Package: DemogAnr (via r-universe)

July 23, 2026

Title Demographic Analysis

Version 0.1.0

Description Tool for demographic modeling and analysis, combining stochastic simulation-based projections with classic demographic methods. Provides utilities for disaggregating population data using Karup-King interpolation, fitting Brass relational logit models, calculating fertility and mortality metrics, building life tables, and performing hierarchical subnational population projections. Relational life table estimation and interpolation methods are described in Brass (1975) ``Methods for Estimating Fertility and Mortality from Limited and Defective Data", Preston et al. (2001, ISBN:978-0631226161) ``Demography: Measuring and Modeling Population Processes", Siegel and Swanson (2004, ISBN:978-0126419559) ``The Methods and Materials of Demography", and Raftery et al. (2012) ``Bayesian probabilistic population projections for all countries" <doi:10.1073/pnas.1211452109>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1)

Imports dplyr, stats, tidyr

Suggests testthat (>= 3.0.0)

LazyData true

Config/testthat/edition 3

NeedsCompilation no

Author Edward Owusu Manu [aut, cre]

Maintainer Edward Owusu Manu <edward.manu@statsghana.gov.gh>

Repository https://cran.r-universe.dev

Date/Publication 2026-07-23 14:37:44 UTC

RemoteUrl https://github.com/cran/DemogAnr

RemoteRef HEAD

RemoteSha 354d0e868e52d5a285f27fa4729a24c056290341

Contents

`brass_logit` *Brass Relational Logit Model*

Description

Fits the Brass relational logit model to an observed series of *cumulative* probabilities of dying $q(x) = 1 - l(x)/l(0)$ using a standard life table. The observed logits are regressed on the logits of the standard, and the fitted relationship (alpha, beta) is used to predict a complete, smoothed series of $q(x)$, including for ages where the observed value is missing (`NA`).

Usage

```
brass_logit(
  data,
  qx_col,
  age_col,
  standard,
  standards_data = NULL,
  verbose = FALSE
)
```

Arguments

`data`	A data frame containing the observed cumulative probabilities of dying $q(x)$ and ages.
`qx_col`	Column name for the cumulative probability of dying $q(x) = 1 - l(x)$ in the data frame. May contain `NA` for ages to be estimated.
`age_col`	Column name for age in the data frame.
`standard`	Standard to use ("African" or "GeneralUN").
`standards_data`	Standards dataset (default: included standards).
`verbose`	Logical. If `TRUE`, prints detailed status messages to the console.

Details

The Brass logit of a survivorship value is $Y(x) = 0.5 \log\{(1 - l(x))/l(x)\} = 0.5 \log\{q(x)/(1 - q(x))\}$, and predicted probabilities are recovered with the inverse transform $q(x) = \exp(2Y)/(1 + \exp(2Y))$.

Note that the input must be cumulative probabilities of dying by age x (i.e. $1-l(x)$), which increase with age, and not the age-interval probabilities ${}_nq_x$. Values of exactly 0 or 1 (for example $q(0) = 0$) are excluded from the fit.

Value

A list containing the fitted model (`model`), the alpha/beta `coefficients`, and `data`: the input data with a `predicted_qx` column of smoothed/completed cumulative probabilities and a `predicted_lx` column of the implied survivorship $l(x) = 1 - q(x)$.

References

Brass, W. (1971). On the scale of mortality. In W. Brass (Ed.), *Biological Aspects of Demography* (pp. 69-110). London: Taylor & Francis.

Brass, W. (1975). *Methods for Estimating Fertility and Mortality from Limited and Defective Data*. Chapel Hill: Carolina Population Center, University of North Carolina.

Moultrie, T., Dorrington, R., Hill, A., Hill, K., Timaeus, I., & Zaba, B. (Eds.). (2013). *Tools for Demographic Estimation*. Paris: International Union for the Scientific Study of Population (IUSSP). https://demographicestimation.iussp.org/

Newell, C. (1988). *Methods and Models in Demography*. New York: Guilford Press. (Chapter on model life tables.)

