Package ‘LEdecomp’

November 11, 2025

Type Package

Title Decompose Life Expectancy by Age (and Cause)

Version 1.0.4

Date 2025-11-06

Description A set of all-cause and cause-specific life expectancy sensitivity and decomposition methods, including Arriaga (1984) <doi:10.2307/2061029>, others documented by Ponnapalli (2005) <doi:10.4054/DemRes.2005.12.7>, lifetable, numerical, and other algorithmic approaches such as Horiuchi et al (2008) <doi:10.1353/dem.0.0033>, or Andreev et al (2002) <doi:10.4054/DemRes.2002.7.14>.

License GPL-3

LazyLoad yes

LazyData true

RoxygenNote 7.3.2

Imports numDeriv, Rdpack, DemoDecomp, data.table, ggplot2

Depends R (>= 4.3)

RdMacros Rdpack

BugReports `https://github.com/timriffe/LEdecomp/issues`

Encoding UTF-8

URL `https://github.com/timriffe/LEdecomp`

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation no

Author Tim Riffe [aut, cre] (0000-0002-2673-4622),
David Atance [aut] (0000-0001-5860-0584),
Josep Lledo [aut] (0000-0002-7475-8549)

Maintainer Tim Riffe <`tim.riffe@gmail.com`>

Repository CRAN

Date/Publication 2025-11-11 10:10:02 UTC

Contents

`abridge_mx` *Abridge a single-year mx schedule to 0,1,5,... using lifetable quantities*

Description

Abridge a single-year mx schedule to 0,1,5,... using lifetable quantities

Usage

```
abridge_mx(mx, age, sex = "t", closeout = TRUE)
```

Arguments

`mx`	numeric vector of single-year mortality rates (ages 0,1,2,...)
`age`	numeric vector of the same length as mx, usually 0:(n-1)
`sex`	character, passed to mx_to_ax(), default "t"
`closeout`	logical, passed to lifetable helpers, default TRUE

Value

numeric vector of abridged mx at ages c(0, 1, 5, 10, ...)

`ald_to_Lx` *Calculate the lifetable exposure*

Description

`Lx` is defined as the integration of `lx` in the interval `[x,x+n)`, where `n` is the width of the interval. There are many approximations for this. Here we use HMD Method Protocol equation 78. You can think of `Lx` as lifetable exposure, or person-years lived in each age interval.

Usage

```
ald_to_Lx(ax, lx, dx, nx)
```

Arguments

`ax`	numeric vector of ax, average time spent in the age interval by those that die in the interval
`lx`	numeric vector of lx, lifetable survivorship at exact ages.
`dx`	numeric vector of dx, the lifetable deaths distribution.
`nx`	age interval width, assumes 1 by default

Value

numeric vector of `Lx` values

References

Wilmoth JR, Andreev K, Jdanov D, Glei DART, Boe C, Bubenheim M, Philipov D, Shkolnikov V, Vachon P, Winant C, M B (2021). “Methods protocol for the human mortality database. Version 6.” *University of California, Berkeley, and Max Planck Institute for Demographic Research, Rostock. URL: http://mortality.org [version 26/01/2021].*

`arriaga` *classic Arriaga decomposition*

Description

Following the notation given in Preston et al (2000), Arriaga’s decomposition method can written as:

$${}_n\Delta_x = \frac{l_x^1}{l_0^1} \cdot \left(\frac{{}_nL_x^2}{l_x^2} - \frac{{}_nL_x^1}{l_x^1} \right) + \frac{T_{x+n}^2}{l_0^1} \cdot \left(\frac{l_x^1}{l_x^2} - \frac{l_{x+n}^1}{l_{x+n}^2} \right)$$

Usage

```
arriaga(
  mx1,
  mx2,
  age = 0:(length(mx1) - 1),
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

A little-known property of this decomposition method is that it is directional, in the sense that we are comparing a movement of `mx1` to `mx2`, and this is not exactly symmetrical with a comparison of `mx2` with `mx1`. Note also, if decomposing in reverse from the usual, you may need a slight adjustment to the closeout value in order to match sums properly (see examples for a demonstration).

setting `closeout` to `TRUE` will result in value of `1/mx` for the final age group, of `ax` and a value of 1 for the closeout of `qx`.

Value

`cc` numeric vector with one element per age group, and which sums to the total difference in life expectancy between population 1 and 2.

References

Arriaga EE (1984). “Measuring and explaining the change in life expectancies.” *Demography*, **21**, 83–96. Preston S, Heuveline P, Guillot M (2000). *Demography: measuring and modeling population processes*. Wiley-Blackwell.

Examples

```
a <- .001
b <- .07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
cc <- arriaga(mx1, mx2, age = x)
e01 <- mx_to_e0(mx1, age = x)
e02 <- mx_to_e0(mx2, age = x)
(delta <- e02 - e01)
sum(cc)


 plot(x, cc)

# asymmetrical with a decomposition in the opposite direction
cc2 <- -arriaga(mx1 = mx2, mx2 = mx1, age = x)
plot(x, cc)
lines(x,cc2)
# also we lose some precision?
sum(cc);sum(cc2)
# found it!
delta-sum(cc2); cc2[length(cc2)] / 2

# But this is no problem if closeout = FALSE
-arriaga(mx1 = mx2, mx2 = mx1, age = x,closeout=FALSE) |> sum()
arriaga(mx1 = mx1, mx2 = mx2, age = x,closeout=FALSE) |> sum()
```

`arriaga_sym` *Estimate sensitivity of life expectancy using a symmetrical Arriaga approach.*

Description

This approach conducts a classic Arriaga decomposition in both directions, averaging the (sign-adjusted) result, i.e. `a_avg = (arriaga(mx1,mx2, ...) - arriaga(mx2, mx1, ...)) / 2`. #@note The final age group's contribution from the reversed decomposition is halved before averaging. This empirical adjustment ensures symmetry and numeric stability, though the theoretical basis requires further exploration.

Usage

```
arriaga_sym(
  mx1,
  mx2,
  age = 0:(length(mx1) - 1),
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Value

numeric vector of contributions summing to the gap in life expectancy implied by `mx1` and `mx2`.

See Also

`arriaga`

Examples

```
a <- .001
b <- .07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
d <- arriaga_sym(mx1, mx2, age = x)

e01 <- mx_to_e0(mx1,age=x)
e02 <- mx_to_e0(mx2,age=x)
(Delta <- e02 - e01)
sum(d)

d12 <- arriaga(mx1, mx2, age = x)
d21 <- arriaga(mx2, mx1, age = x) # direction opposite

plot(x, d, type= 'l')
  lines(x, d12, col = "red")
  lines(x, -d21, col = "blue")
```

`available_methods` *List available decomposition methods*

Description

Returns a table of all implemented methods, their function name, and category.

Usage

```
available_methods(category = NULL)
```

Arguments

`category`	character. one of `"direct"`, `"direct_sen"`, `"opt_ok"`, or `"general"`

Value

A data frame of available decomposition methods.

Examples

```
available_methods()
```

`chandrasekaran_II` *II approach of Chandrasekaran decomposition approach*

Description

Following the notation given in Ponnapalli (2005), and the decomposition method can written as:

$${}_n\Delta_x = \frac{(e_x^2 - e_x^1)(l_x^2 + l_x^1)}{2} - \frac{(e_{x+n}^2 - e_{x+n}^1)(l_{x+n}^2 + l_{x+n}^1)}{2} - \frac{{}_nL_x^1}{l_x^1} + \frac{T_{x+n}^2}{l_0^1}\left(\frac{l_x^1}{l_x^2} - \frac{l_{x+n}^1}{l_{x+n}^2}\right)$$

where ${}_n\Delta_x$ is the contribution of rate differences in age x to the difference in life expectancy implied by `mx1` and `mx2`. This formula can be averaged between 'effect interaction deferred' and 'effect interaction forwarded' from the Ponnapalli (2005).

