

Package: demor (via r-universe)

May 27, 2026

Title Methods for Demographic Analysis

Version 1.0.10

Description Implements life tables, fertility and mortality indicators, decomposition methods, Lee-Carter mortality forecasting, Leslie matrices, and population pyramids for demographic analysis. Methods are described in Preston et al. (2001, ISBN:1557864519) and Ustyuzhanin (2025) [<doi:10.17323/demreview.v12i4.30415>](https://doi.org/10.17323/demreview.v12i4.30415).

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports dplyr, forecast, ggplot2, magrittr, tidyr, scales, splines

Depends R (>= 4.0.0)

LazyData true

LazyDataCompression gzip

LazyLoad true

Suggests knitr, rmarkdown, stringr, testthat

URL <https://vadvu.github.io/demor/>, <https://github.com/vadvu/demor>

BugReports <https://github.com/vadvu/demor/issues>

Config/testthat/edition 3

Config/pak/sysreqs libicu-dev

Repository <https://vadvu.r-universe.dev>

Date/Publication 2026-05-27 21:08:30 UTC

RemoteUrl <https://github.com/vadvu/demor>

RemoteRef HEAD

RemoteSha 316e7539db8b4deb40b1d93b799a0866c2fedfe0

Contents

ages	2
asdt	3
asdtex	4
ccm	5
decomp	6
edagger	8
fert.approx	8
get_rosbris	10
gini	12
hli	12
leecart	13
leslie	16
LT	17
mac	18
mdecomp	19
mdecompex	20
med.age	21
MLT	21
mort.approx	22
plot.mdecomp	24
plot_pyr	25
rosbris.codes	26
sle_stand	26
summary.leslie	27
tatfr	28
tfr	29
yll	30
Index	32

ages

Split continuous ages to age groups

Description

Split continuous ages to age groups

Usage

```
ages(x, groups, char = FALSE, below_min_val = NA)
```

Arguments

x	Numeric array. Values of age that should be grouped
groups	Numeric array. Values of lower boundaries of age groups
char	Logical. Should output be numeric (FALSE) or character (TRUE)? By default, FALSE
below_min_val	What value to return for $x < \min(\text{groups})$ when <code>char = FALSE</code> ? Default is NA.

Value

If `char = TRUE`, a factor of the same length as `x` with age-group labels. Otherwise, a numeric vector of the same length as `x` with the lower boundaries from `groups`; values below the first group receive `below_min_val`.

Examples

```
groups <- c(0, 1, 5, 10, 15, 20, 24, 45, 85)
x <- 0:100
ages(x, groups, TRUE)
```

asdt	<i>Associated single decrement life table (ASDT) for causes of death (cause-deleted life table)</i>
------	---

Description

Associated single decrement life table (ASDT) for causes of death (cause-deleted life table)

Usage

```
asdt(age, m_all, m_i, full = FALSE, method = "chiang1968", ...)
```

Arguments

age	Numeric array of age intervals; for full life table = <code>0:100</code> ; for concise life table = <code>c(0:1, seq(5, 85, 5))</code>
m_all	Numeric array with age specific mortality rates of all causes of death (usual <code>mx</code>).
m_i	Numeric array with age specific mortality rates of some cause of death (<code>i</code>)
full	Logical. Is full table needed? TRUE for full, FALSE for concise. By default, FALSE
method	Character. The method of ASDT construction to use. Now just "chiang1968" is supported.
...	Optional. Additional arguments for <code>LT()</code> function.

Value

If `full = TRUE`, a data frame with the full associated single decrement life table, including the standard life-table columns and the cause-deleted columns `r_not_i`, `p_not_i`, `l_not_i`, `a_not_i`, `d_not_i`, `L_not_i`, `T_not_i`, and `ex_without_i`. If `full = FALSE`, a reduced data frame with columns `age`, `r_not_i`, `lx`, `qx`, `ax`, `ex`, `p_not_i`, `l_not_i`, `a_not_i`, and `ex_without_i`.

References

Chiang, L. (1968). *Introduction to Stochastic Processes in Biostatistics*. New York: John Wiley and Sons.

Examples

```
data(asdtex)
asdt(
  age = asdtex$age,
  m_all = asdtex$m_all,
  m_i = asdtex$m_circulatory
)[1:3, ]
```

asdtex

Data on mortality of US men in 2002 by some causes

Description

Data on mortality of US men in 2002 by some causes

Usage

```
asdtex
```

Format**A dataframe:**

age age

neoplasms age specific mortality rate (asmr) from Neoplasms

circulatory asmr from Diseases of the circulatory system

respiratory asmr from Diseases of the respiratory system

digestive asmr from Diseases of the digestive system

other asmr from other causes

all overall asmr

Source

Andreev & Shkolnikov spreadsheet, can be seen on this [webpage](#)

ccm

*Cohort Component Model for Projecting Population.***Description**

Cohort Component Model for Projecting Population.

Usage

```
ccm(
  Mx.f,
  Mx.m = NULL,
  Fx,
  Ix.f = NULL,
  Ix.m = NULL,
  age.mx,
  age.fx,
  N0.f,
  N0.m = NULL,
  srb = 100/205,
  ...
)
```

Arguments

Mx.f	Matrix or dataframe with mx for females, where row represents age, column represents a forecasted period. Thus, first column should be the first projected period, while each row represents age-specific mortality rate.
Mx.m	Optional. Matrix or dataframe with mx for males, where row represents age, column represents a forecasted period. Thus, first column should be the first projected period, while each row represents age-specific mortality rate.
Fx	Matrix or dataframe with fx, where row represents age, column represents a forecasted period. Thus, first column should be the first projected period, while each row represents age-specific fertility rate.
Ix.f	Optional. Matrix or dataframe with net number of female migrants, where row represents age, column represents a forecasted period. Thus, first column should be the first projected period, while each row represents net number of migrants.
Ix.m	Optional. Matrix or dataframe with net number of male migrants, where row represents age, column represents a forecasted period. Thus, first column should be the first projected period, while each row represents net number of migrants.
age.mx	Numeric vector. Age groups for Mx.f rows.
age.fx	Numeric vector. Age groups for Fx rows.
N0.f	Numeric vector. Female population in the initial period by age groups.
N0.m	Optional. Numeric vector. Male population in the initial period by age groups.
srb	Numeric. Sex ratio at birth for females. By default, it is 100/205.
...	Optional. Additional arguments for leslie() function.

