Package 'tcensReg'

January 8, 2020

Title MLE of a Truncated Normal Distribution with Censored Data

Version 0.1.5

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. (2019) <arXiv:1911.11221>. 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 6.1.1

Imports stats, maxLik

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

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 2020-01-08 20:50:03 UTC

R topics documented:

rtnorm 2
tcensReg 3

tcensReg_gradient 4
tcensReg_gradient_sepvar 5
tcensReg_gradient_sepvar_maxLik 5
tcensReg_hess 6
tcensReg_llike 7
tcensReg_llike_sepvar 7
tcensReg_llike_sepvar_maxLik 8
tcensReg_newton 9
tcensReg_optim 10
tcensReg_sepvar 11

Index **13**

`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. Sampling is performed by first drawing from a random uniform distribution to generate c.d.f. probabilities and then the inverse density function is applied to generate observations.

Usage

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

Arguments

`n`	Numeric scaler 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

Note that if the mean `mu` is specified as a vector then the standard deviation `sigma` must have the same length or be a scalar indicating that all samples have constant standard deviation.

Value

Returns a vector of samples drawn from the specified distribition eqaul 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. It uses analytically derived values of the gradient vector and Hessian matrix to iteratively solve for the maximum likelihood using Newton-Raphson methods with step halving line search. This function can also be used with censored only (similar to censReg), truncated only (similar to truncreg), or uncensored and untruncated gaussian models.

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 tcensReg_newton such as max_iter, step_max, or epsilon.

Details

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 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.

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
tcensReg(y ~ 1, v = 0.25, a = 0)
```

`tcensReg_gradient` *Gradient Vector for Truncated Normal Distribution with Censoring with Linear Equation Mean*

Description

Gradient Vector for Truncated Normal Distribution with Censoring with Linear Equation Mean

Usage

```
tcensReg_gradient(theta, y, X, a = -Inf, v = NULL)
```

Arguments

`theta`	Numeric vector numeric vector containing estimates of beta and log sigma
`y`	Numeric vector with the observed truncated and censored outcomes
`X`	Numeric design matrix
`a`	Numeric scalar indicating the truncation value
`v`	Numeric scalar indicating the censoring value

Value

Vector of gradient values with p-1 beta parameters and log sigma for the nth iterate

---

`tcensReg_gradient_sepvar`

Gradient Vector for Truncated Normal Distribution with Censoring with Linear Equation Mean for J Independent Truncated Normals with Seprate Variance

---

Description

Gradient Vector for Truncated Normal Distribution with Censoring with Linear Equation Mean for J Independent Truncated Normals with Seprate Variance

Usage

```
tcensReg_gradient_sepvar(theta, y, X, group, a = -Inf, v = NULL)
```

Arguments

`theta`	Numeric vector numeric vector containing estimates of beta and log sigma
`y`	Numeric vector with the observed truncated and censored outcomes
`X`	Numeric design matrix
`group`	Character vector identifying the group membership for the independent truncated normal variables. This defines the `J` groups.
`a`	Numeric scalar indicating the truncation value
`v`	Numeric scalar indicating the censoring value

Value

Vector of gradient values with p-1 beta parameters and log sigma for the nth iterate

---

`tcensReg_gradient_sepvar_maxLik`

Gradient Vector for Truncated Normal Distribution with Censoring with Linear Equation Mean for J Independent Truncated Normals with Seprate Variance

---

Description

Gradient Vector for Truncated Normal Distribution with Censoring with Linear Equation Mean for J Independent Truncated Normals with Seprate Variance

Usage

```
tcensReg_gradient_sepvar_maxLik(theta, y, X, group, left_trunc = -Inf,
  v = NULL)
```

Arguments

theta	Numeric vector numeric vector containing estimates of beta and log sigma
y	Numeric vector with the observed truncated and censored outcomes
X	Numeric design matrix
group	Character vector identifying the group membership for the independent truncated normal variables. This defines the J groups.
left_trunc	Numeric scalar indicating the truncation value
v	Numeric scalar indicating the censoring value

Value

Vector of gradient values with p-1 beta parameters and log sigma for the nth iterate

---

tcensReg_hess *Hessian Matrix for Truncated Normal Distribution with Censoring with Linear Equation Mean*

Description

Hessian Matrix for Truncated Normal Distribution with Censoring with Linear Equation Mean

Usage

```
tcensReg_hess(theta, y, X, a = -Inf, v = NULL)
```

Arguments

theta	Numeric vector numeric vector containing estimates of beta and log sigma
y	Numeric vector with the observed truncated and censored outcomes
X	Numeric design matrix
a	Numeric scalar indicating the truncation value
v	Numeric scalar indicating the censoring value

Value

Matrix of Hessian values for the nth iterate

tcensReg_llike *Log Likelihood for Truncated Normal Distribution with Censoring with Linear Equation Mean*

Description

tcensReg_llike is a supporting function that is used to calculate the log likelihood value at the nth iteration of theta. If a and/or v are not specified then the corresponding censored only, truncated only, or gaussian log likelihood will be used. This function is called as part of the Newton-Raphson algorithm in tcensReg_newton.

Usage

