

Package: ltRISK (via r-universe)

October 14, 2026

Title Estimate Lifetime and Age-Conditional Risk of Developing or Dying from Cancer

Version 0.1.0

Description Estimates lifetime and age-conditional risks of developing or dying from cancer using population-based cancer registry data. Four established approaches for lifetime risk estimation are implemented: the DevCan method developed by the Surveillance, Epidemiology, and End Results (SEER) Program, the Wun method, the Adjusted for Multiple Primary Cancers (AMP) method, and a cumulative risk approach with a predefined upper age limit. The package provides a unified interface through the `calc_ltr()` function, which supports both gamma-based and delta-method confidence intervals, as well as the piecewise mid-age group joinpoint (PMAJ) model for age-specific rate modeling.

License MIT + file LICENSE

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

Depends R (>= 4.0.0)

Imports memoise, parallel, stats, utils

LazyData true

URL <https://github.com/gigu003/ltRISK>

BugReports <https://github.com/gigu003/ltRISK/issues>

Suggests dplyr, testthat (>= 3.0.0), knitr, rmarkdown

Config/testthat/edition 3

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

Repository <https://gigu003.r-universe.dev>

Date/Publication 2026-08-15 16:02:41 UTC

RemoteUrl <https://github.com/gigu003/ltrRISK>
RemoteRef HEAD
RemoteSha 765db198c28a6dfa1ac56f9922e7af41111aedf6

Contents

amp	2
calc_ltr	3
calc_ltr_df	6
clear_ltr_cache	8
cumulative	9
devcan	10
format_risk_ci	11
GCO_Today_2022	13
get_risk	14
pairwise_ztest	15
pmaj	16
seer_example_data	17
trend_test	18
wun	19
ztest	21
Index	23

amp	<i>Compute probabilities used for estimation of lifetime risk</i>
-----	---

Description

Estimates the lifetime and age-conditional probabilities of developing cancer, adjusted for multiple primary cancers, using a competing risks framework based on the method described in Sasieni et al. (2011).

Usage

```
amp(  
  ages,  
  cancer,  
  cancer_death,  
  death,  
  pys,  
  last_age_widths = Inf,  
  type = "developing",  
  maj_method = "pmaj",  
  pmaj_sub_interval = 0.5  
)
```

Arguments

ages	Starting ages of each age interval (e.g., 0, 5, 10, ..., 85).
cancer	Number of cancer diagnoses in each age interval. For AMP, this may include multiple primary cancers.
cancer_death	Number of deaths due to cancer in each age interval.
death	Number of all deaths (all causes combined) in each age interval.
pys	Person-years at risk corresponding to each age interval.
last_age_widths	Width of the last age group (default = Inf).
type	Characters "developing" or "dying" indicate estimate the probability of developing cancer or dying from it.
maj_method	Character string. The method to use for smoothing the rates: "pmaj" for Piece-wise Mid-Age Group Joinpoint, "constant" for simple piece-wise constant. Default is "pmaj". Note: To approximate the exact MAJ (Mid-Age Group Joinpoint) method, use "pmaj" with a small value for pmaj_sub_interval (e.g., 0.01), as MAJ requires numerical integration for exact computation but can be closely approximated this way.
pmaj_sub_interval	Numeric. Sub-interval size for pmaj (default = 0.5 years).

Value

Contribution of risk of developing cancer or dying from it in each age group.

References

Sasieni PD, Shelton J, Ormiston-Smith N, Thomson CS, Silcocks PB. What is the lifetime risk of developing cancer?: the effect of adjusting for multiple primaries. *Br J Cancer*, 2011;105:460–465. DOI: 10.1038/bjc.2011.250

calc_ltr

Estimate disease risk and its confidence interval

Description

Estimate disease risk using population-based cancer registry data, including age-specific cancer incidence, cancer mortality, all-cause mortality, and corresponding population data. Available methods include adjusting for multiple primary cancers (AMP), the DevCan method recommended by Surveillance, Epidemiology, and End Results (SEER), and the Wun method.

