

Package ‘coxmnar’

July 31, 2026

Title Cox Regression with Missing not at Random Failure Indicators

Version 0.1.0

Description Implements estimation for the Cox (1972, 1975) [<doi:10.1111/j.2517-6161.1972.tb00899.x>](https://doi.org/10.1111/j.2517-6161.1972.tb00899.x) [<doi:10.1093/biomet/62.2.269>](https://doi.org/10.1093/biomet/62.2.269) proportional hazards model when the failure indicator (cause of failure) is missing not at random (MNAR), following the two adjusted imputation-based estimating equations of Liu and Liu (2026) [<doi:10.1007/s11222-026-10857-1>](https://doi.org/10.1007/s11222-026-10857-1). Also provided for comparison are the full-data partial-likelihood estimator of Andersen and Gill (1982) [<doi:10.1214/aos/1176345976>](https://doi.org/10.1214/aos/1176345976), the complete-case estimator, and the missing-at-random imputation estimator of Liu and Wang (2010, Statistica Sinica, 20, 1125-1142). The probability models for the failure indicator and for the missingness mechanism are estimated jointly by maximum likelihood following Sun, Xie, and Liang (2013) [<doi:10.1007/s11425-012-4492-x>](https://doi.org/10.1007/s11425-012-4492-x), and a Nadaraya-Watson kernel-smoothed estimator of the missingness propensity is constructed following Qiu, Chen, and Zhou (2015) [<doi:10.1016/j.spl.2014.12.006>](https://doi.org/10.1016/j.spl.2014.12.006). Both an asymptotic (sandwich-type) variance estimator and a nonparametric bootstrap variance estimator are provided. When failure indicators are fully observed the estimators reduce algebraically to the classical Cox partial-likelihood estimator.

License GPL (>= 3)

Encoding UTF-8

Language en-US

Depends R (>= 4.1.0)

Imports survival (>= 3.5-0), stats, Rdpack

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

RoxygenNote 7.3.3

VignetteBuilder knitr

RdMacros Rdpack

Config/testthat/edition 3

NeedsCompilation no

Author Shikhar Tyagi [aut, cre] (ORCID:
 <<https://orcid.org/0000-0003-1606-0844>>),
 Arvind Pandey [aut],
 Bhupendra Singh [aut],
 Vrijesh Tripathi [aut]
Maintainer Shikhar Tyagi <shikhar1093tyagi@gmail.com>
Repository CRAN
Date/Publication 2026-07-31 15:10:02 UTC

Contents

coxmnar-package	2
coxmnar	3
coxmnar-methods	5
coxmnar_compare	6
coxmnar_control	7
simulate_coxmnar_data	8
Index	10

coxmnar-package	<i>coxmnar: Cox Regression with Missing Not at Random Failure Indicators</i>
-----------------	--

Description

Implements estimation for the Cox proportional hazards model when failure indicators are missing not at random (MNAR) following the methodology of Liu and Liu (2026).

Author(s)

Maintainer: Shikhar Tyagi <shikhar1093tyagi@gmail.com> (**ORCID**)
Authors:
 • Arvind Pandey <arvindmzu@gmail.com>
 • Bhupendra Singh <bhupendra.rana@gmail.com>
 • Vrijesh Tripathi <vrijesh.tripathi@uwi.edu>

References

Liu Y, Liu K (2026). “Estimation in the Cox proportional hazards model with missing not at random failure indicators.” *Statistics and Computing*, **36**, 112. doi:[10.1007/s11222026108571](https://doi.org/10.1007/s11222026108571).

Description

Fits the Cox proportional hazards model when failure indicators (causes of failure) are missing not at random (MNAR), using the adjusted imputation estimators of Liu and Liu (2026). Also supports complete-case, missing-at-random (MAR), and benchmark full-data estimation.

Usage

```
coxmnar(
  formula,
  time,
  cause = NULL,
  observed = NULL,
  data,
  method = c("ai", "ac", "mar", "cc", "full"),
  m_formula = NULL,
  pi_formula = NULL,
  kernel = c("gaussian", "triangular", "epanechnikov", "quartic", "triweight"),
  bandwidth = NULL,
  se_method = c("bootstrap", "asymptotic", "both", "none"),
  B = 500L,
  control = coxmnar_control(),
  ties = c("efron", "breslow"),
  x = FALSE,
  seed = NULL
)
```

Arguments

formula	Formula specifying covariates, e.g. <code>cause ~ Z1 + Z2</code> or <code>~ Z1 + Z2</code> .
time	Name of time column in data or numeric vector of observed times.
cause	Optional name of failure indicator column in data or numeric vector (1=event, 0=censored, NA=missing).
observed	Optional name of missingness indicator column <code>xi</code> (1=observed, 0=missing). Inferred from NAs in <code>cause</code> if NULL.
data	Data frame containing the variables.
method	Estimation method: "ai" (Adjusted-Imputation MNAR), "ac" (Adjusted-Complete MNAR), "mar" (Missing-at-Random), "cc" (Complete-Case), "full" (Full-Data benchmark). Default is "ai".
m_formula	One-sided formula for outcome model $m(w, \theta)$. Default is <code>~ time + all covariates</code> .
pi_formula	One-sided formula for propensity model $\pi(w, \alpha)$. Default is <code>~ time + all covariates</code> .

