Package: squire (via r-universe)

October 12, 2026

Type Package

Title SEIR transmission model of COVID-19

Version 0.7.1

Description An extended model of the SEIR model used in the Imperial College London Report into the global impact of COVID-19 and strategies for mitigation and suppression (https://www.imperial.ac.uk/mrc-global-infectious-disease-analysis/covid-19/report-12-global-impact-covid-19/).
Extensions now include healthcare treatment pathways and excess mortality.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Suggests testthat (>= 2.1.0), covr, readxl, knitr, odin.js, rmarkdown, mockery

RoxygenNote 7.1.2

Imports odin (>= 1.2.4), dde (>= 1.0.2), dplyr, tidyr, rlang, furrr, magrittr, purrr, crayon, methods, ggplot2, viridis, coda, mvtnorm, drjacoby

Depends R (>= 3.1.0)

VignetteBuilder knitr

Remotes mrc-ide/dde, mrc-ide/odin, mrc-ide/odin.js, mrc-ide/drjacoby

Config/pak/sysreqs cmake git make libgit2-dev libicu-dev libuv1-dev libssl-dev libx11-dev

Repository https://ncov-ic.r-universe.dev

Date/Publication 2022-09-08 09:14:12 UTC

RemoteUrl https://github.com/mrc-ide/squire

RemoteRef master

RemoteSha 9fd16c495b2e571a309f35745a602a66ebc9dfb7

Contents

adjusted_eigen . . . 3
align . . . 4
beta_est . . . 5
beta_est_explicit . . . 5
beta_est_simple . . . 6
calibrate . . . 7
check_squire . . . 10
check_time_change . . . 10
compare_output . . . 11
conduct_replicate . . . 11
contact_matrices . . . 13
country_specific_healthcare_capacity . . . 13
create_master_chain . . . 14
default_durations . . . 14
default_probs . . . 15
deterministic_model . . . 15
div_pop . . . 16
drjacoby_mcmc . . . 16
elderly_pop . . . 20
explicit_model . . . 20
extract_deaths . . . 21
extract_hospital_occ . . . 21
extract_ICU_occ . . . 22
extract_infection_incidence . . . 22
format_deterministic_output . . . 23
format_output . . . 23
format_output_simple_model . . . 24
get_elderly_population . . . 25
get_healthcare_capacity . . . 25
get_lmic_countries . . . 26
get_mixing_matrix . . . 26
get_population . . . 27
income_group . . . 27
income_strata_healthcare_capacity . . . 28
init_check . . . 28
init_check_explicit . . . 29
intervention_dates_for_odin . . . 29
jc_prop_update . . . 30
matrix_check . . . 31
matrix_set . . . 31
parameters_explicit_SEEIR . . . 32
parameters_simple_SEEIR . . . 35
parse_country_severity . . . 36
particle_filter . . . 37
particle_filter_data . . . 38
plot.squire_scan . . . 39

plot.squire_simulation . . . 39
pmcmc . . . 40
population . . . 45
pos_num . . . 45
print.squire_simulation . . . 46
process_contact_matrix . . . 46
process_contact_matrix_scaled_age . . . 47
projection_plotting . . . 47
projections . . . 48
run_deterministic_comparison . . . 50
run_deterministic_SEIR_model . . . 51
run_explicit_SEEIR_model . . . 54
run_particle_filter . . . 58
run_simple_SEEIR_model . . . 59
sample_3d_grid_scan . . . 60
sample_drjacoby . . . 61
sample_grid_scan . . . 62
sample_pmcmc . . . 63
scan_R0_date . . . 64
scan_R0_date_Meff . . . 65
simple_model . . . 67
summary.squire_simulation . . . 68
trigger_projections . . . 68

Index **70**

`adjusted_eigen` *Compute age-adjusted eigenvalue for mixing matrix*

Description

Compute age-adjusted eigenvalue for mixing matrix

Usage

```
adjusted_eigen(dur_IMild, dur_ICase, prob_hosp, mixing_matrix)
```

Arguments

`dur_IMild`	Duration of mild infectiousness (days)
`dur_ICase`	Delay between symptom onset and requiring hospitalisation (days)
`prob_hosp`	Probability of hospitilisation by ages
`mixing_matrix`	Mixing matrix

Value

Eigenvalue

`align` *Align deaths*

Description

Quick method to align a set of simulations to a cumulative deaths total

Usage

```
align(
  deaths,
  reporting_fraction = 1,
  country = NULL,
  population = NULL,
  contact_matrix_set = NULL,
  seeding_age_groups = c("35-40", "40-45", "45-50", "50-55"),
  min_seeding_cases = 5,
  max_seeding_cases = 50,
  R0 = 3,
  R0_scan = NULL,
  replicates = 100,
  ...
)
```

Arguments

`deaths`	Number of observed deaths
`reporting_fraction`	Reporting fraction. Numeric for what proportion of the total deaths the reported deaths represent. E.g. 0.5 results in the model calibrating to twice the deaths provided by `deaths`
`country`	Character for country beign simulated. WIll be used to generate `population` and `contact_matrix_set` if unprovided. Either `country` or `population` and `contact_matrix_set` must be provided.
`population`	Population vector (for each age group). Default = NULL, which will cause population to be sourced from `country`
`contact_matrix_set`	Contact matrices used in simulation. Default = NULL, which will generate this based on the `country`.
`seeding_age_groups`	Age groups for seeding
`min_seeding_cases`	Minimum seeding cases
`max_seeding_cases`	Maximum seeding cases

R0	R0 to be passed to run_explicit_SEEIR_model. Default = 3
R0_scan	Vector or R0 values to sample from to introduce uncertainty in predictions. Default = NULL, which will not scan. If provided, the first value in R0 will be drawn from R0_scan
replicates	Replicates to be passed to run_explicit_SEEIR_model. Default = 100
...	Further aguments for run_explicit_SEEIR_model()

Value

List of time adjusted squire_simulations

`beta_est` *Estimate beta for a squire model using model parameters*

Description

Estimate beta for a squire model using model parameters

Usage

```
beta_est(squire_model, model_params, R0)
```

Arguments

squire_model	A squire model. Default = explicit_SEEIR()
model_params	Squire model parameters, from a call to generate_parameters().
R0	Basic reproduction number

Value

Beta parameter

`beta_est_explicit` *Estimate beta parameter for explicit model*

Description

Estimate beta parameter for explicit model

Usage

```
beta_est_explicit(dur_IMild, dur_ICase, prob_hosp, mixing_matrix, R0)
```

Arguments

dur_IMild	Duration of mild infectiousness (days)
dur_ICase	Delay between symptom onset and requiring hospitalisation (days)
prob_hosp	Probability of hospitilisation by ages
mixing_matrix	Mixing matrix
R0	Basic reproduction number

Value

Beta parameter

---

beta_est_simple *Estimate beta parameter*

Description

Estimate beta parameter

Usage

```
beta_est_simple(duration_infectiousness, mixing_matrix, R0)
```

Arguments

duration_infectiousness	Duration of infectiousness (days)
mixing_matrix	Mixing matrix
R0	Basic reproduction number

Value

Beta parameter

`calibrate` *Calibrate via particle filter grid search using time series of deaths*

Description

Calibrate via particle filter grid search using time series of deaths

Usage

```
calibrate(
  data,
  R0_min,
  R0_max,
  R0_step,
  R0_prior = NULL,
  first_start_date,
  last_start_date,
  day_step,
  Meff_min = 1,
  Meff_max = 1,
  Meff_step = 0.1,
  squire_model = explicit_model(),
 Rt_func = function(R0_change, R0, Meff) {    exp(log(R0) - Meff * (1 - R0_change)) },
  pars_obs = NULL,
  forecast = 0,
  n_particles = 100,
  reporting_fraction = 1,
  treated_deaths_only = FALSE,
  replicates = 100,
  date_R0_change = NULL,
  R0_change = NULL,
  date_ICU_bed_capacity_change = NULL,
  baseline_ICU_bed_capacity = NULL,
  ICU_bed_capacity = NULL,
  date_hosp_bed_capacity_change = NULL,
  baseline_hosp_bed_capacity = NULL,
  hosp_bed_capacity = NULL,
  date_contact_matrix_set_change = NULL,
  baseline_contact_matrix = NULL,
  contact_matrix_set = NULL,
  country = NULL,
  population = NULL,
  dt = 0.1,
  ...
)
```

Arguments

data	Deaths data to fit to. See example_deaths.csv and particle_filter_data()
R0_min	Minimum value of R0 in the search
R0_max	Maximum value of R0 in the search
R0_step	Step to increment R0 between min and max
R0_prior	Prior for R0. Default = NULL, which is a flat prior. Should be provided as a list with first argument the distribution function and the second the function arguments (excluding quantiles which are worked out based on R0_min and R0_max), e.g. 'list("func" = dnorm, args = list("mean"= 3.5, "sd"= 3))'.
first_start_date	Earliest start date as 'yyyy-mm-dd'
last_start_date	Latest start date as 'yyyy-mm-dd'
day_step	Step to increment date in days
Meff_min	Minimum value of Meff (Movement effect size) in the search. Default = 1, which is the same as Meff_max. If Meff_max and Meff_min are the same then only a 2d grid search is performed based on R0 and start_date
Meff_max	Maximum value of Meff (Movement effect size) in the search. Default = 1, which is the same as Meff_min. If Meff_max and Meff_min are the same then only a 2d grid search is performed based on R0 and start_date
Meff_step	Step to increment Meff (Movement effect size) between min and max. Default = 0.1
squire_model	A squire model. Default = explicit_SEIR()
Rt_func	Function for converting R0, Meff and R0_change. Function must have names arguments of R0, Meff and R0_change. Default is linear relationship on the log scale given by exp(log(R0) - Meff*(1-R0_change)).
pars_obs	list of parameters to use for the comparison function.
forecast	Number of days to forecast forward. Default = 0
n_particles	Number of particles. Positive Integer. Default = 100
reporting_fraction	Reporting fraction. Numeric for what proportion of the total deaths the reported deaths represent. E.g. 0.5 results in the model calibrating to twice the deaths provided by data$deaths
treated_deaths_only	Boolean for whether likelihood is based only on deaths that occur from health-care systems, i.e. are treated. Default = FALSE, which uses all deaths.
replicates	Replicates to be run. Default = 100
date_R0_change	Calendar dates at which R0_change occurs. Defaut = NULL, i.e. no change in R0
R0_change	Numeric vector for relative changes in R0. Default = NULL, i.e. no change in R0

`date_ICU_bed_capacity_change`
: Calendar dates at which ICU bed capacity changes set in `model_params` change. Defaut = NULL, i.e. no change

`baseline_ICU_bed_capacity`
: The starting number of ICU beds before the epidemic started. Default = NULL, which will use the hospital beds data for the country provided. If no country is provided then this is 3/100 of hospital beds

`ICU_bed_capacity`
: Number of ICU beds at each date specified in `date_ICU_bed_capacity_change`. Must be same length as `date_ICU_bed_capacity_change`.

`date_hosp_bed_capacity_change`
: Calendar dates at which hospital bed capacity changes set in `model_params` change. Defaut = NULL, i.e. no change

`baseline_hosp_bed_capacity`
: The starting number of hospital beds before the epidemic started. Default = NULL, which will use the hospital beds data for the country provided. If no country is provided then this is 5/1000 of the population

`hosp_bed_capacity`
: Number of hospital beds at each date specified in `date_hosp_bed_capacity_change`. Must be same length as `date_hosp_bed_capacity_change`.

`date_contact_matrix_set_change`
: Calendar dates at which the contact matrices set in `model_params` change. Defaut = NULL, i.e. no change

`baseline_contact_matrix`
: The starting contact matrix prior to any changes due to interventions or otherwise. Default = NULL, which will use the contact matrix associated with the coutnry provided.

