most assay endpoint-chemical combinations are non-responsive. Here, only the hits are treated as potential indicators of bioactivity. This bioactivity does not have a direct toxicological interpretation. The October 2015 release (invitrodb_v2) of the ToxCast and Tox21 data were used for this analysis. This object contains just the chemicals in Wambaugh et al. (2019) and only the quantiles across all assays for the ACC.
Wang et al. 2018 Wang et al. (2018) screened the blood of 75 pregnant women for the presence of environmental organic acids (EOAs) and identified mass spectral features corresponding to 453 chemical formulae of which 48 could be mapped to likely structures. Of the 48 with tentative structures the identity of six were confirmed with available chemical standards.
### Description

Microtiter Plate Well Descriptions for Armitage et al. (2014) model from Honda et al. (2019)

### Usage

```
well_param
```

### Format

A data frame / data table with 11 rows and 8 variables:

- **sysID**: Identifier for each multi-well plate system
- **well_desc**: Well description
- **well_number**: Number of wells on plate
- **area_bottom**: Area of well bottom in mm\(^2\)
- **cell_yield**: Number of cells
- **diam**: Diameter of well in mm
- **v_total**: Total volume of well in uL
- **v_working**: Working volume of well in uL

### Author(s)

Greg Honda

### Source


### References


Wetmore2012

Published toxicokinetic predictions based on in vitro data from Wetmore et al. 2012.

Description
This data set overlaps with Wetmore.data and is used only in Vignette 4 for steady state concentration.

Usage
Wetmore2012

Format
A data.frame containing 13 rows and 15 columns.

References

wfl

WHO weight-for-length charts

Description
Charts giving weight-for-length percentiles for boys and girls under age 2.

Usage
wfl

Format
a data.table with 262 rows and 4 variables:

Sex  "Male" or "Female"
Length Recumbent length in cm
P2.3 The 2.3rd percentile weight in kg for the corresponding sex and recumbent length
P97.7 The 97.7th percentile weight in kg for the corresponding sex and recumbent length

Details
For infants under age 2, weight class depends on weight for length percentile. #'

Underweight <2.3rd percentile
Normal weight 2.3rd-97.7th percentile
Obese >=97.7th percentile
Source

Index

* 1compartment
  calc_analytic_css_1comp, 27
  calc_elimination_rate, 34
  calc_half_life, 41
  calc_total_clearance, 71
  calc_vdist, 72
  convert_httkpop_1comp, 85
  parameterize_1comp, 158
  propagate_invitrouv_1comp, 183
  solve_1comp, 192

* 3compartment
  calc_analytic_css_3comp, 28
  parameterize_3comp, 161
  propagate_invitrouv_3comp, 184
  solve_3comp, 196

* 3compss
  calc_analytic_css_3compss, 30
  parameterize_steadystate, 174

* Dynamic
  scale_dosing, 188

* Export
  export_pbtk_jarnac, 96
  export_pbtk_sbml, 97

* Literature
  get_lit_cheminfo, 108
  get_lit_css, 109
  get_lit_oral_equiv, 111
  get_wetmore_cheminfo, 115
  get_wetmore_css, 116
  get_wetmore_oral_equiv, 118

* Monte-Carlo
  calc_mc_css, 52
  calc_mc_oral_equiv, 57
  calc_mc_tk, 62
  create_mc_samples, 90
  get_lit_css, 109
  get_lit_oral_equiv, 111
  get_wetmore_css, 116
  get_wetmore_oral_equiv, 118
  monte_carlo, 155

* Parameter
  available_rblood2plasma, 18
  calc_elimination_rate, 34
  calc_fetal_phys, 36
  calc_half_life, 41
  calc_hep_clearance, 45
  calc_hepatic_clearance, 43
  calc_ionization, 48
  calc_krbc2pu, 50
  calc_maternal_bw, 51
  calc_rblood2plasma, 65
  calc_total_clearance, 71
  calc_vdist, 72
  get_rblood2plasma, 113
  lump_tissues, 150
  parameterize_1comp, 158
  parameterize_3comp, 161
  parameterize_fetal_pbtk, 163
  parameterize_gas_pbtk, 166
  parameterize_pbtk, 170
  parameterize_schmitt, 173
  parameterize_steadystate, 174
  predict_partitioning_schmitt, 181

* Retrieval
  get_cheminfo, 103
  get_lit_cheminfo, 108
  get_wetmore_cheminfo, 115

* Solve
  calc_analytic_css, 24
  calc_stats, 67
  calc_tkstats, 69
  honda.ivive, 121
  solve_1comp, 192
  solve_3comp, 196
  solve_fetal_pbtk, 200
  solve_gas_pbtk, 203
  solve_model, 208
  solve_pbtk, 211