Examples

```
# Observed cumulative probabilities of dying q(x) = 1 - l(x),
# with gaps (NA) to be filled by the model:
observed_data <- data.frame(
  age = c(1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50),
  qx = c(0.10, 0.15, NA, 0.19, 0.21, NA, 0.26, 0.30, NA, 0.38, 0.44)
)

model <- brass_logit(
```

```
  data = observed_data,
  qx_col = "qx",
  age_col = "age",
  standard = "African"
)

model$coefficients
model$data
```

`data` *Grouped population data by 5-year age groups*

Description

Example population counts in 5-year age groups (0-4 to 75-79) for the years 2021 to 2035, used to demonstrate `karup_king()`.

Usage

```
data
```

Format

A data frame with 16 rows (age groups) and 16 variables:

2021 Numeric. Population in the age group in 2021.

2022 Numeric. Population in the age group in 2022.

2023 Numeric. Population in the age group in 2023.

2024 Numeric. Population in the age group in 2024.

2025 Numeric. Population in the age group in 2025.

2026 Numeric. Population in the age group in 2026.

2027 Numeric. Population in the age group in 2027.

2028 Numeric. Population in the age group in 2028.

2029 Numeric. Population in the age group in 2029.

2030 Numeric. Population in the age group in 2030.

2031 Numeric. Population in the age group in 2031.

2032 Numeric. Population in the age group in 2032.

2033 Numeric. Population in the age group in 2033.

2034 Numeric. Population in the age group in 2034.

2035 Numeric. Population in the age group in 2035.

age_col Character. The 5-year age group, e.g. "0-4", "5-9", ..., "75-79".

Source

Simulated data based on population structures from the Ghana Statistical Service.

`dem.cdr` *Calculate CDR and ASDR*

Description

This function calculates the Crude Death Rate (CDR) and Age-Specific Death Rates (ASDR).

Usage

```
dem.cdr(
  data,
  type,
  age_col = NULL,
  population_col,
  deaths_col,
  verbose = FALSE
)
```

Arguments

`data`	A data frame containing demographic data.
`type`	A string or vector specifying the type(s) of death rate calculation. Options: "CDR", "ASDR". Use "all" to calculate both.
`age_col`	The column name for age groups (used for "ASDR").
`population_col`	The column name for total population.
`deaths_col`	The column name for total deaths.
`verbose`	Logical. If TRUE, prints progress messages during execution.

Value

A list of named outputs for the requested death rate calculations.

References

Preston, S. H., Heuveline, P., & Guillot, M. (2001). *Demography: Measuring and Modeling Population Processes*. Oxford: Blackwell Publishers. ISBN 978-0631226161. (Chapter 2: Age-Specific Rates and Probabilities.)

Newell, C. (1988). *Methods and Models in Demography*. New York: Guilford Press.

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.). San Diego: Elsevier Academic Press. ISBN 978-0126419559. (Chapter on mortality measures.)

Examples

```
data <- data.frame(
  age = c("0-4", "5-9", "10-14"),
  population = c(10000, 12000, 11000),
  deaths = c(100, 50, 30)
)
dem.cdr(data, type = "all", age_col = "age",
                    population_col = "population", deaths_col = "deaths")
```

`dem.fert` *Calculate Fertility Metrics*

Description

This function calculates various fertility metrics (CBR, GFR, ASFR, TFR) based on user input and a dataframe with demographic data.

Usage

```
dem.fert(
  data,
  type,
  age_col = NULL,
  population_col,
  women_col = NULL,
  births_col,
  age_interval = 5,
  verbose = FALSE
)
```

Arguments

`data`	A dataframe containing demographic data.
`type`	A string or vector specifying the type(s) of fertility calculation. Options: "CBR", "GFR", "ASFR", "TFR". Use "all" to calculate all metrics.
`age_col`	The column name for age groups (used for "ASFR" and "TFR").
`population_col`	The column name for total population.
`women_col`	The column name for number of women (used for "GFR", "ASFR", and "TFR").
`births_col`	The column name for live births.
`age_interval`	Width (in years) of the age groups, used to compute TFR from the ASFRs (default 5, for standard 5-year age groups).
`verbose`	Logical. If TRUE, prints progress messages during execution.

Value

A list of named outputs for the requested fertility calculations and the modified dataframe with ASFR.

