```
2379  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c housemodel-mread-source.cpp -o housemodel-mread-source.o
2380  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c mrgsolve.cpp -o mrgsolve.o
2381  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c mrgsolve_init.cpp -o mrgsolve_init.o
2382  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c odepack_dlsoda.cpp -o odepack_dlsoda.o
2383  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c odeproblem.cpp -o odeproblem.o
2384  #> gfortran -fpic -pipe -O2 -mtune=native -c opk_dlsoda_mrg.f -o opk_dlsoda_mrg.o
2385  #> gfortran -fpic -pipe -O2 -mtune=native -c opkda1_mrg.f -o opkda1_mrg.o
2386  #> gfortran -fpic -pipe -O2 -mtune=native -c opkda2_mrg.f -o opkda2_mrg.o
2387  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c quick.cpp -o quick.o
2388  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -shared -
      L/usr/local/lib/R/lib -L/usr/local/lib -o mrgsolve.so RcppExports.o dataobject.o datarecord.o devtran.o housemodel-
      mread-source.o mrgsolve.o mrgsolve_init.o odepack_dlsoda.o odeproblem.o opk_dlsoda_mrg.o opkda1_mrg.o opkda2_mrg.o
      quick.o -L/usr/local/lib/R/lib -lRlapack -L/usr/local/lib/R/lib -lRblas -lgfortran -lm -lquadmath -lgfortran -lm -
      lquadmath -L/usr/local/lib/R/lib -lR
2389  #> installing to /home/docker/R/mrgsolve/libs
2390  #> ** R
2391  #> ** data
2392  #> ** inst
2393  #> ** preparing package for lazy loading
2394  #> Creating a generic function for ‘as.list’ from package ‘base’ in package ‘mrgsolve’
2395  #> Creating a generic function for ‘as.data.frame’ from package ‘base’ in package ‘mrgsolve’
2396  #> Creating a generic function for ‘as.matrix’ from package ‘base’ in package ‘mrgsolve’
2397  #> Creating a generic function for ‘plot’ from package ‘graphics’ in package ‘mrgsolve’
2398  #> Creating a generic function for ‘labels’ from package ‘base’ in package ‘mrgsolve’
2399  #> Creating a generic function for ‘nrow’ from package ‘base’ in package ‘mrgsolve’
2400  #> Creating a new generic function for ‘rename’ in package ‘mrgsolve’
2401  #> Creating a generic function for ‘tail’ from package ‘utils’ in package ‘mrgsolve’
2402  #> Creating a generic function for ‘head’ from package ‘utils’ in package ‘mrgsolve’
2403  #> Creating a generic function for ‘subset’ from package ‘base’ in package ‘mrgsolve’
2404  #> Creating a generic function for ‘summary’ from package ‘base’ in package ‘mrgsolve’
2405  #> Creating a generic function for ‘update’ from package ‘stats’ in package ‘mrgsolve’
2406  #> ** help
2407  #> *** installing help indices
2408  #> ** building package indices
2409  #> ** testing if installed package can be loaded
2410  #> * creating tarball
2411  #> packaged installation of ‘mrgsolve’ as ‘mrgsolve_0.8.5_R_x86_64-pc-linux-gnu.tar.gz’
2412  #> * DONE (mrgsolve)
2413  #> >
2414  #> >
```

Running R CMD check

```
2416  #> About to run xvfb-run san.sh mrgsolve_0.8.5.tar.gz
2417  #> * installing to library ‘/home/docker/R’
2418  #> * installing *source* package ‘mrgsolve’ ...
2419  #> ** package ‘mrgsolve’ successfully unpacked and MD5 sums checked
2420  #> ** libs
2421  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c RcppExports.cpp -o RcppExports.o
2422  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c dataobject.cpp -o dataobject.o
2423  #> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -
      I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -
      I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native
      -c datarecord.cpp -o datarecord.o
```

```
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native -c devtran.cpp -o devtran.o
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native -c housemodel-mread-source.cpp -o housemodel-mread-source.o
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native -c mrgsolve.cpp -o mrgsolve.o
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native -c mrgsolve_init.cpp -o mrgsolve_init.o
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native -c odepack_dlsoda.cpp -o odepack_dlsoda.o
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native -c odeproblem.cpp -o odeproblem.o
#> gfortran -fpic -pipe -O2 -mtune=native -c opk_dlsoda_mrg.f -o opk_dlsoda_mrg.o
#> gfortran -fpic -pipe -O2 -mtune=native -c opkda1_mrg.f -o opkda1_mrg.o
#> gfortran -fpic -pipe -O2 -mtune=native -c opkda2_mrg.f -o opkda2_mrg.o
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -I/usr/local/lib/R/include -DNDEBUG -I../inst/include -I/usr/local/include -I"/home/docker/R/Rcpp/include" -I"/home/docker/R/RcppArmadillo/include" -I"/home/docker/R/BH/include" -fpic -pipe -Wall -pedantic -O2 -mtune=native -c quick.cpp -o quick.o
#> g++ -fsanitize=address,undefined,bounds-strict -fno-omit-frame-pointer -fno-sanitize=object-size,vptr -shared -L/usr/local/lib/R/lib -L/usr/local/lib -o mrgsolve.so RcppExports.o dataobject.o datarecord.o devtran.o housemodel-mread-source.o mrgsolve.o mrgsolve_init.o odepack_dlsoda.o odeproblem.o opk_dlsoda_mrg.o opkda1_mrg.o opkda2_mrg.o quick.o -L/usr/local/lib/R/lib -lRlapack -L/usr/local/lib/R/lib -lRblas -lgfortran -lm -lquadmath -lgfortran -lm -lquadmath -L/usr/local/lib/R/lib -lR
#> installing to /home/docker/R/mrgsolve/libs
#> ** R
#> ** data
#> ** inst
#> ** preparing package for lazy loading
#> Creating a generic function for ‘as.list’ from package ‘base’ in package ‘mrgsolve’
#> Creating a generic function for ‘as.data.frame’ from package ‘base’ in package ‘mrgsolve’
#> Creating a generic function for ‘as.matrix’ from package ‘base’ in package ‘mrgsolve’
#> Creating a generic function for ‘plot’ from package ‘graphics’ in package ‘mrgsolve’
#> Creating a generic function for ‘labels’ from package ‘base’ in package ‘mrgsolve’
#> Creating a generic function for ‘nrow’ from package ‘base’ in package ‘mrgsolve’
#> Creating a new generic function for ‘rename’ in package ‘mrgsolve’
#> Creating a generic function for ‘tail’ from package ‘utils’ in package ‘mrgsolve’
#> Creating a generic function for ‘head’ from package ‘utils’ in package ‘mrgsolve’
#> Creating a generic function for ‘subset’ from package ‘base’ in package ‘mrgsolve’
#> Creating a generic function for ‘summary’ from package ‘base’ in package ‘mrgsolve’
#> Creating a generic function for ‘update’ from package ‘stats’ in package ‘mrgsolve’
#> ** help
#> *** installing help indices
#> ** building package indices
#> ** testing if installed package can be loaded
#> * DONE (mrgsolve)
#> Running tests
#> R Under development (unstable) (2016-09-18 r71304) -- "Unsuffered Consequences"
#> Copyright (C) 2016 The R Foundation for Statistical Computing
#> Platform: x86_64-pc-linux-gnu (64-bit)
#> R is free software and comes with ABSOLUTELY NO WARRANTY.
#> You are welcome to redistribute it under certain conditions.
#> Type 'license()' or 'licence()' for distribution details.
#> Natural language support but running in an English locale
#> R is a collaborative project with many contributors.
#> Type 'contributors()' for more information and
#> 'citation()' on how to cite R or R packages in publications.
#> Type 'demo()' for some demos, 'help()' for on-line help, or
#> 'help.start()' for an HTML browser interface to help.
#> Type 'q()' to quit R.
#> > tools:::.runPackageTestsR()
#> Running ‘testthat.R’
#> R Under development (unstable) (2016-09-18 r71304) -- "Unsuffered Consequences"
#> Copyright (C) 2016 The R Foundation for Statistical Computing
#> Platform: x86_64-pc-linux-gnu (64-bit)
#> R is free software and comes with ABSOLUTELY NO WARRANTY.
```

```
#> You are welcome to redistribute it under certain conditions.
#> Type 'license()' or 'licence()' for distribution details.
#> R is a collaborative project with many contributors.
#> Type 'contributors()' for more information and
#> 'citation()' on how to cite R or R packages in publications.
#> Type 'demo()' for some demos, 'help()' for on-line help, or
#> 'help.start()' for an HTML browser interface to help.
#> Type 'q()' to quit R.
#> > # Copyright (C) 2013 - 2017 Metrum Research Group, LLC
#> > #
#> > # This file is part of mrgsolve.
#> > #
#> > # mrgsolve is free software: you can redistribute it and/or modify it
#> > # under the terms of the GNU General Public License as published by
#> > # the Free Software Foundation, either version 2 of the License, or
#> > # (at your option) any later version.
#> > #
#> > # mrgsolve is distributed in the hope that it will be useful, but
#> > # WITHOUT ANY WARRANTY; without even the implied warranty of
#> > # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
#> > # GNU General Public License for more details.
#> > #
#> > # You should have received a copy of the GNU General Public License
#> > # along with mrgsolve. If not, see .
#> >
#> > Sys.setenv("R_TESTS" = "")
#> > library(testthat)
#> > library(mrgsolve)
#> >
#> > test_check("mrgsolve", reporter="summary")
#> Attaching package: 'dplyr'
#> The following object is masked from 'package:testthat':
#> matches
#> The following objects are masked from 'package:stats':
#> filter, lag
#> The following objects are masked from 'package:base':
#> intersect, setdiff, setequal, union
#> Lagged doses: .....
#> test-annot: ..................................................