Usage

```
chandrasekaran_II(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

setting `closeout` to `TRUE` will result in value of `1/mx` for the final age group, of `ax` and a value of 1 for the closeout of `qx`. This function gives numerically identical results to `arriaga_sym()`, `lopez_ruzicka_sym()`, and `chandrasekaran_III()`.

Value

`cc` numeric vector with one element per age group, and which sums to the total difference in life expectancy between population 1 and 2.

References

Arriaga EE (1984). "Measuring and explaining the change in life expectancies." *Demography*, **21**, 83–96. Preston S, Heuveline P, Guillot M (2000). *Demography: measuring and modeling population processes*. Wiley-Blackwell. Ponnapalli KM (2005). "A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth." *Demographic Research*, **12**, 141–172.

Examples

```
a <- .001
b <- .07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
cc <- chandrasekaran_II(mx1, mx2, age = x)
e01 <- mx_to_e0(mx1, age = x)
e02 <- mx_to_e0(mx2, age = x)
(delta <- e02 - e01)
sum(cc)


 plot(x, cc)
```

`chandrasekaran_III` *Chandrasekaran III decomposition*

Description

Implements the Chandrasekaran III decomposition as described in Ponnapalli (2005), which combines multiple directional effects into a symmetric average. The method constructs a decomposition of the difference in life expectancy into four parts: the main effect, the operative effect, their average (exclusive effect), and a non-linear interaction term. These are calculated based on life table values. Let e_x^i denote remaining life expectancy at age x for population i, and l_x^i the number of survivors to age x. Then:

- **Main effect**:
$$\frac{l_x^1}{l_x^2}\left[l_x^2(e_x^2 - e_x^1) - l_{x+n}^2(e_{x+n}^2 - e_{x+n}^1)\right]$$
- **Operative effect**:
$$\frac{l_x^2}{l_x^1}\left[l_x^1(e_x^2 - e_x^1) - l_{x+n}^1(e_{x+n}^2 - e_{x+n}^1)\right]$$

- **Exclusive effect**:

$$\frac{\text{Main effect} + \text{Operative effect}}{2}$$

- **Interaction effect**:

$$(e^2_{x+n} - e^1_{x+n}) \cdot \frac{1}{2}\left[\frac{l^1_x \cdot l^2_{x+n}}{l^2_x} + \frac{l^2_x \cdot l^1_{x+n}}{l^1_x} - (l^1_{x+n} + l^2_{x+n})\right]$$

The final contribution by age group is the sum of exclusive and interaction effects.

Usage

```
chandrasekaran_III(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This decomposition gives numerically identical results to `arriaga_sym()`, `lopez_ruzicka_sym()`, and `chandrasekaran_II()`, despite conceptual differences in their derivation. Included here for methodological completeness.

Value

Numeric vector of contributions by age group that sum to the total difference in life expectancy between the two mortality schedules.

References

Ponnapalli KM (2005). “A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth.” *Demographic Research*, **12**, 141–172.

See Also

`chandrasekaran_II`, `arriaga_sym`, `lopez_ruzicka_sym`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
cc <- chandrasekaran_III(mx1, mx2, age = x)
e01 <- mx_to_e0(mx1, age = x)
e02 <- mx_to_e0(mx2, age = x)
(delta <- e02 - e01)
sum(cc)

plot(x, cc, type = "l")
```

`LEdecomp` *Function for applying different Life-Expectancy decomposition and sensitivity methods*

Description

A variety of exact or asymptotically exact life expectancy decomposition methods are implemented. Also, several life-expectancy decomposition sensitivity methods are implemented to answer how each age will change with an increase/decrease in life expectancy. See the package README and references for details.

Usage

```
LEdecomp(
  mx1,
  mx2,
  age = NULL,
  nx = NULL,
  n_causes = NULL,
  cause_names = NULL,
  sex1 = "t",
  sex2 = sex1,
 method = c("lifetable", "arriaga", "arriaga_sym", "sen_arriaga", "sen_arriaga_sym",
```

```
    "sen_arriaga_inst", "sen_arriaga_inst2", "sen_arriaga_sym_inst",
    "sen_arriaga_sym_inst2", "chandrasekaran_ii", "sen_chandrasekaran_ii",
   "sen_chandrasekaran_ii_inst", "sen_chandrasekaran_ii_inst2", "chandrasekaran_iii",
    "sen_chandrasekaran_iii", "sen_chandrasekaran_iii_inst",
    "sen_chandrasekaran_iii_inst2", "lopez_ruzicka", "lopez_ruzicka_sym",
    "sen_lopez_ruzicka", "sen_lopez_ruzicka_sym", "sen_lopez_ruzicka_inst",
    "sen_lopez_ruzicka_inst2",
     "horiuchi", "stepwise", "numerical"),
  closeout = TRUE,
  opt = TRUE,
  tol = 1e-10,
  Num_Intervals = 20,
  symmetrical = TRUE,
  direction = "both",
  perturb = 1e-06,
  ...
)
```

Arguments

`mx1`	numeric. Age-structured mortality rates for population 1 (vector, matrix, or data.frame). See Details section for more info.
`mx2`	numeric. Age-structured mortality rates for population 2 (same shape as `mx1`).
`age`	integer. Lower bound of each age group. If `NULL`, it will be inferred from data (see Details).
`nx`	integer vector of age intervals (defaults to 1 when missing).
`n_causes`	integer or `NULL`. If provided with stacked vectors, forces the number of causes (columns).
`cause_names`	optional character vector of length `n_causes` giving labels for causes. Alternatively detected from `colnames(mx1)` in case given as a `matrix` or `data.frame`
`sex1`	character. `"m"`,`"f"`, or `"t"`, affects a0 treatment.
`sex2`	character. `"m"`,`"f"`, or `"t"`, affects a0 treatment.
`method`	character. One of the methods in `method_registry$method`.
`closeout`	logical. Close out at top age (`TRUE`) or assume closed final age group (`FALSE`).
`opt`	logical. For lifetable, numerical, and instantaneous sensitivity-based methods, optimize rate averaging to eliminate the decomposition residual?
`tol`	numeric. Tolerance for rate-averaging optimization.
`Num_Intervals`	integer. For methods that discretize an integral (e.g., Horiuchi).
`symmetrical`	logical. For stepwise replacement only: average 1 to 2 and 2 to 1?
`direction`	character. For stepwise replacement: "up", "down", or "both".
`perturb`	numeric. Small perturbation for numerical derivatives.
`...`	optional arguments passed to `numDeriv::grad()` or other internals #' @details Input dimensions are flexible to accommodate different coding styles and data layouts: **Accepted forms of `mx1` and `mx2`:**

- **Vector:** A single all-cause mortality schedule, one value per age. In this case `age` must be the same length, or can be omitted and will default to `0:(length(mx1)-1)`, unless we detect you might be using abridged age groups.
- **Matrix:** Rows represent ages, columns represent causes of death. Row names, if any, and if numeric, are interpreted as ages and override the supplied `age` argument or our inferences. Column names are retained in the output for clarity.
- **Data frame (wide):** Same layout as a matrix, with an optional column named `age`.
- **Stacked vector:** A long, concatenated vector representing causes stacked on top of each other (i.e., all ages for cause 1, then all ages for cause 2, and so on). If you don't specify age, we try to detect age and the number of causes. But please specify age- it could be stacked also, or not! For example, when used inside a tidy pipeline you might write `mutate(LEdecomp(mx1, mx2, age))` where `age` is repeated for each cause, i.e. the code might look the same as if you were dealing with all-cause data. But in that case be careful data are ordered consistently.

Age detection and inference:

- If `age` is supplied explicitly, it is used as given.
- If missing, `LEdecomp()` attempts to infer it from (in order): row names, names of the input vector, a column named `"age"` in a data frame, or heuristics for single-year (0:100) or abridged (0,1,5,10,. . .) schedules.
- If `age` is repeated (e.g., `c(0:100, 0:100, 0:100)`), the function assumes a stacked structure and collapses `age` to its unique sorted values. The number of repetitions becomes `n_causes`.

Return shape:

The output mirrors the input form:

- If the inputs were vectors, outputs are vectors.
- If inputs were matrices or data frames (wide), outputs are matrices.
- If inputs were stacked vectors, outputs are stacked vectors in the same order.

Value

An object of class `"LEdecomp"`:

- `mx1`, `mx2`, `age`, `sex1`, `sex2`, `method`, `closeout`, `opt`, `tol`, `Num_Intervals`, `symmetrical`, `direction`, `perturb`
- `sens`: vector/matrix of sensitivities (same dimensions as inputs)
- `LE1`, `LE2`: life expectancy for `mx1` and `mx2`
- `LEdecomp`: vector/matrix of contributions (same shape as inputs)

References

Arriaga EE (1984). "Measuring and explaining the change in life expectancies." *Demography*, **21**, 83–96. Chandrasekaran C (1986). "Assessing the effect of mortality change in an age group

on the expectation of life at birth." *Janasamkhya*, **4**(1), 1–9. Preston S, Heuveline P, Guillot M (2000). *Demography: measuring and modeling population processes*. Wiley-Blackwell. Ponnapalli KM (2005). "A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth." *Demographic Research*, **12**, 141–172. Horiuchi S, Wilmoth JR, Pletcher SD (2008). "A decomposition method based on a model of continuous change." *Demography*, **45**(4), 785–801. Andreev EM, Shkolnikov VM, Begun AZ (2002). "Algorithm for decomposition of differences between aggregate demographic measures and its application to life expectancies, healthy life expectancies, parity-progression ratios and total fertility rates." *Demographic research*, **7**, 499–522.

See Also

`sen_e0_mx_lt()`, `arriaga()`, `arriaga_sym()`, `sen_arriaga()`, `sen_arriaga_sym()`

Examples