Details

The model is calculated in matrix form as

$$\mathbf{N}_{t+h} = \mathbf{L}_t(\mathbf{N}_t + \mathbf{I}_t/2) + \mathbf{I}_t/2$$

where \mathbf{N}_t is a column vector of population for time t with h as a step of projection (it is the length of age interval), \mathbf{L}_t is Leslie matrix and \mathbf{I}_t is a column vector of net migration.

Note that the model assumes that in \mathbf{N} all age intervals are the same (i.e. $age = \{0 - 1, 1 - 4, 5 - 9, \dots\}$ is not permitted). Fortunately (and thanks to me), the function handles such situations automatically by transforming all age groups into a unified standard.

Value

If `Mx.m` is `NULL`, a numeric matrix with projected female population, where rows are age groups and columns are periods $0:h$. If male mortality is supplied, a list with matrices `female`, `male`, and `all`, each with the same row/column structure.

Examples

```
age.mx <- seq(0, 80, 5)
age.fx <- seq(15, 45, 5)
Mx.f <- matrix(
  rep(seq(0.005, 0.12, length.out = length(age.mx)), 2),
  nrow = length(age.mx),
  ncol = 2
)
Fx <- matrix(
  rep(c(0.02, 0.08, 0.11, 0.09, 0.05, 0.02, 0.005), 2),
  nrow = length(age.fx),
  ncol = 2
)
N0.f <- round(100000 * exp(-0.04 * age.mx))
ccm(Mx.f = Mx.f, Fx = Fx, age.mx = age.mx, age.fx = age.fx, N0.f = N0.f)
```

 decomp

Age decomposition of mortality: single decrement process

Description

Age decomposition of mortality: single decrement process

Usage

```
decomp(mx1, mx2, sex = "m", age, method = "andreev", ax1 = NULL, ax2 = NULL)
```

Arguments

mx1	Numeric array with age specific mortality rates of population 1 (base population).
mx2	Numeric array with age specific mortality rates of population 2 (compared population).
sex	Character. Sex. "m" for males and "f" for females. By default, sex = "m".
age	Numeric array of age intervals; for full life table = 0:100; for concise life table = c(0:1, seq(5, 85, 5))
method	Character. Decomposition method. "andreev" (1982), "arriaga" (1984) or "pollard" (1982) - slightly different in their results. By default, method = "andreev".
ax1	Optional. Numeric array with ax for the 1st population. By default, it is a the middle of the interval, while ax for age 0 is modeled as in Andreev & Kingkade (2015).
ax2	Optional. Numeric array with ax for the 2nd population. By default, it is a the middle of the interval, while ax for age 0 is modeled as in Andreev & Kingkade (2015).

Details

Example of decomposition using Andreev (1982) formulas:

$$\Delta_x = l_x^2(e_x^2 - e_x^1) - l_{x+n}^2(e_{x+n}^2 - e_{x+n}^1), \quad x \neq \omega$$

$$\Delta_\omega = l_\omega^2(e_\omega^2 - e_\omega^1)$$

where Δ_x is an absolute contribution of age x to difference in e_0 between the second and the first population. e_x^i, l_x^i are life table functions for population i . ω is the last age group. Note, $e_0^2 - e_0^1 = \sum_x^\omega \Delta_x$

Value

A data frame with one row per age group. It always contains age, ex12, and ex12_prc, where ex12 is the absolute age contribution to the life-expectancy difference and ex12_prc is the percentage contribution. Additional columns depend on method and contain intermediate life-table quantities used in the decomposition.

References

1. Arriaga, E. E. (1984). Measuring and explaining the change in life expectancies. *Demography*, 21, 83-96.
2. Andreev, E. M. (1982). Metod komponent v analize prodolzhitel'nosti zhizni. *Vestnik statistiki*, 9, 42-47.
3. Pollard, J. H. (1982). The expectation of life and its relationship to mortality. *Journal of the Institute of Actuaries*, 109(2), 225-240.

See Also

[mdecomp\(\)](#) for age and cause decomposition

Examples

```
age <- 0:5
mx1 <- c(0.02, 0.01, 0.012, 0.015, 0.02, 0.03)
mx2 <- c(0.018, 0.009, 0.011, 0.014, 0.019, 0.028)
decomp(mx1, mx2, age = age)$ex12
```

edagger

e-dagger

Description

e-dagger

Usage

```
edagger(age, mx, ...)
```

Arguments

age	Numeric array of age intervals
mx	Numeric array with age specific mortality rates.
...	Optional. Additional arguments for <code>LT()</code> function.

Value

A named numeric vector of the same length as age, where each element is $e^{\backslash dagger}_x$, the average remaining life years lost because of death from age x onward.

Examples

```
age <- 0:5
mx <- c(0.02, 0.01, 0.012, 0.015, 0.02, 0.03)
edagger(age, mx)[1:3]
```

fert.approx

Fertility models for ASFR approximation

Description

Fertility models for ASFR approximation

Usage

```
fert.approx(fx, age, model, start = NULL, se = FALSE, alpha = 0.05, bn = 1000)
```

Arguments

fx	Numeric vector of age specific fertility rates.
age	Numeric vector of ages.
model	Character. Model name to be estimated. Now "Hadwiger", "Gamma", "Brass" and "Beta" are supported.
start	Numeric vector with user-specific values of parameters for optimization. Default is NULL (choose automatically)
se	Logical. Should bootstrapped variance for ASFR approximation be calculated. Default is FALSE for no bootstrap.
alpha	Numeric. Used if se = TRUE, the level of uncertainty. By default, alpha = 0.05 for 95% CI.
bn	Numeric. Used if se = TRUE, number of bootstrap samples. By default, bn = 1000.

