Package ‘swash’

June 3, 2025

Type Package

Title Swash-Backwash Model for the Single Epidemic Wave

Version 1.2.1

Date 2025-05-29

Author Thomas Wieland [aut, cre] (ORCID:
<https://orcid.org/0000-0001-5168-9846>)

Maintainer Thomas Wieland <geowieland@googlemail.com>

Depends R (>= 3.5.0), lubridate

Description The Swash-Backwash Model for the Single Epidemic Wave was developed by Cliff and Haggett (2006) <doi:10.1007/s10109-006-0027-8> to model the velocity of spread of infectious diseases across space. This package enables the calculation of the Swash-Backwash Model for user-supplied panel data on regional infections. The package also provides additional functions for bootstrap confidence intervals, country comparison, visualization of results, and data management.

License GPL (>= 2)

Imports methods

NeedsCompilation no

Repository CRAN

Date/Publication 2025-06-03 07:10:02 UTC

Contents

`swash-package` *Implementation of the Swash-Backwash Model for the Single Epidemic Wave and additional functions in R*

Description

Swash-Backwash Model for the single epidemic wave (Cliff and Haggett 2006) with additional functions for bootstrap confidence intervals and data management

Details

The Swash-Backwash Model for the Single Epidemic Wave is the spatial equivalent of the classic epidemiological SIR (Susceptible-Infected-Recovered) model. It was developed by Cliff and Haggett (2006) to model the velocity of spread of infectious diseases across space. Current applications can be found, for example, in Smallman-Raynor et al. (2022a,b). This package enables the calculation of the Swash-Backwash Model for user-supplied panel data on regional infections. The core of this is the `swash()` function, which calculates the model and creates a model object of the `sbm` class defined in this package. This class can be used to visualize results (`summary()`, `plot()`) and calculate bootstrap confidence intervals for the model estimates (`confint(sbm)`). The package also contains additional helper functions.

Author(s)

Thomas Wieland

References

Swash-Backwash Model:

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022a) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022b) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

Basics of epidemiological modeling:

Bonifazi G et al. (2021) A simplified estimate of the effective reproduction number Rt using its relation with the doubling time and application to Italian COVID-19 data. *The European Physical Journal Plus* 136, 386. doi:10.1140/epjp/s13360021013396

Li, MY (2018) An Introduction to Mathematical Modeling of Infectious Diseases. doi:10.1007/9783319721224

Nishiura H, Chowell G (2009) The effective reproduction number as a prelude to statistical estimation of time-dependent epidemic trends. In Chowell G, Hyman JM, Bettencourt LMA (eds.) Mathematical and statistical estimation approaches in epidemiology, 103–121. doi:10.1007/97890-48123131_5

Pell B, Kuang Y, Viboud C, Chowell G (2018) Using phenomenological models for forecasting the 2015 Ebola challenge. Epidemics 22, 62–70. doi:10.1016/j.epidem.2016.11.002

Spatio-temporal analysis and modeling of infectious diseases:

Bourdin S, Jeanne L, Nadou F, Noiret G (2021) Does lockdown work? A spatial analysis of the spread and concentration of Covid-19 in Italy. Regional Studies, 55, 1182–1193. doi:10.1080/00343404.2021.1887471

Chowell G, Viboud C, Hyman JM, Simonsen L (2015) The Western Africa ebola virus disease epidemic exhibits both global exponential and local polynomial growth rates. *PLOS Currents Outbreaks*, ecurrents.outbreaks.8b55f4bad99ac5c5db3663e916803261. doi:10.1371/currents.outbreaks.8b55f4bad99ac5c5db36

Viboud C, Bjørnstad ON, Smith DL, Simonsen L, Miller MA, Grenfell BT (2006) Synchrony, Waves, and Spatial Hierarchies in the Spread of Influenza. *Science* 312,447-451. doi:10.1126/science.1125237

Wieland T (2020) Flatten the Curve! Modeling SARS-CoV-2/COVID-19 Growth in Germany at the County Level. *REGION* 7(2), 43–83. doi:10.18335/region.v7i2.324

Wieland T (2020) A phenomenological approach to assessing the effectiveness of COVID-19 related nonpharmaceutical interventions in Germany. *Safety Science* 131, 104924. doi:10.1016/j.ssci.2020.104924

Panel data:

Greene, WH (2012) *Econometric Analysis*. Ch. 11.

Wooldridge, JM (2012) *Introductory Econometrics. A Modern Approach*. Ch. 13.

Bootstrapping und bootstrap confidence intervals:

Efron B, Tibshirani RJ (1993) An Introduction to the Bootstrap.