Usage

```
calc_ltr(
  ages,
  cancer,
  cancer_death,
  death,
  pys,
  age_start = min(ages),
  age_end = Inf,
  age_combine = c("cross", "pairwise"),
  alpha = 0.05,
  risk_func = "devcan",
  ci_method = "gamma",
  variance_method = "auto",
  multiplier = 100,
  digits = 6,
  return_variance = FALSE,
  ...
)
```

Arguments

ages	Starting ages of each age interval (e.g., 0, 5, 10, ..., 85).
cancer	Number of cancer diagnoses in each age interval. For AMP, this may include multiple primary cancers.
cancer_death	Number of deaths due to cancer in each age interval.
death	Number of all deaths (all causes combined) in each age interval.
pys	Person-years at risk corresponding to each age interval.
age_start	Starting age(s) used to calculate risk over one or multiple age ranges.
age_end	Ending age(s) used to calculate risk over one or multiple age ranges.
age_combine	How age_start and age_end are combined. "cross" (default) returns every valid combination. "pairwise" pairs elements positionally, allowing scalar recycling.
alpha	Alpha level specifying the confidence level for estimating the risk confidence interval.
risk_func	Function used to estimate cancer risk. Options are "devcan", "amp", "wun" or "cumulative", with "devcan" as the default. The legacy alias "cumu" is also accepted.
ci_method	Method used to estimate confidence intervals for the risk. Options are "gamma", "delta", or "none", with "gamma" as the default. Use "none" to skip variance and confidence-interval calculations.
variance_method	Method used to estimate the risk variance. "auto" uses an analytic gradient for DevCan, AMP, Wun, and cumulative-risk models and otherwise falls back to "finite_difference". Analytic gradients support constant and PMAJ rates. Wun

	models with surgery correction use finite differences. "analytic" requires an implemented analytic gradient. Default is "auto".
multiplier	Numeric value to scale the risk and confidence interval. Default is 100, which reports risk as a percentage.
digits	Integer indicating the number of decimal places to round the results. Default is 6.
return_variance	Logical. If TRUE, also return the estimated variance and standard error of the risk on the same scale as multiplier. These values are not rounded by digits, so they can be used for downstream statistical tests. Default is FALSE.
...	Additional arguments passed to the risk estimation function, depending on risk_func; see the corresponding model function for available parameters. For age-specific rates, DevCan, AMP, and cumulative risk default to maj_method = "pmaj", while Wun defaults to maj_method = "constant" for backward compatibility. The default pmaj_sub_interval is 0.5 years. Input counts must be non-negative, pys must be positive, and cancer deaths must not exceed all-cause deaths. Variances treat cancer incidence, cancer deaths, and other deaths as independent Poisson count components and treat person-years as fixed. Gamma intervals return zero limits for an age range with zero estimated risk and no positive finite local perturbation.

Value

A data frame containing the following variables:

- start — starting age of the age range for the risk.
- end — ending age of the age range for the risk.
- risk — estimated risk value.
- lower — lower bound of the confidence interval.
- upper — upper bound of the confidence interval.
- variance — estimated variance of the risk, returned only when return_variance = TRUE.
- se — estimated standard error of the risk, returned only when return_variance = TRUE.

References

- Sasieni PD, Shelton J, Ormiston-Smith N, Thomson CS, Silcocks PB. What is the lifetime risk of developing cancer?: the effect of adjusting for multiple primaries. *Br J Cancer*, 2011;105:460–465. DOI: 10.1038/bjc.2011.250
- Fay MP, Pfeiffer R, Cronin KA, Le C, Feuer EJ. (2003). *Age-conditional probabilities of developing cancer. Statistics in Medicine*, 22(11):1837-1848. DOI: 10.1002/sim.1428.
- Fay M P. *Estimating age conditional probability of developing disease from surveillance data[J]*. Population Health Metrics, 2004, 2(1): 6.

Examples

```
breast <- seer_example_data[seer_example_data$site=="Breast",]
calc_ltr(ages = breast$ages, cancer = breast$cancer,
         cancer_death = breast$cancer_death, death = breast$death,
         pys = breast$pys, maj_method = "constant",
         age_start = c(0, 30, 50, 70), age_end = c(30, 50, 70, Inf),
         digits = 4)
```

calc_ltr_df

Estimate disease risk from a data frame

Description

calc_ltr_df() is a data-frame interface to [calc_ltr\(\)](#). It lets users provide one data frame and specify the columns containing age-specific cancer incidence, cancer mortality, all-cause mortality, and person-years.