Details

This function runs least squares optimization (using default `optim`) of the selected fertility function with 1e-06 as tolerance parameter.

f_x is age-specific fertility rate for age x .

Hadwiger model:

The model is as follows:

$$f_x = \frac{ab}{c} \frac{c^{3/2}}{x} \exp[-b^2(\frac{c}{x} + \frac{x}{c} - 2)]$$

where a, b, c are estimated parameters that do not have demographic interpretation. Sometimes c is interpreted as mean age at childbearing.

Gamma model:

The model is as follows:

$$f_x = \frac{R}{\Gamma(b)c^b} (x-d)^{b-1} \exp[-(\frac{x-d}{c})]$$

where R, b, c, d are estimated parameters. Γ is gamma function. R can be interpreted as fertility level (TFR) and d as mean age at childbearing.

Brass model:

The model is as follows:

$$f_x = c(x-d)(d+w-x)$$

where c, d, w are estimated parameters.

Beta model:

The model is as follows:

$$f_x = \frac{R}{B(A, C)} (\beta - \alpha)^{-(A+C-1)} (x - \alpha)^{(A-1)} (\beta - x)^{(B-1)}$$

where $B(A, C)$ is beta function, R, β, α are estimated parameters, which can be interpreted as fertility level (TFR) and max and min age of childbearing respectively. A, C are

$$C = \left(\frac{(v - \alpha)(\beta - v)}{\tau^2} - 1 \right) \frac{\beta - v}{\beta - \alpha}$$

$$A = C \frac{v - \alpha}{v - \beta}$$

where v, τ^2 are estimated parameters, where v can be interpreted as mean age at childbearing. Thus, Beta model uses 5 parameters $R, \beta, \alpha, v, \tau^2$, where only τ^2 has no demographic interpretation.

Value

A list with two components: `model`, a list describing the fitted fertility model (type, fitted params, rmse, and, if `se = TRUE`, bootstrap covmat and parameter percentile intervals `prc`); and `predicted`, a data frame with observed and fitted age-specific fertility rates. When `se = TRUE`, `predicted` also includes bootstrap standard errors and percentile intervals.

References

Peristera, P., & Kostaki, A. (2007). Modeling fertility in modern populations. *Demographic Research*, 16, 141-194.

Examples

```
age <- seq(15, 45, 5)
fx <- c(0.03, 0.10, 0.14, 0.12, 0.07, 0.03, 0.01)
fert.approx(fx = fx, age = age, model = "Hadwiger", se = FALSE)
```

get_rosbris

Download RosBris Data

Description

Download age-specific mortality, fertility, and population data from the Russian Fertility and Mortality Database (RosBris) and return them as long-format data frames for use in `demor`.

Usage

```
get_rosbris(
  dataset = c("mortality_1", "mortality_5", "fertility_1", "fertility_5"),
  refresh = FALSE
)
```

Arguments

dataset	Character. Dataset to download and parse. One of "mortality_1", "mortality_5", "fertility_1", or "fertility_5".
refresh	Logical. If FALSE (default), previously downloaded RosBris archives are reused from the local cache directory. If TRUE, archives are downloaded again from the RosBris website.

Details

The function downloads official .zip archives from the RosBris website, stores them in a user cache directory created by `tools::R_user_dir()`, reads .txt tables from the archives, and converts them to long-format data frames.

Returned data have the following structure:

- "mortality_1": year, code, territory, sex, age, mx, N, Dx, name.
- "mortality_5": year, code, territory, sex, age, mx, N, Dx, name.
- "fertility_1": year, code, territory, age, fx, N, Bx, name.
- "fertility_5": year, code, territory, age, fx, fx1, fx2, fx3, fx4, fx5, N, Bx, Bx1, Bx2, Bx3, Bx4, Bx5, name.

At the moment, `get_rosbris()` uses the legacy RosBris series corresponding to the periods 1989–2014 and 2015–2022. Updated post-census series are not used by this function.

Value

A data frame with one row per year-region-age combination. Column structure depends on dataset; see **Details**.

Source

Russian Fertility and Mortality Database. Center for Demographic Research, Moscow (Russia). Available at <https://www.nes.ru/research-main/research-centers/demogr/demogr-fermort-data>

See Also

[rosbris.codes](#)

Examples

```
if (interactive()) {
  mort <- get_rosbris("mortality_5")
  rus2010 <- subset(
    mort,
    year == 2010 & code == 1100 & sex == "m" & territory == "t"
  )
  head(rus2010)
}
```

`gini` *Gini coefficient of a life table*

Description

Gini coefficient of a life table

Usage

```
gini(age, mx, ...)
```

Arguments

`age` Numeric array of age intervals
`mx` Numeric array with age specific mortality rates.
`...` Optional. Additional arguments for `LT()` function.

Value

A list with two components: `Gini`, itself a list with the relative Gini coefficient `G0` and the absolute Gini coefficient `G0_abs`; and `plot`, a data frame with columns `Fx` and `Phix` for drawing the Lorenz curve.

Examples

```
age <- 0:5
mx <- c(0.02, 0.01, 0.012, 0.015, 0.02, 0.03)
gini(age, mx)$Gini
```

`hli` *The Human Life Indicator (HLI)*

Description

The Human Life Indicator (HLI)

Usage

```
hli(age, mx, ...)
```

Arguments

`age` Numeric array of age intervals; for full life table = `0:100`; for concise life table = `c(0:1, seq(5, 85, 5))`
`mx` Numeric array with age specific mortality rates.
`...` Optional. Additional arguments for `LT()` function.