#> Dedicated bioavailability tests: ...
#> test-cache: ...
#> Testing carry_out: ......
#> Testing carry_out of tran / PK dosing items: ..........
#> Testing data_set: ..............
#> Data set record sorting: ..
#> Tests for dataobject - data: ..
#> ......Use dplyr generics on mrgsims objects: ..............
#> Testing infusion inputs:
#> Test $ENV: ...................
#> ...................CMTN block gives compartment numbers: ..
#> Fixed parameters: ...
#> Infusions: ..........
#> Infusions that end too close to observations.: ..
#> Infusion with ss flag: .
#> test-initials: ..............
#> Loading a model via mread:
#> Running a 1 cmt PK model with initial value of 1000: ......
#> Test knobs: .....
#> Testing matlist operations: ..........
#> test-annot: .....
#> test-modlib models: ........................
#> test-modspec: ..................................
#> Testing new model specification file: ............................
#> valid_data_set tests: ........................
#> Testing NMXML functionality:
#> Testing obsonly and obsaug arguments: .........
#> Test opts: ..
#> Compare PKMODEL with equivalent ODEs.: ...
#> Testing that plots can be generated from output objects:
#> PLUGIN: Rcpp: ...
#> .Rename via carry.out #30: .......
#> test-request: ........
#> Testing various request settings: ............
#> R RNG respected via set.seed(): ....
#> .......test-update: ...........
#> Updates: general and simulation times: .....
#> Test updates: parameters and initials: ....................
```

```
#> Test updates: solver settings: .......
#> Test updates: events: ...
#> Testing parameter updates via data:
#> Testing parameter updates via idata: ..
#> Events-based sim same results as data set-based sim: .
#> Time-varying data items passed in via data set: .
#> util functions: ....
#> Test utils: ......................
#> Set F via F_CMT: ......
#> Set ALAG via ALAG_CMT: ............
#> DONE ===============================================================================
#> >
#> >
#> > proc.time()
#> user system elapsed
#> 30.028 4.768 36.706
#> Running examples
#> > tools:::.createExdotR('mrgsolve', system.file(package = 'mrgsolve'))
#> Extracting from parsed Rd's ..........
#> >
#> >
#> Model: housemodel.cpp
#> Dim: 482 x 4
#> Time: 0 to 120
#> ID: 1
#> ID time RESP CP
#> [1,] 1 0.00 50.00 0.00
#> [2,] 1 0.00 50.00 0.00
#> [3,] 1 0.25 42.29 12.87
#> [4,] 1 0.50 32.69 22.25
#> [5,] 1 0.75 25.29 29.04
#> [6,] 1 1.00 20.05 33.91
#> [7,] 1 1.25 16.45 37.37
#> [8,] 1 1.50 14.01 39.78
#> [[1]]
#> [,1] [,2]
#> [1,] 1 11
#> [2,] 11 3
#> [[2]]
#> [,1] [,2]
#> [1,] 2 22
#> [2,] 22 4
#> [[1]]
#> [,1]
#> [1,] 1
#> [[2]]
#> [,1]
#> [1,] 1
#> [[1]]
#> [,1] [,2] [,3]
#> [1,] 1 0 0
#> [2,] 0 11 0
#> [3,] 0 0 3
#> ID time cmt evid amt ii addl
#> 1 1 0 1 1 100 0 0
#> 2 1 0 1 1 200 0 0
#> 3 2 0 1 1 100 0 0
#> 4 2 0 1 1 200 0 0
#> 5 3 0 1 1 100 0 0
#> 6 3 0 1 1 200 0 0
#> 7 4 24 1 1 300 0 0
#> 8 5 24 1 1 300 0 0
#> 9 6 0 1 1 1000 8 10
#> 10 7 0 1 1 1000 8 10
#> 11 8 0 1 1 1000 8 10
#> ID time cmt evid amt
#> 1 1 0 1 1 100
#> 2 2 0 1 1 200
#> 3 3 0 1 1 300
#> # A tibble: 4 × 5
#> ID end delta add GRP
#>
#> 1 1 24 6 1
#> 2 2 48 6 0
#> 3 3 72 6 1
#> 4 4 96 6 0
#> $GRP_1
```

```
#> start: 0 end: 48 delta: 6 offset: 0 min: 0 max: 111
#> $GRP_0
#> start: 0 end: 24 delta: 6 offset: 0 min: 0 max: 135
#> attr(,"descol")
#> [1] "GRP"
#> $GRP_1
#> [1] 0 6 12 18 24 30 36 42 48 111
#> $GRP_0
#> [1] 0 6 12 18 24 122 124 135
#> $ID_1
#> [1] 0 6 12 18 24 122 124 135
#> $ID_2
#> [1] 0 6 12 18 24 30 36 42 48 111
#> $ID_3
#> [1] 0 6 12 18 24 30 36 42 48 54 60 66 72 99
#> $ID_4
#> [1] 0 6 12 18 24 30 36 42 48 54 60 66 72 78 84 88 90 96
#> time cmt amt evid rate ii addl ID arm
#> 1 0 1 100 1 0 0 0 1 2
#> 2 0 1 300 1 100 12 10 2 1
#> 3 0 1 100 1 0 0 0 3 2
#> 4 0 1 300 1 100 12 10 4 1
#> 5 0 1 100 1 0 0 0 5 2
#> 6 0 1 300 1 100 12 10 6 1
#> 7 0 1 100 1 0 0 0 7 2
#> 8 0 1 300 1 100 12 10 8 1
#> 9 0 1 100 1 0 0 0 9 2
#> 10 0 1 300 1 100 12 10 10 1
#> Model file: housemodel.cpp
#> $PARAM
#> @annotated
#> CL : 1 : Clearance (L/hr)
#> VC : 20 : Volume of distribution (L)
#> KA : 1.2 : Absorption rate constant (1/hr)
#> F1 : 1.0 : Bioavailability fraction (.)
#> WT : 70 : Weight (kg)
#> SEX : 0 : Covariate female sex
#> WTCL : 0.75 : Exponent WT on CL
#> WTVC : 1.00 : Exponent WT on VC
#> SEXCL: 0.7 : Prop cov effect on CL
#> SEXVC: 0.85 : Prop cov effect on VC
#> KIN : 100 : Resp prod rate constant (1/hr)
#> KOUT : 2 : Resp elim rate constant (1/hr)
#> IC50 : 10 : Conc giving 50% max resp (ng/ml)
#> $MAIN
#> F_GUT = F1;
#> double CLi = exp(log(CL) + WTCL*log(WT/70) + log(SEXCL)*SEX + ECL);
#> double VCi = exp(log(VC) + WTVC*log(WT/70) + log(SEXVC)*SEX + EVC);
#> double KAi = exp(log(KA) + EKA);
#> double KOUTi = exp(log(KOUT) + EKOUT);
#> RESP_0 = KIN/KOUTi;
#> $ODE
#> dxdt_GUT = -KAi*GUT;
#> dxdt_CENT = KAi*GUT - (CLi/VCi)*CENT;
#> dxdt_RESP = KIN*(1-INH) - KOUTi*RESP;
#> $TABLE
#> double DV = CP*exp(EXPO);
#> Model file: housemodel.cpp
#> $PARAM
#> @annotated
#> CL : 1 : Clearance (L/hr)
#> VC : 20 : Volume of distribution (L)
#> KA : 1.2 : Absorption rate constant (1/hr)
#> F1 : 1.0 : Bioavailability fraction (.)