Usage

```
calc_ltr_df(
  data,
  ages = "ages",
  cancer = "cancer",
  cancer_death = "cancer_death",
  death = "death",
  pys = "pys",
  by = NULL,
  parallel = FALSE,
  workers = NULL,
  cluster = NULL,
  cache = c("retain", "clear", "none"),
  ...
)
```

Arguments

data	A data frame containing the input variables.
ages, cancer, cancer_death, death, pys	Column names in data. Each can be supplied as a character string, a bare column name, or a character variable.
by	Optional character vector of grouping columns. When supplied, calc_ltr_df() calculates risks separately within each group and returns the grouping columns together with the risk estimates. If data is a <code>dplyr::grouped_df</code> and by = NULL, grouping variables are detected automatically.

parallel	Logical. If TRUE, calculate independent groups in parallel using a cross-platform PSOCK cluster. Parallel execution is used only when there is more than one group and more than one worker. Default is FALSE.
workers	Number of parallel worker processes. NULL chooses up to one fewer than the detected logical cores, capped by the number of groups. workers = 1 always uses the serial path. Ignored when parallel = FALSE. Because each PSOCK worker is a separate R process, parallel execution is most useful for many computationally intensive groups (for example, AMP or DevCan gamma intervals) and uses additional memory.
cluster	Optional PSOCK cluster created by <code>parallel::makePSOCKcluster()</code> . When supplied, it is reused and is not stopped by <code>calc_ltr_df()</code> . This is useful for several consecutive grouped calculations. workers is ignored when cluster is supplied.
cache	Cache policy. "retain" (default) retains memoised results for later calls. "clear" clears caches after the complete calculation. "none" additionally clears caches after every group, limiting peak cache growth in long grouped jobs. With an external cluster, clearing also occurs on workers while leaving the cluster running.
...	Additional arguments passed to <code>calc_ltr()</code> , such as risk_func, type, maj_method, ci_method, age_start, and age_end.

Value

A data frame returned by `calc_ltr()`.

Examples

```
# One population using the package's standard column names
breast <- seer_example_data[seer_example_data$site == "Breast", ]
calc_ltr_df(
  breast,
  maj_method = "constant",
  ci_method = "none",
  age_start = c(0, 30, 50, 70),
  age_end = c(30, 50, 70, Inf),
  age_combine = "pairwise",
  digits = 4
)

# Calculate independent male and female risks in one call
all_by_sex <- seer_example_data[
  seer_example_data$site == "All" & seer_example_data$sex %in% c(1, 2),
]
calc_ltr_df(
  all_by_sex,
  by = "sex",
  maj_method = "constant",
  ci_method = "delta",
  age_start = 40,
  age_end = Inf,
```

```

    return_variance = TRUE
  )

  # Input columns can have other names
  custom <- data.frame(
    age_group = breast$ages,
    cases = breast$cancer,
    cancer_deaths = breast$cancer_death,
    all_deaths = breast$death,
    population = breast$pys
  )
  calc_ltr_df(
    custom,
    ages = age_group,
    cancer = cases,
    cancer_death = "cancer_deaths",
    death = all_deaths,
    pys = population,
    ci_method = "none",
    maj_method = "constant"
  )

  ## Not run:
  # Parallel grouped calculation and reusable cluster
  calc_ltr_df(
    seer_example_data,
    by = c("site", "sex"),
    ci_method = "delta",
    parallel = TRUE,
    workers = 2
  )

  cl <- parallel::makePSOCKcluster(2)
  calc_ltr_df(
    seer_example_data,
    by = c("site", "sex"),
    ci_method = "none",
    cluster = cl
  )
  parallel::stopCluster(cl)

  ## End(Not run)

```

clear_ltr_cache

Manage litRISK in-memory caches

Description

clear_ltr_cache() removes memoised risk-model results and PMAJ grid geometries from the current R process. It can also clear the corresponding caches on an existing PSOCK cluster. ltr_cache_info() reports the number of cached entries in the current process.

Usage

```
clear_ltr_cache(cluster = NULL)

ltr_cache_info()
```

Arguments

cluster Optional cluster created by `parallel::makePSOCKcluster()`. When supplied, caches are also cleared on every worker. The cluster remains running.

Value

`clear_ltr_cache()` invisibly returns `NULL`. `ltr_cache_info()` returns a data frame with cache names and entry counts.

cumulative	<i>Calculate cumulative rate and risk.</i>
------------	--

Description

This function computes age-specific rates and cumulative risk contributions across age intervals based on cancer incidence or cancer mortality counts. It converts interval counts into rates using person-years, and then calculates the contribution of each age group to the cumulative risk using $1 - \exp(-r_x \cdot n_x)$, where r_x is the age-specific rate and n_x is the width of the age interval.