Details

It is calculated as

$$HLI = \prod_{x=\alpha}^{\omega} (x + a_x)^{d_x}$$

where α, ω are the first and last age groups, x is age, a_x, d_x are life table functions (s.t. $\sum_{x=\alpha}^{\omega} d_x = 1$).

Value

A length-1 numeric value giving the Human Life Indicator, i.e. the geometric mean age at death implied by the life table.

References

Ghislandi, S., Sanderson, W.C., & Scherbov, S. (2019). A Simple Measure of Human Development: The Human Life Indicator. *Population and Development Review*, 45, 219–233.

Examples

```
age <- 0:5
mx <- c(0.02, 0.01, 0.012, 0.015, 0.02, 0.03)
hli(age, mx)
```

leecart

Lee-Carter model

Description

Lee-Carter model

Usage

```
leecart(
  data,
  n = 10,
  alpha = 0.05,
  model = "RWwD",
  ax_method = "classic",
  bx_method = "classic",
  boot = FALSE,
  bn = 1000,
  ktadj = "none",
  ...
)
```

Arguments

<code>data</code>	Dataframe in the long format with the following columns: <code>age</code> , <code>year</code> , <code>m_x</code> (age specific mortality rates). For some types of <code>ktadj</code> argument <code>N</code> (population at age <code>x</code>) and <code>D_x</code> (number of deaths at age <code>x</code>) columns should also be presented.
<code>n</code>	Numeric. Forecasted horizon
<code>alpha</code>	Numeric. The level of uncertainty. By default, <code>alpha = 0.05</code> for 95% CI.
<code>model</code>	Character. Model type for <code>kt</code> forecasting. Can be "RWwD" for random walk with drift (by default, for original Lee-Carter model) or "ARIMA" for ARIMA model which parameters are chosen automatically by <code>forecast::auto.arima()</code> .
<code>ax_method</code>	Character. Method for <code>a_x</code> calculation. Can be "classic" from original Lee-Carter model (by default), "last" or "last_smooth". See details.
<code>bx_method</code>	Character. Method for <code>b_x</code> calculation. Can be "classic" from original Lee-Carter model (by default) and "rotate" for rotating <code>b_x</code> (Li et al., 2013).
<code>boot</code>	Logical. Should bootstrap estimates for uncertainty be used? FALSE by default.
<code>bn</code>	Numeric. Used if <code>boot = TRUE</code> , number of bootstrap samples. By default, <code>bn = 1000</code> .
<code>ktadj</code>	Character. Type of <code>kt</code> adjustment. It can be set to 'none' (default, no adjustment), 'Dmin', 'e0min', 'poisson' or 'edaggermin' (see Details). Note that 'Dmin' and 'poisson' require data on the age-specific number of deaths (<code>D_x</code> column in the data) and the age-specific population (<code>N</code> column in the data).
<code>...</code>	Optional. Additional arguments for <code>LT()</code> function.

Details

The `model` argument specifies the forecasting method.

- `model = "RWwD"` – classic random walk option
- `model = "ARIMA"` for selecting a more complex time series model

The `ax_method` argument allows to control how `ax` is calculated.

- `ax_method = "classic"` – classic option with the average of the logarithm of mortality rates (but there is so-called "jump-off bias").
- `ax_method = "last"` uses the logarithm of mortality for the last available year (as proposed in Lee & Miller, 2001).
- `ax_method = "last_smooth"` uses data for the last year with smoothing (see Ševčíková et al., 2016, p. 288).

The `bx_method` argument allows to control how `bx` is calculated.

- `bx_method = "classic"` for the original method.
- `bx_method = "rotate"` for the rotational variant (see Li et al., 2013).

The `ktadj` argument allows to control how `kt` is calculated.

- `ktadj = "none"` for no adjustment.

- `ktadj = "Dmin"` for minimizing the deviance of predicted/actual annual deaths (as proposed in the original Lee-Carter paper). This method requires data on the age-specific number of deaths (Dx column in the data) and the age-specific population (N column in the data).
- `ktadj = "e0min"` for minimizing the deviance of predicted/actual life expectancy (as proposed in Lee & Miller, 2001).
- `ktadj = "poisson"` for minimizing the deviance from a Poisson model, where the dependent variable is the age-specific annual number of deaths (as proposed in Booth et al., 2002). This method requires data on the age-specific number of deaths (Dx column in the data) and the age-specific population (N column in the data).
- `ktadj = "edaggermin"` for minimizing the deviance of predicted/actual edagger (see `edagger()`) as proposed in Rabbi & Mazzuco, 2021.

Value

A list with four components: `model`, the fitted time-series model used to forecast `kt`; `kt`, a data frame with historical and forecast mortality index values and confidence intervals; `ex0`, a data frame with observed, fitted, and forecast life expectancy at birth and confidence intervals; and `mx`, a data frame with observed, fitted, and forecast age-specific mortality rates and confidence intervals.

References

- Booth, H., Maindonald, J., & Smith, L. (2002). Applying Lee-Carter under conditions of variable mortality decline. *Population Studies*, 56(3), 325-336. doi:10.1080/00324720215935
- Lee, R. D., & Carter, L. R. (1992). Modeling and forecasting US mortality. *Journal of the American Statistical Association*, 87(419), 659-671. doi:10.1080/01621459.1992.10475265
- Lee, R., & Miller, T. (2001). Evaluating the performance of the Lee-Carter method for forecasting mortality. *Demography*, 38(4), 537-549. doi:10.1353/dem.2001.0036
- Li, N., Lee, R., & Gerland, P. (2013). Extending the Lee-Carter Method to Model the Rotation of Age Patterns of Mortality Decline for Long-Term Projections. *Demography*, 50(6), 2037-2051. doi:10.1007/s1352401302322
- Rabbi, A. M. F., & Mazzuco, S. (2021). Mortality forecasting with the Lee-Carter method: Adjusting for smoothing and lifespan disparity. *European Journal of Population*, 37(1), 97-120. doi:10.1007/s10680020095599
- Ševčíková, H., Li, N., Kantorová, V., Gerland, P., & Raftery, A. E. (2016). Age-Specific Mortality and Fertility Rates for Probabilistic Population Projections. In R. Schoen (Ed.), *Dynamic Demographic Analysis* (Vol. 39, pp. 285-310). Springer International Publishing. doi:10.1007/9783319266039_15