kernel	Kernel function for Nadaraya-Watson estimator: "gaussian", "triangular", "epanechnikov", "quartic", "triweight". Default is "gaussian".
bandwidth	Optional numeric scalar bandwidth for Nadaraya-Watson estimator. Calculated automatically if NULL.
se_method	Standard error estimation method: "bootstrap", "asymptotic", "both", "none". Default is "bootstrap".
B	Number of bootstrap replications. Default is 500L.
control	List of control parameters from <code>coxmnar_control</code> .
ties	Tie-handling method: "efron" or "breslow". Default is "efron".
x	Logical indicating whether to return the covariate matrix. Default is FALSE.
seed	Optional random seed for bootstrap reproducibility.

Value

An object of S3 class "coxmnar", containing estimated coefficients, variance-covariance matrix, standard errors, z-scores, p-values, fitted probability models, baseline hazard, and model diagnostics.

References

- Liu Y, Liu K (2026). "Estimation in the Cox proportional hazards model with missing not at random failure indicators." *Statistics and Computing*, **36**, 112. doi:10.1007/s11222026108571.
- Andersen PK, Gill RD (1982). "Cox's regression model for counting processes: a large sample study." *The Annals of Statistics*, **10**(4), 1100–1120. doi:10.1214/aos/1176345976.
- Liu C, Wang Q (2010). "Semiparametric estimation for regression coefficients in the Cox model with failure indicators missing at random." *Statistica Sinica*, **20**(3), 1125–1142.
- Sun Z, Xie T, Liang H (2013). "Statistical inference for right-censored data with nonignorable missing censoring indicators." *Science China Mathematics*, **56**(6), 1263–1278. doi:10.1007/s11425012-4492x.
- Qiu Z, Chen X, Zhou Y (2015). "A kernel-assisted imputation estimating method for the additive hazards model with missing censoring indicator." *Statistics & Probability Letters*, **98**, 89–97. doi:10.1016/j.spl.2014.12.006.

Examples

```
set.seed(123)
sim_data <- simulate_coxmnar_data(n = 100, beta0 = 0.2, mechanism = "mnar", seed = 123)
fit <- coxmnar(cause ~ Z, time = "time", data = sim_data, method = "ai", B = 10L, seed = 123)
print(fit)
summary(fit)
```

Description

Print, summary, prediction, confidence intervals, survival curves, and plotting for `coxmnar` objects.

Usage

```
## S3 method for class 'coxmnar'
print(x, ...)

## S3 method for class 'coxmnar'
summary(object, conf.level = 0.95, ...)

## S3 method for class 'summary.coxmnar'
print(x, ...)

## S3 method for class 'coxmnar'
coef(object, ...)

## S3 method for class 'coxmnar'
vcov(object, ...)

## S3 method for class 'coxmnar'
confint(object, parm, level = 0.95, ...)

## S3 method for class 'coxmnar'
predict(
  object,
  newdata = NULL,
  type = c("lp", "risk", "survival", "cumhaz"),
  times = NULL,
  ...
)

## S3 method for class 'coxmnar'
survfit(formula, newdata = NULL, ...)

## S3 method for class 'coxmnar'
plot(x, compare_km = TRUE, ...)

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

## S3 method for class 'summary.coxmnar_compare'
print(x, ...)
```

Arguments

<code>x</code>	An object of class <code>coxmnar</code> .
<code>...</code>	Additional arguments passed to methods.
<code>object</code>	An object of class <code>coxmnar</code> .
<code>conf.level</code>	Confidence level for summary. Default is 0.95.
<code>parm</code>	Specification of parameters to include in <code>confint</code> .
<code>level</code>	Confidence level for <code>confint</code> . Default is 0.95.
<code>newdata</code>	Optional data frame of new covariate values for <code>predict</code> or <code>survfit</code> .
<code>type</code>	Prediction type: <code>"lp"</code> (linear predictor), <code>"risk"</code> (relative risk $\exp(lp)$), <code>"survival"</code> , <code>"cumhaz"</code> . Default is <code>"lp"</code> .
<code>times</code>	Numeric vector of times at which to evaluate survival/cumulative hazard predictions.
<code>formula</code>	Object of class <code>coxmnar</code> when dispatched from <code>survfit</code> .
<code>compare_km</code>	Logical indicating whether to overlay Kaplan-Meier curve in plot. Default is <code>TRUE</code> .

Value

Various objects depending on the method called.

<code>coxmnar_compare</code>	<i>Compare Multiple Coxmnar Estimation Methods</i>
------------------------------	--

Description

Convenience wrapper to fit and compare multiple estimation methods on the same dataset.

Usage

```
coxmnar_compare(
  formula,
  time,
  data,
  methods = c("ai", "ac", "mar", "cc", "full"),
  ...
)
```

Arguments

<code>formula</code>	Formula specifying covariates.
<code>time</code>	Name of time column or numeric vector.
<code>data</code>	Data frame.
<code>methods</code>	Vector of method names to fit. Default is <code>c("ai", "ac", "mar", "cc", "full")</code> .
<code>...</code>	Additional arguments passed to <code>coxmnar</code> .

