Package 'gcemod'

October 29, 2026

Type Package

Title Generalized Competing Event Models with Lunn-McNeil Testing

Version 0.3.1

Date 2026-10-28

Description Fits generalized competing event (GCE) models and estimates covariate effects on omega-plus, the ratio of the hazard for an event of interest to the hazard for a competing event, on both the cause-specific (Cox) and subdistribution (Fine-Gray) hazard scales. Confidence intervals and p-values are obtained from the Lunn-McNeil (1995) stacked (augmented) data approach. The package builds GCE risk scores from the model linear predictor, identifies risk-score cutpoints that maximize the separation in omega-plus between groups, and produces cumulative incidence (``alligator'') plots by risk group and calibration plots of predicted versus observed omega-plus. It also compares covariate effects across the primary, competing, and total (composite) events, and estimates covariate effects on the ratio of cumulative incidence functions (a cumulative-incidence-scale GCE model). Methods follow Carmona et al. (2014) <doi:10.1016/j.ijrobp.2014.03.047>, Mell et al. (2024) <doi:10.1016/j.eururo.2023.01.020>, and Lunn and McNeil (1995) <doi:10.2307/2532940>.

License GPL (>= 2)

Encoding UTF-8

Language en-US

LazyData true

Depends R (>= 3.5.0)

Imports survival, cmprsk, ggplot2 (>= 3.5.2), patchwork, stats

Suggests testthat (>= 3.0.0), knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

Config/roxygen2/version 8.0.0

NeedsCompilation no

Author Loren Mell [aut, cre],
Morten Overgaard [ctb] (proposed the stacked pseudo-observation estimator used by gcecif())

Maintainer Loren Mell <lmell@ucsd.edu>

Repository CRAN

Date/Publication 2026-10-29 20:40:27 UTC

Contents

gcecif . . . 2
gcecox . . . 4
gcefg . . . 5
gce_alligator . . . 7
gce_calibration . . . 8
gce_cutpoints . . . 9
gce_omegatime . . . 10
gce_performance . . . 12
gce_riskscore . . . 13
gce_scaling . . . 14
gce_tidy . . . 14
hn . . . 15
lunnmcneil . . . 16
prostate . . . 18

Index **19**

`gcecif` *Generalized competing event model on the cumulative-incidence-ratio scale*

Description

Fits a generalized competing event (GCE) model on the *cumulative incidence* scale, estimating covariate effects on the ratio of cumulative incidences $\rho(t) = F_1(t)/F_2(t)$ at a landmark time `t`, where F_1 and F_2 are the cumulative incidence functions of the event of interest and the competing event. $\rho(t)$ is the odds that a subject's *realized* event by time `t` is the event of interest rather than the competing event, so `gcecif()` is a cumulative-incidence-scale companion to the cause-specific (`gcecox`) and subdistribution (`gcefg`) omega+ models and returns a compatible `"gcemod"` object.

Effects are estimated by a *stacked pseudo-observation regression*: each subject contributes one jackknife pseudo-observation for $F_1(t)$ and one for $F_2(t)$ (both mean-type functionals of the Aalen-Johansen estimator, for which conditional unbiasedness of pseudo-observations is established), and a single log-link GEE is fit to the stacked pairs with an event-type indicator. The main effects model $\log F_2(t \mid z)$; the event-type interaction coefficients estimate $\log \rho(t \mid z) = \log F_1(t \mid z) - \log F_2(t \mid z)$ directly. Because each subject appears twice, a cluster-robust (subject) sandwich variance is used, which captures the dependence between a subject's two pseudo-observations induced by the shared all-cause survivor. This is the cumulative-incidence-scale analogue of the stacked Lunn-McNeil

construction used on the two hazard scales. A GCE risk score is formed from the interaction linear predictor, so `exp(riskscore)` is a subject's ρ relative to the average subject (see `gce_riskscore`, `gce_scaling`).

Usage

```
gcecif(Time, Ind, Cov, M = 5, t, conf.level = 0.95)
```

Arguments

`Time`	numeric vector of follow-up times.
`Ind`	a data frame of event indicators: column 1 = event of interest, column 2 = competing event (`0`/`1`).
`Cov`	a data frame of covariates (factors are expanded automatically).
`M`	number of bins for the ρ risk plot (default `5`).
`t`	landmark time at which ρ is evaluated.
`conf.level`	confidence level (default `0.95`).

