Package 'WAreg'

March 6, 2026

Type Package

Title While-Alive Regression for Composite Endpoints with Cluster-Robust Inference

Version 0.1.0

Description Provides estimation and inference for while-alive regression models targeting the while-alive loss rate for composite endpoints that include recurrent events and a terminal event. The implementation supports flexible time-varying covariate effects through user-selected time bases, including B-splines, natural splines, M-splines, step functions, truncated linear bases, interval-local bases, and piecewise polynomials. Inference can be performed using cluster-robust variance estimators for cluster-randomized trials, with subject-level (IID) variance as a special case. The package includes prediction and plotting utilities and K-fold cross-validation for selecting basis and tuning parameters. Methodology is based on Fang et al. (2025) <doi:10.1093/biostatistics/kxaf047>.

License GPL-3

Encoding UTF-8

Depends R (>= 4.1)

Imports dplyr, tidyr, tibble, ggplot2, survival, nleqslv, splines, MASS, magrittr, rlang

Suggests splines2, testthat (>= 3.0.0), knitr, rmarkdown

Config/testthat/edition 3

URL https://github.com/fancy575/WAreg

BugReports https://github.com/fancy575/WAreg/issues

RoxygenNote 7.3.3

LazyData true

NeedsCompilation no

Author Xi Fang [aut, cre], Hajime Uno [aut], Fan Li [aut]

Maintainer Xi Fang <x.fang@yale.edu>

Repository CRAN

Date/Publication 2026-03-06 13:50:07 UTC

Contents

`crt_dt` *Clustered Recurrent-Time Dataset: crt_dt*

Description

A simulated dataset of clustered recurrent events with terminal/censoring outcomes and covariates, suitable for examples and tests.

Usage

```
data(crt_dt)
```

Format

A data frame with the following columns:

id Integer subject ID (within the whole sample).

cluster Integer cluster ID.

time Numeric event/censoring time.

status Integer event type indicator: `0` = censored, `1` = recurrent type 1, `2` = recurrent type 2, `3` = death (terminal).

trt Cluster-level treatment indicator carried to subjects (e.g., 0/1).

Z1 Numeric covariate.

Z2 Numeric covariate.

Details

Rows represent observed events (including censoring and death) for each subject. Multiple rows per `id` indicate multiple recurrent events; terminal/censoring rows mark the end of observation for that subject.

Source

Generated by the package's simulation utilities.

Examples

```
data(crt_dt)
head(crt_dt)
```

`irt_dt` *Individual Recurrent-Time Dataset: irt_dt*

Description

A simulated dataset of recurrent events with terminal/censoring outcomes and covariates, organized in long format.

Usage

```
data(irt_dt)
```

Format

A data frame with the following columns:

id Integer subject ID (within the whole sample).

time Numeric event/censoring time.

status Integer event type indicator: `0` = censored, `1` = recurrent type 1, `2` = recurrent type 2, `3` = death (terminal).

trt Cluster-level treatment indicator carried to subjects (e.g., 0/1).

Z1 Numeric covariate.

Z2 Numeric covariate.

Details

Long-format events: each row is an event (or censoring/death) for a subject.

Source

Generated by the package's simulation utilities.

Examples

```
data(irt_dt)
head(irt_dt)
```

plot.WA *Plot while-alive trajectory or a covariate's time-varying effect*

Description

Plot while-alive trajectory or a covariate's time-varying effect

Usage

```
## S3 method for class 'WA'
plot(
  x,
  newdata,
  t_seq,
  id = 1,
  mode = c("wa", "cov"),
  covariate = NULL,
  ylab_wa = "While-alive loss rate",
  ylab_cov = NULL,
  xlab = "Time",
  level = 0.95,
  smooth = FALSE,
  span = 0.3,
  ...
)
```

Arguments

x	A "WA" object.
newdata	Data used to rebuild the RHS design (same columns as in the model).
t_seq	Times to plot over (numeric vector).
id	Row index of newdata to use for the while-alive trajectory (mode = "wa").
mode	"wa" to plot the while-alive loss rate, or "cov" to plot a specific covariate's time-varying effect.
covariate	Character; covariate name (must appear on RHS) when mode="cov".
ylab_wa	Y-axis label for while-alive plot.
ylab_cov	Y-axis label for covariate-effect plot; default "Effect of <covariate> on \u03B7(t)".
xlab	X-axis label.
level	Confidence level for ribbons (default 0.95).
smooth	Logical; if TRUE, apply LOESS smoothing to the displayed curve/CI.
span	LOESS span used when smooth=TRUE.
...	Unused.

Value

A **ggplot2** object.

