Package 'ipcwswitch'

October 13, 2022

Title Inverse Probability of Censoring Weights to Deal with Treatment Switch in Randomized Clinical Trials

Version 1.0.4

Maintainer Nathalie Graffeo <nathalie.graffeo@univ-amu.fr>

Description Contains functions for formatting clinical trials data and implementing inverse probability of censoring weights to handle treatment switches when estimating causal treatment effect in randomized clinical trials.

Depends R (>= 2.10), survival (>= 2.42)

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports stats

NeedsCompilation no

Author Nathalie Graffeo [aut, cre],
Aurelien Latouche [aut],
Sylvie Chevret [aut]

Repository CRAN

Date/Publication 2021-02-17 08:30:02 UTC

R topics documented:

`cens.ipw` *Censoring patient initiating the other arm treatment and building a treatment censoring indicator cens*

Description

Censoring patient initiating the other arm treatment and building a treatment censoring indicator cens

Usage

```
cens.ipw(
  data,
  id,
  tstart,
  tstop,
  event,
  censTime,
  arm,
  realtrt = FALSE,
  trt.start = NULL,
  trt.stop = NULL
)
```

Arguments

`data`	a dataframe containing the following variables
`id`	the patient's id
`tstart`	the date of the beginning of the follow-up (in numeric format)
`tstop`	the date of the end of the follow-up (in numeric format)
`event`	the indicator of failure (a death is denoted by 1 at the end of the follow-up)
`censTime`	the chosen time to censor the patients (in numeric format)
`arm`	the randomized treatment (2-levels factor)
`realtrt`	the randomized treatment (2-levels factor)
`trt.start`	the time of initiation of the randomized treatment (NULL by default)
`trt.stop`	the time of termination of the randomized treatment (NULL by default)

Value

a dataframe in the long format, with the data being censored according to the input date, censTime. a treatment censoring indicator, cens, is thus added to the previous dataset to indicate such a switch. Note that this function provides the option to include in the data the treatment really taken with the corresponding dates. Then, the treatment really taken is a 3-levels factor, i.e., the two from the randomized arms and a third indicating the no-treatment case (None).

References

Graffeo, N., Latouche, A., Le Tourneau C., Chevret, S. (2019) "ipcwswitch: an R package for inverse probability of censoring weighting with an application to switches in clinical trials". Computers in biology and medicine, 111, 103339. doi : "10.1016/j.compbiomed.2019.103339"

See Also

SHIdat, timesTokeep, wideToLongTDC

Examples

```
# To obtain the times parameter, we can apply the timesTokeep function on the same
# dataframe in the wide format
kept.t <- timesTokeep(toydata, id = "id",
tstart = "randt", tstop = "lastdt",
mes.cov = list(c("ps1", "ps2", "ps3")),
time.cov = list(c("randt", "dt2", "dt3")))
# Now, we can build the long format
toy.long <- wideToLongTDC(data = toydata, id = "id",
tstart = "randt", tstop = "lastdt", event = "status",
bas.cov = c("age", "arm", "swtrtdt"),
mes.cov = list(TDconf = c("ps1", "ps2", "ps3")),
time.cov = list(c("randt", "dt2", "dt3")),
times = kept.t[[1]])
# Put dates in numeric format with tstart at 0
toy.long$tstart <- as.numeric(toy.long$tstart)
toy.long$tstop <- as.numeric(toy.long$tstop)
toy.long$swtrtdt <- as.numeric(toy.long$swtrtdt)
tabi <- split(toy.long, toy.long$id)
L.tabi   <- length(tabi)
tablist <- lapply(1:L.tabi, function(i){
    refstart <- tabi[[i]]$tstart[1]
    tabi[[i]]$tstart  <- tabi[[i]]$tstart - refstart
    tabi[[i]]$tstop <- tabi[[i]]$tstop - refstart
    tabi[[i]]$swtrtdt <- tabi[[i]]$swtrtdt - refstart
    return(tabi[[i]])
    })
    toy.long <- do.call( rbind, tablist )
# Patients are censored when initiating the other arm treatment, that is, at time swtrtdt
toy.long2 <- cens.ipw(toy.long, id = "id", tstart = "tstart", tstop = "tstop",
event = "event", arm = "arm",
realtrt = FALSE, censTime ="swtrtdt")
# Before censoring:
toy.long
# Ater censoring:
toy.long2
```

ipcw *Computing the stabilized IPCweights*

Description

Computing the stabilized IPCweights

Usage

```
ipcw(
  data,
  id,
  tstart,
  tstop,
  cens,
  arm,
  bas.cov,
  conf,
  trunc = NULL,
  type = "kaplan-meier"
)
```