Details

Because $F_k(t) = \int_0^t S(u^-)\, h_k(u)\, du$ with S the all-cause survivor, $\rho(t)$ is a survival-weighted time-average of the instantaneous cause-specific hazard ratio $\omega^+(u)$; under a time-constant ω^+ it equals ω^+ exactly. The ratio of cumulative incidences is a non-collapsible, marginal-over-time measure, so `exp(coef)` is interpreted as a conditional association with the realized event mix, not a mechanism-level per-cause effect.

The estimating function is *linear* in the two per-cause (mean-type) pseudo-observations, so its conditional unbiasedness follows from the standard pseudo-observation theory (Graw et al. 2009; Overgaard et al. 2017); pseudo-observations of the nonlinear ratio itself are not used. The working model comprises a log-linear model for $F_2(t \mid z)$ (the nuisance main effects, reported as `$coef_f2`) and a log-linear model for $\rho(t \mid z)$ (the target interactions, reported as `$coef`).

The estimand is defined for a genuine competing-risks setting, in which both $F_1(t)$ and $F_2(t)$ are appreciable at the landmark `t`. If the competing event is effectively absent, or `t` precedes appreciable competing-event incidence, $\log \rho$ is weakly identified and the fit may fail or return very large standard errors; this should be read as a signal that the competing-risks premise does not hold.

Value

An object of class `"gcemod"` (subclass `"gcecif"`) with: `coef` (log ρ ratios), `result` (ratio table with CIs and p-values), `omnibus` (Wald test that all ρ ratios equal 1), `riskscore` (interaction linear predictor, mean-centered), `scaling` (covariate means/SDs and per-SD effects), `F1`, `F2`, `rho0` (marginal $F_1(t)/F_2(t)$), `rho` (per-subject fitted ρ), `pseudo` (a two-column data frame of per-cause pseudo-observations), `coef_f2` (the log-linear $F_2(t \mid z)$ nuisance model), `converged`, and the design matrix, time, and cause coding used by `gce_tidy`.

References

Andersen PK, Klein JP, Rosthoj S (2003). Generalized linear models for correlated pseudo-observations, with applications to multi-state models. *Biometrika* 90:15-27.

Graw F, Gerds TA, Schumacher M (2009). On pseudo-values for regression analysis in competing risks models. *Lifetime Data Analysis* 15:241-255.

Overgaard M, Parner ET, Pedersen J (2017). Asymptotic theory of generalized estimating equations based on jack-knife pseudo-observations. *The Annals of Statistics* 45:1988-2015.

Overgaard M, Parner ET, Pedersen J (2018). Estimating the variance in a pseudo-observation scheme with competing risks. *Scandinavian Journal of Statistics* 45:923-940.

See Also

gcecox, gcefg, gce_tidy, gce_riskscore

Examples

```
data(prostate)
Ind <- data.frame(event = as.integer(prostate$status == 1),
                  competing = as.integer(prostate$status == 2))
Cov <- prostate[, c("age", "psa", "gleason", "t2b", "comorbidity")]
fit <- gcecif(prostate$time, Ind, Cov, t = 10)
summary(fit)
gce_tidy(fit)
```

---

gcecox *Generalized competing event model via cause-specific Cox regression*

Description

Fits a generalized competing event (GCE) model on the cause-specific hazard scale and estimates covariate effects on ω^+ (event of interest vs. competing event) using the Lunn-McNeil stacked model (see lunnmcneil), with 95% confidence intervals, Wald p-values, and an omnibus test. A GCE risk score is formed from the model linear predictor, and per-subject $\omega = \omega^+/(1+\omega^+)$ is returned for comparison with observed values.

Usage

```
gcecox(
  Time,
  Ind,
  Cov,
  M = 5,
  t,
  conf.level = 0.95,
  cluster.se = TRUE,
  standardize = FALSE
)
```

Arguments

`Time`	numeric vector of follow-up times.
`Ind`	a data frame of event indicators. Column 1 = event of interest, column 2 = competing event (`0`/`1`). A third column, if present, is ignored.
`Cov`	a data frame of covariates.
`M`	number of bins for the ω^+ risk plot (default `5`).
`t`	time point at which per-subject ω / ω^+ are evaluated.
`conf.level`	confidence level (default `0.95`).
`cluster.se`	logical; cluster-robust SEs in the Lunn-McNeil model (default `TRUE`).
`standardize`	logical; controls only the *reported* omega+ ratio scale (default `FALSE` = per natural covariate unit; `TRUE` = per 1 SD). The risk score is *always* the normalized (mean-centered) linear predictor regardless of this setting, so cutpoints, plots, and the predicted-vs-observed omega calibration are unaffected. Covariate means/SDs are available in `$scaling` (see `gce_scaling`).

