Title  Ordinal Regression Analysis for Continuous Scales
Version  2.0.1
Author  Maurizio Manuguerra [aut, cre],
        Gillian Heller [aut]
Maintainer  Maurizio Manuguerra <maurizio.manuguerra@mq.edu.au>
Description  A regression framework for response variables which are continuous
             self-rating scales such as the Visual Analog Scale (VAS) used in pain
             assessment, or the Linear Analog Self-Assessment (LASA) scales in quality
             of life studies. These scales measure subjects' perception of an intangible
             quantity, and cannot be handled as ratio variables because of their inherent
             non-linearity. We treat them as ordinal variables, measured on a continuous
             scale. A function (the g function) connects the scale with an underlying
             continuous latent variable. The link function is the inverse of the CDF of the
             assumed underlying distribution of the latent variable. A variety of
             link functions are currently implemented.
Depends  R (>= 3.3.0), boot, splines, Deriv
License  GPL (>= 2)
LazyData  true
RoxygenNote  6.1.0
Encoding  UTF-8
NeedsCompilation  no
Repository  CRAN
Date/Publication  2019-02-13 23:20:03 UTC

R topics documented:

  ordinalCont-package ................................................. 2
 anova.ocm ........................................................... 3
  ANZ0001 ............................................................ 4
  ANZ0001.sub ....................................................... 5
  coef.ocm ........................................................... 6
  deriv_link ......................................................... 7
Ordinal regression analysis is a convenient tool for analyzing ordinal response variables in the presence of covariates. We extend this methodology to the case of continuous self-rating scales such as the Visual Analog Scale (VAS) used in pain assessment, or the Linear Analog Self-Assessment (LASA) scales in quality of life studies. Subjects are typically given a linear scale of 100 mm and asked to put a mark where they perceive themselves. These scales measure subjects' perception of an intangible quantity, and cannot be handled as ratio variables because of their inherent nonlinearity. Instead we treat them as ordinal variables, measured on a continuous scale. We express the likelihood in terms of a function (the “g function”) connecting the scale with an underlying continuous latent variable. In the current version the g function is expressed with monotone increasing I-splines (Ramsey 1988). The link function is the inverse of the CDF of the assumed underlying distribution of the latent variable. Currently the logit link, which corresponds to a standard logistic distribution, is implemented. (This implies a proportional odds model.) The likelihood is maximized using the MI algorithm (Ma, 2010). Fixed- and mixed-effects models are implemented in the function ocm.
Author(s)
Maurizio Manuguerra, Gillian Heller

References


Description
Comparison of continuous ordinal models using likelihood ratio tests.

Usage
### S3 method for class 'ocm'
anova(object, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>an object of class ocm</td>
</tr>
<tr>
<td>...</td>
<td>one or more additional ocm objects</td>
</tr>
</tbody>
</table>

Details
Likelihood ratio testing of nested models is performed.

Value
The method returns an object of class anova.ocm and data.frame, reporting for each model, in hierarchical order:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>no.par</td>
<td>number of parameters</td>
</tr>
<tr>
<td>AIC</td>
<td>Akaike information criterion</td>
</tr>
<tr>
<td>loglik</td>
<td>log-likelihood</td>
</tr>
</tbody>
</table>
**Description**

The complete ANZ0001 trial data set

**Usage**

```r
data(ANZ0001)
```

**Format**

A data frame with 2473 rows and 11 variables

**Details**

The ANZ0001 trial, conducted by the ANZ Breast Cancer Trials Group, is an unblinded, multi-centre, randomized trial with three chemotherapy treatment arms, concluded in 2005 (Stockler et al 2007). Health-related quality of life measures (Overall quality of life, Physical Well-Being, Mood, Pain, Nausea and Vomiting, Appetite) are assessed at each chemotherapy treatment cycle, from randomization until disease progression, when treatment is interrupted. The treatments Intermittent Capecitabine (IC) and Continuous Capecitabine (CC) are compared with the standard combination treatment CMF, each with its own protocol. There is no maximum duration of treatment, but it is interrupted on disease progression, or when patient intolerance or unacceptable toxicity are recorded. The data set is extracted from the ANZ0001 trial and contains information from 292 patients with complete quality of life measurements.