Usage

```
cumulative(
  ages,
  cancer,
  cancer_death,
  death,
  pys,
  last_age_widths = Inf,
  type = "developing",
  maj_method = "pmaj",
  pmaj_sub_interval = 0.5
)
```

Arguments

ages	Starting ages of each age interval (e.g., 0, 5, 10, ..., 85).
cancer	Number of cancer diagnoses in each age interval. For AMP, this may include multiple primary cancers.
cancer_death	Number of deaths due to cancer in each age interval.
death	Number of all deaths (all causes combined) in each age interval.

pys	Person-years at risk corresponding to each age interval.
last_age_widths	Width of the last age group (default = Inf).
type	Characters "developing" or "dying" indicate estimate the probability of developing cancer or dying from it.
maj_method	Character string. The method to use for smoothing the rates: "pmaj" for Piece-wise Mid-Age Group Joinpoint, "constant" for simple piece-wise constant. Default is "pmaj". Note: To approximate the exact MAJ (Mid-Age Group Joinpoint) method, use "pmaj" with a small value for pmaj_sub_interval (e.g., 0.01), as MAJ requires numerical integration for exact computation but can be closely approximated this way.
pmaj_sub_interval	Numeric. Sub-interval size for pmaj (default = 0.5 years).

Value

A list of class "cumu" containing:

ages Vector of starting ages.

widths Vector of widths for each age interval.

rate Age-specific rates, computed as count / pys.

See Also

[get_risk.cumu](#) for obtaining cumulative risks across arbitrary age ranges.

devcan

Compute probabilities used for estimation of lifetime risk

Description

Compute probabilities used for estimation of lifetime risk

Usage

```
devcan(
  ages,
  cancer,
  death,
  cancer_death,
  pys,
  type = "developing",
  no_other_death = FALSE,
  maj_method = "pmaj",
  pmaj_sub_interval = 0.5
)
```

Arguments

ages	Starting ages of each age interval (e.g., 0, 5, 10, ..., 85).
cancer	Number of cancer diagnoses in each age interval. For AMP, this may include multiple primary cancers.
death	Number of all deaths (all causes combined) in each age interval.
cancer_death	Number of deaths due to cancer in each age interval.
pys	Person-years at risk corresponding to each age interval.
type	Characters "developing" or "dying" indicate estimate the probability of developing cancer or dying from it.
no_other_death	Logical. If TRUE, assumes absence of other causes of death (i.e., non-cancer mortality = 0). Default = FALSE.
maj_method	Character string. The method to use for smoothing the rates: "pmaj" for Piece-wise Mid-Age Group Joinpoint, "constant" for simple piece-wise constant. Default is "pmaj". Note: To approximate the exact MAJ (Mid-Age Group Joinpoint) method, use "pmaj" with a small value for pmaj_sub_interval (e.g., 0.01), as MAJ requires numerical integration for exact computation but can be closely approximated this way.
pmaj_sub_interval	Numeric. Sub-interval size for pmaj (default = 0.5 years).

Value

Contribution of risk of developing cancer or dying from it in each age group.

References

Fay MP, Pfeiffer R, Cronin KA, Le C, Feuer EJ. (2003). *Age-conditional probabilities of developing cancer. Statistics in Medicine*, 22(11):1837-1848. DOI: 10.1002/sim.1428.

Fay M P. *Estimating age conditional probability of developing disease from surveillance data[J]*. Population Health Metrics, 2004, 2(1): 6.

format_risk_ci

Format risk estimates with 95% confidence intervals

Description

format_risk_ci() combines risk estimates and their lower and upper confidence interval bounds into a character vector suitable for tables. When risk is a data frame, the formatted value is added as a new column.

Usage

```
format_risk_ci(
  risk,
  lower = NULL,
  upper = NULL,
  risk_col = "risk",
  lower_col = "lower",
  upper_col = "upper",
  name = "risk_95ci",
  digits = NULL,
  conf_level = 95,
  show_conf_level = FALSE,
  sep = ", ",
  na = NA_character_
)
```