```
## Simple reproducible setup
set.seed(123)
a <- 0.001
b <- 0.07

## 1) Vector (single cause), single-year ages
age <- 0:50
mx1 <- a * exp(age * b)
mx2 <- (a / 2) * exp(age * b)

res_vec <- LEdecomp(
  mx1 = mx1, mx2 = mx2,
  age = age, nx = rep(1, length(age)),
  sex1 = "t", method = "sen_arriaga", opt = TRUE
)
round(sum(res_vec$LEdecomp), 4)

## 2) Matrix (multiple causes): rows = age, cols = causes
##    Build 3 causes with random positive weights per age
k <- 3
w1 <- matrix(runif(length(age) * k, 0.9, 1.1), nrow = length(age)); w1 <- w1 / rowSums(w1)
w2 <- matrix(runif(length(age) * k, 0.9, 1.1), nrow = length(age)); w2 <- w2 / rowSums(w2)
mx1_mat <- (mx1) * w1
mx2_mat <- (mx2) * w2
colnames(mx1_mat) <- colnames(mx2_mat) <- paste0("c", 1:k)
rownames(mx1_mat) <- rownames(mx2_mat) <- as.character(age)

res_mat <- LEdecomp(
  mx1 = mx1_mat, mx2 = mx2_mat,
  age = age, nx = rep(1, length(age)),
  sex1 = "t", method = "sen_arriaga", opt = TRUE
)
## Check: row-summed cause contributions equal all-cause result
res_all <- LEdecomp(
  mx1 = mx1, mx2 = mx2,
  age = age, nx = rep(1, length(age)),
```

```
  sex1 = "t", method = "sen_arriaga", opt = TRUE
)
all.equal(rowSums(res_mat$LEdecomp), res_all$LEdecomp, tolerance = 1e-7)

## 3) Data frame (wide): same as matrix but with an 'age' column
df1 <- data.frame(age = age, mx1_mat, check.names = FALSE)
df2 <- data.frame(age = age, mx2_mat, check.names = FALSE)
res_df <- LEdecomp(
  mx1 = df1, mx2 = df2,
  age = NULL, nx = rep(1, length(age)),
  sex1 = "t", method = "sen_arriaga", opt = TRUE
)
all.equal(res_df$LEdecomp, res_mat$LEdecomp, tolerance = 1e-8)

## 4) Stacked vector (long/concatenated): all ages for cause 1, then cause 2, etc.
##    If 'age' is repeated per cause, LEdecomp infers n_causes and collapses age.
mx1_stack <- as.vector(mx1_mat)  # column-major flattening
mx2_stack <- as.vector(mx2_mat)
age_rep   <- rep(age, k)         # typical tidy pipeline: age repeated per cause

res_stack <- LEdecomp(
  mx1 = mx1_stack, mx2 = mx2_stack,
  age = age_rep, nx = NULL,
  sex1 = "t", method = "sen_arriaga", opt = TRUE
)
## Output is a stacked vector matching the matrix baseline when flattened
all.equal(res_stack$LEdecomp, c(res_mat$LEdecomp), tolerance = 1e-8)

## 5) Abridged ages (0,1,5,10,...,110): inference when labels are missing
age_ab <- c(0L, 1L, seq.int(5L, 110L, by = 5L))
nx_ab  <- c(diff(age_ab), tail(diff(age_ab), 1L))
mx1_ab <- a * exp(age_ab * b)
mx2_ab <- (a / 2) * exp(age_ab * b)

## Explicit abridged example
res_ab_explicit <- LEdecomp(
  mx1 = mx1_ab, mx2 = mx2_ab,
  age = age_ab, nx = nx_ab,
  sex1 = "t", method = "sen_arriaga", opt = TRUE
)

## Unlabeled abridged vector of the same length: age and nx inferred
res_ab_infer <- LEdecomp(
  mx1 = mx1_ab, mx2 = mx2_ab,
  age = NULL, nx = NULL,
  sex1 = "t", method = "sen_arriaga", opt = TRUE
)
all.equal(res_ab_infer$age, as.numeric(age_ab))
all.equal(res_ab_infer$LEdecomp, res_ab_explicit$LEdecomp, tolerance = 1e-8)

## 6) Rownames override age when they look like ages
##    Here we give the wrong 'age' but set rownames to "0","1",...,"50".
wrong_age <- age + 10
```

```
mx1_rn <- mx1_mat; mx2_rn <- mx2_mat
rownames(mx1_rn) <- rownames(mx2_rn) <- as.character(age)
res_rn <- suppressWarnings(LEdecomp(
  mx1 = mx1_rn, mx2 = mx2_rn,
  age = wrong_age, nx = rep(1, length(age)),
  sex1 = "t", method = "sen_arriaga", opt = TRUE
))
all.equal(res_rn$age, as.numeric(age))

## 7) List available methods
available_methods()
```

`lL_to_ex` *calculate remaining life expectancy* `ex` *for each age*

Description

Here we combine HMD Method Protocol equations 79 and 80. We calculate all the remaining years left to live at each age, then condition this on survival to each age.

Usage

```
lL_to_ex(lx, Lx)
```

Arguments

`lx`	numeric vector of lifetable survivors at exact age `x`
`Lx`	numeric vector of lifetable exposure `Lx`

Value

numeric vector of remaining life expectancy `ex`

References

Wilmoth JR, Andreev K, Jdanov D, Glei DART, Boe C, Bubenheim M, Philipov D, Shkolnikov V, Vachon P, Winant C, M B (2021). “Methods protocol for the human mortality database. Version 6.” *University of California, Berkeley, and Max Planck Institute for Demographic Research, Rostock. URL: http://mortality.org [version 26/01/2021].*

`lopez_ruzicka` *Lopez-Ruzicka decomposition*

Description

Implements the decomposition of life expectancy proposed by Lopez and Ruzicka, as described in Ponnapalli (2005). This method expresses the difference in life expectancy between two mortality schedules in terms of an exclusive effect and an interaction effect, using life table quantities.

Let e_x^i denote remaining life expectancy at age x for population i, and l_x^i the number of survivors to age x. Then:

- **Exclusive effect**:
$$\frac{l_x^1}{l_x^2}\left[l_x^2(e_x^2 - e_x^1) - l_{x+n}^2(e_{x+n}^2 - e_{x+n}^1)\right]$$
- **Interaction effect**:
$$(e_{x+n}^2 - e_{x+n}^1) \cdot \left[\frac{l_x^1 \cdot l_{x+n}^2}{l_x^2} - l_{x+n}^1\right]$$

The total contribution to life expectancy difference in age group x is the sum of the exclusive and interaction effects.

Usage

```
lopez_ruzicka(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This method produces **numerically identical results** to `arriaga()`.

Value

Numeric vector of contributions by age group that sum to the total difference in life expectancy between the two mortality schedules.

References

Ponnapalli KM (2005). "A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth." *Demographic Research*, **12**, 141–172.

See Also

`arriaga`, `chandrasekaran_III`, `lopez_ruzicka_sym`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
cc <- lopez_ruzicka(mx1, mx2, age = x)
sum(cc)
```

`lopez_ruzicka_sym` *Symmetric Lopez-Ruzicka decomposition*

Description

Implements a symmetric version of the Lopez-Ruzicka decomposition by averaging the results from the forward and reverse directions. That is, `lopez_ruzicka_sym(mx1, mx2)` returns

$$\frac{1}{2}\left(\text{lopez\_ruzicka}(mx1, mx2) - \text{lopez\_ruzicka}(mx2, mx1)\right)$$

This symmetric adjustment ensures that the decomposition is directionally neutral.

Usage

```
lopez_ruzicka_sym(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
```

```
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This symmetric version gives **numerically identical results** to `arriaga_sym()`, `chandrasekaran_II()`, and `chandrasekaran_III()`.

Value

Numeric vector of contributions by age group that sum to the total difference in life expectancy between the two mortality schedules.

References

Ponnapalli KM (2005). "A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth." *Demographic Research*, **12**, 141–172.

