Package ‘reconstructKM’

November 25, 2020

Type Package
Title Reconstruct Individual-Level Data from Published KM Plots
Version 0.3.0
Description Functions for reconstructing individual-level data (time, status, arm) from Kaplan-MEIER curves published in academic journals (e.g. NEJM, JCO, JAMA). The individual-level data can be used for re-analysis, meta-analysis, methodology development, etc. This package was used to generate the data for commentary such as Sun, Rich, & Wei (2018) <doi:10.1056/NEJMc1808567>. Please see the vignette for a quickstart guide.

Encoding UTF-8
LazyData true
Depends R (>= 3.5.0)
Imports stats, dplyr, magrittr, survival, rlang, stats4, survminer
License GPL-3
RoxygenNote 7.1.1
Suggests knitr, rmarkdown, testthat
VignetteBuilder knitr
NeedsCompilation no
Author Ryan Sun [aut, cre]
Maintainer Ryan Sun <ryansun.work@gmail.com>
Repository CRAN
Date/Publication 2020-11-25 13:40:02 UTC

R topics documented:

add_clicks ......................................................... 2
CIC_reconstruct .................................................. 3
format_raw_tabs .................................................. 4
integrate_survdat ............................................... 4
KM_reconstruct .................................................. 5
add_clicks

Description

When there are more clicks in the composite (overall) outcome curve, we need to add them to the subdistribution curves. Find the time points in the composite data that are furthest away from the times in clicksDF, add these times to clicksDF with 0 jumps in cuminc.

Usage

```r
add_clicks(clicksDF, targetTimes, nAdd)
```

Arguments

- **clicksDF**: A data frame with the two columns, time and cuminc.
- **targetTimes**: A vector of times from the composite KM plot.
- **nAdd**: Number of times to add to clicksDF.

Value

An augmented clicksDF with extra rows (no cuminc jumps in those extra times).

Examples

```r
clicksDF <- data.frame(time=0:10, cuminc=seq(from=0, to=1, by=0.1))
add_clicks(clicksDF, targetTimes = runif(n=14, min=0, max=10), nAdd=5)
```
CIC_reconstruct

Reconstruct cumulative incidence curves

Description

In competing risks situations, papers may provide one overall KM plot for the composite outcome of event 1 or event 2 as well as cumulative incidence plots for the each event separately. We can use these three plots to reconstruct individual level data with event-specific labels (censored, event 1, or event 2). Can also handle the case when the CIC for event 2 is not given. Run this separately for each arm.

Usage

CIC_reconstruct(overallIPD, clicks1, arm, clicks2 = NULL)

Arguments

overallIPD The individual patient data from the overall (composite outcome) plot that has already been processed through reconstructKM. Should have three columns: time, status, and arm.

clicks1 A data.frame with "time" and "cuminc" columns that are output from the digitizing software, similar to what you would input for reconstructKM except it’s a cumulative incidence function for a specific event, not a survival function (make sure first click is (0,0)).

arm The arm corresponding to clicks1 and possibly clicks2.

clicks2 Same as clicks1 but for the second event if it’s provided. Default is null.

Value

An augmented version of overallIPD that additionally gives the cause of the event (cause 1 or cause 2) as a fourth "event" column.

Examples

data(pembro_clicks)
data(pembro_NAR)
augTabs <- format_raw_tabs(raw_NAR=pembro_NAR, raw_surv=pembro_clicks)
reconstruct <- KM_reconstruct(aug_NAR=augTabs$aug_NAR, aug_surv=augTabs$aug_surv)
IPD <- data.frame(arm=1, time=reconstruct$IPD_time, status=reconstruct$IPD_event)
clicks1 <- dplyr::mutate(pembro_clicks, cuminc=1-survival)
CIC_reconstruct(overallIPD = IPD, clicks1 = clicks1, arm=1, clicks2=NULL)
**format_raw_tabs**

*Format raw survival and NAR tables so they are ready for reconstruction algorithm*

**Description**

Augment a raw number at risk table with the necessary information to run the reconstruction algorithm.

**Usage**

```r
format_raw_tabs(raw_NAR, raw_surv, tau = NULL)
```

**Arguments**

- `raw_NAR`: A data frame with the columns 'time' and 'NAR' at least.
- `raw_surv`: A data frame with the columns 'time' and 'survival' at least.
- `tau`: End of follow-up time, defaults to last time in NAR table.