Examples

```
age <- 0:20
year <- 2000:2009
lc_data <- expand.grid(age = age, year = year)
lc_data$mx <- exp(-7 + 0.09 * lc_data$age - 0.02 * (lc_data$year - min(year)))
fit <- leecart(lc_data, n = 2)
head(fit$kt)
```

 leslie

Leslie Matrix

Description

Leslie Matrix

Usage

```
leslie(mx, fx, age.mx, age.fx, srb = 100/205, fin = TRUE, ...)
```

Arguments

mx	Numeric array of age specific mortality rates.
fx	Numeric array of age specific fertility rates.
age.mx	Numeric array of ages for mx.
age.fx	Numeric array of ages for fx.
srb	Numeric. Sex ratio at birth. Usually it is assumed that for males it is 105/205, for females it is 100/205. By default, it is 100/205.
fin	Logical. Should the survival rate for the last age-group be nonzero? By default it is FALSE, so the last survival rate is 0 as in classical model. Otherwise, it is T_x/T_{x-1} .
...	Optional. Additional arguments for <code>LT()</code> function.

Value

A square numeric matrix of class `leslie` with one row and one column per age group. The first row contains fertility contributions and the subdiagonal contains survival ratios.

See Also

`summary.leslie()` for leslie output that calculates λ , r , w and v .

Examples

```
mx <- c(0.02, 0.01, 0.012, 0.015, 0.02)
fx <- c(0.05, 0.08)
leslie(mx = mx, fx = fx, age.mx = 0:4, age.fx = 1:2)
```

LT *Life table*

Description

Life table

Usage

LT(age, sex = "m", mx, ax = NULL, w = NULL, l0 = 1)

Arguments

age	Numeric array of age intervals; for full life table = 0:100; for concise life table = c(0:1, seq(5, 85, 5))
sex	Character. Sex. "m" for males or "f" for females. By default = "m".
mx	Numeric array with age specific mortality rates.
ax	Optional. Numeric array with ax. By default, it is the middle of the interval, while ax for age 0 is modeled as in Andreev & Kingkade (2015).
w	Optional. Numeric array with weights for each age interval for calculating weighted life expectancy (wex).
l0	Numeric. Life table radix. By default, = 1 but it can be any positive real number. In "human" demography tradition it is 100'000, in "ecological" and "evolutionary" demography tradition it is 1.

Details

By default, a_x for age 0 (first entity in ax) is modeled as in Andreev & Kingkade (2015, p. 390, see table 3-2).

The weighted life expectancy is calculated as follows:

$$e_x^w = \frac{\sum_{i=x}^{\omega} L_i w_i}{l_x}$$

where ω is the last age, w is weight s.t. $w \in [0, 1]$, and other variables are life table functions.

Value

A numeric matrix with one row per age group. Standard columns are age, mx, ax, qx, lx, dx, Lx, Tx, and ex. If w is supplied, additional columns w, wLx, and wex are appended, where wex is weighted life expectancy.

References

Andreev, E. M., & Kingkade, W. W. (2015). Average age at death in infancy and infant mortality level: Reconsidering the Coale-Demeny formulas at current levels of low mortality. *Demographic Research*, 33, 363-390.

See Also

[MLT\(\)](#) for Multiple Decrement Life Table.

Examples

```
# Minimal toy example
age <- 0:5
mx <- c(0.02, 0.01, 0.012, 0.015, 0.02, 0.03)
LT(age = age, sex = "m", mx = mx)

if (interactive()) {
  # Real RosBris data via get_rosbris(): Russian males, 2010
  rus2010 <- subset(get_rosbris("mortality_5"),
    year == 2010 & code == 1100 & sex == "m" & territory == "t"
  )
  LT(age = rus2010$age, sex = "m", mx = rus2010$mx)
}
```

 mac

Mean Age at Childbearing (MAC)

Description

Mean Age at Childbearing (MAC)

Usage

```
mac(fx, age)
```

Arguments

fx Numeric array of age specific fertility rates.
 age Numeric array of ages. For example, 15:55 for 1-year age-groups

Value

A length-1 numeric value giving the mean age at childbearing implied by fx.

Examples

```
age <- seq(15, 45, 5)
fx <- c(0.03, 0.10, 0.14, 0.12, 0.07, 0.03, 0.01)
mac(fx, age)
```

mdecomp

*Age and cause decomposition of differences in life expectancies***Description**

Age and cause decomposition of differences in life expectancies

Usage

```
mdecomp(mx1, mx2, age, method = "andreev", ...)
```

Arguments

mx1	List of numeric arrays. 1st array should be all-cause nmx in the 1st population, other arrays are cause-specific nmx in the 1st population
mx2	List of numeric arrays. 1st array should be all-cause nmx in the 2nd population, other arrays are cause-specific nmx in the 2nd population
age	Numeric array of age intervals; for full life table = 0:100; for concise life table = c(0:1, seq(5, 85, 5))
method	Character. Decomposition method. "andreev" (1982) or "arriaga" (1984) - slightly different in their results. By default, method = "andreev".
...	Optional. Additional arguments for decomp() .

Details

The contribution of each cause c to the absolute difference in life expectancies between the first and second population is calculated as

$$\Delta_{x,c} = \frac{m_{x,c}^1 - m_{x,c}^2}{m_x^1 - m_x^2} \times \Delta_x$$

where Δ_x is contribution of age x to difference $e_0^2 - e_0^1$ from function [decomp\(\)](#), $m_{x,c}^i$ is age-specific mortality rate for population i from cause c , and m_x^i is total age-specific mortality rate.