See Also

`lopez_ruzicka`, `arriaga_sym`, `chandrasekaran_II`, `chandrasekaran_III`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
d <- lopez_ruzicka_sym(mx1, mx2, age = x)

# compare to arriaga_sym()
d2 <- arriaga_sym(mx1, mx2, age = x)
all.equal(d, d2)
```

`lx_to_dx` *Calculate the lifetable death distribution*

Description

Minus the decumulation of the survival curve gives the death distribution. Or the element-wise product of `lx` and the conditional death probabilities `qx` gives the same thing.

Usage

```
lx_to_dx(lx)
```

Arguments

`lx`	numeric vector of lifetable survivors at exact age `x`

Value

numeric vector of `dx` values

`mx_to_ax` *produce single-age `ax` values*

Description

We assume mid-interval `ax` except for age 0 and potentially the open age group. `ax` is defined as the average years lived in each age interval by those that die within the interval, and it is used to increase the precision of lifetable estimates. We allow ourselves the midpoint rule for single ages because it has little leverage. If we were working with abridged ages then we would need to use a more sophisticated method.

Usage

```
mx_to_ax(
  mx,
  nx = rep(1, length(mx)),
  age = 0:(length(mx) - 1),
  sex = "t",
  closeout = TRUE
)
```

Arguments

`mx`	numeric vector of the mortality rates (central death rates)
`nx`	age interval width, assumes 1 by default
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`sex`	character: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`closeout`	logical. Default `TRUE`.

Details

For the case of Total sex, we estimate the male and female $a(0)$ using the Andreev-Kingkade rule of thumb, and then average them. We assume a value of 1/2 for all other ages, unless `closeout = TRUE`, in which case we close with `1/mx` for the final value.

Value

numeric vector of `ax` values, the same length as mx

References

Andreev EM, Kingkade WW (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.

`mx_to_e0` *calculate life expectancy at birth from mortality rates*

Description

We follow the full chain of standard lifetable column calculations to translate `mx` to `ex`, then select the first element of `ex`. If `min(age) > 0`, then we return remaining life expectancy at the lowest given age.

Usage

```
mx_to_e0(mx, age, sex = "t", nx = rep(1, length(age)), closeout = TRUE)
```

Arguments

`mx`	numeric vector of the mortality rates (central death rates)
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`sex`	character: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`nx`	age interval width, assumes 1 by default
`closeout`	logical. Default `TRUE`.

Value

numeric scalar of `e0`

`mx_to_ex` *calculate remaining life expectancy from mortality rates*

Description

We follow the full chain of standard lifetable column calculations to translate `mx` to `ex`.

Usage

```
mx_to_ex(mx, age, nx = rep(1, length(age)), sex = "t", closeout = TRUE)
```

Arguments

`mx`	numeric vector of the mortality rates (central death rates)
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	age interval width, assumes 1 by default
`sex`	character: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`closeout`	logical. Default `TRUE`.

Value

numeric vector of `ex`, the same length as `mx`

`mx_to_qx` *produce single-age* `qx` *values*

Description

`qx` gives conditional death probabilities, in this case forced to be consistent with a set of `mx` and `ax` values per HMD Method Protocol eq 71.

Usage

```
mx_to_qx(mx, ax, nx = rep(1, length(mx)), closeout = TRUE)
```

Arguments

`mx`	numeric vector of the mortality rates (central death rates)
`ax`	numeric vector of `ax` values
`nx`	age interval width, assumes 1 by default
`closeout`	logical. Default `TRUE`.

Value

numeric vector of `qx`, the same length as `mx`

`plot.LEdecomp` *Plot Life-Expectancy Decomposition Results (ggplot2)*

Description

Plot contributions (or sensitivities) to a life expectancy difference by age or by age and cause using ggplot2. This is just for a quick default plot method.

Usage

```
## S3 method for class 'LEdecomp'
plot(
  x,
  what = c("LEdecomp", "sens"),
  geom = c("auto", "line", "bar"),
  col = NULL,
  lwd = 1.2,
  xlab = "Age",
  ylab = NULL,
  main = NULL,
  legend = TRUE,
  legend_pos = "right",
  abridged_midpoints = FALSE,
  layout = c("overlay", "facet"),
  ncol = NULL,
  ...
)
```

Arguments

`x`	An object of class `"LEdecomp"`.
`what`	Which series to plot: `"LEdecomp"` (default) or `"sens"`.
`geom`	Plot geometry: "auto", "line", or "bar". If "auto", "bar" is used for what = "LEdecomp" and "line" for what = "sens".
`col`	Optional vector of colors for causes. If `NULL`, a fixed palette is used and recycled as needed.
`lwd`	Line width for cause lines (default 1.2).
`xlab`, `ylab`, `main`	Axis labels and main title. If `ylab` is `NULL`, the default is `"Difference explained (years)"` for `what` = `"LEdecomp"` and `"Sensitivity d(e0)/d(mx)"` for `what` = `"sens"`.
`legend`	Logical. Show legend (primarily relevant for `layout` = `"overlay"`).

`legend_pos`	Legend position passed to `theme(legend.position = ...)`. Accepts `"none"`, `"left"`, `"right"`, `"bottom"`, `"top"`, or numeric coordinates `c(x, y)` in [0,1] inside the panel.
`abridged_midpoints`	Logical. If `TRUE` and ages are abridged (`0, 1, 5, 10, ...`), plot against bin midpoints instead of lower bounds.
`layout`	Plot layout for cause-of-death results: `"overlay"` (all causes in one panel) or `"facet"` (one panel per cause).
`ncol`	Number of columns to use when `layout = "facet"`. If `NULL`, a default is chosen based on the number of causes.
`...`	Reserved for future use.

Details

By default, if `what = "LEdecomp"` we plot using bars (`geom = "bar"`), but you can override this. For bar plots, recall it's the area, not the height that the eye reads; for this reason, if your data is in non-single ages, we divide out the interval width, so that the implied uniform graduation to single ages still sums to the gap. If `what = "sens"` note we plot on a single-age scale even if the data are in abridged ages.

Value

Invisibly returns the ggplot object (after printing).

Examples

```
data("US_data_CoD", package = "LEdecomp")
allc <- subset(US_data_CoD, Period == 2010 & cause == "All-causes") |>
  as.data.frame()

# Make Female vs Male all-cause schedules, Age 0:100
ac_w <- reshape(allc[, c("Gender","Age","mxt")],
                timevar = "Gender", idvar = "Age", direction = "wide")
names(ac_w) <- sub("^mxt\\.", "", names(ac_w))
ac_w <- ac_w[order(ac_w$Age), ]

dec_ac <- LEdecomp(
  mx1 = ac_w$Male,
  mx2 = ac_w$Female,
  age = 0:100,
  method = "sen_arriaga"
)

# Simple single-line plot

plot(dec_ac, main = "All-cause Arriaga, 2010 Female vs Male")

## End(Not run)
## Example 2: Cause of death, one year, Female vs Male
cod <- subset(US_data_CoD, Period == 2010 & cause != "All-causes")
```

```
cod_w <- reshape(cod[, c("Gender","Age","cause","mxt")],
                 timevar = "Gender", idvar = c("cause","Age"),
                 direction = "wide")|>
  as.data.frame()
names(cod_w) <- sub("^mxt\\.", "", names(cod_w))
cod_w <- cod_w[order(cod_w$cause, cod_w$Age), ]

dec_cod <- LEdecomp(
  mx1 = cod_w$Male,
  mx2 = cod_w$Female,
  age = 0:100,
  n_causes = length(unique(cod_w$cause)),
  cause_names = unique(cod$cause_id),
  method = "sen_arriaga"
)

# Overlay of all causes

plot(dec_cod, layout = "overlay", main = "Arriaga CoD, 2010 Female vs Male", legend.pos = "top")

# Facet by cause (3 columns)
plot(dec_cod, layout = "facet", ncol = 3, main = "Arriaga by cause (faceted)")


## Example 3: How to add an all-cause total line yourself (overlay)

p <- plot(dec_cod, layout = "overlay", main = "Overlay with manual Total")
y_mat <- if (is.matrix(dec_cod$LEdecomp)) dec_cod$LEdecomp else
  matrix(dec_cod$LEdecomp, nrow = length(dec_cod$age))
total <- rowSums(y_mat)
p + ggplot2::geom_line(
  data = data.frame(age = dec_cod$age, total = total),
  mapping = ggplot2::aes(x = .data$age, y = .data$total),
  inherit.aes = FALSE, color = "black", linewidth = 1.1)
```

`qx_to_lx` *Calculate the survival curve*

Description

The survival curve is calculated as the cumulative product of the conditional survival probabilities, which are the complement of conditional death probabilities, qx, except we take care to start with a clean 1. This function no radix option. `lx` with a radix of 1 can be interpreted as the probability of surviving from birth to age `x`.

Usage

```
qx_to_lx(qx, radix = 1)
```

Arguments

`qx`	numeric vector of conditional death probabilities
`radix`	initial lifetable cohort size, $l(0)$. Default 1.

Value

numeric vector of `lx` values

`rcumsum` *top-down cumulative sums, as in the lifetable Tx*

Description

Why write `x |> rev() |> cumsum() |> rev()` when you can just write `rcumsum(x)`?

Usage

```
rcumsum(x)
```

Arguments

`x`	numeric vector

Value

numeric vector the same length as `x`

`sen_arriaga` *the sensitivity implied by a classic Arriaga decomposition*

Description

The sensitivity of life expectancy to a perturbation in mortality rates can be derived by dividing the Arriaga decomposition result Δ by the difference `mx2-mx1`.

$$s_x = \frac{\Delta}{{}_nM_x^2 -_n M_x^1}$$

Usage

```
sen_arriaga(
  mx1,
  mx2,
  age = 0:(length(mx1) - 1),
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Value

`s` numeric vector with one element per age group, and which gives the sensitivity values for each age.

References

Arriaga EE (1984). “Measuring and explaining the change in life expectancies.” *Demography*, **21**, 83–96. Preston S, Heuveline P, Guillot M (2000). *Demography: measuring and modeling population processes*. Wiley-Blackwell.