Arguments

<code>risk</code>	Numeric vector of risk estimates, or a data frame containing risk estimates and confidence interval bounds.
<code>lower</code>	Numeric vector of lower confidence interval bounds. When <code>risk</code> is a data frame, this can be the lower-bound column name.
<code>upper</code>	Numeric vector of upper confidence interval bounds. When <code>risk</code> is a data frame, this can be the upper-bound column name.
<code>risk_col</code>	Column name containing risk estimates when <code>risk</code> is a data frame. Default is "risk".
<code>lower_col</code>	Column name containing lower confidence interval bounds when <code>risk</code> is a data frame. Default is "lower".
<code>upper_col</code>	Column name containing upper confidence interval bounds when <code>risk</code> is a data frame. Default is "upper".
<code>name</code>	Column name for the formatted output when <code>risk</code> is a data frame. Default is "risk_95ci".
<code>digits</code>	Optional integer indicating the number of decimal places to display. If <code>NULL</code> and <code>risk</code> is a data frame returned by <code>calc_ltr()</code> or <code>calc_ltr_df()</code> , the calculation's <code>digits</code> setting is inherited. Otherwise, values are converted with <code>as.character()</code> without additional rounding.
<code>conf_level</code>	Confidence level displayed in the formatted text. Default is 95.
<code>show_conf_level</code>	Logical. If <code>TRUE</code> , include the confidence level label such as 95% CI before the interval bounds. Default is <code>FALSE</code> .
<code>sep</code>	Separator between lower and upper confidence interval bounds. Default is ", ".
<code>na</code>	Character value used when any of <code>risk</code> , <code>lower</code> , or <code>upper</code> is missing for an observation. Default is <code>NA_character_</code> .

Value

A character vector in the form risk (lower, upper), or a data frame with an added formatted column when risk is a data frame.

Examples

```
format_risk_ci(risk = 12.3, lower = 10.1, upper = 14.5)
format_risk_ci(risk = 12.345, lower = 10.123, upper = 14.567, digits = 1)

breast <- seer_example_data[seer_example_data$site == "Breast", ]
res <- calc_ltr(
  ages = breast$ages, cancer = breast$cancer,
  cancer_death = breast$cancer_death, death = breast$death,
  pys = breast$pys, maj_method = "constant",
  ci_method = "delta", age_start = 0, age_end = Inf,
  digits = 2
)
res$risk_95ci <- format_risk_ci(res$risk, res$lower, res$upper, digits = 2)
format_risk_ci(res, digits = 2)
```

GCO_Today_2022

*GCO_Today_2022***Description**

A data frame containing the number of cancer cases, cancer deaths (cancer mortality), all-cause deaths, and population counts for the world total, stratified by age group and cancer site with both sexes combined. The cancer cases and cancer deaths are obtained from the Global Cancer Observatory Today, while the population counts and all-cause deaths are sourced from the World Population Prospects 2022.

Usage

```
GCO_Today_2022
```

Format

A data frame with 702 rows and 8 variables:

site_code integer: Cancers include the code of cancer sites.
icd10 character: ICD-10 codes corresponding to cancer sites.
site_abbr character: Cancer site abbreviated description.
ages integer: Starting age of each age group.
cancer numeric: Number of (registered) cancer cases.
cancer_death numeric: Number of cancer deaths (cancer mortality).
death numeric: Number of deaths (all-cause mortality).
pys numeric: The size of the mid-year population.

Details

The data is collected from two main sources: cancer cases and deaths from the Global Cancer Observatory Today, and population and all-cause deaths from the World Population Prospects 2022. Seven site-age cells have a missing cancer or cancer-death estimate in the source data. The all-cancers- excluding-NMSC series is complete because it is derived from the difference between all cancers and NMSC when necessary.

Source

The cancer cases and cancer deaths are obtained from the Global Cancer Observatory Today <https://gco.iarc.fr/today/en>, while the population counts and all-cause deaths are sourced from the World Population Prospects 2022 <https://population.un.org/wpp/>.

References

Global Cancer Observatory Today: <https://gco.iarc.fr/today/en> World Population Prospects 2022: <https://population.un.org/wpp/>

Examples

```
# Load the dataset
data(GCO_Today_2022)

# Display the first few rows
head(GCO_Today_2022)
```

get_risk

Calculate the risk value for a specific age range

Description

Calculate the risk value for a specific age range

Usage

```
get_risk(object, age_start, age_end, age_combine = c("cross", "pairwise"))

## S3 method for class 'devcan'
get_risk(object, age_start, age_end, age_combine = c("cross", "pairwise"))

## S3 method for class 'amp'
get_risk(object, age_start, age_end, age_combine = c("cross", "pairwise"))

## S3 method for class 'wun'
get_risk(object, age_start, age_end, age_combine = c("cross", "pairwise"))

## S3 method for class 'cumu'
get_risk(object, age_start, age_end, age_combine = c("cross", "pairwise"))
```

Arguments

object	Returned object from a risk model calculated using the devcan, amp, or wun function.
age_start	Starting age(s) used to calculate risk over one or multiple age ranges.
age_end	Ending age(s) used to calculate risk over one or multiple age ranges.
age_combine	How age vectors are combined: "cross" or "pairwise".