Ramachandran KM, Tsokos CP (2021) Mathematical Statistics with Applications in R (Third Edition). Ch. 13.3.1 (Bootstrap confidence intervals). doi:10.1016/B9780128178157.000130

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

CH_covidwave1 <-
  swash (
    data = COVID19Cases_geoRegion,
    col_cases = "entries",
    col_date = "datum",
    col_region = "geoRegion"
    )
# Swash-Backwash Model for Swiss COVID19 cases
# Spatial aggregate: NUTS 3 (cantons)

summary(CH_covidwave1)
# Summary of Swash-Backwash Model

plot(CH_covidwave1)
# Plot of Swash-Backwash Model edges and total epidemic curve
```

---

as_balanced — *Correction of Non-balanced Panel Dataset with Regional Infection Data*

Description

This function corrects non-balanced input panel data by replacing missing entries with a user-given constant (e.g., 0).

Usage

```
as_balanced(
  data,
  col_cases,
  col_date,
  col_region,
  fill_missing = 0
  )
```

Arguments

`data`	`data.frame` with regional infection data
`col_cases`	Column containing the cases (numeric)
`col_date`	Column containing the time points (e.g., days)
`col_region`	Column containing the unique identifier of the regions (e.g., name, NUTS 3 code)
`fill_missing`	Constant to fill missing values (default and recommended: 0)

Details

The Swash-Backwash Model for the Single Epidemic Wave does not necessarily require balanced panel data in order for the calculations to be carried out. However, for a correct estimation it is implicitly assumed that the input data is balanced. The function corrects non-balanced panel data. It is executed automatically whithin the `swash()` function (when using the function `is_balanced()`), but can also be used separately.

Value

`data`	Corrected input dataset (`data.frame`)

Author(s)

Thomas Wieland

References

Swash-Backwash Model:

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

Panel data:

Greene, WH (2012) *Econometric Analysis*. Ch. 11.

Wooldridge, JM (2012) *Introductory Econometrics. A Modern Approach*. Ch. 13.

See Also

`is_balanced`

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

COVID19Cases_geoRegion_balanced <-
  is_balanced(
  data = COVID19Cases_geoRegion,
  col_cases = "entries",
  col_date = "datum",
  col_region = "geoRegion"
)
# Test whether "COVID19Cases_geoRegion" is balanced panel data

COVID19Cases_geoRegion_balanced$data_balanced
# Balanced? TRUE or FALSE

if (COVID19Cases_geoRegion_balanced$data_balanced == FALSE) {
  COVID19Cases_geoRegion <-
    as_balanced(
    COVID19Cases_geoRegion,
    col_cases = "entries",
    col_date = "datum",
    col_region = "geoRegion"
  )
}
# Correction of dataset "COVID19Cases_geoRegion"
# not necessary as parameter balance of is_balanced is set TRUE by default
```

---

compare_countries *Two-country Comparison of Swash-Backwash Model Parameters*

Description

This function enables bootstrap estimates for the mean difference of Swash-Backwash Model parameters of two countries to be compared.

Usage

```
compare_countries(
  sbm1,
  sbm2,
```

```
country_names = c("Country 1", "Country 2"),
indicator = "R_0A",
iterations = 20,
samples_ratio = 0.8,
alpha = 0.05,
replace = TRUE
)
```

Arguments

sbm1	A sbm object for country 1
sbm2	A sbm object for country 2
country_names	list with user-given country names (two entries)
indicator	character, indicator to be analyzed ("S_A", "I_A", "R_A", "t_LE", "t_LE", or "R_0A" (default and recommended: "R_0A"))
iterations	Number of iterations for resampling (default: 100)
samples_ratio	Proportion of regions included in each sample (default: 0.8)
alpha	Significance level α for the confidence intervals (default: 0.05)
replace	Resampling with replacement (TRUE or FALSE, default: TRUE = bootstrap resampling)

Details

The combination of the Swash-Backwash Model and bootstrap resampling allows the estimation of mean differences of a user-specified model parameter (e.g., spatial reproduction number R_{OA}) between two countries. This makes it possible to check whether the spatial spread velocity of a communicable disease is significantly different in one country than in another country. Since the initial data in the Swash-Backwash Model should be balanced, *entity-based bootstrap sampling* is carried out in the `compare_countries()` function. This means that not, for example, 80% of all observations are included in each sample at a sample ratio equal to p = 0.8, but rather all observations for 80% of the regions. For both countries, B bootstrap samples (default: 100) are drawn for which the Swash-Backwash Model is calculated. Based on the distribution of indicators, confidence intervals are calculated at the user-specified significance level α. The `compare_countries()` function calculates the differences of the user's desired indicator between the two samples, D, and also calculates α confidence intervals for this.

Value

object of class countries, see countries-class

Author(s)

Thomas Wieland

References

Swash-Backwash Model:

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

Bootstrapping und bootstrap confidence intervals:

Efron B, Tibshirani RJ (1993) An Introduction to the Bootstrap.

