Package 'gtapssp'

April 10, 2024

Title What the Package Does (One Line, Title Case)

Version 0.0.0.9000

Description What the package does (one paragraph).

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports data.table,
dplyr,
tidyr,
tools

R topics documented:

`calc_beers` *Perform Beers Interpolation or Subdivision*

Description

This function implements the Beers interpolation or subdivision methods, either in their ordinary or modified forms. It generates interpolated or subdivided points from a given data set. The Beers interpolation method, first introduced in "The Record of the American Institute of Actuaries" (Vol. 34, Part I, 1945), is a six-term formula designed to minimize the fifth differences in interpolated results. More details can be found in "The Methods and Materials of Demography, Volume 2" (page 877).

Usage

```
calc_beers(values, method = "ordinary")
```

Arguments

values	A numeric vector of data points for interpolation or subdivision.
method	A character string specifying the method to be used. (default = "ordinary") Must be one of "ordinary", "modified", "subidvision_ordinary", or "subidvision_modified". - "ordinary": In interpolation, include original data points unchanged; in subdivision, each set of 5 subdivided values sums to the original data point. - "modified": In interpolation, includes some smoothing with only the first and last points unchanged; in subdivision, similar smoothing occurs.

Value

A numeric vector with interpolated or subdivided values.

Examples

```
calc_beers(c(1, 2, 3, 4, 5), "ordinary")
calc_beers(c(1, 2, 3, 4, 5), "modified")
calc_beers(c(1, 2, 3, 4, 5), "subidvision_ordinary")
calc_beers(c(1, 2, 3, 4, 5), "subidvision_modified")
```

---

interpolate_beers *Fill Gaps Using Beers Method Interpolation*

Description

This function applies the Beers method interpolation on a grouped data frame. It requires each group to have at least 6 non-NA values and assumes data to be in 5-year intervals. Groups with fewer than 6 non-NA values are excluded from interpolation.

Usage

```
interpolate_beers(input_df, groups, year, values, method = "ordinary")
```

Arguments

input_df	A data frame containing the data to be interpolated.
groups	A vector of column names to group by.
year	The name of the column containing year information.
values	The name of the column containing values for interpolation.
method	A character string specifying the method to be used. Must be one of "ordinary", "modified", "subidvision_ordinary", or "subidvision_modified". - "ordinary" (default): In interpolation, include original data points unchanged; in subdivision, each set of 5 subdivided values sums to the original data point. - "modified": In interpolation, includes some smoothing with only the first and last points unchanged; in subdivision, similar smoothing occurs.

Value

A data frame with interpolated values using the Beers method.

Examples

```
data <- data.frame(
  Scenario = rep(c("Scenario1", "Scenario2"), each = 5),
  Region = rep(c("Region1", "Region2"), each = 5),
  year = rep(c(2000, 2005, 2010, 2015, 2020), 2),
  value = rnorm(10)
)

filled_data <- interpolate_beers(
  input_df = data,
  groups = c("Scenario", "Region"),
  year = "year",
  values = "value"
)
```

---

interpolate_spline *Fill Gaps with Spline Interpolation*

Description

This function takes a data frame and performs cubic spline interpolation to fill in missing year gaps for specified groups. Groups with fewer than 2 non-NA values are excluded from interpolation.

Usage

```
interpolate_spline(input_df, groups, year, values, method = "fmm")
```

Arguments

input_df	A data frame containing the data to be processed.
groups	A vector of column names to group by (to loop applying spline).
year	The name of the column containing years.
values	The name of the numeric column containing values for interpolation.
method	The method of interpolation. Possible values: Must be one of "fmm", "natural", "periodic", "monoH.FC" or "hyman". • "fmm" (default): Forsythe, Malcolm, and Moler method. An exact cubic spline is fitted through the four points at each end of the data. This is used for determining end conditions. Not suitable for extrapolation. • "natural": Natural spline method. Uses natural splines for interpolation. Linear extrapolation outside the range of x using the slope of the interpolating curve at the nearest data point. • "periodic": Periodic spline method. Suitable for periodic data. • "monoH.FC": Monotone Hermite spline according to Fritsch and Carlson. Ensures the spline is monotone (increasing or decreasing) if the data are monotone. • "hyman": Monotone cubic spline using Hyman filtering of an "fmm" fit for strictly monotonic inputs.

Value

A data frame with gaps in years filled using spline interpolation.

Examples

```
data <- data.frame(
  Scenario = rep(c("Scenario1", "Scenario2"), each = 5),
  Region = rep(c("Region1", "Region2"), each = 5),
  year = rep(c(2000, 2005, 2010, 2015, 2020), 2),
  value = rnorm(10)
)

filled_data <- interpolate_spline(
  input_df = data,
  groups = c("Scenario", "Region"),
  year = "year",
  values = "value"
)
```

`read_csv_from_zip` *Combine CSV Files from ZIP Archives*

Description

This function searches for ZIP files in a specified directory that match a given pattern. It then extracts CSV files from these ZIP archives that match another specified pattern. The function can either combine these CSV files vertically into a single data frame or return them as separate data frames in a list. Each data frame in the list is named after its respective CSV file.

Usage

```
read_csv_from_zip(zip_dir, zip_pattern, csv_pattern, combine_vertically = TRUE)
```

Arguments

`zip_dir`	Directory containing the ZIP files.
`zip_pattern`	Pattern to match the ZIP file names.
`csv_pattern`	Pattern to match the CSV file names inside the ZIP archives.
`combine_vertically`	Logical, if TRUE, the function combines all CSV files vertically into a single data frame. If FALSE, returns a list of data frames, each named after its corresponding CSV file.

Value

Either a single combined data frame or a list of data frames, depending on the value of `combine_vertically`.

Examples

```
# Assuming ZIP files are in 'path/to/zip/files'
combined_data <- combine_csv_from_zip(
  zip_dir = "path/to/zip/files",
  zip_pattern = "zip_file_pattern",
  csv_pattern = "csv_file_pattern",
  combine_vertically = TRUE
```

```
)

# To get a list of data frames instead of a combined one
data_list <- combine_csv_from_zip(
  zip_dir = "path/to/zip/files",
  zip_pattern = "zip_file_pattern",
  csv_pattern = "csv_file_pattern",
  combine_vertically = FALSE
)
```

Index