Package: qlifetable (via r-universe)

June 29, 2024

Title Managing and Building of Quarterly Life Tables

Version 0.0.2-5

Maintainer Jose M. Pavía <jose.m.pavia@uv.es>

Description Manages, builds and computes statistics and datasets for the construction of quarterly (sub-annual) life tables by exploiting micro-data from either a general or an insured population. References: Pavía and Lledó (2022) <doi:10.1111/rssa.12769>. Pavía and Lledó (2023) <doi:10.1017/asb.2023.16>. Acknowledgements: The authors wish to thank Consellería de Educación, Universidades y Empleo, Generalitat Valenciana (grant AICO/2021/257), Ministerio de Ciencia e Innovación (grant PID2021-128228NB-I00) and Fundación Mapfre (grant 'Modelización espacial e intra-anual de la mortalidad en España. Una herramienta automática para el cálculo de productos de vida') for supporting this research.

License EPL

Encoding UTF-8

Imports stats, methods

Suggests ggplot2, scales, gridExtra

LazyData true

RoxygenNote 7.2.3

Depends R (>= 3.5.0)

NeedsCompilation no

Author Jose M. Pavía [aut, cre] (<https://orcid.org/0000-0002-0129-726X>), Josep Lledó [aut] (<https://orcid.org/0000-0002-7475-8549>)

Repository CRAN

Date/Publication 2024-06-28 07:30:07 UTC

Contents

`annual2quarterly` *Estimates quarterly life tables*

Description

Given an annual life table and a set of estimates seasonal-ageing indexes, estimates the four quarterly life tables associated to the annual life table.

Usage

```
annual2quarterly(table.a, SAIs, mx = FALSE, min.age = 0, max.age = 100)
```

Arguments

`table.a` A data.frame corresponding to the reference annual life table. The life table can be defined via via death probabilities (‘qx‘, default) or death rates (‘mx‘). The first column of ‘table.a‘ refers to ‘age‘, and the second column to either ‘mx‘ rates or ‘qx‘ probabilities. In case of using death probabilities (‘qx‘), ‘table.a‘ can have an optional third column, which refers to the the average number of years lived for those dying with age x, ‘ax‘. If this last column is missing ax is assumed to be constant and equal to 0.5.

SAIs	An object output of the compute_SAI function.
mx	A ‘TRUE/FALSE‘ argument informing whether ‘table.a‘ is either an annual life table of death rates or an annual table of death probabilities. Default, ‘FALSE‘.
min.age	A non-negative integer informing about the initial minimal age for which quarterly tables must be computed. This minimum age can be increased depending on the ages for which there are values in ‘table.a‘ and in ‘SAIs‘. Default, 0.
max.age	A positive integer informing about the initial maximum age for which quarterly tables must be computed. This maximum age can be decreased depending on the ages for which there are values in ‘table.a‘ and in ‘SAIs‘. Default, 100.

Value

A data frame with ten columns

age	Integer age to which death rates and probabilities corresponds.
quarter.age	Age quarter to which death rates and probabilities corresponds.
mx.quarter.birth.1	Death rates corresponding to people born during the first quarter of the year.
mx.quarter.birth.2	Death rates corresponding to people born during the second quarter of the year.
mx.quarter.birth.3	Death rates corresponding to people born during the third quarter of the year.
mx.quarter.birth.4	Death rates corresponding to people born during the four quarter of the year.
qx.quarter.birth.1	Death Probabilites corresponding to people born during the first quarter of the year.
qx.quarter.birth.2	Death Probabilites corresponding to people born during the second quarter of the year.
qx.quarter.birth.3	Death Probabilites corresponding to people born during the third quarter of the year.
qx.quarter.birth.4	Death Probabilites corresponding to people born during the four quarter of the year.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

Examples

```
# This can take a while
 t.stock <- time_exposed_stock(pop_2006$date.birth, 2006, "forward")
 t.stock <- t.stock[t.stock$age <= 100, ]
 temp <- quarterly_variables(death_2006$date.birth, death_2006$date.death)
 e.death <- count_events_quarter(temp)
 e.death <- e.death[e.death$age <= 100, ]
 t.birth <- time_exposed_newborns(birth_2006$date.birth)
 out <- crude_mx(t.stock, e.death, time.birth = t.birth)
 SAI.example <- compute_SAI(out, out)
 dx <- tapply(e.death$number.events, e.death$age, sum)
 Lax <- tapply(t.stock$time.exposed, t.stock$age, sum)
 table.ex <- data.frame(age = 0:100, mx.a = dx/Lax)
 example <- annual2quarterly(table.a = table.ex, SAIs = SAI.example, mx = TRUE)
```

`birth_2006` *Half of male births recorded in the Comunitat Valenciana (Spain) during 2006.*

Description

Data frame containing the dates of birth of a random sample composed by half of all male babies born in the region of Comunitat Valenciana (Spain) during 2006.

Usage

```
data(birth_2006)
```

Format

A data frame containing 13,626 rows and 1 column:

date.birth Dates of birth in format "yyyy-mm-dd".

Source

National Statistical Institute, https://www.ine.es/.

`compute_SAI` *Estimates seasonal-ageing indexes from quarterly tables of crude rates estimates.*

Description

Given a set of quarterly tables of crude rates estimates corresponding to several years (periods), this function computes their corresponding seasonal-ageing indexes.

Usage

```
compute_SAI(x, y, ..., margins = FALSE, min.age = 0, max.age = 100)
```

Arguments

`x`	A data.frame/list output of either `crude_mx`, `crude_mx_sh2` or `crude_mx_sh3` functions corresponding to a period (typically a year).
`y`	A data.frame/list output of either `crude_mx`, `crude_mx_sh2` or `crude_mx_sh3` functions corresponding to another period (typically a different year than the one corresponding to ‘x‘).
`...`	Further output(s) of either `crude_mx`, `crude_mx_sh2` or `crude_mx_sh3` functions corresponding to another period(s) (typically different year(s) than the ones corresponding to ‘x‘ and ‘y‘).
`margins`	A ‘TRUE/FALSE‘ argument informing whether or not the marginal seasonal and ageing indexes should be also computed. Default, ‘FALSE‘.
`min.age`	A non-negative integer informing about the initial minimal age for which SAIs must be computed. This minimum age can be increased depending on the ages for which there are crude mx estimates (exposed-at-risk) in the objects ‘x‘, ‘y‘, ‘...‘. Default, 0.
`max.age`	A positive integer informing about the initial maximum age for which SAIs must be computed. This maximum age can be decreased depending on the maximum ages for which there are crude mx estimates (exposed-at-risk) in the objects ‘x‘, ‘y‘, ‘...‘. Default, 100.

Value

When ‘margins = FALSE‘ a data frame with the (raw, normalized and linearized) estimated seasonal-ageing indexes (SAI) corresponding to the set of integer ages determined by ‘min.age‘ and ‘max.age‘ and the exposed-at-risk in the mx crude estimates for each combination of age and calendar quarter. The data frame has the following components:

`age`	Integer age to which the SAIs corresponds.
`quarter.age`	Age quarter to which the SAIs corresponds.
`quarter.calendar`	Calendar (time, season) quarter to which the SAIs corresponds.