#> WT : 70 : Weight (kg)
#> SEX : 0 : Covariate female sex
#> WTCL : 0.75 : Exponent WT on CL
#> WTVC : 1.00 : Exponent WT on VC
#> SEXCL: 0.7 : Prop cov effect on CL
#> SEXVC: 0.85 : Prop cov effect on VC
#> KIN : 100 : Resp prod rate constant (1/hr)
#> KOUT : 2 : Resp elim rate constant (1/hr)
#> IC50 : 10 : Conc giving 50% max resp (ng/ml)
#> $TABLE
#> double DV = CP*exp(EXPO);
#> [,1] [,2] [,3]
#> [1,] 0.1 0.0 0.0
```

```
#> [2,] 0.0 0.2 0.0
#> [3,] 0.0 0.0 0.3
#> [,1] [,2]
#> [1,] 0.50 0.01
#> [2,] 0.01 0.20
#> [,1] [,2]
#> [1,] 0.5000000 0.2751182
#> [2,] 0.2751182 0.2000000
#> [1] 2
#> [1] "A" "B" "C"
#> [1] "A" "B" "C"
#> [1] "A" "B" "C"
#> Model: housemodel.cpp
#> Dim: 2410 x 7
#> Time: 0 to 120
#> ID: 5
#> ID time GUT CENT RESP DV CP
#> [1,] 2 0.00 0.000 0.000 50.00 0.0000 0.0000
#> [2,] 2 0.00 10.000 0.000 50.00 0.0000 0.0000
#> [3,] 2 0.25 7.408 2.575 49.86 0.1287 0.1287
#> [4,] 2 0.50 5.488 4.450 49.56 0.2225 0.2225
#> [5,] 2 0.75 4.066 5.808 49.23 0.2904 0.2904
#> [6,] 2 1.00 3.012 6.783 48.93 0.3391 0.3391
#> [7,] 2 1.25 2.231 7.474 48.67 0.3737 0.3737
#> [8,] 2 1.50 1.653 7.956 48.46 0.3978 0.3978
#> ID amt cmt time addl ii rate evid
#> 1 1 1000 1 0 3 24 0 1
#> 2 2 1000 2 0 0 0 20 1
#> 3 3 1000 1 0 0 0 0 1
#> 4 3 500 1 24 0 0 0 1
#> 5 3 500 1 48 0 0 0 1
#> 6 3 1000 1 72 0 0 0 1
#> Model: housemodel.cpp
#> Dim: 2414 x 7
#> Time: 0 to 120
#> ID: 5
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.00 0.0 0.0 50.00 0.00 0.00
#> [2,] 1 0.00 1000.0 0.0 50.00 0.00 0.00
#> [3,] 1 0.25 740.8 257.5 42.29 12.87 12.87
#> [4,] 1 0.50 548.8 445.0 32.69 22.25 22.25
#> [5,] 1 0.75 406.6 580.8 25.29 29.04 29.04
#> [6,] 1 1.00 301.2 678.3 20.05 33.91 33.91
#> [7,] 1 1.25 223.1 747.4 16.45 37.37 37.37
#> [8,] 1 1.50 165.3 795.6 14.01 39.78 39.78
#> Model: housemodel.cpp
#> Dim: 2414 x 7
#> Time: 0 to 120
#> ID: 5
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.00 0.0 0.0 50.00 0.00 0.00
#> [2,] 1 0.00 1000.0 0.0 50.00 0.00 0.00
#> [3,] 1 0.25 740.8 257.5 42.29 12.87 12.87
#> [4,] 1 0.50 548.8 445.0 32.69 22.25 22.25
#> [5,] 1 0.75 406.6 580.8 25.29 29.04 29.04
#> [6,] 1 1.00 301.2 678.3 20.05 33.91 33.91
#> [7,] 1 1.25 223.1 747.4 16.45 37.37 37.37
#> [8,] 1 1.50 165.3 795.6 14.01 39.78 39.78
#> time cmt amt evid ii addl
#> 1 0 1 100 1 168 3
#> 2 48 1 100 1 168 3
#> 3 96 1 100 1 168 3
#> time cmt amt evid ii addl
#> 1 24 1 50 1 168 23
#> 2 72 1 100 1 168 23
#> Events:
#> time cmt amt evid
#> 1 0 1 1000 1
#> Events:
#> time cmt amt evid ii addl
#> 1 0 1 1000 1 0 0
#> 2 12 1 500 1 12 10
#> Events:
#> ID time cmt amt evid
#> 1 1 0 1 100 1
#> 2 2 0 1 100 1
#> 3 3 0 1 100 1
```

```
#> 4 4 0 1 100 1
#> 5 5 0 1 100 1
#> 6 6 0 1 100 1
#> 7 7 0 1 100 1
#> 8 8 0 1 100 1
#> 9 9 0 1 100 1
#> 10 10 0 1 100 1
#> Model: housemodel.cpp
#> Dim: 132 x 3
#> Time: 0 to 24.65
#> ID: 12
#> ID time CP
#> [1,] 1 0.00 0.00000
#> [2,] 1 0.25 0.04552
#> [3,] 1 0.57 0.08624
#> [4,] 1 1.12 0.12643
#> [5,] 1 2.02 0.15072
#> [6,] 1 3.82 0.15121
#> [7,] 1 5.10 0.14348
#> [8,] 1 7.03 0.13101
#> Model: housemodel.cpp
#> Dim: 4814 x 3
#> Time: 0 to 240
#> ID: 5
#> ID time CP
#> [1,] 1 0.00 0.00
#> [2,] 1 0.00 0.00
#> [3,] 1 0.25 12.87
#> [4,] 1 0.50 22.25
#> [5,] 1 0.75 29.04
#> [6,] 1 1.00 33.91
#> [7,] 1 1.25 37.37
#> [8,] 1 1.50 39.78
#> Loading required package: dplyr
#> Attaching package: ‘dplyr’
#> The following objects are masked from ‘package:stats’:
#> filter, lag
#> The following objects are masked from ‘package:base’:
#> intersect, setdiff, setequal, union
#> Joining, by = "ID"
#> ID amt cmt time addl ii rate evid CL
#> 1 1 1000 1 0 3 24 0 1 1.252153
#> 2 2 1000 2 0 0 0 20 1 1.313282
#> 3 3 1000 1 0 0 0 0 1 1.991284
#> 4 3 500 1 24 0 0 0 1 1.991284
#> 5 3 500 1 48 0 0 0 1 1.991284
#> 6 3 1000 1 72 0 0 0 1 1.991284
#> 7 4 2000 2 0 2 48 100 1 2.117796
#> 8 5 1000 1 0 0 0 0 1 1.475523
#> 9 5 5000 1 24 0 0 60 1 1.475523
#> Model: housemodel.cpp
#> Dim: 4814 x 4
#> Time: 0 to 240
#> ID: 5
#> ID time CL CP
#> [1,] 1 0.00 1.252 0.00
#> [2,] 1 0.00 1.252 0.00
#> [3,] 1 0.25 1.252 12.85
#> [4,] 1 0.50 1.252 22.18
#> [5,] 1 0.75 1.252 28.89
#> [6,] 1 1.00 1.252 33.66
#> [7,] 1 1.25 1.252 37.01
#> [8,] 1 1.50 1.252 39.30
#> detaching ‘package:dplyr’
#> Model file: housemodel.cpp
#> $PROB
#>
#> # `mrgsolve` housemodel
#>
#> This model is compiled with `mrgsolve`.
#>
#> - Author: Metrum Research Group, LLC
#> - Description: Generic indirect response PK/PD model
#> - Covariates: Weight, female sex
#> - Random effects: CL, VC, KA, KOUT
#> - Error model: exponential
#>
```

```
2862   #>
2863   #>
2864   #> $PLUGIN base
2865   #>
2866   #> $PARAM @annotated
2867   #> CL : 1 : Clearance (L/hr)
2868   #> VC : 20 : Volume of distribution (L)
2869   #> KA : 1.2 : Absorption rate constant (1/hr)
2870   #> F1 : 1.0 : Bioavailability fraction (.)
2871   #> WT : 70 : Weight (kg)
2872   #> SEX : 0 : Covariate female sex
2873   #> WTCL : 0.75 : Exponent WT on CL
2874   #> WTVC : 1.00 : Exponent WT on VC
2875   #> SEXCL: 0.7 : Prop cov effect on CL
2876   #> SEXVC: 0.85 : Prop cov effect on VC
2877   #> KIN : 100 : Resp prod rate constant (1/hr)
2878   #> KOUT : 2 : Resp elim rate constant (1/hr)
2879   #> IC50 : 10 : Conc giving 50% max resp (ng/ml)
2880   #>
2881   #> $CMT @annotated
2882   #> GUT : Dosing compartment (mg)
2883   #> CENT : Central compartment (mg)
2884   #> RESP : Response (unitless)
2885   #>
2886   #> $OMEGA @labels ECL EVC EKA EKOUT
2887   #> 0 0 0 0
2888   #>
2889   #> $SIGMA @labels EXPO
2890   #> 0
2891   #>
2892   #> $SET end=120, delta=0.25
2893   #>
2894   #>
2895   #> $GLOBAL
2896   #> #define CP (CENT/VCi)
2897   #> #define INH (CP/(IC50+CP))
2898   #>
2899   #> typedef double localdouble;
2900   #>
2901   #> $MAIN
2902   #> F_GUT = F1;
2903   #>
2904   #> double CLi = exp(log(CL) + WTCL*log(WT/70) + log(SEXCL)*SEX + ECL);
2905   #> double VCi = exp(log(VC) + WTVC*log(WT/70) + log(SEXVC)*SEX + EVC);
2906   #> double KAi = exp(log(KA) + EKA);
2907   #> double KOUTi = exp(log(KOUT) + EKOUT);
2908   #>
2909   #> RESP_0 = KIN/KOUTi;
2910   #>
2911   #> $ODE
2912   #> dxdt_GUT = -KAi*GUT;
2913   #> dxdt_CENT = KAi*GUT - (CLi/VCi)*CENT;
2914   #> dxdt_RESP = KIN*(1-INH) - KOUTi*RESP;
2915   #>
2916   #> $TABLE
2917   #> double DV = CP*exp(EXPO);
2918   #>
2919   #> $CAPTURE @annotated
2920   #> DV: Dependent variable (ng/ml)
2921   #> CP: Plasma concentration (ng/ml)
2922   #>
2923   #> ID CL VC KA KOUT IC50 FOO
2924   #> 1 1 1.050 47.80 0.8390 2.450 1.280 4
2925   #> 2 2 0.730 30.10 0.0684 2.510 1.840 6
2926   #> 3 3 2.820 23.80 0.1180 3.880 2.480 5
2927   #> 4 4 0.552 26.30 0.4950 1.180 0.977 2
2928   #> 5 5 0.483 4.36 0.1220 2.350 0.483 10
2929   #> 6 6 3.620 39.80 0.1260 1.890 4.240 1
2930   #> 7 7 0.395 12.10 0.0317 1.250 0.802 8
2931   #> 8 8 1.440 31.20 0.0931 4.030 1.310 7
2932   #> 9 9 2.570 18.20 0.0570 0.862 1.950 3
2933   #> 10 10 2.000 6.51 0.1540 3.220 0.699 9
2934   #> Model: housemodel.cpp
2935   #> Dim: 4810 x 7
2936   #> Time: 0 to 120
2937   #> ID: 10
2938   #> ID time GUT CENT RESP DV CP
```

```
#> [1,] 1 0.00 0 0 40.82 0 0
#> [2,] 1 0.25 0 0 40.82 0 0
#> [3,] 1 0.50 0 0 40.82 0 0
#> [4,] 1 0.75 0 0 40.82 0 0
#> [5,] 1 1.00 0 0 40.82 0 0
#> [6,] 1 1.25 0 0 40.82 0 0
#> [7,] 1 1.50 0 0 40.82 0 0
#> [8,] 1 1.75 0 0 40.82 0 0
#> Model: housemodel.cpp
#> Dim: 4810 x 7
#> Time: 0 to 120
#> ID: 10
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.00 0 0 40.82 0 0
#> [2,] 1 0.25 0 0 40.82 0 0
#> [3,] 1 0.50 0 0 40.82 0 0
#> [4,] 1 0.75 0 0 40.82 0 0
#> [5,] 1 1.00 0 0 40.82 0 0
#> [6,] 1 1.25 0 0 40.82 0 0
#> [7,] 1 1.50 0 0 40.82 0 0
#> [8,] 1 1.75 0 0 40.82 0 0
#> Model initial conditions (N=3):
#> name value . name value
#> CENT (2) 0 | RESP (3) 50
#> GUT (1) 0 | . ... .