Arguments

data	a dataframe containing the following variables
id	the patient's id
tstart	the date of the beginning of the follow-up (in numeric format, with the first being equal at 0)
tstop	the date of the end of the follow-up (in numeric format)
cens	the indicator of treatment censoring (denoted by 1 at the end of the follow-up)
arm	the randomized treatment (2-levels factor)
bas.cov	a vector the baseline covariates
conf	a vector of time-dependent confounders
trunc	an optional fraction for the weights. For instance, when trunc = 0.01, the left tail is truncated to the 1st percentile and the right tail is truncated to the 99th percentile
type	a character string specifying the type of survival curve. The default is type=`kaplan-meier`

Value

the initial dataframe data with stabilized IPCweights as additional arguments. By default, the untruncated stabilized weights are given. If the trunc option is not NULL then the truncated stabilized weights are also given.

References

Graffeo, N., Latouche, A., Le Tourneau C., Chevret, S. (2019) "ipcwswitch: an R package for inverse probability of censoring weighting with an application to switches in clinical trials". Computers in biology and medicine, 111, 103339. doi : "10.1016/j.compbiomed.2019.103339"

See Also

SHIdat

Examples

```
## Not run
# ipcw(toy.rep, tstart = tstart, tstop = tstop, cens = cens,
# arm="arm",
# bas.cov = c("age"),
# conf = c("TDconf"), trunc = 0.05)

# see ?SHIdat for a complete example
```

`replicRows` *Function to replicate the rows so that each patients' follow-up is split according to all event times (times parameter) up to each patient's end time*

Description

Function to replicate the rows so that each patients' follow-up is split according to all event times (times parameter) up to each patient's end time

Usage

```
replicRows(data, tstart, tstop, event, cens, times1, times2, arm)
```

Arguments

`data`	a dataframe containing the following variables
`tstart`	the date of the beginning of the follow-up (in numeric format, with the first being equal at 0)
`tstop`	the date of the end of the follow-up (in numeric format)
`event`	the indicator of failure (a death is denoted by 1 at the end of the follow-up)
`cens`	the indicator of treatment censoring (denoted by 1 at the end of the follow-up)
`times1`	a vector of times (in numeric format) indicating the times according to which the rows have to be split for patients in the first arm
`times2`	a vector of times (in numeric format) indicating the times according to which the rows have to be split for patients in the second arm
`arm`	the randomized treatment (2-levels factor)

Value

a formatted dataframe with the rows replicated according to the provided times parameter

References

Graffeo, N., Latouche, A., Le Tourneau C., Chevret, S. (2019) "ipcwswitch: an R package for inverse probability of censoring weighting with an application to switches in clinical trials". Computers in biology and medicine, 111, 103339. doi : "10.1016/j.compbiomed.2019.103339"

