Package ‘EpiReport’

May 11, 2020

Type Package
Title Epidemiological Report
Version 0.1.1
Date 2020-05-11
Description Drafting an epidemiological report in 'Microsoft Word' format for a given disease, similar to the Annual Epidemiological Reports published by the European Centre for Disease Prevention and Control. Through standalone functions, it is specifically designed to generate each disease specific output presented in these reports and includes:
- Table with the distribution of cases by Member State over the last five years;
- Seasonality plot with the distribution of cases at the European Union / European Economic Area level, by month, over the past five years;
- Trend plot with the trend and number of cases at the European Union / European Economic Area level, by month, over the past five years;
- Age and gender bar graph with the distribution of cases at the European Union / European Economic Area level.
Two types of datasets can be used:
- The default dataset of salmonella 2012-2016 data;
- Any dataset specified as described in the vignette.

Depends R (>= 3.4.0)
License EUPL
Encoding UTF-8
LazyData true
RoxygenNote 7.1.0
Imports officer, flextable, zoo, png, dplyr, tidyr, ggplot2, extrafont, graphics, utils, knitr (>= 1.20), rmarkdown
VignetteBuilder knitr


NeedsCompilation no
**Author** Lore Merdrignac [aut, ctr, cre] (Author of the package and original code),
   Tommi Karki [aut, fnd],
   Esther Kissling [aut, ctr],
   Joana Gomes Dias [aut, fnd] (Project manager)

**Maintainer** Lore Merdrignac <l.merdrignac@epiconcept.fr>

**Repository** CRAN

**Date/Publication** 2020-05-11 16:00:07 UTC

**R topics documented:**

- **AERparams** .......................................................... 2
- cleanECDCTable ..................................................... 4
- cleanMeasureCode .................................................... 4
- filterDisease ........................................................ 6
- getAER ................................................................. 6
- getAgeGender ......................................................... 8
- getMap ................................................................. 10
- getSeason ............................................................. 11
- getTableByMS ......................................................... 13
- getTemplate ........................................................ 14
- getTrend .............................................................. 15
- includeMap .......................................................... 17
- MSCode ............................................................... 18
- orderQuasinum ....................................................... 18
- plotAge ............................................................... 19
- plotAgeGender ....................................................... 20
- plotSeasonality ..................................................... 22
- plotTS12MAvg ....................................................... 23
- previewMap .......................................................... 24
- SALM2016 ............................................................ 24
- shapeECDCFlexTable ............................................... 26
- toCapTitle .......................................................... 26

**Index** 28

| AERparams | Dataset describing the parameters for the epidemiological report production |

**Description**

A dataset describing the parameters to be used for each output of each disease report for all 53 health topics included in TESSy
Usage

AERparams

Format

A data frame with 53 rows (corresponding to the 53 health topics) and 24 variables:

- **HealthTopic**: Disease code that should match with the health topic code from the disease-specific dataset e.g. ANTH, SALM, etc.
- **DG** (optional): Disease group e.g. FWD
- **DP** (optional): Disease programme e.g. FWD
- **Label**: Disease label to be used in the report e.g. salmonellosis, anthrax
- **FrequencyCategory** (optional): Frequency of the disease e.g. VERY RARE, NON-RARE, etc.
- **MeasurePopulation**: Type of population presented for this disease i.e. ALL or CONFIRMED cases
- **DatePublicAtlas**: Date of latest availability in the public access of the Atlas
- **TableUse**: Type of table to present in the report i.e. NO table, ASR table presenting age-standardised rates, RATE table presenting rates or COUNT table presenting the number of cases only.
- **TableRatesLabel**: Label to use in the table for rates e.g. RATE PER 100000 POPULATION
- **TableRatesNoDecimals**: Number of decimals to use when presenting rates
- **TableASRNoDecimals**: Number of decimails to use when presenting ASR
- **AgeGenderUse**: Type of age and gender bar graph to present i.e. NO graph, AG-COUNT Bar graph presenting the number of cases by age and gender, AG-RATE Bar graph presenting the rates of cases by age and gender, AG-PROP Bar graph presenting the proportion of cases by age and gender, A-RATE Bar graph presenting the rates of cases by age.
- **AgeGenderBarGraphLabel**: Label to use in the age and gender bar graph
- **AgeGenderGraphNoDecimals**: Number of decimals to use when presenting rates in the age and gender bar graph
- **TSTrendGraphUse**: Logical Y/N specifying whether to include a line graph describing the trend of the disease over the time
- **TSSeasonalityGraphUse**: Logical Y/N specifying whether to include a line graph describing the seasonality of the disease
- **TSSpecific**: Logical Y/N for specific line graph inclusion
- **MapNumbersUse**: Logical Y/N specifying whether to include the map presenting the number of cases by Member State
- **MapRatesUse**: Logical Y/N specifying whether to include the map presenting the rates of cases by Member State
- **MapRatesNoDecimals**: (optional) Number of decimals to use for presenting maps
- **MapASRUse**: Logical Y/N specifying whether to include the map presenting the age-standardised rates of cases by Member State
- **MapASRNoDecimals**: (optional) Number of decimals to use for presenting maps
- **Transmission**: Not implemented yet
- **TransmissionNoDecimals**: Not implemented yet
### cleanECDCTable