See Also

`arriaga`

Examples

```
a <- .001
b <- .07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
cc <- arriaga(mx1, mx2, age = x)
# examples can come from above too
s <- sen_arriaga(mx1, mx2, age = x)

plot(x, s)

cc_check <- s * (mx2 - mx1)

plot(x,cc)
lines(x,cc_check)
```

`sen_arriaga_instantaneous`

Estimate sensitivity of life expectancy for a set of mortality rates

---

Description

This implementation gives a good approximation of the sensitivity of life expectancy to perturbations in mortality rates (central death rates). Since the Arriaga approach requires two versions of mortality rates `mx1`, `mx2`, we create these by slightly perturbing `mx` up and down. Then we calculate the Arriaga sensitivity in each direction and take the average. Specifically, we create `mx1` and `mx2` as

$$m_x^1 = m_x \cdot \left(\frac{1}{1-h} \right)$$

$$m_x^2 = m_x \cdot (1-h)$$

where `h` is small value given by the argument `perturb`.

Usage

```
sen_arriaga_instantaneous(
  mx,
  age = 0:(length(mx1) - 1),
  sex = "t",
  nx = rep(1, length(mx)),
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	numeric vector of mortality rates (central death rates)
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`sex`	character Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`nx`	integer vector of age intervals, default 1.
`perturb`	numeric constant, a very small number
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

A minor correction might be needed for the final age group for the case of the reverse-direction Arriaga sensitivity. Note also for values of `perturb` (h) that are less than `1e-7` we might lose stability in results.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

Examples

```
a   <- .001
b   <- .07
x   <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
mx  <- (mx1 + mx2) / 2
s     <- sen_arriaga_instantaneous(mx, age = x)
s1    <- sen_arriaga_instantaneous(mx1, age = x)
s2    <- sen_arriaga_instantaneous(mx2, age = x)
s1_d  <- sen_arriaga(mx1, mx2, age = x)
s2_d  <- sen_arriaga(mx2, mx1, age = x)
delta <- mx2 - mx1
# dots give our point estimate of sensitivity at the average of the rates,
# which is different from the

plot(x,s*delta, ylim = c(0,.3))
lines(x,s1*delta,col = "red")
lines(x,s2*delta,col = "blue")
# the sensitivity of the average is different
# from the average of the sensitivities!
lines(x, ((s1+s2)) / 2 * delta)
# and these are different from the directional sensitivities
# covering the whole space from mx1 to mx2:
lines(x, s1_d*delta, col = "red", lty =2)
lines(x, s2_d*delta, col = "blue", lty =2)
```

`sen_arriaga_instantaneous2` *Estimate sensitivity of life expectancy for a set of mortality rates by perturbing in the log space.*

Description

This is a second approach for estimating the sensitivity for a single set of rates. Here, rather than directly expanding and contracting rates to convert `mx` into `mx1` and `mx2` we instead shift the logged mortality rates up and down by the factor `perturb` = `h`. Specifically:

$$m_x^1 = e^{\ln(m_x)+h}$$

$$m_x^2 = e^{\ln(m_x)-h}$$

Usage

```
sen_arriaga_instantaneous2(
  mx,
  age = 0:(length(mx1) - 1),
  sex = "t",
  nx = rep(1, length(mx)),
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	numeric vector of mortality rates (central death rates)
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`sex`	character Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`nx`	integer vector of age intervals, default 1.
`perturb`	numeric constant, a very small number
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`

See Also

`sen_arriaga_instantaneous`

Examples

```
a   <- .001
b   <- .07
x   <- 0:100
mx <- a * exp(x * b)
# the multiplicative perturbation:
s1 <- sen_arriaga_instantaneous(mx)
s2 <- sen_arriaga_instantaneous2(mx)
plot(x,
     s1 - s2,
     pch = 16,
     cex=.5,
     main = "very similar")
```

`sen_arriaga_sym` *Estimate sensitivity of life expectancy using a symmetrical Arriaga approach.*

Description

This approach conducts a classic Arriaga decomposition in both directions, averaging the (sign-adjusted) result, i.e. `a_avg = (arriaga(mx1,mx2, ...) - arriaga(mx2, mx1, ...)) / 2`, then approximates the sensitivity by dividing out the rate differences, i.e. `s = a_avg / (mx2 - mx1)`. A resulting decomposition will be exact because the two arriaga directions are exact, but this method might be vulnerable to 0s in the denominator.

Usage

```
sen_arriaga_sym(
  mx1,
  mx2,
  age = 0:(length(mx1) - 1),
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Value

numeric vector of life expectancy sensitivity to perturbations in mx evaluated at the average of `mx1` and `mx2`.

See Also

`arriaga`

Examples

```
a <- .001
b <- .07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
s <- sen_arriaga_sym(mx1, mx2, age = x)

e01 <- mx_to_e0(mx1,age=x)
e02 <- mx_to_e0(mx2,age=x)
(Delta <- e02 - e01)
deltas <- mx2- mx1
sum(deltas * s)
mx_avg <- (mx1+mx2) / 2

mx_avg <- (mx1 + mx2) / 2
plot(x, s, type = 'l')
lines(x, sen_arriaga_instantaneous(mx_avg, age=x),col = "blue")
```

`sen_arriaga_sym_instantaneous`

Instantaneous sensitivity via symmetrical Arriaga decomposition

---

Description

Estimates the sensitivity of life expectancy to small changes in age-specific mortality rates using the symmetrical Arriaga decomposition. This is done by applying a small multiplicative perturbation to the input mortality rates and using the symmetrical sensitivity function `sen_arriaga_sym()`.

Specifically, the function constructs:

$$m_x^1 = m_x \cdot \left(\frac{1}{1-h}\right)$$

$$m_x^2 = m_x \cdot (1-h)$$

and applies `sen_arriaga_sym(mx1, mx2, ...)` to the result.

Usage

```
sen_arriaga_sym_instantaneous(
  mx,
  age = 0:(length(mx1) - 1),
  sex = "t",
  nx = rep(1, length(mx)),
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`sex`	Character; "m" for male, "f" for female, or "t" for total.
`nx`	integer vector of age intervals, default 1.
`perturb`	Numeric; a small constant determining the perturbation size (default 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This function yields an instantaneous approximation to the derivative of life expectancy with respect to mortality, evaluated at the input schedule. Because `sen_arriaga_sym()` is itself symmetrical, only the "forward" perturbation is required.

Value

numeric vector of life expectancy sensitivity to perturbations in `mx`.

See Also

`sen_arriaga_sym`, `sen_arriaga_sym_instantaneous2`, `sen_lopez_ruzicka_instantaneous`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_arriaga_sym_instantaneous(mx, age = x)

plot(x, s, type = "l")
```

`sen_arriaga_sym_instantaneous2`

Estimate sensitivity of life expectancy for a set of mortality rates by perturbing in the log space.

Description

This is a second approach for estimating the sensitivity for a single set of rates. Here, rather than directly expanding and contracting rates to convert `mx` into `mx1` and `mx2` we instead shift the logged mortality rates up and down by the factor `perturb` = `h`. Specifically:

$$m_x^1 = e^{\ln(m_x)+h}$$

$$m_x^2 = e^{\ln(m_x)-h}$$

Usage

```
sen_arriaga_sym_instantaneous2(
  mx,
  age = 0:(length(mx1) - 1),
  sex = "t",
  nx = rep(1, length(mx)),
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	numeric vector of mortality rates (central death rates)
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`sex`	character Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`nx`	integer vector of age intervals, default 1.
`perturb`	numeric constant, a very small number
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Value

numeric vector of life expectancy sensitivity to perturbations in `mx`.

See Also

`sen_arriaga_instantaneous`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_arriaga_sym_instantaneous2(mx, age = x)

plot(x, s, type = "l")
```

`sen_chandrasekaran_II` *Sensitivity from Chandrasekaran II decomposition*

Description

Computes the sensitivity of life expectancy to changes in age-specific mortality rates using the Chandrasekaran II decomposition approach described by Ponnapalli (2005). The sensitivity is obtained by dividing the age-specific contributions (from `chandrasekaran_II()`) by the differences in mortality rates (`mx2 - mx1`). This yields a pointwise estimate of the derivative of life expectancy with respect to each age-specific mortality rate evaluated at an imagined midpoint between the first a second set of mortality rates.

Usage

```
sen_chandrasekaran_II(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This give numerically identical results to `sen_arriaga_sym()`, `sen_lopez_ruzicka_sym()`, and `sen_chandrasekaran_III()`.

Value

A numeric vector of sensitivity values by age group.

numeric vector of sensitivity of life expectancy to perturbations in `mx` between `mx1` and `mx2`.

References

Ponnapalli KM (2005). "A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth." *Demographic Research*, **12**, 141–172.