**Value**

A list with `aug_NAR` and `aug_surv`, properly cleaned tables that can be used as input in `KM_reconstruct()`.

**Examples**

```r
data(pembro_clicks)
data(pembro_NAR)
augTabs <- format_raw_tabs(raw_NAR=pembro_NAR, raw_surv=pembro_clicks)
```

**integrate_survdat**

*Integrate area under curve for single arm*

**Description**

Calculate nonparametric RMST for a single arm up to `tau` for data.frame with time and status

**Usage**

```r
integrate_survdat(dat, tau, alpha = 0.05)
```

**Arguments**

- `dat`: Data frame of time-to-event data which MUST have the columns 'time' and 'status' exactly.
- `tau`: The cutoff time, a scalar
- `alpha`: Level for confidence interval
**Value**

data.frame with rows for RMST and RMTL and columns for estimate, std err, pvalue, and CI

**Examples**

time <- rnorm(100)
status <- rbinom(n=100, size=1, prob=0.5)
dat <- data.frame(time=time, status=status)
integrate_survdat(dat=dat, tau=2)

---

**Description**

Reconstruct individual-level data from augmented survival table and augmented NAR table, with augmentation performed by format_raw_tabs().

**Usage**

KM_reconstruct(aug_NAR, aug_surv)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>aug_NAR</td>
<td>A data frame processed through format_raw_tabs().</td>
</tr>
<tr>
<td>aug_surv</td>
<td>A data frame processed through format_raw_tabs().</td>
</tr>
</tbody>
</table>

**Value**

A list including IPD_time, IPD_event, n_hat=n_hat, KM_hat, n_cen, n_event, int_censor

**Examples**

data(pembro_NAR)
data(pembro_clicks)
augTabs <- format_raw_tabs(raw_NAR=pembro_NAR, raw_surv=pembro_clicks)
KM_reconstruct(aug_NAR=augTabs$aug_NAR, aug_surv=augTabs$aug_surv)
nonparam_rmst  
*Calculate RMST for each arm as well as contrast*

**Description**

Non-parametric RMST function that allows for the tau (follow-up time) to be arbitrarily large. Uno package restricts it to be min(last observed event in either arm). Provides estimate, SE, CI for each arm. Provides same for difference in arms (and also p-value).

**Usage**

```r
nonparam_rmst(dat, tau, alpha = 0.05)
```

**Arguments**

- `dat`: Data frame of time-to-event data which MUST have the columns 'time', 'arm', and 'status'
- `tau`: How long of a follow-up to consider, i.e. we integrate the survival functions from 0 to tau
- `alpha`: Confidence interval is given for (alpha/2, 1-alpha/2) percentiles

**Value**

A list including data.frame of results in each arm (RMST, RMTL, SE, pvalue, CI) as well as data.frame of results for Arm1 - Arm0 RMST.

**Examples**

```r
time <- rnorm(100)
status <- rbinom(n=100, size=1, prob=0.5)
arm <- c(rep(1, 50), rep(0, 50))
dat <- data.frame(time=time, status=status, arm=arm)
nonparam_rmst(dat=dat, tau=1, alpha=0.05)
```

---

pbo_clicks  
Pembrolizumab example OS KM reconstruction clicks - placebo arm

**Description**

A dataset containing the clicks used to reconstruct the placebo OS KM curve.

**Usage**

```r
data(pbo_clicks)
```
### pbo_NAR

**Format**

A data frame with 96 rows and 2 variables, time (event time in months) and survival (probability of OS)

**References**

Gandhi et al. NEJM 2018;378(22):2078-2092

---

### pbo_NAR

Pembrolizumab example OS NAR table - placebo arm

**Description**

A dataset containing the number at risk information for the placebo OS KM curve.

**Usage**

```r
data(pbo_NAR)
```

**Format**

A data frame with 8 rows and 2 variables, time (time in months) and NAR (number still at risk)

**References**

Gandhi et al. NEJM 2018;378(22):2078-2092

---

### pembro_clicks

Pembrolizumab example OS KM reconstruction clicks - pembrolizumab arm

**Description**

A dataset containing the clicks used to reconstruct the pembrolizumab OS KM curve.