Value

Risks according to the age ranges.

pairwise_ztest	<i>Pairwise Z tests for risk differences among groups</i>
----------------	---

Description

pairwise_ztest() performs pairwise large-sample Z tests for risk estimates across two or more independent groups. The input should be a grouped result returned by `calc_ltr_df()` with `return_variance = TRUE`.

Usage

```
pairwise_ztest(
  x,
  group,
  ref = NULL,
  alpha = 0.05,
  alternative = c("two.sided", "less", "greater"),
  p_adjust_method = "holm",
  digits = 6
)
```

Arguments

x	A grouped data frame returned by <code>calc_ltr_df()</code> with <code>return_variance = TRUE</code> .
group	Character string giving the grouping column in x.
ref	Optional reference group value. If NULL, all pairwise comparisons are performed. If supplied, each non-reference group is compared against ref.
alpha	Alpha level for the confidence interval of each difference.
alternative	Alternative hypothesis. Options are "two.sided", "less", or "greater". The alternative is applied to <code>risk_compare - risk_ref</code> .
p_adjust_method	Method passed to <code>stats::p.adjust()</code> for multiplicity adjustment. Use "none" to return unadjusted P values.
digits	Integer indicating the number of decimal places used to round the returned estimates and test statistics.

Value

A data frame containing group labels, matched age ranges, risks, risk difference, standard error of the difference, Z statistic, raw P value, adjusted P value, and confidence interval for the difference.

Examples

```
data("seer_example_data")
all_sites <- seer_example_data[
  seer_example_data$site == "All" & seer_example_data$sex %in% c(1, 2),
]
risks <- calc_ltr_df(
  all_sites,
  by = "sex",
  ages = ages,
  cancer = cancer,
  cancer_death = cancer_death,
  death = death,
  pys = pys,
  maj_method = "constant",
  return_variance = TRUE
)
pairwise_ztest(risks, group = "sex")
```

pmaj

Smooth a Single Rate Using Piecewise Mid-Age Group Joinpoint (PMAJ) Method

Description

This function implements the Piecewise Mid-Age Group Joinpoint (PMAJ) smoothing method to approximate a smoothed rate (e.g., cancer incidence) over finer sub-intervals. It uses linear interpolation between midpoints of original age intervals to create piecewise constant rates on a finer grid.

Usage

```
pmaj(ages, rate_hat, maj_method = "pmaj", pmaj_sub_interval = 0.5)
```

Arguments

ages	Starting ages of the original age intervals (e.g., 0, 5, 10, ..., 95).
rate_hat	Estimated rates for the original intervals (e.g., cancer incidence rates).
maj_method	The method to use for smoothing the rates, default is "pmaj". • "pmaj" for Piecewise Mid-Age Group Joinpoint; • "constant" for simple piecewise constant. Note: To approximate the exact MAJ (Mid-Age Group Joinpoint) method, use "pmaj" with a small value for pmaj_sub_interval (e.g., 0.01), as MAJ requires numerical integration for exact computation but can be closely approximated this way.

`pmaj_sub_interval`

The width of sub-intervals for the PMAJ approximation (default = 0.5 years).

Value

Numeric vector of smoothed rates on the fine grid. When an original age interval is not evenly divisible by `pmaj_sub_interval`, the final sub-interval is shortened so that the original age boundaries are retained.

References

Fay MP. Estimating age conditional probability of developing disease from surveillance data. *Popul Health Metr.* 2004 Jul 27;2(1):6. doi: 10.1186/1478-7954-2-6. PMID: 15279675; PMCID: PMC517510.