`contact_matrix_set`
: List of contact matrices to be used from the dates provided in `date_contact_matrix_set_change`.Must be same length as `date_contact_matrix_set_change`

`country`
: Character for country beign simulated. WIll be used to generate `population` and `contact_matrix_set` if unprovided. Either `country` or `population` and `contact_matrix_set` must be provided.

`population`
: Population vector (for each age group). Default = NULL, which will cause population to be sourced from `country`

`dt`
: Time Step. Default = 0.1

`...`
: Further aguments for the model parameter function. If using the `explicit_model` (default) this will be `parameters_explicit_SEEIR`.

Value

List of dated squire simulations

check_squire *Check object is an squire_simulation*

Description

Check object is an squire_simulation

Usage

```
check_squire(x)
```

Arguments

x	an onject

check_time_change *Check time change inputs are correct*

Description

Check time change inputs are correct

Usage

```
check_time_change(tt, time_period)
```

Arguments

tt	Time change points
time_period	Length of simulation

Value

Nothing if check pass

compare_output *Compare model to death data*

Description

Compare the model to death data for use with the particle filter

Usage

```
compare_output(model, pars_obs, data, type = "explicit_SEEIR_model")
```

Arguments

model	An odin_model object
pars_obs	Parameters for the observations
data	The data to be compared against
type	The class of the model. At the moment this can only be explicit_SEIR but as more models come online we can use this parameter to control the type of comparison function generated.

conduct_replicate *Handles simulating replicates within* projections

Description

Handles simulating replicates within projections

Usage

```
conduct_replicate(
  x,
  r = NULL,
  t_steps = NULL,
  R0 = NULL,
  R0_change = NULL,
  tt_R0 = NULL,
  contact_matrix_set = NULL,
  contact_matrix_set_change = NULL,
  tt_contact_matrix = NULL,
  hosp_bed_capacity = NULL,
  hosp_bed_capacity_change = NULL,
  tt_hosp_beds = NULL,
  ICU_bed_capacity = NULL,
  ICU_bed_capacity_change = NULL,
  tt_ICU_beds = NULL,
  model_user_args = NULL
)
```

Arguments

x	Simulation replicate number
r	Output of pmcmc or calibrate
t_steps	List of time steps to be run for each replicate
R0	Numeric vector for R0 from t = 0 in the calibration. E.g. R0 = c(2, 1). Default = NULL, which will use R0_change to alter R0 if provided.
R0_change	Numeric vector for relative changes in R0 relative to the final R0 used in the calibration (i.e. at t = 0 in the calibration) E.g. R0 = c(0.8, 0.5). Default = NULL, which will use R0 to parameterise changes in R0 if provided.
tt_R0	Change time points for R0
contact_matrix_set	Contact matrices used in simulation. Default = NULL, which will use contact_matrix_set_change to alter the contact matrix if provided.
contact_matrix_set_change	Numeric vector for relative changes in the contact matrix realtive to the final contact matrix used in the calibration (i.e. at t = 0 in the calibration). E.g. contact_matrix_set_change = c(0.8, 0.5). Default = NULL, which will use contact_matrix_set to parameterise changes in contact matrices if if provided.
tt_contact_matrix	Time change points for matrix change. Default = 0
hosp_bed_capacity	Numeric vector for hospital bed capacity from t = 0 in the calibration. Default = NULL, which will use hosp_bed_capacity_change to alter hosp_bed_capacity if provided.
hosp_bed_capacity_change	Numeric vector for relative changes in hospital bed capacity relative to the final hospital bed capacity used in the calibration (i.e. at t = 0 in the calibration). E.g. hosp_bed_capacity_change = c(0.8, 0.5). Default = NULL, which will use hosp_bed_capacity to parameterise changes in hospital bed capacity if provided.
tt_hosp_beds	Change time points for hosp_bed_capacity
ICU_bed_capacity	Numeric vector for ICU bed capacity from t = 0 in the calibration. Default = NULL, which will use ICU_bed_capacity_change to alter ICU_bed_capacity if provided.
ICU_bed_capacity_change	Numeric vector for relative changes in ICU bed capacity relative to the final ICU bed capacity used in the calibration (i.e. at t = 0 in the calibration). E.g. ICU_bed_capacity_change = c(0.8, 0.5). Default = NULL, which will use ICU_bed_capacity to parameterise changes in ICU bed capacity if provided.
tt_ICU_beds	Change time points for ICU_bed_capacity
model_user_args	List of other parameters to be passed to the model to be run. Default = NULL. An example would be:

```
list( list( "prob_severe" = runif(17), "tt_dur_get_ox_survive" = c(0,
10), "gamma_get_ox_survive" = 0.2), list( "prob_severe" = runif(17),
"tt_dur_get_mv_survive" = c(0, 5), "gamma_get_mv_survive" = 0.1) )
```

The list should be the same length as the number of replicates in the simulations. Each list element should then be a list with elements named to match the arguments expected by the odin model with `r`. Above would be suitable to set the model parameters for a simulation with 2 replicates. You do not have to have the same arguments in each list.

`contact_matrices` *Contact matrices*

Description

A list of contact matrices representing different ...

Usage

```
contact_matrices
```

Format

A list of contact matrices

`country_specific_healthcare_capacity`
Country specific healthcare capacity

Description

Hospital and ICU beds per 1000 pop for various countries

Usage

```
country_specific_healthcare_capacity
```

Format

A data.frame

create_master_chain *create a master chain from a pmcmc_list object*

Description

create a master chain from a pmcmc_list object

Usage

```
create_master_chain(x, burn_in)
```

Arguments

x	a pmcmc_list object
burn_in	an integer denoting the number of samples to discard from each chain

default_durations *Return the default hospital durations for modelling*

Description

Return the default hospital durations for modelling

Usage

```
default_durations()
```

Details

- tt_dur_get_ox_survive = 0
- tt_dur_get_mv_survive = 0
- tt_dur_get_ox_die = 0
- tt_dur_get_mv_die = 0
- dur_get_ox_survive = 9
- dur_get_ox_die = 9
- dur_not_get_ox_survive = 9 * 0.5
- dur_not_get_ox_die = 9 * 0.5
- dur_get_mv_survive = 14.8
- dur_get_mv_die = 11.1
- dur_not_get_mv_survive = 14.8 * 0.5
- dur_not_get_mv_die = 1
- dur_rec = 3

- dur_R = Inf
- dur_E = 4.6
- dur_IMild = 2.1
- dur_ICase = 4.5

Value

List of default durations

`default_probs` *Return the default probabilities for modelling*

Description

Return the default probabilities for modelling

Usage

```
default_probs()
```

Value

list of default probabilities

`deterministic_model` *Simple SEEIR model creation. We will use this structure to ensure that model fitting is flexible in the future as more models are added*

Description

Create a simple model

Usage

```
deterministic_model()
```

div_pop *Divide matrix by population*

Description

Divide matrix by population

Usage

```
div_pop(contact, population)
```

Arguments

contact	Matrix
population	Population vector

Value

Matrix

drjacoby_mcmc *Run an MCMC Sampler using drjacoby*

Description

The drjacoby mcmc sampler is very similar to [[pmcmc]] but there are a few subtle differences that meant it was easier to have a separate function for using drjacoby for the mcmc process

Usage

```
drjacoby_mcmc(
  data,
  n_mcmc,
  log_likelihood = NULL,
  log_prior = NULL,
  n_particles = 100,
  steps_per_day = 4,
  output_proposals = FALSE,
  n_chains = 1,
  squire_model = explicit_model(),
 pars_obs = list(phi_cases = 1, k_cases = 2, phi_death = 1, k_death = 2, exp_noise =
    1e+06),
 pars_init = list(start_date = as.Date("2020-02-07"), R0 = 2.5, Meff = 2, Meff_pl = 3,
    R0_pl_shift = 0),
 pars_min = list(start_date = as.Date("2020-02-01"), R0 = 0, Meff = 1, Meff_pl = 2,
```

```
    R0_pl_shift = -2),
  pars_max = list(start_date = as.Date("2020-02-20"), R0 = 5, Meff = 3, Meff_pl = 4,
    R0_pl_shift = 5),
  pars_discrete = list(start_date = TRUE, R0 = FALSE, Meff = FALSE, Meff_pl = FALSE,
    R0_pl_shift = FALSE),
  reporting_fraction = 1,
  treated_deaths_only = FALSE,
  country = NULL,
  population = NULL,
  contact_matrix_set = NULL,
  baseline_contact_matrix = NULL,
  date_contact_matrix_set_change = NULL,
  R0_change = NULL,
  date_R0_change = NULL,
  hosp_bed_capacity = NULL,
  baseline_hosp_bed_capacity = NULL,
  date_hosp_bed_capacity_change = NULL,
  ICU_bed_capacity = NULL,
  baseline_ICU_bed_capacity = NULL,
  date_ICU_bed_capacity_change = NULL,
  date_vaccine_change = NULL,
  baseline_max_vaccine = NULL,
  max_vaccine = NULL,
  date_vaccine_efficacy_infection_change = NULL,
  baseline_vaccine_efficacy_infection = NULL,
  vaccine_efficacy_infection = NULL,
  date_vaccine_efficacy_disease_change = NULL,
  baseline_vaccine_efficacy_disease = NULL,
  vaccine_efficacy_disease = NULL,
  Rt_args = NULL,
  burnin = 0,
  replicates = 100,
  forecast = 0,
  drjacoby_list = list(),
  ...
)
```

Arguments

data	Data to fit to. This must be constructed with particle_filter_data
n_mcmc	number of mcmc mcmc iterations to perform
log_likelihood	function to calculate log likelihood, must take named parameter vector as input, allow passing of implicit arguments corresponding to the main function arguments. Returns a named list, with entries: - $log_likelihood, a single numeric - $sample_state, a numeric vector corresponding to the state of a single particle, chosen at random, at the final time point for which we have data. If NULL, calculated using the function calc_loglikelihood.
log_prior	function to calculate log prior, must take named parameter vector as input, re-

turns a single numeric. If NULL, uses uninformative priors which do not affect the posterior

`n_particles` Number of particles (considered for both the PMCMC fit and sampling from posterior)

`steps_per_day` Number of steps per day

`output_proposals`
Logical indicating whether proposed parameter jumps should be output along with results

`n_chains` number of MCMC chains to run

`squire_model` A squire model to use

`pars_obs` list of parameters to use in comparison with `compare`. Must be a list containing, e.g. list(phi_cases = 0.1, k_cases = 2, phi_death = 1, k_death = 2, exp_noise = 1e6)

`pars_init` named list of initial inputs for parameters being sampled

`pars_min` named list of lower reflecting boundaries for parameter proposals

`pars_max` named list of upper reflecting boundaries for parameter proposals

`pars_discrete` named list of logicals, indicating if proposed jump should be discrete

`reporting_fraction`
Reporting fraction. Numeric for what proportion of the total deaths the reported deaths represent. E.g. 0.5 results in the model calibrating to twice the deaths provided by `data$deaths`

`treated_deaths_only`
Boolean for whether likelihood is based only on deaths that occur from health-care systems, i.e. are treated. Default = FALSE, which uses all deaths.

`country` Character for country beign simulated. WIll be used to generate `population` and `contact_matrix_set` if unprovided. Either `country` or `population` and `contact_matrix_set` must be provided.

`population` Population vector (for each age group). Default = NULL, which will cause population to be sourced from `country`

`contact_matrix_set`
List of contact matrices to be used from the dates provided in `date_contact_matrix_set_change`.Must be same length as `date_contact_matrix_set_change`

`baseline_contact_matrix`
The starting contact matrix prior to any changes due to interventions or otherwise. Default = NULL, which will use the contact matrix associated with the coutnry provided.

