

Package ‘CohortSurvival’

April 2, 2024

Title Estimate Survival from Common Data Model Cohorts

Version 0.4.0

Description Estimate survival using data mapped to the Observational Medical Outcomes Partnership common data model. Survival can be estimated based on user-defined study cohorts.

License Apache License (≥ 2)

Encoding UTF-8

RoxygenNote 7.3.1

Imports CDMConnector ($\geq 1.3.0$), omopgenerics, checkmate, cli, DBI, dplyr, dbplyr, magrittr, lubridate, broom, PatientProfiles, visOmopResults, rlang ($\geq 0.4.11$), survival, scales, stringr, tibble, tidy

Suggests testthat ($\geq 3.0.0$), CodelistGenerator, roxygen2, knitr, tictoc, rmarkdown, ggplot2, duckdb

Config/testthat/edition 3

Config/testthat/parallel true

VignetteBuilder knitr

URL <https://darwin-eu-dev.github.io/CohortSurvival/>

NeedsCompilation no

Author Edward Burn [aut, cre] (<<https://orcid.org/0000-0002-9286-1128>>),
Kim Lopez [aut] (<<https://orcid.org/0000-0002-8462-8668>>),
Marti Catala [ctb] (<<https://orcid.org/0000-0003-3308-9905>>),
Xintong Li [ctb] (<<https://orcid.org/0000-0003-3308-9905>>),
Danielle Newby [ctb] (<<https://orcid.org/0000-0002-3001-1478>>)

Maintainer Edward Burn <edward.burn@endorms.ox.ac.uk>

Repository CRAN

Date/Publication 2024-04-02 17:35:00 UTC

R topics documented:

addCohortSurvival	2
asSurvivalResult	3
benchmarkCohortSurvival	4
estimateCompetingRiskSurvival	6
estimateSingleEventSurvival	8
generateDeathCohortSet	10
mockMGUS2cdm	11
plotSurvival	12
splitNameLevel	13
survivalParticipants	14
tableSurvival	15
uniteNameLevel	15

Index	17
--------------	-----------

addCohortSurvival	<i>Add survival information to a cohort table</i>
-------------------	---

Description

Add survival information to a cohort table

Usage

```
addCohortSurvival(
  x,
  cdm,
  outcomeCohortTable,
  outcomeCohortId = 1,
  outcomeDateVariable = "cohort_start_date",
  outcomeWashout = Inf,
  censorOnCohortExit = FALSE,
  censorOnDate = NULL,
  followUpDays = Inf
)
```

Arguments

x	cohort table to add survival information
cdm	CDM reference
outcomeCohortTable	The outcome cohort table of interest.
outcomeCohortId	ID of event cohorts to include. Only one outcome (and so one ID) can be considered.

outcomeDateVariable Variable containing date of outcome event
 outcomeWashout Washout time in days for the outcome
 censorOnCohortExit If TRUE, an individual's follow up will be censored at their cohort exit
 censorOnDate if not NULL, an individual's follow up will be censored at the given date
 followUpDays Number of days to follow up individuals (lower bound 1, upper bound Inf)

Value

Two additional columns will be added to x. The "time" column will contain number of days to censoring. The "status" column will indicate whether the patient had the event (value: 1), or did not have the event (value: 0)

Examples

```

cdm <- mockMGUS2cdm()
cdm$mgus_diagnosis <- cdm$mgus_diagnosis %>%
  addCohortSurvival(
    cdm = cdm,
    outcomeCohortTable = "death_cohort",
    outcomeCohortId = 1
  )

```

asSurvivalResult *A tidy implementation of the summarised_characteristics object.*

Description

A tidy implementation of the summarised_characteristics object.

Usage

```
asSurvivalResult(result)
```

Arguments

result A summarised_characteristics object.

Value

A tibble with a tidy version of the summarised_characteristics object.