**Cleaning the final table**

**Description**

Cleaning the final table: identifying missing reports with '-', replacing the Member State codes with Member State names (see correspondence table MSCode), identifying not reporting Member States with '.'.

**Usage**

```r
cleanECDCTable(
  x,
  Country = EpiReport::MSCode$Country,
  GeoCode = EpiReport::MSCode$GeoCode
)
```

**Arguments**

- `x`: dataframe, dataset to clean
- `Country`: character vector, full names of the countries / Member States (e.g. Austria, Belgium, etc.) that will replace the GeoCodes included the x dataframe (Default MSCode$Country)
- `GeoCode`: character vector, corresponding GeoCode of each Member State (e.g. AT, BE, etc.) to replace with the country full names (Default MSCode$GeoCode)

**Value**

cleaned ECDC dataframe

**See Also**

Global function: `getTableByMS`
Default dataset `MSCode`

---

### cleanMeasureCode

**Clean the MeasureCode variable**

**Description**

Clean the MeasureCode variable and replace the specific codes with the generic ones (e.g. ACCUTE.AGE_GENDER.RATE will be replaced by CONFIRMED.AGE_GENDER.RATE)

**Usage**

```r
cleanMeasureCode(var)
```
**cleanMeasureCode**

**Arguments**

```
var character string vector variable, variable to clean
```

**Details**

- **ALL.COUNT** will replace the following codes:
  - ALL.DOMESTIC.COUNT
  - AGELT1.COUNT

- **ALL.RATE** will replace the following codes:
  - ALL.DOMESTIC.AGE.RATE

- **ALL.AGE.RATE** will replace the following codes:
  - ALL.DOMESTIC.AGE.RATE

- **ALL.AGESTANDARDISED.RATE** will replace the following codes:
  - ALL.DOMESTIC.AGESTANDARDISED.RATE

- **CONFIRMED.COUNT** will replace the following codes:
  - ALL.LABCONFIRMED.COUNT
  - CONFIRMED.LABCONFIRMED.COUNT
  - CONFIRMED.AGELT1.COUNT
  - TYPHOID.COUNT

- **CONFIRMED.RATE** will replace the following codes:
  - CONFIRMED.LABCONFIRMED.RATE
  - CONFIRMED.AGELT1.RATE
  - TYPHOID.RATE

- **CONFIRMED.AGESTANDARDISED.RATE** will replace the following codes:
  - CONFIRMED.LABCONFIRMED.AGESTANDARDISED.RATE

- **CONFIRMED.AGE_GENDER.RATE** will replace the following codes:
  - CONFIRMED.LABCONFIRMED.AGE_GENDER.RATE
  - TYPHOID.AGE_GENDER.RATE
  - ACCUTE.AGE_GENDER.RATE

**Value**

cleaned vector variable

**See Also**

* SALM2016 *

**Examples**

```
x <- EpiReport::SALM2016
x$MeasureCode <- cleanMeasureCode(x$MeasureCode)
```
filterDisease  
*Filter disease parameters*

**Description**
Filter the table of parameters for the report on the given disease

**Usage**
```
filterDisease(dis, reportParameters)
```

**Arguments**
- `dis` character string, disease code
- `reportParameters` dataset of parameters for the report (default AERparams)

**Value**
dataframe with one row (from the AERparams dataframe) corresponding to the parameters of the selected disease

**See Also**
AERparams

**Examples**
```
disease <- "SALM"
reportParameters <- EpiReport::AERparams
reportParameters <- filterDisease(disease, reportParameters)
```

getAER  
*Get full disease-specific epidemiological report*

**Description**
Function to generate the 'Microsoft Word' epidemiological report (similar to the ECDC Annual Epidemiological Report (AER)) including all disease-specific outputs at each output-specific bookmarks exact location.