The variables are as follows:

- **randno**: patient ID number
- **cycleno**: chemotherapy cycle number
- **age**: age of patient at entry to study
bsa

Body Surface Area (m$^2$)

treatment
treatment received by patient (1,2,3)

overall

Overall quality of life as recorded by the patient on a LASA scale, normalized to (0, 1)

phys

Physical Well-Being as recorded by the patient on a LASA scale, normalized to (0, 1)

mood

Mood as recorded by the patient on a LASA scale, normalized to (0, 1)

pain

Pain as recorded by the patient on a LASA scale, normalized to (0, 1)

nausvom

Nausea and Vomiting as recorded by the patient on a LASA scale, normalized to (0, 1)

appetite

Appetite as recorded by the patient on a LASA scale, normalized to (0, 1)

References

Stockler, M., T. Sourjina, P. Grimison, V. Gebski, M. Byrne, V. Harvey, P. Francis et al. “A randomized trial of capecitabine (C) given intermittently (IC) rather than continuously (CC) compared to classical CMF as first-line chemotherapy for advanced breast cancer (ABC).” In ASCO Annual Meeting Proceedings, vol. 25, no. 18_suppl, p. 1031. 2007.

Description

A subset from the ANZ0001 trial data set

Usage

data(ANZ0001.sub)

Format

A data frame with 428 rows and 11 variables

Details

The ANZ0001 trial, conducted by the ANZ Breast Cancer Trials Group, is an unblinded, multi-centre, randomized trial with three chemotherapy treatment arms, concluded in 2005 (Stockler et al 2007). Health-related quality of life measures (Overall quality of life, Physical Well-Being, Mood, Pain, Nausea and Vomiting, Appetite) are assessed at each chemotherapy treatment cycle, from randomization until disease progression, when treatment is interrupted. The treatments Intermittent Capecitabine (IC) and Continuous Capecitabine (CC) are compared with the standard combination treatment CMF, each with its own protocol. There is no maximum duration of treatment, but it is interrupted on disease progression, or when patient intolerance or unacceptable toxicity are recorded. The data set is extracted from the ANZ0001 trial and contains information from a subset of 292 patients with complete quality of life measurements, limited to cycle numbers 0 and 5.

The variables are as follows:

randno

patient ID number

cycleno

chemotherapy cycle number, either 0 (initial assessment) or 1 (fifth cycle).
age  age of patient at entry to study
bsa  Body Surface Area (m$^2$)
treatment  treatment received by patient (1,2,3)
overall  Overall quality of life as recorded by the patient on a LASA scale, normalized to (0, 1)
phys  Physical Well-Being as recorded by the patient on a LASA scale, normalized to (0, 1)
mood  Mood as recorded by the patient on a LASA scale, normalized to (0, 1)
pain  Pain as recorded by the patient on a LASA scale, normalized to (0, 1)
ausvom  Nausea and Vomiting as recorded by the patient on a LASA scale, normalized to (0, 1)
appetite  Appetite as recorded by the patient on a LASA scale, normalized to (0, 1)

References

Stockler, M., T. Sourjina, P. Grimison, V. Gebski, M. Byrne, V. Harvey, P. Francis et al. “A randomized trial of capecitabine (C) given intermittently (IC) rather than continuously (CC) compared to classical CMF as first-line chemotherapy for advanced breast cancer (ABC).” In ASCO Annual Meeting Proceedings, vol. 25, no. 18_suppl, p. 1031. 2007.

---

**coef.ocm**

**Extract Model Coefficients**

**Description**

`coef.ocm` is the ordinalCont specific method for the generic function `coeff`, which extracts model coefficients from objects of class `ocm`.

**Usage**

```r
## S3 method for class 'ocm'
coef(object, ...)
```

**Arguments**

- `object`  an object of class `ocm`, usually, a result of a call to `ocm`.
- `...`  further arguments passed to or from other methods.

**Value**

A named numeric vector with the coefficients extracted from the model object.