`SAI.raw`	Estimates of raw seasonal-ageing indexes for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.
`SAI.norm`	Estimates of seasonal-ageing indexes, attained after normalizing raw SAIs estimates, for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.
`SAI.lin`	Final estimates of seasonal-ageing indexes, attained after linearizing normalized SAIs estimates, for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

When ‘margins = TRUE‘ the output is a list with three data frames ‘SAI‘, ‘SAI.age‘ and ‘SAI.quarter‘. ‘SAI‘ is defined as just described above. ‘SAI.age‘ and ‘SAI.quarter‘ contains, in a similar vein than ‘SAI‘, the estimated marginal SAIs, corresponding to, respectively, the age quarter and the calendar (time, season) quarter.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

Examples

```
# This can take a while
 t.stock <- time_exposed_stock(pop_2006$date.birth, 2006, "forward")
 t.stock <- t.stock[t.stock$age <= 100, ]
 temp <- quarterly_variables(death_2006$date.birth, death_2006$date.death)
 e.death <- count_events_quarter(temp)
 e.death <- e.death[e.death$age <= 100, ]
 t.birth <- time_exposed_newborns(birth_2006$date.birth)
 out <- crude_mx(t.stock, e.death, time.birth = t.birth)
 SAI.example <- compute_SAI(out, out)


# Fast example
dates.b <- c("2017-05-13", "2018-04-12", "2018-12-01")
t.stock <- time_exposed_stock(dates.b, year = 2020, type = "backward")
dates.bd <- c("2018-04-12")
dates.d <- c("2020-05-23")
x <- quarterly_variables(dates.bd, dates.d)
e.death <- count_events_quarter(x)
t.death <- time_exposed_outs(x)
out <- crude_mx(t.stock, e.death, t.death)
SAI.example <- compute_SAI(out, out)
```

coord_age *Time elapsed (in years) since last birthday and the date of the event*

Description

Computes the time(s) elapsed (in years) between the date(s) of last birthday and the date(s) of event(s) for each member of a population. The age coordinate(s) corresponding to an 1x1-year Lexis diagram, using by default the same length of year employed to compute the (related) time coordinate(s).

Usage

```
coord_age(
  date.birth,
  date.event,
  random.b = TRUE,
  random.e = TRUE,
  constant.age.year = FALSE
)
```

Arguments

`date.birth` A character vector with the dates of birth in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.b = FALSE‘, or a random hour by default.

`date.event` A character vector with the dates of events in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.e = FALSE‘, or a random hour, by default. This vector must have either length 1, when the aim is to compute for all the members of the population the age coordinate in an 1x1-year Lexis diagram in the same temporal point, or the same length as ‘date.birth‘, when the aim is to compute for each member of the population the age coordinate in the moment of the event (e.g., death).

`random.b` A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the birth occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.birth‘ even if those have been declared. By default, TRUE.

`random.e` A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the event occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.event‘ even if those have been declared. By default, TRUE.

`constant.age.year`
A ‘TRUE/FALSE‘ argument indicating whether the length of the year should be constant, 365.25 days, or variable, depending on the time lived for the person in

each year since her/his dates of birth and event. By default, FALSE. The advantage of using a non-constant (person-dependent) length of year is congruence when estimating time exposed at risk: in each year the time exposed along the time and age axes will coincide.

Value

A numeric vector of the same length as data.birth

Note

If 'constant.age.year = FALSE' (default), the length of the year for each person is computed as a weighted average of the lengths of the years that the person has lived between his/her dates of birth and event using as weight the time lived for the person during each year.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

`coord_time`, `exact_age`

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
dates.e <- c("2002-03-23", "2009-04-12", "2019-01-01")
coord_age(dates.b, dates.e)
```

---

coord_time — *Time elapsed (in years) since the beginning of the year*

Description

Computes the time(s) elapsed (in years) between the beginning of the year and the date(s) of the event(s). The time coordinate(s) in a Lexis diagram.

Usage

```
coord_time(date.event, random.e = TRUE)
```

Arguments

date.event
: A character vector with the dates of events in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.e = FALSE‘, or a random hour, by default.

random.e
: A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the event occurs within the day is randomly selected. This overwrites "hour:min:secs" in ‘date.event‘ even if this has been declared.

Value

A numeric vector of the same length as data.event

Note

The length of the year is 365 days in non-leap years and 366 days in leap years.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

coord_age, exact_age

Examples

```
dates <- c("2002-03-23", "2009-04-12", "2019-01-01")
coord_time(dates)
dates <- "2019-01-01 14:00:00"
coord_time(dates, FALSE)
```

`count_events_quarter` *Data frame of number of events occurring in each Lexis-diagram quarter*

Description

Computes for each integer age and each combination of age and seasonal quarter the number of events occurring in the population. The computation is performed using the associated data frame of quarterly variables corresponding to the population obtained using the `quarterly_variables` function.

Usage

```
count_events_quarter(x)
```

Arguments

`x`	A data.frame output of the `quarterly_variables` function.

Value

A data frame with the number of events for each (potential) combination of integer age and age and season quarter of the input dataset. The data frame has the following components:

`age`	Integer age to which the time exposed at risk corresponds.
`quarter.age`	Age quarter to which the time exposed at risk corresponds.
`quarter.calendar`	Calendar (time, season) quarter to which the time exposed at risk corresponds.
`number.events`	Number of events that occurred during the quarter determined for the combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

Note

The structure of the dataset is similar to those obtained using the `time_exposed_outs`, `time_exposed_ins` and `time_exposed_stock` functions.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
dates.e <- c("2002-03-23", "2009-04-12", "2019-01-01")
x <- quarterly_variables(dates.b, dates.e)
out <- count_events_quarter(x)
```

---

crude_mx — *Data frame(s) of crude estimates of quarterly (and annual) central rates of mortality estimated from quarterly summary statistics.*

Description

Computes for a general/insured population crude estimates of quarterly (and annual) central rates of mortality by age (in a set of integer ages) for each combination of age and calendar quarter, given data sets of times exposed-at-risk of stocks of population and of number of deaths by each integer age for each combination of age and seasonal quarter, along with data sets of times exposed-at-risk (or non-exposed) of immigrants/new policies, emigrants/exits/deaths and/or newborns.

Usage

```
crude_mx(
  time.stock,
  events.death,
  time.death = NULL,
  time.outs = NULL,
  time.ins = NULL,
  time.birth = NULL,
  type = "forward",
  annual = FALSE
)
```

Arguments

`time.stock` A data frame containing the total exposure-at-risk times for the stock of the population/portfolio, measured either since the start of the year (‘type = "forward"‘) or from the end of the year (‘type = "backward"‘) throughout the target period (typically a year). This data frame must contain the times for a set of consecutive integer ages for each combination of age and season/calendar quarter. Typically, this data frame is an output of the function `time_exposed_stock` or a data frame with the same structure. The set of consecutive integer ages in this object determines the range of ages for which crude rates of mortality are estimated by the function.