Value

An object of class `"gcemod"` (subclass `"gcecox"`) with: `coef` (log ω^+ ratios), `result` (ratio table with CIs and p-values), `omnibus` (Lunn-McNeil global test), `riskscore` (linear predictor), `omega` / `omegaplus` (per-subject values at `t`), `omegaplot` and `omegatimeplot` (ggplot objects), and the fitted Lunn-McNeil model.

See Also

`gcefg`, `lunnmcneil`, `gce_cutpoints`

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
Cov <- hn[, c("age", "smoker", "t_cat", "n_cat", "p16")]
fit <- gcecox(hn$time, Ind, Cov, M = 5, t = 5)
summary(fit)
```

---

gcefg *Generalized competing event model via Fine-Gray regression*

Description

Fits a generalized competing event (GCE) model on the subdistribution hazard scale and estimates covariate effects on ω^+ using a Lunn-McNeil-style stacked Fine-Gray model (see `lunnmcneil` with `type = "finegray"`), giving 95% confidence intervals, Wald p-values, and an omnibus test from one joint model. A GCE risk score is formed from the model linear predictor, and per-subject ω^+ / ω at time `t` are returned for comparison with observed values.

Usage

```
gcefg(
  Time,
  Ind,
  Cov,
  M = 5,
  t,
  conf.level = 0.95,
  standardize = FALSE,
  se = c("robust", "boot"),
  B = 200
)
```

Arguments

`Time`	numeric vector of follow-up times.
`Ind`	a data frame of event indicators. Column 1 = event of interest, column 2 = competing event (`0`/`1`).
`Cov`	a data frame of covariates (factors are expanded to indicators).
`M`	number of bins for the ω^+ risk plot (default `5`).
`t`	time point at which per-subject ω^+ / ω are evaluated.
`conf.level`	confidence level (default `0.95`).
`standardize`	logical; controls only the *reported* omega+ ratio scale (default `FALSE` = per natural covariate unit; `TRUE` = per 1 SD). The risk score is always the normalized (mean-centered) linear predictor. Covariate means/SDs are in `$scaling` (see `gce_scaling`).
`se`	standard-error method passed to `lunnmcneil`: `"robust"` (default; cluster-robust on subject, censoring weights treated as fixed, which errs on the conservative side) or `"boot"` (subject bootstrap that re-estimates the censoring weights within each resample, so the variance additionally accounts for their estimation, at higher computational cost).
`B`	number of bootstrap resamples when `se = "boot"` (default `200`).

Value

An object of class `"gcemod"` (subclass `"gcefg"`) with the same components as `gcecox` on the sub-distribution scale.

See Also

`gcecox`, `lunnmcneil`

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
```

```
Cov <- hn[, c("age", "smoker", "t_cat", "n_cat", "p16")]
fit <- gcefg(hn$time, Ind, Cov, M = 5, t = 5)
summary(fit)
```

---

gce_alligator *Alligator plot: cumulative incidence by risk group and event type*

Description

Plots the cause-specific cumulative incidence functions (CIFs) of the event of interest and the competing event over time on a single panel, with risk groups distinguished by color and event types by line type (event of interest solid, competing event dashed). In higher-risk groups the event-of-interest CIF rises above the competing-event CIF and the curves separate like an alligator's open jaws.

The x-axis is truncated at the model's readout time `t`, with tick marks at 1-unit (e.g. 1-year) intervals. The y-axis auto-scales its cap to the smallest of 25/50/75/100% that covers the peak cumulative incidence (so low-incidence settings are not compressed), with ticks at 10% (or 5% when capped at 25%). Numbers at risk for each group are shown in an aligned panel below the plot. Gray's test across groups is attached.

