Package 'ehep'

December 17, 2021

Type Package

Title Ethiopia Health Extension Program Capacity Modelling

Version 0.1.3

Author Charles Eliot, Brittany Hagedorn

Maintainer Charles Eliot <charles.eliot@gatesfoundation.org>

Description This work allows modeling of the burdens placed on healthcare
workers by the addition of new services to the Ethiopian healthcare system.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

Imports jsonlite,
readxl,
data.table,
assertthat,
dplyr,
magrittr,
ggplot2

R topics documented:

Index   **13**

.computeDemographicsProjection

Compute Demographics Projection

Description

Use an initial population pyramid, fertility rates, and mortality rates to predict future population pyramids

Usage

```
.computeDemographicsProjection(
  initial_population_pyramid,
  fertility_rates,
  mortality_rates,
  years,
  debug = FALSE
)
```

Arguments

initial_population_pyramid
                 Population pyramids dataframe. Must have `$Male` and `$Female` fields

fertility_rates
                 Fertility rates

mortality_rates
                 Mortality rates

years   Vector of years to model

debug   Flag for debugging output

Value

Demographics time-series

ClinicalTaskTime   *Calculate Annual Time For Clinical Task*

Description

Calculate Annual Time For Clinical Task

Usage

```
ClinicalTaskTime(tasks, taskID, demographics, year)
```

Arguments

tasks	Dataframe of task parameters (as returned by `loadTaskParameters`)
taskID	Task ID string
demographics	List of population pyramid dataframes
year	Year (index into `demographics` list)

Value

Annual time in minutes

ClinicalTaskTimesGroup

Calculate Clinical Task Times

Description

Calculate clinical task times for a group of tasks over a spread of years

Usage

```
ClinicalTaskTimesGroup(tasks, taskIDs, demographics, years)
```

Arguments

tasks	Dataframe of task parameters (as returned by `loadTaskParameters`)
taskIDs	Vector of task ID strings
demographics	List of population pyramid dataframes
years	Vector of years (usually `globalPackageEnvironment$years`)

Value

Dataframe of annual times in minutes

---

computeBirths *Compute Births*

Description

Compute the total number of births for a year, given the female population pyramid and the annual fertility rates per age.

Usage

```
computeBirths(female_population, rates)
```

Arguments

female_population

Vector of female population pyramid

rates   Exploded vector of annual fertility rates

Value

Total number of expected births

---

computeDeaths *Compute Deaths*

Description

Compute the total number of deaths for a year, given the population pyramids for females and males, and the annual rates per age for both sexes.

Usage

```
computeDeaths(population, rates)
```

Arguments

population   Dataframe of female and male population pyramids, as returned by, for example, `loadInitialPopulation`

rates   List with exploded vectors of annual death rates, as returned by `explodeMortalityRates`

Value

List of expected deaths, one vector for each sex

ComputeDemographicsProjection

Compute Demographics Projection

Description

Use the population pyramid, fertility rates, and mortality rates loaded globally to predict future population pyramids. Drops through to .computeDemographicsProjection()

Usage

```
ComputeDemographicsProjection(...)
```

Value

Demographics time-series

explodeFertilityRates   *Convert Fertility Rates From Banded To Per-Age*

Description

Birth rates are reported in age bands - 15-19 years, 20-29 years, etc. explodeFertilityRates converts the vector of banded rates into a vector with one rate per year of age.

Usage

```
explodeFertilityRates(banded_annual_rates)
```

Arguments

banded_annual_rates

Rates reported in age buckets

Value

List of vectors of per-year-of-age rates, for males and females

---

explodeMortalityRates   *Convert Mortality Rates From Banded To Per-Age*

Description

Mortality rates are reported in age bands - 1-4 years, 5-9 years, etc. explodeMortalityRates converts the vector of banded rates into a vector with one rate per year of age.

Usage

```
explodeMortalityRates(banded_annual_rates)
```

Arguments

banded_annual_rates
                    Rates reported in age buckets

Value

List of vectors of per-year-of-age rates, for males and females

---

generateFertilityRates
                    Generate Time-Series of Fertility Rates

Description

Take the initial value and change rate information returned be a call to loadPopulationChangeParameters and derive a database of annual fertility rates stratified by population age cohort.