`events.death` A data frame with the number of deaths recorded in the population/portfolio during the target period (typically a year) in a set of integer ages for each combination of age and season/calendar quarter. Typically, this is an output of the function `count_events_quarter` or a data frame with the same structure. If the range of ages does not cover the range of ages in ‘time.stock‘, zeros are imputed for the missing ages.

time.death
: A data frame containing either the total non-exposure-at-risk times (when ‘type = "forward"‘) or the total exposure-at-risk times (when ‘type = "backward"‘) during the target period of the deaths recorded in the population/portfolio. Typically, this data frame is an output of the function time_exposed_ins when ‘type = "forward"‘ or an output of the function time_exposed_outs when ‘type = "backward"‘ or a data frame with the same structure. Default, ‘NULL‘. If no ‘time.death‘ data.frame is provided the total (non-) exposure-at-risk times are computed using the information in ‘events.death‘ under the hypothesis of uniform distribution of deaths within each age-calendar quarter.

time.outs
: A data frame containing either the total non-exposure-at-risk times (when ‘type = "forward"‘) or the total exposure-at-risk times (when ‘type = "backward"‘) during the target period of the emigrants/exits recorded in the population/portfolio. Typically, this data frame is an output of the function time_exposed_ins when ‘type = "forward"‘ or an output of the function time_exposed_outs when ‘type = "backward"‘ or a data frame with the same structure. Default, ‘NULL‘.

time.ins
: A data frame containing either the total exposure-at-risk times (when ‘type = "forward"‘) or the total non-exposure-at-risk times (when ‘type = "backward"‘) during the target period (typically a year) of the immigrants/new policies recorded in the population/portfolio. Typically, this data frame is an output of the function time_exposed_ins when ‘type = "forward"‘ and an output of the function time_exposed_outs when ‘type = "backward"‘ or a data frame with the same structure. Default, ‘NULL‘.

time.birth
: A data frame containing the total exposure-at-risk times of the newborns recorded in the population during the target period. Typically, this data frame is an output of the function time_exposed_newborns or a data frame with the same structure. Default, ‘NULL‘.

type
: A character string informing if the total time exposed to risk of the stock of population/portfolio has been computed since the beginning of the year (‘"forward"‘) or from the end of the year (‘"backward"‘). Default, ‘"forward"‘.

annual
: A character string informing whether the annual crude central rates of mortality should also be computed. Default, ‘FALSE‘.

Value

When ‘annual = FALSE‘ a data frame with estimated crude central rates of mortality for the set of integer ages determined by ‘time.stock‘ for each combination of age and calendar quarter. The data frame has the following components:

age
: Integer age to which the crude central rate of mortality corresponds.

quarter.age
: Age quarter to which the crude central rate of mortality corresponds.

quarter.calendar
: Calendar (time, season) quarter to which the crude central rate of mortality corresponds.

exposed
: Total exposure-at-risk times in the target population for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

deaths
: Number of deaths in the target population for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

mx Estimated crude central rate of mortality corresponding to the combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

When ‘annual = TRUE‘ the output is a list with two data frames ‘mx.quarterly‘ and ‘mx.annual‘. ‘mx.quarterly‘ is a data frame with the estimated quarterly crude central rates of mortality as just described and ‘mx.annual‘ is the corresponding data frame with the estimated annual crude central rates of mortality.

Note

First, it is the responsibility of the user to assure that all the times exposed-at-risk and number of events correspond to the same target period (typically, the same year). Second, If the date of reference of the stock of population/portfolio is the beginning of the year (‘type = "forward"‘), the total time exposed at risk for each integer age in each age and calendar quarter is computed through: time_exposed_stock(P) + time_exposed_ins(I) - time_exposed_ins(E) - time_exposed_ins(D) + time_exposed_newborns(B). On the other hand, if the stock of population/portfolio is referenced at the end of the target year (‘type = "backward"‘), the total time exposed at risk for each age in each age and seasonal quarter is computed slightly different: time_exposed_stock(P) - time_exposed_outs(I) + time_exposed_outs(E) + time_exposed_outs(D). In the above expressions P, I, E, D and B represent, respectively, stocks of population/portfolio, immigrants/new policies, emigrants/exits, deaths and births after being transformed using the function `quarterly_variables`.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

`crude_mx_sh2`, `crude_mx_sh2`.

Examples

```
## Not run:
# Do not run, it can take a while
 t.stock <- time_exposed_stock(pop_2006$date.birth, 2006, "forward")
 t.stock <- t.stock[t.stock$age <= 100, ]
 temp <- quarterly_variables(death_2006$date.birth, death_2006$date.death)
 e.death <- count_events_quarter(temp)
 e.death <- e.death[e.death$age <= 100, ]
 t.birth <- time_exposed_newborns(birth_2006$date.birth)
 out <- crude_mx(t.stock, e.death, time.birth = t.birth)

## End(Not run)
dates.b <- c("2017-05-13", "2018-04-12", "2018-12-01")
```

```
t.stock <- time_exposed_stock(dates.b, year = 2020, type = "backward")
dates.bd <- c("2018-04-12")
dates.d <- c("2020-05-23")
x <- quarterly_variables(dates.bd, dates.d)
e.death <- count_events_quarter(x)
t.death <- time_exposed_outs(x)
out <- crude_mx(t.stock, e.death, t.death)
```

---

crude_mx_sh2 *Data frame(s) of crude estimates of quarterly (and annual) central rates of mortality estimated using shortcut 2.*

Description

Computes for a general/insured population crude estimates of quarterly (and annual) central rates of mortality by age (in a set of integer ages) for each combination of age and calendar quarter, using age aggregated data of population stocks and age-season quarterly counts of deaths by employing shortcut 2, based on equation (2.7), proposed in Pavia and Lledo (2023).

Usage

```
crude_mx_sh2(pop.start, pop.end, events.death, annual = FALSE)
```

Arguments

pop.start	A data frame, corresponding to the target population/portfolio, containing the stock of population/portfolio by age collected at the beginning of the year. This data frame must have two columns. The first column refers to age and the second to the number of people in the population/portfolio corresponding to each age.
pop.end	A data frame, corresponding to the target population/portfolio, containing the stock of population/portfolio by age collected at the end of the year. This data frame must have two columns. The first column refers to age and the second to the number of people in the population/portfolio corresponding to each age.
events.death	A data frame with the number of deaths recorded in the population/portfolio during the target year in a set of integer ages for each combination of age and season/calendar quarter. Typically, this is an output of the function count_events_quarter or a data frame with the same structure. If the range of ages in ‘events.death‘ does not cover the range of ages in the intersection of ages defined by ‘pop.start‘ and ‘pop.end‘, zeros are imputed for the missing ages.
annual	A character string informing whether the annual crude central rates of mortality should also be computed. Default, ‘FALSE‘.

Value

When ‘annual = FALSE‘ a data frame with estimated crude central rates of mortality for each combination of age and calendar quarter in the set of integer ages determined by the intersection of ages of ‘pop.start‘ and ‘pop.end‘. The data frame has the following components:

`age` Integer age to which the crude central rate of mortality corresponds.

