Package ‘tcensReg’

July 1, 2020

Title MLE of a Truncated Normal Distribution with Censored Data

Version 0.1.7

Description Maximum likelihood estimation (MLE) of parameters assuming an underlying left truncated normal distribution with left censoring described in Williams, J, Kim, H, and Crespi, C. (2020) <doi:10.1186/s12874-020-01032-9>. Censoring is assumed to occur above the truncation threshold meaning that only censored observations are observed. Additional maximum likelihood estimation procedures are implemented to solve left censored only and left truncated only problems.

Depends R (>= 3.3)

License MIT + file LICENSE

Encoding UTF-8

LazyData false

RoxygenNote 7.1.0

Imports stats, maxLik, Rdpack

Suggests knitr, rmarkdown, testthat (>= 2.1.0), ggplot2, viridis, future.apply, tictoc, censReg, truncreg, microbenchmark

RdMacros Rdpack

VignetteBuilder knitr

BugReports `https://github.com/williazo/tcensReg/issues`

URL `https://github.com/williazo/tcensReg`

NeedsCompilation no

Author Justin Williams [aut, cre]

Maintainer Justin Williams <`williazo@ucla.edu`>

Repository CRAN

Date/Publication 2020-07-01 17:40:03 UTC

R topics documented:

`pseudo_r2` *Pseudo R2 for tcensReg Objects*

Description

Implementation of various methods for calculating pseudo R2 values popular with censored observations

Usage

```
pseudo_r2(obj, type = c("mckelvey_zavoina"))
```

Arguments

`obj` Object of class `tcensReg`

`type` Character value indicating the type of pseudo R2 to calculate. Currently only mckelvey_zavoina is available

Details

When comparing goodness of fit between methods for censored observations pseudo R^2 is often the preferred metric (Veall and Zimmermann 1996). While there are many different types of pseudo R^2 measures available this function implements those that are particularly relevant for censored observations. Below is a description. of the currently available methods within this function.

McKelvey-Zavoina:

This measure of pseudo R^2 is from McKelvey and Zavoina (1975) and is the optimal metric suggested for limited dependent variables from Veall and Zimmermann (1996). The formula is shown below

$$R^2_{mz} = \frac{\sum_{i=1}^{N}(\hat{y}_i - \bar{\hat{y}}_i)^2}{\sum_{i=1}^{N}(\hat{y}_i - \bar{\hat{y}}_i)^2 + N\hat{\sigma}}$$

Value

List with numeric value representing the pseudo R^2 and type of pseudo R^2 calculated

References

McKelvey RD, Zavoina W (1975). "A statistical model for the analysis of ordinal level dependent variables." *Journal of mathematical sociology*, **4**(1), 103-120.

Veall MR, Zimmermann KF (1996). "Pseudo-R2 measures for some common limited dependent variable models." *Journal of Economic Surveys*, **10**(3), 241-259.

Examples

```
#truncated normal underlying data
y_star <- rtnorm(n = 1000, mu = 0.5, sd = 1, a = 0)

#apply censoring
y <- ifelse(y_star <= 0.25, 0.25, y_star)

#find MLE estimates
mod_result <- tcensReg(y ~ 1, v = 0.25, a = 0)

pseudo_r2(mod_result, type="m")
```

rtnorm *Simulate Random Left-Truncated Normal Distribution*

Description

This function is used to generate random samples from left-truncated normal distribution with specified mean and variance.

Usage

```
rtnorm(n, mu, sd, a)
```

Arguments

n	Numeric scalar representing the number of observations. Must be greater than or equal to 1.
mu	Mean value of the underlying normal random variable
sd	Standard deviation of underlying normal random variable
a	Numeric vector indicating the left-truncation value.

Details

Our goal is to draw samples from the left truncated normal random variable Y_i^*. We define this distribution as

$$Y_i^* \sim TN(\mu, \sigma^2, a)$$

Sampling is performed by first drawing from a random variable Z with a uniform distribution on the interval $[0, 1]$ to generate cumulative density probabilities, p. Then the inverse density function of the truncated normal random variable is applied to generate our desired truncated normal observations.

This inverse truncated normal function is shown below:

$$Y_i^* = \Phi^{-1}\left\{p \times \left[1 - \Phi(\frac{a-\mu}{\sigma})\right] + \Phi(\frac{a-\mu}{\sigma})\right\} \times \sigma + \mu,$$

where p represents the probabilities sampled from the uniform distribution.