#> Model initial conditions (N=3):
#> name value . name value
#> CENT (2) 0 | . ... .
#> [1] "cmt_list"
#> attr(,"package")
#> [1] "mrgsolve"
#> [1] 0
#> $GUT
#> [1] 0
#> $CENT
#> [1] 0
#> $RESP
#> [1] 50
#> GUT CENT RESP
#> 1 0 0 50
#> Model: housemodel
#> Batch (head):
#> ID CL
#> 1 1 1
#> 2 2 2
#> 3 3 3
#> [ CL ]
#> Head:
#> ID time GUT CENT RESP DV CP CL
#> 1 1 0.00 1000.0000 0.0000 50.00000 0.00000 0.00000 1
#> 2 1 0.25 740.8182 257.4883 42.29342 12.87441 12.87441 1
#> 3 1 0.50 548.8116 445.0417 32.69422 22.25208 22.25208 1
#> 4 1 0.75 406.5697 580.8258 25.29397 29.04129 29.04129 1
#> 5 1 1.00 301.1942 678.2976 20.05159 33.91488 33.91488 1
#> [ CL ]Dropping knobs: VC
#> Model: housemodel
#> Batch (head):
#> ID CL
#> 1 1 1
#> 2 2 2
#> 3 3 3
#> [ CL ]
#> Head:
#> ID time GUT CENT RESP DV CP CL
#> 1 1 0.00 0 0 50 0 0 1
#> 2 1 0.25 0 0 50 0 0 1
#> 3 1 0.50 0 0 50 0 0 1
#> 4 1 0.75 0 0 50 0 0 1
#> 5 1 1.00 0 0 50 0 0 1
#> [ CL ]Model: housemodel
#> Batch (head):
#> ID CL
#> 1 1 1
#> 2 2 2
#> 3 3 3
#> [ CL ]
#> Head:
```

```
#> ID time GUT CENT RESP DV CP CL
#> 1 1 0.00 0 0 50 0 0 1
#> 2 1 0.25 0 0 50 0 0 1
#> 3 1 0.50 0 0 50 0 0 1
#> 4 1 0.75 0 0 50 0 0 1
#> 5 1 1.00 0 0 50 0 0 1
#> [ CL ] [,1] [,2] [,3]
#> [1,] 1 0.0 0
#> [2,] 0 2.2 0
#> [3,] 0 0.0 333
#> [,1] [,2]
#> [1,] 1.0 1.1
#> [2,] 1.1 2.2
#> [,1] [,2] [,3] [,4]
#> [1,] 0 0 0 0
#> [2,] 0 0 0 0
#> [3,] 0 0 0 0
#> [4,] 0 0 0 0
#> [,1] [,2]
#> [1,] 1.100000 1.022052
#> [2,] 1.022052 2.200000
#> [,1] [,2]
#> [1,] 1.000 0.657
#> [2,] 0.657 1.000
#> Model: housemodel.cpp
#> Dim: 90 x 7
#> Time: 0 to 22
#> ID: 1
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.00 0.0 0.0 50.00 0.00 0.00
#> [2,] 1 0.00 1000.0 0.0 50.00 0.00 0.00
#> [3,] 1 0.25 740.8 257.5 42.29 12.87 12.87
#> [4,] 1 0.50 548.8 445.0 32.69 22.25 22.25
#> [5,] 1 0.75 406.6 580.8 25.29 29.04 29.04
#> [6,] 1 1.00 301.2 678.3 20.05 33.91 33.91
#> [7,] 1 1.25 223.1 747.4 16.45 37.37 37.37
#> [8,] 1 1.50 165.3 795.6 14.01 39.78 39.78
#> Model: housemodel.cpp
#> Dim: 132 x 7
#> Time: 0 to 24.65
#> ID: 12
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.00 4.020000 0.000 50.00 0.00000 0.00000
#> [2,] 1 0.25 2.978089 1.035 49.95 0.04552 0.04552
#> [3,] 1 0.57 2.028470 1.961 49.81 0.08624 0.08624
#> [4,] 1 1.12 1.048417 2.875 49.57 0.12643 0.12643
#> [5,] 1 2.02 0.356038 3.428 49.33 0.15072 0.15072
#> [6,] 1 3.82 0.041060 3.439 49.25 0.15121 0.15121
#> [7,] 1 5.10 0.008838 3.263 49.28 0.14348 0.14348
#> [8,] 1 7.03 0.000872 2.980 49.34 0.13101 0.13101
#> Model: housemodel.cpp
#> Dim: 120 x 7
#> Time: 0.25 to 24.65
#> ID: 12
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.25 2.978e+00 1.035 49.95 0.04552 0.04552
#> [2,] 1 0.57 2.028e+00 1.961 49.81 0.08624 0.08624
#> [3,] 1 1.12 1.048e+00 2.875 49.57 0.12643 0.12643
#> [4,] 1 2.02 3.560e-01 3.428 49.33 0.15072 0.15072
#> [5,] 1 3.82 4.106e-02 3.439 49.25 0.15121 0.15121
#> [6,] 1 5.10 8.838e-03 3.263 49.28 0.14348 0.14348
#> [7,] 1 7.03 8.720e-04 2.980 49.34 0.13101 0.13101
#> [8,] 1 9.05 7.723e-05 2.703 49.40 0.11884 0.11884
#> Model: housemodel.cpp
#> Dim: 5904 x 8
#> Time: 0 to 120
#> ID: 12
#> ID time a.u.g GUT CENT RESP DV CP
#> [1,] 1 0.00 1 0.000 0.000 50.00 0.00000 0.00000
#> [2,] 1 0.00 0 4.020 0.000 50.00 0.00000 0.00000
#> [3,] 1 0.25 1 2.978 1.035 49.95 0.04552 0.04552
#> [4,] 1 0.25 0 2.978 1.035 49.95 0.04552 0.04552
#> [5,] 1 0.50 1 2.206 1.790 49.84 0.07870 0.07870
#> [6,] 1 0.57 0 2.028 1.961 49.81 0.08624 0.08624
#> [7,] 1 0.75 1 1.634 2.337 49.73 0.10274 0.10274
#> [8,] 1 1.00 1 1.211 2.729 49.61 0.12001 0.12001
#> Model: housemodel.cpp
```

```
#> Dim: 482 x 5
#> Time: 0 to 120
#> ID: 1
#> ID time CENT DV CP
#> [1,] 1 0.00 0.0 0.00 0.00
#> [2,] 1 0.00 0.0 0.00 0.00
#> [3,] 1 0.25 257.5 12.87 12.87
#> [4,] 1 0.50 445.0 22.25 22.25
#> [5,] 1 0.75 580.8 29.04 29.04
#> [6,] 1 1.00 678.3 33.91 33.91
#> [7,] 1 1.25 747.4 37.37 37.37
#> [8,] 1 1.50 795.6 39.78 39.78
#> Model: housemodel.cpp
#> Dim: 482 x 4
#> Time: 0 to 120
#> ID: 1
#> ID time RESP CP
#> [1,] 1 0.00 50.00 0.00
#> [2,] 1 0.00 50.00 0.00
#> [3,] 1 0.25 42.29 12.87
#> [4,] 1 0.50 32.69 22.25
#> [5,] 1 0.75 25.29 29.04
#> [6,] 1 1.00 20.05 33.91
#> [7,] 1 1.25 16.45 37.37
#> [8,] 1 1.50 14.01 39.78
#> [1] "mrgsims"
#> attr(,"package")
#> [1] "mrgsolve"
#> Model: housemodel.cpp
#> Dim: 481 x 7
#> Time: 0 to 120
#> ID: 1
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.00 100.00 0.00 50.00 0.000 0.000
#> [2,] 1 0.25 74.08 25.75 48.68 1.287 1.287
#> [3,] 1 0.50 54.88 44.50 46.18 2.225 2.225
#> [4,] 1 0.75 40.66 58.08 43.61 2.904 2.904
#> [5,] 1 1.00 30.12 67.83 41.38 3.391 3.391
#> [6,] 1 1.25 22.31 74.74 39.58 3.737 3.737
#> [7,] 1 1.50 16.53 79.56 38.18 3.978 3.978
#> [8,] 1 1.75 12.25 82.83 37.14 4.141 4.141
#> Model: housemodel
#> ID time GUT CENT RESP DV CP
#> 1 1 0.00 100.00000 0.00000 50.00000 0.000000 0.000000
#> 2 1 0.25 74.08182 25.74883 48.68223 1.287441 1.287441
#> 3 1 0.50 54.88116 44.50417 46.18005 2.225208 2.225208
#> 4 1 0.75 40.65697 58.08258 43.61333 2.904129 2.904129
#> 5 1 1.00 30.11942 67.82976 41.37943 3.391488 3.391488
#> 6 1 1.25 22.31302 74.74256 39.57649 3.737128 3.737128
#> Model: housemodel
#> ID time GUT CENT RESP DV CP
#> 476 1 118.75 9.202240e-44 0.2753340 49.92950 0.01376670 0.01376670
#> 477 1 119.00 5.342789e-44 0.2719137 49.93038 0.01359569 0.01359569
#> 478 1 119.25 2.453278e-44 0.2685360 49.93124 0.01342680 0.01342680
#> 479 1 119.50 2.498865e-44 0.2652002 49.93209 0.01326001 0.01326001
#> 480 1 119.75 1.869677e-44 0.2619058 49.93293 0.01309529 0.01309529
#> 481 1 120.00 1.163038e-44 0.2586523 49.93377 0.01293262 0.01293262
#> -------- mrgsolve model object (unix) --------
#> Project: /home/docker/R/mrgsolve/project
#> source: housemodel.cpp
#> shared object: mrgsolve
#> compile date:
#> Time: start: 0 end: 120 delta: 0.25
#> > add:
#> > tscale: 1
#> Compartments: GUT CENT RESP [3]
#> Parameters: CL VC KA F1 WT SEX
#> > WTCL WTVC SEXCL SEXVC KIN KOUT
#> > IC50 [13]
#> Omega: 4x4
#> Sigma: 1x1
#> Solver: atol: 1e-08 rtol: 1e-08
#> > maxsteps: 2000 hmin: 0 hmax: 0
#> [1] 481 7
#> [1] "ID" "time" "GUT" "CENT" "RESP" "DV" "CP"
#> [1] 0.00000000 1.28744129 2.22520840 2.90412917 3.39148807 3.73712819
#> [7] 3.97797182 4.14136927 4.24757982 4.31160957 4.34457305 4.35470182
```

```
#> [13] 4.34809176 4.32925658 4.30153797 4.26740933 4.22870090 4.18676645