`quarter.age` Age quarter to which the crude central rate of mortality corresponds.

`quarter.calendar`
Calendar (time, season) quarter to which the crude central rate of mortality corresponds.

`exposed` Total exposure-at-risk times in the target population, calulated according to shortcut 2, for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

`deaths` Number of deaths in the target population for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

`mx` Estimated crude central rate of mortality corresponding to the combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

When ‘annual = TRUE‘ the output is a list with two data frames ‘mx.quarterly‘ and ‘mx.annual‘. ‘mx.quarterly‘ is a data frame with the estimated quarterly crude central rates of mortality as just described and ‘mx.annual‘ is the corresponding data frame with the estimated annual crude central rates of mortality.

Note

It is the responsibility of the user to assure that both stocks of population (which determine the ages for which estimates are computed) and number of events correspond to the same year.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2023). Shortcuts for the construction of sub-annual life tables. *ASTIN Bulletin: The Journal of the International Actuarial Association*, 53(2), 332-350. doi:10.1017/asb.2023.16

See Also

`crude_mx`, `crude_mx_sh3`.

Examples

```
# This can take a while
 pop <- 2005 - as.numeric(substr(pop_2006$date.birth, 1 , 4))
 pop <- as.data.frame(table(pop))
 pop[, 1] <- as.numeric(as.character(pop[, 1]))
 temp <- quarterly_variables(death_2006$date.birth, death_2006$date.death)
 e.death <- count_events_quarter(temp)
 out <- crude_mx_sh2(pop, pop, e.death)


# Fast example
```

```
pop.1 <- data.frame(age = c(40, 41), people = c(4134, 4353))
pop.2 <- data.frame(age = c(40, 41), people = c(4250, 4213))
dates.b <- c("1980-04-12")
dates.d <- c("2020-08-23")
x <- quarterly_variables(dates.b, dates.d)
e.death <- count_events_quarter(x)
out <- crude_mx_sh2(pop.1, pop.2, e.death)
```

`crude_mx_sh3` *Data frame(s) of crude estimates of quarterly (and annual) central rates of mortality estimated using shortcut 3.*

Description

Computes for a general/insured population crude estimates of quarterly (and annual) central rates of mortality by age (in a set of integer ages) for each combination of age and calendar quarter, using quarterly age aggregated data of population stocks and age-season quarterly counts of deaths, immigrants/new policies (entries) and emigrants/lapses/expirations (exits) by employing shortcut 3, based on equation (2.9), proposed in Pavia and Lledo (2023).

Usage

```
crude_mx_sh3(
  pop.start,
  pop.end,
  events.death,
  events.out,
  events.in,
  annual = FALSE
)
```

Arguments

`pop.start` A data frame, corresponding to the target population/portfolio, containing by age the stock of population/portfolio by quarter collected at the beginning of the year. This data frame must have three columns. The first refers to integer ages (e.g., 20, 21, 22,...), the second to the quarter (with values 1, 2, 3 and 4) and the third to the number of people in the population/portfolio at the beginning of the year corresponding to each integer age and quarter.

`pop.end` A data frame, corresponding to the target population/portfolio, containing by age the stock of population/portfolio by quarter collected at the end of the year. This data frame must have three columns. The first refers to integer ages (e.g., 20, 21, 22,...), the second to the quarter (with values 1, 2, 3 and 4) and the third to the number of people in the population/portfolio at the end of the year corresponding to each integer age and quarter.

events.death
: A data frame with the number of deaths recorded in the population/portfolio during the target year in a set of integer ages for each combination of age and season/calendar quarter. Typically, this is an output of the function count_events_quarter or a data frame with the same structure. If the range of ages in ‘events.death‘ does not cover the range of ages in the intersection of ages defined by ‘pop.start‘ and ‘pop.end‘, zeros are imputed for the missing ages.

events.out
: A data frame with the number of exits (emigrants/lapses/expirations) recorded in the population/portfolio during the target year in a set of integer ages for each combination of age and season/calendar quarter. Typically, this is an output of the function count_events_quarter or a data frame with the same structure. If the range of ages in ‘events.out‘ does not cover the range of ages in the intersection of ages defined by ‘pop.start‘ and ‘pop.end‘, zeros are imputed for the missing ages.

events.in
: A data frame with the number of entries (immigrants/new policies) recorded in the population/portfolio during the target year in a set of integer ages for each combination of age and season/calendar quarter. Typically, this is an output of the function count_events_quarter or a data frame with the same structure. If the range of ages in ‘events.in‘ does not cover the range of ages in the intersection of ages defined by ‘pop.start‘ and ‘pop.end‘, zeros are imputed for the missing ages.

annual
: A character string informing whether the annual crude central rates of mortality should also be computed. Default, ‘FALSE‘.

Value

When ‘annual = FALSE‘ a data frame with estimated crude central rates of mortality for each combination of age and calendar quarter in the set of integer ages determined by the intersection of ages in ‘pop.start‘ and ‘pop.end‘. The data frame has the following components:

age
: Integer age to which the crude central rate of mortality corresponds.

quarter.age
: Age quarter to which the crude central rate of mortality corresponds.

quarter.calendar
: Calendar (time, season) quarter to which the crude central rate of mortality corresponds.

exposed
: Total exposure-at-risk times in the target population, calulated according to shortcut 2, for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

deaths
: Number of deaths in the target population for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

mx
: Estimated crude central rate of mortality corresponding to the combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

When ‘annual = TRUE‘ the output is a list with two data frames ‘mx.quarterly‘ and ‘mx.annual‘. ‘mx.quarterly‘ is a data frame with the estimated quarterly crude central rates of mortality as just described and ‘mx.annual‘ is the corresponding data frame with the estimated annual crude central rates of mortality.

Note

It is the responsibility of the user to assure that both stocks of population (which determine the ages for which estimates are computed) and number of events correspond to the same year.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2023). Shortcuts for the construction of sub-annual life tables. *ASTIN Bulletin: The Journal of the International Actuarial Association*, 53(2), 332-350. doi:10.1017/asb.2023.16

See Also

crude_mx, crude_mx_sh2.

