Package ‘i2extras’

October 30, 2020

Type     Package
Title    Functions to Work with 'incidence2' Objects
Version  0.0.2
Description Provides functions to work with 'incidence2' objects, including a
simplified interface for trend fitting and peak estimation. This package is
part of the RECON (<https://www.repidemicsconsortium.org/>) toolkit for
outbreak analysis.

URL      https://www.repidemicsconsortium.org/i2extras/
BugReports https://github.com/reconhub/i2extras/issues
Encoding UTF-8
License MIT + file LICENSE
LazyData true
RoxygenNote 7.1.1
Imports utils, stats, MASS, incidence2, trending, dplyr, tibble,
rlang, tidyr, vctrs, ellipsis, ggplot2
Suggests testthat, outbreaks, knitr, rmarkdown, covr
VignetteBuilder knitr
NeedsCompilation no
Author    Tim Taylor [aut, cre] (<https://orcid.org/0000-0002-8587-7113>),
          Thibaut Jombart [aut],
          Zhian N. Kamvar [aut] (<https://orcid.org/0000-0003-1458-7108>)
Maintainer Tim Taylor <tim.taylor@hiddenelephants.co.uk>
Repository CRAN
Date/Publication 2020-10-30 09:00:06 UTC

R topics documented:

  add_rolling_average ........................................... 2
  bootstrap .......................................................... 3
add_rolling_average

Description

add_rolling_average() adds a rolling average to an incidence2::incidence() object. If x is a grouped this will be a dplyr::rowwise() type object. If x is not grouped this will be a subclass of tibble.

Usage

add_rolling_average(x, ...)

## Default S3 method:
add_rolling_average(x, ...)

## S3 method for class 'incidence2'
add_rolling_average(x, before = 2, ...)

Arguments

x                   An incidence2::incidence object.

...                  Not currently used.

before              how many prior dates to group the current observation with. Default is 2 days.

Value

An object of class incidence2_rolling.

Note

If groups are present the average will be calculated across each grouping, therefore care is required when plotting.
Examples

```r
if (requireNamespace("outbreaks", quietly = TRUE) && requireNamespace("incidence2", quietly = TRUE)) {
  data(ebola_sim_clean, package = "outbreaks")
  dat <- ebola_sim_clean$linelist

  inci <- incidence2::incidence(dat,
      date_index = date_of_onset,
      interval = "week",
      last_date = "2014-10-05",
      groups = gender)

  ra <- add_rolling_average(inci, before = 2)
  plot(ra, color = "white")

  inci2 <- incidence2::regroup(inci)
  ra2 <- add_rolling_average(inci2, before = 2)
  plot(ra, color = "white")
}
```

bootstrap

bootstrap incidence time series

Description

This function can be used to bootstrap incidence2::incidence objects. Bootstrapping is done by sampling with replacement the original input dates. See details for more information on how this is implemented.

Usage

```r
bootstrap(x, randomise_groups = FALSE)
```

Arguments

- `x`: An incidence2::incidence object.
- `randomise_groups`: A logical indicating whether groups should be randomised as well in the resampling procedure; respective group sizes will be preserved, but this can be used to remove any group-specific temporal dynamics. If FALSE (default), data are resampled within groups.

Details

As original data are not stored in incidence2::incidence objects, the bootstrapping is achieved by multinomial sampling of date bins weighted by their relative incidence.
estimate_peak

Estimate the peak date of an incidence curve using bootstrap

Description

This function can be used to estimate the peak of an epidemic curve stored as `incidence2::incidence` object, using bootstrap. See `bootstrap()` for more information on the resampling.

Usage

```r
estimate_peak(x, n = 100, alpha = 0.05, progress = TRUE)
```

Arguments

- `x` An `incidence2::incidence` object.
- `n` The number of bootstrap datasets to be generated; defaults to 100.
- `alpha` The type 1 error chosen for the confidence interval; defaults to 0.05.
- `progress` Should a progress bar be displayed (default = TRUE)
estimate_peak

Details
Input dates are resampled with replacement to form bootstrapped datasets; the peak is reported for each, resulting in a distribution of peak times. When there are ties for peak incidence, only the first date is reported.

Note that the bootstrapping approach used for estimating the peak time makes the following assumptions:

- the total number of event is known (no uncertainty on total incidence)
- dates with no events (zero incidence) will never be in bootstrapped datasets
- the reporting is assumed to be constant over time, i.e. every case is equally likely to be reported

Value
A tibble with the following columns:

- observed_date: the date of peak incidence of the original dataset.
- observed_count: the peak incidence of the original dataset.
- estimated: the mean peak time of the bootstrap datasets.
- lower_ci/upper_ci: the confidence interval based on bootstrap datasets.
- peaks: a nested tibble containing the the peak times of the bootstrapped datasets.

Author(s)
Thibaut Jombart <thibautjombart@gmail.com>, with inputs on caveats from Michael Höhle.

See Also
bootstrap() for the bootstrapping underlying this approach and find_peak() to find the peak in a single incidence2::incidence object.

Examples
if (requireNamespace("outbreaks", quietly = TRUE) &&
    requireNamespace("incidence2", quietly = TRUE)) {

  withAutoprint( {
    # load data and create incidence
    data(fluH7N9_china_2013, package = "outbreaks")
    i <- incidence2::incidence(fluH7N9_china_2013, date_index = date_of_onset)
    i

    # one simple bootstrap
    x <- bootstrap(i)
    x

    # find 95% CI for peak time using bootstrap
    peak_data <- estimate_peak(i)
    peak_data
    summary(peak_data$peaks)
  })
}
find_peak

Find the peak date of an incidence curve

Description
This function can be used to find the peak of an epidemic curve stored as an incidence2::incidence object.