See Also

cens.ipw, SHIdat, timesTokeep, wideToLongTDC

Examples

```
# To obtain the times parameter, we can apply the timesTokeep function on the same
# dataframe in the wide format
kept.t <- timesTokeep(toydata, id = "id",
tstart = "randt", tstop = "lastdt",
mes.cov = list(c("ps1", "ps2", "ps3")),
time.cov = list(c("randt", "dt2", "dt3")))
# Now, we can build the long format
toy.long <- wideToLongTDC(data = toydata, id = "id",
tstart = "randt", tstop = "lastdt", event = "status",
bas.cov = c("age", "arm", "swtrtdt"),
mes.cov = list(TDconf = c("ps1", "ps2", "ps3")),
time.cov = list(c("randt", "dt2", "dt3")),
times = kept.t[[1]])
# Put dates in numeric format with tstart at 0
toy.long$tstart <- as.numeric(toy.long$tstart)
toy.long$tstop <- as.numeric(toy.long$tstop)
toy.long$swtrtdt <- as.numeric(toy.long$swtrtdt)
tabi <- split(toy.long, toy.long$id)
L.tabi   <- length(tabi)
tablist <- lapply(1:L.tabi, function(i){
    refstart <- tabi[[i]]$tstart[1]
    tabi[[i]]$tstart  <- tabi[[i]]$tstart - refstart
    tabi[[i]]$tstop <- tabi[[i]]$tstop - refstart
    tabi[[i]]$swtrtdt <- tabi[[i]]$swtrtdt - refstart
    return(tabi[[i]])
    })
    toy.long <- do.call( rbind, tablist )
# Patients are censored when initiating the other arm treatment, that is, at time swtrtdt
toy.long2 <- cens.ipw(toy.long, id = "id", tstart = "tstart", tstop = "tstop",
event = "event", arm = "arm",
realtrt = FALSE, censTime ="swtrtdt")
# We collect all event times (death for both arms and treatment censoring according to the trt arm)
rep.times1 <- unique(c(toy.long2$tstop[toy.long2$cens==1 & toy.long2$arm == "A"],
toy.long2$tstop[toy.long2$event==1]))
rep.times2 <- unique(c(toy.long2$tstop[toy.long2$cens==1 & toy.long2$arm == "B"],
toy.long2$tstop[toy.long2$event==1]))
# to put times in same order as arms levels
```

```
levels(toy.long2[, "arm"])
# Now, we can replicate the rows
toy.rep   <- replicRows(toy.long2, tstart = "tstart", tstop = "tstop",
                        event = "event", cens = "cens",
                        times1 = rep.times1, times2 = rep.times2,
                        arm = "arm")
toy.rep
```

SHIdat *A real example dataset from the randomized clinical trial SHIVA*

Description

Dataset SHIdat contains an anonymized excerpt of data from the SHIVA01 trial. This was the first randomized cli,ical trial that aimed at comparing molecularly targeted therapy based on tumour profiling (MTA) versus conventiona therapy (CT) for advanced cancer. A switch to the other arm was scheduled to be proposed at disease progression for patients in both treatment groups.

Usage

```
data("SHIdat")
```

Format

A data frame with 197 observations on the following 306 variables.

`id` a numeric vector corresponding to the patient's identifier

`bras.f` a vector containing the patient's randomized arm

`agerand` a numeric vector containing patient's age (in years) at randomization

`sex.f` a vector containing the patient's gender

`tt_Lnum` a numeric vector containing the number of previous lines of treatment

`rmh_alea.c` a numeric vector containing the Royal Marsden Hospital score segregated into two categories

`pathway.f` a vector the molecular pathway altered (pathway.f: the hormone receptors pathway, the PI3K/ AKT/mTOR pathway, and the RAF/MEK pathway)

`myps.v2`,`ps.v3`,`ps1.v1`,`ps1.v2`,`ps1.v3`,`ps1.v4`,`ps1.v5`,`ps1.v6`,`ps1.v7`,`ps1.v8`,`ps1.v9`,`ps1.v10`,`ps1.v11`,`ps1.v12`, `ps1.v`
numeric vectors containing the ECOG performance status measured at the randomization visit, the visit before the potential switch and the planned visits (maximum number of planned visits: 21)

`mytran.v1`,`tran.v2`,`tran.v3`,`tran.v4`,`tran.v5`,`tran.v6`,`tran.v7`,`tran.v8`,`tran.v9`,`tran.v10`,`tran.v11`,`tran.v12`, `tran`
numeric vectors containing the use of platelet transfusions at each of the potential 21 planned visits

`myttc.v2`,`ttc.v3`,`ttc1.v1`,`ttc1.v2`,`ttc1.v3`,`ttc1.v4`,`ttc1.v5`,`ttc1.v6`,`ttc1.v7`,`ttc1.v8`,`ttc1.v9`,`ttc1.v10`, `ttc1.v11`
: numeric vectors containing the presence of concomitant treatments at the randomization visit, the visit before the potential switch and the planned visits (maximum number of planned visits: 21)

`tox.t1`,`tox.t2`,`tox.t3`,`tox.t4`,`tox.t5`,`tox.t6`,`tox.t7`,`tox.t8`,`tox.t9`,`tox.t10`,`tox.t11`,`tox.t12`,`tox.t13`,`tox.t14`, `tox`
: numeric vectors corresponding to the presence of an adveerse event. `tox.ti` contains 1 if the patient started an adverse event linked with the treatment at `datetox.ti`, 0 if the patient ended an adverse event linked with the treatment at `datetox.ti`, and NA otherwise