#> [19] 4.14260681 4.09696135 4.05037581 4.00325248 3.95588742 3.90849799
#> [25] 3.86124331 3.81423936 3.76757018 3.72129619 3.67546029 3.63009249
#> [31] 3.58521323 3.54083588 3.49696861 3.45361577 3.41077886 3.36845734
#> [37] 3.32664914 3.28535111 3.24455927 3.20426913 3.16447573 3.12517389
#> [43] 3.08635822 3.04802320 3.01016327 2.97277280 2.93584619 2.89937783
#> [49] 2.86336215 2.82779361 2.79266673 2.75797606 2.72371622 2.68988189
#> [55] 2.65646780 2.62346875 2.59087958 2.55869522 2.52691065 2.49552090
#> [61] 2.46452107 2.43390631 2.40367186 2.37381298 2.34432501 2.31520334
#> [67] 2.28644343 2.25804077 2.22999094 2.20228955 2.17493227 2.14791483
#> [73] 2.12123300 2.09488263 2.06885958 2.04315979 2.01777925 1.99271399
#> [79] 1.96796010 1.94351371 1.91937100 1.89552819 1.87198156 1.84872743
#> [85] 1.82576217 1.80308219 1.78068394 1.75856393 1.73671869 1.71514483
#> [91] 1.69383896 1.67279775 1.65201792 1.63149623 1.61122946 1.59121444
#> [97] 1.57144806 1.55192722 1.53264887 1.51360999 1.49480763 1.47623883
#> [103] 1.45790070 1.43979037 1.42190501 1.40424182 1.38679805 1.36957097
#> [109] 1.35255788 1.33575613 1.31916310 1.30277620 1.28659285 1.27061054
#> [115] 1.25482676 1.23923904 1.22384497 1.20864213 1.19362814 1.17880065
#> [121] 1.16415735 1.14969596 1.13541421 1.12130987 1.10738074 1.09362463
#> [127] 1.08003941 1.06662295 1.05337314 1.04028793 1.02736526 1.01460312
#> [133] 1.00199952 0.98955248 0.97726006 0.96512034 0.95313142 0.94129144
#> [139] 0.92959853 0.91805087 0.90664666 0.89538412 0.88426148 0.87327700
#> [145] 0.86242899 0.85171572 0.84113554 0.83068679 0.82036783 0.81017706
#> [151] 0.80011287 0.79017371 0.78035801 0.77066425 0.76109090 0.75163648
#> [157] 0.74229948 0.73307851 0.72397206 0.71497874 0.70609713 0.69732585
#> [163] 0.68866353 0.68010882 0.67166037 0.66331687 0.65507702 0.64693952
#> [169] 0.63890311 0.63096653 0.62312854 0.61538791 0.60774343 0.60019392
#> [175] 0.59273819 0.58537508 0.57810343 0.57092212 0.56383000 0.55682600
#> [181] 0.54990900 0.54307792 0.53633170 0.52966928 0.52308962 0.51659170
#> [187] 0.51017449 0.50383700 0.49757824 0.49139722 0.48529299 0.47926458
#> [193] 0.47331106 0.46743150 0.46162497 0.45589057 0.45022741 0.44463459
#> [199] 0.43911125 0.43365652 0.42826954 0.42294950 0.41769554 0.41250684
#> [205] 0.40738260 0.40232202 0.39732429 0.39238865 0.38751432 0.38270054
#> [211] 0.37794656 0.37325163 0.36861503 0.36403602 0.35951389 0.35504794
#> [217] 0.35063746 0.34628177 0.34198019 0.33773204 0.33353666 0.32939341
#> [223] 0.32530162 0.32126065 0.31726989 0.31332870 0.30943647 0.30559259
#> [229] 0.30179645 0.29804748 0.29434507 0.29068866 0.28707767 0.28351153
#> [235] 0.27998969 0.27651161 0.27307672 0.26968451 0.26633444 0.26302598
#> [241] 0.25975862 0.25653184 0.25334515 0.25019805 0.24709004 0.24402064
#> [247] 0.24098937 0.23799575 0.23503932 0.23211961 0.22923617 0.22638856
#> [253] 0.22357631 0.22079900 0.21805619 0.21534746 0.21267237 0.21003051
#> [259] 0.20742147 0.20484484 0.20230021 0.19978720 0.19730540 0.19485443
#> [265] 0.19243391 0.19004346 0.18768270 0.18535127 0.18304880 0.18077493
#> [271] 0.17852931 0.17631158 0.17412141 0.17195844 0.16982233 0.16771277
#> [277] 0.16562941 0.16357193 0.16154000 0.15953332 0.15755156 0.15559443
#> [283] 0.15366160 0.15175279 0.14986768 0.14800600 0.14616744 0.14435172
#> [289] 0.14255855 0.14078766 0.13903877 0.13731160 0.13560589 0.13392136
#> [295] 0.13225777 0.13061483 0.12899231 0.12738994 0.12580748 0.12424467
#> [301] 0.12270128 0.12117706 0.11967177 0.11818519 0.11671707 0.11526718
#> [307] 0.11383531 0.11242123 0.11102471 0.10964554 0.10828350 0.10693838
#> [313] 0.10560997 0.10429806 0.10300245 0.10172293 0.10045931 0.09921138
#> [319] 0.09797896 0.09676185 0.09555985 0.09437279 0.09320047 0.09204271
#> [325] 0.09089934 0.08977017 0.08865503 0.08755374 0.08646613 0.08539203
#> [331] 0.08433127 0.08328369 0.08224913 0.08122741 0.08021839 0.07922190
#> [337] 0.07823779 0.07726590 0.07630609 0.07535820 0.07442209 0.07349760
#> [343] 0.07258460 0.07168294 0.07079248 0.06991308 0.06904460 0.06818692
#> [349] 0.06733989 0.06650338 0.06567726 0.06486140 0.06405568 0.06325997
#> [355] 0.06247414 0.06169808 0.06093165 0.06017475 0.05942724 0.05868903
#> [361] 0.05795998 0.05723999 0.05652894 0.05582673 0.05513324 0.05444836
#> [367] 0.05377199 0.05310403 0.05244436 0.05179288 0.05114950 0.05051411
#> [373] 0.04988661 0.04926691 0.04865491 0.04805051 0.04745362 0.04686414
#> [379] 0.04628198 0.04570706 0.04513927 0.04457855 0.04402478 0.04347790
#> [385] 0.04293781 0.04240442 0.04187767 0.04135746 0.04084370 0.04033634
#> [391] 0.03983527 0.03934043 0.03885173 0.03836911 0.03789248 0.03742177
#> [397] 0.03695691 0.03649783 0.03604444 0.03559669 0.03515450 0.03471781
#> [403] 0.03428654 0.03386062 0.03344000 0.03302460 0.03261436 0.03220922
#> [409] 0.03180911 0.03141397 0.03102374 0.03063836 0.03025776 0.02988189
#> [415] 0.02951069 0.02914411 0.02878207 0.02842454 0.02807144 0.02772273
#> [421] 0.02737835 0.02703825 0.02670238 0.02637068 0.02604310 0.02571958
#> [427] 0.02540009 0.02508456 0.02477296 0.02446522 0.02416131 0.02386118
#> [433] 0.02356477 0.02327204 0.02298295 0.02269745 0.02241550 0.02213705
#> [439] 0.02186206 0.02159048 0.02132228 0.02105741 0.02079583 0.02053750
#> [445] 0.02028238 0.02003043 0.01978161 0.01953588 0.01929320 0.01905354
#> [451] 0.01881685 0.01858310 0.01835226 0.01812428 0.01789914 0.01767679
#> [457] 0.01745721 0.01724035 0.01702619 0.01681469 0.01660581 0.01639953
#> [463] 0.01619581 0.01599462 0.01579594 0.01559971 0.01540593 0.01521456
#> [469] 0.01502556 0.01483891 0.01465458 0.01447253 0.01429275 0.01411521
```

```
#> [475] 0.01393986 0.01376670 0.01359569 0.01342680 0.01326001 0.01309529
#> [481] 0.01293262
#> Events:
#> time cmt amt addl ii evid
#> 1 0 1 1000 5 24 1
#> -------- mrgsolve model object (unix) --------
#> Project: /home/docker/R/mrgsolve/project
#> source: housemodel.cpp
#> shared object: mrgsolve
#> compile date:
#> Time: start: 0 end: 120 delta: 0.1
#> > add:
#> > tscale: 1
#> Compartments: GUT CENT RESP [3]
#> Parameters: CL VC KA F1 WT SEX
#> > WTCL WTVC SEXCL SEXVC KIN KOUT
#> > IC50 [13]
#> Omega: 4x4
#> Sigma: 1x1
#> Solver: atol: 1e-08 rtol: 1e-08
#> > maxsteps: 2000 hmin: 0 hmax: 0
#> Model file: housemodel.cpp
#> $PROB
#>
#> # `mrgsolve` housemodel
#>
#> This model is compiled with `mrgsolve`.