Examples

```
ages <- seq(0, 95, 5)
cancer <- c(0, 0, 1, 9, 43, 335, 1116, 2670, 5183, 7392, 8012, 7341, 7010,
7651, 8060, 7146, 4754, 2574, 952, 273)
pys <- c(4052953, 4032790, 3784789, 3810986, 3675646, 4138795, 4575728,
4831799, 4578168, 3906260, 3054146, 2353577, 1981443, 1988371, 1838556,
1541002, 1083867, 629172, 299128, 114178)
rate_hat_c <- cancer / pys
pmaj(ages, rate_hat_c)
pmaj(ages, rate_hat_c, maj_method = "constant")
# To approximate MAJ, use small sub-interval with PMAJ
pmaj(ages, rate_hat_c, pmaj_sub_interval = 0.01)
```

`seer_example_data`

Example Cancer Data from SEER

Description

Example Cancer Data from SEER

Usage

`seer_example_data`

Format

A data frame with 40 rows and 7 variables:

site Character. Breast for breast cancer, All for all sites.

period Character. Time period corresponding to data.

sex Integer. Sex indicator (0 = Both, 1 = Male, 2 = Female).

ages Integer. The starting age of the interval (e.g., 0, 1, 5...).

cancer Integer. Count of first primary incident cancer cases.

cancer_death Integer. Deaths specifically attributed to this cancer site.

death Integer. Total deaths from all causes in the age interval.

pys Numeric. Person-years at risk, typically the mid-year population.

Source

[doi:10.1002/sim.1428](https://doi.org/10.1002/sim.1428)

References

Fay MP, Pfeiffer R, Cronin KA, Le C, Feuer EJ. (2003). Age-conditional probabilities of developing cancer. *Statistics in Medicine*, 22(11), 1837–1848.

National Cancer Institute. DevCan 6.7.5: SEER 21 Incidence and Mortality, 2019–2022 (2020 Excluded).

Examples

```
data(seer_example_data)
# View the structure of the Fay et al. example data
head(seer_example_data)
```

trend_test	<i>Test linear trend in risk across ordered groups</i>
------------	--

Description

`trend_test()` performs a weighted linear trend test for risk estimates across three or more ordered groups. The input should usually be a grouped result returned by `calc_ltr_df()` with `return_variance = TRUE`.

Usage

```
trend_test(
  x,
  group,
  score = NULL,
  alpha = 0.05,
  alternative = c("two.sided", "less", "greater"),
  digits = 6
)
```

Arguments

x	A data frame returned by <code>calc_ltr_df()</code> with <code>return_variance = TRUE</code> .
group	Character string giving the ordered grouping column in x.
score	Optional numeric scores for the ordered groups. If NULL, numeric group values are used directly, factor levels are converted to 1, 2, ..., and character groups are scored by order of first appearance with a warning. A named numeric vector can be used to explicitly map group values to scores.
alpha	Alpha level for the confidence interval of the slope.
alternative	Alternative hypothesis for the slope. Options are "two.sided", "less", or "greater".
digits	Integer indicating the number of decimal places used to round returned estimates and test statistics.

Value

A data frame with one row per age range containing the number of groups, trend slope, standard error, Z statistic, P value, and confidence interval for the slope.

wun

Compute probabilities used for estimation of lifetime risk

Description

Compute probabilities used for estimation of lifetime risk

Usage

```
wun(
  ages,
  cancer,
  cancer_death,
  death,
  pys,
  correct_for_surgery = FALSE,
  H = NULL,
  Ch = NULL,
  type = "developing",
  cohort_size = 1,
  last_age_widths = Inf,
  maj_method = "constant",
  pmaj_sub_interval = 0.5
)
```

Arguments

ages	Starting ages of each age interval (e.g., 0, 5, 10, ..., 85).
cancer	Number of first primary cancer diagnoses in each age interval.
cancer_death	Number of deaths due to cancer in each age interval.
death	Number of all deaths (all causes combined) in each age interval.
pys	Person-years at risk corresponding to each age interval.
correct_for_surgery	Logical value, whether correct for factors.
H	Numeric vector of the number of surgeries (e.g., hysterectomies) in each interval. Required if <code>correct_for_surgery = TRUE</code> . Corresponds to H_x in the paper, representing the total number of hysterectomies performed in the interval.
Ch	Numeric vector of the number of new cancer cases treated by the surgery in each interval. Required if <code>correct_for_surgery = TRUE</code> . Corresponds to Ch_x in the paper, representing the number of new cancer cases treated by a hysterectomy.
type	Characters "developing" or "dying" indicate estimate the probability of developing cancer or dying from it.
cohort_size	Size of the cohort.
last_age_widths	Width of the last age group (default = Inf).
maj_method	Character string. The age-specific rate model: "constant" preserves the original Wun age intervals, while "pmaj" uses the Piecewise Mid-Age Group Joinpoint approximation. Default is "constant" for backward compatibility.
pmaj_sub_interval	Numeric. Sub-interval size for PMAJ (default = 0.5 years). Ignored when <code>maj_method = "constant"</code> .

