Package ‘CohortSurvival’

June 29, 2024

Title Estimate Survival from Common Data Model Cohorts

Version 0.5.2

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 (>= 1.3.0), omopgenerics (>= 0.2.0), checkmate, cli, DBI, dplyr, dbplyr, magrittr, lubridate, broom, PatientProfiles, visOmopResults (>= 0.3.0), rlang (>= 0.4.11), survival, scales, stringr, tibble, tidyr, purrr

Suggests testthat (>= 3.0.0), CodelistGenerator, roxygen2, knitr, tictoc, rmarkdown, ggplot2, patchwork, cmprsk, duckdb, gt, flextable

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/0000-0002-6872-5804>), Danielle Newby [ctb] (<https://orcid.org/0000-0002-3001-1478>)

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

Repository CRAN

Date/Publication 2024-06-29 21:20:02 UTC
addCohortSurvival

Add survival information to a cohort table

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.
addCompetingRiskCohortSurvival

```
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

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

addCompetingRiskCohortSurvival

Add competing risk survival information to a cohort table

Description

Add competing risk survival information to a cohort table

Usage

```r
addCompetingRiskCohortSurvival(
   x, 
   cdm, 
   outcomeCohortTable, 
   outcomeCohortId = 1, 
   outcomeDateVariable = "cohort_start_date", 
   outcomeWashout = Inf, 
   outcomeCensorOnCohortExit = FALSE, 
   outcomeCensorOnDate = NULL, 
   outcomeFollowUpDays = Inf, 
)```


```r
addCompetingRiskCohortSurvival(
  competingOutcomeCohortTable,
  competingOutcomeCohortId = 1,
  competingOutcomeDateVariable = "cohort_start_date",
  competingOutcomeWashout = Inf,
  competingOutcomeCensorOnCohortExit = FALSE,
  competingOutcomeCensorOnDate = NULL,
  competingOutcomeFollowUpDays = 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
- `outcomeCensorOnCohortExit` If TRUE, an individual’s follow up will be censored at their cohort exit
- `outcomeCensorOnDate` if not NULL, an individual’s follow up will be censored at the given date
- `outcomeFollowUpDays` Number of days to follow up individuals (lower bound 1, upper bound Inf)
- `competingOutcomeCohortTable` The outcome cohort table of interest.
- `competingOutcomeCohortId` ID of event cohorts to include. Only one 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
- `competingOutcomeCensorOnCohortExit` If TRUE, an individual’s follow up will be censored at their cohort exit
- `competingOutcomeCensorOnDate` if not NULL, an individual’s follow up will be censored at the given date
- `competingOutcomeFollowUpDays` 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 outcome event (value: 1), competing event (value: 2) or did not have the event/is censored (value: 0)
Examples

cdm <- mockMGUS2cdm()
crsurvivaldata <- cdm$mgus_diagnosis %>%
  addCompetingRiskCohortSurvival(
    cdm = cdm,
    outcomeCohortTable = "progression",
    outcomeCohortId = 1,
    competingOutcomeCohortTable = "death_cohort",
    competingOutcomeCohortId = 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

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
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

```r
estimateCompetingRiskSurvival(
  cdm,
  targetCohortTable,
  targetCohortId = NULL,
  outcomeCohortTable,
  outcomeCohortId = NULL,
  outcomeDateVariable = "cohort_start_date",
  eventGap,
  estimateGap,
  minCellCount,
  returnParticipants
)
```

Value

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

Examples

```r
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

outcomeWashout = Inf,
competingOutcomeCohortTable,
competingOutcomeCohortId = NULL,
competingOutcomeDateVariable = "cohort_start_date",
competingOutcomeWashout = Inf,
censorOnCohortExit = FALSE,
censorOnDate = NULL,
followUpDays = Inf,
strata = NULL,
eventGap = 30,
estimateGap = 1,
restrictedMeanFollowUp = NULL,
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
estimateSingleEventSurvival

**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.

**restrictedMeanFollowUp**
Number of days of follow-up to take into account when calculating restricted mean for all cohorts.