See Also

`chandrasekaran_II`, `sen_arriaga`, `sen_arriaga_sym`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
s <- sen_chandrasekaran_II(mx1, mx2, age = x)

# Check that multiplying sensitivity by rate difference approximates the decomposition
cc_check <- s * (mx2 - mx1)
cc <- chandrasekaran_II(mx1, mx2, age = x)

plot(x, cc, type = "l")
lines(x, cc_check, col = "red", lty = 2)
```

`sen_chandrasekaran_III`

Sensitivity from Chandrasekaran III decomposition

Description

Computes the implied sensitivity of life expectancy to changes in age-specific mortality rates using the Chandrasekaran III decomposition approach described by Ponnapalli (2005). The sensitivity is obtained by dividing the age-specific contributions (from `chandrasekaran_III()`) by the differences in mortality rates (`mx2 - mx1`). This yields a pointwise estimate of the derivative of life expectancy with respect to each age-specific mortality rate evaluated at an imagined midpoint between the first a second set of mortality rates.

Usage

```
sen_chandrasekaran_III(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
```

```
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This gives numerically identical results to `sen_arriaga_sym()`, `sen_lopez_ruzicka_sym()`, and `sen_chandrasekaran_II()`.

Value

A numeric vector of sensitivity values by age group.

References

Ponnapalli KM (2005). “A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth.” *Demographic Research*, **12**, 141–172.

See Also

`chandrasekaran_III`, `sen_chandrasekaran_II`, `sen_arriaga_sym`, `sen_lopez_ruzicka_sym`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
s <- sen_chandrasekaran_III(mx1, mx2, age = x)

# Check that multiplying sensitivity by rate difference approximates the decomposition
cc_check <- s * (mx2 - mx1)
cc <- chandrasekaran_III(mx1, mx2, age = x)
```

```
plot(x, cc, type = "l")
lines(x, cc_check, col = "red", lty = 2)
```

`sen_chandrasekaran_III_instantaneous`

Instantaneous sensitivity via Chandrasekaran III decomposition

Description

Estimates the sensitivity of life expectancy to small changes in mortality rates using the Chandrasekaran III decomposition. This is done by perturbing the input mortality rates up and down by a small factor and computing directional sensitivity from the result.

Specifically:

$$m_x^1 = m_x \cdot \left(\frac{1}{1-h} \right)$$

$$m_x^2 = m_x \cdot (1-h)$$

and applies `sen_chandrasekaran_III(mx1, mx2, ...)` to the result.

Usage

```
sen_chandrasekaran_III_instantaneous(
  mx,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; "m" for male, "f" for female, or "t" for total.
`perturb`	Numeric; a small constant determining the perturbation size (default: 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This approach provides an approximation of the derivative of life expectancy with respect to each age-specific mortality rate, evaluated near the input `mx`. It gives numerically identical results to `sen_arriaga_sym_instantaneous()`, `sen_lopez_ruzicka_instantaneous()`, and `sen_chandrasekaran_II_instanta`

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

See Also

`sen_chandrasekaran_III`, `sen_chandrasekaran_III_instantaneous2`, `sen_arriaga_sym_instantaneous`, `sen_lopez_ruzicka_instantaneous`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_chandrasekaran_III_instantaneous(mx, age = x)

plot(x, s, type = "l")
```

---

sen_chandrasekaran_III_instantaneous2

Log-space instantaneous sensitivity via Chandrasekaran III decomposition

Description

Estimates the sensitivity of life expectancy to small changes in mortality rates using the Chandrasekaran III decomposition and log-transformed perturbations. The method perturbs `mx` up and down in log space and averages the directional sensitivities to approximate the derivative.

Specifically:

$$m_x^1 = \exp(\ln m_x + h)$$

$$m_x^2 = \exp(\ln m_x - h)$$

and applies `sen_chandrasekaran_III(mx1, mx2, ...)` and `sen_chandrasekaran_III(mx2, mx1, ...)`, returning their average.

Usage

```
sen_chandrasekaran_III_instantaneous2(
  mx,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; "m" for male, "f" for female, or "t" for total.
`perturb`	Numeric; a small constant determining the perturbation size (default: 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This version uses symmetric log-space perturbations. It gives numerically identical results to `sen_arriaga_sym_instantaneous2()`, `sen_lopez_ruzicka_instantaneous2()`, and `sen_chandrasekaran_II_instan`

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

See Also

`sen_chandrasekaran_III_instantaneous`, `sen_arriaga_sym_instantaneous2`, `sen_lopez_ruzicka_instantaneous`
`sen_chandrasekaran_II_instantaneous2`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_chandrasekaran_III_instantaneous2(mx, age = x)

plot(x, s, type = ”l”)
```

`sen_chandrasekaran_II_instantaneous`

Instantaneous sensitivity via Chandrasekaran II decomposition

Description

Estimates the sensitivity of life expectancy to small changes in mortality rates using the Chandrasekaran II decomposition. This is done by perturbing the input mortality rates up and down by a small factor and calculating the directional sensitivity.

Specifically, the function constructs:

$$m_x^1 = m_x \cdot \left(\frac{1}{1-h}\right)$$

$$m_x^2 = m_x \cdot (1-h)$$

and applies `sen_chandrasekaran_II(mx1, mx2, ...)` to the result.

Usage

```
sen_chandrasekaran_II_instantaneous(
  mx,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; "m" for male, "f" for female, or "t" for total.
`perturb`	Numeric; a small constant determining the perturbation size (default 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This approach gives a reasonable approximation of the derivative of life expectancy with respect to each age-specific mortality rate. It gives numerically identical results to `sen_arriaga_sym_instantaneous()`, `sen_lopez_ruzicka_instantaneous()`, and `sen_chandrasekaran_III_instantaneous()`.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`

See Also

`sen_chandrasekaran_II`, `sen_chandrasekaran_II_instantaneous2`, `sen_arriaga_sym_instantaneous`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_chandrasekaran_II_instantaneous(mx, age = x)

plot(x, s, type = "l")
```

`sen_chandrasekaran_II_instantaneous2` *Log-space instantaneous sensitivity via Chandrasekaran II decomposition*

Description

Estimates the sensitivity of life expectancy to small changes in mortality rates using the Chandrasekaran II decomposition. This variant perturbs the mortality rates in **log space**, creating two versions of `mx` by adding and subtracting a small constant to `log(mx)`, then exponentiating.

Specifically:

$$m_x^1 = \exp\left(\ln m_x + h\right)$$
$$m_x^2 = \exp\left(\ln m_x - h\right)$$

and applies `sen_chandrasekaran_II(mx1, mx2, ...)` to the result.

Usage

```
sen_chandrasekaran_II_instantaneous2(
  mx,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; "m" for male, "f" for female, or "t" for total.
`perturb`	Numeric; a small constant determining the perturbation size (default 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This approach provides a log-linear perturbation of the mortality schedule and can be used to estimate the derivative of life expectancy with respect to logged mortality rates. It gives numerically identical results to `sen_arriaga_sym_instantaneous2()`, `sen_lopez_ruzicka_instantaneous2()`, and `sen_chandrasekaran_III_instantaneous2()`.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

See Also

`sen_chandrasekaran_II_instantaneous`, `sen_arriaga_sym_instantaneous2`, `sen_lopez_ruzicka_instantaneous2`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_chandrasekaran_II_instantaneous2(mx, age = x)

plot(x, s, type = "l")
```

`sen_e0_mx_lt` *A direct approximation of the sensitivity of life expectancy at birth to changes in mortality.*

Description

This function tries to get the direct discrete life expectancy sensitivity to $m(x)$, in continuous math it's $-l(x)e(x)$, we just need to find the best approx with a discrete lifetable. This direct lifetable-based calculation requires a few approximations to get a usable value whenever we're working with discrete data. In continuous notation, we know that the sensitivity $s(x)$

$$s(x) = -l(x)e(x)$$

but it is not obvious what to use from a discrete lifetable. In this implementation, we use $L(x)$ and an $a(x)$-weighted average of successive $e(x)$ values, specifically, we calculate:

$$s_x = -L_x \cdot (e_x \cdot (1 - a_x) + e_{x+1} \cdot a_x)$$

This seems to be a very good approximation for ages >0, but we still have a small, but unaccounted-for discrepancy in age 0, at least when comparing with also-imperfect numerical derivatives.

Usage

```
sen_e0_mx_lt(
  mx,
  age = 0:(length(mx) - 1),
  nx = rep(1, length(mx)),
  sex = "t",
  closeout = TRUE
)
```

Arguments

`mx`	numeric vector of the mortality rates (central death rates)
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	age interval width, assumes 1 by default
`sex`	character: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`closeout`	logical. Default `TRUE`.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

Examples

```
x <- 0:100
mx <- 0.001 * exp(x * 0.07)
sl <-  sen_e0_mx_lt(mx,age=x,sex='t',closeout=TRUE)
sn <- numDeriv::grad(mx_to_e0, mx, age=x, sex = 't', closeout=TRUE)

plot(x,sl)
lines(x,sn)

# examine residuals:
sl - sn
# Note discrepancies in ages >0 are due to numerical precision only

plot(x, sl - sn, main = "still uncertain what accounts for the age 0 discrepancy")
```

`sen_lopez_ruzicka` *Sensitivity from Lopez-Ruzicka decomposition*

Description

Computes the sensitivity of life expectancy to changes in age-specific mortality rates using the Lopez-Ruzicka decomposition approach. The sensitivity is calculated by dividing the age-specific contributions (from `lopez_ruzicka()`) by the differences in mortality rates (`mx2 - mx1`). This gives a pointwise estimate of the derivative of life expectancy with respect to each age-specific mortality rate, evaluated at an imagined midpoint between the two input rate schedules.

Usage

```
sen_lopez_ruzicka(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This method gives numerically identical results to `sen_arriaga()`.