References

Preston, S. H., Heuveline, P., & Guillot, M. (2001). *Demography: Measuring and Modeling Population Processes*. Oxford: Blackwell Publishers. ISBN 978-0631226161. (Chapter 5: Fertility and Reproduction.)

Newell, C. (1988). *Methods and Models in Demography*. New York: Guilford Press.

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.). San Diego: Elsevier Academic Press. ISBN 978-0126419559. (Chapter on fertility measures.)

Examples

```
demo_data <- data.frame(
  age = c("15-19", "20-24", "25-29"),
  population = c(10000, 12000, 11000),
  women = c(5000, 6000, 5500),
  live_births = c(200, 400, 300)
)
# Calculate ASFR
dem.fert(demo_data, type = "ASFR", age_col = "age",
             population_col = "population", women_col = "women", births_col = "live_births")
# Calculate all metrics
dem.fert(demo_data, type = "all", age_col = "age",
             population_col = "population", women_col = "women", births_col = "live_births")
```

---

dem.mmr *Calculate Maternal Mortality Metrics*

Description

This function calculates the maternal mortality rate (maternal deaths per 100,000 women of reproductive age) and the maternal mortality ratio (maternal deaths per 100,000 live births).

Usage

```
dem.mmr(data, deaths_col, births_col, pop_women, verbose = FALSE)
```

Arguments

data	A dataframe containing demographic data.
deaths_col	The column name for maternal deaths.
births_col	The column name for live births.
pop_women	The column containing the number of women of reproductive age.
verbose	Logical. If TRUE, prints progress messages during execution.

Value

An object of class dem_mmr: a list with results (the maternal mortality rate MMR per 100,000 women, and the maternal mortality ratio MMR_Ratio per 100,000 live births) and modified_data (the input data with per-row mm_rate and mm_ratio columns).

References

World Health Organization (1992). *International Statistical Classification of Diseases and Related Health Problems, 10th Revision (ICD-10)*. Geneva: World Health Organization. (Definitions of maternal death and maternal mortality measures.)

Preston, S. H., Heuveline, P., & Guillot, M. (2001). *Demography: Measuring and Modeling Population Processes*. Oxford: Blackwell Publishers. ISBN 978-0631226161.

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.). San Diego: Elsevier Academic Press. ISBN 978-0126419559. (Chapter on health demography and maternal mortality.)

Examples

```
demo_data <- data.frame(
  age=c("15-24", "25-34", "35-44"),
  maternal_deaths = c(5, 10, 15),
  live_births = c(5000, 7000, 6000),
  women =c(20000, 22000, 32000)
)
dem.mmr(demo_data, deaths_col = "maternal_deaths", births_col = "live_births", pop_women = "women")
```

---

dm.chm *Calculate Child Mortality Metrics by Age*

Description

This function calculates child mortality metrics (neonatal, infant, child, under-5) based on single-age data. It requires columns for age, denominator (live births or population), and deaths.

Usage

```
dm.chm(
  data,
  age_col,
  live_births,
  deaths_col,
  type = c("neonatal", "infant", "child", "under5"),
  age_in_months = NULL,
  verbose = FALSE
)
```

Arguments

data	A dataframe containing demographic data.
age_col	The column name representing the age (in years or months for neonates).
live_births	The column name for the denominator at each age (live births, or mid-year population if births are unavailable).

deaths_col	The column name representing deaths at each age.
type	The type of mortality calculation to perform: "neonatal", "infant", "child", or "under5".
age_in_months	(Optional) The column name for age in months for neonatal mortality.
verbose	Logical. If TRUE, prints progress messages during execution.

Details

The result is expressed per 1,000 of the supplied denominator. If the denominator is live births (as is conventional for neonatal and infant mortality), the result is a rate per 1,000 live births. If a mid-year population is supplied instead (often the only option for the child and under-5 groups from census data), the result is a death *rate* per 1,000 population, which is not identical to the cohort probability of dying (e.g. 5q0) reported by survey programmes such as the DHS.

Value

A numeric value representing the mortality rate per 1,000 live births.