Usage

```
gce_alligator(
  object,
  groups = 2,
  method = c("optimal", "quantile"),
  criterion = c("omegaplus", "cif"),
  xlab = "Time (years)",
  ylab = "Cumulative incidence",
  group.lab = "Risk group",
  risk.table = TRUE,
  ymax = NULL
)
```

Arguments

`object`	a `"gcemod"` object from `gcecox`, `gcefg`, or `gcecif`.
`groups`	a `gce_cutpoints` result, a grouping factor of length n, or a single integer giving the number of risk groups (default `2`). When an integer is given, the groups are derived with `gce_cutpoints` using `method`.
`method`	grouping rule used when `groups` is an integer: `"optimal"` (default) or `"quantile"` (equal-sized groups). Ignored when `groups` is a cutpoints object or a factor.
`criterion`	optimal-search criterion when `groups` is an integer and `method = "optimal"`: `"omegaplus"` (default, log-ω^+ separation; not tail-sensitive) or `"cif"` (joint Gray statistic on the observed cumulative incidence of both events). See `gce_cutpoints`.
`xlab, ylab`	axis labels.

group.lab	legend title for the risk groups (default "Risk group").
risk.table	logical; show numbers at risk in an aligned panel below the plot (default TRUE).
ymax	optional numeric upper cap for the y-axis (0-1). If NULL (default) the cap auto-scales to 25/50/75/100% based on the peak incidence.

Value

A ggplot/patchwork object. The Gray's-test result, the plotted CIF data, the numbers-at-risk table, and the cutpoints are attached as attributes "grays_test", "data", "n.risk", and "cutpoints".

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
fit <- gcecox(hn$time, Ind, hn[, c("age", "t_cat", "p16")], M = 5, t = 5)
gce_alligator(fit, groups = 2)
```

---

gce_calibration *Calibration of predicted vs. observed omega+ by risk rank*

Description

Groups subjects into equal-sized rank groups on the GCE risk score and compares the model-predicted quantity with an observed estimate in each group. For omega+ models (gcecox, gcefg) the predicted ω^+ (or ω) is compared with observed cause-specific Nelson-Aalen cumulative hazards at time t: $\omega^+_{obs} = H_1(t)/H_2(t)$ and $\omega_{obs} = H_1(t)/(H_1(t) + H_2(t))$. For a gcecif fit the predicted cumulative-incidence ratio ρ (or share π_1) is compared with the observed Aalen-Johansen cumulative incidence within the group: $\rho_{obs} = F_1(t)/F_2(t)$ and $\pi_{1,obs} = F_1(t)/(F_1(t)+F_2(t))$. A calibration plot (predicted vs. observed with the identity line) and the underlying table are returned.

Usage

```
gce_calibration(object, which = NULL, groups = 5)
```

Arguments

object	a "gcemod" object from gcecox, gcefg, or gcecif.
which	quantity to calibrate. For omega+ models "omegaplus" (default) or "omega"; for a gcecif fit "rho" (default) or "share". If NULL the model-appropriate default is used.
groups	number of rank groups (default 5).

Value

A ggplot object (predicted vs. observed). The calibration table is attached as attribute "data".

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
fit <- gcecox(hn$time, Ind, hn[, c("age", "t_cat", "p16")], M = 5, t = 5)
gce_calibration(fit, which = "omegaplus", groups = 5)
```

`gce_cutpoints` *Risk-score cutpoints for GCE risk stratification*

Description

Splits subjects into ordered risk groups on the GCE risk score. With `method = "optimal"` the cutpoint(s) maximize a separation criterion by a greedy forward search; with `method = "quantile"` equal-sized groups are used.

The default criterion, `"omegaplus"`, maximizes the between-group separation in **log** ω^+ (equivalently, the risk score / linear predictor). This is the additive scale on which the joint event-of-interest-vs-competing ω^+ contrast is defined, and – unlike separation measured on the exponential ω^+ scale – it is **not** sensitive to a few extreme high-risk subjects, so the search does not chase high-risk tails and tends to return balanced, well-separated groups. The alternative `"cif"` maximizes a joint Gray's test statistic (summed across both competing events) on the observed cumulative incidence.

Usage

```
gce_cutpoints(
  object,
  groups = 2,
  method = c("optimal", "quantile"),
  criterion = c("omegaplus", "cif"),
  min.prop = 0.1,
  nperm = 0,
  labels = NULL
)
```

Arguments

`object`	a `"gcemod"` object from `gcecox`, `gcefg`, or `gcecif`. For a `gcecif` fit the per-group summary reports mean ρ and share instead of ω^+ / ω; the `"omegaplus"` criterion still means separation on the risk-score (linear-predictor) scale.
`groups`	number of groups (default 2).
`method`	`"optimal"` (default) or `"quantile"`.
`criterion`	optimal-search criterion: `"omegaplus"` (default, log-ω^+ separation) or `"cif"` (joint Gray statistic on observed cumulative incidence of both events).
`min.prop`	minimum proportion of subjects per group (default `0.1`).

