

Package: daedalus.compare (via r-universe)

May 11, 2026

Title Run Multiple Scenarios of the Daedalus Model

Version 0.0.6

Description Run multiple pandemic response scenarios using the daedalus package and access epidemiological and economic outcomes.

License MIT + file LICENSE

URL <https://github.com/jameel-institute/daedalus.compare>

BugReports <https://github.com/jameel-institute/daedalus.compare/issues>

Imports checkmate, cli, daedalus, daedalus.data, data.table, distributional, purrr, rlang, scales, stats

Suggests dplyr, ggplot2, knitr, rmarkdown, spelling, testthat (>= 3.0.0)

VignetteBuilder knitr

Remotes jameel-institute/daedalus

Config/Needs/website jameel-institute/jameelinst.rpkg.theme

Config/testthat/edition 3

Encoding UTF-8

Language en-GB

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Repository <https://jameel-institute.r-universe.dev>

Date/Publication 2026-04-30 13:11:52 UTC

RemoteUrl <https://github.com/jameel-institute/daedalus.compare>

RemoteRef HEAD

RemoteSha a4618f496d7d9afcc0d52e3555d22b29c5e1cd5c

Contents

ci	2
get_cost_data	2
get_econ_cost_data	3
get_epicurve_data	3
get_summary_data	4
make_infection_samples	5
NAMES_VECTOR_INF_PARAMS	6
run_scenarios	6

Index	8
--------------	----------

ci	<i>Calculate an interval</i>
----	------------------------------

Description

Calculate an interval

Usage

```
ci(x, level = 95)
```

Arguments

x	A numeric vector.
level	The interval level as a numeric vector with values in the range [0, 100]. Defaults to 95 for the 95% interval.

Value

A numeric vector of the same length as level.

get_cost_data	<i>Get cost data from DAEDALUS scenarios</i>
---------------	--

Description

Get cost data from DAEDALUS scenarios

Usage

```
get_cost_data(dt, disease_tags = "default", format = c("long", "wide"))
```

Arguments

dt	A <data.table> resulting from run_scenarios().
disease_tags	A character vector giving names for replicates within each scenario.
format	A string for whether the data should be returned in "long" or "wide" format. Returning a 'wide' data.frame is only recommended when there is a small number of disease_tags, typically a triplet of "low", "medium", and "high" risk scenarios.

Value

A <data.frame> summarising epidemic costs over the timeframe of the modelled epidemics.
 A <data.frame> of the costs for each model scenario.

get_econ_cost_data *Get economic cost details from an output data.table*

Description

Get economic cost details from an output data.table

Usage

```
get_econ_cost_data(dt)
```

Arguments

dt	A <data.table> resulting from run_scenarios().
----	--

Value

A <data.frame> summarising economic costs over the timeframe of the modelled epidemics, broken down into costs due to restrictions and illness-related absences.

get_epicurve_data *Get epidemiological curves from model data*

Description

Get epidemiological curves from model data

Usage

```
get_epicurve_data(dt, disease_tags = "default", format = c("long", "wide"))
```

Arguments

dt	A <data.table> resulting from run_scenarios().
disease_tags	A character vector giving names for replicates within each scenario.
format	A string for whether the data should be returned in "long" or "wide" format. Returning a 'wide' data.frame is only recommended when there is a small number of disease_tags, typically a triplet of "low", "medium", and "high" risk scenarios.

Value

A <data.frame> of individuals in each epidemiological compartments over the timeframe of the modelled epidemics.

get_summary_data	<i>Get summary data from DAEDALUS scenarios</i>
------------------	---

Description

Get summary data from DAEDALUS scenarios

Usage

```
get_summary_data(dt, disease_tags, format = c("long", "wide"), ...)
```

Arguments

dt	A <data.table> resulting from run_scenarios().
disease_tags	A character vector giving names for replicates within each scenario.
format	A string for whether the data should be returned in "long" or "wide" format. Returning a 'wide' data.frame is only recommended when there is a small number of disease_tags, typically a triplet of "low", "medium", and "high" risk scenarios.
...	Additional arguments passed to daedalus::get_epidemic_summary() . If passing groups, only format = "long" is supported.