`ddn` a vector containing the date of latest news

`debttCO` a vector containing the date of initiation of the other arm treatment

`ddeath` a vector containg the death date

`ddt.v1` a vector containing the date of initiation of the randomized treatment

`datt` a vector containing the date of the interruption of the randomized treatment

`dexac.v2` a vector containing the date of randomization

`dexac.v3` a vector containing the date of the visit before the potential switch

`dexac1.v1`,`dexac1.v2`,`dexac1.v3`,`dexac1.v4`,`dexac1.v5`,`dexac1.v6`,`dexac1.v7`,`dexac1.v8`,`dexac1.v9`, `dexac1.v10`,`dex`
: vectors containing the dates of the potential 21 planned visits

`datetox.t1`,`datetox.t2`,`datetox.t3`,`datetox.t4`,`datetox.t5`,`datetox.t6`,`datetox.t7`,`datetox.t8`, `datetox.t9`,`date`
: vectors containing the dates related to adverse events (as explained above)

`CO` a vector containing 1 if the patient changed treatment arm (i.e., did a switch)

`progDate` a vector containing the date of a potential progression

`progStatus` a vector containing 1 if the patient did a progression (and 0 otherwise)

`status` a vector containing the patient's status at the date of latest news (1 if died, 0 otherwise)

Details

Note that some variables were built from the original data for illustration purpose. We provided an excerpt containing only the covariates that are useful for our analysis. Note also that the SHIVA data were anonymized.

Acknowledgments: we thank the patients who volunteered to participate in this study for their dedication and the study-site staff who cared for them. This work is supported by grant ANR-10-EQPX-03 from the Agence Nationale de la Recherche (Investissements d'avenir) and Site de Recherche Integre contre le Cancer (SiRIC). High-throughput sequencing was done by the NGS platform of the Institut Curie, supported by grants ANR-10-EQPX-03 and ANR-10-INBS-09-08 from the Agence Nationale de la Recherche (Investissements d'avenir) and the Canceropole Ile-de-France.

References

Le Tourneau, C., Delord, J. P., Goncalves, A., et al. (2015). "Molecularly targeted therapy based on tumour molecular profiling versus conventional therapy for advanced cancer (SHIVA): a multicentre, open-label, proof-of-concept, randomised, controlled phase 2 trial". The Lancet Oncology, 16(13), 1324-1334. doi : "10.1016/S1470-2045(15)00188-6"

Graffeo, N., Latouche, A., Le Tourneau C., Chevret, S. (2019) "ipcwswitch: an R package for inverse probability of censoring weighting with an application to switches in clinical trials". Computers in biology and medicine, 111, 103339. doi : "10.1016/j.compbiomed.2019.103339"