`nperm`	number of permutations for the selection-adjusted p-value (default `0` = not computed). When `nperm > 0`, the maximally-selected joint Gray statistic is permuted (risk score vs. outcome) to give a p-value accounting for the data-driven choice of cutpoint.
`labels`	optional character labels for the groups (low to high risk).

Value

A list with: `groups` (ordered factor), `cutpoints` (on the risk-score scale), `summary` (per-group n, events, mean ω^+ / ω), and `test`. The `test` element reports the Lunn-McNeil ω^+ contrast between groups – a test that the ratio of cause-specific (or subdistribution) hazards differs across groups – with the univariable ω^+ ratio estimate and CI (`$omegaplus`), the joint test (`$statistic`, `$df`, `$p.value`), the per-event Gray tests for comparison (`$individual_gray`), and a selection-adjusted `$p.perm` when `nperm > 0`.

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
fit <- gcecox(hn$time, Ind, hn[, c("age", "t_cat", "p16")], M = 5, t = 5)
gce_cutpoints(fit, groups = 2)

# with a selection-adjusted permutation p-value (slower)
gce_cutpoints(fit, groups = 2, nperm = 200)
```

`gce_omegatime` *GCE contrasts over time: a visual check of relative-hazard constancy*

Description

Plots a nonparametric estimate of a chosen GCE contrast over follow-up, as a visual assessment of the proportional relative-hazards (PRH) assumption and of the time range over which it holds approximately. All estimates are built from the Nelson-Aalen cause-specific cumulative hazards $\Lambda_k(t)$ and the Aalen-Johansen cumulative incidences $F_k(t)$:

`"omega"` **(default)** $\Lambda_1(t)/\{\Lambda_1(t)+\Lambda_2(t)\}$, the bounded cumulative hazard share, plotted on $[0,1]$.

`"omegaplus"` $\omega^+(t) = \Lambda_1(t)/\Lambda_2(t)$, the cause-specific relative hazard, on a log scale.

`"omegatilde"` $\tilde{\Lambda}_1(t)/\{\tilde{\Lambda}_1(t)+\tilde{\Lambda}_2(t)\}$, the subdistribution analogue of `"omega"`, with $\tilde{\Lambda}_k(t) = -\log\{1-F_k(t)\}$, on $[0,1]$.

`"omegatildeplus"` $\tilde{\omega}^+(t) = \tilde{\Lambda}_1(t)/\tilde{\Lambda}_2(t)$, the subdistribution relative hazard, on a log scale.

`"rho"` $\rho(t) = F_1(t)/F_2(t)$, the cumulative-incidence ratio, on a log scale.

`"pi"` $\pi(t) = F_1(t)/\{F_1(t)+F_2(t)\} = \rho(t)/\{1+\rho(t)\}$, the bounded share form of ρ (so $\operatorname{logit}\pi = \log\rho$), i.e. the probability that a realized event by t is the event of interest, on $[0,1]$.

If the cause-specific relative hazard is constant in time then $\Lambda_1(t) = \omega^+ \Lambda_2(t)$ and, because $F_k(t) = \int_0^t S(u^-)\lambda_k(u)\,du$, also $F_1(t) = \omega^+ F_2(t)$: the `"omega"`, `"omegaplus"`, `"rho"`, and `"pi"` displays are then exactly flat, and drift in any of them delineates where a single time-constant summary is and is not adequate. The subdistribution displays are flat only approximately (exactly so when both incidences are small, since $\tilde{\Lambda}_k \approx F_k$).

With `groups`, one curve is drawn per risk group. Approximately parallel group curves on a log-scaled contrast indicate a time-constant group *contrast*—the modeling assumption of the stacked estimators—even where the individual curves drift.

Usage