#>
#> - Author: Metrum Research Group, LLC
#> - Description: Generic indirect response PK/PD model
#> - Covariates: Weight, female sex
#> - Random effects: CL, VC, KA, KOUT
#> - Error model: exponential
#>
#>
#>
#> $PLUGIN base
#>
#> $PARAM @annotated
#> CL : 1 : Clearance (L/hr)
#> VC : 20 : Volume of distribution (L)
#> KA : 1.2 : Absorption rate constant (1/hr)
#> F1 : 1.0 : Bioavailability fraction (.)
#> WT : 70 : Weight (kg)
#> SEX : 0 : Covariate female sex
#> WTCL : 0.75 : Exponent WT on CL
#> WTVC : 1.00 : Exponent WT on VC
#> SEXCL: 0.7 : Prop cov effect on CL
#> SEXVC: 0.85 : Prop cov effect on VC
#> KIN : 100 : Resp prod rate constant (1/hr)
#> KOUT : 2 : Resp elim rate constant (1/hr)
#> IC50 : 10 : Conc giving 50% max resp (ng/ml)
#>
#> $CMT @annotated
#> GUT : Dosing compartment (mg)
#> CENT : Central compartment (mg)
#> RESP : Response (unitless)
#>
#> $OMEGA @labels ECL EVC EKA EKOUT
#> 0 0 0 0
#>
#> $SIGMA @labels EXPO
#> 0
#>
#> $SET end=120, delta=0.25
#>
#>
#> $GLOBAL
#> #define CP (CENT/VCi)
#> #define INH (CP/(IC50+CP))
#>
#> typedef double localdouble;
#>
#> $MAIN
#> F_GUT = F1;
#>
#> double CLi = exp(log(CL) + WTCL*log(WT/70) + log(SEXCL)*SEX + ECL);
```

```
#> double VCi = exp(log(VC) + WTVC*log(WT/70) + log(SEXVC)*SEX + EVC);
#> double KAi = exp(log(KA) + EKA);
#> double KOUTi = exp(log(KOUT) + EKOUT);
#>
#> RESP_0 = KIN/KOUTi;
#>
#> $ODE
#> dxdt_GUT = -KAi*GUT;
#> dxdt_CENT = KAi*GUT - (CLi/VCi)*CENT;
#> dxdt_RESP = KIN*(1-INH) - KOUTi*RESP;
#>
#> $TABLE
#> double DV = CP*exp(EXPO);
#>
#> $CAPTURE @annotated
#> DV: Dependent variable (ng/ml)
#> CP: Plasma concentration (ng/ml)
#>
#> [1] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0 1.1
#> [13] 1.2 1.3 1.4 1.5 1.6 1.7 1.8 1.9 2.0 2.1 2.2 2.3
#> [25] 2.4 2.5 2.6 2.7 2.8 2.9 3.0 3.1 3.2 3.3 3.4 3.5
#> [37] 3.6 3.7 3.8 3.9 4.0 4.1 4.2 4.3 4.4 4.5 4.6 4.7
#> [49] 4.8 4.9 5.0 5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8 5.9
#> [61] 6.0 6.1 6.2 6.3 6.4 6.5 6.6 6.7 6.8 6.9 7.0 7.1
#> [73] 7.2 7.3 7.4 7.5 7.6 7.7 7.8 7.9 8.0 8.1 8.2 8.3
#> [85] 8.4 8.5 8.6 8.7 8.8 8.9 9.0 9.1 9.2 9.3 9.4 9.5
#> [97] 9.6 9.7 9.8 9.9 10.0 10.1 10.2 10.3 10.4 10.5 10.6 10.7
#> [109] 10.8 10.9 11.0 11.1 11.2 11.3 11.4 11.5 11.6 11.7 11.8 11.9
#> [121] 12.0 12.1 12.2 12.3 12.4 12.5 12.6 12.7 12.8 12.9 13.0 13.1
#> [133] 13.2 13.3 13.4 13.5 13.6 13.7 13.8 13.9 14.0 14.1 14.2 14.3
#> [145] 14.4 14.5 14.6 14.7 14.8 14.9 15.0 15.1 15.2 15.3 15.4 15.5
#> [157] 15.6 15.7 15.8 15.9 16.0 16.1 16.2 16.3 16.4 16.5 16.6 16.7
#> [169] 16.8 16.9 17.0 17.1 17.2 17.3 17.4 17.5 17.6 17.7 17.8 17.9
#> [181] 18.0 18.1 18.2 18.3 18.4 18.5 18.6 18.7 18.8 18.9 19.0 19.1
#> [193] 19.2 19.3 19.4 19.5 19.6 19.7 19.8 19.9 20.0 20.1 20.2 20.3
#> [205] 20.4 20.5 20.6 20.7 20.8 20.9 21.0 21.1 21.2 21.3 21.4 21.5
#> [217] 21.6 21.7 21.8 21.9 22.0 22.1 22.2 22.3 22.4 22.5 22.6 22.7
#> [229] 22.8 22.9 23.0 23.1 23.2 23.3 23.4 23.5 23.6 23.7 23.8 23.9
#> [241] 24.0 24.1 24.2 24.3 24.4 24.5 24.6 24.7 24.8 24.9 25.0 25.1
#> [253] 25.2 25.3 25.4 25.5 25.6 25.7 25.8 25.9 26.0 26.1 26.2 26.3
#> [265] 26.4 26.5 26.6 26.7 26.8 26.9 27.0 27.1 27.2 27.3 27.4 27.5
#> [277] 27.6 27.7 27.8 27.9 28.0 28.1 28.2 28.3 28.4 28.5 28.6 28.7
#> [289] 28.8 28.9 29.0 29.1 29.2 29.3 29.4 29.5 29.6 29.7 29.8 29.9
#> [301] 30.0 30.1 30.2 30.3 30.4 30.5 30.6 30.7 30.8 30.9 31.0 31.1
#> [313] 31.2 31.3 31.4 31.5 31.6 31.7 31.8 31.9 32.0 32.1 32.2 32.3
#> [325] 32.4 32.5 32.6 32.7 32.8 32.9 33.0 33.1 33.2 33.3 33.4 33.5
#> [337] 33.6 33.7 33.8 33.9 34.0 34.1 34.2 34.3 34.4 34.5 34.6 34.7
#> [349] 34.8 34.9 35.0 35.1 35.2 35.3 35.4 35.5 35.6 35.7 35.8 35.9
#> [361] 36.0 36.1 36.2 36.3 36.4 36.5 36.6 36.7 36.8 36.9 37.0 37.1
#> [373] 37.2 37.3 37.4 37.5 37.6 37.7 37.8 37.9 38.0 38.1 38.2 38.3
#> [385] 38.4 38.5 38.6 38.7 38.8 38.9 39.0 39.1 39.2 39.3 39.4 39.5
#> [397] 39.6 39.7 39.8 39.9 40.0 40.1 40.2 40.3 40.4 40.5 40.6 40.7
#> [409] 40.8 40.9 41.0 41.1 41.2 41.3 41.4 41.5 41.6 41.7 41.8 41.9
#> [421] 42.0 42.1 42.2 42.3 42.4 42.5 42.6 42.7 42.8 42.9 43.0 43.1
#> [433] 43.2 43.3 43.4 43.5 43.6 43.7 43.8 43.9 44.0 44.1 44.2 44.3
#> [445] 44.4 44.5 44.6 44.7 44.8 44.9 45.0 45.1 45.2 45.3 45.4 45.5
#> [457] 45.6 45.7 45.8 45.9 46.0 46.1 46.2 46.3 46.4 46.5 46.6 46.7
#> [469] 46.8 46.9 47.0 47.1 47.2 47.3 47.4 47.5 47.6 47.7 47.8 47.9
#> [481] 48.0 48.1 48.2 48.3 48.4 48.5 48.6 48.7 48.8 48.9 49.0 49.1
#> [493] 49.2 49.3 49.4 49.5 49.6 49.7 49.8 49.9 50.0 50.1 50.2 50.3
#> [505] 50.4 50.5 50.6 50.7 50.8 50.9 51.0 51.1 51.2 51.3 51.4 51.5
#> [517] 51.6 51.7 51.8 51.9 52.0 52.1 52.2 52.3 52.4 52.5 52.6 52.7
#> [529] 52.8 52.9 53.0 53.1 53.2 53.3 53.4 53.5 53.6 53.7 53.8 53.9
#> [541] 54.0 54.1 54.2 54.3 54.4 54.5 54.6 54.7 54.8 54.9 55.0 55.1
#> [553] 55.2 55.3 55.4 55.5 55.6 55.7 55.8 55.9 56.0 56.1 56.2 56.3
#> [565] 56.4 56.5 56.6 56.7 56.8 56.9 57.0 57.1 57.2 57.3 57.4 57.5
#> [577] 57.6 57.7 57.8 57.9 58.0 58.1 58.2 58.3 58.4 58.5 58.6 58.7
#> [589] 58.8 58.9 59.0 59.1 59.2 59.3 59.4 59.5 59.6 59.7 59.8 59.9
#> [601] 60.0 60.1 60.2 60.3 60.4 60.5 60.6 60.7 60.8 60.9 61.0 61.1
#> [613] 61.2 61.3 61.4 61.5 61.6 61.7 61.8 61.9 62.0 62.1 62.2 62.3
#> [625] 62.4 62.5 62.6 62.7 62.8 62.9 63.0 63.1 63.2 63.3 63.4 63.5
#> [637] 63.6 63.7 63.8 63.9 64.0 64.1 64.2 64.3 64.4 64.5 64.6 64.7
#> [649] 64.8 64.9 65.0 65.1 65.2 65.3 65.4 65.5 65.6 65.7 65.8 65.9
#> [661] 66.0 66.1 66.2 66.3 66.4 66.5 66.6 66.7 66.8 66.9 67.0 67.1
#> [673] 67.2 67.3 67.4 67.5 67.6 67.7 67.8 67.9 68.0 68.1 68.2 68.3
#> [685] 68.4 68.5 68.6 68.7 68.8 68.9 69.0 69.1 69.2 69.3 69.4 69.5
#> [697] 69.6 69.7 69.8 69.9 70.0 70.1 70.2 70.3 70.4 70.5 70.6 70.7
```

```
#> [709] 70.8 70.9 71.0 71.1 71.2 71.3 71.4 71.5 71.6 71.7 71.8 71.9
#> [721] 72.0 72.1 72.2 72.3 72.4 72.5 72.6 72.7 72.8 72.9 73.0 73.1
#> [733] 73.2 73.3 73.4 73.5 73.6 73.7 73.8 73.9 74.0 74.1 74.2 74.3
