Package 'ModalCens'

March 11, 2026

Type Package

Title Parametric Modal Regression with Right Censoring

Version 0.1.0

Description Implements parametric modal regression for continuous positive
distributions of the exponential family under right censoring.
Provides functions to link the conditional mode to a linear predictor
using reparameterizations for Gamma, Beta, Weibull, and Inverse Gaussian
families. Includes maximum likelihood estimation via numerical
optimization, asymptotic inference based on the observed Fisher
information matrix, and model diagnostics using randomized quantile
residuals.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports stats, graphics

URL https://github.com/chedgala/ModalCens

BugReports https://github.com/chedgala/ModalCens/issues

NeedsCompilation no

Author Christian Galarza [aut, cre],
Victor Lachos [aut]

Maintainer Christian Galarza <chedgala@espol.edu.ec>

Repository CRAN

Date/Publication 2026-03-11 16:40:22 UTC

Contents

---

modal_cens | *Modal Regression Fit with Right Censoring*

Description

Fits a parametric modal regression model under right censoring by linking the conditional mode M_i to a linear predictor via a family-specific link function $g(M_i) = \mathbf{x}_i^\top \boldsymbol{\gamma}$.

The density of each family is reparameterized so that M_i appears explicitly, obtained by solving $\partial \log f(y_i)/\partial y_i|_{y_i=M_i} = 0$. The resulting mode-parameter mappings are:

Family	Link	Mode mapping
Gamma	log	$\alpha = \phi^{-1} + 1, \backslash \, \text{scale} = M_i \cdot \phi$
Beta	logit	$\alpha = \phi + 1.01, \backslash \, \beta_p = (\alpha - 1)/M_i - \alpha + 2$
Weibull	log	$k = \phi + 1.01, \backslash \, \lambda = M_i \, (k/(k-1))^{1/k}$
Inv. Gauss	log	$\mu = [(M_i^{-2}) - 3/(\lambda M_i)]^{-1/2}, \backslash \, \lambda > 3M_i$

The censored log-likelihood is

$$\ell(\boldsymbol{\gamma}, \phi) = \sum_{i=1}^{n} \big[(1 - \delta_i) \log f(y_i) + \delta_i \log S(y_i)\big],$$

where $\delta_i = 1$ for right-censored observations and $S(\cdot) = 1 - F(\cdot)$ is the survival function. Maximization is performed via the BFGS algorithm with analytical Hessian for asymptotic standard errors.

Usage

```
modal_cens(formula, data, cens, family = "gamma")
```

Arguments

formula	An object of class "formula" (e.g., y ~ x1 + x2).
data	A data frame containing the variables in the model. Must not contain missing values (NA) in any variable used by formula. No imputation is performed; the function stops with an error if NAs are detected.
cens	A binary numeric vector indicating censoring status: 1 = right-censored (only a lower bound y_i is observed), 0 = fully observed. Must not contain NAs and must have the same length as nrow(data).
family	A character string naming the distribution family: "gamma", "beta", "weibull", or "invgauss". Default is "gamma".

Value

An object of class ″ModalCens″ containing:

coefficients Estimated regression coefficients (beta vector).

phi Estimated dispersion/shape parameter (on original scale).

vcov Full covariance matrix (p+1 x p+1), including log_phi.

vcov_beta Covariance matrix for regression coefficients only (p x p).

fitted.values Estimated conditional modes.

residuals Randomized quantile residuals (Dunn-Smyth).

loglik Maximized log-likelihood value.

n Number of observations used in fitting.

n_par Total number of estimated parameters.

family Distribution family used.

cens Censoring indicator vector as supplied.

call The matched call.

terms The model terms object.

y Response variable.

Available methods

Objects of class ″ModalCens″ support the following S3 methods:

summary() Coefficient table with standard errors, z-values, and p-values; dispersion parameter; log-likelihood; AIC; BIC; and pseudo R-squared.

plot() Two-panel diagnostic plot: (1) residuals vs.\ fitted modes, distinguishing observed and censored points; (2) normal Q-Q plot of randomized quantile residuals.

coef() Estimated regression coefficients $\hat{\gamma}$.

vcov() Full $(p+1) \times (p+1)$ covariance matrix including log_phi. Use object$vcov_beta for the $p \times p$ submatrix of regression coefficients only.

residuals() Randomized quantile residuals (Dunn & Smyth, 1996), which follow $\mathcal{N}(0,1)$ under the correct model, even under censoring.

logLik() Maximized log-likelihood value.

AIC() / BIC() Information criteria computed as $-2\ell + k \cdot p$ with $k = 2$ or $k = \log n$.

Author(s)

Christian Galarza and Victor Lachos

Examples

```
data(trees, package = "datasets")
set.seed(007)

# Simulate 15% right-censoring (for didactic purposes)
q85      <- quantile(trees$Volume, 0.85)
cens_ind <- as.integer(trees$Volume > q85)  # 1 = censored, 0 = observed

# (0,1) rescaling for Beta
ys      <- (trees$Volume - min(trees$Volume) + 1) / (max(trees$Volume) - min(trees$Volume) + 2)
q85b     <- quantile(ys, 0.85)
cens_b   <- as.integer(ys > q85b)

# Build datasets
df_orig <- data.frame(y = pmin(trees$Volume, q85), cens = cens_ind,
                      Girth = trees$Girth, Height = trees$Height)
df_beta <- data.frame(y = pmin(ys, q85b), cens = cens_b,
                      Girth = trees$Girth, Height = trees$Height)

# Fit all families
datasets <- list(gamma = df_orig, weibull = df_orig,
                 invgauss = df_orig, beta = df_beta)
models <- list(); aic_values <- list()

for (f in names(datasets)) {
  mod <- try(modal_cens(y ~ Girth + Height, data = datasets[[f]],
                        cens = datasets[[f]]$cens, family = f), silent = TRUE)
  if (!inherits(mod, "try-error")) { models[[f]] <- mod; aic_values[[f]] <- AIC(mod) }
}

# Select best model by AIC and analyse
best <- names(which.min(aic_values))
cat("Best family:", best, "| AIC =", round(aic_values[[best]], 3), "\n")

fit <- models[[best]]
summary(fit)
plot(fit)
confint(fit)
```

Index