Value

A list of class "wun" containing:

- age: Model age boundaries. These are the input ages for the constant model and the finer PMAJ boundaries for the PMAJ model.
- contrib_a: Contributions to new cancers (a_x).
- condi_p: Conditional probabilities.
- l_cf: Cancer-free (and surgery-free if applicable) alive at start.
- d: Non-cancer deaths among cancer-free.
- s: Non-cancer surgeries (if `correct_for_surgery = TRUE`).

References

Wun, L. M., Merrill, R. M., & Feuer, E. J. (1998). Estimating lifetime and age-conditional probabilities of developing cancer. *Lifetime Data Analysis*, 4(2), 169-186.

ztest

*Test the difference in risk between two groups***Description**

`ztest()` compares risk estimates from two independent groups using a large-sample Z test. The inputs should be results returned by `calc_ltr()` or `calc_ltr_df()` with `return_variance = TRUE`.

Usage

```
ztest(
  x,
  y = NULL,
  group = NULL,
  ref = NULL,
  compare = NULL,
  alpha = 0.05,
  alternative = c("two.sided", "less", "greater"),
  digits = 6
)

## S3 method for class 'data.frame'
ztest(
  x,
  y = NULL,
  group = NULL,
  ref = NULL,
  compare = NULL,
  alpha = 0.05,
  alternative = c("two.sided", "less", "greater"),
  digits = 6
)
```

Arguments

<code>x</code>	A data frame returned by <code>calc_ltr()</code> with <code>return_variance = TRUE</code> , or a grouped result returned by <code>calc_ltr_df()</code> .
<code>y</code>	Optional. A second data frame returned by <code>calc_ltr()</code> with <code>return_variance = TRUE</code> . If <code>NULL</code> , <code>group</code> must be supplied and the two groups are selected from <code>x</code> .
<code>group</code>	Optional character string giving the grouping column in <code>x</code> . Used when <code>y = NULL</code> .
<code>ref</code>	Reference group value when <code>group</code> is supplied.
<code>compare</code>	Comparison group value when <code>group</code> is supplied. The reported difference is <code>risk_compare - risk_ref</code> .
<code>alpha</code>	Alpha level for the confidence interval of the difference.

alternative	Alternative hypothesis. Options are "two.sided", "less", or "greater". For grouped input, the alternative is applied to $\text{risk_compare} - \text{risk_ref}$; for two-input usage it is applied to $\text{risk_x} - \text{risk_y}$.
digits	Integer indicating the number of decimal places used to round the returned estimates and test statistics.

Value

A data frame containing group labels, matched age ranges, risks, risk difference, standard error of the difference, Z statistic, P value, and confidence interval for the difference.

Examples

```
data("seer_example_data")
breast <- seer_example_data[seer_example_data$site == "Breast", ]
female <- breast[breast$sex == 2, ]
male <- breast[breast$sex == 1, ]
if (nrow(female) > 0 && nrow(male) > 0) {
  risk_female <- calc_ltr(
    female$ages, female$cancer, female$cancer_death, female$death,
    female$pys, maj_method = "constant", return_variance = TRUE
  )
  risk_male <- calc_ltr(
    male$ages, male$cancer, male$cancer_death, male$death,
    male$pys, maj_method = "constant", return_variance = TRUE
  )
  ztest(risk_male, risk_female)
}
```

Index

- * **GCO**
 - GCO_Today_2022, [13](#)
- * **GLOBOCAN**
 - GCO_Today_2022, [13](#)
- * **datasets**
 - seer_example_data, [17](#)
- amp, [2](#)
- as.character(), [12](#)
- calc_ltr, [3](#)
- calc_ltr(), [6](#), [7](#), [12](#), [21](#)
- calc_ltr_df, [6](#)
- calc_ltr_df(), [12](#), [15](#), [18](#), [19](#), [21](#)
- clear_ltr_cache, [8](#)
- cumulative, [9](#)
- devcan, [10](#)
- format_risk_ci, [11](#)
- GCO_Today_2022, [13](#)
- get_risk, [14](#)
- get_risk.cumu, [10](#)
- ltr_cache_info (clear_ltr_cache), [8](#)
- pairwise_ztest, [15](#)
- parallel::makePSOCKcluster(), [7](#), [9](#)
- pmaj, [16](#)
- seer_example_data, [17](#)
- stats::p.adjust(), [15](#)
- trend_test, [18](#)
- wun, [19](#)
- ztest, [21](#)