Value

A data frame of class `c("mdecomp", "data.frame")` with one row per age group. Column `age` contains ages, `ex12` contains the overall age contribution to the life-expectancy difference, and the remaining columns contain cause-specific contributions named after the cause-specific elements of `mx1` and `mx2`.

See Also

[decomp\(\)](#) for just age decomposition and [plot.mdecomp\(\)](#) for graph of mdecomp results

Examples

```

data(mdecompex)
usa <- subset(mdecompex, cnt == "usa")
eng <- subset(mdecompex, cnt == "eng")
dec <- mdecomp(
  mx1 = list(
    all = usa$all,
    neoplasms = usa$neoplasms,
    circulatory = usa$circulatory
  ),
  mx2 = list(
    all = eng$all,
    neoplasms = eng$neoplasms,
    circulatory = eng$circulatory
  ),
  age = usa$age
)
dec[1:3, ]

```

mdecompex

Data on mortality of US and England and Wales men in 2002 by some causes

Description

Data on mortality of US and England and Wales men in 2002 by some causes

Usage

mdecompex

Format**A dataframe:**

age age

neoplasms age specific mortality rate (asmr) from Neoplasms

circulatory asmr from Diseases of the circulatory system

respiratory asmr from Diseases of the respiratory system

digestive asmr from Diseases of the digestive system

other asmr from other causes

all overall asmr

cnt country: usa - US, eng - England and Wales

Source

Andreev & Shkolnikov spreadsheet, can be seen on this [webpage](#)

med.age *Median age calculation*

Description

Median age calculation

Usage

```
med.age(N, age)
```

Arguments

N Numeric array. Population counts by age groups (from young to old)

age Numeric array. Lower bounds of age groups, same length as N

Value

A length-1 numeric value giving the estimated median age of the population represented by N.

Examples

```
N <- c(100, 90, 80, 70, 60)
age <- seq(0, 20, 5)
med.age(N, age)
```

MLT *Multiple Decrement Life Table*

Description

Multiple Decrement Life Table

Usage

```
MLT(age, mx, ...)
```

Arguments

age Numeric array of age intervals; for full life table = 0:100; for concise life table = c(0:1, seq(5, 85, 5))

mx List of numeric arrays. 1st array should be all-cause mx in the population, other arrays are cause-specific mx in the population.

... Other parameters for the function `LT()` including `ax` (by default, the middle of the interval), `sex` (by default = "m" - males), `l0` (by default = 1).

Value

A numeric matrix extending the output of `LT()`. In addition to the standard life-table columns, for each cause `i` in `mx[-1]` it adds `qx_i`, `dx_i`, `lx_i`, and `ex_no_i`, corresponding to cause-specific death probabilities, deaths, survivors, and cause-deleted life expectancy.

See Also

`LT()` for usual life table calculation

Examples

```
data(asdtex)
mx_causes <- list(
  all = asdtex$all,
  neoplasms = asdtex$neoplasms,
  circulatory = asdtex$circulatory
)
MLT(age = asdtex$age, mx = mx_causes)[1:3, c("age", "mx", "qx_neoplasms", "ex_no_neoplasms")]
```

 mort.approx

Mortality models for mx approximation

Description

Mortality models for mx approximation

Usage

```
mort.approx(mx, age, model = c("Brass", "Gompertz"), standard.mx = NULL, ...)
```

Arguments

<code>mx</code>	Numeric vector of age specific mortality rates.
<code>age</code>	Numeric vector of ages.
<code>model</code>	Character. Model name to be estimated. Now "Gompertz" and "Brass" are supported.
<code>standard.mx</code>	Numeric vector of age specific mortality rates for standard population. Default is NULL.
<code>...</code>	Used only for Brass model. Other parameters for the function <code>LT()</code> including <code>ax</code> (by default, the middle of the interval), <code>sex</code> (by default = "m" - males), <code>l0</code> (by default = 1).

Details

This function runs least squares optimization of the selected mortality function using Gauss-Newton algorithm with 2000 maximum iterations and 1e-07 as tolerance parameter. For "Gompertz" usual OLS estimator is used.

Gompertz model:

The model is as follows:

$$m(age) = \alpha e^{\beta age}$$

Brass model:

The model is as follows:

$$y(age) = \alpha + \beta y^S(age)$$

where

$$y(age) = \frac{1}{2} \ln \left[\frac{q(age)}{1 - q(age)} \right]$$

and subscript S defines that function is for standard population. To get mx from qx usual formula is used:

$$m(age) = \frac{q(age)}{n - q(age)(n - a(age))}$$

where n is the size of age interval and a(x) is a parameter from life table.

Value

A list with two components: model, the fitted lm or nls object; and predicted, a data frame with columns age and mx.pred containing the fitted mortality schedule.

References

Preston, S. H., Heuveline, P., & Guillot, M. (2001). Demography: Measuring and Modeling Population Processes. Blackwell Publishers.

Examples

```
age <- seq(40, 80, 10)
mx <- c(0.003, 0.005, 0.009, 0.018, 0.036)
standard.mx <- c(0.0025, 0.004, 0.007, 0.014, 0.03)
mort.approx(mx = mx, age = age, model = "Gompertz")
mort.approx(mx = mx, age = age, model = "Brass", standard.mx = standard.mx)
```

`plot.mdecomp`*Plot for mdecomp function*

Description

Plot for mdecomp function

Usage

```
## S3 method for class 'mdecomp'  
plot(x, return.data = FALSE, ...)
```

Arguments

<code>x</code>	A result of age and cause decomposition from <code>mdecomp()</code> .
<code>return.data</code>	Should the data be returned (TRUE) or the ggplot2 plot (default, FALSE).
<code>...</code>	Ignored.

Value

If `return.data = FALSE`, a ggplot2 object with stacked bars of age- and cause-specific contributions. If `return.data = TRUE`, a data frame with columns `age`, `ex12`, and `group` used to build the plot.