* Statistics
  calc_stats, 67
  calc_tkstats, 69

* Steady-State
  calc_css, 32
  calc.mc_css, 52
  calc.mc_oral_equiv, 57

* cheminformatics
INDEX

get_chem_id, 106

* datasets
  EPA.ref, 93
  Tables.Rdata.stamp, 218
  wfl, 228

* data
  armitage_input, 16
  aylward2014, 19
  bmiage, 21
  chem.invivo.PK.aggregate.data, 74
  chem.invivo.PK.data, 75
  chem.invivo.PK.summary.data, 78
  chem.physical_and_invitro.data, 81
  concentration_data_Linakis2020, 85
  dawson2021, 93
  fetalpcs, 98
  Frank2018invivo, 99
  hct_h, 119
  howgate, 122
  hw_H, 137
  johnson, 145
  kapraun2019, 145
  mcnally_dt, 153
  mcdt, 154
  metabolism_data_Linakis2020, 155
  Obach2008, 157
  onlyp, 157
  pc.data, 177
  pearce2017regression, 178
  pharma, 178
  physiology.data, 179
  pksim.pcs, 180
  pradeep2020, 180
  pregnonpregaucs, 183
  scr_h, 189
  supptab1_Linakis2020, 217
  supptab2_Linakis2020, 217
  tissue.data, 218
  wambaugh2019, 220
  wambaugh2019.nhanes, 222
  wambaugh2019.raw, 223
  wambaugh2019.seem3, 225
  wambaugh2019.tox21, 225
  wang2018, 226
  well_param, 227
  Wetmore2012, 228

* dynamic
  calc_mc_tk, 62

* httk-pop
  age_draw_smooth, 9
  blood_mass_correct, 20
  blood_weight, 20
  bmiage, 21
  body_surface_area, 22
  bone_mass_age, 23
  brain_mass, 24
  ckd_epi_eq, 84
  convert_h ttkpop_1comp, 85
  estimate_gfr, 94
  estimate_gfr_ped, 95
  estimate_hematocrit, 95
  gen_age_height_weight, 100
  gen_height_weight, 101
  gen_serum_creatinine, 102
  get_gfr_category, 107
  get_weight_class, 114
  hct_h, 119
  hematocrit_infants, 120
  httkpop, 123
  httkpop_biotophys_default, 127
  httkpop_direct_resample, 128
  httkpop_direct_resample_inner, 129
  httkpop_generate, 131
  httkpop_mc, 135
  httkpop_virtual_indiv, 136
  hw_H, 137
  is_in_inclusive, 144
  kidney_mass_children, 146
  liver_mass_children, 147
  lung_mass_children, 152
  mcnally_dt, 153
  mcdt, 154
  pancreas_mass_children, 158
  rfun, 186
  rmed0non0u95, 186
  scr_h, 189
  skeletal_muscle_mass, 190
  skeletal_muscle_mass_children, 191
  skin_mass_bosgra, 192
  spleen_mass_children, 216
  tissue_masses_flows, 219
  tissue_scale, 220

* in-vitro
  calc_hep_fu, 47
  invitro_mc, 140

* monte-carlo
  httkpop_biotophys_default, 127
  httkpop_direct_resample, 128
  httkpop_direct_resample_inner, 129
  httkpop_generate, 131
  httkpop_mc, 135
  httkpop_virtual_indiv, 136
  invitro_mc, 140
  propagate_invitrouv_1comp, 183
propagate_invitrouv_3comp, 184
propagate_invitrouv_pbtk, 184

* package
  htttk-package, 6

* pbtk
  calc_analytic_css_pbtk, 31
  parameterize_pbtk, 170
  propagate_invitrouv_pbtk, 184
  solve_pbtk, 211

* physiology
  calc_hep_bioavailability, 44

* schmitt
  parameterize_schmitt, 173

* simulation
  calc_mc_tk, 62

add_chemtable, 7, 17
age_draw_smooth, 9
armitage_estimate_sarea, 10
armitage_eval, 11
armitage_input, 16
augment_table, 17
available_rblood2plasma, 18
Aylward2014 (aylward2014), 19
aylward2014, 19

blood_mass_correct, 20
blood_weight, 20, 20
bmiage, 21
body_surface_area, 22
bone_mass_age, 23
brain_mass, 24

calc_analytic_css, 24, 54, 174
calc_analytic_css_1comp, 27
calc_analytic_css_3comp, 28
calc_analytic_css_3comps, 30
calc_analytic_css_pbtk, 31
calc_css, 32, 171
calc_elimination_rate, 34, 183
calc_fetal_phys, 36
calc_half_life, 41
calc_hep_bioavailability, 44
calc_hep_clearance, 43, 45, 184, 185
calc_hep_fu, 47
calc_hepatic_clearance, 43
calc_ionization, 48
calc_krbc2pu, 50
calc_maternal_bw, 51
calc_mc_css, 52, 58, 59, 174
calc_mc_oral_equiv, 57, 174
calc_mc_tk, 62
calc_rblood2plasma, 65
calc_stats, 67
calc_tkstats, 67, 69
calc_total_clearance, 71
calc_vdist, 72
CAS.checksum, 73
chem.invivo.PK.aggregate.data, 74
chem.invivo.PK.data, 75
chem.invivo.PK.summary.data, 78
chem.physical_and_invitro.data, 81, 104
ckd_epi_eq, 84
concentration_data_Linakis2020, 85
convert_httkpop_1comp, 85
convert_solve_x, 86
convert_units, 87, 88
create_mc_samples, 52, 58, 86, 90