#> [745] 74.4 74.5 74.6 74.7 74.8 74.9 75.0 75.1 75.2 75.3 75.4 75.5
#> [757] 75.6 75.7 75.8 75.9 76.0 76.1 76.2 76.3 76.4 76.5 76.6 76.7
#> [769] 76.8 76.9 77.0 77.1 77.2 77.3 77.4 77.5 77.6 77.7 77.8 77.9
#> [781] 78.0 78.1 78.2 78.3 78.4 78.5 78.6 78.7 78.8 78.9 79.0 79.1
#> [793] 79.2 79.3 79.4 79.5 79.6 79.7 79.8 79.9 80.0 80.1 80.2 80.3
#> [805] 80.4 80.5 80.6 80.7 80.8 80.9 81.0 81.1 81.2 81.3 81.4 81.5
#> [817] 81.6 81.7 81.8 81.9 82.0 82.1 82.2 82.3 82.4 82.5 82.6 82.7
#> [829] 82.8 82.9 83.0 83.1 83.2 83.3 83.4 83.5 83.6 83.7 83.8 83.9
#> [841] 84.0 84.1 84.2 84.3 84.4 84.5 84.6 84.7 84.8 84.9 85.0 85.1
#> [853] 85.2 85.3 85.4 85.5 85.6 85.7 85.8 85.9 86.0 86.1 86.2 86.3
#> [865] 86.4 86.5 86.6 86.7 86.8 86.9 87.0 87.1 87.2 87.3 87.4 87.5
#> [877] 87.6 87.7 87.8 87.9 88.0 88.1 88.2 88.3 88.4 88.5 88.6 88.7
#> [889] 88.8 88.9 89.0 89.1 89.2 89.3 89.4 89.5 89.6 89.7 89.8 89.9
#> [901] 90.0 90.1 90.2 90.3 90.4 90.5 90.6 90.7 90.8 90.9 91.0 91.1
#> [913] 91.2 91.3 91.4 91.5 91.6 91.7 91.8 91.9 92.0 92.1 92.2 92.3
#> [925] 92.4 92.5 92.6 92.7 92.8 92.9 93.0 93.1 93.2 93.3 93.4 93.5
#> [937] 93.6 93.7 93.8 93.9 94.0 94.1 94.2 94.3 94.4 94.5 94.6 94.7
#> [949] 94.8 94.9 95.0 95.1 95.2 95.3 95.4 95.5 95.6 95.7 95.8 95.9
#> [961] 96.0 96.1 96.2 96.3 96.4 96.5 96.6 96.7 96.8 96.9 97.0 97.1
#> [973] 97.2 97.3 97.4 97.5 97.6 97.7 97.8 97.9 98.0 98.1 98.2 98.3
#> [985] 98.4 98.5 98.6 98.7 98.8 98.9 99.0 99.1 99.2 99.3 99.4 99.5
#> [997] 99.6 99.7 99.8 99.9 100.0 100.1 100.2 100.3 100.4 100.5 100.6 100.7
#> [1009] 100.8 100.9 101.0 101.1 101.2 101.3 101.4 101.5 101.6 101.7 101.8 101.9
#> [1021] 102.0 102.1 102.2 102.3 102.4 102.5 102.6 102.7 102.8 102.9 103.0 103.1
#> [1033] 103.2 103.3 103.4 103.5 103.6 103.7 103.8 103.9 104.0 104.1 104.2 104.3
#> [1045] 104.4 104.5 104.6 104.7 104.8 104.9 105.0 105.1 105.2 105.3 105.4 105.5
#> [1057] 105.6 105.7 105.8 105.9 106.0 106.1 106.2 106.3 106.4 106.5 106.6 106.7
#> [1069] 106.8 106.9 107.0 107.1 107.2 107.3 107.4 107.5 107.6 107.7 107.8 107.9
#> [1081] 108.0 108.1 108.2 108.3 108.4 108.5 108.6 108.7 108.8 108.9 109.0 109.1
#> [1093] 109.2 109.3 109.4 109.5 109.6 109.7 109.8 109.9 110.0 110.1 110.2 110.3
#> [1105] 110.4 110.5 110.6 110.7 110.8 110.9 111.0 111.1 111.2 111.3 111.4 111.5
#> [1117] 111.6 111.7 111.8 111.9 112.0 112.1 112.2 112.3 112.4 112.5 112.6 112.7
#> [1129] 112.8 112.9 113.0 113.1 113.2 113.3 113.4 113.5 113.6 113.7 113.8 113.9
#> [1141] 114.0 114.1 114.2 114.3 114.4 114.5 114.6 114.7 114.8 114.9 115.0 115.1
#> [1153] 115.2 115.3 115.4 115.5 115.6 115.7 115.8 115.9 116.0 116.1 116.2 116.3
#> [1165] 116.4 116.5 116.6 116.7 116.8 116.9 117.0 117.1 117.2 117.3 117.4 117.5
#> [1177] 117.6 117.7 117.8 117.9 118.0 118.1 118.2 118.3 118.4 118.5 118.6 118.7
#> [1189] 118.8 118.9 119.0 119.1 119.2 119.3 119.4 119.5 119.6 119.7 119.8 119.9
#> [1201] 120.0
#> Model parameters (N=13):
#> name value . name value
#> CL 0.5 | SEXCL 0.7
#> F1 1 | SEXVC 0.85
#> IC50 10 | VC 20
#> KA 1.2 | WT 70
#> KIN 100 | WTCL 0.75
#> KOUT 2 | WTVC 1
#> SEX 0 | . .
#> Model initial conditions (N=3):
#> name value . name value
#> CENT (2) 0 | RESP (3) 50
#> GUT (1) 0 | . ... .
#> Model: housemodel.cpp
#> Dim: 1682 x 7
#> Time: 0 to 168
#> ID: 1
#> ID time GUT CENT RESP DV CP
#> [1,] 1 0.0 0.0 0.0 50.00 0.000 0.000
#> [2,] 1 0.0 1000.0 0.0 50.00 0.000 0.000
#> [3,] 1 0.1 886.9 112.9 48.04 5.647 5.647
#> [4,] 1 0.2 786.6 212.8 44.32 10.641 10.641
#> [5,] 1 0.3 697.7 301.1 40.24 15.056 15.056
#> [6,] 1 0.4 618.8 379.2 36.30 18.958 18.958
#> [7,] 1 0.5 548.8 448.1 32.67 22.405 22.405
#> [8,] 1 0.6 486.8 509.0 29.41 25.448 25.448
#> Model: housemodel
#> ID time GUT CENT RESP DV CP
#> 1 1 0.0 0.0000 0.0000 50.00000 0.000000 0.000000
#> 2 1 0.0 1000.0000 0.0000 50.00000 0.000000 0.000000
#> 3 1 0.1 886.9204 112.9355 48.03655 5.646775 5.646775
#> 4 1 0.2 786.6279 212.8183 44.31899 10.640917 10.640917
#> 5 1 0.3 697.6763 301.1252 40.24379 15.056258 15.056258
#> 6 1 0.4 618.7834 379.1657 36.29753 18.958286 18.958286
#> Model: housemodel
```

```
#> ID time GUT CENT RESP DV CP
#> 1677 1 167.5 1.218836e-22 671.4739 11.36504 33.57369 33.57369
#> 1678 1 167.6 1.077787e-22 669.7973 11.38701 33.48987 33.48987
#> 1679 1 167.7 9.538413e-23 668.1249 11.40901 33.40625 33.40625
#> 1680 1 167.8 8.441714e-23 666.4567 11.43104 33.32283 33.32283
#> 1681 1 167.9 7.466548e-23 664.7926 11.45310 33.23963 33.23963
#> 1682 1 168.0 6.610140e-23 663.1327 11.47519 33.15664 33.15664
#> [1] 1682 7
#> 'data.frame': 1 obs. of 7 variables:
#> $ ID : num 1
#> $ time: num 72
#> $ GUT : num 1000
#> $ CENT: num 1037
#> $ RESP: num 8
#> $ DV : num 51.8
#> $ CP : num 51.8
#> Model: housemodel
#> ID time evid amt cmt GUT CENT RESP DV CP
#> 1 1 0.0 0 0 0 0.0000 0.0000 50.00000 0.000000 0.000000
#> 2 1 0.0 1 1000 1 1000.0000 0.0000 50.00000 0.000000 0.000000
#> 3 1 0.1 0 0 0 886.9204 112.9355 48.03655 5.646775 5.646775
#> 4 1 0.2 0 0 0 786.6279 212.8183 44.31899 10.640917 10.640917
#> 5 1 0.3 0 0 0 697.6763 301.1252 40.24379 15.056258 15.056258
#> 6 1 0.4 0 0 0 618.7834 379.1657 36.29753 18.958286 18.958286
#> Events:
#> time cmt amt rate evid ii addl
#> 1 0 2 500 10 1 0 0
#> 2 54 1 100 0 1 8 10
#> ID amt cmt time addl ii rate evid
#> 1 1 1000 1 0 3 24 0 1
#> 2 2 1000 2 0 0 0 20 1
#> 3 3 1000 1 0 0 0 0 1
#> 4 3 500 1 24 0 0 0 1
#> 5 3 500 1 48 0 0 0 1
#> 6 3 1000 1 72 0 0 0 1
#> 7 4 2000 2 0 2 48 100 1
#> 8 5 1000 1 0 0 0 0 1
#> 9 5 5000 1 24 0 0 60 1
#> ID CL VC KA KOUT IC50 FOO
#> 1 1 1.050 47.80 0.8390 2.450 1.280 4
#> 2 2 0.730 30.10 0.0684 2.510 1.840 6
#> 3 3 2.820 23.80 0.1180 3.880 2.480 5
#> 4 4 0.552 26.30 0.4950 1.180 0.977 2
#> 5 5 0.483 4.36 0.1220 2.350 0.483 10
#> 6 6 3.620 39.80 0.1260 1.890 4.240 1
#> 7 7 0.395 12.10 0.0317 1.250 0.802 8
#> 8 8 1.440 31.20 0.0931 4.030 1.310 7
#> 9 9 2.570 18.20 0.0570 0.862 1.950 3
#> 10 10 2.000 6.51 0.1540 3.220 0.699 9
#> Compiling irm1 ... done.