Examples

```
# This can take a while
 pop <- 2005 - as.numeric(substr(pop_2006$date.birth, 1 , 4))
 pop <- as.data.frame(table(pop))
 pop[, 1] <- as.numeric(as.character(pop[, 1]))
 pop[, 2] <- pop[, 2]/4
 pop <- cbind(apply(pop, 2, rep, each = 4), quarter = rep(1:4, nrow(pop)))[, c(1, 3, 2)]
 pop <- as.data.frame(pop)
 temp <- quarterly_variables(death_2006$date.birth, death_2006$date.death)
 e.death <- count_events_quarter(temp)
 temp <- quarterly_variables(emi_2006$date.birth, emi_2006$date.emi)
 e.emi <- count_events_quarter(temp)
 temp <- quarterly_variables(immi_2006$date.birth, immi_2006$date.immi)
 e.immi <- count_events_quarter(temp)
 out <- crude_mx_sh3(pop, pop, e.death, e.emi, e.immi)


# Fast example
pop.1 <- data.frame(age = c(rep(40, 4), rep(41, 4)), quarter = rep(1:4, 2), people = c(4134, 4353))
pop.2 <- data.frame(age = c(rep(40, 4), rep(41, 4)), quarter = rep(1:4, 2), people = c(4250, 4213))
dates.b <- c("1980-04-12")
dates.d <- c("2020-08-23")
x <- quarterly_variables(dates.b, dates.d)
e.death <- count_events_quarter(x)
dates.b <- c("1980-05-12")
dates.e <- c("2020-06-23")
x <- quarterly_variables(dates.b, dates.e)
e.emi <- count_events_quarter(x)
dates.b <- c("1980-07-12")
dates.i <- c("2020-12-10")
x <- quarterly_variables(dates.b, dates.i)
```

```
e.immi <- count_events_quarter(x)
out <- crude_mx_sh3(pop.1, pop.2, e.death, e.emi, e.immi)
```

`death_2006` *Half of male deaths recorded in Comunitat Valenciana (Spain) during year 2006.*

Description

Data frame containing the dates of birth and death of a random sample composed by half of all men who died in the region of Comunitat Valenciana (Spain) during the year 2006.

Usage

```
data(death_2006)
```

Format

A data frame containing 10,146 rows and 2 columns:

date.birth Dates of birth in format "yyyy-mm-dd".

date.death Dates of death in format "yyyy-mm-dd".

Source

National Statistical Institute, https://www.ine.es/.

`distribute_excess` *Randomly distributes the excess of recorded births in a given day*

Description

Randomly distributes a number of births equivalent to the excess of registered births on a given day of a year among the different days of that year.

Usage

```
distribute_excess(
  date.birth,
  day = "01-01",
  maximum.excess = 50,
  date.event = NULL
)
```

Arguments

- `date.birth` A character vector with the dates of birth in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population.
- `day` A character vector in format "mm-dd" with the day of the year for which the (assumed) excess must be randomly distributed. By default, "01-01".
- `maximum.excess` A numeric value indicating the percentage of births registered above the average to be surpassed in the target day in order to consider that in that day an excess of births has been artificially recorded.
- `date.event` A character vector with the dates of events in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") linked to the population of births. By default, NULL. When the dates of births are linked to some dates of events and both occur in the same year, it can happen that some imputed dates of births be posterior to the dates of events. The inclusion of this argument (when different of NULL) avoids this possibility happening.

Value

A numeric vector of the same length and order as data.birth.

Note

We consider that in a day an excess of births has been registered if the percentage of the number of births recorded in that day surpasses the average number of births registered during the days of the corresponding year in a amount higher than ‘maximum.excess‘.

An excess usually happens in official statistics on the first of January. This occurs as a consequence of established protocols in country border systems, because when an immigrant does not know her/his day of birth, the border officials usually record them as January, 1. This provokes an artificial peak of dates of births in that date.

Author(s)

Josep Lledo <josep.lledo@uv.es>

Jose M. Pavia <pavia@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

`time_exposed_ins`, `time_exposed_outs`

Examples

```
dates <- c("1920-05-13", "1999-04-12", "2019-01-01", "2019-01-01",
           "2022-01-01", "2022-01-01", "2022-01-01", "2022-01-01", "2022-01-01")
distribute_excess(dates)
```

`emi_2006` *Male emigrants recorded in Comunitat Valenciana (Spain) during year 2006.*

Description

Data frame containing the dates of birth and emigration of a random sample composed by half of all men who emigrated from the region of Comunitat Valenciana (Spain) during the year 2006.

Usage

```
data(emi_2006)
```

Format

A data frame containing 18,006 rows and 2 columns:

date.birth Dates of birth in format "yyyy-mm-dd".

date.emi Dates of death in format "yyyy-mm-dd".

Source

National Statistical Institute, https://www.ine.es/.

`exact_age` *Time elapsed (in years) since the dates of birth and event.*

Description

Computes the time(s) elapsed (in years) between the date(s) of birth and the date(s) of event(s).

Usage

```
exact_age(
  date.birth,
  date.event,
  random.b = TRUE,
  random.e = TRUE,
  constant.age.year = FALSE
)
```

Arguments

`date.birth` A character vector with the dates of birth in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.b = FALSE‘, or a random hour by default.

`date.event` A character vector with the dates of events in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.e = FALSE‘, or a random hour, by default. This vector must have either length 1, when the aim is to compute the exact age of all the members of the population in the same temporal point or the same length as ‘date.birth‘ when the aim is to compute for each member of the population the exact age in the moment of the event (e.g., death).

`random.b` A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the birth occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.birth‘ even if those have been declared. By default, TRUE.

`random.e` A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the event occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.event‘ even if those have been declared. By default, TRUE.

`constant.age.year` A ‘TRUE/FALSE‘ argument indicating whether the length of the year should be constant, 365.25 days, or variable, depending on the time lived for the person in each year since her/his dates of birth and event. By default, FALSE. The advantage of using a non-constant (person-dependent) length of year is congruence when estimating time exposed at risk: in each year the time exposed along the time and age axes will coincide.

Value

A numeric vector of the same length as data.birth

Note

If ‘constant.age.year = FALSE‘ (default), the length of the year for each person is computed as a weighted average of the lengths of the years that the person has lived between the dates of birth and event using as weight the time lived for the person during each year.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

coord_age, coord_time

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
dates.e <- c("2002-03-23", "2009-04-12", "2019-01-01")
exact_age(dates.b, dates.e)
```

---

immi_2006 — *Half of the male immigrants recorded in Comunitat Valenciana (Spain) during year 2006.*

Description

Data frame containing the dates of birth and immigration of a random sample composed by half of all men who immigrate to the region of Comunitat Valenciana (Spain) during the year 2006.

Usage

```
data(immi_2006)
```

Format

A data frame containing 48,431 rows and 2 columns:

date.birth Dates of birth in format "yyyy-mm-dd".

date.immi Dates of death in format "yyyy-mm-dd".

Source

National Statistical Institute, https://www.ine.es/.

plot.qlifetable — *Graphical representation in a 4x4 raster of a qlifetable data frame.*

Description

Plot method for a data frame of events or time exposed occurring in each Lexis-diagram quarter for a set of ages. This is a plot method for the objects typically obtained using the function `count_events_quarter` or whatever of the time_exposed_ functions (e.g., `time_exposed_outs`).

Usage