(for further information on the outputs and the corresponding bookmarks, please see the package vignette "The Epidemiological Report Package" with browseVignettes("EpiReport"))
getAER

Usage

getaER(
  template = file.path(system.file(package = "EpiReport"),
      "template/AER_template.docx"),
  outputPath = getwd(),
  x = EpiReport::SALM2016,
  disease = "SALM",
  year = 2016,
  reportParameters = EpiReport::AERparams,
  MSCode = EpiReport::MSCode,
  pathPNG = system.file("maps", package = "EpiReport")
)

Arguments

template character string, the empty 'Word' document template in which to include the table and plots disease-specific outputs. Default value is the empty template included in the package. See getTemplate().

outputPath character string, the full path where to generate the epidemiological report 'Word' output. Default value is the current working directory getwd().

x dataframe, raw disease-specific dataset (see specification of the dataset in the package vignette with browseVignettes("EpiReport")) (default SALM2016)

disease character string, disease code (default "SALM"). Please make sure the disease code is included in the disease-specific dataset x in the HealthTopicCode variable.

year numeric, year to produce the report for (default 2016). Please make sure the year is included in the disease-specific dataset x in the TimeCode variable.

reportParameters dataframe, dataset including the required parameters for the report production (default AERparams) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))

MSCode dataframe, correspondence table of GeoCode names and codes (default MSCode) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))

pathPNG character string, the full path to the folder containing the maps (in PNG) to include in the final report

Value

A 'Word' document

See Also

Default template: getTemplate
Default datasets: MSCode AERparams SALM2016
Disease-specific outputs: getTableByMS getSeason getTrend getMap getAgeGender
getAgeGender

Examples

# --- Generating the AER report using the default Salmonellosis dataset
getAER()

## Not run:
# --- Or using external data (example below)
ZIKV2016 <- read.table("data/ZIKV2016.csv", sep = ",", header = TRUE, stringsAsFactors = FALSE)
output <- "C:/EpiReport/doc/
pathMap <- "C:/EpiReport/maps/
getAER(disease = "ZIKV", year = 2016, x = ZIKV2016, outputPath = output, pathPNG = pathMap)

## End(Not run)

---

getAgeGender

Get disease-specific age and gender bar graph

Description

Function returning the age and gender bar graph that will be included in the epidemiological report at the bookmark location 'BARGPH_AGE_GENDER_BOOKMARK' of the template report.

The bar graph presents the distribution of cases at EU/EEA level using either:

- AG-COUNT: The number of cases by age and gender
- AG-RATE: The rate per 100 000 cases by age and gender
- AG-PROP: The proportion of cases by age and gender
- A-RATE: The rate per 100 000 cases by age only

The choice of the type of bar graph is set in the report parameters table AERparams.

Usage

getAgeGender(
  x = EpiReport::SALM2016,
  disease = "SALM",
  year = 2016,
  reportParameters = EpiReport::AERparams,
  geoCode = "EU_EEA31",
  index = 1,
  doc
)

getAgeGender

Arguments

x    dataframe, raw disease-specific dataset (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport")) (default SALM2016)

disease    character string, disease code (default "SALM"). Please make sure the disease code is included in the disease-specific dataset x in the HealthTopicCode variable.

year    numeric, year to produce the graph for (default 2016). Please make sure the year is included in the disease-specific dataset x in the TimeCode variable.

reportParameters    dataframe, dataset including the required parameters for the graph and report production (default AERparams) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))

geoCode    character string, GeoCode to run the analysis on (default "EU_EEA31")

index    integer, figure number

doc    'Word' document (see 'officer' package) in which to add the graph at the bookmark location. If doc is missing, getAgeGender returns the ggplot2 object.

Value

'Word' doc or a ggplot2 object

See Also

Global function for the full epidemiological report: getAER
Required Packages: ggplot2 officer
Internal functions: plotAgeGender plotAge
Default datasets: AERparams

Examples

# --- Plot using the default dataset
getAgeGender()

# --- Plot using external dataset
# --- Please see examples in the vignette
browseVignettes(package = "EpiReport")
getMap

Get disease-specific map: distribution of cases by Member State

Description

Function returning the disease-specific PNG map previously created and stored in a specific folder (see pathPNG argument) and that will be included in the epidemiological report at the bookmark location of the template report, depending of the type of map. Three type of maps can be included in the report:

- Bookmark 'MAP_NB_BOOKMARK': Distribution of cases by country. An additional caption will be included at the location of the bookmark 'MAP_NB_CAPTION'.
- Bookmark 'MAP_RATE_BOOKMARK': Distribution of cases per 100 000 population by country. An additional caption will be included at the location of the bookmark 'MAP_RATE_CAPTION'.
- Bookmark 'MAP_ASR_BOOKMARK': Distribution of cases using age-strandardised rates per 100 000 population by country. An additional caption will be included at the location of the bookmark 'MAP_ASR_CAPTION'.