```
gce_omegatime(
  object,
  which = c("omega", "omegaplus", "omegatilde", "omegatildeplus", "rho", "pi"),
  groups = NULL,
  method = "optimal",
  xmax = NULL
)
```

Arguments

`object`	a `"gcemod"` object from `gcecox`, `gcefg`, or `gcecif`.
`which`	the contrast to plot: `"omega"` (default), `"omegaplus"`, `"omegatilde"`, `"omegatildeplus"`, `"rho"`, or `"pi"`. See Description.
`groups`	optional grouping: a `gce_cutpoints` result, a grouping factor of length n, or a single integer number of risk groups (derived with `gce_cutpoints`). `NULL` (default) draws one curve for the whole sample.
`method`	grouping rule used when `groups` is an integer: `"optimal"` (default) or `"quantile"`.
`xmax`	optional upper limit of the time axis; the curves are truncated at `xmax`. Defaults to `NULL` (full follow-up). Late follow-up is often carried by few subjects at risk, so truncating (e.g. at or somewhat beyond the readout time `t`) can give a fairer picture of the range that matters for the fitted summary.

Value

A `ggplot` object. The plotted step-function data (group, time, value) are attached as attribute `"data"`. The model's readout time `t` is marked with a dashed vertical line.

See Also

`gce_cutpoints`, `gce_alligator`

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
fit <- gcecox(hn$time, Ind, hn[, c("age", "t_cat", "p16")], M = 5, t = 5)
```

```
gce_omegatime(fit)
gce_omegatime(fit, which = "rho")
gce_omegatime(fit, which = "omegaplus", groups = 2)
```

`gce_performance` *Discrimination (C-index) for a GCE model*

Description

Computes the cause-specific concordance (C-index) of the GCE risk score for the event of interest. For calibration (predicted vs. observed ω^{+}), see `gce_calibration`.

Usage

```
gce_performance(object, conf.level = 0.95)
```

Arguments

`object`	a `"gcemod"` object.
`conf.level`	confidence level for the C-index interval (default `0.95`).

Details

The C-index is computed with `concordance` using the event of interest as the outcome and the GCE risk score as the predictor. Higher risk scores denote higher risk; the statistic is oriented so that values above 0.5 indicate discrimination in the expected direction, and the `"flipped"` attribute of the result records whether the orientation was reversed.

Value

An object of class `"gce_performance"`: a named numeric vector with elements `estimate`, `se`, `lower`, and `upper`, carrying attributes `conf.level` and `flipped`. Printing the object displays the C-index with its confidence interval.

See Also

`gce_calibration`

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
fit <- gcecox(hn$time, Ind, hn[, c("age", "t_cat", "p16")], M = 5, t = 5)
gce_performance(fit)
```

`gce_riskscore` *Normalized risk score and relative omega+ for new subjects*

Description

Computes the GCE risk score for new subjects using the covariate means/SDs and coefficients stored in a fitted model. The score is the normalized (mean-centered) linear predictor, so `exp(riskscore)` is the subject's omega+ relative to the *average* subject in the fitting data (a value of 1 = same as average, 2 = twice the average omega+). For cause-specific (`gcecox`) models the absolute predicted omega+ and omega at the model's time `t` are also returned.

Usage

```
gce_riskscore(object, newdata)
```

Arguments

`object`	a `"gcemod"` object from `gcecox`, `gcefg`, or `gcecif`.
`newdata`	a data frame containing the covariates used to fit `object` (same column names). Factor levels not seen in the fitting data are treated as the reference level.

Value

A data frame with `riskscore` and `exp(riskscore)` relative to the average subject: `omegaplus_rel` for omega+ models (with absolute `omegaplus` and `omega` at time `t` for cause-specific fits), or `rho_rel` for a `gcecif` fit.

See Also

`gce_scaling`

Examples

```
data(prostate)
Ind <- data.frame(as.integer(prostate$status == 1), as.integer(prostate$status == 2))
fit <- gcecox(prostate$time, Ind, prostate[, c("age", "psa", "gleason")], M = 5, t = 10)
gce_riskscore(fit, newdata = prostate[1:5, ])
```

gce_scaling *Covariate scaling (means and SDs) used by the GCE risk score*

Description

Returns each covariate's mean and standard deviation (from the data used to fit the model), together with the per-SD log-omega+ coefficient and omega+ ratio. These are the quantities needed to reproduce the normalized risk score for new subjects (see `gce_riskscore`) and to interpret how far a subject sits from the average patient.

Usage

```
gce_scaling(object)
```

Arguments

`object`	a "gcemod" object from `gcecox`, `gcefg`, or `gcecif`.

Value

A data frame with columns `term`, `mean`, `sd`, `coef_perSD` (log ratio per 1 SD) and the per-SD ratio (`omegaplus_ratio_perSD` for omega+ models, `rho_ratio_perSD` for a `gcecif` fit).

See Also

`gce_riskscore`

Examples