References

Pollard, A. H., Yusuf, F., & Pollard, G. N. (1990). *Demographic Techniques* (3rd ed.). Sydney: Pergamon Press.

Preston, S. H., Heuveline, P., & Guillot, M. (2001). *Demography: Measuring and Modeling Population Processes*. Oxford: Blackwell Publishers. ISBN 978-0631226161. (Chapters 1-2: rates versus probabilities of dying.)

Examples

```
demo_data <- data.frame(
  age = 0:10,
  population = c(1000, 950, 900, 850, 800, 750, 700, 650, 600, 550, 500),
  deaths = c(30, 20, 15, 10, 5, 2, 1, 1, 0, 0, 0)
)
dm.chm(demo_data,age_col = "age",
                                  live_births = "population",
                                  deaths_col = "deaths",
                                  type = "under5")
```

---

first_coef *Karup-King coefficients for the first age group*

Description

The standard Karup-King third-difference osculatory interpolation multipliers used to split the *first* 5-year age group into single years. Rows are the five single-year ages within the group; columns are the weights applied to the first, second, and third 5-year groups.

Usage

```
first_coef
```

Format

A 5 x 3 data frame of numeric coefficients.

Source

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.), Appendix C.

---

gphc2010 *Ghana 2010 Population and Housing Census mortality data*

Description

Population and deaths by age group from the Ghana 2010 Population and Housing Census, used to demonstrate `lifetable()`.

Usage

```
gphc2010
```

Format

A data frame with 21 rows and 3 variables:

Age Numeric. Lower bound of the age group (0, 1, 5, 10, ...).

Pop Numeric. Population in the age group.

Deaths Numeric. Deaths in the age group.

Source

Ghana Statistical Service, 2010 Population and Housing Census.

`interpolation` *Smart Interpolation Function*

Description

Performs interpolation on numeric data, supporting linear, quadratic, Lagrange, and cubic spline methods. Can handle both vector and dataframe inputs.

Usage

```
interpolation(
  x,
  y,
  x_new,
  data = NULL,
  y_cols = NULL,
  method = c("auto", "linear", "quadratic", "lagrange", "spline"),
  verbose = FALSE
)
```

Arguments

`x`	Numeric vector of x-values.
`y`	Numeric vector of y-values (ignored if using a dataframe).
`x_new`	Numeric vector of new x-values to interpolate.
`data`	Optional dataframe containing y-values to interpolate.
`y_cols`	Character vector of column names in `data` containing y-values.
`method`	Character. Interpolation method: `"auto"`, `"linear"`, `"quadratic"`, `"lagrange"`, or `"spline"`. Default is `"auto"`: linear for 2 points, quadratic for 3, and a natural cubic spline for more than 3 points. High-degree Lagrange polynomials are numerically unstable for many points (Runge phenomenon), so `"lagrange"` is only used when requested explicitly.
`verbose`	Logical. If `TRUE`, prints detailed status messages to the console during interpolation.

Value

If vectors `x` and `y` are provided, returns a named numeric vector of interpolated values. If a dataframe is provided, returns the original dataframe with additional interpolated columns.

References

Shryock, H. S., Siegel, J. S., & Larmon, E. A. (1973). *The Methods and Materials of Demography*. Washington, DC: US Bureau of the Census. (Appendix C: Interpolation and graduation.)

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.). San Diego: Elsevier Academic Press. ISBN 978-0126419559. (Appendix C: Selected General Methods.)

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). *Numerical Recipes: The Art of Scientific Computing* (3rd ed.). Cambridge: Cambridge University Press. ISBN 978-0521880688. (Chapter 3: Interpolation and Extrapolation.)

Examples

```
# Example 1: Interpolating between points (vector input)
x <- c(1, 2, 3, 4, 5)
y <- c(2, 4, 6, 8, 10)
x_new <- c(2.5, 3.5, 4.5)
interpolation(x, y, x_new, method = "linear")

# Example 2: Interpolating for a dataframe
data <- data.frame(
  category = c("A", "B", "C"),
  y1 = c(2, 5, 9),
  y2 = c(3, 6, 10)
)
x_new <- c(2.5, 3.5)
interpolation(x = c(1, 2), data = data, y_cols = c("y1", "y2"), x_new = x_new, method = "spline")
```

---

karup_king *Split Age Groups into Single-Year Ages*

Description

This function takes grouped population data (e.g., 5-year age groups) and distributes the population into single-year age groups using the Karup-King third-difference osculatory interpolation coefficients. The user can specify the column containing age groups, one or more population columns, and (optionally) custom coefficient matrices for the first, middle, and last groups.