Examples

```
ex_dt <- crt_dt[crt_dt$cluster %in% c(1,2,3,4,7,10), ]
fit <- WA_fit(survival::Surv(time, status) ~ trt + Z1 + Z2,
              data = ex_dt, id="id", cluster="cluster",
              knots=seq(0, max(ex_dt$time), length.out=6),
              tau_grid=seq(0, max(ex_dt$time), length.out=6),
              basis="bz", degree=1, link="log",
              w_recur=c(1,1), w_term=2, ipcw="km")
nd <- unique(ex_dt[, c("trt","Z1","Z2")])
plot(fit, newdata = nd,
     t_seq = seq(0, max(fit$tau_grid), length.out = 200),
     id = 1, mode = "wa", smooth = TRUE)
```

---

predict.WA *Predict while-alive loss rates*

Description

Predict while-alive loss rates

Usage

```
## S3 method for class 'WA'
predict(object, newdata, t_seq, level = 0.95, ...)
```

Arguments

object	A "WA" object.
newdata	Data frame with columns matching the RHS of the fitted model. Predictions are computed for the rows of newdata.
t_seq	Numeric vector of times at which to evaluate predictions.
level	Confidence level for pointwise intervals (default 0.95).
...	Unused.

Value

A data frame with columns id (row index in newdata), t, mu (predicted while-alive rate), and CI columns lb, ub.

Examples

```
ex_dt <- crt_dt[crt_dt$cluster %in% c(1,2,3,4,7,10), ]
fit <- WA_fit(survival::Surv(time, status) ~ trt + Z1 + Z2,
              data = ex_dt, id="id", cluster="cluster",
              knots=seq(0, max(ex_dt$time), length.out=6),
              tau_grid=seq(0, max(ex_dt$time), length.out=6),
              basis="bz", degree=1, link="log",
              w_recur=c(1,1), w_term=2, ipcw="km")
nd <- unique(ex_dt[, c("trt","Z1","Z2")])
pred <- predict(fit, newdata = nd, t_seq = seq(0, max(fit$tau_grid), by = 0.2))
head(pred)
```

`summary.WA` *Summarize a WA object*

Description

Summarize a WA object

Usage

```
## S3 method for class 'WA'
summary(object, ...)
```

Arguments

`object`	A `"WA"` object from `WA_fit`.
`...`	Unused.

Value

An object of class `"summary.WA"` containing configuration and a coefficient table with estimates, standard errors, and z-scores.

Examples

```
ex_dt <- crt_dt[crt_dt$cluster %in% c(1,2,3,4,7,10), ]
fit <- WA_fit(survival::Surv(time, status) ~ trt + Z1 + Z2,
              data = ex_dt, id="id", cluster="cluster",
              knots=seq(0, max(ex_dt$time), length.out=6),
              tau_grid=seq(0, max(ex_dt$time), length.out=6),
              basis="bz", degree=1, link="log",
              w_recur=c(1,1), w_term=2, ipcw="km")
summary(fit)
```

WA_cv *K-fold cross-validation for WA configuration selection*

Description

Runs K-fold CV over a grid of basis types, degrees, interior-knot counts, and link functions. For each configuration, fits the model on K-1 folds and accumulates the prediction error (PE) on the held-out fold using `WA_PE()` (IPCW computed on the training subjects).

Usage

```
WA_cv(
  formula,
  data,
  id,
  cluster = NULL,
  basis_set = c("il", "pl", "bz"),
  degree_vec = 1:2,
  n_int_vec = c(0, 2, 4),
  knot_scheme = c("equidist", "quantile"),
  link_set = c("log"),
  time_range = NULL,
  tau_grid = NULL,
  w_recur,
  w_term,
  ipcw = c("cox", "km"),
  ipcw_formula = ~1,
  K = 5,
  seed = 1L,
  verbose = TRUE
)
```

Arguments

`formula`	A `Surv(time, status) ~ RHS` formula; see `WA_fit`.
`data`	Long-format data frame; see `WA_fit`.
`id`	Character scalar; subject ID column name; see `WA_fit`.
`cluster`	Optional character scalar; cluster column name; see `WA_fit`.
`basis_set`	Character vector of candidate bases.
`degree_vec`	Integer vector of candidate degrees.
`n_int_vec`	Integer vector of interior-knot counts; 0 means boundaries only.
`knot_scheme`	`"equidist"` or `"quantile"` to construct interior knots.
`link_set`	Character vector of candidate links (subset of `c("log","identity")`).
`time_range`	Optional numeric length-2 vector `c(tmin, tmax)`. If `NULL`, inferred from `data`.

tau_grid	Optional numeric vector; if NULL, a default dense grid over time_range is created.
w_recur	recurrent-event weights
w_term	Numeric scalar; terminal-event weight; see WA_fit.
ipcw	IPCW method ("cox" or "km") for PE computation.
ipcw_formula	One-sided RHS formula for IPCW Cox model (if ipcw="cox").
K	Number of folds.
seed	RNG seed for fold assignment.
verbose	Logical; show a text progress bar and per-fold messages.