(see ECDC reports https://ecdc.europa.eu/en/annual-epidemiological-reports)

Usage

getMap(
  disease = "SALM",
  year = 2016,
  reportParameters = EpiReport::AERparams,
  index = 1,
  pathPNG = system.file("maps", package = "EpiReport"),
  doc
)

Arguments

disease character string, disease code (default "SALM").
year numeric, year to produce the map for (default 2016).
reportParameters
dataframe, dataset including the required parameters for the map and report production (default AERparams) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))
index integer, figure number
pathPNG character string, full path to the folder containing the maps in PNG (default 'maps' folder included in the package system.file("maps",package = "EpiReport"))
doc 'Word' document (see 'officer' package) in which to add the maps at the bookmark location. If doc is missing, getMap returns a preview of the PNG image.
getSeason

Value

'Word' doc an image preview

See Also

Global function for the full epidemiological report: getAER
Required Packages: officer
Internal functions: includeMap previewMap
Default datasets: AERparams

Examples

# --- Preview of the PNG map using the default Salmonellosis dataset
getMap()

# --- Plot using external PNG image
# --- Please see examples in the vignette
browseVignettes(package = "EpiReport")

getSeason

Get disease-specific seasonality graph: distribution of cases by month

Description

Function returning the plot describing the seasonality of the disease that will be included in the epidemiological report at the bookmark location 'TS_SEASON_BOOKMARK' of the template report.

The graph includes the distribution of cases at EU/EEA level, by month, over the past five years, with:

- The number of cases by month in the reference year (green solid line)
- The mean number of cases by month in the four previous years (grey dashed line)
- The minimum number of cases by month in the four previous years (grey area)
- The maximum number of cases by month in the four previous years (grey area)

(see ECDC reports https://ecdc.europa.eu/en/annual-epidemiological-reports)

Usage

getSeason(
  x = EpiReport::SALM2016,
  disease = "SALM",
  year = 2016,
  reportParameters = EpiReport::AERparams,
  MSCode = EpiReport::MSCode,
  index = 1,
  doc
)
getSeason

Arguments

x dataframe, raw disease-specific dataset (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport")) (default SALM2016)
disease character string, disease code (default "SALM"). Please make sure the disease code is included in the disease-specific dataset x in the HealthTopicCode variable.
year numeric, year to produce the graph for (default 2016). Please make sure the year is included in the disease-specific dataset x in the TimeCode variable.
reportParameters dataframe, dataset including the required parameters for the graph and report production (default AERparams) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))
MSCoDe dataframe, correspondence table of GeoCode names and codes (default MSCode) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))
index integer, figure number
doc 'Word' document (see 'officer' package) in which to add the graph at the bookmark location. If doc is missing, getSeason returns the ggplot2 object.

Value

'Word' doc or a ggplot2 object

See Also

Global function for the full epidemiological report: getAER
Required Packages: ggplot2 officer
Internal functions: plotSeasonality
Default datasets: AERparams MSCode

Examples

# --- Plot using the default dataset
getSeason()

# --- Plot using external dataset
# --- Please see examples in the vignette
browseVignettes(package = "EpiReport")
getTableByMS

Get disease-specific table: distribution of cases by Member State (GeoCode)

Description

Function returning the table ('flextable') that will be included in the epidemiological report at the bookmark location 'TABLE1_BOOKMARK' of the template report. An additional caption will be included at the location of the bookmark 'TABLE1_CAPTION'. (see Table 1 of the ECDC annual reports [https://ecdc.europa.eu/en/annual-epidemiological-reports](https://ecdc.europa.eu/en/annual-epidemiological-reports))

Usage

getableByMS(
  x = EpiReport::SALM2016,
  disease = "SALM",
  year = 2016,
  reportParameters = EpiReport::AERparams,
  MSCode = EpiReport::MSCode,
  index = 1,
  doc
)

Arguments

- **x** dataframe, raw disease-specific dataset (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport")) (default SALM2016)
- **disease** character string, disease code (default "SALM"). Please make sure the disease code is included in the disease-specific dataset x in the HealthTopicCode variable.
- **year** numeric, year to produce the table for (default 2016). Please make sure the year is included in the disease-specific dataset x in the TimeCode variable.
- **reportParameters** dataframe, dataset including the required parameters for the report production (default AERparams) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))
- **MSCode** dataframe, correspondence table of GeoCode names and codes (default MSCode) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))
- **index** integer, figure number
- **doc** 'Word' document (see officer package) in which to add the table at the bookmark location. If doc is missing, getTable returns the flextable table object.
getTemplate