Examples

```
cdm <- mockMGUS2cdm()
surv <- estimateSingleEventSurvival(
  cdm = cdm,
  targetCohortTable = "mgus_diagnosis",
  targetCohortId = 1,
  outcomeCohortTable = "death_cohort",
  outcomeCohortId = 1,
  eventGap = 7
) %>%
  asSurvivalResult()
```

benchmarkCohortSurvival

Estimate performance of estimateSurvival function for benchmarking

Description

Estimate performance of estimateSurvival function for benchmarking

Usage

```
benchmarkCohortSurvival(
  cdm,
  targetSize,
  outcomeSize,
  outcomeDateVariable = "cohort_start_date",
  competingOutcomeSize = NULL,
  competingOutcomeDateVariable = "cohort_start_date",
  censorOnCohortExit = FALSE,
  censorOnDate = NULL,
  followUpDays = Inf,
  strata = NULL,
  eventGap = 30,
  estimateGap = 1,
  minCellCount = 5,
  returnParticipants = FALSE
)
```

Arguments

cdm	CDM reference
targetSize	number of people in the target cohort table
outcomeSize	number of people in the outcome cohort table

outcomeDateVariable	Variable containing date of outcome event
competingOutcomeSize	number of people in the competing outcome cohort table
competingOutcomeDateVariable	Variable containing date of competing event
sensorOnCohortExit	If TRUE, an individual's follow up will be censored at their cohort exit
sensorOnDate	if not NULL, an individual's follow up will be censored at the given date
followUpDays	Number of days to follow up individuals (lower bound 1, upper bound Inf)
strata	strata
eventGap	Days between time points for which to report survival estimates. First day will be day zero with risk estimates provided for times up to the end of follow-up, with a gap in days equivalent to eventGap.
estimateGap	vector of time points at which to give survival estimates, if NULL estimates at all times are calculated
minCellCount	The minimum number of events to reported, below which results will be obscured. If 0, all results will be reported.
returnParticipants	Either TRUE or FALSE. If TRUE, references to participants from the analysis will be returned allowing for further analysis.

Value

tibble with performance of estimateSurvival function information, according to the selected input parameters

Examples

```
cdm <- mockMGUS2cdm()
cdm$condition_occurrence <- cdm$death_cohort %>%
dplyr::rename("condition_start_date" = "cohort_start_date",
              "condition_end_date" = "cohort_end_date") %>%
  dplyr::compute()
surv_timings <- benchmarkCohortSurvival(
  cdm, targetSize = 100, outcomeSize = 20)
```

```
estimateCompetingRiskSurvival
```

Estimate survival for a given event and competing risk of interest using cohorts in the OMOP Common Data Model

Description

Estimate survival for a given event and competing risk of interest using cohorts in the OMOP Common Data Model

Usage

```
estimateCompetingRiskSurvival(
  cdm,
  targetCohortTable,
  targetCohortId = NULL,
  outcomeCohortTable,
  outcomeCohortId = NULL,
  outcomeDateVariable = "cohort_start_date",
  outcomeWashout = Inf,
  competingOutcomeCohortTable,
  competingOutcomeCohortId = NULL,
  competingOutcomeDateVariable = "cohort_start_date",
  competingOutcomeWashout = Inf,
  censorOnCohortExit = FALSE,
  censorOnDate = NULL,
  followUpDays = Inf,
  strata = NULL,
  eventGap = 30,
  estimateGap = 1,
  minimumSurvivalDays = 1,
  minCellCount = 5,
  returnParticipants = FALSE
)
```

Arguments

cdm	CDM reference
targetCohortTable	targetCohortTable
targetCohortId	targetCohortId
outcomeCohortTable	The outcome cohort table of interest.
outcomeCohortId	ID of event cohorts to include. Only one outcome (and so one ID) can be considered.