`date_contact_matrix_set_change`
Calendar dates at which the contact matrices set in `model_params` change. Defaut = NULL, i.e. no change

`R0_change` Numeric vector for relative changes in R0. Default = NULL, i.e. no change in R0

`date_R0_change` Calendar dates at which R0_change occurs. Defaut = NULL, i.e. no change in R0

hosp_bed_capacity
: Number of hospital beds at each date specified in date_hosp_bed_capacity_change. Must be same length as date_hosp_bed_capacity_change.

baseline_hosp_bed_capacity
: The starting number of hospital beds before the epidemic started. Default = NULL, which will use the hospital beds data for the country provided. If no country is provided then this is 5/1000 of the population

date_hosp_bed_capacity_change
: Calendar dates at which hospital bed capacity changes set in model_params change. Defaut = NULL, i.e. no change

ICU_bed_capacity
: Number of ICU beds at each date specified in date_ICU_bed_capacity_change. Must be same length as date_ICU_bed_capacity_change.

baseline_ICU_bed_capacity
: The starting number of ICU beds before the epidemic started. Default = NULL, which will use the hospital beds data for the country provided. If no country is provided then this is 3/100 of hospital beds

date_ICU_bed_capacity_change
: Calendar dates at which ICU bed capacity changes set in model_params change. Defaut = NULL, i.e. no change

date_vaccine_change
: Date that vaccine doses per day change. Default = NULL.

baseline_max_vaccine
: Baseline vaccine doses per day. Default = NULL

max_vaccine
: Time varying maximum vaccine doeses per day. Default = NULL.

date_vaccine_efficacy_infection_change
: Date that vaccine efficacy against infection changes. Default = NULL.

baseline_vaccine_efficacy_infection
: Baseline vaccine effacy against infection. Default = NULL

vaccine_efficacy_infection
: Time varying vaccine efficacy against infection. Default = NULL.

date_vaccine_efficacy_disease_change
: Date that vaccine efficacy against disease changes. Default = NULL.

baseline_vaccine_efficacy_disease
: Baseline vaccine efficacy against disease Default = NULL

vaccine_efficacy_disease
: Time varying vaccine efficacy against infection. Default = NULL.

Rt_args
: List of arguments to be passed to evaluate_Rt_pmcmc for calculating Rt. Current arguments are available in Rt_args_list

burnin
: number of iterations to discard from the start of MCMC run when sampling from the posterior for trajectories

replicates
: number of trajectories (replicates) to be returned that are being sampled from the posterior probability results produced by run_mcmc_chain to select parameter set. For each parmater set sampled, run particle filter with n_particles and sample 1 trajectory

forecast	Number of days to forecast forward. Default = 0
drjacoby_list	List of arguments to pass to [[drjacoby::run_mcmc]] that are not data, df_params, misc, loglike, logprior, burnin or samples
...	Further aguments for the model parameter function. If using the explicit_model (default) this will be parameters_explicit_SEEIR.

Details

Run a drjacoby mcmc sampler with the Squire model setup

---

elderly_pop *Elderly Population*

Description

Population of 80-84, 85-89 and 90+ by age group and country

Usage

```
elderly_pop
```

Format

A data.frame

---

explicit_model *Explicit SEEIR model creation. We will use this structure to ensure that model fitting is flexible in the future as more models are added*

Description

Create an explicit model

Usage

```
explicit_model()
```

extract_deaths *Extract deaths from model output*

Description

Extract deaths from model output

Usage

```
extract_deaths(x, reduce_age = TRUE, date_0 = NULL)
```

Arguments

x	squire_simulation object
reduce_age	Collapse age-dimension
date_0	Date of time 0, if specified a date column will be added

Value

Formatted long data.frame

extract_hospital_occ *Extract hospital bed occupancy from model output*

Description

Extract hospital bed occupancy from model output

Usage

```
extract_hospital_occ(x, reduce_age = TRUE, date_0 = NULL)
```

Arguments

x	squire_simulation object
reduce_age	Collapse age-dimension
date_0	Date of time 0, if specified a date column will be added

Value

Formatted long data.frame

extract_ICU_occ *Extract ICU bed occupancy from model output*

Description

Extract ICU bed occupancy from model output

Usage

```
extract_ICU_occ(x, reduce_age = TRUE, date_0 = NULL)
```

Arguments

x	squire_simulation object
reduce_age	Collapse age-dimension
date_0	Date of time 0, if specified a date column will be added

Value

Formatted long data.frame

extract_infection_incidence *Extract infection incidence from model output*

Description

Extract infection incidence from model output

Usage

```
extract_infection_incidence(x, reduce_age = TRUE, date_0 = NULL)
```

Arguments

x	squire_simulation object
reduce_age	Collapse age-dimension
date_0	Date of time 0, if specified a date column will be added

Value

Formatted long data.frame

`format_deterministic_output` *Format deterministic model output as data.frame*

Description

Format deterministic model output as data.frame

Usage

```
format_deterministic_output(x)
```

Arguments

`x`	squire_simulation object

Value

Formatted long data.frame

`format_output` *Format model output as data.frame*

Description

Format model output as data.frame

Usage

```
format_output(
  x,
  var_select = NULL,
  reduce_age = TRUE,
  combine_compartments = TRUE,
  date_0 = NULL
)
```

Arguments

`x`	squire_simulation object
`var_select`	Vector of compartment names, e.g. `c("S", "R")`. In addition a number of summary compartment can be requested. These include: • "deaths" Daily Deaths • "infections" Daily Infections • "hospital_occupancy" Occupied Hospital Beds

- "ICU_occupancy" Occupied ICU Beds
- "hospital_demand Required Hospital Beds
- "ICU_demand Required ICU Beds

`reduce_age` Collapse age-dimension, calculating the total in the compartment.

`combine_compartments`
Collapse compartments of same type together (e.g. E1 and E2 -> E)

`date_0` Date of time 0, if specified a date column will be added

Value

Formatted long data.frame

`format_output_simple_model`
Format model output from simple as data.frame

Description

Format model output from simple as data.frame

Usage

```
format_output_simple_model(
  x,
  var_select = NULL,
  reduce_age = TRUE,
  combine_compartments = TRUE,
  date_0 = NULL
)
```

Arguments

`x` squire_simulation object

`var_select` Vector of compartment names, e.g. `c("S", "R")`

`reduce_age` Collapse age-dimension

`combine_compartments`
Collapse compartments of same type together (e.g. E1 and E2 -> E)

`date_0` Date of time 0, if specified a date column will be added

Value

Formatted long data.frame

get_elderly_population

Get elderly population data (5 year age-breakdown for 80-84, 85-89 and 90+)

Description

Get elderly population data (5 year age-breakdown for 80-84, 85-89 and 90+)

Usage

```
get_elderly_population(country = NULL, iso3c = NULL, simple_SEIR = FALSE)
```

Arguments

country	Country name
iso3c	ISO 3C Country Code
simple_SEIR	Logical. Is the population for the simple_SEIR. Default = FALSE

Value

Population data.frame

get_healthcare_capacity

Get healthcare capacity data

Description

Get healthcare capacity data

Usage

```
get_healthcare_capacity(country, simple_SEIR = FALSE)
```

Arguments

country	Country name
simple_SEIR	Logical. Is the population for the simple_SEIR. Default = FALSE

Value

Healthcare capacity data

get_lmic_countries *Get supported LMIC countries*

Description

Get supported LMIC countries

Usage

```
get_lmic_countries()
```

Value

vector of support LMIC

get_mixing_matrix *Get mixing matrix*

Description

Get mixing matrix

Usage

```
get_mixing_matrix(country = NULL, iso3c = NULL)
```

Arguments

country	Country name
iso3c	ISO 3C Country Code

Value

Age mixing matrix

get_population *Get population data*

Description

Get population data

Usage

```
get_population(country = NULL, iso3c = NULL, simple_SEIR = FALSE)
```

Arguments

country	Country name
iso3c	ISO 3C Country Code
simple_SEIR	Logical. Is the population for the simple_SEIR. Default = FALSE

Value

Population data.frame

income_group *Income group*

Description

World Bank income group for all countries

Usage

```
income_group
```

Format

A data.frame

income_strata_healthcare_capacity *Income strata healthcare capacity*

Description

Average Hospital and ICU beds per 1000 pop for different World Bank income groups

Usage

```
income_strata_healthcare_capacity
```

Format

A data.frame

init_check *Check and set up initial values*

Description

Check and set up initial values

Usage

```
init_check(init, population)
```

Arguments

init	Data.frame of initial conditions
population	Population vector (for each age group)

Value

Checked initial values data.frame

init_check_explicit *Check and set up initial values for explicit model*

Description

Check and set up initial values for explicit model

Usage

```
init_check_explicit(init, population, seeding_cases = 20)
```

Arguments

`init`	Data.frame of initial conditions. Default = NULL
`population`	Population vector (for each age group). Default = NULL, which will cause population to be sourced from `country`
`seeding_cases`	Initial number of cases seeding the epidemic

Value

Checked initial values data.frame

intervention_dates_for_odin *Prepare intervention timing for odin*

Description

Prepare dates of intervention for use with odin. This function exists to make explicit how time changes through the model relative to the data and to odin's internal clock.

Usage

```
intervention_dates_for_odin(
  dates,
  change,
  start_date,
  steps_per_day,
  starting_change = 1
)
```

Arguments

dates	Dates (or ISO-formatted strings for conversion with as.Date at which intervention changes.
change	Variable that is changing at each of dates.
start_date	The date to start the simulation from..
steps_per_day	The number of steps per day
starting_change	The first value to use for change in the case that all provided dates are after start_date

Details

If start date is after elements in dates, these will be trimmed accordinly and the final change value used as the value one day after start date.

---

jc_prop_update *update_sigma*

Description

Involved in the Johnstone-Change optimisation within the Metropolis-Hastings MCMC. Function to iteratively update the scaling factor and covariance matrix involved in the proposal distribution.

Usage

```
jc_prop_update(
  accepted,
  i,
  current_sf,
  previous_mu,
  current_parameters,
  current_covariance_matrix,
  required_acceptance_ratio
)
```

Arguments

accepted	whether or not the most recent parameter proposal was accepted
i	the iteration number
current_sf	the current scaling factor
previous_mu	running average of the MCMC parameters
current_parameters	current parameters
current_covariance_matrix	current covariance matrix
required_acceptance_ratio	required acceptance ratio

matrix_check *Check dimensions of inputs*

Description

Check dimensions of inputs

Usage

```
matrix_check(population, contact_matrix_set)
```

Arguments

population	Population vector (for each age group)
contact_matrix_set	Contact matrices used in simulation

Value

Null if checks pass

matrix_set *Process set of contact matrices -> mixing matrices*

Description

Process set of contact matrices -> mixing matrices

Usage

```
matrix_set(contact_matrix_set, population)
```

Arguments

contact_matrix_set	Set of contact matrices
population	Vector of populaion by age

Value

Processed set of mixing matrices

parameters_explicit_SEEIR *Parmaters for explicit SEEIR model*

Description

Parmaters for explicit SEEIR model

Usage

```
parameters_explicit_SEEIR(
  country = NULL,
  population = NULL,
  tt_contact_matrix = 0,
  contact_matrix_set = NULL,
  R0 = 3,
  tt_R0 = 0,
  beta_set = NULL,
  time_period = 365,
  dt = 0.1,
  init = NULL,
  seeding_cases = NULL,
  prob_hosp = NULL,
  prob_severe = NULL,
  prob_non_severe_death_treatment = NULL,
  prob_non_severe_death_no_treatment = NULL,
  prob_severe_death_treatment = NULL,
  prob_severe_death_no_treatment = NULL,
  p_dist = probs$p_dist,
  walker_params = FALSE,
  dur_E = NULL,
  dur_IMild = NULL,
  dur_ICase = NULL,
  dur_get_ox_survive = NULL,
  tt_dur_get_ox_survive = NULL,
  dur_get_ox_die = NULL,
  tt_dur_get_ox_die = NULL,
  dur_not_get_ox_survive = NULL,
  dur_not_get_ox_die = NULL,
  dur_get_mv_survive = NULL,
  tt_dur_get_mv_survive = NULL,
  dur_get_mv_die = NULL,
  tt_dur_get_mv_die = NULL,
  dur_not_get_mv_survive = NULL,
  dur_not_get_mv_die = NULL,
  dur_rec = NULL,
  dur_R = NULL,
```

```
  hosp_bed_capacity = NULL,
  ICU_bed_capacity = NULL,
  tt_hosp_beds = 0,
  tt_ICU_beds = 0
)
```