**Usage**

```r
data(pembro_clicks)
```

**Format**

A data frame with 97 rows and 2 variables, time (event time in months) and survival (probability of OS)

**References**

Gandhi et al. NEJM 2018;378(22):2078-2092
pembro_NAR  
**Pembrolizumab example OS NAR table - pembrolizumab arm**

**Description**
A dataset containing the number at risk information for the pembrolizumab OS KM curve.

**Usage**
data(pembro_NAR)

**Format**
A data frame with 8 rows and 2 variables, time (time in months) and NAR (number still at risk)

**References**
Gandhi et al. NEJM 2018;378(22):2078-2092

---

print_cox_outputs  
**Print outputs from Cox regression**

**Description**
Just a wrapper to get quantities out of a call to coxph()

**Usage**
print_cox_outputs(cox_fit, print_output = TRUE)

**Arguments**
- **cox_fit**  
  A model fitted with coxph()
- **print_output**  
  Print summary to screen if TRUE

**Value**
A list including beta, HR, SE, and CI

**Examples**
```r
time <- rnorm(100)
status <- rbinom(n=100, prob=0.5, size=1)
arm <- c(rep(1,50), rep(0,50))
temp_cox <- survival::coxph(survival::Surv(time, status) ~ arm)
print_cox_outputs(temp_cox)
```
remove_clicks

Remove clicks from subdistribution curves for reconstructing CIC

Description

When there are fewer clicks in the composite (overall) outcome curve, we need to remove them from the subdistribution curves. Find the time points in the subdistribution data that are furthest away from the composite curve times, remove those times.

Usage

remove_clicks(clicksDF, targetTimes, nRemove)

Arguments

- clicksDF: A data frame with the two columns time and cuminc.
- targetTimes: A vector of times from the composite KM plot.
- nRemove: Number of times to remove from clicksDF.

Value

A clicksDF with fewer rows.

Examples

```r
clicksDF <- data.frame(time=0:10, cuminc=seq(from=0, to=1, by=0.1))
remove_clicks(clicksDF, targetTimes = runif(n=7, min=0, max=10), nRemove=3)
```

weibull_rmst

RMST using Weibull fit

Description

RMST for time-to-event data under parametric Weibull fit for data in each arm separately. Also can provide CI for RMST estimate and difference in RMST.

Usage

weibull_rmst(num_boots = 1000, dat, tau, alpha, find_pval = FALSE, seed = NULL)
Arguments

num_boots  Number of bootstrap iterations
dat        Data frame of time-to-event data which MUST have the columns 'time', 'arm', and 'status'
tau       How long of a follow-up to consider, i.e. we integrate the survival functions from 0 to tau
alpha     Confidence interval is given for (alpha/2, 1-alpha/2) percentiles
find_pval  Boolean, if TRUE then does bootstrap under the null to find p-value of mean difference and RMST difference
seed      For reproducibility

Value

A list including out_tab (estimate and CI in both arms), trt_rmst, pbo_rmst, diff_rmst, trt_CI, pbo_CI, diff_CI. Assumes trt coded as arm 1 and placebo coded as arm 0.

Examples

time <- rexp(100)
status <- rbinom(n=100, prob=0.5, size=1)
arm <- c( rep(1, 50), rep(0, 50))
dat <- data.frame(time=time, status=status, arm=arm)
weibull_rmst(dat=dat, tau=1, alpha=0.05, num_boots=200)

weimle1

Fit Weibull distribution parameters using MLE

Description

Fit the shape and scale parameters for a Weibull distribution to the time-to-event data using MLE.

Usage

weimle1(time, status)

Arguments

time A vector of event times
status A vector of 0-1 censoring status, 0 for censored, 1 for observed

Value

A list including out (the return from mle()), shape, and scale
Examples

```r
time <- rexp(100)
status <- rbinom(n=100, size=1, prob=0.5)
weimle1(time=time, status=status)
```
Index

* datasets
  pbo_clicks, 6
  pbo_NAR, 7
  pembro_clicks, 7
  pembro_NAR, 8

add_clicks, 2
CIC_reconstruct, 3
format_raw_tabs, 4
integrate_survdat, 4
KM_reconstruct, 5
nonparam_rmst, 6
pbo_clicks, 6
pbo_NAR, 7
pembro_clicks, 7
pembro_NAR, 8
print_cox_outputs, 8
remove_clicks, 9
weibull_rmst, 9
weimle1, 10