Ramachandran KM, Tsokos CP (2021) Mathematical Statistics with Applications in R (Third Edition). Ch. 13.3.1 (Bootstrap confidence intervals). doi:10.1016/B9780128178157.000130

See Also

swash, countries-class

Examples

```
data(COVID19Cases_geoRegion)
# Get Swiss COVID19 cases at NUTS 3 level

data(Oesterreich_Faelle)
# Get Austrian COVID19 cases at NUTS 3 level
# (first wave, same final date as in Swiss data: 2020-05-31)

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

CH_covidwave1 <-
  swash (
    data = COVID19Cases_geoRegion,
    col_cases = "entries",
    col_date = "datum",
    col_region = "geoRegion"
    )
# Swash-Backwash Model for Swiss COVID19 cases
# Spatial aggregate: NUTS 3 (cantons)

AT_covidwave1 <-
  swash (
    data = Oesterreich_Faelle,
```

```
    col_cases = "Faelle",
    col_date = "Datum",
    col_region = "NUTS3"
  )
# Swash-Backwash Model for Austrian COVID19 cases
# Spatial aggregate: NUTS 3

AT_vs_CH <-
  compare_countries(
    CH_covidwave1,
    AT_covidwave1,
    country_names = c("Switzerland", "Austria"))
# Country comparison Switzerland vs. Austria
# default config: 20 iterations, alpha = 0.05, sample ratio = 80%,
# indicator: R_0A

summary(AT_vs_CH)
# Summary of country comparison

plot(AT_vs_CH)
# Plot of country comparison
```

---

confint-methods *Methods for Function* `confint`

Description

Methods for function `confint`

Methods

`signature(object = "sbm", iterations = 100, samples_ratio = 0.8, alpha = 0.05, replace = TRUE)`
: Creates bootstrap confidence intervals for `sbm` objects. The argument `iterations` indicates the number of bootstrap samples which are drawn. Since the initial data in the Swash-Backwash Model should be balanced, *entity-based bootstrap sampling* is carried out. This means that not, for example, 80% of all observations are included in each sample at a sample ratio equal to $p = 0.8$ (`samples_ratio = 0.8`), but rather all observations for 80% of the regions. The significance level for the confidence intervals α is set by the argument `alpha` (default: 0.05, which corresponds to a 95% confidence level).

Author(s)

Thomas Wieland

References

Swash-Backwash Model:

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

Bootstrapping und bootstrap confidence intervals:

Efron B, Tibshirani RJ (1993) An Introduction to the Bootstrap.

Ramachandran KM, Tsokos CP (2021) Mathematical Statistics with Applications in R (Third Edition). Ch. 13.3.1 (Bootstrap confidence intervals). doi:10.1016/B9780128178157.000130

See Also

sbm_ci-class

---

countries-class   *Class* "countries"

Description

The class "countries" contains the results of a two-country comparison analysis using the Swash-Backwash Model, including two "sbm_ci" classes for each country. Use summary(countries) and plot(countries) for results summary and plotting, respectively.

Objects from the Class

Objects can be created by calls of the form new("countries", ...). Objects can be created by the function compare_countries(sbm1, sbm2).

Slots

sbm_ci1: Object of class "sbm_ci" Results of "confint(sbm1)" for country 1

sbm_ci2: Object of class "sbm_ci" Results of "confint(sbm1)" for country 2

D: Object of class "numeric" Results: Difference D between the samples with respect to the chosen indicator

D_ci: Object of class "numeric" Results: α confidence intervals of D

config: Object of class "list" Configuration details for bootstrap sampling

country_names: Object of class "character" User-stated country names

indicator: Object of class "character" User-stated indicator to be tested

Methods

plot `signature(x = "countries")`: Plots the results of a two-country comparison with the Swash-Backwash Model

show `signature(object = "countries")`: Prints an `countries` object; use `summary(sbm_ci)` for results

summary `signature(object = "countries")`: Prints a summary of a `countries` object (results of the two-country comparison)

Author(s)

Thomas Wieland

Examples

```
showClass("countries")
```

`COVID19Cases_geoRegion` *Switzerland Daily COVID-19 cases by region*

Description

A dataset containing COVID-19 cases by region (NUTS 3 = cantons) and time periods (days) for Switzerland (Source: Federal Office of Public Health FOPH).

Usage

```
data(COVID19Cases_geoRegion)
```

Format

A data.frame with multiple columns:

geoRegion (character) Region for which the data was collected.

datum (Date) Date of record.

entries (integer) Number of reported cases on this date.

sumTotal (integer) Cumulative case numbers.

timeframe_14d (logical) Indicates whether the time period covers the last 14 days.

timeframe_all (logical) Indicates whether the time period covers all previous data.

offset_last7d (integer) Offset of the last 7 days.

sumTotal_last7d (integer) Cumulative case numbers of the last 7 days.

offset_last14d (integer) Offset of the last 14 days.

sumTotal_last14d (integer) Cumulative case numbers of the last 14 days.

offset_last28d (integer) Offset of the last 28 days.

sumTotal_last28d (integer) Cumulative case numbers of the last 28 days.

sum7d (numeric) Sum of the last 7 days.

sum14d (numeric) Sum of the last 14 days.

mean7d (numeric) Average of the last 7 days.

mean14d (numeric) Average of the last 14 days.

entries_diff_last_age (integer) Difference from the last age group.

pop (integer) Population of the region.

inz_entries (numeric) Incidence of the entries.

inzsumTotal (numeric) Incidence of cumulative cases.

inzmean7d (numeric) Incidence of the 7-day average.

inzmean14d (numeric) Incidence of the 14-day average.

inzsumTotal_last7d (numeric) Incidence of cumulative cases in the last 7 days.

inzsumTotal_last14d (numeric) Incidence of cumulative cases in the last 14 days.

inzsumTotal_last28d (numeric) Incidence of cumulative cases in the last 28 days.

inzsum7d (numeric) Incidence of the last 7 days.

inzsum14d (numeric) Incidence of the last 14 days.

sumdelta7d (numeric) Difference in sums of the last 7 days.

inzdelta7d (numeric) Difference in incidence of the last 7 days.

type (character) Type of recorded data (e.g., COVID-19 cases).

type_variant (character) Variant of the data type.

version (character) Version of the data collection.

datum_unit (character) Unit of date specification (e.g., day).

entries_letzter_stand (integer) Last known count of entries.

entries_neu_gemeldet (integer) Newly reported entries.

entries_diff_last (integer) Difference in last entries.