Arguments

country	Character for country beign simulated. WIll be used to generate population and contact_matrix_set if unprovided. Either country or population and contact_matrix_set must be provided.
population	Population vector (for each age group). Default = NULL, which will cause population to be sourced from country
tt_contact_matrix	Time change points for matrix change. Default = 0
contact_matrix_set	Contact matrices used in simulation. Default = NULL, which will generate this based on the country.
R0	Basic Reproduction Number. Default = 3
tt_R0	Change time points for R0. Default = 0
beta_set	Alternative parameterisation via beta rather than R0. Default = NULL, which causes beta to be estimated from R0
time_period	Length of simulation. Default = 365
dt	Time Step. Default = 0.1
init	Data.frame of initial conditions. Default = NULL
seeding_cases	Initial number of cases seeding the epidemic
prob_hosp	Probability of hospitalisation by age. Default, NULL, will use c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)
prob_severe	Probability of developing severe symptoms by age. Default, NULL, will use c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)
prob_non_severe_death_treatment	Probability of death from non severe treated infection. Default, NULL, will use c(0.181354223, 0.181354223, 0.181354223, 0.137454906,0.121938236, 0.122775613, 0.136057441, 0.160922182, 0.196987378,0.242011054, 0.289368845, 0.326537862, 0.337229819, 0.309082553,0.243794865, 0.160480254, 0.057084366)
prob_non_severe_death_no_treatment	Probability of death in non severe hospital inections that aren't treated. Default, NULL, will use rep(0.5, 17)
prob_severe_death_treatment	Probability of death from severe infection that is treated. Default, NULL, will use c(0.226668959, 0.252420241, 0.281097009, 0.413005389,0.518451493,

0.573413613, 0.576222065, 0.54253573, 0.493557696,0.447376527, 0.416666608, 0.411186639, 0.443382594, 0.538718871,0.570434076, 0.643352843, 0.992620047)

prob_severe_death_no_treatment
: Probability of death from severe infection that is not treated. Default, NULL, will use rep(0.95, 17)

p_dist
: Preferentiality of age group receiving treatment relative to other age groups when demand exceeds healthcare capacity.

walker_params
: Boolean for using parameters in Walker et al. Default = FALSE, which uses parameter update as of November 2020. For full information see parameters vignette

dur_E
: Mean duration of incubation period (days). Default = 4.6

dur_IMild
: Mean duration of mild infection (days). Default = 2.1

dur_ICase
: Mean duration from symptom onset to hospitil admission (days). Default = 4.5

dur_get_ox_survive
: Mean duration of oxygen given survive. Default = 9

tt_dur_get_ox_survive
: Times at which dur_get_ox_survive changes (Default = 0 = doesn't change)

dur_get_ox_die
: Mean duration of oxygen given death. Default = 9

tt_dur_get_ox_die
: Times at which dur_get_ox_die changes (Default = 0 = doesn't change)

dur_not_get_ox_survive
: Mean duration without oxygen given survive. Default = 4.5

dur_not_get_ox_die
: Mean duration without oxygen given death. Default = 4.5

dur_get_mv_survive
: Mean duration of ventilation given survive. Default = 14.8

tt_dur_get_mv_survive
: Times at which dur_get_mv_survive changes (Default = 0 = doesn't change)

dur_get_mv_die
: Mean duration of ventilation given death. Default = 11.1

tt_dur_get_mv_die
: Times at which dur_get_mv_die changes (Default = 0 = doesn't change)

dur_not_get_mv_survive
: Mean duration without ventilation given survive. Default = 7.4

dur_not_get_mv_die
: Mean duration without ventilation given death. Default = 1

dur_rec
: Duration of recovery after coming off ventilation. Default = 3

dur_R
: Mean duration of immunity (days). Default = Inf

hosp_bed_capacity
: General bed capacity. Can be single number or vector if capacity time-varies.

ICU_bed_capacity
: ICU bed capacity. Can be single number or vector if capacity time-varies.

tt_hosp_beds
: Times at which hospital bed capacity changes (Default = 0 = doesn't change)

tt_ICU_beds
: Times at which ICU bed capacity changes (Default = 0 = doesn't change)

Details

All durations are in days.

Value

Paramater List

Parameter Updates

Parameters detailing the age-dependent probability of disease severity and durations of hospital durations have been updated in v0.5.0 of `squire` to reflect the changing understanding of COVID-19 transmission. Parameter arguments are by default equal to `NULL`, which causes the new updated parameters specified in `default_probs` and `default_durations` to be used. If any provided parameters are not `NULL`, these will be used. In order to ease previous fits and code, function argument `walker_params` will use the parameters described in Walker et al. Science. 2020 which can be viewed within the function `parse_country_severity`

`parameters_simple_SEEIR` *Parmeters for the simple SEEIR model*

Description

Parmeters for the simple SEEIR model

Usage

```
parameters_simple_SEEIR(
  R0 = 3,
  tt_R0 = 0,
  dt = 0.1,
  init = NULL,
  dur_E = 4.58,
  dur_I = 2.09,
  day_return = FALSE,
  population,
  contact_matrix_set,
  tt_contact_matrix = 0,
  time_period = 365
)
```

Arguments

`R0`	Basic reproduction number
`tt_R0`	Change time points for R0
`dt`	Time step

init	Data.frame of initial conditions
dur_E	Mean duration of incubation period (days)
dur_I	Mean duration of infectious period (days) in simple model
day_return	Logical, do we want to return outut after each day rather than each dt. Default = FALSE
population	Population vector (for each age group)
contact_matrix_set	Contact matrices used in simulation
tt_contact_matrix	Time change points for matrix change
time_period	Length of simulation

Value

Paramater List

`parse_country_severity` *Parse country severity parameters*

Description

Parse country severity parameters

Usage

```
parse_country_severity(
  country = NULL,
  prob_hosp = NULL,
  prob_severe = NULL,
  prob_non_severe_death_treatment = NULL,
  prob_severe_death_treatment = NULL,
  prob_non_severe_death_no_treatment = NULL,
  prob_severe_death_no_treatment = NULL,
  walker_params = FALSE
)
```

Arguments

country	Character for country beign simulated. WIll be used to generate `population` and `contact_matrix_set` if unprovided. Either `country` or `population` and `contact_matrix_set` must be provided.
prob_hosp	Probability of hospitalisation by age. Default, NULL, will use `c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)`

prob_severe Probability of developing severe symptoms by age. Default, NULL, will use c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)

prob_non_severe_death_treatment
Probability of death from non severe treated infection. Default, NULL, will use c(0.181354223, 0.181354223, 0.181354223, 0.137454906,0.121938236, 0.122775613, 0.136057441, 0.160922182, 0.196987378,0.242011054, 0.289368845, 0.326537862, 0.337229819, 0.309082553,0.243794865, 0.160480254, 0.057084366)

prob_severe_death_treatment
Probability of death from severe infection that is treated. Default, NULL, will use c(0.226668959, 0.252420241, 0.281097009, 0.413005389,0.518451493, 0.573413613, 0.576222065, 0.54253573, 0.493557696,0.447376527, 0.416666608, 0.411186639, 0.443382594, 0.538718871,0.570434076, 0.643352843, 0.992620047)

prob_non_severe_death_no_treatment
Probability of death in non severe hospital inections that aren't treated. Default, NULL, will use rep(0.5, 17)

prob_severe_death_no_treatment
Probability of death from severe infection that is not treated. Default, NULL, will use rep(0.95, 17)

walker_params Boolean for using parameters in Walker et al. Default = FALSE, which uses parameter update as of November 2020. For full information see parameters vignette

---

particle_filter *Run a particle filter*

Description

Run a particle filter

Usage

```
particle_filter(
  data,
  model,
  compare,
  n_particles,
  forecast_days = 0,
  save_particles = FALSE,
  full_output = FALSE,
  save_sample_state = FALSE,
  save_end_states = FALSE
)
```

Arguments

data	Data to fit to. This must be constructed with particle_filter_data
model	An odin model, used to generate stochastic samples
compare	A function to generate log-weights
n_particles	Number of particles
forecast_days	Number of days to forecast forward from end states. Requires that save_particles is TRUE.
save_particles	Logical, indicating if we save full particle histories (this is slower).
full_output	Logical, indicating whether the full model output, including the state and the declared outputs are returned. Deafult = FALSE
save_sample_state	Logical, indicating whether we should save a single particle, chosen at random, at the final time point for which we have data
save_end_states	Logical, indicating whether we should save all particles at the final time point for which we have data

---

particle_filter_data *Prepare particle filter data*

Description

Prepare data for use with the particle filter. This function exists to make explicit how time changes through the model relative to the data and to odin's internal clock.

Usage

```
particle_filter_data(data, start_date, steps_per_day)
```

Arguments

data	A data.frame of observed data. There must be a column date, containing dates (or ISO-formatted strings for conversion with as.Date.
start_date	The date to start the simulation from. Must be earlier than the first date in data.
steps_per_day	The number of steps per day

plot.squire_scan *squire scan plot*

Description

squire scan plot

Usage

```
## S3 method for class 'squire_scan'
plot(x, ..., what = "likelihood", log = FALSE, show = c(1, 2))
```

Arguments

x	An squire_scan object
...	additional arguments affecting the plot produced.
what	What scan outputs are we plotting of "likelihood" vs "probability"
log	Should the axes by plotted on log scale
show	Which dimensions of the scan to show. Default = c(1, 2)

plot.squire_simulation *squire simulation plot*

Description

squire simulation plot

Usage

```
## S3 method for class 'squire_simulation'
plot(
  x,
  var_select = NULL,
  replicates = FALSE,
  summarise = TRUE,
  ci = TRUE,
  q = c(0.025, 0.975),
  summary_f = mean,
  x_var = "t",
  particle_fit = FALSE,
  ...
)
```

Arguments

x	An squire_simulation object
var_select	Vector of variable names to plot (default is all)
replicates	Plot replicates
summarise	Logical, add summary line
ci	logical add confidence interval ribbon
q	Quantiles for upper and lower of interval ribbon
summary_f	Function to summarise each compartment passed to the fun argument of stat_summary
x_var	X variable to use for plotting (default is "t", but can be set to, "date", if date_0 provided), which will cause the date to be plotted rather than time.
particle_fit	If the squire_simulation provided is the result of running the particle filter, do we want to just plot the fit. Default = FALSE
...	additional arguments affecting the plot produced.

---

pmcmc *Run a Particle MCMC Sampler within the Squire Framework*

Description

The user inputs initial parameter values for R0, Meff, and the start date The log prior likelihood of these parameters is calculated based on the user-defined prior distributions. The log likelihood of the data given the initial parameters is estimated using a particle filter, which has two functions: - Firstly, to generate a set of 'n_particles' samples of the model state space, at time points corresponding to the data, one of which is selected randomly to serve as the proposed state sequence sample at the final data time point. - Secondly, to produce an unbiased estimate of the likelihood of the data given the proposed parameters. The log posterior of the initial parameters given the data is then estimated by adding the log prior and log likelihood estimate.