Details
The current version of the 'EpiReport' package includes three types of table (see detailed specification of the tables in the package vignette with browseVignettes(package = "EpiReport")):

- **COUNT** - Table presenting the number of cases by Member State (GeoCode) over a 5-year period;
- **RATE** - Table presenting the number of cases and rates by Member State (GeoCode) over a 5-year period;
- **ASR** - Table presenting the number of cases and rates by Member State (GeoCode) over a 5-year period, including age-standardised rates for the most recent year.

Value
'Word' doc or `flextable` object (see 'flextable' package)

See Also
Global function for the full epidemiological report: `getAER`
Required Packages: `flextable` officer
Internal functions: `shapeECDCFlexTable` `cleanECDCTable`
Default datasets: `AERparams` `MSCode`

Examples
```r
# --- Draft the table using the default Salmonellosis dataset
getTableByMS()
```

---

getTemplate

Get epidemiological report (empty) template

Description
Function to export the generic 'Microsoft Word' empty template (included in the 'EpiReport' package) used to produce the epidemiological report similar to the ECDC Annual Epidemiological Report (AER). The modified version of the template can then be used to produce the final epidemiological report using `getAER(template = 'NewTemplate.docx', ...)` (see the package vignette "The Epidemiological Report Package" with browseVignettes("EpiReport")) (see ECDC annual epidemiological reports [https://ecdc.europa.eu/en/annual-epidemiological-reports](https://ecdc.europa.eu/en/annual-epidemiological-reports))

Usage
`getTemplate(output_path)`

Arguments
- `output_path` character string, the full path where to create the 'Word' output. Default location will be the current working directory (default `getwd()`)
getTrend

Value
A 'Word' document

See Also
getAER

Examples

# --- Export the template in the default folder: working directory
getTemplate()

# --- Or specify the full path
getTemplate(output_path = getwd())

---

getTrend

Get disease-specific trend plot: trend and number of cases by month

Description
Function returning the plot describing the trend of the disease over time that will be included in the epidemiological report at the bookmark location 'TS_TREND_BOOKMARK' on the template report.

The graph includes the number of cases at EU/EEA level, by month, over the past five years, with:

- The number of cases by month over the 5-year period (grey solid line)
- The 12-month moving average of the number of cases by month (green solid line)

(see ECDC reports https://ecdc.europa.eu/en/annual-epidemiological-reports)

Usage

getTrend(
  x = EpiReport::SALM2016,
  disease = "SALM",
  year = 2016,
  reportParameters = EpiReport::AERparams,
  MSCode = EpiReport::MSCode,
  index = 1,
  doc
)
getTrend

Arguments

x   dataframe, raw disease-specific dataset (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport")) (default SALM2016)

disease   character string, disease code (default "SALM"). Please make sure the disease code is included in the disease-specific dataset x in the HealthTopicCode variable.

year   numeric, year to produce the graph for (default 2016). Please make sure the year is included in the disease-specific dataset x in the TimeCode variable.

reportParameters   dataframe, dataset including the required parameters for the graph and report production (default AERparams) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))

MSCode   dataframe, correspondence table of GeoCode names and codes (default MSCode) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))

index   integer, figure number

doc   'Word' document (see officer package) in which to add the graph at the bookmark location. If doc is missing, getTrend returns the ggplot2 object.

Value

'Word' doc or a ggplot2 preview

See Also

Global function for the full epidemiological report: getAER
Required Packages: ggplot2 officer
Internal functions: plotTS12MAvg
Default datasets: AERparams MSCode

Examples

# --- Plot using the default dataset
getTrend()

# --- Plot using external dataset
# --- Please see examples in the vignette
browseVignettes(package = "EpiReport")
includeMap

Including PNG map in the 'Microsoft Word' template

Description

Function including the disease-specific PNG map in the 'Word' document at the specific bookmark location.