Usage

```
karup_king(
  df,
  age_col = "age_group",
  pops = "population",
  first_coef = NULL,
  middle_coef = NULL,
  last_coef = NULL,
  verbose = FALSE
)
```

Arguments

df	A data frame containing grouped population data.
age_col	A string specifying the column in df that contains age group labels in the form "start-end" (e.g., "15-19"). Open-ended groups (e.g. "85+") are not supported and should be excluded or closed before calling.
pops	A string or vector of strings specifying one or more columns in df that contain population data.
first_coef	A 5x3 matrix of coefficients for splitting the first age group. Defaults to the packaged Karup-King multipliers (first_coef).
middle_coef	A 5x3 matrix of coefficients for splitting middle age groups. Defaults to the packaged Karup-King multipliers (middle_coef).
last_coef	A 5x3 matrix of coefficients for splitting the last age group. Defaults to the packaged Karup-King multipliers (last_coef).
verbose	Logical. If TRUE, prints detailed status messages to the console during interpolation.

Details

Each coefficient matrix has 5 rows (one per single year within the 5-year group) and 3 columns (weights applied to the neighbouring 5-year groups). The defaults are the standard Karup-King multipliers (Siegel & Swanson 2004, Appendix C): first_coef for the first group (columns: current, next, next+1), middle_coef for interior groups (columns: previous, current, next), and last_coef for the last group (columns: previous-1, previous, current).

Value

A data frame with single-year ages and corresponding population estimates for each specified population column. Ages beyond the range of the input age groups (overflow ages) are removed.

References

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.). San Diego: Elsevier Academic Press. ISBN 978-0126419559. (Appendix C: Karup-King osculatory interpolation multipliers.)

Shryock, H. S., Siegel, J. S., & Larmon, E. A. (1973). *The Methods and Materials of Demography*. Washington, DC: US Bureau of the Census. (Appendix C: Interpolation and graduation.)

Examples

```
data(data)
head(data)

# Split the 5-year age groups into single years for the 2021 population,
# using the packaged Karup-King coefficients:
single <- karup_king(df = data, age_col = "age_col", pops = "2021")
head(single)
```

```
# Population totals are (approximately) preserved:
sum(data[["2021"]])
sum(single[["2021"]])
```

`last_coef` *Karup-King coefficients for the last age group*

Description

The standard Karup-King third-difference osculatory interpolation multipliers used to split the *last* 5-year age group into single years. Rows are the five single-year ages within the group; columns are the weights applied to the antepenultimate, penultimate, and last 5-year groups.

Usage

```
last_coef
```

Format

A 5 x 3 data frame of numeric coefficients.

Source

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.), Appendix C.

`lifetable` *lifetable*

Description

This function computes a complete life table based on given mortality rates or raw mortality data. It takes a dataframe with the necessary inputs and computes standard life table metrics.

Usage

```
lifetable(
  data,
  age = "Age",
  nMx = NULL,
  pop = NULL,
  Dx = NULL,
  sex = c("male", "female"),
  radix = 1e+05,
  verbose = FALSE
)
```

Arguments

`data`	A data frame containing age-specific mortality data.
`age`	String: name of the column with the lower bound of each age group, e.g., 0, 1, 5, 10, ...
`nMx`	String (optional): name of the column with observed mortality rates. If not provided, `pop` and `Dx` must be given to compute it.
`pop`	String (optional): name of the column with total population (exposure) in each age group. Used to compute `nMx` if not given.
`Dx`	String (optional): name of the column with deaths in each age group. Used to compute `nMx` if not given.
`sex`	Character: "male" (default) or "female". Determines the Coale-Demeny coefficients used for a_0 and ${}_4a_1$.
`radix`	Numeric: the life table radix, i.e. survivors at exact age 0 (default 100,000).
`verbose`	Logical. If `TRUE`, prints detailed status messages to the console during computation.