```
data(prostate)
Ind <- data.frame(as.integer(prostate$status == 1), as.integer(prostate$status == 2))
fit <- gcecox(prostate$time, Ind, prostate[, c("age", "psa", "gleason")], M = 5, t = 10)
gce_scaling(fit)
```

gce_tidy *Tidy a GCE model into a data frame of estimates*

Description

With `effects = FALSE` (the default) the omega+ ratio estimates are returned. With `effects = TRUE` a covariate-effects comparison table is returned that juxtaposes, for each covariate, the hazard ratio for the primary (event-of-interest) hazard, the competing-event hazard, the total (composite, any-event) hazard, and the relative hazard ratio (the omega+ ratio) from the GCE model, each with a confidence interval. This lets the investigator see whether a covariate acts mainly on the event of interest, the competing event, both, or on their balance.

Usage

```
gce_tidy(object, effects = FALSE, ...)
```

Arguments

object	a "gcemod" object from gcecox, gcefg, or gcecif. For a gcecif fit the relative column is the cumulative-incidence ratio (rho ratio) and the primary/competing/total columns are cause-specific hazard ratios.
effects	logical; if TRUE, return the primary/competing/total/ relative effects comparison table instead of the omega+ ratio table. Requires an object fit with **gcemod** >= 0.3.0 (which stores the design matrix). Default FALSE.
...	unused.

Details

The individual- and total-event columns are estimated on the same hazard scale as the fitted model: cause-specific Cox models for a gcecox fit, and Fine-Gray subdistribution models for a gcefg fit. The total (composite) event has no competing event, so its subdistribution and cause-specific hazards coincide and a Cox model on the composite endpoint is used in both cases.

Value

If effects = FALSE, a data frame with columns term, log_ratio, ratio, conf.low, conf.high, p.value. If effects = TRUE, a data frame with columns term, primary, competing, total, and relative, each a formatted "HR (low, high)" string.

Examples

```
data(hn)
Ind <- data.frame(event = as.integer(hn$status == 1),
                  competing = as.integer(hn$status == 2))
fit <- gcecox(hn$time, Ind, hn[, c("age", "t_cat", "p16")], M = 5, t = 5)
gce_tidy(fit)
gce_tidy(fit, effects = TRUE)
```

---

hn *Head-and-neck cancer competing-risks example dataset*

Description

An example head-and-neck cancer competing-risks dataset used to illustrate the **gcemod** functions. The data are provided for illustration only and do not represent real patients. The event of interest is cancer recurrence and the competing event is death without recurrence; recurrence is the more frequent event (recurrence-dominant, $\omega^{+} > 1$). Contrast with prostate.

Usage

```
hn
```

Format

A data frame with 1000 rows and 11 variables:

id sequential subject identifier

time follow-up time in years

status competing-risks status: 0 = censored, 1 = cancer recurrence (event of interest), 2 = death without recurrence (competing event)

age age at diagnosis (years)

ps ECOG performance status (factor: 0, 1, 2)

female sex indicator (1 = female)

smoker smoking indicator (1 = more than 10 pack-years)

t_cat T category (factor: 1-4)

n_cat N category (factor: 0-3)

p16 p16 status (1 = positive)

site primary site (factor: oropharynx, larynx, hypopharynx, oralcavity)

`lunnmcneil` *Lunn-McNeil stacked-model estimation of omega+ ratios*

Description

Estimates covariate effects on ω^+ (the ratio of the hazard for the event of interest to the hazard for the competing event) from a single stacked ("augmented") model, following Lunn and McNeil (1995). Each subject contributes one record per event type; the model is stratified by event type and the covariate-by-event-type interaction estimates $\log(\omega^+ \text{ ratio})$ with model-based standard errors, per-covariate Wald tests, and an omnibus multivariate Wald test.

Usage