Usage

includeMap(
    disease,  # character string, disease code (default "SALM").
    year,     # numeric, year to produce the graph for (default 2016).
    reportParameters,  # dataframe, dataset including the required parameters for the graph and report production (default AERparams) (see specification of the dataset in the package vignette with browseVignettes(package = "EpiReport"))
    index,    # integer, figure number
    pathPNG,  # character string, full path to the folder containing the maps in PNG (default 'maps' folder included in the package system.file("maps", package = "EpiReport").
    doc,      # 'Word' document (see 'officer' package) in which to add the maps at the bookmark location
    pop,      # character string, label of the type of population to use in the caption (e.g. confirmed)
    namePNGsuffix,  # character string, suffix of the PNG file name of the map (i.e. "COUNT", "RATE" or "AGESTANDARDISED").
    unit,   # character string, label of the unit used in the caption (e.g. "per 100 000 population")
    mapBookmark,  # character string, label of the bookmark where to add the map in the 'Word' document
    captionBookmark  # character string, label of the bookmark where to add the caption in the 'Word' document
)
**MSCode**

*Dataset correspondence table between country names and country code*

**Description**

Dataframe providing the correspondence table of the geographical code GeoCode used in the disease dataset, and the geographical label Country to use throughout the report. Additional information on the EU/EEA affiliation is also available in column EUEEA.

**Usage**

MSCode

**Format**

A data frame with 32 rows and 3 variables:

- **Country**  Full name of the country / Member State e.g. Austria, Belgium, etc.
- **GeoCode**  Associated code (see GeoCode variable on the SALM2016 internal dataset) e.g. AT, BE, BG, etc.
- **EUEEA**   For each Member State, variable specifying in the country is part of the EU or EEA.

**See Also**

SALM2016

---

**orderQuasinum**

*Order 'quasinumerical' categorical vectors (increasing order)*

**Description**

A function to order 'quasinumerical' (i.e. categorical with values such as "15-30" or "<18") integer vectors into increasing order. Currently handles away the following non-numerical characters "-", "+". 

**Usage**

orderQuasinum(x)
Arguments

- **x**: character vector with 'quasinumerical' values

Author(s)

Tommi Karki

See Also

Used in `getAgeGender` and `plotAgeGender / plotAge`

Examples

```r
age1 <- c("<1", "1-15", "16-25", ">65", "26-65")
age2 <- c("0-4", "5-10", ">65", "25-64", "11-25")
age3 <- c("5-10", ">65", "25-64", "11-25", "<=4")
age4 <- c(">=65", "<18", "18-64")
age5 <- c("5-10", "+65", "25-64", "11-25", "0-4")

age1
orderQuasinum(age1)
age2
orderQuasinum(age2)
age3
orderQuasinum(age3)
age4
orderQuasinum(age4)
age5
orderQuasinum(age5)
```

---

**plotAge**

*Age bar graph*

Description

This function draws a bar graph by age group (or possibly other grouping).

The bar graph presents the distribution of cases at EU/EEA level using the rate per 100 000 cases by age.

Expects aggregated data.

Usage

```r
plotAge(
  data,
  xvar = "XLabel",
  yvar = "YValue",
  fill_color1 = "#65B32E",
  ytitle = "Rate"
)
```
plotAgeGender

**Arguments**

- **data**: dataframe containing the variables to plot
- **xvar**: character string, name of the variable to plot on the x-axis in quotes (default "XLabel")
- **yvar**: character string, name of the variable to plot on the y-axis in quotes (default "YValue")
- **fill_color1**: character string, hexadecimal colour to use in the graph; (default to ECDC green 
  "#65B32E")
- **ytitle**: character string, y-axis title; (default "Rate").

**See Also**

Global function: `getAgeGender`

Required Packages: `ggplot2`

**Examples**

```r
# --- Create dummy data
mydat <- data.frame(AgeGroup = c("0-25", "26-65", "65+"),
                    NumberOfCases = c(54, 32, 41))

# --- Plot the dummy data
plotAge(mydat,
        xvar = "AgeGroup",
        yvar = "NumberOfCases",
        ytitle = "Number of cases")
```

---

**plotAgeGender**

*Age and Gender bar graph*

**Description**

This function draws a bar graph of the distribution of cases by age group and gender (or possibly other grouping).

The bar graph presents the distribution of cases at EU/EEA level using either:

- **AG-COUNT**: The number of cases by age and gender
- **AG-RATE**: The rate per 100 000 cases by age and gender
- **AG-PROP**: The proportion of cases by age and gender

Expects aggregated data.
Usage

plotAgeGender(
  data,
  xvar = "XLabel",
  yvar = "ZValue",
  group = "YLabel",
  fill_color1 = "#65B32E",
  fill_color2 = "#7CBDC4",
  ytitle = "Rate"
)

Arguments

data dataframe containing the variables to plot
xvar character string, name of the variable to plot on the x-axis in quotes (default "XLabel")
yvar character string, name of the variable to plot on the y-axis in quotes (default "ZValue")
group character string, name of the grouping variable in quotes, e.g. gender. (default "YLabel")
fill_color1 character string, hexadecimal colour to use in the graph for bar 1; (default to ECDC green "#65B32E")
fill_color2 character string, hexadecimal colour to use in the graph for bar 2; (default to ECDC blue "#7CBDC4")
ytitle character string, y-axis title; (default "Rate").