The pMCMC sampler then proceeds as follows, for n_mcmc iterations: At each loop iteration the pMCMC sampler pefsorms three steps: 1. Propose new candidate samples for R0, Meff, Meff_pl, and start_date based on the current samples, using the proposal distribution (currently multivariate Gaussian with user-input covariance matrix (proposal_kernel), and reflecting boundaries defined by pars_min, pars_max) 2. Calculate the log prior of the proposed parameters, Use the particle filter to estimate log likelihood of the data given the proposed parameters, as described above, as well as proposing a model state space. Add the log prior and log likelihood estimate to estimate the log posterior of the proposed parameters given the data. 3. Metropolis-Hastings step: The joint canditate sample (consisting of the proposed parameters and state space) is then accepted with probability min(1, a), where the acceptance ratio is simply the ratio of the posterior likelihood of the proposed parameters to the posterior likelihood of the current parameters. Note that by choosing symmetric proposal distributions by including reflecting boundaries, we avoid the the need to include the proposal likelihood in the MH ratio.

If the proposed parameters and states are accepted then we update the current parameters and states to match the proposal, otherwise the previous parameters/states are retained for the next iteration.

After generating the pMCMC simulation, there are replicates specific iterations sampled based on the posterior probability. The parameters from those iterations are then used to generate new trajectories within the squire_model framework.

Usage

```
pmcmc(
  data,
  n_mcmc,
  log_likelihood = NULL,
  log_prior = NULL,
  n_particles = 100,
  steps_per_day = 4,
  output_proposals = FALSE,
  n_chains = 1,
  squire_model = explicit_model(),
 pars_obs = list(phi_cases = 1, k_cases = 2, phi_death = 1, k_death = 2, exp_noise =
    1e+06),
 pars_init = list(start_date = as.Date("2020-02-07"), R0 = 2.5, Meff = 2, Meff_pl = 3,
    R0_pl_shift = 0),
 pars_min = list(start_date = as.Date("2020-02-01"), R0 = 0, Meff = 1, Meff_pl = 2,
    R0_pl_shift = -2),
 pars_max = list(start_date = as.Date("2020-02-20"), R0 = 5, Meff = 3, Meff_pl = 4,
    R0_pl_shift = 5),
 pars_discrete = list(start_date = TRUE, R0 = FALSE, Meff = FALSE, Meff_pl = FALSE,
    R0_pl_shift = FALSE),
 proposal_kernel = NULL,
 scaling_factor = 1,
 reporting_fraction = 1,
 treated_deaths_only = FALSE,
 country = NULL,
 population = NULL,
 contact_matrix_set = NULL,
 baseline_contact_matrix = NULL,
 date_contact_matrix_set_change = NULL,
 R0_change = NULL,
 date_R0_change = NULL,
 hosp_bed_capacity = NULL,
 baseline_hosp_bed_capacity = NULL,
 date_hosp_bed_capacity_change = NULL,
 ICU_bed_capacity = NULL,
 baseline_ICU_bed_capacity = NULL,
 date_ICU_bed_capacity_change = NULL,
 date_vaccine_change = NULL,
 baseline_max_vaccine = NULL,
 max_vaccine = NULL,
 date_vaccine_efficacy_infection_change = NULL,
 baseline_vaccine_efficacy_infection = NULL,
 vaccine_efficacy_infection = NULL,
 date_vaccine_efficacy_disease_change = NULL,
 baseline_vaccine_efficacy_disease = NULL,
 vaccine_efficacy_disease = NULL,
 Rt_args = NULL,
```

```
  burnin = 0,
  replicates = 100,
  forecast = 0,
  required_acceptance_ratio = 0.23,
  start_adaptation = round(n_mcmc/2),
  gibbs_sampling = FALSE,
  gibbs_days = NULL,
  ...
)
```

Arguments

data	Data to fit to. This must be constructed with particle_filter_data
n_mcmc	number of mcmc mcmc iterations to perform
log_likelihood	function to calculate log likelihood, must take named parameter vector as input, allow passing of implicit arguments corresponding to the main function arguments. Returns a named list, with entries: - $log_likelihood, a single numeric - $sample_state, a numeric vector corresponding to the state of a single particle, chosen at random, at the final time point for which we have data. If NULL, calculated using the function calc_loglikelihood.
log_prior	function to calculate log prior, must take named parameter vector as input, returns a single numeric. If NULL, uses uninformative priors which do not affect the posterior
n_particles	Number of particles (considered for both the PMCMC fit and sampling from posterior)
steps_per_day	Number of steps per day
output_proposals	Logical indicating whether proposed parameter jumps should be output along with results
n_chains	number of MCMC chains to run
squire_model	A squire model to use
pars_obs	list of parameters to use in comparison with compare. Must be a list containing, e.g. list(phi_cases = 0.1, k_cases = 2, phi_death = 1, k_death = 2, exp_noise = 1e6)
pars_init	named list of initial inputs for parameters being sampled
pars_min	named list of lower reflecting boundaries for parameter proposals
pars_max	named list of upper reflecting boundaries for parameter proposals
pars_discrete	named list of logicals, indicating if proposed jump should be discrete
proposal_kernel	named matrix of proposal covariance for parameters
scaling_factor	numeric for starting scaling factor for covariance matrix. Default = 1
reporting_fraction	Reporting fraction. Numeric for what proportion of the total deaths the reported deaths represent. E.g. 0.5 results in the model calibrating to twice the deaths provided by data$deaths

treated_deaths_only
: Boolean for whether likelihood is based only on deaths that occur from health-care systems, i.e. are treated. Default = FALSE, which uses all deaths.

country
: Character for country beign simulated. WIll be used to generate population and contact_matrix_set if unprovided. Either country or population and contact_matrix_set must be provided.

population
: Population vector (for each age group). Default = NULL, which will cause population to be sourced from country

contact_matrix_set
: List of contact matrices to be used from the dates provided in date_contact_matrix_set_change.Must be same length as date_contact_matrix_set_change

baseline_contact_matrix
: The starting contact matrix prior to any changes due to interventions or otherwise. Default = NULL, which will use the contact matrix associated with the coutnry provided.

date_contact_matrix_set_change
: Calendar dates at which the contact matrices set in model_params change. Defaut = NULL, i.e. no change

R0_change
: Numeric vector for relative changes in R0. Default = NULL, i.e. no change in R0

date_R0_change
: Calendar dates at which R0_change occurs. Defaut = NULL, i.e. no change in R0

hosp_bed_capacity
: Number of hospital beds at each date specified in date_hosp_bed_capacity_change. Must be same length as date_hosp_bed_capacity_change.

baseline_hosp_bed_capacity
: The starting number of hospital beds before the epidemic started. Default = NULL, which will use the hospital beds data for the country provided. If no country is provided then this is 5/1000 of the population

date_hosp_bed_capacity_change
: Calendar dates at which hospital bed capacity changes set in model_params change. Defaut = NULL, i.e. no change

ICU_bed_capacity
: Number of ICU beds at each date specified in date_ICU_bed_capacity_change. Must be same length as date_ICU_bed_capacity_change.

baseline_ICU_bed_capacity
: The starting number of ICU beds before the epidemic started. Default = NULL, which will use the hospital beds data for the country provided. If no country is provided then this is 3/100 of hospital beds

date_ICU_bed_capacity_change
: Calendar dates at which ICU bed capacity changes set in model_params change. Defaut = NULL, i.e. no change

date_vaccine_change
: Date that vaccine doses per day change. Default = NULL.

baseline_max_vaccine
: Baseline vaccine doses per day. Default = NULL

`max_vaccine` Time varying maximum vaccine doeses per day. Default = NULL.

`date_vaccine_efficacy_infection_change`
Date that vaccine efficacy against infection changes. Default = NULL.

`baseline_vaccine_efficacy_infection`
Baseline vaccine effacy against infection. Default = NULL

`vaccine_efficacy_infection`
Time varying vaccine efficacy against infection. Default = NULL.

`date_vaccine_efficacy_disease_change`
Date that vaccine efficacy against disease changes. Default = NULL.

`baseline_vaccine_efficacy_disease`
Baseline vaccine efficacy against disease Default = NULL

`vaccine_efficacy_disease`
Time varying vaccine efficacy against infection. Default = NULL.

`Rt_args` List of arguments to be passed to `evaluate_Rt_pmcmc` for calculating Rt. Current arguments are available in `Rt_args_list`

`burnin` number of iterations to discard from the start of MCMC run when sampling from the posterior for trajectories

`replicates` number of trajectories (replicates) to be returned that are being sampled from the posterior probability results produced by `run_mcmc_chain` to select parameter set. For each parmater set sampled, run particle filter with `n_particles` and sample 1 trajectory

`forecast` Number of days to forecast forward. Default = 0

`required_acceptance_ratio`
Desired MCMC acceptance ratio

`start_adaptation`
Iteration number to begin RM optimisation of scaling factor at

`gibbs_sampling` Whether or not to use the Gibbs Sampler for start_date

`gibbs_days` Number of days either side of the start_date parameter to evaluate likelihood at

`...` Further aguments for the model parameter function. If using the `explicit_model` (default) this will be `parameters_explicit_SEEIR`.

Details

Run a pmcmc sampler with the Squire model setup (i.e. include the various model parameters for the odin model to generate curves)

Value

squire_simulation

output Trajectories from the sampled pMCMC parameter iterations.

parameters Model parameters use for squire

model Squire model used

inputs Inputs into the squire model for the pMCMC.

pMCMC_results An mcmc object generated from pmcmc and contains:

inputs List of inputs

chains List that include:

results Matrix of accepted parameter samples, rows = iterations as well as log prior, (particle filter estimate of) log likelihood and log posterior

states Matrix of compartment states

acceptance_rate MCMC acceptance rate

ess MCMC chain effective sample size

rhat MCMC Diagnostics

interventions Contains the interventions that can be called with projections.

replicate_parameters contains the parameter values for the sampled pMCMC parameter iterations used to generate the squire_model trajectories

---

`population` *Population*

Description

Population by age group and country

Usage

```
population
```

Format

A data.frame

`pos_num` *Check argument is a single positive numeric*

Description

Check argument is a single positive numeric

Usage

```
pos_num(x, name = deparse(substitute(x)))
```

Arguments

x	argument
name	Name of argument

Value

Nothing if check pass

print.squire_simulation *squire simulation print*

Description

squire simulation print

Usage

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

Arguments

x	An iccm_simulation object
...	additional arguments affecting the summary produced.

process_contact_matrix *Process a contact matrix*

Description

Process a contact matrix

Usage

```
process_contact_matrix(contact_matrix, population)
```

Arguments

contact_matrix	A contact matrix
population	Vector of population by age

Value

Processed matrix

process_contact_matrix_scaled_age *Process a contact matrix with an extra*

Description

Process a contact matrix with an extra

Usage

```
process_contact_matrix_scaled_age(contact_matrix, population)
```

Arguments

contact_matrix	A contact matrix
population	Vector of population by age

Value

Processed matrix

projection_plotting *Plot projections against each other*

Description

Plot projections against each other

Usage

```
projection_plotting(
  r_list,
  scenarios,
  add_parms_to_scenarios = TRUE,
  var_select = NULL,
  replicates = FALSE,
  summarise = TRUE,
  ci = TRUE,
  q = c(0.025, 0.975),
  summary_f = mean,
  date_0 = Sys.Date(),
  x_var = "t",
  ...
)
```

Arguments

r_list	List of different projection runs from projections
scenarios	Character vector describing the different scenarios.
add_parms_to_scenarios	Logical. Should the parameters used for the projection runs be added to scenarios. Default = TRUE
var_select	Vector of variable names to plot (default is all)
replicates	Plot replicates
summarise	Logical, add summary line
ci	logical add confidence interval ribbon
q	Quantiles for upper and lower of interval ribbon
summary_f	Function to summarise each compartment passed to the fun argument of stat_summary
date_0	Date of time 0, if specified a date column will be added
x_var	X variable to use for plotting (default is "t", but can be set to, "date", if date_0 provided), which will cause the date to be plotted rather than time.
...	additional arguments passed to format_output

---

projections *Provide projections from calibrated simulations by changing RO, contact matrices or bed availability.*

Description

This extends previous projections as you can pass in lists of each argument that then get passed to each simulation replicate.