Usage

```
generateFertilityRates(popChangeParamsList = NULL)
```

Arguments

sheetName   Population parameters list (see loadPopulationChangeParameters).

Value

Dataframe of annual fertility rates

---

generateMortalityRates

Generate Time-Series of Mortality Rates

Description

Take the initial value and change rate information returned by a call to loadPopulationChangeParameters and derive a database of annual mortality rates stratified by age.

Usage

```
generateMortalityRates(popChangeParamsList = NULL)
```

Arguments

sheetName   Population parameters list (see loadPopulationChangeParameters).

Value

Dataframe of annual mortality rates

InitializeHealthcareTasks

Initialize Healthcare Task Information

Description

Read the healthcare task information from the model inputs Excel file, and save to a location in the global package environment.

Usage

```
InitializeHealthcareTasks()
```

Arguments

sheetName   Sheet name from the model input Excel file

Value

Data frame of healthcare task parameters

InitializePopulation   *Initialize Population Data*

Description

Load basic population information into the global package environment, from which it can be used for later processing.

Usage

```
InitializePopulation()
```

Value

NULL (invisible)

Examples

```
## Not run:
ehep::InitializePopulation()

## End(Not run)
```

---

loadGlobalConfig   *Load Global Configuration*

Description

Finds and loads global configuration data from a JSON file, including things like the default locations of data files.

Usage

```
loadGlobalConfig(path = "./globalconfig.json")
```

Arguments

path   Location of global configuration file

Value

NULL (invisible)

`loadInitialPopulation` *Load Initial Population*

Description

Read the initial population pyramid from the model inputs Excel file. The name and location of the model inputs Excel file is loaded from the global configuration JSON file.

Usage

```
loadInitialPopulation(sheetName = "TotalPop")
```

Arguments

sheetName   Sheet name from the model input Excel file

Value

Population pyramid data frame. Values are rounded to integers.

`loadPopulationChangeParameters`
Load Population Change Parameters

Description

Load Population Change Parameters

Usage

```
loadPopulationChangeParameters(sheetName = "PopValues")
```

Arguments

sheetName   Sheet name from model input Excel file.

Value

List with two vectors: `initValues` and `changeRates`

---

loadSeasonalityCurves   *Load Seasonality Curves*

Description

Read the seasonality curves from the model inputs Excel file. The name and location of the model inputs Excel file is loaded from the global configuration JSON file.

Usage

```
loadSeasonalityCurves(sheetName = "SeasonalityCurves")
```

Arguments

sheetName   Sheet name from the model input Excel file

Value

Seasonality curves data frame.

---

loadTaskParameters   *Load Healthcare Task Information*

Description

Read the healthcare task information from the model inputs Excel file. The name and location of the model inputs Excel file is loaded from the global configuration JSON file.

Usage

```
loadTaskParameters(sheetName = "TaskValues")
```

Arguments

sheetName   Sheet name from the model input Excel file

Value

Data frame of healthcare task parameters

PlotPopulationCurve *Plot A Single Population Curve*

Description

Plot A Single Population Curve

Usage

```
PlotPopulationCurve(pop, xaxis = NULL, color = NULL)
```

Arguments

pop	Population values as a numeric vector
xaxis	Vector of age values for the x axis
color	Plot color

Value

ggplot grob

PlotPopulationCurves *Plot A Family Of Population Curves*

Description

Plot A Family Of Population Curves

Usage

```
PlotPopulationCurves(..., xaxis = NULL, colors = NULL, shapes = NULL)
```

Arguments

...	Population vectors
xaxis	Vector of age values for the x axis
colors	Vector of colors for each population line
shapes	Vector of shapes for each population point

Value

ggplot grob

Trace *Turn Package Tracing On/Off*

Description

Turn Package Tracing On/Off

Usage

```
Trace(state = NULL)
```

Arguments

state TRUE/FALSE to set tracing state, NULL or empty to return current state.

Value

Original state

Examples

```
oldState <- Trace()
print(oldState)
```

TraceMessage *Log A Trace Message*

Description

Log A Trace Message

Usage

```
TraceMessage(msgString)
```

Arguments

msgString Trace message

Index