Details

Interval widths n are derived from the age column (e.g. ages 0, 1, 5, 10, ... give widths 1, 4, 5, ...). Probabilities of dying are obtained from the death rates with the standard conversion ${}_nq_x = n \cdot {}_nM_x/(1 + (n - {}_na_x) \cdot {}_nM_x)$. The average person-years lived by those dying in the interval (${}_na_x$) use the Coale-Demeny model values for ages 0 and 1-4 (which depend on `sex` and the level of infant mortality; see Preston et al. 2001, Table 3.3), $n/2$ for other closed intervals, and $1/{}_nM_x$ for the open-ended interval. Person-years lived are ${}_nL_x = n \cdot l_{x+n} + {}_na_x \cdot {}_nd_x$ for closed intervals and l_x/M_x for the open-ended interval.

Value

A list containing:

`metrics` A list with the life expectancy at birth and the total life table deaths.

`lifetable` A dataframe of the complete life table.

References

Preston, S. H., Heuveline, P., & Guillot, M. (2001). *Demography: Measuring and Modeling Population Processes*. Oxford: Blackwell Publishers. ISBN 978-0631226161. (Chapter 3: The Life Table and Single Decrement Processes; Table 3.3 for the Coale-Demeny separation factors.)

Chiang, C. L. (1984). *The Life Table and Its Applications*. Malabar, FL: Robert E. Krieger Publishing.

Coale, A. J., & Demeny, P. (1983). *Regional Model Life Tables and Stable Populations* (2nd ed.). New York: Academic Press. (Source of the a0 and 4a1 separation factors.)

Examples

```
# Example computing rates from deaths and population
data(gphc2010)
res <- lifetable(data = gphc2010, age = "Age", pop = "Pop", Dx = "Deaths")
res$metrics
head(res$lifetable)
```

`lifetable_nqx` *Compute Life Table Using Age and nqx*

Description

This function computes a life table from a data frame containing age groups and age-interval probabilities of dying (`nqx`). It calculates key life table components such as `lx`, `dx`, `Lx`, `Tx`, and `ex`.

Usage

```
lifetable_nqx(data, age, nqx, l0 = 1e+05, verbose = FALSE)
```

Arguments

`data`	A data frame containing age and nqx columns.
`age`	A string specifying the column name for age groups (lower bounds, e.g. 0, 1, 5, 10, ...). Ages must be strictly increasing.
`nqx`	A string specifying the column name for the probabilities of dying between age `x` and `x+n` (`nqx`). The last value should be 1 (open-ended group).
`l0`	Radix of the life table; survivors at age 0 (default = 100,000).
`verbose`	Logical. If `TRUE`, prints detailed status messages to the console during computation.

Details

Interval widths n are derived from the age column (e.g. ages 0, 1, 5, 10, ... give widths 1, 4, 5, ...). Person-years lived in each closed interval are approximated as ${}_nL_x = n(l_x - d_x/2)$, i.e. deaths are assumed to occur at the midpoint of the interval. For the open-ended last age group, person-years are approximated with the classical rule of thumb $L_{x+} = l_x \log_{10}(l_x)$ (equivalent to $e_x \approx \log_{10} l_x$), which is used when no death rate for the open interval is available. If death rates are available, `lifetable()` provides a more precise treatment of the open interval.

Value

A data frame with the computed life table, including:

`Age` Age groups.

`n` Width of each age interval (`Inf` for the last, open-ended group).

`nqx` Probability of dying between age `x` and `x+n`.

`lx` Number of survivors at the beginning of each age group.

`dx` Number of deaths within each age group.

`Lx` Person-years lived within each age group.

`Tx` Cumulative person-years lived above a specific age group.

`ex` Life expectancy at the beginning of each age group.

References

Preston, S. H., Heuveline, P., & Guillot, M. (2001). *Demography: Measuring and Modeling Population Processes*. Oxford: Blackwell Publishers. ISBN 978-0631226161. (Chapter 3: The Life Table and Single Decrement Processes.)

