

Package ‘cotram’

March 30, 2020

Title Count Transformation Models

Version 0.2-0

Date 2020-03-30

Description Count transformation models featuring parameters interpretable as discrete hazard ratios, odds ratios, reverse-time discrete hazard ratios, or transformed expectations. An appropriate data transformation for a count outcome and regression coefficients are simultaneously estimated by maximising the exact discrete log-likelihood using the computational framework provided in package 'mlt', technical details are given in Siegfried & Hothorn (2020) <DOI:10.1111/2041-210X.13383>.

Depends tram (>= 0.2-6), mlt (>= 1.0-5)

Imports variables (>= 1.0-2), basefun (>= 1.0-5)

Suggests TH.data, knitr, lattice, colorspace, multcomp

VignetteBuilder knitr

Encoding UTF-8

License GPL-2

NeedsCompilation no

Author Sandra Siegfried [aut],
Torsten Hothorn [aut, cre] (<<https://orcid.org/0000-0001-8301-0471>>)

Maintainer Torsten Hothorn <Torsten.Hothorn@R-project.org>

Repository CRAN

Date/Publication 2020-03-30 19:00:02 UTC

R topics documented:

confband	2
cotram	3
cotram-methods	5

Index	7
--------------	----------

 confband

Confidence Bands

Description

Confidence bands for transformation, distribution, survivor or cumulative hazard functions

Usage

```
confband(object, newdata, level = 0.95, ...)
## S3 method for class 'cotram'
confband(object, newdata, level = 0.95,
         type = c("trafo", "distribution", "survivor", "cumhazard"),
         smooth = FALSE, q = NULL, K = 20, cheat = K, ...)
```

Arguments

object	an object of class cotram .
newdata	a data frame of observations.
level	the confidence level.
type	the function to compute the confidence band for.
smooth	logical; if TRUE a smoothed function of type is returned.
q	quantiles at which to evaluate the model.
K	number of grid points the function is evaluated at (in the absence of q and smooth = TRUE).
cheat	number of grid points the function is evaluated at when using the quantile obtained for K grid points (in the absence of q and smooth = TRUE).
...	additional arguments to confint.glht .

Details

The function is evaluated at the count response or at K grid points and simultaneous confidence intervals are then interpolated in order to construct the band.

Value

For each row in newdata the function and corresponding confidence band evaluated at the count response (or K or cheat grid points) is returned.

Examples

```
data("birds", package = "TH.data")

### fit count transformation model with cloglog link
m_birds <- cotram(SG5 ~ AOT + AFS + GST + DBH + DWC + LOG, data = birds,
                 method = "cloglog")

### compute asymptotic confidence bands for the distribution function
### for the first observation
confband(m_birds, newdata = birds[1, ], type = "distribution")
```

cotram

Count Transformation Models

Description

Likelihood-based count transformation models for fully parameterised discrete conditional distribution functions. The link function governing the interpretation of the predictor can be chosen and results in discrete hazard ratios, odds ratios, reverse time hazard ratios or conditional expectation of transformed counts.

Usage

```
cotram(formula, data, method = c("logit", "cloglog", "loglog", "probit"),
       log_first = TRUE, plus_one = log_first, prob = 0.9,
       subset, weights, offset, cluster, na.action = na.omit, ...)
```

Arguments

formula	an object of class "formula": a symbolic description of the model structure to be fitted. The details of model specification are given under tram and in the package vignette.
data	an optional data frame, list or environment (or object coercible by <code>as.data.frame</code> to a data frame) containing the variables in the model. If not found in <code>data</code> , the variables are taken from <code>environment(formula)</code> .
method	character specifying the choice of the link function, mapping the transformation function into probabilities. Available choices include the logit, complementary log-log, log-log or probit link. The different link functions govern the interpretation of the linear predictor. Details of the interpretation can be found in the package vignette.
prob	probability giving the quantile of the response defining the upper limit of the support of a smooth Bernstein polynomial.
log_first	logical; if TRUE, a Bernstein polynomial is defined on the log-scale.
plus_one	logical; if TRUE, a Bernstein polynomial of $(y + 1)$ is defined.

subset	an optional vector specifying a subset of observations to be used in the fitting process.
weights	an optional vector of weights to be used in the fitting process. Should be NULL or a numeric vector. If present, the weighted log-likelihood is maximised.
offset	this can be used to specify an <code>_a priori_</code> known component to be included in the linear predictor during fitting. This should be NULL or a numeric vector of length equal to the number of cases.
cluster	optional factor with a cluster ID employed for computing clustered covariances.
na.action	a function which indicates what should happen when the data contain NAs. The default is set to <code>na.omit</code> .
...	additional arguments to <code>tram</code> .