Notes:

- If the mean, `mu`, is specified as a vector then the standard deviation, `sigma`, must have either:
  1. same length as `mu`
  2. be a scalar, indicating that all samples have constant standard deviation

Value

Returns a vector of samples drawn from the left truncated normal distribution equal to length n.

Examples

```
#zero truncated normal data with mean 0.5 and standard deviation 1
y_star <- rtnorm(n = 100, mu = 0.5, sd = 1, a = 0)
```

`tcensReg` *Regression Method for Truncated Normal Distribution with Censored Data*

Description

This function is used to find estimates from a linear equation assuming that the underlying distribution is truncated normal and the data has subsequently been censored data.

Usage

```
tcensReg(
  formula,
  a = -Inf,
  v = NULL,
  data = sys.frame(sys.parent()),
  method = c("CG", "Newton", "BFGS"),
  ...
)
```

Arguments

`formula` Object of class `formula` which symbolically describes the model to be fit

`a` Numeric scalar indicating the truncation value. Initial value is -Inf indicating no truncation

`v` Numeric scalar indicating the censoring value. Initially set to NULL indicating no censoring

`data` Data.frame that contains the outcome and corresponding covariates. If none is provided then assumes objects are in user's environment.

`method` Character value indicating which optimization routine to perform. Choices include `Newton`, `BFGS`, and `CG`. See details for explanation on each method.

`...` Additional arguments from such as `max_iter`, `step_max`, or `epsilon`. See details for how to define these additional arguments.

Details

This estimation procedure returns maximum likelihood estimates under the presence of *left* truncation and/or *left* censoring. It builds upon currently available methods for limited dependent variables by relaxing the assumption of a latent normal distribution and instead allows the this underlying distribution to potentially be a latent **truncated** normal distribution.

To indicate left censoring the user should specify the parameter `v`, and to indicate left truncation specify the parameter `a`. If specifying both left censoring and left truncation note that there is an implicit restriction that $a < \nu$.

Below is a brief description of the types of distributions that can be fit along with the assumed data generating process for the observed outcome, Y.

Latent Distributions:

The tcensReg function allows user to specify one of four combinations of distributional assumptions with or without censoring. These are listed below along with the necessary arguments needed to fit this model.

Truncated Normal with Censoring:
This is the main model that this package is designed to fit and introduced in (Williams et al. 2019).It assumes

$$Y_i^* \sim TN(\mu, \sigma^2, a)$$

where TN indicates a left truncated normal random variable, truncated at the value a.
This underlying truncated normal random variable is then *left* censored at the value ν to create the censored observations Y such that

$$Y_i = \nu 1_{\{Y_i^* \leq \nu\}} + Y_i^* 1_{\{Y_i^* > \nu\}}$$

Required Arguments:

- `a`: left truncation value
- `v`: left censoring value

Normal with Censoring:

This model is commonly referred to as the Tobit model (Tobin 1958). This model assumes that the data is generated from a latent normal random variable Y_i^*, i.e.,

$$Y_i^* \sim N(\mu, \sigma^2)$$

This underlying normal random variable is then *left* censored at the value ν to create the censored observations Y such that

$$Y_i = \nu 1_{\{Y_i^* \leq \nu\}} + Y_i^* 1_{\{Y_i^* > \nu\}}$$

Required Arguments:

- `v`: left censoring value

This procedure can also be fit using the `censReg` package by Henningsen (2010).

Truncated Normal:

This model assumes that there is no censored observations, but that the data are left truncated as originally described by Hald (1949).

Therefore, we assume that the observed values follow

$$Y_i \tilde{} TN(\mu, \sigma^2, a)$$

where TN indicates a left truncated normal random variable, truncated at the value a.

Required Arguments:

- `a`: left truncation value

This procedure can also be fit using the `truncreg` package by Croissant and Zeileis (2018).

Normal:

This model assumes that there is no left censoring and no left truncation.

Maximum likelihood estimates are returned based on the assumption that the random variable follows the distribution

$$Y_i \tilde{} N(\mu, \sigma^2)$$

Required Arguments: None

This procedure can also be fit using the command `lm` in base R.