**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**

```r
cdm <- mockMGUS2cdm()
surv <- estimateCompetingRiskSurvival(
  cdm = cdm,
  targetCohortTable = "mgus_diagnosis",
  targetCohortId = 1,
  outcomeCohortTable = "progression",
  outcomeCohortId = 1,
  competingOutcomeCohortTable = "death_cohort",
  competingOutcomeCohortId = 1,
  eventGap = 7
)
```

---

**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,
  restrictedMeanFollowUp = NULL,
  minimumSurvivalDays = 1,
  minCellCount = 5,
  returnParticipants = FALSE
)
```

Arguments

cdm
  CDM reference

targetCohortTable
  targetCohortTable

targetCohortId
  ID of event cohorts to include. Only one outcome (and so one ID) can be considered.

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)

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.
generateDeathCohortSet

**Description**

To create a death cohort

**Usage**

generateDeathCohortSet(cdm, name, cohortTable = NULL, cohortId = NULL)

**Arguments**

- **cdm**: CDM reference
- **name**: name for the created death cohort table
- **cohortTable**: name of the cohort table to create a death cohort for
- **cohortId**: name of the cohort table to create a death cohort for

**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
Value

A cohort table with a death cohort in cdm

Examples

```r
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"
)

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

cdm2 <- generateDeathCohortSet(cdm=cdm2,
```
Create mock CDM reference with survival::mgus2 dataset

Usage
mockMGUS2cdm()

Value
CDM reference containing data from the survival::mgus2 dataset

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

Additional arguments for the function `tableSurvival()`

Usage
optionsTableSurvival()

Value
The default .options named list.

Examples
```r
{
  optionsTableSurvival()
}
```
**plotSurvival**  
*Plot survival results*

**Description**

Plot survival results

**Usage**

```r
plotSurvival(
  result,
  x = "time",
  xscale = "days",
  ylim = c(0, NA),
  cumulativeFailure = FALSE,
  ribbon = TRUE,
  facet = NULL,
  colour = NULL,
  riskTable = FALSE,
  riskInterval = 30
)
```

**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
- **riskTable**: Whether to print risk table below the plot
- **riskInterval**: Interval of time to print risk table below the plot

**Value**

A plot of survival probabilities over time
Examples

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

---

**survivalParticipants**  
*Participants contributing to a survival analysis*

**Description**

Participants contributing to a survival analysis

**Usage**

```r
survivalParticipants(result)
```

**Arguments**

- `result`  
  Result object

**Value**

References to the study participants contributing to a given analysis

**Examples**

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

Description
Table with survival summary

Usage

```r
tableSurvival(
x, 
times = NULL, 
timeScale = "days", 
splitStrata = TRUE, 
header = c("estimate"), 
type = "gt", 
groupColumn = NULL, 
.options = list()
)
```

Arguments

- `x` Result from `estimateSingleEventSurvival` or `estimateCompetingRiskSurvival`
- `times` Times at which to report survival in the summary table
- `timeScale` Time unit to report survival in: days, months or years
- `splitStrata` If TRUE strata will be split into columns, otherwise "strata_name" and "strata_level" columns will be kept.
- `header` A vector containing which elements should go into the header. Allowed are: cdm_name, group, strata, additional, variable, estimate, and settings.
- `type` Type of desired formatted table, possibilities: "gt", "flextable", and "tibble".
- `groupColumn` Columns to use as group labels.
- `.options` Named list with additional formatting options. `CohortSurvival::optionsTableSurvival()` shows allowed arguments and their default values.

Value
A tibble containing a summary of observed survival in the required units

Examples

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

addCohortSurvival, 2
addCompetingRiskCohortSurvival, 3
asSurvivalResult, 5
benchmarkCohortSurvival, 6
estimateCompetingRiskSurvival, 7
estimateSingleEventSurvival, 9
generateDeathCohortSet, 11
mockMGUS2cdm, 13
optionsTableSurvival, 13
plotSurvival, 14
survivalParticipants, 15
tableSurvival, 16