Value

A numeric vector of sensitivity values by age group.

References

Ponnapalli KM (2005). “A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth.” *Demographic Research*, **12**, 141–172.

See Also

`lopez_ruzicka`, `sen_arriaga`, `sen_chandrasekaran_III`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a / 2 * exp(x * b)
s <- sen_lopez_ruzicka(mx1, mx2, age = x)

# Check that multiplying sensitivity by rate difference reproduces the decomposition
cc_check <- s * (mx2 - mx1)
cc <- lopez_ruzicka(mx1, mx2, age = x)

plot(x, cc, type = "l")
lines(x, cc_check, col = "red", lty = 2)
```

`sen_lopez_ruzicka_instantaneous`

Instantaneous sensitivity via Lopez-Ruzicka decomposition

Description

Estimates the sensitivity of life expectancy to small changes in mortality rates using the Lopez-Ruzicka decomposition. This is done by perturbing the input mortality rates up and down by a small factor and averaging the resulting directional sensitivities to approximate a symmetric derivative.

Specifically:

$$m_x^1 = m_x \cdot \left(\frac{1}{1-h}\right)$$

$$m_x^2 = m_x \cdot (1-h)$$

and applies `sen_lopez_ruzicka(mx1, mx2, ...)` and `sen_lopez_ruzicka(mx2, mx1, ...)`, returning their average.

Usage

```
sen_lopez_ruzicka_instantaneous(
  mx,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; "m" for male, "f" for female, or "t" for total.
`perturb`	Numeric; a small constant determining the perturbation size (default: 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This method gives numerically identical results to `sen_arriaga_sym_instantaneous()`, `sen_chandrasekaran_II_instar` and `sen_chandrasekaran_III_instantaneous()`.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

See Also

`sen_lopez_ruzicka`, `sen_lopez_ruzicka_instantaneous2`, `sen_arriaga_sym_instantaneous`, `sen_chandrasekaran_II_instantaneous`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_lopez_ruzicka_instantaneous(mx, age = x)

plot(x, s, type = ”l”)
```

`sen_lopez_ruzicka_instantaneous2`

Log-space instantaneous sensitivity via Lopez-Ruzicka decomposition

Description

Estimates the sensitivity of life expectancy to small changes in mortality rates using the Lopez-Ruzicka decomposition and log-space perturbation. This is done by shifting the log of the input mortality rates up and down by a small constant, then exponentiating, and computing the average directional sensitivity.

Specifically:

$$m_x^1 = \exp(\ln m_x + h)$$

$$m_x^2 = \exp(\ln m_x - h)$$

and applies `sen_lopez_ruzicka(mx1, mx2, ...)` and `sen_lopez_ruzicka(mx2, mx1, ...)`, returning their average.

Usage

```
sen_lopez_ruzicka_instantaneous2(
  mx,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; "m" for male, "f" for female, or "t" for total.
`perturb`	Numeric; a small constant determining the perturbation size (default: 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This approach gives numerically identical results to `sen_arriaga_sym_instantaneous2()`, `sen_chandrasekaran_II_ins` and `sen_chandrasekaran_III_instantaneous2()`.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

See Also

`sen_lopez_ruzicka_instantaneous`, `sen_arriaga_sym_instantaneous2`, `sen_chandrasekaran_III_instantaneous`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_lopez_ruzicka_instantaneous2(mx, age = x)
```

```
plot(x, s, type = "l")
```

`sen_lopez_ruzicka_sym` *Sensitivity from symmetric Lopez-Ruzicka decomposition*

Description

Computes the sensitivity of life expectancy to changes in age-specific mortality rates using the symmetric version of the Lopez-Ruzicka decomposition, as described by Ponnapalli (2005). The sensitivity is obtained by dividing the symmetric decomposition result by the differences in mortality rates (`mx2 - mx1`). This yields a pointwise estimate of the derivative of life expectancy with respect to each age-specific mortality rate evaluated at an imagined midpoint between the first and second set of mortality rates.

Usage

```
sen_lopez_ruzicka_sym(
  mx1,
  mx2,
  age = (1:length(mx1)) - 1,
  nx = rep(1, length(mx1)),
  sex1 = "t",
  sex2 = sex1,
  closeout = TRUE
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This method gives numerically identical results to `sen_arriaga_sym()`, `sen_chandrasekaran_II()`, and `sen_chandrasekaran_III()`.

Value

A numeric vector of sensitivity values by age group.

References

Ponnapalli KM (2005). “A comparison of different methods for decomposition of changes in expectation of life at birth and differentials in life expectancy at birth.” *Demographic Research*, **12**, 141–172.

See Also

`lopez_ruzicka_sym`, `sen_arriaga_sym`, `sen_chandrasekaran_II`, `sen_chandrasekaran_III`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
s <- sen_lopez_ruzicka_sym(mx1, mx2, age = x)

# Check equivalence with symmetric Arriaga
s2 <- sen_arriaga_sym(mx1, mx2, age = x)
all.equal(s, s2)
```

---

sen_lopez_ruzicka_sym_instantaneous

Instantaneous sensitivity via symmetrical Lopez-Ruzicka decomposition

Description

Estimates the instantaneous sensitivity of life expectancy to small proportional changes in mortality rates, using the symmetrical Lopez-Ruzicka decomposition. This implementation perturbs the rates up and down around a central value and applies the symmetrical decomposition to the result.

Specifically, the function constructs:

$$m_x^1 = m_x \cdot \left(\frac{1}{1-h} \right)$$

$$m_x^2 = m_x \cdot (1-h)$$

and applies `sen_lopez_ruzicka_sym(mx1, mx2, ...)` to the result.

Usage

```
sen_lopez_ruzicka_sym_instantaneous(
  mx,
  age = 0:(length(mx) - 1),
  nx = rep(1, length(mx)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; `"m"` for male, `"f"` for female, or `"t"` for total.
`perturb`	Numeric; a small constant determining the perturbation size (default 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This gives a pointwise estimate of the derivative of life expectancy with respect to each age-specific mortality rate, evaluated symmetrically around the given mortality schedule. It gives numerically identical results to e.g. `sen_arriaga_sym_instantaneous()` and `sen_chandrasekaran_II_instantaneous()`.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

See Also

`sen_lopez_ruzicka_sym`, `sen_lopez_ruzicka_sym_instantaneous2`, `sen_arriaga_sym_instantaneous`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_lopez_ruzicka_sym_instantaneous(mx, age = x)

plot(x, s, type = "l")
```

sen_lopez_ruzicka_sym_instantaneous2

Log-scale instantaneous sensitivity via symmetrical Lopez-Ruzicka decomposition

Description

Estimates the instantaneous sensitivity of life expectancy using symmetric perturbations in log-scale mortality rates, based on the Lopez-Ruzicka decomposition.

Specifically, the function constructs:

$$m_x^1 = \exp(\log m_x + h)$$

$$m_x^2 = \exp(\log m_x - h)$$

and applies `sen_lopez_ruzicka_sym(mx1, mx2, ...)` to the result.

Usage

```
sen_lopez_ruzicka_sym_instantaneous2(
  mx,
  age = 0:(length(mx) - 1),
  nx = rep(1, length(mx)),
  sex = "t",
  perturb = 1e-06,
  closeout = TRUE
)
```

Arguments

`mx`	Numeric vector of mortality rates (central death rates).
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.
`sex`	Character; `"m"` for male, `"f"` for female, or `"t"` for total.
`perturb`	Numeric; a small constant determining the perturbation size (default 1e-6).
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.

Details

This method gives results equivalent to `sen_lopez_ruzicka_sym_instantaneous()` and `sen_arriaga_sym_instantaneou` and is preferred when working with log-transformed mortality schedules.

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

See Also

`sen_lopez_ruzicka_sym`, `sen_lopez_ruzicka_sym_instantaneous`, `sen_arriaga_sym_instantaneous2`

Examples

