Package: tv (via r-universe)

September 18, 2024

Title Tools for Creating Time-Varying Datasets

Version 2.1.0

Date 2024-02-16

Description Create a time-varying dataset using features, exposure, and look back specifications.

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

Imports lubridate, dplyr (>= 1.1.1), magrittr, parallelly, rlang

Depends R (>= 3.6.0)

VignetteBuilder knitr

License GPL (>= 2)

RoxygenNote 7.2.3

LazyData true

Encoding UTF-8

Config/testthat/edition 3

Repository https://eheinzen.r-universe.dev

RemoteUrl https://github.com/eheinzen/tv

RemoteRef HEAD

RemoteSha 12332face47a5760a47733772d51b4e2f692c4f4

Contents

tv *Create a time-varying dataset*

Description

Create a time-varying dataset

Usage

```
time_varying(
  x,
  specs,
  exposure,
  ...,
  grid.only = FALSE,
  time_units = c("days", "seconds"),
  id = "pat_id",
  sort = NA,
  n_cores = parallelly::availableCores(omit = 1)
)

check_tv_data(x, time_units, id, sort)

check_tv_exposure(x, expected_ids, time_units, id, ..., check_overlap = TRUE)

check_tv_specs(specs, expected_features = NULL)
```

Arguments

x	A data.frame with four columns: <id>, "feature", "datetime", "value"
specs	a data.frame with four columns: "feature", "use_for_grid", "lookback_start", "lookback_end", "aggregation". See details below.
exposure	a data.frame with (at least) three columns: <id>, "exposure_start", "exposure_stop"
...	Other arguments. Currently just passes check_overlap.
grid.only	Should just the grid be computed and returned? Useful only for debugging
time_units	What time units should be used? Seconds or days
id	The id to use. Default is "pat_id"
sort	Logical, indicating whether to sort the data before performing the analysis. By default (NA), sorting is only done when useful (that is: x$datetime is a POSIXct and time_units == "days"). A warning is issued when x$datetime is a Date to make the user aware that the input ought to be sorted to get the right answer.
n_cores	Number of cores to use. If slurm is being used, it checks the SLURM_CPUS_PER_TASK variable. Else it defaults to 1, for no parallelization.
expected_ids	A vector of expected ids based on the data.

check_overlap Should overlap be checked among exposure rows? A potentially costly operation, so you can opt out of it if you're really sure.

expected_features
A vector of expected features based on the data.

Details

The defaults for specs are to use everything for the grid creation, and to set `lookback_start=0`, with a message in both cases. Currently supported aggregation functions include counting ("count" or "n"), last-value-carried forward ("last value" or "lvcf"), any/none ("any" or "binary"), time since ("time since" or "ts"), min/max/mean, and the special "event" (for which look backs are ignored).

The look back window begins at `row_start - lookback_end` and ends at `row_start - lookback_start`. Passing NA to either look back changes the corresponding window boundary to `exposure_start`.

Value

A data.frame, with one row per grid value and one column per feature specification (plus grid columns).

Examples

```
data(tv_example)
time_varying(tv_example$data, tv_example$specs, tv_example$exposure,
             time_units = "days", id = "mcn")
```

---

tv_aggregation *Time-varying aggregation functions*

Description

Time-varying aggregation functions

Usage

```
tv_count(value, ...)

tv_any(value, ...)

tv_lvcf(value, datetime, ...)

tv_ts(datetime, current_time, ...)

tv_min(value, ...)

tv_max(value, ...)

tv_mean(value, ...)
```

```
tv_median(value, ...)

tv_sum(value, ...)
```

Arguments

value	A vector of values
...	Other arguments (not used at this time)
datetime	A datetime
current_time	The current grid row's time

Value

A scalar, indicating the corresponding aggregation over `value` or `datetime`.

---

tv_example *Example data for time-varying*

Description

Example data for time-varying

Usage

```
tv_example
```

Format

A list

data The data

specs The specs

See Also

tv

Index