Package ‘exDE’

November 18, 2022

Type Package

Title Extensible Differential Equations for Mosquito-Borne Pathogen Modeling

Version 1.0.0

Description Provides tools to set up modular ordinary and delay differential equation models for mosquito-borne pathogens, focusing on malaria. Modular design is achieved by S3 dispatch on parameter lists for each component which is used to compute the full set of differential equations which may be solved using any of the packages for numerical simulation of differential equations in R. The methods implemented by this package are described in Wu et al. (2022) <doi:10.1101/2022.11.07.22282044>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.1.9000

URL `https://dd-harp.github.io/exDE/`

BugReports `https://github.com/dd-harp/exDE/issues`

Imports deSolve, expm, MASS

Suggests ggplot2, data.table, knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author Sean L. Wu [aut, cre] (<`https://orcid.org/0000-0002-5781-9493`>), David L. Smith [aut] (<`https://orcid.org/0000-0003-4367-3849`>)

Maintainer Sean L. Wu <`slwood89@gmail.com`>

Repository CRAN

Date/Publication 2022-11-18 10:00:04 UTC

R topics documented:

`approx_equal` *Check if two numeric values are approximately equal*

Description

Check if two numeric values are approximately equal

Usage

```
approx_equal(a, b, tol = sqrt(.Machine$double.eps))
```

Arguments

a	a numeric object
b	a numeric object
tol	the numeric tolerance

Value

a logical value

diag_inverse — *Invert a diagonal matrix*

Description

Invert a diagonal matrix which is passed as a vector. If any elements are zero, set them to one.

Usage

```
diag_inverse(x)
```

Arguments

x	a numeric vector

Value

a diagonal matrix

dLdt — *Derivatives for aquatic stage mosquitoes*

Description

This method dispatches on the type of `pars$Lpar`.

Usage

```
dLdt(t, y, pars, eta)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
eta	vector giving number of eggs being laid in each larval habitat

Value

a numeric vector of length `pars$L_ix`

`dLdt.basic` *Derivatives for aquatic stage mosquitoes*

Description

Implements dLdt for the basic competition model.

Usage

```
## S3 method for class 'basic'
dLdt(t, y, pars, eta)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`eta`	vector giving number of eggs being laid in each larval habitat

Value

a numeric vector

`dLdt.trace` *Derivatives for aquatic stage mosquitoes*

Description

Implements dLdt for the trace (forced emergence) model.

Usage

```
## S3 method for class 'trace'
dLdt(t, y, pars, eta)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`eta`	vector giving number of eggs being laid in each larval habitat

Value

a numeric vector

`dMYZdt` *Derivatives for adult mosquitoes*

Description

This method dispatches on the type of `pars$MYZpar`.

Usage

```
dMYZdt(t, y, pars, Lambda, kappa, MosyBehavior)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`Lambda`	emergence rate of adult mosquitoes
`kappa`	net infectiousness of human population
`MosyBehavior`	values returned by MosquitoBehavior, potentially modified by control VectorControl

Value

a numeric vector

`dMYZdt.RM_dde` *Derivatives for adult mosquitoes*

Description

Implements dMYZdt for the generalized RM DDE model.

Usage

```
## S3 method for class 'RM_dde'
dMYZdt(t, y, pars, Lambda, kappa, MosyBehavior)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
Lambda	emergence rate of adult mosquitoes
kappa	net infectiousness of human population
MosyBehavior	values returned by MosquitoBehavior, potentially modified by control Vector-Control

Value

a numeric vector

`dMYZdt.RM_ode` *Derivatives for adult mosquitoes*

Description

Implements dMYZdt for the generalized RM ODE model.

Usage

```
## S3 method for class 'RM_ode'
dMYZdt(t, y, pars, Lambda, kappa, MosyBehavior)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
Lambda	emergence rate of adult mosquitoes
kappa	net infectiousness of human population
MosyBehavior	values returned by MosquitoBehavior, potentially modified by control Vector-Control

Value

a numeric vector

dXdt *Derivatives for human population*

Description

This method dispatches on the type of `pars$Xpar`.