Usage
find_peak(x, regroup = TRUE)

Arguments
- x: An incidence2::incidence object.
- regroup: If TRUE (default), any groups will be regrouped before finding a peak. If FALSE, separate peaks will be found for each group.

Value
The date of the (first) highest incidence in the data.

Author(s)
Thibaut Jombart <thibautjombart@gmail.com>, Zhian N. Kamvar <zkamvar@gmail.com>

See Also
estimate_peak() for bootstrap estimates of the peak time

Examples
if (requireNamespace("outbreaks", quietly = TRUE) &
   requireNamespace("incidence2", quietly = TRUE)) {

  withAutoprint( {
    # load data and create incidence
    data(fluH7N9_china_2013, package = "outbreaks")
    i <- incidence2::incidence(fluH7N9_china_2013, date_index = date_of_onset)
    i
    find_peak(i)
  })
}
fit_curve

Fit an epi curve

Description
Fit an epi curve

Usage
fit_curve(dat, model, ...)

## Default S3 method:
fit_curve(dat, model, ...)

## S3 method for class 'incidence2'
fit_curve(dat, model = c("poisson", "negbin"), alpha = 0.05, ...)

Arguments
dat An incidence2::incidence object.
model The regression model to fit (can be "poisson" or "negbin").
... Additional arguments to pass to stats::glm() for model = "poisson" or MASS::glm.nb() for model = "negbin".
alpha Value of alpha used to calculate confidence intervals; defaults to 0.05 which corresponds to a 95% confidence interval.

Value
An object of class incidence2_fit.

growth_rate Calculate growth/decay rate

description Calculate growth/decay rate

Usage
growth_rate(x, ...)

## Default S3 method:
growth_rate(x, ...)

## S3 method for class 'incidence2_fit'
growth_rate(x, alpha = 0.05, growth_decay_time = TRUE, ...)
Arguments

x The output of function `fit_curve.incidence2()`.

... Not currently used.

alpha Value of alpha used to calculate confidence intervals; defaults to 0.05 which corresponds to a 95% confidence interval.

growth_decay_time Should a doubling/halving time and corresponding confidence intervals be added to the output. Default TRUE.

---

is_okerr Error handling for incidence2_fit objects

Description

These functions are used to filter successful model fits from those that errored or gave warnings.

Usage

```r
is_ok(x, ...)
```

## Default S3 method:
```r
is_ok(x, ...)
```

## S3 method for class `incidence2_fit`
```r
is_ok(x, include_warnings = FALSE, ...)
```

```r
is_error(x, ...)
```

## Default S3 method:
```r
is_error(x, ...)
```

## S3 method for class `incidence2_fit`
```r
is_error(x, ...)
```

```r
is_warning(x, ...)
```

## Default S3 method:
```r
is_warning(x, ...)
```

## S3 method for class `incidence2_fit`
```r
is_warning(x, ...)
```
Arguments

x  The output of function fit_curve.incidence2().

include_warnings  Include results in output that triggered warnings but not errors. Defaults to FALSE.

Details

The following accessors are available:

- `is_ok()`: returns rows from an incidence2_fit object that did not error (and optionally produce a warning).
- `is_error()`: returns rows from an incidence2_fit object that errored.
- `is_warning()`: returns rows from an incidence2_fit object that produced warnings.

plot.incidence2_fit  Plot a fitted epicurve

Description

Plot a fitted epicurve

Usage

```r
## S3 method for class 'incidence2_fit'
plot(x, include_warnings = FALSE, ci = TRUE, pi = FALSE, ...)
```

Arguments

- `x`  An incidence2_fit object created by `fit()`.
- `include_warnings`  Include results in plot that triggered warnings but not errors. Defaults to FALSE.
- `ci`  Plot confidence intervals (defaults to TRUE).
- `pi`  Plot prediction intervals (defaults to FALSE).
- `...`  Additional arguments to be passed to `incidence2::plot.incidence2()` or `incidence2::facet_plot()`.

Value

An incidence plot with the addition of a fitted curve. This will be facetted if the object is grouped.
plot.incidence2_rolling

*Plot a rolling average incidence object*

Description

Plot a rolling average incidence object

Usage

```r
## S3 method for class 'incidence2_rolling'
plot(x, ...)
```

Arguments

- `x` An `incidence2_ra` object created by `add_rolling_average()`.
- `...` Additional arguments to be passed to `incidence2::plot.incidence2()` or `incidence2::facet_plot()`.

Value

An incidence plot with the addition of a rolling average. This will be faceted if the object is grouped.
Index

add_rolling_average. 2
add_rolling_average(). 2, 10

bootstrap. 3
bootstrap(). 4, 5

dplyr::rowwise(). 2

estimate_peak. 4
estimate_peak(). 6

find_peak. 6
find_peak(). 4, 5
fit(). 9
fit_curve. 7
fit_curve.incidence2(). 8, 9

growth_rate. 7

incidence2::facet_plot(). 9, 10
incidence2::incidence. 2–7
incidence2::incidence(). 2
incidence2::plot.incidence2(). 9, 10
incidence2_fit. 9
incidence2_fit(fit_curve). 7
is_err.incidence2_fit(is_okerr). 8
is_error(is_okerr). 8
is_ok(is_okerr). 8
is_okerr. 8
is_warning(is_okerr). 8

MASS::glm.nb(). 7

plot.incidence2_fit. 9
plot.incidence2_rolling. 10
stats::glm(). 7