Package 'grattanInflators'

December 1, 2023

Title Inflators for Australian Policy Analysis

Version 0.5.0

Description Using Australian Bureau of Statistics indices, provides functions that convert historical, nominal statistics to real, contemporary values without worrying about date input quality, performance, or the ABS catalogue.

License GPL-2

Encoding UTF-8

Depends R (>= 4.0.0)

Imports data.table, fy, hutils, tools, utils

RoxygenNote 7.2.0

Suggests distributional, fable, fabletools, tinytest, withr

NeedsCompilation yes

Author Hugh Parsonage [aut, cre]

Maintainer Hugh Parsonage <hugh.parsonage@gmail.com>

Repository CRAN

Date/Publication 2023-12-01 21:10:06 UTC

R topics documented:

abs-conn *ABS Connections*

Description

The package uses the catalogue mirrored at https://github.com/HughParsonage/ABS-Catalogue. These functions expose the guts of the package's method to connect to this mirror.

Each inflator, plus the 'adjustment', is associated with an ABS Series ID.

Usage

```
content2series_id(
  broad_cat = c("cpi", "lfi", "wpi"),
 adjustment = c("original", "seasonal", "trend", "trimmed-mean", "monthly-original",
    "monthly-seasonal", "monthly-excl-volatile")
)

download_data(series_id = NULL)

when_last_updated()
```

Arguments

`broad_cat, adjustment`
: Definitions to identify the Series ID. If any are multiple, the result is of the cartesian join, **not** the component-wise values.

`series_id`
: The Series ID desired. For `download_data`, if `NULL`, the default, downloads all files required.

Value

`content2series_id` A character vector, the Series ID identified by 'broad_cat' and 'adjustment'

`download_data` Called for its side-effect, downloading the data required. If successful, returns zero.

`when_last_updated` The date the downloaded data was last retrieved, or the string `"Never"` if the file does not exist.

cpi_inflator *CPI inflator*

Description

CPI inflator

Usage

```
cpi_inflator(
  from = NULL,
  to = NULL,
  series = c("seasonal", "original", "trimmed.mean", "monthly-original",
    "monthly-seasonal", "monthly-excl-volatile"),
  fy_month = 3L,
  x = NULL,
  check = 1L,
  nThread = getOption("grattanInflators.nThread", 1L)
)

cpi_seasonal(..., FORECAST = FALSE, LEVEL = "mean")

cpi_original(..., FORECAST = FALSE, LEVEL = "mean")

cpi_trimmed_mean(..., FORECAST = FALSE, LEVEL = "mean")

cpi_monthly_original(..., FORECAST = FALSE, LEVEL = "mean")

cpi_monthly_seasonal(..., FORECAST = FALSE, LEVEL = "mean")

cpi_monthly_excl_volatile(..., FORECAST = FALSE, LEVEL = "mean")
```

Arguments

from, to	Times for which the inflator is desired. If NULL, a date range close to the previous year is used.
series	Which CPI series to use.
fy_month	An integer 1-12, the month to be used for years and financial years in from or to. Since the CPI is a quarterly series, specifying a year is ambiguous. For financial years, the month is the month of the financial year, so for example fy_month = 9 and "2015-16" means Sep-2015, while fy_month = 6 means Jun-2016.
x	(Advanced) A vector that will be inflated in-place. If NULL, the default, the return vector is simply the inflation factor for 'from'.
check	integer(1) If 0L, no checks are performed, and clearly invalid inputs result in NA in the output. If check = 1L an error is performed for bad input; check = 2L is more thorough.
nThread	Number of threads to use.
...	Set of date-rate pairs for custom CPI series in the future.
FORECAST	Whether the series should be extended via an ETS forecast.
LEVEL	If 'FORECAST = TRUE' what prediction interval should be used. ('LEVEL = 20' means the lower end of an 80% prediction interval.) If 'LEVEL = "mean"' (the default), the central estimate is used.

Value

If ‘x‘ is ‘NULL‘, the default, a numeric vector matching the lengths of ‘from‘ and ‘to‘ equal to the inflators by which nominal prices dated ‘from‘ must be multiplied so that they are in ‘to‘ real terms.

If ‘x‘ is numeric, it is taken to be prices dated ‘from‘ and the value returned is ‘x‘ in ‘to‘ real terms.

Examples

```
cpi_inflator(1995, 2019)  # Inflation from 1995 to 2019
cpi_inflator("2015-16", "2016-17")
cpi_inflator("2015-01-01", "2016-01-01")

if (Sys.Date() < as.Date("2029-01-01")) {
  cpi_inflator("2030-01-01", "2031-01-01",
               series = cpi_original(2030, 0.1))
  cpi_inflator("2030-01-01", "2031-01-01",
               series = cpi_original(0.1))
  cpi_inflator("2030-01-01", "2032-01-01",
               series = cpi_original(2030, 0.1, 2031, 0.1, 2032, 0))
}
```

`custom-series` *Custom series*

Description

Used when the true series is not appropriate, as when a forecast is desired and the series is required beyond the original series.

Usage

```
dr2index(index, d1, r1, ...)
```

Arguments

`index`	An index (i.e. a data.table with columns `date` and `value`, where `date` is an arithmetic sequence of monthly, quarterly, or annual dates), and `value` is the indexed value for that date.
`d1`	A single date or value representing a date.
`r1`	The desired rate of increase for the index from the last date in `index` to the end of `d1`. For example, `d1 = 2025` and `r1 = 0` applied to a monthly `index` would keep `value` constant until `2025-12-01`.
`...`	A set of date-rate pairs.