Details

The data is included as it was published in by the Swiss Federal Office of Public Health (Bundesamt fuer Gesundheit, BAG). Note that the reporting date equals the date of SARS-CoV-2 testing.

Source

Federal Office of Public Health FOPH (2023) COVID-19 Dashboard Source Data. https://www.covid19.admin.ch/api/data/documentation (retrieved 2023-06-28)

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

COVID19Cases_geoRegion_balanced <-
  is_balanced(
  data = COVID19Cases_geoRegion,
  col_cases = "entries",
  col_date = "datum",
  col_region = "geoRegion"
)
# Test whether "COVID19Cases_geoRegion" is balanced panel data

COVID19Cases_geoRegion_balanced$data_balanced
# Balanced? TRUE or FALSE
```

`growth` *Logistic Growth Models for Regional Infections*

Description

Estimates N logistic growth models for N regions.

Usage

```
growth(
  object,
  S_iterations = 10,
  S_start_est_method = "bisect",
  seq_by = 10,
  nls = TRUE
  )
```

Arguments

`object`	object of class `sbm`
`S_iterations`	Number of iterations for saturation value search
`S_start_est_method`	Method for saturation value search, either "bisect" or "trial_and_error"
`seq_by`	No of segments for the "trial_and_error" estimation of the saturation value
`nls`	Nonlinear estimation? `TRUE` or `FALSE`

Details

The function estimates logistic growth models for regional infections based on a sbm object. See logistic_growth for further details.

Value

list with two entries:

results: Object of class "data.frame" Results of the logistic growth models (coefficients and derivates)

logistic_growth_models:
Object of class "list" List with N entries for N growth models resp. loggrowth objects

Author(s)

Thomas Wieland

References

Wieland T (2020) Flatten the Curve! Modeling SARS-CoV-2/COVID-19 Growth in Germany at the County Level. *REGION* 7(2), 43–83. doi:10.18335/region.v7i2.324

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

CH_covidwave1 <-
  swash (
    data = COVID19Cases_geoRegion,
    col_cases = "entries",
    col_date = "datum",
    col_region = "geoRegion"
    )
# Swash-Backwash Model for Swiss COVID19 cases
# Spatial aggregate: NUTS 3 (cantons)

CH_covidwave1_growth <- growth(CH_covidwave1)
CH_covidwave1_growth
# Logistic growth models for sbm object CH_covidwave1
```

growth-methods — *Methods for Function* `growth`

Description

Methods for function `growth`

Methods

`signature(object = "sbm", S_iterations = 10, S_start_est_method = "bisect", seq_by = 10, nls = TRUE)`
: Estimation of N logistic growth models for N regions. Both OLS and NLS estimation are estimated by default (set `nls = FALSE` to skip NLS estimation). Parameters `S_iterations`, `S_start_est_method`, and `seq_by` are used to control the saturation value estimation (see `logistic_growth`).

Author(s)

Thomas Wieland

hist_ci — *Creating Histograms with Confidence Intervals*

Description

Plot of a histogram of a given vector `x` and the related confidence intervals (`lower`, `upper`).

Usage

```
hist_ci(
  x,
  alpha = 0.05,
  col_bars = "grey",
  col_ci = "red",
  ...
  )
```

Arguments

`x`	A numeric vector
`alpha`	Significance level α for 1-α*100 confidence intervals
`col_bars`	Color of bars in histogram
`col_ci`	Color of lines for confidence interval
`...`	Additional arguments passed to `barplot()`

Details

Helper function for `plot(sbm_ci)`, but may be used separately.

Value

Histogram plot, no returned value

Author(s)

Thomas Wieland

Examples

```
numeric_vector <- c(1,9,5,6,3,10,20,6,9,14,3,5,8,6,11)
# any numeric vector

hist_ci(numeric_vector)
```

`is_balanced` *Test whether Panel Dataset with Regional Infection Data is Balanced*

Description

The function tests whether the input panel data with regional infections is balanced.

Usage

```
is_balanced(
  data,
  col_cases,
  col_date,
  col_region,
  as_balanced = TRUE,
  fill_missing = 0
  )
```

Arguments

`data`	`data.frame` with regional infection data
`col_cases`	Column containing the cases (numeric)
`col_date`	Column containing the time points (e.g., days)
`col_region`	Column containing the unique identifier of the regions (e.g., name, NUTS 3 code)
`as_balanced`	Boolean argument which indicates whether non-balanced panel data shall be balanced (default: TRUE)
`fill_missing`	Constant to fill missing values (default and recommended: 0)

Details

The Swash-Backwash Model for the Single Epidemic Wave does not necessarily require balanced panel data in order for the calculations to be carried out. However, for a correct estimation it is implicitly assumed that the input data is balanced. The function tests whether the panel data is balanced. It is executed automatically whithin the `swash()` function (using automatic correction with `as_balanced = TRUE`), but can also be used separately.