Chiang, C. L. (1984). *The Life Table and Its Applications*. Malabar, FL: Robert E. Krieger Publishing.

Examples

```
# Example data
data <- data.frame(
  age = c(0, 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85),
  nqx = c(0.05, 0.01, 0.005, 0.002, 0.003, 0.004, 0.005, 0.007, 0.010, 0.015,
          0.020, 0.030, 0.050, 0.070, 0.100, 0.150, 0.200, 0.300, 1.000)
)
life_table <- lifetable_nqx(data, age = "age", nqx = "nqx")
print(life_table)
```

---

math_project *Hierarchical Subnational Population Projection*

Description

Projects lower-level geographies (e.g., Districts) while constraining them to higher-level control totals (e.g., Regions or National) provided within the dataset.

Usage

```
math_project(
  data,
  sub_group_col,
  parent_group_col = NULL,
  pop_cols,
  time_vals,
  target_times,
 method = c("linear", "exponential", "geometric", "constant_share", "shift_share",
    "share_of_growth"),
```

```
  parent_totals_cols = NULL,
  verbose = FALSE
)
```

Arguments

`data`	Dataframe containing the subnational population data.
`sub_group_col`	Character. Column name for the lower-level geography (e.g., "District").
`parent_group_col`	Character. Optional. Column name for the higher-level geography (e.g., "Region"). If NULL, assumes a single national parent group.
`pop_cols`	Character vector of length 2. Column names for Base Year and Launch Year populations.
`time_vals`	Numeric vector of length 2. The years corresponding to pop_cols.
`target_times`	Numeric vector. The future years to project.
`method`	Character. The projection method.
`parent_totals_cols`	Character vector. Optional. Column names in `data` containing the pre-calculated projected totals for the parent group at `target_times`. Must match the length of `target_times`.
`verbose`	Logical. If `TRUE`, prints detailed status messages to the console during projection.

Value

A dataframe appending the projected columns to the original data.

References

Smith, S. K., Tayman, J., & Swanson, D. A. (2013). *A Practitioner's Guide to State and Local Population Projections*. Dordrecht: Springer. doi:10.1007/9789400775510 (Trend extrapolation, constant-share, shift-share, and share-of-growth methods.)

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.). San Diego: Elsevier Academic Press. ISBN 978-0126419559. (Chapter on population projections.)

Rowland, D. T. (2003). *Demographic Methods and Concepts*. Oxford: Oxford University Press. ISBN 978-0198752639. (Chapter on population projections.)

Examples

```
districts <- data.frame(
  Region = c("A", "A", "B", "B"),
  District = c("A1", "A2", "B1", "B2"),
  pop2010 = c(10000, 20000, 15000, 5000),
  pop2020 = c(12000, 26000, 16500, 5500)
)
```

```
# Exponential extrapolation of each district
math_project(districts,
  sub_group_col = "District", parent_group_col = "Region",
  pop_cols = c("pop2010", "pop2020"), time_vals = c(2010, 2020),
  target_times = c(2025, 2030), method = "exponential"
)
```

`middle_coef` *Karup-King coefficients for middle age groups*

Description

The standard Karup-King third-difference osculatory interpolation multipliers used to split *interior* 5-year age groups into single years. Rows are the five single-year ages within the group; columns are the weights applied to the previous, current, and next 5-year groups.

Usage

```
middle_coef
```

Format

A 5 x 3 data frame of numeric coefficients.

Source

Siegel, J. S., & Swanson, D. A. (Eds.). (2004). *The Methods and Materials of Demography* (2nd ed.), Appendix C.

`project_population` *Basic Probabilistic Population Projection*

Description

This function projects population trajectories for multiple administrative regions using a simple stochastic (simulation-based) exponential growth model, inspired by the probabilistic projection literature (e.g. Raftery et al. 2012), but without a full Bayesian model.

For each region, `num_samples` growth-rate trajectories are simulated. The annual growth rate of each trajectory is the sum of a random baseline, a fertility effect (increasing in TFR), a mortality effect (decreasing in the death rate), a migration effect (with the sign of net migration), and random noise. Each trajectory is propagated independently through all years, so that uncertainty accumulates over the projection horizon, and the results are summarised per region and year.