Value

An object of class "coxmnar_compare", containing a list of fits and a comparison table.

Examples

```
set.seed(123)
sim_data <- simulate_coxmnar_data(n = 100, beta0 = 0.2, mechanism = "mnar", seed = 123)
comp <- coxmnar_compare(cause ~ Z, time = "time", data = sim_data, B = 10L, seed = 123)
summary(comp)
```

coxmnar_control	<i>Control Parameters for coxmnar</i>
-----------------	---------------------------------------

Description

Specifies numerical optimization, tolerance, and control parameters for [coxmnar](#).

Usage

```
coxmnar_control(
  maxit = 200L,
  tol = 1e-08,
  eps = 1e-06,
  nn_fallback_k = NULL,
  boot_max_attempts_factor = 10L,
  optim_method = "L-BFGS-B",
  step_halving_max = 20L,
  on_failure = c("error", "warn"),
  verbose = FALSE
)
```

Arguments

maxit	Maximum number of iterations for optimization/Newton-Raphson. Default is 200L.
tol	Convergence tolerance for Newton-Raphson score norm. Default is 1e-8.
eps	Small positive probability clamp parameter. Default is 1e-6.
nn_fallback_k	Number of nearest neighbors for Nadaraya-Watson fallback. Default is 'max(10, ceiling(0.1*n))'.
boot_max_attempts_factor	Maximum multiplier for bootstrap replicate attempts. Default is 10L.
optim_method	Optimization method passed to optim . Default is "L-BFGS-B".
step_halving_max	Maximum number of step-halvings in Newton-Raphson solver. Default is 20L.
on_failure	Character string specifying action on convergence failure: "error" (default) or "warn".
verbose	Logical indicating whether to print diagnostic messages. Default is FALSE.

Value

A validated control list.

Examples

```
ctrl <- coxmnr_control(maxit = 100L, tol = 1e-6)
```

simulate_coxmnar_data *Simulate Survival Data with MNAR or MAR Failure Indicators*

Description

Generates synthetic survival data following the Cox proportional hazards model and missingness data-generating processes (DGPs) described in Sections 3.1-3.3 of Liu and Liu (2026).

Usage

```
simulate_coxmnar_data(
  n,
  beta0 = 0.2,
  mechanism = c("mnar", "mar"),
  covariate_dist = function(n) stats::runif(n, 0, 15),
  censoring_rate = 0.2,
  missing_rate = 0.2,
  conditionally_independent_censoring = FALSE,
  extended_missingness = FALSE,
  seed = NULL
)
```

Arguments

n	Sample size.
beta0	True regression coefficient scalar or vector. Default is 0.2.
mechanism	Missingness mechanism: "mnar" (Missing Not At Random) or "mar" (Missing At Random). Default is "mnar".
covariate_dist	Function generating covariate values for sample size 'n'. Default generates Uniform(0, 15).
censoring_rate	Target censoring rate in (0, 1). Default is 0.2.
missing_rate	Target missing indicator rate in (0, 1). Default is 0.2.
conditionally_independent_censoring	Logical indicating whether censoring depends on covariate Z. Default is FALSE.
extended_missingness	Logical indicating whether missingness depends on observed time X (Section 3.3). Default is FALSE.
seed	Optional random seed.

Value

A data frame with columns 'time', 'cause' (with 'NA' where 'xi=0'), 'Z' (or 'Z1'), and attributes recording ground truth parameters.

References

Liu Y, Liu K (2026). "Estimation in the Cox proportional hazards model with missing not at random failure indicators." *Statistics and Computing*, **36**, 112. doi:[10.1007/s11222026108571](https://doi.org/10.1007/s11222026108571).

Examples

```
set.seed(123)
sim_df <- simulate_coxmnar_data(n = 200, beta0 = 0.2, mechanism = "mnar", seed = 123)
head(sim_df)
```

Index

`coef.coxmnar` (`coxmnar-methods`), [5](#)
`confint.coxmnar` (`coxmnar-methods`), [5](#)
`coxmnar`, [3](#), [5–7](#)
`coxmnar-methods`, [5](#)
`coxmnar-package`, [2](#)
`coxmnar_compare`, [6](#)
`coxmnar_control`, [4](#), [7](#)

`optim`, [7](#)

`plot.coxmnar` (`coxmnar-methods`), [5](#)
`predict.coxmnar` (`coxmnar-methods`), [5](#)
`print.coxmnar` (`coxmnar-methods`), [5](#)
`print.summary.coxmnar`
 (`coxmnar-methods`), [5](#)
`print.summary.coxmnar_compare`
 (`coxmnar-methods`), [5](#)

`simulate_coxmnar_data`, [8](#)
`summary.coxmnar` (`coxmnar-methods`), [5](#)
`summary.coxmnar_compare`
 (`coxmnar-methods`), [5](#)
`survfit.coxmnar` (`coxmnar-methods`), [5](#)

`vcov.coxmnar` (`coxmnar-methods`), [5](#)