Usage

```
dXdt(t, y, pars, EIR)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
EIR	vector giving the per-capita entomological inoculation rate for each strata

Value

a numeric vector

dXdt.hMoI *Derivatives for human population*

Description

Implements dXdt for the hybrid MoI model.

Usage

```
## S3 method for class 'hMoI'
dXdt(t, y, pars, EIR)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
EIR	vector giving the per-capita entomological inoculation rate for each strata

Value

a numeric vector

dXdt.SIP — *Derivatives for human population*

Description

Implements dXdt for the SIP model.

Usage

```
## S3 method for class 'SIP'
dXdt(t, y, pars, EIR)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`EIR`	vector giving the per-capita entomological inoculation rate for each strata

Value

a numeric vector

dXdt.SIS — *Derivatives for human population*

Description

Implements dXdt for the SIS model.

Usage

```
## S3 method for class 'SIS'
dXdt(t, y, pars, EIR)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`EIR`	vector giving the per-capita entomological inoculation rate for each strata

Value

a numeric vector

ExogenousForcing — *Modify parameters due to exogenous forcing*

Description

This method dispatches on the type of `pars$EXpar`.

Usage

```
ExogenousForcing(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

none

ExogenousForcing.null — *Modify parameters due to exogenous forcing*

Description

Implements ExogenousForcing for the null model of exogenous forcing (do nothing)

Usage

```
## S3 method for class 'null'
ExogenousForcing(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

none

F_alpha — *Number of newly emerging adults from each larval habitat*

Description

This method dispatches on the type of `pars$Lpar`.

Usage

```
F_alpha(t, y, pars)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment

Value

a numeric vector of length `nHabitats`

F_alpha.basic — *Number of newly emerging adults from each larval habitat*

Description

Implements F_alpha for the basic competition model.

Usage

```
## S3 method for class 'basic'
F_alpha(t, y, pars)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment

Value

a numeric vector of length `nHabitats`

F_alpha.trace *Number of newly emerging adults from each larval habitat*

Description

Implements F_alpha for the trace (forced emergence) model.

Usage

```
## S3 method for class 'trace'
F_alpha(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length nHabitats

F_beta *Biting distribution matrix*

Description

This method dispatches on the type of pars$Xpar.

Usage

```
F_beta(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length nStrata

F_beta.hMoI *Biting distribution matrix*

Description

Implements F_beta for the hybrid MoI model.

Usage

```
## S3 method for class 'hMoI'
F_beta(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a matrix of dimensions `nStrata` by `nPatches`

F_beta.SIP *Biting distribution matrix*

Description

Implements F_beta for the SIP model.

Usage

```
## S3 method for class 'SIP'
F_beta(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a matrix of dimensions `nStrata` by `nPatches`

`F_beta.SIS` *Biting distribution matrix*

Description

Implements F_beta for the SIS model.

Usage

```
## S3 method for class 'SIS'
F_beta(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a matrix of dimensions `nStrata` by `nPatches`

`F_beta_lag` *Lagged biting distribution matrix*

Description

This method dispatches on the type of `pars$Xpar`.

Usage

```
F_beta_lag(t, y, pars, lag)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
lag	duration of lag `t-lag`

Value

a numeric vector of length `nStrata`

F_beta_lag.hMoI *Lagged biting distribution matrix*

Description

Implements F_beta_lag for the hybrid MoI model.

Usage

```
## S3 method for class 'hMoI'
F_beta_lag(t, y, pars, lag)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
lag	duration of lag `t-lag`

Value

a matrix of dimensions `nStrata` by `nPatches`

F_beta_lag.SIP *Lagged biting distribution matrix*

Description

Implements F_beta_lag for the SIP model.

Usage

```
## S3 method for class 'SIP'
F_beta_lag(t, y, pars, lag)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
lag	duration of lag `t-lag`

Value

a matrix of dimensions `nStrata` by `nPatches`

F_beta_lag.SIS — *Lagged biting distribution matrix*

Description

Implements F_beta_lag for the SIS model.