outcomeDateVariable	Variable containing date of outcome event
outcomeWashout	Washout time in days for the outcome
competingOutcomeCohortTable	The competing outcome cohort table of interest.
competingOutcomeCohortId	ID of event cohorts to include. Only one competing outcome (and so one ID) can be considered.
competingOutcomeDateVariable	Variable containing date of competing outcome event
competingOutcomeWashout	Washout time in days for the competing outcome
censorOnCohortExit	If TRUE, an individual's follow up will be censored at their cohort exit
censorOnDate	if not NULL, an individual's follow up will be censored at the given date
followUpDays	Number of days to follow up individuals (lower bound 1, upper bound Inf)
strata	strata
eventGap	Days between time points for which to report survival events, which are grouped into the specified intervals.
estimateGap	Days between time points for which to report survival estimates. First day will be day zero with risk estimates provided for times up to the end of follow-up, with a gap in days equivalent to eventGap.
minimumSurvivalDays	Minimum number of days required for the main cohort to have survived
minCellCount	The minimum number of events to reported, below which results will be obscured. If 0, all results will be reported.
returnParticipants	Either TRUE or FALSE. If TRUE, references to participants from the analysis will be returned allowing for further analysis.

Value

tibble with survival information for desired cohort, including: time, people at risk, survival probability, cumulative incidence, 95 CIs, strata and outcome. A tibble with the number of events is outputted as an attribute of the output

Examples

```
cdm <- mockMGUS2cdm()
surv <- estimateCompetingRiskSurvival(
  cdm = cdm,
  targetCohortTable = "mgus_diagnosis",
  targetCohortId = 1,
  outcomeCohortTable = "progression",
  outcomeCohortId = 1,
```

```

    competingOutcomeCohortTable = "death_cohort",
    competingOutcomeCohortId = 1,
    eventGap = 7
)

```

```
estimateSingleEventSurvival
```

Estimate survival for a given event of interest using cohorts in the OMOP Common Data Model

Description

Estimate survival for a given event of interest using cohorts in the OMOP Common Data Model

Usage

```

estimateSingleEventSurvival(
  cdm,
  targetCohortTable,
  targetCohortId = NULL,
  outcomeCohortTable,
  outcomeCohortId = NULL,
  outcomeDateVariable = "cohort_start_date",
  outcomeWashout = Inf,
  censorOnCohortExit = FALSE,
  censorOnDate = NULL,
  followUpDays = Inf,
  strata = NULL,
  eventGap = 30,
  estimateGap = 1,
  minimumSurvivalDays = 1,
  minCellCount = 5,
  returnParticipants = FALSE
)

```

Arguments

cdm	CDM reference
targetCohortTable	targetCohortTable
targetCohortId	targetCohortId
outcomeCohortTable	The outcome cohort table of interest.
outcomeCohortId	ID of event cohorts to include. Only one outcome (and so one ID) can be considered.

outcomeDateVariable	Variable containing date of outcome event
outcomeWashout	Washout time in days for the outcome
sensorOnCohortExit	If TRUE, an individual's follow up will be censored at their cohort exit
sensorOnDate	if not NULL, an individual's follow up will be censored at the given date
followUpDays	Number of days to follow up individuals (lower bound 1, upper bound Inf)
strata	strata
eventGap	Days between time points for which to report survival events, which are grouped into the specified intervals.
estimateGap	Days between time points for which to report survival estimates. First day will be day zero with risk estimates provided for times up to the end of follow-up, with a gap in days equivalent to eventGap.
minimumSurvivalDays	Minimum number of days required for the main cohort to have survived
minCellCount	The minimum number of events to reported, below which results will be obscured. If 0, all results will be reported.
returnParticipants	Either TRUE or FALSE. If TRUE, references to participants from the analysis will be returned allowing for further analysis.