```
## S3 method for class 'qlifetable'
plot(
  x,
  ...,
  range.ages = NULL,
  key = "numbers",
  decimal.digits = 2,
  color.palette = "grey",
  alpha.max = 1,
  alpha.min = 0.4,
  color.values = "black",
  big.mark = NULL,
  size.values = 3,
  legend.name = NULL,
  name.labels.age = c("Q1", "Q2", "Q3", "Q4"),
  name.labels.season = c("Winter", "Spring", "Summer", "Autumn"),
  show.plot = TRUE
)
```

Arguments

`x`	A data frame of quarterly summary statistics. Typically an output of the function `count_events_quarter` or whatever of the time_exposed_ functions (e.g., `time_exposed_outs`).
`...`	Other arguments passed on to methods. Not currently used.
`range.ages`	A vector of integers informing the aggregation of ages for which the graphical representation should be plotted. Default, NULL, the agggregation of all ages is shown.
`key`	Type of statistic to be presented in the plot. Either "numbers" or relative "percentages". Default, "numbers".
`decimal.digits`	Integer indicating the number of decimal places to be shown. Default, 2.
`color.palette`	Background base color for cells. Default, "grey".
`alpha.max`	A number in the interval [0, 1]. Maximum level of transparency to be applied for the background to build the palette. Default, 1.

`alpha.min`	A number in the interval [0, 1]. Minimum level of transparency to be applied for the background to build the palette. Default, 0.4.
`color.values`	Base color for numbers printed in each cell. Default, "black".
`big.mark`	A character string indicating the symbol to be used as thousand separator. Default, NULL.
`size.values`	A number indicating the font size to be used for inner-cells values. Default, 3.
`legend.name`	Name to be use as name in the legend. Default, NULL.
`name.labels.age`	Names to be used for the (y) age axis. Default, c("Q1", "Q2", "Q3", "Q4").
`name.labels.season`	Names to be used for the (x) season axis. Default, c("Winter", "Spring", "Summer", "Autumn").
`show.plot`	A TRUE/FALSE indicating if the plot should be displayed as a side-effect. By default, TRUE.

Value

Invisibly returns the (ggplot) description of the plot, which is a list with components that contain the plot itself, the data, information about the scales, panels, etc.

Note

ggplot2 is needed to be installed for this function to work.

Author(s)

Josep Lledo <josep.lledo@uv.es>

Jose M. Pavia, <pavia@uv.es>

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
dates.e <- c("2002-03-23", "2009-04-12", "2019-01-01")
x <- quarterly_variables(dates.b, dates.e)
out <- time_exposed_outs(x)
p <- plot(out, show.plot = FALSE)
```

`plot.SAI` *Graphical representation of a SAI object.*

Description

Plot method for a SAI object. This is a plot method for objects obtained using the `compute_SAI` function or the `SAI_shortcut_1` function.

Usage

```
## S3 method for class 'SAI'
plot(
  x,
  ...,
  min.age = 1,
  max.age = 100,
  decimal.digits = 1,
  color.main = "black",
  color.lm = "blue",
  color.hline = "red",
  size.labels.y = 8,
  axis.age.ticks,
  limits.joint,
  breaks.joint,
  limits.ageing,
  breaks.ageing,
  limits.seasonal,
  breaks.seasonal,
  name.labels.ageing = c("Q1", "Q2", "Q3", "Q4"),
  name.labels.season = c("Winter", "Spring", "Summer", "Autumn"),
  title.season = "Seasonal effects",
  title.ageing = "Ageing effects",
  x.title = "Age",
  show.plot = TRUE
)
```

Arguments

`x`	An object output of the `compute_SAI` or `SAI_shortcut_1` functions.
`...`	Other arguments passed on to methods. Not currently used.
`min.age`	A positive integer informing about the initial minimal age of the range of ages to be represented. This minimum age can be increased depending on the ages available in the SAI estimates. Default, 1.
`max.age`	A positive integer informing about the initial maximum age of the range of ages to be represented. This maximum age can be decreased depending on the ages available in the SAI estimates. Default, 100.
`decimal.digits`	Integer indicating the number of decimal places to be shown. Default, 1.
`color.main`	Color of the line corresponding to the estimated, normalized SAI. Default, "black".
`color.lm`	Color of the lm line. Default, "blue".
`color.hline`	Color of the hline (reference) line. Default, "red".
`size.labels.y`	Number informing the size of the labels of SAI axis. Default, 8.
`axis.age.ticks`	Optional vector with the values to be presented in the age axis.
`limits.joint`	Optional vector of length 2 with the limits for the SAI axis of the main plot.
`breaks.joint`	Optional vector with the values to be presented in the SAI axis of the main plot.

`limits.ageing` Optional vector of length 2 with the limits for the SAI axis of the marginal ageing SAI plot.

`breaks.ageing` Optional vector with the values to be presented in the SAI axis of the marginal ageing SAI plot.

`limits.seasonal` Optional vector of length 2 with the limits for the SAI axis of the marginal seasonal (calendar) SAI plot.

`breaks.seasonal` Optional vector with the values to be presented in the SAI axis of the marginal seasonal (calendar) SAI plot.

`name.labels.ageing` Names to be used for the (y) ageing axis. Default, c("Q1", "Q2", "Q3", "Q4").

`name.labels.season` Names to be used for the (x) season (calendar) axis. Default, c("Winter", "Spring", "Summer", "Autumn").

`title.season` Name to be used for identifying the seasonal marginal panels. Default, "Seasonal effects".

`title.ageing` Name to be used for identifying the ageing marginal panels. Default, "Ageing effects".

`x.title` Name to be used for the x axis. Default, "Age".

`show.plot` A TRUE/FALSE value indicating whether the plot should be displayed as a side-effect. By default, TRUE.

Value

Invisibly returns the grob description of the plot, which is a list with components that contain the plot itself, the data, information about the scales, panels, etc.

Note

ggplot2 and gridExtra packages are needed to be installed for this function to work.

Author(s)

Jose M. Pavia, <pavia@uv.es>

Examples

```
# This can take a while
 t.stock <- time_exposed_stock(pop_2006$date.birth, 2006, "forward")
 t.stock <- t.stock[t.stock$age <= 100, ]
 temp <- quarterly_variables(death_2006$date.birth, death_2006$date.death)
 e.death <- count_events_quarter(temp)
 e.death <- e.death[e.death$age <= 100, ]
 t.birth <- time_exposed_newborns(birth_2006$date.birth)
 out <- crude_mx(t.stock, e.death, time.birth = t.birth)
 SAI.example <- compute_SAI(out, out)
 p <- plot(SAI.example, show.plot = FALSE)
```

pop_2006 — *Half of the male population living in the Comunitat Valenciana (Spain) on January, 1 2006.*

Description

Data frame containing the dates of birth of a random sample composed by half of all men who were alive on 1 January 2006 in the region of Comunitat Valenciana (Spain).

Usage

```
data(pop_2006)
```

Format

A data frame containing 1,197,154 rows and 1 column:

date.birth Dates of birth in format "yyyy-mm-dd".

Source

Instituto Valenciano de Estadistica (IVE) https://pegv.gva.es/es/.

quarterly_variables — *Data frame of quarterly variables*

Description

Computes punctual risk coordinates in the Lexis diagram and quarterly biometric variables of a population.