Usage

```
projections(
  r,
  time_period = NULL,
  R0 = NULL,
  R0_change = NULL,
  tt_R0 = 0,
  contact_matrix_set = NULL,
  contact_matrix_set_change = NULL,
  tt_contact_matrix = 0,
  hosp_bed_capacity = NULL,
  hosp_bed_capacity_change = NULL,
  tt_hosp_beds = 0,
  ICU_bed_capacity = NULL,
  ICU_bed_capacity_change = NULL,
  tt_ICU_beds = 0,
  to_be_run = TRUE,
  model_user_args = NULL
)
```

Arguments

r	Calibrated {squire_simulation} object.
time_period	How many days is the projection. Deafult = NULL, which will carry the projection forward from t = 0 in the calibration (i.e. the number of days set in calibrate using forecast)
R0	Numeric vector for R0 from t = 0 in the calibration. E.g. R0 = c(2, 1). Default = NULL, which will use R0_change to alter R0 if provided.
R0_change	Numeric vector for relative changes in R0 relative to the final R0 used in the calibration (i.e. at t = 0 in the calibration) E.g. R0 = c(0.8, 0.5). Default = NULL, which will use R0 to parameterise changes in R0 if provided.
tt_R0	Change time points for R0
contact_matrix_set	Contact matrices used in simulation. Default = NULL, which will use contact_matrix_set_change to alter the contact matrix if provided.
contact_matrix_set_change	Numeric vector for relative changes in the contact matrix realtive to the final contact matrix used in the calibration (i.e. at t = 0 in the calibration). E.g. contact_matrix_set_change = c(0.8, 0.5). Default = NULL, which will use contact_matrix_set to parameterise changes in contact matrices if if provided.
tt_contact_matrix	Time change points for matrix change. Default = 0
hosp_bed_capacity	Numeric vector for hospital bed capacity from t = 0 in the calibration. Default = NULL, which will use hosp_bed_capacity_change to alter hosp_bed_capacity if provided.
hosp_bed_capacity_change	Numeric vector for relative changes in hospital bed capacity relative to the final hospital bed capacity used in the calibration (i.e. at t = 0 in the calibration). E.g. hosp_bed_capacity_change = c(0.8, 0.5). Default = NULL, which will use hosp_bed_capacity to parameterise changes in hospital bed capacity if provided.
tt_hosp_beds	Change time points for hosp_bed_capacity
ICU_bed_capacity	Numeric vector for ICU bed capacity from t = 0 in the calibration. Default = NULL, which will use ICU_bed_capacity_change to alter ICU_bed_capacity if provided.
ICU_bed_capacity_change	Numeric vector for relative changes in ICU bed capacity relative to the final ICU bed capacity used in the calibration (i.e. at t = 0 in the calibration). E.g. ICU_bed_capacity_change = c(0.8, 0.5). Default = NULL, which will use ICU_bed_capacity to parameterise changes in ICU bed capacity if provided.
tt_ICU_beds	Change time points for ICU_bed_capacity
to_be_run	List of logicals for whether each replicate should be run. Default = TRUE, which causes all replictes to be run.

model_user_args

List of other parameters to be passed to the model to be run. Default = NULL. An example would be:

`list( list( "prob_severe" = runif(17), "tt_dur_get_ox_survive" = c(0, 10), "gamma_get_ox_survive" = 0.2), list( "prob_severe" = runif(17), "tt_dur_get_mv_survive" = c(0, 5), "gamma_get_mv_survive" = 0.1) )`

The list should be the same length as the number of replicates in the simulations. Each list element should then be a list with elements named to match the arguments expected by the odin model with `r`. Above would be suitable to set the model parameters for a simulation with 2 replicates. You do not have to have the same arguments in each list.

Details

The user can specify changes to R0, contact matrices and bed provision, which will come into effect from the current day in the calibration. These changes can either set these to be specific values or change them relative to their values in the original simulation. If no change is requested, the simulation will use parameters chosen for the calibration run. This extends previous versions of `projections` as you can now pass in lists of each argument that then get passed to each simulation replicate.

---

run_deterministic_comparison

Run Deterministic model comparison

Description

Create a deterministic model and compare to data

Usage

```
run_deterministic_comparison(
  data,
  squire_model,
  model_params,
  model_start_date = "2020-02-02",
 obs_params = list(phi_cases = 0.1, k_cases = 2, phi_death = 1, k_death = 2, exp_noise
    = 1e+06),
  forecast_days = 0,
  save_history = FALSE,
  return = "ll"
)
```

Arguments

data	to fit to.
squire_model	A squire model to use

model_params	Squire model parameters. Created from a call to one of the parameters_<type>_model functions.
model_start_date	Date to run model simulations from
obs_params	List of parameters used for comparing model to data
forecast_days	Days ahead to include in output
save_history	Whether to save full history. Default = FALSE
return	Set return depending on what is needed. 'full' and "sample" gives the entire output, 'll' gives the log-likelihood

Value

Results from particle filter

`run_deterministic_SEIR_model` *Run the deterministic explicit SEIR model*

Description

Run the deterministic explicit SEIR model

Usage

```
run_deterministic_SEIR_model(
  country = NULL,
  population = NULL,
  tt_contact_matrix = 0,
  contact_matrix_set = NULL,
  R0 = 3,
  tt_R0 = 0,
  beta_set = NULL,
  time_period = 365,
  dt = 0.1,
  day_return = FALSE,
  replicates = 10,
  init = NULL,
  seed = stats::runif(1, 0, 1e+08),
  prob_hosp = NULL,
  prob_severe = NULL,
  prob_non_severe_death_treatment = NULL,
  prob_non_severe_death_no_treatment = NULL,
  prob_severe_death_treatment = NULL,
  prob_severe_death_no_treatment = NULL,
  p_dist = probs$p_dist,
  walker_params = FALSE,
```

```
  dur_E = 4.6,
  dur_IMild = 2.1,
  dur_ICase = 4.5,
  dur_get_ox_survive = NULL,
  tt_dur_get_ox_survive = 0,
  dur_get_ox_die = NULL,
  tt_dur_get_ox_die = 0,
  dur_not_get_ox_survive = NULL,
  dur_not_get_ox_die = NULL,
  dur_get_mv_survive = NULL,
  tt_dur_get_mv_survive = 0,
  dur_get_mv_die = NULL,
  tt_dur_get_mv_die = 0,
  dur_not_get_mv_survive = NULL,
  dur_not_get_mv_die = NULL,
  dur_rec = NULL,
  dur_R = NULL,
  hosp_bed_capacity = NULL,
  ICU_bed_capacity = NULL,
  tt_hosp_beds = 0,
  tt_ICU_beds = 0,
  seeding_cases = NULL,
  mod_gen = explicit_SEIR_deterministic
)
```

Arguments

country	Character for country beign simulated. WIll be used to generate population and contact_matrix_set if unprovided. Either country or population and contact_matrix_set must be provided.
population	Population vector (for each age group). Default = NULL, which will cause population to be sourced from country
tt_contact_matrix	Time change points for matrix change. Default = 0
contact_matrix_set	Contact matrices used in simulation. Default = NULL, which will generate this based on the country.
R0	Basic Reproduction Number. Default = 3
tt_R0	Change time points for R0. Default = 0
beta_set	Alternative parameterisation via beta rather than R0. Default = NULL, which causes beta to be estimated from R0
time_period	Length of simulation. Default = 365
dt	Time Step. Default = 0.1
day_return	Logical, do we want to return outut after each day rather than each dt. Default = FALSE
replicates	Number of replicates

`init` Data.frame of initial conditions. Default = NULL

`seed` Random Number Seed.

`prob_hosp` Probability of hospitalisation by age. Default, NULL, will use `c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)`

`prob_severe` Probability of developing severe symptoms by age. Default, NULL, will use `c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)`

`prob_non_severe_death_treatment`
Probability of death from non severe treated infection. Default, NULL, will use `c(0.181354223, 0.181354223, 0.181354223, 0.137454906,0.121938236, 0.122775613, 0.136057441, 0.160922182, 0.196987378,0.242011054, 0.289368845, 0.326537862, 0.337229819, 0.309082553,0.243794865, 0.160480254, 0.057084366)`

`prob_non_severe_death_no_treatment`
Probability of death in non severe hospital inections that aren't treated. Default, NULL, will use `rep(0.5, 17)`

`prob_severe_death_treatment`
Probability of death from severe infection that is treated. Default, NULL, will use `c(0.226668959, 0.252420241, 0.281097009, 0.413005389,0.518451493, 0.573413613, 0.576222065, 0.54253573, 0.493557696,0.447376527, 0.416666608, 0.411186639, 0.443382594, 0.538718871,0.570434076, 0.643352843, 0.992620047)`

`prob_severe_death_no_treatment`
Probability of death from severe infection that is not treated. Default, NULL, will use `rep(0.95, 17)`

`p_dist` Preferentiality of age group receiving treatment relative to other age groups when demand exceeds healthcare capacity.

`walker_params` Boolean for using parameters in Walker et al. Default = FALSE, which uses parameter update as of November 2020. For full information see parameters vignette

`dur_E` Mean duration of incubation period (days). Default = 4.6

`dur_IMild` Mean duration of mild infection (days). Default = 2.1

`dur_ICase` Mean duration from symptom onset to hospitil admission (days). Default = 4.5

`dur_get_ox_survive`
Mean duration of oxygen given survive. Default = 9

`tt_dur_get_ox_survive`
Times at which dur_get_ox_survive changes (Default = 0 = doesn't change)

`dur_get_ox_die` Mean duration of oxygen given death. Default = 9

`tt_dur_get_ox_die`
Times at which dur_get_ox_die changes (Default = 0 = doesn't change)

`dur_not_get_ox_survive`
Mean duration without oxygen given survive. Default = 4.5

`dur_not_get_ox_die`
Mean duration without oxygen given death. Default = 4.5

dur_get_mv_survive
: Mean duration of ventilation given survive. Default = 14.8

tt_dur_get_mv_survive
: Times at which dur_get_mv_survive changes (Default = 0 = doesn't change)

dur_get_mv_die
: Mean duration of ventilation given death. Default = 11.1

tt_dur_get_mv_die
: Times at which dur_get_mv_die changes (Default = 0 = doesn't change)

dur_not_get_mv_survive
: Mean duration without ventilation given survive. Default = 7.4

dur_not_get_mv_die
: Mean duration without ventilation given death. Default = 1

dur_rec
: Duration of recovery after coming off ventilation. Default = 3

dur_R
: Mean duration of immunity (days). Default = Inf

hosp_bed_capacity
: General bed capacity. Can be single number or vector if capacity time-varies.