See Also

[mdecomp\(\)](#)

Examples

```
data(mdecompex)  
usa <- subset(mdecompex, cnt == "usa")  
eng <- subset(mdecompex, cnt == "eng")  
dec <- mdecomp(  
  mx1 = list(  
    all = usa$all,  
    neoplasms = usa$neoplasms,  
    circulatory = usa$circulatory  
  ),  
  mx2 = list(  
    all = eng$all,  
    neoplasms = eng$neoplasms,  
    circulatory = eng$circulatory  
  ),  
  age = usa$age  
)  
plot(dec)
```

plot_pyr *Plot population pyramid*

Description

Plot population pyramid

Usage

```
plot_pyr(
  popm,
  popf,
  popm2 = NULL,
  popf2 = NULL,
  age,
  prc = FALSE,
  sexn = c("Males", "Females"),
  sexc = c("#ED0000B2", "#00468BB2"),
  age.cont = NULL,
  un.intervals = TRUE
)
```

Arguments

popm	Numeric vector of male population
popf	Numeric vector of female population
popm2	Optional. Numeric vector of second male population to be plotted as line.
popf2	Optional. Numeric vector of second female population to be plotted as line.
age	Numeric vector of ages
prc	Should the population be shown as a percentage rather than in absolute terms? By default, it is FALSE and the plot shows user-specific values in popm and popf.
sexn	Character array of labels for the sexes. By default = c("Males", "Females")
sexc	Character array of colors for the sexes. By default = c("#ED0000B2", "#00468BB2"), which are red and blue
age.cont	Optional. Logical. Should the age axis be considered as continuous scale? Recommend to switch to TRUE if the age intervals are small (1 year) and FALSE otherwise (for ex., when age interval is 5 years). By default function chooses the value by itself.
un.intervals	Logical. Should the age groups be unified? For example, if the first and second groups are 0-1, 1-4, and all other groups are five-year, the function automatically makes the group 1-4 to make all the intervals the same. By default, TRUE.

Value

A ggplot2 object representing a population pyramid. If popm2 and popf2 are supplied, the plot also includes dashed comparison lines for the second population.

Examples

```
plot_pyr(
  popm = c(100, 90, 80, 70, 60),
  popf = c(95, 92, 85, 75, 65),
  age = seq(0, 20, 5)
)
```

rosbris.codes	<i>Rosbris' Region codes</i>
---------------	------------------------------

Description

Rosbris' Region codes

Usage

rosbris.codes

Format**A dataframe:**

n number of Region (sequence number)

name Region name

code Region unique code

Source

Russian Fertility and Mortality Database. Center for Demographic Research, Moscow (Russia). Available at <https://www.nes.ru/demogr-fermort-data>

sle_stand	<i>Data on Standard Life expectancies that is used for YLL calculations</i>
-----------	---

Description

Data on Standard Life expectancies that is used for YLL calculations

Usage

sle_stand

Format**A dataframe:**

stand Standard: 1 - World Health Organization Standard Life Expectancy by single-age; 2 - Global Burden of Disease studies (GBD) and WHO Global Health Estimates (WHO GHE) Standard Life Expectancy by 5-year age groups.

age age

ex Standard Life Expectancy

Source

Martinez, R., Soliz, P., Caixeta, R., Ordunez, P. (2019). Reflection on modern methods: years of life lost due to premature mortality-a versatile and comprehensive measure for monitoring non-communicable disease mortality. *International Journal of Epidemiology*, 48, 1367-1376. doi:10.1093/ije/dyy254

 summary.leslie

Leslie matrix summary

Description

Leslie matrix summary

Usage

```
## S3 method for class 'leslie'
summary(object, d = 1, ...)
```

Arguments

object	A leslie matrix from <code>leslie()</code> .
d	Time step. By default it is 1. Only affects <i>r</i> .
...	Ignored.

Details

The function calculates λ , r , w and v .

- λ – asymptotic growth factor that is dominant eigenvalue.
- r – asymptotic growth rate that is $\ln\lambda/\delta t$.
- w – stable age distribution normalized to 1 s.t. $\sum_x w_x = 1$ where x is age.
- v – reproductive values normalized s.t. $v'w = 1$.

Value

A list of class `summary.leslie` with four components: `lambda` (dominant eigenvalue / asymptotic growth factor), `r` (intrinsic growth rate), `w` (stable age distribution summing to 1), and `v` (reproductive values normalized so that $\text{sum}(v * w) = 1$).

See Also[leslie\(\)](#)**Examples**

```
mx <- c(0.02, 0.01, 0.012, 0.015, 0.02)
fx <- c(0.05, 0.08)
les <- leslie(mx = mx, fx = fx, age.mx = 0:4, age.fx = 1:2)
summary(les)
```

tatfr	<i>Tempo-adjusted total fertility rate (TFR')</i>
-------	---

Description

Tempo-adjusted total fertility rate (TFR')

Usage

```
tatfr(past_fx, present_fx, post_fx, age)
```

Arguments

past_fx	List with numeric arrays of age specific fertility rates for period t-1 by parity
present_fx	List with numeric arrays of age specific fertility rates for period t by parity (it is period of interest)
post_fx	List with numeric arrays of age specific fertility rates for period t+1 by parity
age	Array with numeric values age

Details

This indicator is calculated as follows

$$TFR'_{i,t} = \frac{TFR_{i,t}}{1 - (M_{i,t+1} - M_{i,t-1})/2}$$

where $TFR'_{i,t}$, $TFR_{i,t}$ are tempo-adjusted and usual total fertility rate for parity i and time t respectively, $M_{i,t}$ is mean age at childbearing for parity i and time t . The tempo-adjusted total fertility rate is a sum of parity-specific TFR'_i .

Note, the calculation are done as in footnote 1 in (Bongaarts & Feeney, 2000, p. 563). Unfortunately, the original 1998 article does not provide the exact formula, which has caused some confusion in academic circles.