Value

A data frame with columns: basis, degree, n_int, link, and aggregated PE. Lower PE is better.

WA_fit *While-Alive Regression (WA) for Composite Endpoints*

Description

Fits the while-alive regression model targeting the while-alive loss rate for composite endpoints with recurrent and terminal events. Time-varying covariate effects are represented via user-chosen time bases (e.g., B-spline, piecewise polynomial, interval-local). Robust inference supports cluster-randomized trials (CRTs) via cluster-robust variance; if cluster = NULL, IID (subject-as-cluster) variance is used.

Usage

```
WA_fit(
  formula,
  data,
  id,
  cluster = NULL,
  knots,
  tau_grid,
  basis = c("il", "pl", "bz", "ns", "ms", "st", "tl", "tf"),
  degree = 1,
  link = c("log", "identity"),
  w_recur,
  w_term,
  ipcw = c("km", "cox"),
  ipcw_formula = ~1
)
```

Arguments

formula	A Surv(time, status) ~ RHS formula. time and status must exist in data. The RHS contains baseline covariates (no explicit time-varying covariates here; time-variation is induced via the chosen basis).
data	Long-format data frame with one row per *event/checkpoint* per subject, containing time, status, id, optional cluster, and RHS covariates.
id	Character scalar; subject ID column name.
cluster	Optional character scalar; cluster column name for CRT-robust inference. If NULL, IID inference treats each subject as its own cluster.
knots	Numeric vector (length ≥ 2) specifying the basis boundaries and optional interior knots that define the time basis shape.
tau_grid	Numeric vector of evaluation times used to stack the estimating equations. Independent of knots.
basis	One of "il","pl","bz","ns","ms","st","tl","tf": interval-local ("il"), piecewise polynomial ("pl"), B-spline ("bz"), natural spline ("ns"), M-spline ("ms", requires **splines2**), step ("st"), truncated linear ("tl"), or time-fixed ("tf").
degree	Integer degree for bases that use it (e.g., "bz", "pl", "ns", "ms").
link	Link function: "log" (default) or "identity".
w_recur	Numeric vector of weights for each recurrent event type. Its length must match the number of recurrent status codes in data (i.e., excluding 0 for censoring and the max code for terminal).
w_term	Numeric scalar; weight for the terminal event.
ipcw	IPCW method: "km" or "cox".
ipcw_formula	A one-sided formula specifying RHS covariates for the IPCW Cox model when ipcw = "cox" (e.g., ~ x1 + x2). Ignored for ipcw = "km".

Details

The estimating equations solve $E[Z(t)\{L(t) - \mu_\beta(t)X_{\min}(t)\}V/G] = 0$ over tau_grid, where $L(t)$ is the weighted composite loss (recurrent+terminal), $\mu_\beta(t)$ the while-alive loss rate under the chosen link, $X_{\min}(t) = \min(T, t)$, V the at-risk/terminal indicator, and G the censoring survival modeled via ipcw.

Value

An object of class "WA" with elements:

- est: named coefficient vector.
- vcov: cluster-robust variance matrix.
- se: standard errors.
- converged: logical.
- basis, degree, link, Z_cols, knots, tau_grid, id_var, cluster_var, w_recur, w_term, status_codes, formula.

Examples

```
ex_dt <- crt_dt[crt_dt$cluster %in% c(1,2,3,4,7,10), ]
fit <- WA_fit(
  survival::Surv(time, status) ~ trt + Z1 + Z2,
  data     = ex_dt,
  id       = "id",
  cluster  = "cluster",
  knots    = seq(0, max(ex_dt$time, na.rm = TRUE), length.out = 6),
  tau_grid = seq(0, max(ex_dt$time, na.rm = TRUE), length.out = 6),
  basis    = "bz", degree = 1, link = "log",
  w_recur  = c(1, 1), w_term = 2,
  ipcw     = "km"
)
s <- summary(fit)
nd <- unique(ex_dt[, c("trt","Z1","Z2")])
plot(fit, newdata = nd,
     t_seq = seq(0, max(fit$tau_grid), length.out = 200),
     id = 1, mode = "wa", smooth = TRUE)
```

Index