Usage

```
quarterly_variables(
  date.birth,
  date.event,
  random.b = TRUE,
  random.e = TRUE,
  constant.age.year = FALSE
)
```

Arguments

- `date.birth` A character vector with the dates of birth in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.b = FALSE‘, or a random hour by default.
- `date.event` A character vector with the dates of events in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of a population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.e = FALSE‘, or a random hour, by default. This vector must have either length 1, when the aim is to compute the exact age or the (1x1-Lexis) age coordinate of all the members of the population in the same temporal point or the same length as ‘date.birth‘ when the aim is to compute for each member of the population the exact age or the (1x1-Lexis) age coordinate in the moment of the event (e.g., death).
- `random.b` A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the birth occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.birth‘ even if those have been declared. By default, TRUE.
- `random.e` A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the event occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.event‘ even if those have been declared. By default, TRUE.
- `constant.age.year` A ‘TRUE/FALSE‘ argument indicating whether the length of the year should be constant, 365.25 days, or variable, depending on the time lived for the person in each year since her/his dates of birth and event. By default, FALSE. The advantage of using a non-constant (person-dependent) length of year is congruence when estimating time exposed at risk: in each year the time exposed along the time and age axes will coincide.

Value

A data.frame with the following components:

- `coord.age` Time elapsed, measure in years, between the last birthday and the date when the event happens.
- `coord.time` Time coordinate: time elapsed, measure in years, between the begining of the year and the date when the event happens.
- `age.last.birthday` The integer age at last birthday.
- `exact.age.at.event` Time elapsed, measure in years, between the dates of birth and event.
- `quarter.age` Age quarter when the event happens.
- `quarter.calendar` Calendar (time, season) quarter to which the time exposed at risk corresponds.
- `year` Year when the event happens.

Note

In the age axis, the length of the years are assumed either constant 365.25 days (‘constant.age.year = TRUE‘) or variable (‘constant.age.year = FALSE‘), depending on the person. In the time axis, the length of the year is either 365 in non-leap years and 366 in leap years. The advantage of using a non-constant (person-dependent) length of year in the age axis is that in each year the lengths of the years when computing ‘coord.age‘ and ‘coord.time‘ in both axis are equal.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
dates.e <- c("2002-03-23", "2009-04-12", "2019-01-01")
quarterly_variables(dates.b, dates.e)
```

SAI_shortcut_1 *Estimates seasonal-ageing indexes using shortcut 1.*

Description

Given a set of data frames with the number of deaths recorded in a population/portfolio during several years (periods) in a set of integer ages for each combination of age and season/calendar quarter, this function approximates their corresponding seasonal-ageing indexes by employing shortcut 1, based on equation (2.5), proposed in Pavia and Lledo (2023).

Usage

```
SAI_shortcut_1(x, y, ..., margins = FALSE, min.age = 0, max.age = 100)
```

Arguments

x	A data frame with the number of deaths recorded in the population/portfolio during a period (typically a year) in a set of integer ages for each combination of age and season/calendar quarter. Usually, this is an output of the `count_events_quarter` function (or a data frame with the same structure).

`y` A data frame with the number of deaths recorded in the population/portfolio during another period (typically a different year than the one corresponding to ‘x‘) in a set of integer ages for each combination of age and season/calendar quarter. Usually, this is an output of the `count_events_quarter` function (or a data frame with the same structure).

`...` Further data frames similar to ‘x‘ and ‘y‘ corresponding to another period(s)/year(s).

`margins` A ‘TRUE/FALSE‘ argument informing whether or not the marginal seasonal and ageing indexes should be also computed. Default, ‘FALSE‘.

`min.age` A non-negative integer informing about the initial minimal age for which SAIs must be computed. This minimum age can be increased depending on the ages for which there are crude mx estimates (exposed-at-risk) in the objects ‘x‘, ‘y‘, ‘...‘. Default, 0.

`max.age` A positive integer informing about the initial maximum age for which SAIs must be computed. This maximum age can be decreased depending on the maximum ages for which there are crude mx estimates (exposed-at-risk) in the objects ‘x‘, ‘y‘, ‘...‘. Default, 100.

Value

When ‘margins = FALSE‘ a data frame with the (raw, normalized and linearized) estimated seasonal-ageing indexes (SAI) corresponding to the set of integer ages determined by ‘min.age‘ and ‘max.age‘ and the ages for which there is a least a death in the data frames introduced via the ‘x‘, ‘y‘ and ‘...‘ arguments for each combination of age and calendar quarter. The data frame has the following components:

`age` Integer age to which the SAIs corresponds.

`quarter.age` Age quarter to which the SAIs corresponds.

`quarter.calendar` Calendar (time, season) quarter to which the SAIs corresponds.

`SAI.raw` Estimates of raw seasonal-ageing indexes for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

`SAI.norm` Estimates of seasonal-ageing indexes, attained after normalizing raw SAIs estimates, for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

`SAI.lin` Final estimates of seasonal-ageing indexes, attained after linearizing normalized SAIs estimates, for each combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

When ‘margins = TRUE‘ the output is a list with three data frames ‘SAI‘, ‘SAI.age‘ and ‘SAI.quarter‘. ‘SAI‘ is defined as just described above. ‘SAI.age‘ and ‘SAI.quarter‘ contains, in a similar vein than ‘SAI‘, the estimated marginal SAIs, corresponding to, respectively, the age quarter and the calendar (time, season) quarter.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2023). Shortcuts for the construction of sub-annual life tables. *ASTIN Bulletin: The Journal of the International Actuarial Association*, 53(2), 332-350. doi:10.1017/asb.2023.16

Examples

```
# This can take a while
 temp <- quarterly_variables(death_2006$date.birth, death_2006$date.death)
 e.death <- count_events_quarter(temp)
 SAI.example <- SAI_shortcut_1(e.death, e.death)


# Fast example
dates.b <- c("2017-05-13", "2018-04-12", "2018-01-01")
dates.d <- c("2020-09-23", "2021-10-11", "2021-11-23")
x <- quarterly_variables(dates.b, dates.d)
e.death <- count_events_quarter(x)
SAI.example <- SAI_shortcut_1(e.death, e.death)
```

---

time_exposed_ins   *Data frame of time exposed at risk for a population of immigrants/portfolio entries*

Description

Computes for each integer age and each combination of age and seasonal quarter the total time exposed at risk (in years) of a population of immigrants/new policies (new production) during the year of the event. The computation is performed using the associated data frame of quarterly variables corresponding to the population obtained using the `quarterly_variables` function.

Usage

```
time_exposed_ins(x)
```

Arguments

`x`	A data.frame output of the `quarterly_variables` function.

Value

A data frame with the time exposed at risk for each (potential) combination of integer age and age and season quarter of the input dataset. The data frame has the following components:

`age`	Integer age to which the time exposed at risk corresponds.
`quarter.age`	Age quarter to which the time exposed at risk corresponds.
`quarter.calendar`	Calendar (time, season) quarter to which the time exposed at risk corresponds.
`time.exposed`	Total time (in years) exposed at risk of the population during the quarter determined for the combination of 'age', 'quarter.age' and 'quarter.calendar'.

Note

The time exposed at risk is computed for each death from the beginning of the year in which the event occurred until the moment of occurrence of the event. Please see the note in the `time_exposed_stock` function.

Author(s)

Josep Lledo <josep.lledo@uv.es>

Jose M. Pavia <pavia@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

`time_exposed_stock`, `time_exposed_outs`, `time_exposed_newborns`

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
dates.e <- c("2002-03-23", "2009-04-12", "2019-01-01")
x <- quarterly_variables(dates.b, dates.e)
out <- time_exposed_ins(x)
```

`time_exposed_newborns` *Data frame of time exposed at risk for a population of newborns*

Description

Computes for each combination of age and seasonal quarter the total time exposed at risk (in years) of a population of newborns, during the year of their birth, this is up to the end of the year when they born.