Usage

```
## S3 method for class 'SIS'
F_beta_lag(t, y, pars, lag)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`lag`	duration of lag `t-lag`

Value

a matrix of dimensions `nStrata` by `nPatches`

F_eggs — *Number of eggs laid by adult mosquitoes*

Description

This method dispatches on the type of `pars$MYZpar`.

Usage

```
F_eggs(t, y, pars)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment

Value

a numeric vector of length `nPatches`

F_eggs.RM — *Number of eggs laid by adult mosquitoes*

Description

Implements F_eggs for the generalized RM model.

Usage

```
## S3 method for class 'RM'
F_eggs(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nPatches`

F_EIR — *Entomological inoculation rate on human strata*

Description

This method dispatches on the type of `pars$Xpar`.

Usage

```
F_EIR(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nStrata`

F_EIR.hMoI *Entomological inoculation rate on human strata*

Description

Implements F_EIR for the hybrid MoI model.

Usage

```
## S3 method for class 'hMoI'
F_EIR(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nStrata`

F_EIR.SIP *Entomological inoculation rate on human strata*

Description

Implements F_EIR for the SIP model.

Usage

```
## S3 method for class 'SIP'
F_EIR(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nStrata`

F_EIR.SIS *Entomological inoculation rate on human strata*

Description

Implements F_EIR for the SIS model.

Usage

```
## S3 method for class 'SIS'
F_EIR(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nStrata`

F_kappa *Net infectiousness of human population to mosquitoes*

Description

This method dispatches on the type of `pars$MYZpar`.

Usage

```
F_kappa(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nPatches`

F_kappa.RM_dde *Net infectiousness of human population to mosquitoes*

Description

Implements F_kappa for the generalized RM DDE model.

Usage

```
## S3 method for class 'RM_dde'
F_kappa(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nPatches`

F_kappa.RM_ode *Net infectiousness of human population to mosquitoes*

Description

Implements F_kappa for the generalized RM ODE model.

Usage

```
## S3 method for class 'RM_ode'
F_kappa(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nPatches`

F_tau — *Time spent host seeking/feeding and resting/ovipositing*

Description

This method dispatches on the type of `pars$MYZpar`.

Usage

```
F_tau(t, y, pars)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment

Value

either a numeric vector if the model supports this feature, or NULL

F_tau.RM — *Time spent host seeking/feeding and resting/ovipositing*

Description

Implements F_tau for the generalized RM model.

Usage

```
## S3 method for class 'RM'
F_tau(t, y, pars)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment

Value

NULL

F_x *Size of effective infectious human population*

Description

This method dispatches on the type of pars$Xpar.

Usage

```
F_x(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length nStrata

F_x.hMoI *Size of effective infectious human population*

Description

Implements F_x for the hybrid MoI model.

Usage

```
## S3 method for class 'hMoI'
F_x(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length nStrata

F_x.SIP *Size of effective infectious human population*

Description

Implements F_x for the SIP model.

Usage

```
## S3 method for class 'SIP'
F_x(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nStrata`

F_x.SIS *Size of effective infectious human population*

Description

Implements F_x for the SIS model.

Usage

```
## S3 method for class 'SIS'
F_x(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nStrata`

F_x_lag *Size of lagged effective infectious human population*

Description

This method dispatches on the type of `pars$Xpar`.

Usage

```
F_x_lag(t, y, pars, lag)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`lag`	duration of lag `t-lag`

Value

a numeric vector of length `nStrata`

F_x_lag.hMoI *Size of lagged effective infectious human population*

Description

Implements F_x_lag for the hybrid MoI model.

Usage

```
## S3 method for class 'hMoI'
F_x_lag(t, y, pars, lag)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`lag`	duration of lag `t-lag`

Value

a numeric vector of length `nStrata`

F_x_lag.SIP — *Size of lagged effective infectious human population*

Description

Implements F_x_lag for the SIP model.