**Author(s)**

Maurizio Manuguerra, Gillian Heller
Function to compute the derivatives of the link function needed by the algorithm

### deriv_link

**Description**

Function to compute the derivatives of the link function needed by the algorithm

**Usage**

```r
derv_link(link = c("logit", "probit", "cloglog", "loglog", "cauchit"))
```

**Arguments**

- `link`: One of "logit" (default), "probit", "cloglog", "loglog" or "cauchit".

**Value**

A list with the link function and the 1st, 2nd and 3rd derivatives with respect to the argument

---

Extract the deviance from a fitted Continuous Ordinal Model

### deviance.ocm

**Description**

Extracts the absolute conditional deviance for a fitted ocm object

**Usage**

```r
deviance(object, ...)  
```

**Arguments**

- `object`: ocm object
- `...`: further arguments to be passed to methods

**Details**

The deviance is computed as:

\[ -2\ell \]

where \( \ell \) is the conditional penalized log-likelihood.

**Value**

The value of the deviance extracted from `object`. 
**Author(s)**

Maurizio Manuguerra, Gillian Heller

**See Also**

ocm

**Examples**

```r
## Not run:
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
device(fit.overall)

## End(Not run)
```

---

**extractAIC.ocm**

Extract AIC from a fitted Continuous Ordinal Model

**Description**

Extracts the AIC for a fitted ocm object

**Usage**

```r
## S3 method for class 'ocm'
extractAIC(fit, scale = 0, k = 2, ...)
```

**Arguments**

- `fit` ocm object
- `scale` parameter currently not used. For compatibility with general extractAIC method.
- `k` “weight” of the equivalent degrees of freedom (=: edf) in the AIC formula. Defaults to 2
- `...` further arguments to be passed to methods

**Details**

The generalized AIC is computed:

\[-2\ell + k \cdot edf\]

where \(\ell\) is the log-likelihood, \(k=2\) gives the AIC, and \(k=\log(n)\) gives the BIC.

**Value**

A numeric vector of length 2, with first and second elements giving

- `edf` the “equivalent degrees of freedom” for the fitted model `fit`
- `AIC` the generalized AIC of ocm object `fit`
Author(s)

Maurizio Manuguerra, Gillian Heller

References


See Also

ocm

Examples

## Not run
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
extractAIC(fit.overall)

## End(Not run)

fitted.ocm

Extract Model Fitted Values

Description

fitted.ocm is the ordinalCont specific method for the generic function fitted, which computes model fitted from objects of class ocm.

Usage

## S3 method for class 'ocm'
fitted(object, ...)

Arguments

object an object of class ocm, usually, a result of a call to ocm.
...

further arguments passed to or from other methods.

Details

An object of class ocm is used to compute the probability densities of the continuous ordinal score. The fitted values are the means of such probability density functions. The output is scaled following the original scale of the scores.

Value

Fitted values computed from object.
Author(s)
Maurizio Manuguerra, Gillian Heller

---

**formula.ocm**  
*Model Formulae*

**Description**
formula.ocm is the ordinalCont specific method for the generic function formula, which extracts the model formula from objects of class ocm.

**Usage**

```r
## S3 method for class 'ocm'
formula(x, ...) 
```

**Arguments**

- `x` an object of class ocm, usually, a result of a call to ocm.
- `...` further arguments passed to or from other methods.

**Value**

A symbolic model formula extracted from the model object.

**Author(s)**
Maurizio Manuguerra, Gillian Heller

---

**get_gfun**  
*Estimated g function for a Fitted Model Object*

**Description**
Calculates the estimated g function for a fitted ocm object.

**Usage**

```r
get_gfun(object, ...) 
```

```r
## S3 method for class 'ocm'
get_gfun(object, ...) 
```
**inv_link**

**Description**

Function to compute inverse link functions

**Usage**

```r
inv_link(link = c("logit", "probit", "cloglog", "loglog", "cauchit"))
```

**Arguments**

- `link` One of "logit" (default), "probit", "cloglog", "loglog" or "cauchit".

**Value**

A list with the link function and the 1st, 2nd and 3rd derivatives with respect to the argument.