Value

List with two entries:

`data_balanced`	Result of test (`TRUE` or `FALSE`)
`data`	Input dataset (`data.frame`)

Author(s)

Thomas Wieland

References

Swash-Backwash Model:

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

Panel data:

Greene, WH (2012) *Econometric Analysis*. Ch. 11.

Wooldridge, JM (2012) *Introductory Econometrics. A Modern Approach*. Ch. 13.

See Also

`as_balanced`

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
```

```
# Extract first COVID-19 wave

COVID19Cases_geoRegion_balanced <-
  is_balanced(
  data = COVID19Cases_geoRegion,
  col_cases = "entries",
  col_date = "datum",
  col_region = "geoRegion"
)
# Test whether "COVID19Cases_geoRegion" is balanced panel data

COVID19Cases_geoRegion_balanced$data_balanced
# Balanced? TRUE or FALSE

if (COVID19Cases_geoRegion_balanced$data_balanced == FALSE) {
  COVID19Cases_geoRegion <-
    as_balanced(
    COVID19Cases_geoRegion,
    col_cases = "entries",
    col_date = "datum",
    col_region = "geoRegion"
  )
}
# Correction of dataset "COVID19Cases_geoRegion"
# not necessary as parameter balance of is_balanced is set TRUE by default
```

`loggrowth-class` *Class* `"loggrowth"`

Description

The class `"loggrowth"` contains the results of the logistic_growth() function. Use `summary(sbm)` and `plot(sbm)` for results summary and plotting, respectively.

Objects from the Class

Objects can be created by the function `logistic_growth`.

Slots

`LinModel`**:** Object of class `list` Results of the OLS helper model

`GrowthModel_OLS`**:** Object of class `list` Results of the OLS fit (predicted, parameters, first derivative)

`GrowthModel_NLS`**:** Object of class `list` Results of the NLS fit (predicted, parameters, first derivative)

`t`**:** Object of class `numeric` Input time points data

`y`**:** Object of class `numeric` Input infections data

`config`**:** Object of class `list` Model fit configurations

Methods

plot `signature(x = "loggrowth")`: Plots the results of the logistic growth model (observed, predicted, first derivative)

summary `signature(object = "loggrowth")`: Prints a summary of `loggrowth` objects

Author(s)

Thomas Wieland

References

Wieland T (2020) Flatten the Curve! Modeling SARS-CoV-2/COVID-19 Growth in Germany at the County Level. *REGION* 7(2), 43–83. doi:10.18335/region.v7i2.324

Examples

```
showClass("loggrowth")
```

`logistic_growth` *Logistic Growth Model for Epidemic Data*

Description

Estimation of logistic growth models from cumulative infections data, linearized OLS and/or NLS

Usage

```
logistic_growth(
  y,
  t,
  S = NULL,
  S_start = NULL,
  S_end = NULL,
  S_iterations = 10,
  S_start_est_method = "bisect",
  seq_by = 10,
  nls = TRUE
  )
```

Arguments

`y`	`numeric vector` with cumulative infections data over time
`t`	`vector` of class `numeric` or `Date` with time points or dates
`S`	Saturation value for the model
`S_start`	Start value of the saturation value for estimation
`S_end`	End value of the saturation value for estimation

`S_iterations`	Number of iterations for saturation value search
`S_start_est_method`	Method for saturation value search, either "bisect" or "trial_and_error"
`seq_by`	No of segments for the "trial_and_error" estimation of the saturation value
`nls`	Nonlinear estimation? `TRUE` or `FALSE`

Details

This function allows the estimation of a logistic growth model. The user must specify the dependent variable (cumulative infections) and the time variable (time counter or date values). The estimation is performed using a linearized model as an OLS estimator and as an NLS estimator. For the former, the saturation value can either be specified by the user or found using a search algorithm. The parameters from the OLS fit are used as starting values for the NLS estimation.

Value

object of class `loggrowth-class`

Author(s)

Thomas Wieland

References

Wieland T (2020) Flatten the Curve! Modeling SARS-CoV-2/COVID-19 Growth in Germany at the County Level. *REGION* 7(2), 43–83. doi:10.18335/region.v7i2.324

See Also

`loggrowth-class`

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

COVID19Cases_BS <-
  COVID19Cases_geoRegion[(COVID19Cases_geoRegion$geoRegion == "ZH")
                         & (COVID19Cases_geoRegion$sumTotal > 0),]
# COVID cases for Zurich

loggrowth_BS <- logistic_growth (
  y = as.numeric(COVID19Cases_BS$sumTotal),
```

```
  t = COVID19Cases_BS$datum,
  S = 5557,
  S_start = NULL,
  S_end = NULL,
  S_iterations = 10,
  S_start_est_method = "bisect",
  seq_by = 10,
  nls = TRUE
)
# Logistic growth model with stated saturation value

summary(loggrowth_BS)
# Summary of logistic growth model

plot(loggrowth_BS)
# Plot of logistic growth model
```

`Oesterreich_Faelle` *Austria Daily COVID-19 cases by region 2020-02-26 to 2020-05-31*

Description

A dataset containing COVID-19 cases by region (NUTS 3) and time periods (days) for Austria (Source: BMSGPK).