Usage

```
## S3 method for class 'SIP'
F_x_lag(t, y, pars, lag)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
lag	duration of lag `t-lag`

Value

a numeric vector of length `nStrata`

F_x_lag.SIS — *Size of lagged effective infectious human population*

Description

Implements F_x_lag for the SIS model.

Usage

```
## S3 method for class 'SIS'
F_x_lag(t, y, pars, lag)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
lag	duration of lag `t-lag`

Value

a numeric vector of length `nStrata`

F_Z *Density of infectious mosquitoes*

Description

This method dispatches on the type of `pars$MYZpar`.

Usage

```
F_Z(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nPatches`

F_Z.RM *Density of infectious mosquitoes*

Description

Implements F_Z for the generalized RM model.

Usage

```
## S3 method for class 'RM'
F_Z(t, y, pars)
```

Arguments

t	current simulation time
y	state vector
pars	an environment

Value

a numeric vector of length `nPatches`

F_Z_lag *Density of lagged infectious mosquitoes*

Description

This method dispatches on the type of `pars$MYZpar`.

Usage

```
F_Z_lag(t, y, pars, lag)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`lag`	duration of lag `t-lag`

Value

a numeric vector of length `nPatches`

F_Z_lag.RM *Density of lagged infectious mosquitoes*

Description

Implements F_Z_lag for the generalized RM model.

Usage

```
## S3 method for class 'RM'
F_Z_lag(t, y, pars, lag)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`lag`	duration of lag `t-lag`

Value

a numeric vector of length `nPatches`

---

make_index_L *Add indices for aquatic stage mosquitoes to parameter list*

Description

This method dispatches on the type of `pars$Lpar`. Adds field `L_ix` to parameter list.

Usage

```
make_index_L(pars)
```

Arguments

`pars`	an environment

Value

the modified parameter list

---

make_index_L.basic *Add indices for aquatic stage mosquitoes to parameter list*

Description

Implements make_index_L for basic competition model.

Usage

```
## S3 method for class 'basic'
make_index_L(pars)
```

Arguments

`pars`	an environment

Value

the modified parameter list

make_index_L.trace *Add indices for aquatic stage mosquitoes to parameter list*

Description

Implements make_index_L for trace (forced emergence) model.

Usage

```
## S3 method for class 'trace'
make_index_L(pars)
```

Arguments

pars	an environment

Value

the modified parameter list

make_index_MYZ *Add indices for adult mosquitoes to parameter list*

Description

This method dispatches on the type of `pars$MYZpar`.

Usage

```
make_index_MYZ(pars)
```

Arguments

pars	an environment

Value

the modified parameter list

make_index_MYZ.RM *Add indices for adult mosquitoes to parameter list*

Description

Implements make_index_MYZ for the generalized RM model.

Usage

```
## S3 method for class 'RM'
make_index_MYZ(pars)
```

Arguments

pars	an environment

Value

the modified parameter list

make_index_X *Add indices for human population to parameter list*

Description

This method dispatches on the type of `pars$Xpar`.

Usage

```
make_index_X(pars)
```

Arguments

pars	an environment

Value

the modified parameter list

make_index_X.hMoI *Add indices for human population to parameter list*

Description

Implements make_index_X for the hybrid MoI model.

Usage

```
## S3 method for class 'hMoI'
make_index_X(pars)
```

Arguments

pars	an environment

Value

the modified parameter list

make_index_X.SIP *Add indices for human population to parameter list*

Description

Implements make_index_X for the SIP model.

Usage

```
## S3 method for class 'SIP'
make_index_X(pars)
```

Arguments

pars	an environment

Value

the modified parameter list

make_index_X.SIS *Add indices for human population to parameter list*

Description

Implements make_index_X for the SIS model.

Usage

```
## S3 method for class 'SIS'
make_index_X(pars)
```

Arguments

pars	an environment

Value

the modified parameter list

make_indices *Set indices for generalized spatial model*

Description

Set indices for generalized spatial model

Usage

```
make_indices(pars)
```

Arguments

pars	an environment

Value

none

make_Omega — *Make the mosquito demography matrix*

Description

Make the mosquito demography matrix

Usage