#> -------- mrgsolve model object (unix) --------
#> Project: /home/docker/R/mrgsolve/models
#> source: irm1.cpp
#> shared object: irm1-so-11ca67a66162
#> compile date:
#> Time: start: 0 end: 24 delta: 1
#> > add:
#> > tscale: 1
#> Compartments: EV1 CENT PERIPH RESP EV2 [5]
#> Parameters: CL VC Q VP KA1 KA2
#> > KIN KOUT IC50 IMAX n VMAX
#> > KM [13]
#> Omega: 0x0
#> Sigma: 0x0
#> Solver: atol: 1e-08 rtol: 1e-08
#> > maxsteps: 2000 hmin: 0 hmax: 0
#> Model: irm1.cpp
#> Dim: 26 x 8
#> Time: 0 to 24
#> ID: 1
#> ID time EV1 CENT PERIPH RESP EV2 CP
#> [1,] 1 0 0.0000 0.0 0.000 5.0000 0 0.000
#> [2,] 1 0 300.0000 0.0 0.000 5.0000 0 0.000
#> [3,] 1 1 110.3638 174.6 9.758 1.8054 0 8.732
#> [4,] 1 2 40.6006 217.7 26.371 0.9420 0 10.884
#> [5,] 1 3 14.9361 217.3 41.468 0.7950 0 10.864
#> [6,] 1 4 5.4947 204.5 53.085 0.8014 0 10.227
```

```
#> [7,] 1 5 2.0214 189.9 61.313 0.8448 0 9.496
#> [8,] 1 6 0.7436 176.6 66.774 0.8969 0 8.829
#> $...
#> [,1]
#> 1: 1
#> $...
#> [,1]
#> 1: 1
#> $...
#> [,1] [,2] [,3]
#> 2: 1 0 0
#> 3: 0 2 0
#> 4: 0 0 3
#> $...
#> [,1] [,2]
#> 5: 0.100 0.002
#> 6: 0.002 0.500
#> $A
#> [,1]
#> 1: 1
#> $B
#> [,1] [,2] [,3]
#> 2: 1 0 0
#> 3: 0 2 0
#> 4: 0 0 3
#> $C
#> [,1] [,2]
#> 5: 0.100 0.002
#> 6: 0.002 0.500
#> $...
#> [,1] [,2] [,3] [,4]
#> ECL: 0.1 0.0 0.0 0.0
#> EVC: 0.0 0.2 0.0 0.0
#> EKA: 0.0 0.0 0.3 0.0
#> EKOUT: 0.0 0.0 0.0 0.4
#> [,1] [,2] [,3] [,4]
#> 1: 0.1 0.0 0.0 0.0
#> 2: 0.0 0.2 0.0 0.0
#> 3: 0.0 0.0 0.3 0.0
#> 4: 0.0 0.0 0.0 0.4
#> Model parameters (N=13):
#> name value . name value
#> CL 1 | SEXCL 0.7
#> F1 1 | SEXVC 0.85
#> IC50 10 | VC 20
#> KA 1.2 | WT 70
#> KIN 100 | WTCL 0.75
#> KOUT 2 | WTVC 1
#> SEX 0 | . .
#> Model parameters (N=13):
#> name value . name value
#> CL 1 | F1 1
#> [1] "parameter_list"
#> attr(,"package")
#> [1] "mrgsolve"
#> [1] 1.2
#> $CL
#> [1] 1
#> $VC
#> [1] 20
#> $KA
#> [1] 1.2
#> $F1
#> [1] 1
#> $WT
#> [1] 70
#> $SEX
#> [1] 0
#> $WTCL
#> [1] 0.75
#> $WTVC
#> [1] 1
#> $SEXCL
#> [1] 0.7
#> $SEXVC
#> [1] 0.85
#> $KIN
```

```
#> [1] 100
#> $KOUT
#> [1] 2
#> $IC50
#> [1] 10
#> CL VC KA F1 WT SEX WTCL WTVC SEXCL SEXVC KIN KOUT IC50
#> 1 1 20 1.2 1 70 0 0.75 1 0.7 0.85 100 2 10
#> ID Wt Dose Time dv
#> 1 1 79.6 4.02 0.00 0.74
#> 2 1 79.6 4.02 0.25 2.84
#> 3 1 79.6 4.02 0.57 6.57
#> 4 1 79.6 4.02 1.12 10.50
#> 5 1 79.6 4.02 2.02 9.66
#> 6 1 79.6 4.02 3.82 8.58
#> 7 1 79.6 4.02 5.10 8.36
#> 8 1 79.6 4.02 7.03 7.47
#> 9 1 79.6 4.02 9.05 6.89
#> 10 1 79.6 4.02 12.12 5.94
#> 11 1 79.6 4.02 24.37 3.28
#> 12 2 72.4 4.40 0.00 0.00
#> 13 2 72.4 4.40 0.27 1.72
#> 14 2 72.4 4.40 0.52 7.91
#> 15 2 72.4 4.40 1.00 8.31
#> 16 2 72.4 4.40 1.92 8.33
#> 17 2 72.4 4.40 3.50 6.85
#> 18 2 72.4 4.40 5.02 6.08
#> 19 2 72.4 4.40 7.03 5.40
#> 20 2 72.4 4.40 9.00 4.55
#> 21 2 72.4 4.40 12.00 3.01
#> 22 2 72.4 4.40 24.30 0.90
#> 23 3 70.5 4.53 0.00 0.00
#> 24 3 70.5 4.53 0.27 4.40
#> 25 3 70.5 4.53 0.58 6.90
#> 26 3 70.5 4.53 1.02 8.20
#> 27 3 70.5 4.53 2.02 7.80
#> 28 3 70.5 4.53 3.62 7.50
#> 29 3 70.5 4.53 5.08 6.20
#> 30 3 70.5 4.53 7.07 5.30
#> 31 3 70.5 4.53 9.00 4.90
#> 32 3 70.5 4.53 12.15 3.70
#> 33 3 70.5 4.53 24.17 1.05
#> 34 4 72.7 4.40 0.00 0.00
#> 35 4 72.7 4.40 0.35 1.89
#> 36 4 72.7 4.40 0.60 4.60
#> 37 4 72.7 4.40 1.07 8.60
#> 38 4 72.7 4.40 2.13 8.38
#> 39 4 72.7 4.40 3.50 7.54
#> 40 4 72.7 4.40 5.02 6.88
#> 41 4 72.7 4.40 7.02 5.78
#> 42 4 72.7 4.40 9.02 5.33
#> 43 4 72.7 4.40 11.98 4.19
#> 44 4 72.7 4.40 24.65 1.15
#> 45 5 54.6 5.86 0.00 0.00
#> 46 5 54.6 5.86 0.30 2.02
#> 47 5 54.6 5.86 0.52 5.63
#> 48 5 54.6 5.86 1.00 11.40
#> 49 5 54.6 5.86 2.02 9.33
#> 50 5 54.6 5.86 3.50 8.74
#> 51 5 54.6 5.86 5.02 7.56
#> 52 5 54.6 5.86 7.02 7.09
#> 53 5 54.6 5.86 9.10 5.90
#> 54 5 54.6 5.86 12.00 4.37
#> 55 5 54.6 5.86 24.35 1.57
#> 56 6 80.0 4.00 0.00 0.00
#> 57 6 80.0 4.00 0.27 1.29
#> 58 6 80.0 4.00 0.58 3.08
#> 59 6 80.0 4.00 1.15 6.44
#> 60 6 80.0 4.00 2.03 6.32
#> 61 6 80.0 4.00 3.57 5.53
#> 62 6 80.0 4.00 5.00 4.94
#> 63 6 80.0 4.00 7.00 4.02
#> 64 6 80.0 4.00 9.22 3.46
#> 65 6 80.0 4.00 12.10 2.78
#> 66 6 80.0 4.00 23.85 0.92
#> 67 7 64.6 4.95 0.00 0.15
#> 68 7 64.6 4.95 0.25 0.85
#> 69 7 64.6 4.95 0.50 2.35
```