Dawson2021 (dawson2021), 93
dawson2021, 93

EPA.ref, 93
estimate_gfr, 94
estimate_gfr_ped, 95
estimate_hematocrit, 95, 120
export_pbtk_jarnac, 96
export_pbtk_sbml, 97

fetalPCs (fetalpcs), 98
fetalpcs, 98
Frank2018invivo, 99

gen_age_height_weight, 100
gen_height_weight, 101, 138
gen_serum_creatinine, 102, 189
get_chem_id, 106
get_cheminfo, 103
get_gfr_category, 107
get_invitroPK_param, 107
get_lit_cheminfo, 108, 115
get_lit_css, 109, 116
get_lit_oral_equiv, 111, 118
get_physchem_param, 112
get_rblood2plasma, 113
get_weight_class, 114, 154
get_wetmore_cheminfo, 115
get_wetmore_css, 116
get_wetmore_oral_equiv, 118

hct_h, 119
hematocrit_infants, 120
honda.ivive, 121
howgate, 122
Hpi, 138
hpi, 120, 189
INDEX

httk (httk-package), 6
httk-package, 6
httkpop, 123
httkpop-package (httkpop), 123
httkpop_biotophys_default, 127
httkpop_direct_resample, 128
httkpop_direct_resample_inner, 129
httkpop_generate, 9, 54, 63, 86, 91, 92, 96, 100–102, 120, 125, 129, 130, 131, 135, 137, 138, 189
httkpop_mc, 90, 125, 135
httkpop_virtual_indiv, 136
hw_H, 137

in.list, 138, 143
invitro_mc, 54, 63, 90, 92, 140
is.expocast (in.list), 138
is.httk, 139, 142
is.nhanes (in.list), 138
is.pharma (in.list), 138
is.tox21 (in.list), 138
is.toxcast (in.list), 138
is_in_inclusive, 144

johnson, 145
Kapraun2019 (kapraun2019), 145
kapraun2019, 145
det, 120, 138, 189
kidney_mass_children, 146

liver_mass_children, 147
load_dawson2021, 82, 147
load_pradeep2020, 82, 148
load_sipes2017, 82, 149
lump_tissues, 150
lung_mass_children, 152

mcnally_dt, 153
mecdt, 9, 96, 100–102, 129, 130, 137, 154
metabolism_data_linakis2020, 155
monte_carlo, 90, 155

Obach2008, 157
onlyp, 157

pancreas_mass_children, 158
parameterize_1comp, 158
parameterize_3comp, 161
parameterize_fetal_pbtk, 163
parameterize_gas_pbtk, 166
parameterize_pbtk, 161, 170
parameterize_schmitt, 66, 173, 182
parameterize_steadystate, 174

pc.data, 177
Pearce2017Regression
(pearce2017regression), 178
pearce2017regression, 178
pharma, 178
physiology.data, 161, 166, 170, 179
pksim.pcs, 180
Pradeep2020 (pradeep2020), 180
pradeep2020, 180
predict_partitioning_schmitt, 66, 90, 173, 181
pregnonpregaucs, 183
propagate_invitrouv_1comp, 86, 183
propagate_invitrouv_3comp, 184
propagate_invitrouv_pbtk, 184

r_left_censored_norm, 187
reset_httk, 185
rfun, 186
rmed0non0u95, 186
scale_dosing, 89, 188
scr_h, 189
set_httk_precision, 190
skeletal_mass, 190
skeletal_mass_children, 190, 191

skin_mass_bosgra, 192
solve_1comp, 192
solve_3comp, 161, 196
solve_fetal_pbtk, 200
solve_gas_pbtk, 166, 203
solve_model, 63, 208
solve_pbtk, 34, 170, 171, 211
spleen_mass_children, 216
supptab1_linakis2020, 217
supptab2_linakis2020, 217
svydesign, 9, 96, 100–102, 129, 130, 137

Tables.Rdata.stamp, 218
tissue.data, 161, 170, 173, 218
tissue_masses_flows, 219
tissue_scale, 220

wambaugh2019, 220
wambaugh2019.nhanes, 222
wambaugh2019.raw, 223
wambaugh2019.seem3, 225
wambaugh2019.tox21, 225
Wang2018 (wang2018), 226
wang2018, 226
well_param, 227
Wetmore2012, 228
wfl, 228