```
tcensReg_llike(theta, y, X, a = -Inf, v = NULL)
```

Arguments

theta	Numeric vector containing estimates of beta and log sigma
y	Numeric vector with the observed truncated and censored outcomes
X	Numeric design matrix
a	Numeric scalar indicating the truncation value
v	Numeric scalar indicating the censoring value

Value

Scalar value of the log-likelihood at the nth iterate

tcensReg_llike_sepvar *Log Likelihood for Model for J independent Truncated Normal Variables with Same Mean Structure but Different Variance*

Description

This function is part of the supporting function space to calculate the value of the log-likelihood at the nth iteration of theta. For this parametrization we assume that there exists (p-1) linear parameters for the mean and J separate variance parameters.

Usage

```
tcensReg_llike_sepvar(theta, y, X, group, a = -Inf, v = NULL)
```

Arguments

theta	Numeric vector containing the values at the present iterate for the (p-1) fixed mean parameters and the J log sigma values
y	Numeric vector containing the outcome
X	Numeric design matrix
group	Factor variable used to define the J groups
a	Numeric scalar defining the common truncation value for each Truncated Normal
v	Numeric scalar defining the common censoring value

Value

Scalar value of the log-likelihood at the nth-iterate

---

tcensReg_llike_sepvar_maxLik

Log Likelihood for Model for J independent Truncated Normal Variables with Same Mean Structure but Different Variance

---

Description

This function is part of the supporting function space to calculate the value of the log-likelihood at the nth iteration of theta. For this parametrization we assume that there exists (p-1) linear parameters for the mean and J separate variance parameters.

Usage

```
tcensReg_llike_sepvar_maxLik(theta, y, X, group, left_trunc = -Inf,
  v = NULL)
```

Arguments

theta	Numeric vector containing the values at the present iterate for the (p-1) fixed mean parameters and the J log sigma values
y	Numeric vector containing the outcome
X	Numeric design matrix
group	Factor variable used to define the J groups
left_trunc	Numeric scalar defining the common truncation value for each Truncated Normal
v	Numeric scalar defining the common censoring value

Value

Scalar value of the log-likelihood at the nth-iterate

tcensReg_newton *Newton-Raphson Algorithm for Truncated Normal Distribution with Censoring with Linear Equation Mean*

Description

Newton-Raphson Algorithm for Truncated Normal Distribution with Censoring with Linear Equation Mean

Usage

```
tcensReg_newton(y, X, a = -Inf, v = NULL, epsilon = 1e-04,
  tol_val = 1e-06, max_iter = 100, step_max = 10,
  theta_init = NULL)
```

Arguments

y	Numeric vector with the observed truncated and censored outcomes
X	Numeric design matrix
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
epsilon	Numeric value used to define when the algorithm should stop when the gradient is less then 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
max_iter	Maximum number of iterations for algorithm. Default is 100
step_max	Maximum number of steps when performing line search. Default is 10
theta_init	Initial values of theta provided by the user. If unspecified then calculcates values from OLS regression

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.

`tcensReg_optim` *Optimization for Truncated Normal Distribution with Censoring with Linear Equation Mean using optim*

Description

Optimization for Truncated Normal Distribution with Censoring with Linear Equation Mean using optim

Usage

```
tcensReg_optim(y, X, a = -Inf, v = NULL, method, epsilon = 1e-04,
  tol_val = 1e-06, max_iter = 100, step_max = 10,
  theta_init = NULL)
```

Arguments

`y`	Numeric vector with the observed truncated and censored outcomes
`X`	Numeric design matrix
`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
`method`	Character value indicating which optimization method to perform.
`epsilon`	Numeric value used to define when the algorithm should stop when the gradient is less then 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
`max_iter`	Maximum number of iterations for algorithm. Default is 100
`step_max`	Maximum number of steps when performing line search. Default is 10
`theta_init`	Initial values of theta provided by the user. If unspecified then calculcates values from OLS regression

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.

tcensReg_sepvar *Regression Method for Truncated Normal Distribution with Censoring for Indepentend 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 indiciating 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 from `tcensReg_newton` such as `max_iter`, `step_max`, or `epsilon`.

Details

Currently available optimization routines include conjugate gradient (`CG`), Newton-Raphson type via maxLike 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

censReg, *3*

maxLik, *11*

optim, *3*, *11*

rtnorm, 2

tcensReg, 3
tcensReg_gradient, 4
tcensReg_gradient_sepvar, 5
tcensReg_gradient_sepvar_maxLik, 5
tcensReg_hess, 6
tcensReg_llike, 7
tcensReg_llike_sepvar, 7
tcensReg_llike_sepvar_maxLik, 8
tcensReg_newton, *3*, 9, *11*
tcensReg_optim, 10
tcensReg_sepvar, 11
truncreg, *3*