Value

A <data.frame> summarising epidemiological outcomes: cumulative infections, deaths, and hospitalisations over the timeframe of the modelled epidemics.

```
make_infection_samples
```

Generate multiple <daedalus_infection>s from parameter distributions

Description

Generate multiple <daedalus_infection>s from parameter distributions

Usage

```
make_infection_samples(
  name,
  param_distributions,
  param_ranges = NULL,
  samples = 100
)
```

Arguments

name	An infection name from among <code>daedalus.data::epidemic_names</code> .
param_distributions	A named list of <distribution> class objects provided by distributional , with names corresponding to <code>daedalus.data::infection_parameter_names</code> . These are used to generate samples drawn from each distribution for the corresponding infection parameters. Arguments which vary by age are supported, with the drawn and scaled value treated as the median of the profile vector. See Examples .
param_ranges	An optional named list of two-element vectors, giving the ranges to which samples drawn using <code>param_distributions</code> should be rescaled. This allows easy rescaling of values drawn from bounded distributions (e.g. the Beta distribution). If passed, the list must have names corresponding to <code>param_distribution_names</code> . When empty, the generated values are used without scaling. See Examples .
samples	The number of samples to generate.

Value

A list of <daedalus_infection> objects.

Examples

```
# make 10 infection objects varying in R0, with a skewed distribution
# scaled between 1.0 and 2.0
make_infection_samples(
  "influenza_2009",
  samples = 3,
```

```

list(
  r0 = distributional::dist_beta(2, 5)
),
list(
  r0 = c(0.1, 0.2)
)
)

```

NAMES_VECTOR_INF_PARAMS

Names of age-varying infection parameters

Description

Names of age-varying infection parameters

Usage

NAMES_VECTOR_INF_PARAMS

Format

An object of class character of length 4.

run_scenarios

Run multiple DAEDALUS scenarios

Description

Run multiple DAEDALUS scenarios

Usage

```

run_scenarios(
  country,
  infection,
  response_strategy = NULL,
  vaccination_strategy = NULL,
  time_end = 100,
  initial_state_manual = NULL
)

```

Arguments

country	A country or territory object of class <code><daedalus_country></code> , or a country or territory name from those included in the package; see daedalus.data::country_names , or a country ISO2 or ISO3 code; see daedalus.data::country_codes_iso2c and daedalus.data::country_codes_iso3c . Country-specific data such as the community and workplace contacts, the demography, and the distribution of the workforce into economic sectors is automatically accessed from package data for the relevant country name if it is passed as a string. To override package defaults for country characteristics, pass a <code><daedalus_country></code> object instead. See daedalus_country() for more.
infection	A list of <code><daedalus_infection></code> objects. Must have a minimum length of 2.
response_strategy	A time-limited response strategy specified as a single <code><daedalus_npi></code> created using daedalus::daedalus_timed_npi() , or a list of such objects, or NULL for no response. Lists may include NULL, which is useful when comparing response scenarios against the no-response counterfactual. If a list is passed, all elements must be named, or the list may have no names, in which case synthetic names will be assigned and a message printed to screen. Lists with some elements named and some unnamed are not accepted.
vaccination_strategy	A vaccination strategy specified as a single <code><daedalus_vaccination></code> created using daedalus::daedalus_vaccination() , or a list of such objects, or NULL for no response. Lists may include NULL, which is useful when comparing vaccination scenarios against the no-vaccination counterfactual. If a list is passed, all elements must be named, or the list may have no names, in which case synthetic names will be assigned and a message printed to screen. Lists with some elements named and some unnamed are not accepted.
time_end	A vector of integer-ish numbers giving the durations over which to run scenarios. Each scenario is run for each <code>time_end</code> . The intention is to be able to run response scenarios for different durations, to be able to generate a time-series of how different costs accumulate.
initial_state_manual	An optional named list with the names <code>p_infectious</code> , <code>p_asymptomatic</code> , and <code>p_immune</code> . <code>p_infectious</code> and <code>p_asymptomatic</code> give the proportion of infectious and symptomatic individuals in each age group and economic sector. Defaults to $1e-6$ and 0.0 respectively. <code>p_immune</code> may be a single number in the range $0.0 \leq p_immune \leq 1.0$ or a 4-element vector in that range (the number of age groups in the model), for the proportion of individuals in the population or in each age group that have some pre-existing immunity to infection (reduced susceptibility). See Details for more.

Value

A `<data.table>`, with data held in list-columns.

Index

* constants

NAMES_VECTOR_INF_PARAMS, 6

* output_helpers

get_cost_data, 2

get_econ_cost_data, 3

get_epicurve_data, 3

get_summary_data, 4

ci, 2

daedalus.data::country_codes_iso2c, 7

daedalus.data::country_codes_iso3c, 7

daedalus.data::country_names, 7

daedalus::daedalus_timed_npi(), 7

daedalus::daedalus_vaccination(), 7

daedalus::get_epidemic_summary(), 4

daedalus_country(), 7

get_cost_data, 2

get_econ_cost_data, 3

get_epicurve_data, 3

get_summary_data, 4

make_infection_samples, 5

NAMES_VECTOR_INF_PARAMS, 6

run_scenarios, 6