```
lunnmcneil(
  Time,
  cause,
  Cov,
  event = 1,
  competing = 2,
  type = c("coxph", "finegray"),
  cluster.se = TRUE,
  conf.level = 0.95,
  standardize = FALSE,
  se = c("robust", "boot"),
  B = 200,
  ...
)
```

Arguments

Time	numeric vector of follow-up times.
cause	event indicator per subject: 0 = censored, and the codes in event and competing for the two event types.
Cov	a data frame of covariates (factors are expanded to indicators).
event	code marking the event of interest (default 1).
competing	code marking the competing event (default 2).
type	"coxph" (cause-specific, default) or "finegray" (subdistribution).
cluster.se	logical; cluster-robust (subject) SEs (default TRUE).
conf.level	confidence level (default 0.95).
standardize	logical; controls only the *reported* coefficient scale (default FALSE). FALSE reports omega+ ratios per natural covariate unit; TRUE reports them per 1 SD. This is a reparameterization, so Wald tests, p-values, and the omnibus test are identical either way. The covariate means/SDs ($center, $scale) are always returned.
se	standard-error method: "robust" (default; model-based, cluster-robust on subject, with the censoring weights treated as fixed) or "boot" (subject bootstrap: the stacked model is refit on B resamples of subjects, re-estimating the censoring weights within each resample, so the variance additionally accounts for their estimation). Mainly relevant for type = "finegray"; because estimating the weights reduces the asymptotic variance, the fixed-weights default errs, if at all, on the conservative side.
B	number of bootstrap resamples when se = "boot" (default 200).
...	passed to coxph.

Details

Two hazard scales are supported:

type = "coxph" Cause-specific hazards. The augmented data set duplicates each subject into an "event" stratum and a "competing" stratum; the interaction coefficient is exactly the difference in cause-specific log-hazards, i.e. $\log(\omega^+)$. This is the original Lunn-McNeil construction and the interaction MLE equals the difference of two separate cause-specific Cox fits.

type = "finegray" Subdistribution hazards. Risk-set-weighted data are generated for each event type with finegray, stacked, and fit with a single weighted, event-type-stratified coxph. The interaction estimates the difference in subdistribution log-hazards. Because subjects recur across the weighted risk sets, cluster-robust (subject) standard errors are used. This is an extension of Lunn-McNeil to the subdistribution scale.

Subject records are duplicated in both cases, so cluster-robust standard errors (clustered on subject) are the default.

Value

An object of class "gce_lunnmcneil": a list with the fitted model ($fit), the omega+ coefficient table ($omegaplus), the reference (competing-event) coefficients ($beta_competing), the omnibus test ($omnibus), the design column names ($terms), and the hazard scale ($type).

References

Lunn M, McNeil D (1995) Applying Cox regression to competing risks. *Biometrics* 51:524-32.

Examples

```
data(hn)
Cov <- hn[, c("age", "smoker", "t_cat", "n_cat", "p16")]
lunnmcneil(hn$time, hn$status, Cov, type = "coxph")
lunnmcneil(hn$time, hn$status, Cov, type = "finegray")
```

`prostate` *Prostate cancer competing-risks example dataset*

Description

An example prostate cancer competing-risks dataset used to illustrate the **gcemod** functions. The data are provided for illustration only and do not represent real patients. The event of interest is distant metastasis or prostate-cancer death and the competing event is death from other causes; the competing event is the more frequent (competing-death-dominant, $\omega^{+} < 1$). Contrast with `hn`.

Usage

```
prostate
```

Format

A data frame with 1000 rows and 9 variables:

id sequential subject identifier

time follow-up time in years

status competing-risks status: 0 = censored, 1 = distant metastasis or prostate-cancer death (event of interest), 2 = death from other causes (competing event)

age age at diagnosis (years)

ps performance status (factor: 0, 1)

psa pre-treatment PSA (ng/mL, capped at 20)

gleason Gleason grade group (factor: "<=6", "3+4", "4+3", ">=8")

t2b stage indicator (1 = T2b or higher)

comorbidity significant comorbidity indicator (1 = yes)

Index

∗ **datasets**
hn, 15
prostate, 18

concordance, *12*
coxph, *17*

finegray, *17*

gce_alligator, 7, *11*
gce_calibration, 8, *12*
gce_cutpoints, *5*, *7*, 9, *11*
gce_omegatime, 10
gce_performance, 12
gce_riskscore, *3, 4*, 13, *14*
gce_scaling, *3*, *5, 6*, *13*, 14
gce_tidy, *3, 4*, 14
gcecif, 2, *7–9*, *11*, *13–15*
gcecox, *2*, *4*, 4, *6–9*, *11*, *13–15*
gcefg, *2*, *4, 5*, 5, *7–9*, *11*, *13–15*

hn, 15, *18*

lunnmcneil, *4–6*, 16

prostate, *15*, 18