Details

Likelihood-based estimation of a fully parameterised conditional discrete distribution function for count data, while ensuring interpretability of the linear predictors. The models are defined with a negative shift term relating positive predictors to larger values of the conditional mean. For the model with logistic or cloglog link `exp(-coef())` is the multiplicative change of discrete odds-ratios or hazard ratios. For the model with loglog link `exp(coef())` is the multiplicative change of the reverse time hazard ratios. Applying a transformation model with probit link `coef()` gives the conditional expectation of the transformed link, with transformation function estimated from data.

Value

An object of class `cotram` and `tram`, with corresponding `coef`, `vcov`, `logLik`, `summary`, `print`, `plot` and `predict` methods.

References

- Sandra Siegfried, Torsten Hothorn (2020), Count Transformation Models, *Methods in Ecology and Evolution*, doi: [10.1111/2041210X.13383](https://doi.org/10.1111/2041210X.13383).
- Torsten Hothorn, Lisa Moest, Peter Buehlmann (2018), Most Likely Transformations, *Scandinavian Journal of Statistics*, **45**(1), 110–134, doi: [10.1111/sjos.12291](https://doi.org/10.1111/sjos.12291).
- Torsten Hothorn (2020), Most Likely Transformations: The `mlt` Package, *Journal of Statistical Software*, **92**(1), 1–68, doi: [10.18637/jss.v092.i01](https://doi.org/10.18637/jss.v092.i01).

Examples

```
data("birds", package = "TH.data")
cotram(SG5 ~ AOT + AFS + GST + DBH + DWC + LOG, data = birds)
```

Description

Methods for objects inheriting from class cotram

Usage

```
## S3 method for class 'cotram'
predict(object, newdata = model.frame(object),
        type = c("lp", "trafo", "distribution", "survivor",
                 "density", "logdensity", "cumhazard", "quantile"),
        smooth = FALSE, q = NULL, K = 20, prob = 1:(10-1)/10, ...)
## S3 method for class 'cotram'
plot(x, newdata, type = c("distribution", "survivor", "density",
                          "logdensity", "cumhazard", "quantile", "trafo"),
     confidence = c("none", "band"), level = 0.95,
     smooth = FALSE, q = NULL, K = 20, cheat = K, prob = 1:(10-1)/10,
     col = "black", fill = "lightgrey",
     lty = 1, lwd = 1, add = FALSE, ...)
## S3 method for class 'cotram'
as.mlt(object)
## S3 method for class 'cotram'
logLik(object, parm = coef(as.mlt(object)), fixed = FALSE, newdata, ...)
```

Arguments

object, x	a fitted linear count transformation model inheriting from class cotram.
newdata	an optional data frame of observations.
parm	model parameters.
type	type of prediction, current options include linear predictors ("lp", of x variables in the formula $y \sim x$), transformation functions ("trafo") or distribution functions on the scale of the cdf ("distribution"), survivor function, density function, log-density function, cumulative hazard function or quantile function.
confidence	whether to plot a confidence band (see confband).
level	the confidence level.
smooth	logical; if TRUE a smoothed function of type is returned.
q	quantiles at which to evaluate the model.
prob	probabilities for the evaluation of the quantile function (type = "quantile").
K	number of grid points the function is evaluated at (for smooth = TRUE and in absence of q).
cheat	number of grid points the function is evaluated at when using the quantile obtained for K grid points (in the absence of q and smooth = TRUE).

<code>col</code>	color for the lines to plot.
<code>fill</code>	color for the confidence band.
<code>lty</code>	line type for the lines to plot.
<code>lwd</code>	line width.
<code>add</code>	logical; indicating if a new plot shall be generated (the default).
<code>...</code>	additional arguments to the underlying methods for predict.

Details

`predict` and `plot` can be used to inspect the model on different scales.

See Also

[predict.cotram](#), [confband.cotram](#), [tram-methods](#), [mlt-methods](#), [plot.ctm](#)

Examples

```
data("birds", package = "TH.data")

### fit count transformation model with cloglog link
m_birds <- cotram(SG5 ~ AOT + AFS + GST + DBH + DWC + LOG, data = birds,
                 method = "cloglog")
logLik(m_birds)

### classical likelihood inference
summary(m_birds)

### coefficients of the linear predictor (discrete hazard ratios)
exp(-coef(m_birds))

### compute predicted median along with 10% and 90% quantile for the first
### three observations
nd <- birds[1:3,]
predict(m_birds, newdata = nd, type = "quantile", prob = c(.1, .5, .9),
       smooth = TRUE)

### plot the predicted distribution for these observations
plot(m_birds, newdata = nd, type = "distribution",
     col = c("skyblue", "grey", "seagreen"))
```

Index

*Topic **models**

cotram, 3

*Topic **regression**

cotram, 3

as.mlt.cotram (cotram-methods), 5

confband, 2, 5

confband.cotram, 6

confint.giht, 2

cotram, 2, 3

cotram-methods, 5

logLik.cotram (cotram-methods), 5

plot.cotram (cotram-methods), 5

plot.ctm, 6

predict.cotram, 6

predict.cotram (cotram-methods), 5

tram, 3, 4