```
make_Omega(g, sigma, K, nPatches)
```

Arguments

g	mortality rate
sigma	emigration rate
K	mosquito dispersal matrix
nPatches	number of patches

Value

a matrix of dimensions `nPatches` by `nPatches`

make_parameters_exogenous_null — *Make parameters for the null model of exogenous forcing (do nothing)*

Description

Make parameters for the null model of exogenous forcing (do nothing)

Usage

```
make_parameters_exogenous_null(pars)
```

Arguments

pars	an environment

Value

none

make_parameters_L_basic

Make parameters for basic competition aquatic mosquito model

Description

Make parameters for basic competition aquatic mosquito model

Usage

```
make_parameters_L_basic(pars, psi, phi, theta, L0)
```

Arguments

pars	an environment
psi	maturation rates for each aquatic habitat
phi	density-independent mortality rates for each aquatic habitat
theta	density-dependent mortality terms for each aquatic habitat
L0	initial conditions

Value

a list with class `basic`.

make_parameters_L_trace

Make parameters for trace aquatic mosquito model

Description

Make parameters for trace aquatic mosquito model

Usage

```
make_parameters_L_trace(pars, Lambda)
```

Arguments

pars	an environment
Lambda	vector of emergence rates from each aquatic habitat

Value

a list with class `trace`.

make_parameters_MYZ_RM_dde
Make parameters for generalized RM DDE adult mosquito model

Description

Make parameters for generalized RM DDE adult mosquito model

Usage

```
make_parameters_MYZ_RM_dde(
  pars,
  g,
  sigma,
  calK,
  f,
  q,
  nu,
  eggsPerBatch,
  tau,
  M0,
  G0,
  Y0,
  Z0
)
```

Arguments

pars	an environment
g	mosquito mortality rate
sigma	emigration rate
calK	mosquito dispersal matrix of dimensions nPatches by nPatches
f	feeding rate
q	human blood fraction
nu	oviposition rate of gravid mosquitoes
eggsPerBatch	eggs laid per oviposition
tau	length of extrinsic incubation period
M0	total mosquito density at each patch
G0	gravid mosquito density at each patch
Y0	infected mosquito density at each patch
Z0	infectious mosquito density at each patch

Value

none

make_parameters_MYZ_RM_ode

Make parameters for generalized RM ODE adult mosquito model

Description

Make parameters for generalized RM ODE adult mosquito model

Usage

```
make_parameters_MYZ_RM_ode(
  pars,
  g,
  sigma,
  calK,
  f,
  q,
  nu,
  eggsPerBatch,
  tau,
  M0,
  G0,
  Y0,
  Z0
)
```

Arguments

pars	an environment
g	mosquito mortality rate
sigma	emigration rate
calK	mosquito dispersal matrix of dimensions nPatches by nPatches
f	feeding rate
q	human blood fraction
nu	oviposition rate of gravid mosquitoes
eggsPerBatch	eggs laid per oviposition
tau	length of extrinsic incubation period
M0	total mosquito density at each patch
G0	gravid mosquito density at each patch
Y0	infected mosquito density at each patch
Z0	infectious mosquito density at each patch

Value

none

make_parameters_vc_lemenach
Make parameters for Le Menach ITN model of vector control

Description

This model of ITN based vector control was originally described in https://malariajournal.biomedcentral.com/articles/10.1186/1475-2875-6-10.

Usage

```
make_parameters_vc_lemenach(
  pars,
  tau0_frac = c(0.68/3, 2.32/3),
  r = 0.56,
  s = 0.03,
  phi = function(t) {
     0
 }
)
```

Arguments

pars	an environment
tau0_frac	a numeric vector giving the proportion of time spent in host seeking/bloodfeeding and resting/oviposition
r	probability of mosquito being repelled upon contact with ITN
s	probability of mosquito successfully feeding upon contact with ITN
phi	a function that takes a single argument t and returns the level of ITN coverage at that time

Value

none

make_parameters_vc_null
Make parameters for the null model of vector control (do nothing)

Description

Make parameters for the null model of vector control (do nothing)

Usage

```
make_parameters_vc_null(pars)
```

Arguments

pars	an environment

Value

none

`make_parameters_X_hMoI` *Make parameters for hybrid MoI human model*

Description

MoI stands for Multiplicity of Infection, and refers to malarial superinfection.

Usage