---

**Arguments**

- `object` an `ocm` object
- `...` further arguments to be passed to methods

**Value**

A dataframe containing four columns: the values of the score v, the estimated g function and the 95% CIs

**Author(s)**

Maurizio Manuguerra, Gillian Heller

**See Also**

`ocm`

**Examples**

```r
## Not run:
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
gfun.fit.overall
## End(Not run)
```
Description

Extracts the log-likelihood for a fitted ocm object

Usage

```r
## S3 method for class 'ocm'
logLik(object, ...)
```

Arguments

- `object`: an ocm object
- `...`: further arguments to be passed to methods

Value

The log-likelihood of an ocm object. This is a number with attributes

- `df`: estimated degrees of freedom for the fitted model object. When the model maximizes the penalized likelihood, i.e. smoothing is involved in the g function or the formula contains random effects, the effective degrees of freedom are returned.
- `nobs`: number of observations used in the fitted model object
- `class`: class of the returned object: logLik.ocm

Author(s)

Maurizio Manuguerra, Gillian Heller

See Also

ocm

Examples

```r
## Not run:
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
logLik(fit.overall)

## End(Not run)
```
**model.frame.ocm**  

**Model Frame**

**Description**

model.frame.ocm is the ordinalCont specific method for the generic function model.frame, which return a data.frame with the variables needed to use formula and any ... arguments.

**Usage**

```r
## S3 method for class 'ocm'
model.frame(formula, data, random.terms = TRUE, ...)
```

**Arguments**

- `formula`: a model formula
- `data`: a data.frame containing the variables in formula.
- `random.terms`: a logical indicating if random terms have to be included in the terms object. Defaults to TRUE.
- `...`: a mix of further arguments to pass to the default method.

**Value**

A c("data.frame") with the variables needed to obtain object.

**Author(s)**

Maurizio Manuguerra, Gillian Heller

---

**model.matrix.ocm**  

**Model Matrix**

**Description**

model.matrix.ocm is the ordinalCont specific method for the generic function model.matrix, which extracts the model matrix from objects of class ocm.

**Usage**

```r
## S3 method for class 'ocm'
model.matrix(object, random.terms = TRUE, ...)
```
Arguments

object an object of class ocm, usually, a result of a call to ocm.
random.terms a logical indicating if random terms have to be included in the terms object. Defaults to TRUE.
... further arguments passed to or from other methods.

Value

A design (or model) matrix with the variables needed to obtain the object x, e.g., by expanding factors to a set of dummy variables and expanding interactions similarly.

Author(s)

Maurizio Manuguerra, Gillian Heller

Description

A subset from an Australian chronic neck pain study

Usage

data(neck_pain)

Format

A data frame with 264 rows and 4 variables

Details

A randomized, double-blind, placebo-controlled study of low-level laser therapy (LLLT) in 88 subjects with chronic neck pain was conducted with the aim of determining the efficacy of 300 mW, 830 nm laser in the management of chronic neck pain. Subjects were randomized to receive a course of 14 treatments over 7 weeks with either active or sham laser to tender areas in the neck. The primary outcome measure was change in a 10 cm Visual Analogue Scale (VAS) for pain. Measurements were taken at baseline, at the end of 7 weeks\' treatment and 12 weeks from baseline.

The variables are as follows:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>patient ID number</td>
</tr>
<tr>
<td>vas</td>
<td>Neck pain as recorded by the patient on a VAS scale, normalized to (0, 1)</td>
</tr>
<tr>
<td>laser</td>
<td>laser treatment received by patient, either 1 (active) or 2 (placebo)</td>
</tr>
<tr>
<td>time</td>
<td>the measurement time, either 1 (initial assessment), 2 (after 7 weeks) or 3 (after 12 weeks).</td>
</tr>
</tbody>
</table>
References


---

nobs.ocm

**Extract Model Coefficients**

**Description**

nobs.ocm is the ordinalCont specific method for the generic function nobs, which returns number of observations from objects of class ocm.

**Usage**

```r
## S3 method for class 'ocm'
nobs(object, ...)
```

**Arguments**

- `object` an object of class ocm, usually, a result of a call to ocm.
- `...` further arguments passed to or from other methods.

**Value**

The (numeric) number of observations in the model object.