Value

`index` with dates extended until the last supported date. The final rate supplied is the rate for all dates after the final date.

fast_as_idate *Faster conversion to IDate for common dates*

Description

Faster conversion to IDate for common dates

Usage

```
fast_as_idate(x, incl_day = TRUE, check = 0L, nThread = 1L)
```

Arguments

x	The character vector to convert, in YYYY-mm-dd form only.
incl_day	Whether or not the day is necessary to convert. Set to FALSE when the day component does not matter (or is constantly -01); the day component in the output will be -01.
check	integer: 0, 1, or 2 Level of check to perform. 0 for no checks.
nThread	Number of threads to use.

Value

A vector of class IDate, Date the same length as x.

Examples

```
# For ABS data, we only need to care (and check)
# the year and month
fast_as_idate("2015-12-13", incl_day = FALSE)
```

Inflate *Generic inflator*

Description

Generic inflator

Usage

```
Inflate(
  from,
  to,
  index,
  x = NULL,
  fy_month = 3L,
  check = 2L,
  nThread = getOption("grattanInflators.nThread", 1L)
)
```

Arguments

from, to	Times for which the inflator is desired. If NULL, a date range close to the previous year is used.
index	A table of at least two columns, named date and value. date is the column of times to which from, to will be matched. value is the values that determine the inflation factor.
x	(Advanced) A vector that will be inflated in-place. If NULL, the default, the return vector is simply the inflation factor for 'from'.
fy_month	An integer 1-12, the month to be used for years and financial years in from or to. For financial years, the month is the month of the financial year, so for example fy_month = 9 and "2015-16" means Sep-2015, while fy_month = 6 means Jun-2016.
check	integer(1) If 0L, no checks are performed, and clearly invalid inputs result in NA in the output. If check = 1L an error is performed for bad input; check = 2L is more thorough.
nThread	Number of threads to use.

Value

If 'x' is 'NULL', the default, a numeric vector matching the lengths of 'from' and 'to' equal to the ratio between the corresponding values in the column value.

If 'x' is numeric, those values are multiplied by the inflators, in-place.

---

lf_inflator *Labour force inflator*

Description

Uses the Labour Force Index to provide equivalent sizes of the labour force over different times by multiplying by the simple ratio of the sizes on those dates.

Usage

```
lf_inflator(
  from = NULL,
  to = NULL,
  check = 1L,
  series = lfi_original(),
  x = NULL,
  nThread = getOption("grattanInflators.nThread", 1L)
)

lfi_original(..., FORECAST = FALSE, LEVEL = "mean")

lfi_seasonal(..., FORECAST = FALSE, LEVEL = "mean")

lfi_trend(..., FORECAST = FALSE, LEVEL = "mean")
```

Arguments

from, to	Times for which the inflator is desired. If NULL, a date range close to the previous year is used.
check	integer(1) If 0L, no checks are performed, and clearly invalid inputs result in NA in the output. If check = 1L an error is performed for bad input; check = 2L is more thorough.
series	A call to 'lfi_original()', 'lfi_seasonal()', or 'lfi_trend()'.
x	(Advanced) A vector that will be inflated in-place. If NULL, the default, the return vector is simply the inflation factor for 'from'.
nThread	Number of threads to use.
...	Set of date-rate pairs for custom labour force series in the future.
FORECAST	Whether the series should be extended via an ETS forecast.
LEVEL	If 'FORECAST = TRUE' what prediction interval should be used. ('LEVEL = 20' means the lower end of an 80% prediction interval.) If 'LEVEL = "mean"' (the default), the central estimate is used.

Value

If 'x' is 'NULL', the default, a numeric vector matching the lengths of 'from' and 'to' equal to the relative size of the labour force of 'from' and 'to'.

If 'x' is numeric, it is taken to be the sizes of the labour force on dates 'from' and the value returned is the equivalent size of 'x' on dates 'to' (by simple multiplication).

Examples

```
# The relative size of the labour force in FY 2016-17
# compared to FY 2015-16
lf_inflator("2015-16", "2016-17")
```

wage_inflator *Wage inflator*

Description

Uses the Wage Price Index

Usage

```
wage_inflator(
  from = NULL,
  to = NULL,
  check = 1L,
  series = wpi_original(),
  x = NULL,
  nThread = getOption("grattanInflators.nThread", 1L)
)

wpi_original(..., FORECAST = FALSE, LEVEL = "mean")

wpi_seasonal(..., FORECAST = FALSE, LEVEL = "mean")

wpi_trend(..., FORECAST = FALSE, LEVEL = "mean")
```

Arguments

from, to	Times for which the inflator is desired. If NULL, a date range close to the previous year is used.
check	integer(1) If 0L, no checks are performed, and clearly invalid inputs result in NA in the output. If check = 1L an error is performed for bad input; check = 2L is more thorough.
series	A call to 'wpi_original()', 'wpi_seasonal()', or 'wpi_trend()', defining which wage price index series to use.
x	(Advanced) A vector that will be inflated in-place. If NULL, the default, the return vector is simply the inflation factor for 'from'.
nThread	Number of threads to use.
...	Set of date-rate pairs for custom WPI series in the future.
FORECAST	Whether the series should be extended via an ETS forecast.
LEVEL	If 'FORECAST = TRUE' what prediction interval should be used. ('LEVEL = 20' means the lower end of an 80% prediction interval.) If 'LEVEL = "mean"' (the default), the central estimate is used.

Value

If ‘x‘ is ‘NULL‘, the default, a numeric vector matching the lengths of ‘from‘ and ‘to‘ equal to the inflators by which nominal wages dated ‘from‘ must be multiplied so that they are in ‘to‘ real terms.

If ‘x‘ is numeric, it is taken to be wages dated ‘from‘ and the value returned is ‘x‘ in ‘to‘ real terms.

Index