Value

A list with four components: tatfr (overall tempo-adjusted total fertility rate), tatfr_i (parity-specific tempo-adjusted rates), tfr (overall conventional TFR), and tfr_i (parity-specific conventional rates).

References

Bongaarts, J., & Feeney, G. (1998). On the Quantum and Tempo of Fertility. *Population and Development Review*, 24(2), 271–291. doi:10.2307/2807974

Bongaarts, J., & Feeney, G. (2000). On the Quantum and Tempo of Fertility: Reply. *Population and Development Review*, 26(3), 560–564. doi:10.1111/j.17284457.2000.00560.x

See Also

`tfr()` for TFR and `mac()` for mean age at childbearing calculation.

Examples

```
age <- seq(15, 45, 5)
past_fx <- list(
  c(0.02, 0.05, 0.07, 0.05, 0.03, 0.01, 0.00),
  c(0.01, 0.03, 0.04, 0.03, 0.02, 0.01, 0.00)
)
present_fx <- list(
  c(0.03, 0.06, 0.08, 0.06, 0.03, 0.01, 0.00),
  c(0.01, 0.03, 0.05, 0.04, 0.02, 0.01, 0.00)
)
post_fx <- list(
  c(0.03, 0.05, 0.08, 0.07, 0.04, 0.02, 0.00),
  c(0.01, 0.03, 0.04, 0.04, 0.03, 0.01, 0.00)
)
tatfr(past_fx, present_fx, post_fx, age)
```

tfr

Total Fertility Rate (TFR)

Description

Total Fertility Rate (TFR)

Usage

```
tfr(fx, age.int = 1)
```

Arguments

fx	Numeric array of age specific fertility rates.
age.int	Numeric. Age group: 1 for one-year, 5 for five-year. Any age groups are allowed.

Value

A length-1 numeric value equal to the sum of age-specific fertility rates multiplied by `age.int`.

Examples

```
fx <- c(0.02, 0.08, 0.12, 0.09, 0.04, 0.01, 0.001)
tfr(fx, age.int = 5)
```

yll *Years of Life Lost (YLL) calculation*

Description

Years of Life Lost (YLL) calculation

Usage

```
yll(
  Dx,
  type = c("yll", "yll.p", "yll.r", "asyr"),
  age.int = 5,
  Dx_all = NULL,
  pop = NULL,
  w = NULL,
  standard = NULL
)
```

Arguments

Dx	Array with the number of deaths.
type	Character. Type of YLL to calculate. See details section.
age.int	Numeric. Age interval of Dx. Can be 1 or 5.
Dx_all	Array with the number of all deaths. Used only with yll.p type, where Dx is array with cause-specific deaths.
pop	Array with population. Used only with yll.r and asyr types.
w	Array with population weights for direct standardization. Used only with asyr type.
standard	Data frame. User-specific standard life expectancy to calculate YLL with the following columns: age, ex. Note: the age.int argument should be consistent with the age column in this data frame.

Details

Computes four types of Years of Life Lost (YLL) indicators:

- **Absolute YLL** (type = "yll"):

$$YLL_{x,t,c} = D_{x,t,c} \times SLE_x$$

where x, t, c are age, time, and cause respectively, $D_{x,t,c}$ is deaths in age x at time t from cause c and SLE_x is standard life expectancy at age x

- **YLL proportion** (type = "y11.p"):

$$YLL_{x,t,c}^p = \frac{YLL_{x,t,c}}{YLL_{x,t}}$$

where $YLL_{x,t} = \sum_c YLL_{x,t,c}$ (total YLL across causes)

- **YLL rate** (type = "y11.r"):

$$YLL_{x,t,c}^r = \left(\frac{YLL_{x,t,c}}{N_{x,t}} \right) \times 100'000$$

where $N_{x,t}$ is population in age group x at time t

- **Age-standardized YLL rate** (type = "asyr"):

$$ASYR_{t,c} = \sum_{x=\alpha}^{\omega} (YLL_{x,t,c}^r \times w_x)$$

where α to ω corresponds to first to last age group, w_x is standard population weight for age x .

Value

A list whose components depend on type. For type = "y11", the list contains y11_all (overall YLL) and y11 (age-specific YLL). For type = "y11.p", it contains y11.p_all and y11.p (overall and age-specific YLL proportions). For type = "y11.r", it contains y11.r_all and y11.r (overall and age-specific YLL rates). For type = "asyr", it contains asyr_all and asyr (overall and age-specific age-standardized YLL rates).

References

Martinez, R., Soliz, P., Caixeta, R., & Ordunez, P. (2019). Reflection on modern methods: years of life lost due to premature mortality—a versatile and comprehensive measure for monitoring non-communicable disease mortality. *International Journal of Epidemiology*, 48, 1367–1376.

Examples

```
y11(Dx = rep(1, 19), type = "y11", age.int = 5)$y11_all
```

Index

- * **datasets**
 - asdtex, 4
 - mdecompex, 20
 - rosbris.codes, 26
 - sle_stand, 26
- ages, 2
- asdt, 3
- asdtex, 4
- ccm, 5
- decomp, 6
- decomp(), 19
- edagger, 8
- edagger(), 15
- fert.approx, 8
- get_rosbris, 10
- gini, 12
- hli, 12
- leecart, 13
- leslie, 16
- leslie(), 5, 27, 28
- LT, 17
- LT(), 3, 8, 12, 14, 16, 21, 22
- mac, 18
- mac(), 29
- mdecomp, 19
- mdecomp(), 7, 24
- mdecompex, 20
- med.age, 21
- MLT, 21
- MLT(), 18
- mort.approx, 22
- plot.mdecomp, 24
- plot.mdecomp(), 19
- plot_pyr, 25
- rosbris.codes, 11, 26
- sle_stand, 26
- summary.leslie, 27
- summary.leslie(), 16
- tatfr, 28
- tfr, 29
- tfr(), 29
- y11, 30