**Author(s)**

Maurizio Manuguerra, Gillian Heller

---

ocm

**Ordinal regression for continuous scales**

**Description**

Continuous ordinal regression with logit link using I-splines to model the g function.

**Usage**

```r
ocm(formula, data = NULL, scale = NULL, weights, link = c("logit", "probit", "cloglog", "loglog", "cauchit"), niter = c(500, 500), conv_crit = 0.01, n.int.knots = NULL, order = 4, lambdas = NA)
```
Arguments

- **formula**: a formula expression as for regression models, of the form \( \text{response} \sim \text{predictors} \). Only fixed effects are supported. The model must have an intercept: attempts to remove one will lead to a warning and will be ignored.

- **data**: an optional data frame in which to interpret the variables occurring in the formulas.

- **scale**: a vector of length 2 with the boundaries of the ordinal scale used. If not specified, the range of the data is used, and a warning is displayed.

- **weights**: optional case weights in fitting. Defaults to 1.

- **link**: link function, i.e. the type of location-scale distribution assumed for the latent distribution. The default “logit” link gives the proportional odds model. Other options are "logit", "probit", "cloglog", "loglog", "cauchit".

- **niter**: a vector of length 2 with the maximum number of external and internal iterations used in the fitting algorithm. The internal algorithm estimates the parameters of the model conditional on the current values of \( \lambda \)'s, the smoothing parameters. The external algorithm estimates the values of \( \lambda \)'s conditional on the current estimates of the parameters of the model. Default is \( \lambda_{HUPPLUPPI} \).

- **conv_crit**: the smoothing parameters \( \lambda \)'s convergence criteria for the iterative process. Default is 0.01

- **n.int.knots**: the number of internal knots used to compute the spline bases. The default (NULL) is \( \text{round}((n-1\text{-order})*0.8) \) if in the interval \([8,15]\), and 8 or 15 otherwise.

- **order**: the order of the spline functions. The default is 4 (cubic splines).

- **lambdas**: NA (the default) or a vector of length equal to the number of smoothing terms, including the g function and, optionally, the random effect terms and the smoothers. If “lambdas” is a vector, each element \( \lambda_i \) can be a number, in which case the corresponding term is penalized using \( \lambda_i \) as smoothing parameter, zero, in which case the corresponding term is unpenalized, or NA, in which case the value of \( \lambda_i \) is estimated maximizing the marginal posterior function.

Details

Fits a continuous ordinal regression model using penalized maximum likelihood. The model can contain fixed effects and optionally mixed effects and smoothers. The g function is estimated using monotone increasing I-splines, and the link function is the logit, implying the standard logistic distribution for the latent variable. Penalized maximum likelihood estimation is performed using the MI algorithm and the splines smoothing parameters are estimated maximizing the marginal posterior (details of the iterative process are printed out during the fit).

Value

- an object of type `ocm` with the components listed below. Parameter estimates are in `coefficients`.

- **coefficients**: parameter estimates

- **pars_obj**: an object of class `ocmpars` carrying the parameter estimates and other properties of the regression terms
vcov
H
logLik
penlogLik
v
sample.size
edf
df.residual
nobs
terms
call
data
model.frame
model.matrix
weights
sorting
link
formula
scale

Author(s)
Maurizio Manuguerra, Gillian Heller

References

Examples
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
summary(fit.overall)
## Not run:
plot(fit.overall)
## Smoothers and complete data set
fit.overall.smooth <- ocm(overall ~ age + treatment : s(cycleno), data=ANZ0001, scale=c(0,100))
summary(fit.overall.smooth)
plot(fit.overall.smooth)

## End(Not run)
Description

Draws several summary and diagnostic plots, including the estimated \( g \) function, the estimated density function of the continuous ordinal score for the null model (no covariates), the histogram of the quantile residuals, the normal Q-Q plot and any smoother included in the model.