Optimization Methods:

Currently available optimization routines include conjugate gradient (`CG`), Newton-Raphson (`Newton`), and BFGS (`BFGS`). The default method is set as the conjugate gradient. Both the of the conjugate gradient and BFGS methods are implemented via the general-purpose optimization routine `optim`. These two methods use only the respective likelihood and gradient functions. The Newton-Raphson method uses the likelihood, gradient, and Hessian functions along with line search to achieve the maximum likelihood.

Additional Arguments:

There are additional arguments that the user may provide for controlling the optimization routine.

- max_iter: Maximum number of iterations for optimization routine. Default is 100
- step_max: Maximum number of steps when performing line search. Default is 10
- epsilon: Numeric value used to define algorithm stops, i.e., when evaluated gradient is less than epsilon. Default is 0.001
- tol_val: Tolerance value used to stop the algorithm if the (n+1) and (n) log likelihood is within the tolerance limit.
- theta_init: Numeric vector specifying the initial values to use for the estimated parameters β and $\log(\sigma)$

Value

Returns a list of final estimate of theta, total number of iterations performed, initial log-likelihood, final log-likelihood, estimated variance covariance matrix, information criterion, model design matrix, call, list of total observations and censored observations, and latent distributional assumption.

References

Croissant Y, Zeileis A (2018). *truncreg: Truncated Gaussian Regression Models*. R package version 0.2-5, https://CRAN.R-project.org/package=truncreg.

Hald A (1949). "Maximum likelihood estimation of the parameters of a normal distribution which is truncated at a known point." *Skandinavisk Aktuarietidskrift*, **32**, 119-134.

Henningsen A (2010). "Estimating Censored Regression Models in R using the censReg Package." *R package vignettes*.

Tobin J (1958). "Estimation of relationships for limited dependent variables." *Econometrica*, **26**(1), 24-36.

Williams JR, Kim H, Crespi CM (2019). "Modeling Variables with a Detection Limit using a Truncated Normal Distribution with Censoring." *arXiv preprint arXiv:1911.11221*.

Examples

```
#truncated normal underlying data
y_star <- rtnorm(n = 1000, mu = 0.5, sd = 1, a = 0)

#apply censoring
y <- ifelse(y_star <= 0.25, 0.25, y_star)

#find MLE estimates
trunc_cens_mod <- tcensReg(y ~ 1, v = 0.25, a = 0)
summary(trunc_cens_mod)
```

tcensReg_sepvar *Regression Method for Truncated Normal Distribution with Censoring for Independent Truncated Normal Groups*

Description

This function is used to find estimates from a linear equation assuming that the data is observed from a truncated distribution with left censoring. It uses numerical values of the gradient vector and hessian matrix to solve for the maximum likelihood using maxLik package. This function can also be used with censored only, truncated only, or uncensored and untruncated gaussian models.

Usage

```
tcensReg_sepvar(
  formula,
  a = -Inf,
  v = NULL,
  group_var,
  method = c("BFGS", "maxLik", "CG"),
  theta_init = NULL,
  data = sys.frame(sys.parent()),
  max_iter = 100,
  ...
)
```

Arguments

formula	Object of class `formula` which symbolically describes the model to be fit
a	Numeric scalar indicating the truncation value. Initial value is -Inf indicating no truncation
v	Numeric scalar indicating the censoring value. Initially set to NULL indicating no censoring
group_var	Character scalar indicating a variable in the data.frame that defines the independent groups
method	Character value indicating which optimization routine to perform. Choices include `BFGS`, `maxLik` and `CG`. See details for explanation on each method.
theta_init	Optional initial values for the parameters. Default is to fit a linear regression model.
data	Data.frame that contains the outcome and corresponding covariates. If none is provided then assumes objects are in user's environment.
max_iter	Numeric value indicating the maximum number of iterations to perform.
...	Additional arguments such as `max_iter`, `step_max`, or `epsilon`.

Details

Currently available optimization routines include conjugate gradient (`CG`), Newton-Raphson type via maxLik package (`maxLik`), and BFGS (`BFGS`). The default method is set as the conjugate gradient. Both the of the conjugate gradient and BFGS methods are implemented via the general-purpose optimization `optim`. These two methods use only the respective likelihood and gradient functions. The Newton-Raphson method uses the likelihood, gradient, and Hessian functions along with line search to achieve the maximum likelihood.

Value

Returns a list of final estimate of theta, total number of iterations performed, initial log-likelihood, final log-likelihood, and estimated variance covariance matrix.

Index