See Also

Global function: getAgeGender
Required Packages: ggplot2

Examples

# --- Create dummy data
mydat <- data.frame(Gender=c("F", "F", "M", "M"),
  AgeGroup = c("0-65", "65+", "0-65", "65+"),
  NumberOfCases = c(54,43,32,41))

# --- Plot the dummy data
plotAgeGender(mydat,
  xvar = "AgeGroup",
  yvar = "NumberOfCases",
  group = "Gender",
  ytitle = "Number of cases")
plotSeasonality  
Seasonality line graph

Description

This function draws a line graph describing the seasonality of the selected disease over the past 5 years.
The graph includes the distribution of cases, by month, over the past five years, with:

• `yvar`: The number of cases by month in the reference year (green solid line)
• `mean4years`: The mean number of cases by month in the four previous years (grey dashed line)
• `min4years`: The minimum number of cases by month in the four previous years (grey area)
• `max4years`: The maximum number of cases by month in the four previous years (grey area)

Expects aggregated data and pre-calculated min, max and mean figures.

Usage

```r
plotSeasonality(
  data,
  xvar = "TimeCode",
  yvar = "N",
  min4years = "Min4Years",
  max4years = "Max4Years",
  mean4years = "Mean4Years",
  year = 2016
)
```

Arguments

data  dataframe containing the variables to plot  
xvar  character string, name of the time variable on the x-axis in quotes (default "TimeCode")  
yvar  character string, name of the variable to plot on the y-axis in quotes (default "N"), number of cases by month in the reference year (green solid line)  
min4years  character string, name of the variable to plot in quotes including the minimum number of cases by month over the past 4 years (default "Min4Years")  
max4years  character string, name of the variable to plot in quotes including the maximum number of cases by month over the past 4 years (default "Max4Years")  
mean4years  character string, name of the variable to plot in quotes including the mean of the number of cases by month over the past 4 years (default "Mean4Years")  
year  numeric, year to produce the graph for (default 2016).
plotTS12MAvg

See Also

Global function: getSeason
Required Packages: ggplot2

plotTS12MAvg  Time series with 12-month moving average

Description

This function draws a line graph describing the trend of the selected disease over the past 5 years. The graph includes the trend and number of cases at EU/EEA level, by month, over the past five years, with:

- `yvar`: The number of cases by month over the 5-year period (grey solid line)
- `movAverage`: The 12-month moving average of the number of cases by month (green solid line)

Expects aggregated data and pre-calculated 12-month moving average.

Usage

```r
plotTS12MAvg(data, xvar = "TimeCode", yvar = "N", movAverage = "MAV")
```

Arguments

- `data`: dataframe containing the variables to plot
- `xvar`: character string, name of the time variable to plot on the x-axis in quotes (default "TimeCode")
- `yvar`: character string, name of the variable to plot on the y-axis in quotes (default "N"), number of cases by month over the 5-year period (grey solid line)
- `movAverage`: character string, name of the variable to plot in quotes including the moving average per each time unit (default "MAV")

See Also

Global function: getTrend
Required Packages: ggplot2
Previewing the PNG map

Description

Function previewing the disease-specific PNG map

Usage

`previewMap(disease, year, reportParameters, pathPNG, namePNGsuffix)`

Arguments

disease character string, disease code (default "SALM").
year numeric, year to produce the graph for (default 2016).
reportParameters dataframe, dataset including the required parameters for the graph and report production (default `AERparams`) (see specification of the dataset in the package vignette with `browseVignettes(package = "EpiReport")`)
pathPNG character string, full path to the folder containing the maps in PNG (default 'maps' folder included in the package `system.file("maps",package = "EpiReport")`)
namePNGsuffix character string, suffix of the PNG file name of the map (i.e. "COUNT", "RATE" or "AGESTANDARDISED").