Usage

```r
## S3 method for class 'ocm'
plot(x, plot.only = NULL, CIs = c("vcov", "no", "rnd.x.bootstrap", "fix.x.bootstrap", "param.bootstrap"), R = 100,
     main_gfun = "g function", main_density = "Density function when X=0",
     xlab = "Continuous ordinal scale \[v]\", CIcol = "lightblue",
     individual_plots = F, ...)
```

Arguments

- `x`: an object of class \ocm\.
- `plot.only`: either NULL, in which case all plots are displayed, or a value among "gfun", "quant_resid", "QQplot" or "smoother", in which case only the requested plot is displayed.
- `CIs`: method used for confidence bands for the \( g \) function. "vcov" = Wald [default]; "no" = no CIS; "rnd.x.bootstrap" = random-x bootstrap; "fix.x.bootstrap" = bootstrap with fixed-x resampling; "param.bootstrap" = parametric bootstrap
- `R`: the number of bootstrap replicates. Ignored if CIs="no"
- `main_gfun`: title of the \( g \) function plot. Defaults to “\( g \) function (95% CIs)”
- `main_density`: title of the density function plot. Defaults to “Density function when X=0”
- `xlab`: label of the x axis for the \( g \) function and the density plots. Defaults to “Continuous ordinal scale \[v]\”
- `CIcol`: color of the confidence interval bands. Defaults to “lightblue”
- `individual_plots`: logical. If TRUE, every figure is drawn in a new window. If FALSE (default), the first four figures are drawn in a 2-by-2 array.
- `...`: further arguments passed to or from other methods

Details

The estimated \( g \) function, quantile residual histogram and normal Q-Q plot of an \ocm\ object are plotted. If smoothers are included in the formula, the user has the option to plot them in the same graph or separately. If CIs is not "no", 95% confidence bands are also plotted.
**predict.ocm**

**Author(s)**

Maurizio Manuguerra, Gillian Heller

**See Also**

ocm

**Examples**

```r
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
plot(fit.overall, CIs="vcov")
## Not run:
plot(fit.overall, CIs="rnd.x.bootstrap", R=100)
plot(fit.overall, CIs="fix.x.bootstrap", R=100)
plot(fit.overall, CIs="param.bootstrap", R=100)
## End(Not run)
```

---

**predict.ocm**

*Predict method for Continuous Ordinal Fits*

**Description**

Predicted values based on ocm object

**Usage**

```r
## S3 method for class 'ocm'
predict(object, newdata = NULL, type = c("response",
  "density", "CDF", "quantile", "regressor", "exp_regressor", "hazard",
  "cum_hazard", "survival"), prob = 1:(K - 1)/K, K = 50, ...)
```

**Arguments**

- **object**: an object of class ocm, usually a result of a call to ocm
- **newdata**: optionally, a data frame in which to look for variables with which to predict. Note that all predictor variables should be present, having the same names as the variables used to fit the model. If NULL, predictions are computed for the original dataset.
- **type**: type of prediction. One of "response" (default), "density", "CDF", "quantile", "regressor", "exp_regressor", "hazard", "cum_hazard" or "survival"
- **prob**: probabilities used to evaluate the quantile function (if type="quantile")
- **K**: number of evenly spaced values of v over which the probability density is evaluated (if type="density" or type="CDF") or number of probabilities at which the quantile function is evaluated (if type="quantile"). The default is 50.
- **...**: further arguments passed to or from other methods
Details

An object of class ocm and optionally a new data frame are used to compute the predictions. The estimated parameters of the fitted model and K values of v are used to compute the conditional probability density and the conditional cumulative distribution. If a new data frame is used to make predictions, the individual (random) effects are set to zero, while they are maintained to the estimated values if newdata is NULL.

Value

A vector of predictions, according to the type.

Author(s)

Maurizio Manuguerra, Gillian Heller

See Also

ocm

Examples

## Not run
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
pred <- predict(fit.overall)

## End(Not run)

print.anova.ocm Print anova.ocm objects

Description

Print the results of the comparison of continuous ordinal models in likelihood ratio tests.

Usage

## S3 method for class 'anova.ocm'
print(x, digits = max(getOption("digits") - 2, 3),
    signif.stars = getOption("show.signif.stars"), ...)