Usage

```
data(Oesterreich_Faelle)
```

Format

A data.frame with multiple columns:

NUTS3 (character) Region for which the data was collected.

Datum (Date) Date of record.

Faelle (integer) Number of reported cases on this date.

Details

The original data was originally published by BMSGPK at a smaller spatial scale level (political districts, "Politische Bezirke"). The data was linked to a corresponding shapefile from Statistik Austria (2022), joined to the NUTS3 level via a spatial join, and summed over the Austrian NUTS3 regions. The spatial join is based on polygon centroids of the political districts level; in cases where the centroid was outside the polygon, it was placed inside the polygon manually.

Source

BMSGPK, Oesterreichisches COVID-19 Open Data Informationsportal (2022) COVID-19: Zeitliche Darstellung von Daten zu Covid19-Faellen je Bezirk. https://www.data.gv.at/katalog/dataset/4b71eb3d-7d55-4967-b80d-91a3f220b60c (retrieved 2022-06-23)

Statistik Austria (2022) Politische Bezirke. https://www.data.gv.at/katalog/dataset/stat_gliederung-osterreichs-in-politische-bezirke131e2 (retrieved 2022-06-27)

Wieland T (2022) C19dNUTS: Dataset of Regional COVID-19 Deaths per 100,000 Pop (NUTS). R package v1.0.1. doi:10.32614/CRAN.package.C19dNUTS

Examples

```
data(Oesterreich_Faelle)
# Get Austrian COVID19 cases at NUTS 3 level
# (first wave, same final date as in Swiss data: 2020-05-31)

AT_covidwave1 <-
  swash (
    data = Oesterreich_Faelle,
    col_cases = "Faelle",
    col_date = "Datum",
    col_region = "NUTS3"
  )
# Swash-Backwash Model for Austrian COVID19 cases
# Spatial aggregate: NUTS 3

summary(AT_covidwave1)
# Summary of model results
```

`plot-methods` *Methods for Function* `plot`

Description

Methods for function `plot`

Methods

`signature(x = "sbm")` Plots the results of the Swash-Backwash Model; two plots: edges over time, total infections per time unit

`signature(x = "sbm_ci")` Plots the results of bootstrap confidence intervals for the Swash-Backwash Model; one figure with six plots: S_A, I_A, R_A, t_{FE}, t_{LE}, and R_{0A}

`signature(x = "loggrowth")` Plots the results of the logistic growth model (observed, predicted, first derivative)

Author(s)

Thomas Wieland

plot_regions *Plots of Regional Infections Over Time*

Description

Plots regional infection curves in N plots for N regions.

Usage

```
plot_regions(
  object,
  col = "red",
  scale = FALSE,
  normalize_by_col = NULL,
  normalize_factor = 1
  )
```

Arguments

`object`	object of class `sbm`
`col`	Color of line plot
`scale`	Set y axis of the plots uniformly based on the maximum value across all regions? (boolean, default: `FALSE`)
`normalize_by_col`	Normalize infection numbers by stating a column in the input data frame (e.g., regional population)
`normalize_factor`	Multiply density/incidence with a factor (say, 100,000 inhabitants)

Details

Plots regional infection curves in N plots for N regions, with the number of columns equals 4 and the number of rows is calculated based the size of N.

Value

Plot, only no returned value

Author(s)

Thomas Wieland

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

CH_covidwave1 <-
  swash (
    data = COVID19Cases_geoRegion,
    col_cases = "entries",
    col_date = "datum",
    col_region = "geoRegion"
    )
# Swash-Backwash Model for Swiss COVID19 cases
# Spatial aggregate: NUTS 3 (cantons)

plot_regions(CH_covidwave1)
# Plot of regional infections
```

---

plot_regions-methods *Methods for Function* `plot_regions`

Description

Methods for function `plot_regions`

Methods

`signature(object = "sbm", col = "red", scale = FALSE, normalize_by_col = NULL, normalize_factor = 1)`
: Plots regional infection curves in N plots for N regions, with the number of columns equals 4 and the number of rows is calculated based the size of N. Set the color by the argument `col`. If `scale` is `TRUE`, the y axis of the plots is set uniformly based on the maximum value across all regions. If the input data contains a column to normalize the infection numbers (such as regional population), the user may use this data to normalize the infection numbers by setting `normalize_by_col`. If this density value should be multiplied by a factor (e.g. regional infections per 100,000 inhabitants), this can be set with the argument `normalize_by_col` (default: 1).

Author(s)

Thomas Wieland

print-methods — *Methods for Function* `print`

Description

Methods for function `print`

Methods

`signature(x = "sbm")` Prints an `sbm` object; use `summary(sbm)` for results

`signature(x = "sbm_ci")` Prints an `sbm_ci` object; use `summary(sbm_ci)` for results

quantile_ci — *Computing Quantiles for a given Numeric Vector*

Description

Computes quantiles for a given vector `x` and the related confidence intervals (`lower`, `upper`).

Usage