Value

Preview

See Also

Global function: `getMap`

Dataset including Salmonellosis data for 2012-2016

Description

A dataset containing the data and indicators required to build the epidemiological report for Salmonellosis 2016 TESSy data (default dataset used throughout EpiReport)

Usage

`SALM2016`
Format

A data frame with 60,775 rows and 18 variables:

- **HealthTopicCode**: Disease code e.g. ANTH, SALM, etc.
- **MeasureLabel**: (optional) Label of the measure indicator
- **MeasurePopulation**: Population targeted by the measure indicator
- **MeasureCode**: Code of the measure indicator
- **MeasureId**: (optional) Measure indicator ID
- **MeasureType**: (optional) Type of measure indicator
- **TimeUnit**: Unit of the time variable i.e. Y for yearly data or M for monthly data
- **GeoLevel**: (optional) Geographical level e.g. 1, 2, etc
- **TimeCode**: Time variable including dates in any formats available (according to the unit defined in **TimeUnit**) yearly data (e.g. 2001) or monthly data (e.g. 2001-01)
- **GeoCode**: Geographical level in coded format including country names (e.g. AT for Austria, BE for Belgium, BG for Bulgaria, see also the EpiReport::MSCode table, correspondence table for Member State labels and codes)
- **XValue**: (optional) XValue
- **XLabel**: The label associated with the x-axis in the epidemiological report (see getAgeGender() and plotAgeGender() bar graph for the age variable)
- **YValue**: The value associated with the y-axis in the epidemiological report (see plotAge() bar graph for the variable age, or getTableByMS() for the number of cases, rate or age-standardised rate in the table by Member States by year)
- **YLabel**: The label associated with the y-axis in the epidemiological report (see getAgeGender() and plotAgeGender() bar graph for the grouping variable gender)
- **ZValue**: The value associated with the stratification of XLabel and YLabel in the age and gender bar graph (see getAgeGender() and plotAgeGender())
- **N**: Number of cases (see getTrend() and getSeason() line graph)
- **NMissing**: (optional)
- **NLowerResolution**: (optional)

See Also

The correspondence table for Member State labels and codes MSCode and the functions mentioned above: getAgeGender, plotAgeGender, plotAge, getTableByMS, getTrend and getSeason.
shapeECDCFlexTable  
*Shaping the final table (layout, title, color, font)*

**Description**
Shaping the final table including titles, adding background color, specifying font name and size.

**Usage**
`shapeECDCFlexTable(ft, headers, fsize, fname, maincolor)`

**Arguments**
- `ft` flextable (see `flextable` package), table to shape into ECDC table layout
- `headers` dataframe including the multiple headers to add to the flextable object. Please note that the column `col_keys` should contain the names of the flextable object (i.e. `col_key = names(x)`), accordingly to `set_header_df`.
- `fsize` numeric, font to use (Default 7)
- `fname` character, font name (Default "Tahoma")
- `maincolor` character string, hexadecimal code for the header background color (Default "#69AE23")

**Value**
flextable object (see `flextable` package)

**See Also**
Global function: `getTableByMS`
Required package `flextable`

toCapTitle  
*Capitalise first letter*

**Description**
Capitalise the first letter of a character string in order to use it as title

**Usage**
toCapTitle(str)

**Arguments**
- `str` character string to capitalise as a title
toCapTitle

**Value**

character string

**Examples**

```r
my_title <- "number of salmonellosis cases by age group"
toCapTitle(my_title)
```
Index

*Topic age
  plotAge, 19
  plotAgeGender, 20
*Topic bargraph
  plotAge, 19
  plotAgeGender, 20
*Topic datasets
  AERparams, 2
  MSCode, 18
  SALM2016, 24
*Topic gender
  plotAgeGender, 20
*Topic order
  orderQuasinum, 18
*Topic salmonellosis
  SALM2016, 24
*Topic seasonality
  plotSeasonality, 22
*Topic trend
  plotTS12MAvg, 23

AERparams, 2, 6, 7, 9, 11, 12, 14, 16

cleanECDCTable, 4, 14
cleanMeasureCode, 4
filterDisease, 6
flextable, 14, 26

getAER, 6, 9, 11, 12, 14–16
getAgeGender, 7, 8, 19–21, 25
getMap, 7, 10, 18, 24
getSeason, 7, 11, 23, 25
getTableByMS, 4, 7, 13, 25, 26
getTemplate, 7, 14
getTrend, 7, 15, 23, 25
ggplot2, 9, 12, 16, 20, 21, 23
includeMap, 11, 17
MSCode, 4, 7, 12, 14, 16, 18, 25

officer, 9, 11, 12, 14, 16
orderQuasinum, 18
plotAge, 9, 19, 19, 25
plotAgeGender, 9, 19, 20, 25
plotSeasonality, 12, 22
plotTS12MAvg, 16, 23
previewMap, 11, 24
SALM2016, 5, 7, 18, 24
set_header_df, 26
shapeECDCFlexTable, 14, 26
toCapTitle, 26