Value

tibble with survival information for desired cohort, including: time, people at risk, survival probability, cumulative incidence, 95 CIs, strata and outcome. A tibble with the number of events is outputted as an attribute of the output

Examples

```
cdm <- mockMGUS2cdm()
surv <- estimateSingleEventSurvival(
  cdm = cdm,
  targetCohortTable = "mgus_diagnosis",
  targetCohortId = 1,
  outcomeCohortTable = "death_cohort",
  outcomeCohortId = 1,
  eventGap = 7
)
```

`generateDeathCohortSet`*To create a death cohort*

Description

To create a death cohort

Usage

```
generateDeathCohortSet(  
  cdm,  
  name,  
  cohortTable = NULL,  
  cohortId = NULL,  
  overwrite = FALSE  
)
```

Arguments

<code>cdm</code>	CDM reference
<code>name</code>	name for the created death cohort table
<code>cohortTable</code>	name of the cohort table to create a death cohort for
<code>cohortId</code>	name of the cohort table to create a death cohort for
<code>overwrite</code>	Should the cohort table be overwritten if it already exists?

Value

A cohort table with a death cohort in cdm

Examples

```
library(CDMConnector)  
library(CohortSurvival)  
observation_period <- dplyr::tibble(  
  observation_period_id = c(1, 2, 3, 4, 5,6),  
  person_id = c(1, 2, 3, 4, 5,6),  
  observation_period_start_date = c(  
    rep(as.Date("1980-07-20"),6)  
  ),  
  observation_period_end_date = c(  
    rep(as.Date("2023-05-20"),6)  
  ),  
  period_type_concept_id = c(rep(0,6))  
)
```

```

deathTable <- dplyr::tibble(
  person_id = c(1,2,3),
  death_date = c(as.Date("2020-01-01"),
                 as.Date("2020-01-02"),
                 as.Date("2020-01-01"))

person <- dplyr::tibble(
  person_id = c(1, 2, 3, 4, 5),
  year_of_birth = c(rep("1990", 5)),
  month_of_birth = c(rep("02", 5)),
  day_of_birth = c(rep("11", 5)),
  gender_concept_id = c(rep(0,5)),
  ethnicity_concept_id = c(rep(0,5)),
  race_concept_id = c(rep(0,5))
)

cdm <- omopgenerics::cdmFromTables(
  tables = list(
    person = person,
    observation_period = observation_period,
    death = deathTable
  ),
  cdmName = "mock_es"
)

db <- DBI::dbConnect(duckdb::duckdb(), ":memory:")
cdm2 = CDMConnector::copy_cdm_to(db,
                                cdm,
                                schema = "main",
                                overwrite = TRUE)

attr(cdm2, "cdm_schema") <- "main"
attr(cdm2, "write_schema") <- "main"

cdm2 <- generateDeathCohortSet(cdm=cdm2,
                              name = "death_cohort",
                              overwrite = TRUE)

```

mockMGUS2cdm

Create mock CDM reference with survival::mgus2 dataset

Description

Create mock CDM reference with survival::mgus2 dataset

Usage

```
mockMGUS2cdm()
```

Value

CDM reference containing data from the survival::mgus2 dataset

Examples

```
cdm <- mockMGUS2cdm()
cdm$person
```

plotSurvival

Plot survival results

Description

Plot survival results

Usage

```
plotSurvival(
  result,
  x = "time",
  xscale = "days",
  ylim = c(0, NA),
  cumulativeFailure = FALSE,
  ribbon = TRUE,
  facet = NULL,
  colour = NULL,
  colourName = NULL
)
```

Arguments

result	Survival results
x	Variable to plot on x axis
xscale	X axis scale. Can be "days" or "years".
ylim	Limits for the Y axis
cumulativeFailure	whether to plot the cumulative failure probability instead of the survival probability
ribbon	If TRUE, the plot will join points using a ribbon
facet	Variables to use for facets
colour	Variables to use for colours
colourName	Colour legend name

Value

A plot of survival probabilities over time

Examples

```
cdm <- mockMGUS2cdm()
surv <- estimateSingleEventSurvival(cdm,
                                     targetCohortTable = "mgus_diagnosis",
                                     outcomeCohortTable = "death_cohort")

plotSurvival(surv)
```

splitNameLevel	<i>Split name and level columns into the columns.</i>
----------------	---

Description

Pivots the input dataframe so the values of the name columns are transformed into columns, which values come from the specified level column.