```
quantile_ci(
  x,
  alpha = 0.05
  )
```

Arguments

`x`	A numeric vector
`alpha`	Significance level α for 1-α*100 confidence intervals

Details

Helper function for `plot(sbm_ci)`, but may be used separately.

Value

A numeric vector with lower and upper quantile

Author(s)

Thomas Wieland

Examples

```
numeric_vector <- c(1,9,5,6,3,10,20,6,9,14,3,5,8,6,11)
# any numeric vector

quantile_ci(numeric_vector)
```

R_t — *Effective Reproduction Number for Epidemic Data*

Description

Calculation of the effective reproduction number for infection/surveillance data

Usage

```
R_t(
  infections,
  GP = 4,
  correction = FALSE
  )
```

Arguments

`infections`	`numeric` `vector` with infection data
`GP`	Generation period, in days
`correction`	Correction of values equal to zero? (Recommended)

Details

The function calculates the effective reproduction number (=growth factor), R_t, of an infections time series.

Value

`list` with two entries:

`R_t:` Object of class "`numeric`" R_t values

`infections_data:`
Object of class "`data.frame`" Dataset with infections data and R_t

Author(s)

Thomas Wieland

References

Bonifazi G et al. (2021) A simplified estimate of the effective reproduction number Rt using its relation with the doubling time and application to Italian COVID-19 data. *The European Physical Journal Plus* 136, 386. doi:10.1140/epjp/s13360021013396

Wieland T (2020) A phenomenological approach to assessing the effectiveness of COVID-19 related nonpharmaceutical interventions in Germany. *Safety Science* 131, 104924. doi:10.1016/j.ssci.2020.104924

See Also

logistic_growth

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

COVID19Cases_BS <-
  COVID19Cases_geoRegion[(COVID19Cases_geoRegion$geoRegion == "ZH")
                         & (COVID19Cases_geoRegion$sumTotal > 0),]
# COVID cases for Zurich

Rt_BS <- R_t(infections = COVID19Cases_BS$entries)
# Effective reproduction number

Rt_BS
```

---

sbm-class *Class* "sbm"

Description

The class "sbm" contains the results of the Swash-Backwash Model and the related input data as well as additional information. Use summary(sbm) and plot(sbm) for results summary and plotting, respectively.

Objects from the Class

Objects can be created by the function swash.

Slots

`R_0A`**:** Object of class `"numeric"` Model result: spatial reproduction number R_{0A}

`integrals`**:** Object of class `"numeric"` Model result: integrals S_A, I_A, and R_A

`velocity`**:** Object of class `"numeric"` Model result: velocity measures t_{FE} and t_{LE}

`occ_regions`**:** Object of class `"data.frame"` Model result: Occurence at regional level

`SIR_regions`**:** Object of class `"data.frame"` Model result: Susceptible, infected and recovered regions over time

`cases_by_date`**:** Object of class `"data.frame"` Total cases by date

`cases_by_region`**:** Object of class `"data.frame"` Cumulative cases by region

`input_data`**:** Object of class `"data.frame"` Input data

`data_statistics`**:** Object of class `"numeric"` Diagnostics of input data

`col_names`**:** Object of class `"character"` Original column names in input data

Methods

confint `signature(object = "sbm")`: Creates bootstrap confidence intervals for `sbm` objects.

plot `signature(x = "sbm")`: Plots the results of the Swash-Backwash Model; two plots: edges over time, total infections per time unit

print `signature(x = "sbm")`: Prints an `sbm` object; use `summary(sbm)` for results

show `signature(object = "sbm")`: Prints an `sbm` object; use `summary(sbm)` for results

summary `signature(object = "sbm")`: Prints a summary of `sbm` objects (results of the Swash-Backwash Model)

growth `signature(object = "sbm")`: Estimates logistic growth models from `sbm` objects

Author(s)

Thomas Wieland

References

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

Wieland T (2020) Flatten the Curve! Modeling SARS-CoV-2/COVID-19 Growth in Germany at the County Level. *REGION* 7(2), 43–83. doi:10.18335/region.v7i2.324

Examples

```
showClass("sbm")
```

sbm_ci-class *Class* "sbm_ci"

Description

The class "sbm_ci" contains the results of the Swash-Backwash Model, confidence intervals for the model estimates, and the related input data as well as additional information. Use summary(sbm_ci) and plot(sbm_ci) for results summary and plotting, respectively.

Objects from the Class

Objects can be created by the function confint(sbm).