```
make_parameters_X_hMoI(pars, b, c1, c2, r1, r2, Psi, wf = 1, m10, m20, H)
```

Arguments

pars	an environment
b	transmission probability (efficiency) from mosquito to human
c1	transmission probability (efficiency) from inapparent human infections to mosquito
c2	transmission probability (efficiency) from patent human infections to mosquito
r1	recovery rate from inapparent infections
r2	recovery rate from patent infections
Psi	a matrix of dimensions `nPatches` by `nStrata`
wf	vector of biting weights of length `nStrata`
m10	mean MoI among inapparent human infections
m20	mean MoI among patent human infections
H	size of human population in each strata

Value

a list with class `hMoI`.

make_parameters_X_SIP *Make parameters for SIP human model*

Description

Make parameters for SIP human model

Usage

```
make_parameters_X_SIP(pars, b, c, r, rho, eta, Psi, wf = 1, X0, P0, H)
```

Arguments

pars	an environment
b	transmission probability (efficiency) from mosquito to human
c	transmission probability (efficiency) from human to mosquito
r	recovery rate
rho	probability of successful treatment upon infection
eta	prophylaxis waning rate
Psi	a matrix of dimensions `nPatches` by `nStrata`
wf	vector of biting weights of length `nStrata`
X0	size of infected population in each strata
P0	size of population protected by prophylaxis in each strata
H	size of human population in each strata

Value

a list with class `SIP`.

make_parameters_X_SIS *Make parameters for SIS human model*

Description

Make parameters for SIS human model

Usage

```
make_parameters_X_SIS(pars, b, c, r, Psi, wf = 1, X0, H)
```

Arguments

pars	an environment
b	transmission probability (efficiency) from mosquito to human
c	transmission probability (efficiency) from human to mosquito
r	recovery rate
Psi	a matrix of dimensions `nPatches` by `nStrata`
wf	vector of biting weights of length `nStrata`
X0	size of infected population in each strata
H	size of human population in each strata

Value

a list with class `SIS`.

`metric_calD` *Parasite dispersal by humans*

Description

Compute the `p` by `p` matrix $\mathcal{D}$ whose columns describe how potentially infectious person time from persons in that patch are dispersed across other patches.

$$\mathcal{D} = \text{diag}\left(W\right) \cdot \beta^T \cdot \text{diag}\left(bDH\right) \cdot \beta$$

Usage

```
metric_calD(W, beta, b, D, H)
```

Arguments

W	ambient human population at each patch
beta	the biting distribution matrix
b	transmission efficiency from mosquitoes to humans
D	human transmitting capacity
H	human population size of each strata

Value

a numeric matrix

metric_calR — *Parasite Dispersal through one Parasite Generation (Humans)*

Description

Computes a n by n matrix describing parasite dispersal from infecteds (columns) to infectees (rows).

$$\mathcal{R} = b\beta \cdot \mathcal{V} \cdot \text{diag}\left(W\right) \cdot \beta^T \cdot \text{diag}\left(DH\right)$$

Usage

```
metric_calR(b, beta, calV, W, D, H)
```

Arguments

b	transmission efficiency from mosquitoes to humans
beta	the biting distribution matrix
calV	parasite dispersal by mosquitoes matrix (see metric_calV)
W	ambient human population at each patch
D	human transmitting capacity
H	human population size of each strata

Value

a numeric matrix

metric_calV — *Parasite dispersal by mosquitoes*

Description

Compute the p by p matrix $\mathcal{V}$ whose columns describe how infective bites arising from all the mosquitoes biting a single human on a single day are dispersed to other patches, accounting for movement and mortality.

$$\mathcal{V} = fq\Omega^{-1} \cdot e^{-\Omega\tau} \cdot \text{diag}\left(\frac{fqM}{W}\right)$$

Usage