See Also

cens.ipw, ipcw, replicRows, timesTokeep, wideToLongTDC

Examples

```
# To obtain the times parameter, we can apply the timesTokeep function on the same
# dataframe in the wide format
# names of the repeated measurements
vect.ps   <- c("myps.v2", "ps.v3", c(paste("ps1.v", seq(1,21), sep="")))
vect.ttc  <- c("myttc.v2", "ttc.v3", c(paste("ttc1.v", seq(1,21), sep="")))
vect.tran <- c("mytran.v1", paste("tran.v", seq(2,21), sep=""))
# corresponding dates
dates <- c("dexac.v2", "dexac.v3", c(paste("dexac1.v", seq(21), sep="")))
dates2 <- dates[!(dates %in% c("dexac.v2","dexac.v3"))]

# times to keep
kept.t <- timesTokeep(SHIdat, id = "id",
                      tstart = "dexac.v2", tstop = "ddn",
                      mes.cov = list(vect.ps, vect.ttc, vect.tran),
                      time.cov = list(dates, dates, dates2))
# Now, we can build the long format
SHIlong <- wideToLongTDC(SHIdat, id = "id",
                         tstart = "dexac.v2", tstop = "ddn",
                         event = "status",
                       bas.cov = c("agerand", "sex.f","tt_Lnum", "rmh_alea.c", "pathway.f",
                         "bras.f","debttCO","ddt.v1", "datt"),
                         mes.cov = list(f1=vect.ps, f2=vect.ttc, f3=vect.tran),
                         time.cov = list(dates, dates, dates2),
                         times = kept.t[[1]])

# Put dates in numeric format with tstart at 0
tabi <- split(SHIlong, SHIlong$id)
L.tabi   <- length(tabi)
tablist <- lapply(1:L.tabi, function(i){
  refstart <- tabi[[i]]$tstart[1]

  tabi[[i]]$tstart  <- tabi[[i]]$tstart - refstart
  tabi[[i]]$tstop <- tabi[[i]]$tstop - refstart
  tabi[[i]]$debttCO <- tabi[[i]]$debttCO - refstart # to be used in next step
  tabi[[i]]$ddt.v1 <- tabi[[i]]$ddt.v1 - refstart # to be used in the final step
  tabi[[i]]$datt <- tabi[[i]]$datt - refstart # to be used in the final step

  return(tabi[[i]])
})
SHIlong <- do.call( rbind, tablist )
colnames(SHIlong)[14:16] <- c("ps", "ttc", "tran")

# Eliminating patient not having initiated the treatment arm
SHIlong2 <- SHIlong[!is.na(SHIlong$ddt.v1),]

# Patients are censored when initiating the other arm treatment, that is, at time swtrtdt
```

```
SHIlong2 <- cens.ipw(SHIlong2, id = "id", tstart = "tstart", tstop  = "tstop",
                     event = "event", arm = "bras.f", realtrt = FALSE,
                     censTime ="debttCO")
# We collect all event times
# (death for both arms and treatment censoring according to the trt arm)
replic.times.MTA <-
    unique(c(SHIlong2$tstop[SHIlong2$cens == 1 &
    SHIlong2$bras.f == "MTA"],
    SHIlong2$tstop[SHIlong2$event == 1]))
replic.times.CT <-
    unique(c(SHIlong2$tstop[SHIlong2$cens == 1 &
    SHIlong2$bras.f == "CT"],
    SHIlong2$tstop[SHIlong2$event == 1]))
# to put times in same order as arms levels
levels(SHIlong2[,"bras.f"])
SHIrep  <- replicRows(SHIlong2, tstart = "tstart", tstop = "tstop",
                      event = "event", cens = "cens",
                      times1 = replic.times.MTA, times2=replic.times.CT,
                      arm = "bras.f")

# Estimation of the stabilized weights
library(survival)
SHIres <- ipcw(SHIrep, id = "id", tstart = tstart, tstop = tstop, cens = cens,
               arm = "bras.f",
               bas.cov = c("agerand", "sex.f", "tt_Lnum", "rmh_alea.c", "pathway.f"),
               conf = c("ps", "ttc", "tran"),
               trunc = 0.05, type = 'kaplan-meier')

# To have conventional therapy (CT) as reference
SHIres$bras.f <- relevel(SHIres$bras.f, ref="CT")

# Using the IPCW weights in Cox likelihood...
fit.stab.w <- coxph(Surv(tstart, tstop, event) ~ bras.f + agerand + sex.f +
                      tt_Lnum + rmh_alea.c + pathway.f
                    + cluster(id),
                    data = SHIres, weights = SHIres$weights.trunc)
fit.stab.w
```

`timesTokeep` *Function to keep all event times*

Description

Function to keep all event times

Usage

```
timesTokeep(data, id, tstart, tstop, mes.cov, time.cov)
```

Arguments

data	dataframe containing the following variables
id	patient's id
tstart	date of the beginning of the follow-up (in Date format)
tstop	date of the end of the follow-up (in Date format)
mes.cov	list of vectors, each of them must contain the names (in character format) of the repeated measurements related to one time-dependent covariate
time.cov	list of vectors, each of them must contain the times (in Date format) of the date when the abovementioned measurements were done

Value

list of two lists, one in Date format the other in numeric format. Each of them contains, for each patient, the event time and the times of changes in time-varying covariates

References

Graffeo, N., Latouche, A., Le Tourneau C., Chevret, S. (2019) "ipcwswitch: an R package for inverse probability of censoring weighting with an application to switches in clinical trials". Computers in biology and medicine, 111, 103339. doi : "10.1016/j.compbiomed.2019.103339"