```
a <- 0.001
b <- 0.07
x <- 0:100
mx <- a * exp(x * b)
s <- sen_lopez_ruzicka_sym_instantaneous2(mx, age = x)

plot(x, s, type = "l")
```

`sen_min` *sen_min*

Description

Most sensitivity methods in this packages (`sen_arriaga_sym()` excepted) are approximations; when used in decompositions they will tend to imply residuals. To acheive near-exact additivity for a decomposition using these sensitivity approaches, one can try to find a different weighting of rates from populations 1 and 2, rather than simply taking their arithmetic average. Here we turn this into an optimization problem, where we find the weighting `w` that implies an exactly additive decomposition to an arbitrary degree of tolerance.

$$m_x = m_x^1 * w + m_x^2 * (1 - w)$$

Usage

```
sen_min(
  mx1,
  mx2,
  age,
  sex1,
  sex2 = sex1,
  closeout = TRUE,
  sen_fun = sen_arriaga_instantaneous,
  tol = 1e-10,
  ...
)
```

Arguments

`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)

`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.
`sen_fun`	function name, current options include `sen_arriaga_instantaneous`, `sen_arriaga_instantaneous1`, `sen_arriaga_sym`, `sen_e0_mx_lt`, `sen_num`
`tol`	double. tolerance level for residual, passed to `optimise()`
`...`	optional arguments to pass to `sen_fun()`

Details

We expect the value `w` to be close to .5, and only search the interval `[.4,.6]`. This may need to be revisited in case that proves too narrow.

Value

age-specific sensitivity of life expectancy to changes in mortality rates.

Examples

```
a   <- .001
b   <- .07
x   <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
mx  <- (mx1 + mx2) / 2
s1 <- sen_min(mx1, mx2,
              age = x, sex1 = 't',
              closeout = TRUE,
              sen_fun = sen_arriaga_instantaneous)
s2 <- sen_min(mx1, mx2,
              age = x, sex1 = 't',
              closeout = TRUE,
              sen_fun = sen_e0_mx_lt,
              tol = 1e-12)

# check sums
e01 <- mx_to_e0(mx1,age=x,sex='t',closeout=TRUE)
e02 <- mx_to_e0(mx2,age=x,sex='t',closeout=TRUE)
(gap <- e02 - e01)
delta <- mx2 - mx1
(gap1 <- sum(s1 * delta))
(gap2 <- sum(s2 * delta))
gap2-gap

plot(x, s1, type= 'l')
lines(x, s2, col = 'red', lty = 2, lwd = 2)
```

```
plot(x, s2-s1, main = "age 0 difference is due to imprecision in
lifetable approach for this age")
```

`sen_num` *A numerical approximation of the sensitivity of life expectancy at birth to changes in mortality.*

Description

Here we produce a numerical derivative based on the methods implemented in the `numDeriv::grad()` function. Tweaking the optional arguments of `numDeriv::grad()`, passed in via `...` might lead to greater precision, but this method actually performs usably well with its defaults.

Usage

```
sen_num(
  mx,
  age = (1:length(mx)) - 1,
  nx = rep(1, length(mx)),
  sex = "t",
  closeout = TRUE,
  ...
)
```

Arguments

`mx`	numeric vector of the mortality rates (central death rates)
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	age interval width, assumes 1 by default
`sex`	character: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`closeout`	logical. Default `TRUE`.
`...`	optional arguments to pass to `mx_to_e0()`

Value

numeric vector of sensitivity of life expectancy to perturbations in `mx`.

Examples

```
x <- 0:100
mx <- 0.001 * exp(x * 0.07)
sn <- sen_num(mx,age=x,sex='t',closeout=TRUE)
sa <- sen_arriaga_instantaneous2(mx, age=x,sex='t',perturb = 1e-4)

plot(x,sa)
lines(x,sn)
```

```
# examine residuals:
sn - sa
# Note discrepancies in ages >0 are due to numerical precision only

plot(x, sn - sa, main = "still uncertain what accounts for the age 0 discrepancy")
```

---

sen_resid *sen_resid*

Description

Most sensitivity methods in this packages (`sen_arriaga_sym()` excepted) are approximations; when used in decompositions they will tend to imply residuals. To acheive near-exact additivity for a decomposition using these sensitivity approaches, one can try to find a different weighting of rates from populations 1 and 2, rather than simply taking their arithmetic average. Here we turn this into an optimization problem, where we find the weighting `w` that implies an exactly additive decomposition to an arbitrary degree of tolerance. This function gives said residual, for purposes of optimizing using `sen_min()`. We export this auxiliary function because one might wish to know the value w that balances rates such that the decomposition is exact.

$$m_x = m_x^1 * w + m_x^2 * (1 - w)$$

Usage

```
sen_resid(
  w = 0.5,
  mx1,
  mx2,
  age = 0:(length(mx) - 1),
  nx = rep(1, length(mx)),
  sex1,
  sex2 = sex1,
  closeout = TRUE,
  sen_fun = sen_arriaga_instantaneous,
  ...
)
```

Arguments

`w`	the parameter weight to optimize, default 0.5
`mx1`	numeric vector of the mortality rates (central death rates) for population 1
`mx2`	numeric vector of the mortality rates (central death rates) for population 2
`age`	integer vector of the lower bound of each age group (currently only single ages supported)
`nx`	integer vector of age intervals, default 1.

`sex1`	character either the sex for population 1: Male (`"m"`), Female (`"f"`), or Total (`"t"`)
`sex2`	character either the sex for population 2: Male (`"m"`), Female (`"f"`), or Total (`"t"`) assumed same as `sex1` unless otherwise specified.
`closeout`	logical. Default `TRUE`. Shall we use the HMD Method Protocol to close out the `ax` and `qx` values? See details.
`sen_fun`	function name, current options include `sen_e0_mx_lt`, `sen_arriaga_instantaneous`, `sen_arriaga_instantaneous2`, `sen_arriaga_sym`, `sen_num`,`sen_chandrasekaran_II_instantaneou` ,`sen_chandrasekaran_ii_instantaneous2`,`sen_chandrasekaran_iii_instantaneous`, `sen_chandrasekaran_iii_instantaneous2`,`sen_lopez_ruzicka_instantaneous`, `sen_lopez_ruzicka_instantaneous2`
`...`	optional arguments passed to a given sensitivity function.

Value

age-specific sensitivity of life expectancy to changes in mortality rates.

Examples

```
a   <- .001
b   <- .07
x   <- 0:100
mx1 <- a * exp(x * b)
mx2 <- a/2 * exp(x * b)
w <- optimize(sen_resid,
              mx1 = mx1,
              mx2 = mx2,
              age = x,
              sen_fun = sen_arriaga_instantaneous,
              sex1 = 't',
              sex2 = 't',
              closeout = TRUE,
              interval = c(.4,.6))$minimum
w
```

`US_data` *US Mortality data*

Description

Data from the US total population from the Human Mortality Dataset and from National Center for Health Statistics (NCHS). The use of two dataset is justified because NCHS does not contain information of exposures above age 85. The dataset contains information on mortality rates (mxt), registered deaths (Dxt) and the size of the population at risk of death (Ext) by period, from 2000 to 2020, and by age, from 0 to 100 years, for both males and females.

Usage

```
US_data
```

Format

A data frame with 4242 rows and 6 columns with class `"LEdecompData"` and `"data.frame"` including the following information

- `Age` a vector containing the ages considered in the dataset, 0, 1, ..., 99, and 100.
- `Gender` a vector containing the information regarding the gender, `"Male"` or `"Female"`.
- `Period` a vector containing the periods of the dataset from 2000 to 2020.
- `Ext` a vector containing the size of the population at risk of death by age and period.
- `Dxt` a vector containing the number of registered deaths by age and period.
- `mxt` a vector mortality rates for the corresponding age and period.

Examples

```
#The dataset is executed with the following information
US_data
```

`US_data_CoD` *US cause-of-death Mortality data*

Description

Data from the US total population from the Human Mortality Dataset and from National Center for Health Statistics (NCHS). In this case, we have information with the number of deaths by 18 different causes, for more information please review the cause_id and the National Center for Health Statistics (NCHS). The use of two dataset is justified because NCHS does not contain information of exposures above age 85. The dataset contains information on mortality rates (mxt), registered deaths (Dxt) and the size of the population at risk of death (Ext) by period, from 2000 to 2020, and by age, from 0 to 100 years, for both males and females. In addition, we have the number of deaths by cause between 0 and 100 years of age and between 2000 and 2020.

Usage

```
US_data_CoD
```

Format

A data frame with 80598 rows and 8 columns with class `"LEdecompData"` and `"data.frame"` including the following information

- `Age` a vector containing the ages considered in the dataset, 0, 1, ..., 99, and 100.
- `Gender` a vector containing the information regarding the gender, `"Male"` or `"Female"`.
- `Period` a vector containing the periods of the dataset from 2000 to 2020.

- `Ext` a vector containing the size of the population at risk of death by age and period.
- `Dxt` a vector containing the number of registered deaths by age and period.
- `mxt` a vector mortality rates for the corresponding age and period.
- `cause` a vector containing a brief summary of the corresponding cause of death.
- `cause_id` a vector containing the corresponding identification number for the cause of death.

Examples

```
#The dataset is executed with the following information
US_data_CoD
```

Index