Usage

```
time_exposed_newborns(date.birth, random.b = TRUE)
```

Arguments

`date.birth`	A character vector with the dates of birth in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of the members of the population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.b = FALSE‘, or a random hour by default.

`random.b` A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the birth occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.birth‘ even if those have been declared. By default, TRUE.

Value

A data frame with the time exposed at risk for each (potential) combination of integer age and age and season quarter of the population. The data frame has the following components:

`age` Integer age to which the time exposed at risk corresponds.

`quarter.age` Age quarter to which the time exposed at risk corresponds.

`quarter.calendar`
Calendar (time, season) quarter to which the time exposed at risk corresponds.

`time.exposed` Total time (in years) exposed at risk of the population during the quarter determined for the combination of ‘age‘, ‘quarter.age‘ and ‘quarter.season‘.

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

`time_exposed_stock`, `time_exposed_outs`, `time_exposed_ins`

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
out <- time_exposed_newborns(dates.b)
```

---

time_exposed_outs *Data frame of time exposed at risk for a population of deaths/emigrants/exits (portfolio withdrawals, lapses) during the year of the event.*

Description

Computes for each integer age and each combination of age and seasonal quarter the total time exposed at risk (in years) of a population of deceased/emigrants/exits (portfolio withdrawals, lapses) during the year of the event. The computation is performed using the associated data frame of quarterly variables corresponding to the population obtained using the `quarterly_variables` function.

Usage

```
time_exposed_outs(x)
```

Arguments

x	A data.frame output of the quarterly_variables function.

Value

A data frame with the time exposed at risk for each (potential) combination of integer age and age and season quarter of the input dataset. The data frame has the following components:

age	Integer age to which the time exposed at risk corresponds.
quarter.age	Age quarter to which the time exposed at risk corresponds.
quarter.calendar	Calendar (time, season) quarter to which the time exposed at risk corresponds.
time.exposed	Total time (in years) exposed at risk of the population during the quarter determined for the combination of ‘age‘, ‘quarter.age‘ and ‘quarter.calendar‘.

Note

The time exposed at risk is computed for each person from the beginning of the year in which the event occurred until the moment of occurrence of the event. Please see the note in the time_exposed_stock function.

Author(s)

Josep Lledo <josep.lledo@uv.es>

Jose M. Pavia <pavia@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

time_exposed_stock, time_exposed_newborns, time_exposed_ins

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
dates.e <- c("2002-03-23", "2009-04-12", "2019-01-01")
x <- quarterly_variables(dates.b, dates.e)
out <- time_exposed_outs(x)
```

time_exposed_stock *Data frame of time exposed at risk for a stock of general/insured population*

Description

Computes for each integer age and each combination of age and seasonal quarter the total time exposed at risk (in years) of a (stock) population of survivors (expected survivors) during a given year.

Usage

```
time_exposed_stock(
  date.birth,
  year,
  type,
  random.b = TRUE,
  constant.age.year = FALSE
)
```

Arguments

date.birth
: A character vector with the dates of birth in format either "yyyy-mm-dd" or "yyyy-mm-dd hour:min:secs" (for instance, "2016-01-20 12:00:00") of the members of the population. If "hour:min:secs" is omitted the function imputes either "12:00:00", if ‘random.b = FALSE‘, or a random hour by default.

year
: A numeric vector indicating the year for which the total time exposed at risk (by quarter) of the population is to be computed.

type
: A character argument informing if the total time exposed to risk is computed either since the beginning of the year or from the end of the year, depending when the census (stock) of population (portfolio) has been made. Only two values are allowed: ‘"forward"‘ and ‘"backward"‘. If ‘type = "forward"‘ the time exposed to risk is computed since the beginning of the year (i.e., it is assumed that the population counting has been performed at the beginning of the year of interest). If ‘type = "backward"‘ the time exposed to risk is computed from the end of the year (i.e., it is assumed that the population counting has been performed at the end of the year of interest).

random.b
: A ‘TRUE/FALSE‘ argument indicating whether the exact moment ("hour:min:secs") when the birth occurs within the day is randomly selected. If TRUE, this overwrites "hour:min:secs" in ‘date.birth‘ even if those have been declared. By default, TRUE.

constant.age.year
: A ‘TRUE/FALSE‘ argument indicating whether the length of the year should be constant, 365.25 days, or variable, depending on the time lived for the person in each year since her/his dates of birth and event. By default, FALSE. The advantage of using a non-constant (person-dependent) length of year is congruence

when estimating time exposed at risk: in each year the time exposed along the time and age axes will coincide.

Value

A data frame with the time exposed at risk for each (potential) combination of integer age and age and season/calendar quarter of the population. The data frame has the following components:

- `age` Integer age to which the time exposed at risk corresponds.
- `quarter.age` Age quarter to which the time exposed at risk corresponds.
- `quarter.calendar` Calendar (time, season) quarter to which the time exposed at risk corresponds.
- `time.exposed` Total time (in years) exposed at risk of the population during the quarter determined for the combination of 'age', 'quarter.age' and 'quarter.calendar'.

Note

Using the notation of a general population, denoting by P the stock of population counted either at the beginning or the end of the year, and by E, I, D and B the emigrants, immigrants, deaths and births recorded during the year, to compute the total time exposed to risk they relate as follows:

If the census (stock) of the population is performed at the beginning of the year of interest, it is initially assumed that all the people is going to survive (is going to be at risk) up to the end of the year. In this case 'type = "forward"' should be used and the total time exposed at risk, in each age a and (r, s) quarter, is through: T(a, r, s) = time_exposed_stock(P) + time_exposed_ins(I) - time_exposed_ins(E) - time_exposed_ins(D) + time_exposed_newborns(B).

If the census (stock) of population is performed at the end of the year of interest, only the people who survives up to that date is included in the counting. In this case 'type = "backward"' should be used and the total time exposed at risk, in each age a and (r, s) quarter, is computed using the expression: T(a, r, s) = time_exposed_stock(P) - time_exposed_outs(I) + time_exposed_outs(E) + time_exposed_outs(D).

Author(s)

Jose M. Pavia <pavia@uv.es>

Josep Lledo <josep.lledo@uv.es>

References

Pavia, JM and Lledo, J (2022). Estimation of the Combined Effects of Ageing and Seasonality on Mortality Risk. An application to Spain. *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, 185(2), 471-497. doi:10.1111/rssa.12769

See Also

`time_exposed_ins`, `time_exposed_outs`, `time_exposed_newborns`

Examples

```
dates.b <- c("1920-05-13", "1999-04-12", "2019-01-01")
out <- time_exposed_stock(dates.b, year = 2019, type = "backward")
```

Index