See Also

SHIdat

Examples

```
kept.t <- timesTokeep(toydata, id = "id",
tstart = "randt", tstop = "lastdt",
mes.cov = list(c("ps1", "ps2", "ps3")),
time.cov = list(c("randt", "dt2", "dt3")))
# For example, for patient id=3, to obtain the kept times in Date format:
kept.t[[1]][[3]]
# To obtain the kept times in numeric format:
kept.t[[2]][[3]]
```

---

toydata *A short example dataset*

Description

Dataset toydata contains repeated measurements made in 3 patients. It mimics randomized clinical trials data with two parallel arms with a repeated measurement of a time-varying binary covariate, which could be the time-varying confounder acting both on the survival and treatment censoring.

Usage

```
data("toydata")
```

Format

A data frame with 3 observations on the following 12 variables.

`id` a numeric vector corresponding to the patient's identifier

`randt` a vector containing the date of the randomization visit

`lastdt` a vector containing the date of latest news

`status` a numeric vector. The value equals to 1 if the patient dies at lastdt (and 0 otherwise)

`age` a numeric vector containing patient's age (in years) at randomization

`ps1` a numeric vector containing the values (0 or 1) of a repeated measurement happening on date `randt`. Note that some of them could be missing

`ps2` a numeric vector containing the values (0 or 1) of a repeated measurement happening on date `dt2`. Note that some of them could be missing

`ps3` a numeric vector containing the values (0 or 1) of a repeated measurement happening on date `dt3`. Note that some of them could be missing

`dt2` a vector containing the dates of measurement of `ps2`. Note that some of them could be missing

`dt3` a vector containing the date of measurement `ps3`. Note that some of them could be missing

`arm` a vector containing the patient's randomized arm

`swtrtdt` a vector containing the date when the patient initiates the other arm treatment (NA if does not happen)

References

Graffeo, N., Latouche, A., Le Tourneau C., Chevret, S. (2019) "ipcwswitch: an R package for inverse probability of censoring weighting with an application to switches in clinical trials". Computers in biology and medicine, 111, 103339. doi : "10.1016/j.compbiomed.2019.103339"

Examples

```
data(toydata)
toydata
```

`wideToLongTDC` *Function from wide to long format*

Description

Function from wide to long format

Usage

```
wideToLongTDC(
  data,
  id,
  tstart,
  tstop,
  event,
  bas.cov,
  mes.cov,
  time.cov,
  times
)
```

Arguments

data	a dataframe containing the variables id, tstart, tstop, mes.cov and time.cov
id	the patient's id
tstart	date of the beginning of the follow-up (in Date format)
tstop	date of the end of the follow-up (in Date format)
event	the indicator of failure (a death is denoted by 1 at the end of the follow-up)
bas.cov	a vector containing the names (in character format) of the baseline covariates
mes.cov	a list of vectors, each of them must contain the names (in character format) of the repeated measurements related to one time-dependent covariate
time.cov	a list of vectors, each of them must contain the times (in Date format) of the date when the abovementioned measurements were done
times	a list of vectors. Each of them must contain, for each patient, the event time and the times of changes in time-varying covariates

Value

the long format version of the initial dataframe data. The repeated values included in each vector of the list mes.cov are aggregated in a variable named aas the name of the corresponding list member.

References

Graffeo, N., Latouche, A., Le Tourneau C., Chevret, S. (2019) "ipcwswitch: an R package for inverse probability of censoring weighting with an application to switches in clinical trials". Computers in biology and medicine, 111, 103339. doi : "10.1016/j.compbiomed.2019.103339"

See Also

SHIdat, timesTokeep

Examples

```
# To obtain the times parameter, we can apply the timesTokeep function on the same
# dataframe in the wide format
kept.t <- timesTokeep(toydata, id = "id",
tstart = "randt", tstop = "lastdt",
mes.cov = list(c("ps1", "ps2", "ps3")),
time.cov = list(c("randt", "dt2", "dt3")))
# Now, we can build the long format
toy.long <- wideToLongTDC(data = toydata, id = "id",
tstart = "randt", tstop = "lastdt", event = "status",
bas.cov = c("age", "arm", "swtrtdt"),
mes.cov = list(TDconf = c("ps1", "ps2", "ps3")),
time.cov = list(c("randt", "dt2", "dt3")),
times = kept.t[[1]])
toy.long
```

Index