Usage

```
project_population(
  data,
  future_year,
  base_year,
  region_var = "region",
  subregion_var = "subregion",
  base_pop_var = "base_pop",
  TFR_var = "TFR",
  death_rate_var = "death_rate",
  net_migration_var = "net_migration",
  num_samples = 2000,
  random_seed = 42,
  verbose = FALSE
)
```

Arguments

`data`	A data frame containing projection parameters.
`future_year`	Numeric. The target year for the projection.
`base_year`	Numeric. The base year for the projection.
`region_var`	Character. Column name for the region.
`subregion_var`	Character. Column name for the subregion.
`base_pop_var`	Character. Column name for the base population.
`TFR_var`	Character. Column name for the Total Fertility Rate (TFR).
`death_rate_var`	Character. Column name for the death rate.
`net_migration_var`	Character. Column name for net migration.
`num_samples`	Integer. Number of simulation samples (default: 2000).
`random_seed`	Integer. Random seed for reproducibility (default: 42).
`verbose`	Logical. If `TRUE`, prints detailed status messages to the console during projection.

Value

A data frame with one row per region, subregion, and year, containing the summary statistics of the simulated population trajectories: `lower` (25th percentile), `median`, `mean`, and `upper` (75th percentile).

References

Raftery, A. E., Li, N., Sevcikova, H., Gerland, P., & Heilig, G. K. (2012). Bayesian probabilistic population projections for all countries. *Proceedings of the National Academy of Sciences*, 109(35), 13915-13921. doi:10.1073/pnas.1211452109 (Inspiration for the stochastic/probabilistic projection logic.)

Preston, S. H., Heuveline, P., & Guillot, M. (2001). *Demography: Measuring and Modeling Population Processes*. Oxford: Blackwell Publishers. ISBN 978-0631226161. (Chapter 6: Population Projections.)

Rowland, D. T. (2003). *Demographic Methods and Concepts*. Oxford: Oxford University Press. ISBN 978-0198752639. (Chapter on population projections.)

Examples

```
data(region2000)
proj <- project_population(region2000,
  base_year = 2000, future_year = 2009,
  TFR_var = "TFR", base_pop_var = "base_pop",
  region_var = "Country", death_rate_var = "death_rate",
  net_migration_var = "net_migration",
  subregion_var = "Region",
  num_samples = 1000
)
head(proj)
```

---

region2000 *Regional projection parameters, Ghana 2000*

Description

Base-year demographic parameters for the regions of Ghana in 2000, used to demonstrate `project_population()`.

Usage

```
region2000
```

Format

A data frame with 16 rows and 6 variables:

Country Character. Country name.

Region Character. Region name.

base_pop Numeric. Base-year population.

TFR Numeric. Total fertility rate.

death_rate Numeric. Crude death rate.

net_migration Numeric. Net migration (persons).

Source

Derived from Ghana Statistical Service census publications.

standards *Brass standard life tables*

Description

Survivorship values $l(x)$ (radix 1) and their Brass logits for the African and General United Nations standards, used by `brass_logit()`.

Usage

```
standards
```

Format

A data frame with 24 rows and 5 variables:

Age Numeric. Exact age x.

GeneralUN Numeric. l(x) of the General UN standard.

GeneralUN_logit Numeric. Brass logit of the General UN standard.

African Numeric. l(x) of the African standard.

African_logit Numeric. Brass logit of the African standard.

Source

Brass, W. (1971). On the scale of mortality. In *Biological Aspects of Demography*. London: Taylor & Francis. United Nations model life table publications.

testdata *District-level demographic test data*

Description

Simulated district-level demographic panel data (population, births, deaths, migration) by age group and sex, for testing subnational projection workflows such as `math_project()`.

Usage

```
testdata
```

Format

A data frame with 1224 rows and 10 variables:

region Character. Region name.

district Character. District name.

year Numeric. Calendar year.

age_group Character. Age group label.

sex Character. "Male" or "Female".

population Numeric. Population count.

births Numeric. Live births.

deaths Numeric. Deaths.

in_migration Numeric. In-migrants.

out_migration Numeric. Out-migrants.

Source

Simulated data.

Index