ICU_bed_capacity
: ICU bed capacity. Can be single number or vector if capacity time-varies.

tt_hosp_beds
: Times at which hospital bed capacity changes (Default = 0 = doesn't change)

tt_ICU_beds
: Times at which ICU bed capacity changes (Default = 0 = doesn't change)

seeding_cases
: Initial number of cases seeding the epidemic

mod_gen
: An odin model generation function. Default: 'explicit_SEIR_deterministic'

Value

Simulation output

Examples

```
## Not run:
pop <- get_population("Afghanistan")
m <- get_mixing_matrix("Afghanistan")
run_deterministic_SEIR_model(pop$n, m, c(0, 50), c(3, 3/2), 365, 100000,
1000000)

## End(Not run)
```

---

run_explicit_SEEIR_model

Run the explicit SEEIR model

Description

Run the explicit SEEIR model

Usage

```
run_explicit_SEEIR_model(
  country = NULL,
  population = NULL,
  tt_contact_matrix = 0,
  contact_matrix_set = NULL,
  R0 = 3,
  tt_R0 = 0,
  beta_set = NULL,
  time_period = 365,
  dt = 0.1,
  day_return = FALSE,
  replicates = 10,
  init = NULL,
  seed = stats::runif(1, 0, 1e+08),
  prob_hosp = NULL,
  prob_severe = NULL,
  prob_non_severe_death_treatment = NULL,
  prob_non_severe_death_no_treatment = NULL,
  prob_severe_death_treatment = NULL,
  prob_severe_death_no_treatment = NULL,
  p_dist = probs$p_dist,
  walker_params = FALSE,
  dur_E = NULL,
  dur_IMild = NULL,
  dur_ICase = NULL,
  dur_get_ox_survive = NULL,
  tt_dur_get_ox_survive = 0,
  dur_get_ox_die = NULL,
  tt_dur_get_ox_die = 0,
  dur_not_get_ox_survive = NULL,
  dur_not_get_ox_die = NULL,
  dur_get_mv_survive = NULL,
  tt_dur_get_mv_survive = 0,
  dur_get_mv_die = NULL,
  tt_dur_get_mv_die = 0,
  dur_not_get_mv_survive = NULL,
  dur_not_get_mv_die = NULL,
  dur_rec = NULL,
  dur_R = NULL,
  hosp_bed_capacity = NULL,
  ICU_bed_capacity = NULL,
  tt_hosp_beds = 0,
  tt_ICU_beds = 0,
  seeding_cases = NULL
)
```

Arguments

country	Character for country beign simulated. WIll be used to generate population and contact_matrix_set if unprovided. Either country or population and contact_matrix_set must be provided.
population	Population vector (for each age group). Default = NULL, which will cause population to be sourced from country
tt_contact_matrix	Time change points for matrix change. Default = 0
contact_matrix_set	Contact matrices used in simulation. Default = NULL, which will generate this based on the country.
R0	Basic Reproduction Number. Default = 3
tt_R0	Change time points for R0. Default = 0
beta_set	Alternative parameterisation via beta rather than R0. Default = NULL, which causes beta to be estimated from R0
time_period	Length of simulation. Default = 365
dt	Time Step. Default = 0.1
day_return	Logical, do we want to return outut after each day rather than each dt. Default = FALSE
replicates	Number of replicates
init	Data.frame of initial conditions. Default = NULL
seed	Random Number Seed.
prob_hosp	Probability of hospitalisation by age. Default, NULL, will use c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)
prob_severe	Probability of developing severe symptoms by age. Default, NULL, will use c(0.000840764, 0.001182411, 0.001662887, 0.002338607,0.003288907, 0.004625365, 0.006504897, 0.009148183, 0.012865577,0.018093546, 0.025445917, 0.035785947, 0.050327683, 0.0707785,0.099539573, 0.1399878, 0.233470395)
prob_non_severe_death_treatment	Probability of death from non severe treated infection. Default, NULL, will use c(0.181354223, 0.181354223, 0.181354223, 0.137454906,0.121938236, 0.122775613, 0.136057441, 0.160922182, 0.196987378,0.242011054, 0.289368845, 0.326537862, 0.337229819, 0.309082553,0.243794865, 0.160480254, 0.057084366)
prob_non_severe_death_no_treatment	Probability of death in non severe hospital inections that aren't treated. Default, NULL, will use rep(0.5, 17)
prob_severe_death_treatment	Probability of death from severe infection that is treated. Default, NULL, will use c(0.226668959, 0.252420241, 0.281097009, 0.413005389,0.518451493, 0.573413613, 0.576222065, 0.54253573, 0.493557696,0.447376527, 0.416666608, 0.411186639, 0.443382594, 0.538718871,0.570434076, 0.643352843, 0.992620047)

`prob_severe_death_no_treatment`
: Probability of death from severe infection that is not treated. Default, NULL, will use `rep(0.95, 17)`

`p_dist`
: Preferentiality of age group receiving treatment relative to other age groups when demand exceeds healthcare capacity.

`walker_params`
: Boolean for using parameters in Walker et al. Default = FALSE, which uses parameter update as of November 2020. For full information see parameters vignette

`dur_E`
: Mean duration of incubation period (days). Default = 4.6

`dur_IMild`
: Mean duration of mild infection (days). Default = 2.1

`dur_ICase`
: Mean duration from symptom onset to hospitil admission (days). Default = 4.5

`dur_get_ox_survive`
: Mean duration of oxygen given survive. Default = 9

`tt_dur_get_ox_survive`
: Times at which dur_get_ox_survive changes (Default = 0 = doesn't change)

`dur_get_ox_die`
: Mean duration of oxygen given death. Default = 9

`tt_dur_get_ox_die`
: Times at which dur_get_ox_die changes (Default = 0 = doesn't change)

`dur_not_get_ox_survive`
: Mean duration without oxygen given survive. Default = 4.5

`dur_not_get_ox_die`
: Mean duration without oxygen given death. Default = 4.5

`dur_get_mv_survive`
: Mean duration of ventilation given survive. Default = 14.8

`tt_dur_get_mv_survive`
: Times at which dur_get_mv_survive changes (Default = 0 = doesn't change)

`dur_get_mv_die`
: Mean duration of ventilation given death. Default = 11.1

`tt_dur_get_mv_die`
: Times at which dur_get_mv_die changes (Default = 0 = doesn't change)

`dur_not_get_mv_survive`
: Mean duration without ventilation given survive. Default = 7.4

`dur_not_get_mv_die`
: Mean duration without ventilation given death. Default = 1

`dur_rec`
: Duration of recovery after coming off ventilation. Default = 3

`dur_R`
: Mean duration of immunity (days). Default = Inf

`hosp_bed_capacity`
: General bed capacity. Can be single number or vector if capacity time-varies.

`ICU_bed_capacity`
: ICU bed capacity. Can be single number or vector if capacity time-varies.

`tt_hosp_beds`
: Times at which hospital bed capacity changes (Default = 0 = doesn't change)

`tt_ICU_beds`
: Times at which ICU bed capacity changes (Default = 0 = doesn't change)

`seeding_cases`
: Initial number of cases seeding the epidemic

Details

All durations are in days.

Value

Simulation output

Parameter Updates

Parameters detailing the age-dependent probability of disease severity and durations of hospital durations have been updated in v0.5.0 of `squire` to reflect the changing understanding of COVID-19 transmission. Parameter arguments are by default equal to `NULL`, which causes the new updated parameters specified in `default_probs` and `default_durations` to be used. If any provided parameters are not `NULL`, these will be used. In order to ease previous fits and code, function argument `walker_params` will use the parameters described in Walker et al. Science. 2020 which can be viewed within the function `parse_country_severity`

Examples

```
## Not run:
pop <- get_population("Afghanistan")
m1 <- run_explicit_SEEIR_model(R0 = 2,
population = pop$n, dt = 1,
contact_matrix_set=contact_matrices[[1]])

## End(Not run)
```

---

run_particle_filter *Run particle filter*

Description

Create a model, and fit with the particle filter

Usage

```
run_particle_filter(
  data,
  squire_model,
  model_params,
  model_start_date = "2020-02-02",
 obs_params = list(phi_cases = 0.1, k_cases = 2, phi_death = 1, k_death = 2, exp_noise
    = 1e+06),
  n_particles = 1000,
  forecast_days = 0,
  save_particles = FALSE,
  full_output = FALSE,
  return = "full"
)
```

Arguments

data	to fit to.
squire_model	A squire model to use
model_params	Squire model parameters. Created from a call to one of the parameters_<type>_model functions.
model_start_date	Date to run model simulations from
obs_params	List of parameters used for comparing model to data in the particle filter t
n_particles	Number of particles
forecast_days	Days ahead to include in output
save_particles	Whether to save trajectories
full_output	Logical, indicating whether the full model output, including the state and the declared outputs are returned. Deafult = FALSE
return	Set return depending on what is needed. 'full' gives the entire particle filter output, 'll' gives the log-likelihood, 'sample' gives a sampled particle's trace, 'single' gives the final state

Value

Results from particle filter

---

run_simple_SEEIR_model *Run the SEEIR model*

Description

Run the SEEIR model

Usage

```
run_simple_SEEIR_model(
  R0 = 3,
  tt_R0 = 0,
  dt = 0.1,
  init = NULL,
  dur_E = 4.58,
  dur_I = 2.09,
  day_return = FALSE,
  population,
  contact_matrix_set,
  tt_contact_matrix = 0,
  time_period = 365,
  replicates = 10
)
```

Arguments

R0	Basic reproduction number
tt_R0	Change time points for R0
dt	Time step
init	Data.frame of initial conditions
dur_E	Mean duration of incubation period (days)
dur_I	Mean duration of infectious period (days)
day_return	Logical, do we want to return outut after each day rather than each dt. Default = FALSE
population	Population vector (for each age group)
contact_matrix_set	Contact matrices used in simulation
tt_contact_matrix	Time change points for matrix change
time_period	Length of simulation
replicates	Number of replicates

Value

Simulation output

Examples

```
## Not run:
pop <- get_population("Afghanistan", simple_SEIR = TRUE)
m1 <- run_simple_SEEIR_model(population = pop$n, dt = 1,
R0 = 2,
contact_matrix_set=contact_matrices[[1]])

## End(Not run)
```

---

sample_3d_grid_scan *Sample Grid Scan*

Description

Take a grid search produced by scan_R0_date_Meff and sample n_sample_pairs from the parameter grid uses based on their probability. For each parameter pair chosen, run particle filter with num_particles and sample 1 trajectory

Usage

```
sample_3d_grid_scan(
  scan_results,
  n_sample_pairs = 10,
  n_particles = 100,
  forecast_days = 0,
  full_output = FALSE
)
```

Arguments

scan_results	Output of scan_R0_date_Meff.
n_sample_pairs	Number of parameter pairs to be sampled. This will determine how many trajectories are returned. Integer. Default = 10. This will determine how many trajectories are returned.
n_particles	Number of particles. Positive Integer. Default = 100
forecast_days	Number of days being forecast. Default = 0
full_output	Logical, indicating whether the full model output, including the state and the declared outputs are returned. Deafult = FALSE

Value

list. First element (trajectories) is a 3 dimensional array of trajectories (time, state, tranjectories). Second element (param_grid) is the parameters chosen when sampling from the scan_results grid and the third dimension (inputs) is a list of model inputs.

---

sample_drjacoby *Sample from a drjacoby mcmc*

Description

The drjacoby sample is very similar to [[sample_pmcmc]] but there are a few subtle differences that meant it was easier to have a separate function for using drjacoby for the mcmc process

Usage

```
sample_drjacoby(
  pmcmc_results,
  burnin = 0,
  n_chains,
  log_likelihood = calc_loglikelihood,
  n_trajectories = 10,
  n_particles = 100,
  forecast_days = 0
)
```

Arguments

pmcmc_results	output of run_mcmc_chain; The results from the PMCMC run – can have mutliple chains.
burnin	integer; Number of iterations to discard from the start of MCMC run. Default = 0
n_chains	number of chains that considered. Should inherent from pmcmc.
log_likelihood	function to calculate log likelihood, must take named parameter vector as input, allow passing of implicit arguments corresponding to the main function arguments. Returns a named list, with entries: - $log_likelihood, a single numeric - $sample_state, a numeric vector corresponding to the state of a single particle, chosen at random, at the final time point for which we have data. If NULL, calculated using the function calc_loglikelihood.
n_trajectories	interger; Number of trajectories to be returned. Integer. Default = 10.
n_particles	integer; Number of particles to be considered in the particle filter. Default = 100
forecast_days	integer; number of days being forecast. Default = 0

Details

Sample from a drjacoby mcmc

---

sample_grid_scan *Sample Grid Scan*

Description

Take a grid search produced by scan_R0_date and sample n_sample_pairs from the parameter grid uses based on their probability. For each parameter pair chosen, run particle filter with num_particles and sample 1 trajectory

Usage

```
sample_grid_scan(
  scan_results,
  n_sample_pairs = 10,
  n_particles = 100,
  forecast_days = 0,
  full_output = FALSE
)
```

Arguments

scan_results	Output of scan_R0_date.
n_sample_pairs	Number of parameter pairs to be sampled. This will determine how many trajectories are returned. Integer. Default = 10. This will determine how many trajectories are returned.

n_particles	Number of particles. Positive Integer. Default = 100
forecast_days	Number of days being forecast. Default = 0
full_output	Logical, indicating whether the full model output, including the state and the declared outputs are returned. Deafult = FALSE

Value

list. First element (trajectories) is a 3 dimensional array of trajectories (time, state, tranjectories). Second element (param_grid) is the parameters chosen when sampling from the scan_results grid and the third dimension (inputs) is a list of model inputs.