Slots

R_0A: Object of class "numeric" Model result: spatial reproduction number R_{0A}

integrals: Object of class "numeric" Model result: integrals S_A, I_A, and R_A

velocity: Object of class "numeric" Model result: velocity measures t_{FE} and t_{LE}

occ_regions: Object of class "data.frame" Model result: Occurence at regional level

cases_by_date: Object of class "data.frame" Total cases by date

cases_by_region: Object of class "data.frame" Cumulative cases by region

input_data: Object of class "data.frame" Input data

data_statistics: Object of class "numeric" Diagnostics of input data

col_names: Object of class "character" Column names in input data

integrals_ci: Object of class "list" Confidence intervals for integrals S_A, I_A, and R_A

velocity_ci: Object of class "list" Confidence intervals for velocity measures t_{FE} and t_{LE}

R_0A_ci: Object of class "numeric" Confidence intervals for spatial reproduction number R_{0A}

iterations: Object of class "data.frame" Results of bootstrap sampling iterations

ci: Object of class "numeric" Lower and upper confidence intervals based on user input

config: Object of class "list" Configuration details for bootstrap sampling

Methods

plot signature(x = "sbm_ci"): Plots the results of bootstrap confidence intervals for the Swash-Backwash Model; one figure with six plots: S_A, I_A, R_A, t_{FE}, t_{LE}, and R_{0A}

print signature(x = "sbm_ci"): Prints an sbm_ci object; use summary(sbm_ci) for results

show signature(object = "sbm_ci"): Prints an sbm_ci object; use summary(sbm_ci) for results

summary signature(object = "sbm_ci"): Prints a summary of sbm_ci objects (bootstrap confidence intervals for Swash-Backwash Model estimates)

Author(s)

Thomas Wieland

References

Swash-Backwash Model:

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

Bootstrapping und bootstrap confidence intervals:

Efron B, Tibshirani RJ (1993) An Introduction to the Bootstrap.

Ramachandran KM, Tsokos CP (2021) Mathematical Statistics with Applications in R (Third Edition). Ch. 13.3.1 (Bootstrap confidence intervals). doi:10.1016/B9780128178157.000130

Examples

```
showClass("sbm_ci")
```

---

show-methods   *Methods for Function* `show`

Description

Methods for function `show`

Methods

`signature(object = "sbm")` Prints an `sbm` object; use `summary(sbm)` for results

`signature(object = "sbm_ci")` Prints an `sbm_ci` object; use `summary(sbm_ci)` for results

`summary-methods` *Methods for Function* `summary`

Description

Methods for function `summary`

Methods

`signature(object = "sbm")` Prints a summary of `sbm` objects (results of the Swash-Backwash Model)

`signature(object = "sbm_ci")` Prints a summary of `sbm_ci` objects (bootstrap confidence intervals for Swash-Backwash Model estimates)

`signature(object = "countries")` Prints a summary of a `countries` object built with the function `compare_countries`

`swash` *Swash-Backwash Model for the Single Epidemic Wave*

Description

Analysis of regional infection/surveillance data using the Swash-Backwash Model for the single epidemic wave by Cliff and Haggett (2006)

Usage

```
swash(
  data,
  col_cases,
  col_date,
  col_region
  )
```

Arguments

`data`	`data.frame` with regional infection data
`col_cases`	Column containing the cases (numeric)
`col_date`	Column containing the time points (e.g., days)
`col_region`	Column containing the unique identifier of the regions (e.g., name, NUTS 3 code)

Details

The function performs the analysis of the input panel data using the Swash-Backwash Model. The output is an object of class "sbm". The results can be viewed using summary(sbm). The user must state panel data with daily infections.

Value

object of class sbm-class

Author(s)

Thomas Wieland

References

Cliff AD, Haggett P (2006) A swash-backwash model of the single epidemic wave. *Journal of Geographical Systems* 8(3), 227-252. doi:10.1007/s1010900600278

Smallman-Raynor MR, Cliff AD, Stickler PJ (2022) Meningococcal Meningitis and Coal Mining in Provincial England: Geographical Perspectives on a Major Epidemic, 1929–33. *Geographical Analysis* 54, 197–216. doi:10.1111/gean.12272

Smallman-Raynor MR, Cliff AD, The COVID-19 Genomics UK (COG-UK) Consortium (2022) Spatial growth rate of emerging SARS-CoV-2 lineages in England, September 2020–December 2021. *Epidemiology and Infection* 150, e145. doi:10.1017/S0950268822001285.

See Also

sbm-class

Examples

```
data(COVID19Cases_geoRegion)
# Get SWISS COVID19 cases at NUTS 3 level

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[!COVID19Cases_geoRegion$geoRegion %in% c("CH", "CHFL"),]
# Exclude CH = Switzerland total and CHFL = Switzerland and Liechtenstein total

COVID19Cases_geoRegion <-
  COVID19Cases_geoRegion[COVID19Cases_geoRegion$datum <= "2020-05-31",]
# Extract first COVID-19 wave

CH_covidwave1 <-
  swash (
    data = COVID19Cases_geoRegion,
    col_cases = "entries",
    col_date = "datum",
    col_region = "geoRegion"
    )
# Swash-Backwash Model for Swiss COVID19 cases
# Spatial aggregate: NUTS 3 (cantons)
```

```
summary(CH_covidwave1)
# Summary of Swash-Backwash Model

plot(CH_covidwave1)
# Plot of Swash-Backwash Model edges and total epidemic curve
```

Index