Arguments

x an object of class anova.ocm
digits controls the number of digits to print. Defaults to the maximum of the value returned by (getOption("digits") - 2) and 3
signif.stars a logical. Should the significance stars be printed? Defaults to the value returned by getOption("show.signif.stars")
... further arguments passed to or from other methods
print.ocm

Value

Prints anova.ocm object

Author(s)

Maurizio Manuguerra, Gillian Heller

See Also

ocm, anova.ocm

print.ocm

Print Continuous Ordinal Regression Objects

Description

print.ocm is the ordinalCont specific method for the generic function print, which prints objects of class ocm.

Usage

## S3 method for class 'ocm'
print(x, ...)

Arguments

x an object of class ocm, usually, a result of a call to ocm.
... further arguments passed to or from other methods.

Value

Prints an ocm object.

Author(s)

Maurizio Manuguerra, Gillian Heller

See Also

ocm, summary.ocm
summary.ocl Summary of Continuous Ordinal Fits

Description
Summary method for class ocl

Usage

## S3 method for class 'ocl'

```r
summary(object, full = F, ...)
```

Arguments
- `object` an object of class ocl, usually a result of a call to ocl
- `full` logical, if TRUE (the default) all the parameters are printed; if FALSE, only the fixed effects are printed.
- `...` further arguments passed to or from other methods

Author(s)
Maurizio Manuguerra, Gillian Heller

See Also
ocl, print.ocl

Examples

```r
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
summary(fit.overall)
```

terms.ocl Model Terms

Description
terms.ocl is the ordinalCont specific method for the generic function terms, which extracts model terms from objects of class ocl.

Usage

## S3 method for class 'ocl'

terms(x, random.terms = TRUE, ...)

**vcov.ocm**

**Arguments**

- `x`: an object of class `ocm`, usually, a result of a call to `ocm`.
- `random.terms`: a logical indicating if random terms have to be included in the terms object. Defaults to TRUE.
- `...`: further arguments passed to or from other methods.

**Value**

An object of class `c("terms", "formula")` which contains the terms representation of a symbolic model.

**Author(s)**

Maurizio Manuguerra, Gillian Heller

---

**vcov.Nocm**

**Variance-Covariance Matrix for a Fitted Model Object**

**Description**

Calculates variance-covariance matrix for a fitted `ocm` object

**Usage**

```r
## S3 method for class 'ocm'
vcov(object, ...)
```

**Arguments**

- `object`: an `ocm` object
- `...`: further arguments to be passed to methods

**Details**

For the generalized logistic g-function, the variance-covariance matrix of model parameters includes information on fixed- and random- effect terms and smoothing terms.

**Value**

Variance-covariance matrix of model parameters

**Author(s)**

Maurizio Manuguerra, Gillian Heller

**See Also**

`ocm`
Examples

```r
## Not run:
fit.overall <- ocm(overall ~ cycleno + age + bsa + treatment, data=ANZ0001.sub, scale=c(0,100))
vcov(fit.overall)

## End(Not run)
```
Index

*Topic anova
  anova.ocm, 3
  print.anova.ocm, 20
*Topic datasets
  ANZ0001, 4
  ANZ0001.sub, 5
  neck_pain, 14
*Topic likelihood,
  ocm, 15
  print.ocm, 21
*Topic log-likelihood,
  ocm, 15
*Topic log-likelihood.
  print.ocm, 21
*Topic ordinal
  ocm, 15
*Topic plot
  plot.ocm, 18
*Topic predict
  predict.ocm, 19
*Topic regression.
  ocm, 15
*Topic summary,
  print.anova.ocm, 20
*Topic summary
  summary.ocm, 22

anova.ocm, 3, 21
ANZ0001, 4
ANZ0001.sub, 5

coef.ocm, 6
data.frame, 13
deriv_link, 7
deviance.ocm, 7
extractAIC.ocm, 8
fitted.ocm, 9
formula.ocm, 10

get_gfun, 10
inv_link, 11
logLik.ocm, 12
model.frame.ocm, 13
model.matrix.ocm, 13
neck_pain, 14
nobs.ocm, 15

ocm, 2, 4, 8, 9, 11, 12, 15, 19–23
ordinalCont-package, 2
plot.ocm, 18
predict.ocm, 19
print.anova.ocm, 4, 20
print.ocm, 21, 22

summary.ocm, 21, 22
terms.ocm, 22
vcov.ocm, 23