---

sample_pmcmc *Sample PMCMC*

Description

Sample from the posterior probability results produced by run_mcmc_chain to select parameter set. For each parmater set sampled, run particle filter with num_particles and sample 1 trajectory

Usage

```
sample_pmcmc(
  pmcmc_results,
  burnin = 0,
  n_chains,
  log_likelihood = calc_loglikelihood,
  n_trajectories = 10,
  n_particles = 100,
  forecast_days = 0
)
```

Arguments

pmcmc_results	output of run_mcmc_chain; The results from the PMCMC run – can have mutliple chains.
burnin	integer; Number of iterations to discard from the start of MCMC run. Default = 0
n_chains	number of chains that considered. Should inherent from pmcmc.
log_likelihood	function to calculate log likelihood, must take named parameter vector as input, allow passing of implicit arguments corresponding to the main function arguments. Returns a named list, with entries: - $log_likelihood, a single numeric - $sample_state, a numeric vector corresponding to the state of a single particle, chosen at random, at the final time point for which we have data. If NULL, calculated using the function calc_loglikelihood.
n_trajectories	interger; Number of trajectories to be returned. Integer. Default = 10.
n_particles	integer; Number of particles to be considered in the particle filter. Default = 100
forecast_days	integer; number of days being forecast. Default = 0

Value

trajectories A 3-dimensional array of trajectories (time, state, tranjectories).

sampled_PMCMC_Results The parameters chosen when sampling from the `pmcmc` posteriors

inputs A list of model inputs.

`scan_R0_date` *Grid search of R0 and start date*

Description

Run a grid search of the particle filter over R0 and start date. This is parallelised, first run `plan(multiprocess)` to set this up.

Usage

```
scan_R0_date(
  R0_min,
  R0_max,
  R0_step,
  first_start_date,
  last_start_date,
  day_step,
  data,
  model_params,
 Rt_func = function(R0_change, R0, Meff) {    exp(log(R0) - Meff * (1 - R0_change)) },
  R0_prior = NULL,
  R0_change = NULL,
  date_R0_change = NULL,
  date_contact_matrix_set_change = NULL,
  date_ICU_bed_capacity_change = NULL,
  date_hosp_bed_capacity_change = NULL,
  squire_model = explicit_SEIR(),
  pars_obs = NULL,
  n_particles = 100
)
```

Arguments

`R0_min`	Minimum value of R0 in the search
`R0_max`	Maximum value of R0 in the search
`R0_step`	Step to increment R0 between min and max
`first_start_date`	Earliest start date as 'yyyy-mm-dd'
`last_start_date`	Latest start date as 'yyyy-mm-dd'

day_step	Step to increment date in days
data	Deaths data to fit to. See example_deaths.csv and particle_filter_data()
model_params	Squire model parameters. Created from a call to one of the parameters_<type>_model functions.
Rt_func	Function for converting R0, Meff and R0_change. Function must have names arguments of R0, Meff and R0_change. Default is linear relationship on the log scale given by exp(log(R0) - Meff*(1-R0_change)).
R0_prior	Prior for R0. Default = NULL, which is a flat prior. Should be provided as a list with first argument the distribution function and the second the function arguments (excluding quantiles which are worked out based on R0_min and R0_max), e.g. 'list("func" = dnorm, args = list("mean"= 3.5, "sd"= 3))'.
R0_change	Numeric vector for relative changes in R0. Default = NULL, i.e. no change in R0
date_R0_change	Calendar dates at which R0_change occurs. Defaut = NULL, i.e. no change in R0
date_contact_matrix_set_change	Calendar dates at which the contact matrices set in model_params change. Defaut = NULL, i.e. no change
date_ICU_bed_capacity_change	Calendar dates at which ICU bed capacity changes set in model_params change. Defaut = NULL, i.e. no change
date_hosp_bed_capacity_change	Calendar dates at which hospital bed capacity changes set in model_params change. Defaut = NULL, i.e. no change
squire_model	A squire model. Default = explicit_SEIR()
pars_obs	list of parameters to use for the comparison function.
n_particles	Number of particles. Positive Integer. Default = 100

Value

List of R0 and start date grid values, and normalised probabilities at each point

---

scan_R0_date_Meff *Grid search of R0 and start date*

Description

Run a grid search of the particle filter over R0, start date and Meff. This is parallelised, first run plan(multiprocess) to set this up.

Usage

```
scan_R0_date_Meff(
  R0_min,
  R0_max,
  R0_step,
  first_start_date,
  last_start_date,
  day_step,
  Meff_min,
  Meff_max,
  Meff_step,
  data,
  model_params,
 Rt_func = function(R0_change, R0, Meff) {    exp(log(R0) - Meff * (1 - R0_change)) },
  R0_prior = NULL,
  R0_change = NULL,
  date_R0_change = NULL,
  date_contact_matrix_set_change = NULL,
  date_ICU_bed_capacity_change = NULL,
  date_hosp_bed_capacity_change = NULL,
  squire_model = explicit_SEIR(),
  pars_obs = NULL,
  n_particles = 100
)
```

Arguments

R0_min	Minimum value of R0 in the search
R0_max	Maximum value of R0 in the search
R0_step	Step to increment R0 between min and max
first_start_date	Earliest start date as 'yyyy-mm-dd'
last_start_date	Latest start date as 'yyyy-mm-dd'
day_step	Step to increment date in days
Meff_min	Minimum value of Meff (Movement effect size) in the search
Meff_max	Maximum value of Meff (Movement effect size) in the search
Meff_step	Step to increment Meff (Movement effect size) between min and max
data	Deaths data to fit to. See example_deaths.csv and particle_filter_data()
model_params	Squire model parameters. Created from a call to one of the parameters_<type>_model functions.
Rt_func	Function for converting R0, Meff and R0_change. Function must have names arguments of R0, Meff and R0_change. Default is linear relationship on the log scale given by exp(log(R0) - Meff*(1-R0_change)).

R0_prior — Prior for R0. Default = NULL, which is a flat prior. Should be provided as a list with first argument the distribution function and the second the function arguments (excluding quantiles which are worked out based on R0_min and R0_max), e.g. `list("func" = dnorm, args = list("mean"= 3.5, "sd"= 3))`.

R0_change — Numeric vector for relative changes in R0. Default = NULL, i.e. no change in R0

date_R0_change — Calendar dates at which R0_change occurs. Defaut = NULL, i.e. no change in R0

date_contact_matrix_set_change — Calendar dates at which the contact matrices set in `model_params` change. Defaut = NULL, i.e. no change

date_ICU_bed_capacity_change — Calendar dates at which ICU bed capacity changes set in `model_params` change. Defaut = NULL, i.e. no change

date_hosp_bed_capacity_change — Calendar dates at which hospital bed capacity changes set in `model_params` change. Defaut = NULL, i.e. no change

squire_model — A squire model. Default = `explicit_SEIR()`

pars_obs — list of parameters to use for the comparison function.

n_particles — Number of particles. Positive Integer. Default = 100

Value

List of R0 and start date grid values, and normalised probabilities at each point

---

simple_model *Simple SEEIR model creation. We will use this structure to ensure that model fitting is flexible in the future as more models are added*

Description

Create a simple model

Usage

```
simple_model()
```

summary.squire_simulation *squire simulation summary*

Description

squire simulation summary

Usage

```
## S3 method for class 'squire_simulation'
summary(object, ...)
```

Arguments

object	An squire_simulation object
...	additional arguments affecting the summary produced.

trigger_projections *Project lockdowns based on triggering*

Description

Project lockdowns based on triggering

Usage

```
trigger_projections(
  out,
  trigger_metric = "deaths",
  trigger_value = 150,
  R0_lockdowns = c(0.5, 0.5, 0.5, 0.5),
  lockdown_lengths = c(28, 42, 28, 42),
  max_lockdowns = 4,
  seed = 931L
)
```

Arguments

out	Output of pmcmc or calibrate
trigger_metric	Name of model output to tigger by. Must be one accepted by format_output
trigger_value	Value of trigger_metric at which triggering occurs
R0_lockdowns	Vector of R0 values to be used for each lockdown. Default = c(0.5, 0.5, 0.5, 0.5)

`lockdown_lengths`
: Vector of lengths of each lockdown in days. `Default = c(28, 42, 28, 42)`

`max_lockdowns`
: Maximum number of lockdowns. `Default = 4`

`seed`
: RNG seed to be used. `Default = 931L`

Index

∗ **datasets**
    contact_matrices, 13
    country_specific_healthcare_capacity, 13
    elderly_pop, 20
    income_group, 27
    income_strata_healthcare_capacity, 28
    population, 45

adjusted_eigen, 3
align, 4
as.Date, *30*, *38*

beta_est, 5
beta_est_explicit, 5
beta_est_simple, 6

calibrate, 7, *68*
check_squire, 10
check_time_change, 10
compare_output, 11
conduct_replicate, 11
contact_matrices, 13
country_specific_healthcare_capacity, 13
create_master_chain, 14

default_durations, 14, *35*, *58*
default_probs, 15, *35*, *58*
deterministic_model, 15
div_pop, 16
drjacoby_mcmc, 16

elderly_pop, 20
explicit_model, *9*, 20, *20*, *44*
extract_deaths, 21
extract_hospital_occ, 21
extract_ICU_occ, 22
extract_infection_incidence, 22

format_deterministic_output, 23
format_output, 23, *48*
format_output_simple_model, 24

get_elderly_population, 25
get_healthcare_capacity, 25
get_lmic_countries, 26
get_mixing_matrix, 26
get_population, 27

income_group, 27
income_strata_healthcare_capacity, 28
init_check, 28
init_check_explicit, 29
intervention_dates_for_odin, 29

jc_prop_update, 30

list, *61*, *63*

matrix_check, 31
matrix_set, 31

parameters_explicit_SEEIR, 32
parameters_simple_SEEIR, 35
parse_country_severity, *35*, 36, *58*
particle_filter, 37
particle_filter_data, 38
plot.squire_scan, 39
plot.squire_simulation, 39
pmcmc, 40, *68*
population, 45
pos_num, 45
print.squire_simulation, 46
process_contact_matrix, 46
process_contact_matrix_scaled_age, 47
projection_plotting, 47
projections, 48, *48*

run_deterministic_comparison, 50
run_deterministic_SEIR_model, 51

run_explicit_SEEIR_model, *5*, 54
run_particle_filter, 58
run_simple_SEEIR_model, 59

sample_3d_grid_scan, 60
sample_drjacoby, 61
sample_grid_scan, 62
sample_pmcmc, 63
scan_R0_date, *62*, 64
scan_R0_date_Meff, *60, 61*, 65
simple_model, 67
stat_summary, *40*, *48*
summary.squire_simulation, 68

trigger_projections, 68