Package ‘truncreg’

February 20, 2015

Version 0.2-1
Date 2013-12-24
Title Truncated Gaussian Regression Models
Depends R (>= 1.8.0), maxLik
Suggests survival
Description Estimation of models for truncated Gaussian variables by maximum likelihood.
License GPL (>= 2)

URL http://R-Forge.R-project.org/projects/truncreg/

Author Yves Croissant [aut, cre],
Achim Zeileis [aut]

Maintainer Yves Croissant <yves.croissant@univ-reunion.fr>

Repository CRAN
Repository/R-Forge/Project truncreg
Repository/R-Forge/Revision 5
Repository/R-Forge/DateTimeStamp 2013-12-24 09:59:06
Date/Publication 2013-12-27 13:28:25

R topics documented:

truncreg ................................................................. 2

Index 5
truncreg  

*Truncated Gaussian Response Models*

**Description**

Estimation of models for truncated Gaussian variables by maximum likelihood.

**Usage**

```
truncreg(formula, data, subset, weights, na.action,
         point = 0, direction = "left",
         model = TRUE, y = FALSE, x = FALSE, ...)
```

**Arguments**

- `formula`: a symbolic description of the model to be estimated,
- `data`: the data,
- `subset`: an optional vector specifying a subset of observations,
- `weights`: an optional vector of weights,
- `na.action`: a function which indicates what should happen when the data contains 'NA's,
- `point`: the value of truncation (the default is 0),
- `direction`: the direction of the truncation, either "left" (the default) or "right",
- `model, y, x`: logicals. If TRUE the corresponding components of the fit (model frame, response, model matrix) are returned.
- `...`: further arguments.

**Details**

The model is estimated with the `maxLik` package and the Newton-Raphson method, using analytic gradient and Hessian.

A set of standard extractor functions for fitted model objects is available for objects of class "truncreg", including methods to the generic functions `print`, `summary`, `coef`, `vcov`, `logLik`, `residuals`, `predict`, `fitted`, `model.frame`, and `model.matrix`.

**Value**

An object of class "truncreg", a list with elements:

- `coefficients`: the named vector of coefficients,
- `vcov`: the variance matrix of the coefficients,
- `fitted.values`: the fitted values,
- `logLik`: the value of the log-likelihood,
- `gradient`: the gradient of the log-likelihood at convergence,
truncreg

nobs the number of observations,
call the matched call,
terms the model terms,
model the model frame used (if model = TRUE),
y the response vector (if y = TRUE),
x the model matrix (if x = TRUE),
point the truncation point used,
direction the truncation direction used,
est.stat some information about the estimation (time used, optimization method),

References


See Also

maxlik, mhurdle

Examples

```
# Artificial example

## simulate a data.frame
set.seed(1071)
n <- 10000
sigma <- 4
alpha <- 2
beta <- 1
x <- rnorm(n, mean = 0, sd = 2)
eps <- rnorm(n, sd = sigma)
y <- alpha + beta * x + eps
d <- data.frame(y = y, x = x)

## truncated response
d$yt <- ifelse(d$y > 1, d$y, NA)

## binary threshold response
d$yb <- factor(d$y > 0)
```
truncreg

## censored response

d$yc <- pmax(1, d$y)

## compare estimates for full/truncated/censored/threshold response

fm_full <- lm(y ~ x, data = d)
fm_trunc <- truncreg(yt ~ x, data = d, point = 1, direction = "left")
fm_thresh <- glm(yb ~ x, data = d, family = binomial(link = "probit"))
library("survival")
fm_cens <- survreg(Surv(y, yc > 1, type = "left") ~ x, data = d, dist = "gaussian")

## compare scaled regression coefficients

cbind(
  "True" = c(alpha, beta) / sigma,
  "Full" = coef(fm_full) / summary(fm_full)$sigma,
  "Truncated" = coef(fm_trunc)[1:2] / coef(fm_trunc)[3],
  "Censored" = coef(fm_cens) / fm_cens$scale,
  "Threshold" = coef(fm_thresh)
)

--------------------
## Tobin's durable goods data ##
--------------------

## Tobit model (Tobin 1958)
data("tobin", package = "survival")
tobit <- survreg(Surv(durable, durable > 0, type = "left") ~ age + quant,
                 data = tobin, dist = "gaussian")

## Two-part model (Cragg 1971)
## (see "mhurdle" package for a combined solution)
cragg_probit <- glm(factor(durable > 0) ~ age + quant,
                     data = tobin, family = binomial(link = "logit"))
cragg_trunc <- truncreg(durable ~ age + quant, data = tobin, subset = durable > 0)

## Scaled coefficients

cbind(
  "Tobit" = coef(tobit) / tobit$scale,
  "Binary" = coef(cragg_probit),
  "Truncated" = coef(cragg_trunc)[1:3] / coef(cragg_trunc)[4])

## likelihood ratio test and BIC

ll <- c("Tobit" = tobit$loglik[1],
        "Two-Part" = as.vector(logLik(cragg_probit) + logLik(cragg_trunc)))

ll <- c(4, 3 + 4)
pchisq(2 * diff(ll), df, lower.tail = FALSE)
-2 * ll + log(nrow(tobin)) * df
Index

*Topic regression
  truncreg, 2

  coef, 2
  coef.truncreg(truncreg), 2

  fitted, 2
  fitted.truncreg(truncreg), 2

  logLik, 2
  logLik.truncreg(truncreg), 2

  maxLik, 2, 3
  m hurdle, 3
  model.frame, 2
  model.frame.truncreg(truncreg), 2
  model.matrix, 2
  model.matrix.truncreg(truncreg), 2

  predict, 2
  predict.truncreg(truncreg), 2
  print, 2
  print.summary.truncreg(truncreg), 2
  print.truncreg(truncreg), 2

  residuals, 2
  residuals.truncreg(truncreg), 2

  summary, 2
  summary.truncreg(truncreg), 2

  truncreg, 2

  vcov, 2
  vcov.truncreg(truncreg), 2