```
metric_calV(f, q, Omega, tau, M, W)
```

Arguments

f	the feeding rate
q	fraction of bloodmeals taken on humans
Omega	the mosquito demography matrix
tau	duration of the extrinsic incubation period
M	size of mosquito population in each patch
W	ambient human population at each patch

Value

a numeric matrix

---

metric_calZ *Parasite Dispersal through one Parasite Generation (Mosquitoes)*

Description

Computes a p by p matrix describing parasite dispersal from infecteds (columns) to infectees (rows).

$$\mathcal{Z} = e^{-\Omega\tau} \cdot \text{diag}\left(\frac{fqM}{W}\right) \cdot \mathcal{D} \cdot fq\Omega^{-1}$$

Usage

```
metric_calZ(Omega, tau, f, q, M, W, calD)
```

Arguments

Omega	the mosquito demography matrix
tau	duration of the extrinsic incubation period
f	the feeding rate
q	fraction of bloodmeals taken on humans
M	size of mosquito population in each patch
W	ambient human population at each patch
calD	parasite dispersal by humans matrix (see metric_calD)

Value

a numeric matrix

`MosquitoBehavior` *Compute bloodfeeding and mortality rates*

Description

This method dispatches on the type of `pars$MYZpar`. It should, at a minimum return the values `f`, `q`, `g` (blood feeding rate, human feeding proportion, and mortality rate) at the current time, although it may return vectors of these values at multiple times for models with delay. These baseline values will be modified by the vector control component. The return type is a named list with those 3 values, and `f` should have an attr labeled `time` giving the time(s) in the simulation that these bionomic values correspond to.

Usage

```
MosquitoBehavior(t, y, pars)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment

Value

a list

`MosquitoBehavior.RM` *Compute bloodfeeding and mortality rates*

Description

Implements MosquitoBehavior for the generalized RM model.

Usage

```
## S3 method for class 'RM'
MosquitoBehavior(t, y, pars)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment

Value

a named list

VectorControl — *Modify baseline values due to vector control*

Description

This method dispatches on the type of `pars$VCpar`. It takes the baseline `MosyBehavior` values and modifies them, potentially at multiple time points for models with delay.

Usage

```
VectorControl(t, y, pars, MosyBehavior)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`MosyBehavior`	values returned by MosquitoBehavior

Value

a list

VectorControl.lemenach — *Modify baseline values due to vector control*

Description

Implements VectorControl for the Le Menach ITN model of vector control

Usage

```
## S3 method for class 'lemenach'
VectorControl(t, y, pars, MosyBehavior)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`MosyBehavior`	values returned by MosquitoBehavior

Value

a named list

VectorControl.null — *Modify baseline values due to vector control*

Description

Implements VectorControl for the null model of vector control (do nothing)

Usage

```
## S3 method for class 'null'
VectorControl(t, y, pars, MosyBehavior)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
MosyBehavior	values returned by MosquitoBehavior

Value

a named list

xDE_diffeqn — *Generalized spatial differential equation model*

Description

Compute derivatives for deSolve::ode or deSolve::dede using generic methods for each model component. The arguments `EIR_delta` and `kappa_delta` are for adding external forcing to the system from unmodeled sources. This can arise if humans can acquire infection by traveling outside the spatial domain, and arises for mosquitoes if traveling outside the spatial domain or are being infected by unmodeled (non-human) sources. By default these are set to `NULL` and are turned off.

Usage

```
xDE_diffeqn(t, y, pars, EIR_delta = NULL, kappa_delta = NULL)
```

Arguments

t	current simulation time
y	state vector
pars	an environment
EIR_delta	a vector of values to be added to the internal `EIR`
kappa_delta	a vector of values to be added to the internal `kappa`

Value

a list containing the vector of all state derivatives

`xDE_diffeqn_mosy` *Generalized spatial differential equation model (mosquito only)*

Description

Mirrors xDE_diffeqn but only includes the adult and aquatic mosquito components.

Usage

```
xDE_diffeqn_mosy(t, y, pars, kappa, MosyBehavior)
```

Arguments

`t`	current simulation time
`y`	state vector
`pars`	an environment
`kappa`	a vector
`MosyBehavior`	a list emulating the output of MosquitoBehavior for the appropriate adult mosquito model

Value

a list containing the vector of all state derivatives

Index