Usage

```
splitNameLevel(
  result,
  name = "group_name",
  level = "group_level",
  keep = FALSE,
  overall = FALSE
)
```

Arguments

result	Omop result object (summarised_result or compared_result).
name	Column with the names.
level	Column with the levels.
keep	Whether to keep the original group_name and group_level columns.
overall	Whether to keep overall column if present.

Value

A dataframe with the specified name column values as columns.

Examples

```
x <- dplyr::tibble(
  variable = "number subjects",
  value = c(10, 15, 40, 78),
  group_name = c("sex and age_group", "sex and age_group",
  "sex and age_group", "sex and age_group"),
  group_level = c("Male and <40", "Female and >40", "Male and >40", "Male and <40")
)
x %>% splitNameLevel(name = "group_name",
  level = "group_level",
  keep = FALSE)
```

survivalParticipants *Participants contributing to a survival analysis*

Description

Participants contributing to a survival analysis

Usage

```
survivalParticipants(result)
```

Arguments

result Result object

Value

References to the study participants contributing to a given analysis

Examples

```
cdm <- mockMGUS2cdm()
surv <- estimateSingleEventSurvival(cdm,
  targetCohortTable = "mgus_diagnosis",
  outcomeCohortTable = "death_cohort",
  returnParticipants = TRUE)

survivalParticipants(surv)
```

tableSurvival	<i>Table with survival summary</i>
---------------	------------------------------------

Description

Table with survival summary

Usage

```
tableSurvival(x, times = NULL)
```

Arguments

x	Result from estimateSingleEventSurvival or estimateCompetingRiskSurvival
times	Times at which to report survival in the summary table

Value

A tibble containing a summary of observed survival in days

Examples

```
cdm <- mockMGUS2cdm()
surv <- estimateSingleEventSurvival(cdm,
                                   targetCohortTable = "mgus_diagnosis",
                                   outcomeCohortTable = "death_cohort")
tableSurvival(surv, times = c(50,100,365))
```

uniteNameLevel	<i>Unite one or more columns in name-level format.</i>
----------------	--

Description

Unite one or more columns in name-level format.

Usage

```
uniteNameLevel(
  x,
  cols,
  name = "group_name",
  level = "group_level",
  keep = FALSE
)
```

Arguments

<code>x</code>	Tibble or data.frame.
<code>cols</code>	Columns to aggregate.
<code>name</code>	Column name of the name column.
<code>level</code>	Column name of the level column.
<code>keep</code>	Whether to keep the original columns.

Value

A Tibble with the new columns.

Examples

```
x <- dplyr::tibble(
  variable = "number subjects",
  value = c(10, 15, 40, 78),
  sex = c("Male", "Female", "Male", "Female"),
  age_group = c("<40", ">40", ">40", "<40")
)

x

x %>%
  CohortSurvival::uniteNameLevel(
    cols = c("sex", "age_group"),
    name = "new_column_name",
    level = "new_column_level"
  )
```


Index

`addCohortSurvival`, [2](#)
`asSurvivalResult`, [3](#)

`benchmarkCohortSurvival`, [4](#)

`estimateCompetingRiskSurvival`, [6](#)
`estimateSingleEventSurvival`, [8](#)

`generateDeathCohortSet`, [10](#)

`mockMGUS2cdm`, [11](#)

`plotSurvival`, [12](#)

`splitNameLevel`, [13](#)
`survivalParticipants`, [14](#)

`tableSurvival`, [15](#)

`uniteNameLevel`, [15](#)