Package ‘weightedScores’

October 22, 2015

Version 0.9.5.1

Date 2015-10-22

Title Weighted Scores Method for Regression Models with Dependent Data

Author A. K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk> and H. Joe <harry.joe@ubc.ca>

Maintainer Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>

Depends R (>= 2.0.0), mvtnorm, rootSolve

Description Has functions to implement the weighted scores method and CL1 information criteria as an intermediate step for variable/correlation selection for longitudinal categorical and count data in Nikoloulopoulos, Joe and Chaganty (2011, Biostatistics, 12: 653-665) and Nikoloulopoulos (2015a,2015b).

NeedsCompilation yes

Repository CRAN

Date/Publication 2015-10-22 22:14:54

R topics documented:

`approxbvncdf` *APPROXIMATION OF BIVARIATE STANDARD NORMAL DISTRIBUTION*

Description

Approximation of bivariate standard normal cumulative distribution function (Johnson and Kotz, 1972).

Usage

```
approxbvncdf(r,x1,x2,x1s,x2s,x1c,x2c,x1f,x2f,t1,t2)
```

Arguments

`r`	The correlation parameter of bivariate standard normal distribution.
`x1`	x_1, see details.
`x2`	x_2, see details.
`x1s`	x_1^2.
`x2s`	x_2^2.
`x1c`	x_1^3.
`x2c`	x_2^3.
`x1f`	x_1^4.
`x2f`	x_2^4.
`t1`	$\Phi(x_1)\Phi(x_2)$, where $\Phi(\cdot)$ is the cdf of univariate standard normal distribution.
`t2`	$\phi(x_1)\phi(x_2)$, where $\phi(\cdot)$ is the density of univariate stamdard normal distribution.

Details

The approximation for the bivariate normal cdf is from Johnson and Kotz (1972), page 118. Let $\Phi_2(x_1, x_2; \rho) = Pr(Z_1 \leq x_1,\, Z_2 \leq x_2)$, where (Z_1, Z_2) is bivariate normal with means 0, variances 1 and correlation ρ. An expansion, due to Pearson (1901), is

$$\Phi_2(x_1, x_2; \rho) = \Phi(x_1)\Phi(x_2) + \phi(x_1)\phi(x_2) \sum_{j=1}^{\infty} \rho^j \psi_j(x_1)\psi_j(x_2)/j!$$

where

$$\psi_j(z) = (-1)^{j-1} d^{j-1} \phi(z)/dz^{j-1}.$$

Since

$$\phi'(z) = -z\phi(z), \phi''(z) = (z^2-1)\phi(z), \phi'''(z) = [2z - z(z^2-1)]\phi(z) = (3z - z^3)\phi(z),$$

$$\phi^{(4)}(z) = [3 - 3z^2 - z(3z - z^3)]\phi(z) = (3 - 6z^2 + z^4)\phi(z)$$

we have

$$\Phi_2(x_1, x_2; \rho) = \Phi(x_1)\Phi(x_2) + \phi(x_1)\phi(x_2)[\rho + \rho^2 x_1 x_2/2 + \rho^3(x_1^2-1)(x_2^2-1)/6 + \rho^4(x_1^3-3x_1)(x_2^3-3x_2)/24$$

$$+\rho^5(x_1^4 - 6x_1^2 + 3)(x_2^4 - 6x_2^2 + 3)/120 + \cdots]$$

A good approximation is obtained truncating the series at ρ^3 term for $|\rho| \le 0.4$, and at ρ^5 term for $0.4 < |\rho| \le 0.7$. Higher order terms may be required for $|\rho| > 0.7$.

Value

An approximation of bivariate normal cumulative distribution function.

References

Johnson, N. L. and Kotz, S. (1972) *Continuous Multivariate Distributions.* Wiley, New York.

Pearson, K. (1901) Mathematical contributions to the theory of evolution-VII. On the correlation of characters not quantitatively measureable. *Philosophical Transactions of the Royal Society of London, Series A,* **195**, 1–47.

See Also

`scoreCov`

---

arthritis *Rheumatoid Arthritis Clinical Trial*

Description

Rheumatoid self-assessment scores for 302 patients, measured on a five-level ordinal response scale at three follow-up times.

Usage

```
data(arthritis)
```

Format

A data frame with 888 observations on the following 7 variables:

`id` Patient identifier variable.

`y` Self-assessment score of rheumatoid arthritis measured on a five-level ordinal response scale.

`sex` Coded as (1) for female and (2) for male.

`age` Recorded at the baseline.

`trt` Treatment group variable, coded as (1) for the placebo group and (2) for the drug group.

`baseline` Self-assessment score of rheumatoid arthritis at the baseline.

`time` Follow-up time recorded in months.

Source

Lipsitz, S.R. and Kim, K. and Zhao, L. (1994). Analysis of repeated categorical data using generalized estimating equations. *Statistics in Medicine*, **13**, 1149–1163.

Examples

```
data(arthritis)
```

`bcl` *BIVARIATE COMPOSITE LIKELIHOOD FOR MULTIVARIATE NORMAL COPULA WITH CATEGORICAL AND COUNT REGRESSION*

Description

Bivariate composite likelihood for multivariate normal copula with categorical and count regression.

Usage

```
bcl(r,b,gam,xdat,ydat,id,tvec,margmodel,corstr,link)
bcl.ord(r,b,gam,xdat,ydat,id,tvec,corstr,link)
```

Arguments

`r`	The vector of normal copula parameters.
`b`	The regression coefficients.
`gam`	The uinivariate parameters that are not regression coefficients. That is the parameter γ of negative binomial distribution or the q-dimensional vector of the univariate cutpoints of ordinal model. γ is NULL for Poisson and binary regression.

xdat	$(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept (except for ordinal regression where there is no intercept). This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.
ydat	$(y_1, y_2, \ldots, y_n)^\top$, where the response data vectors y_i, $i = 1, \ldots, n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of data vectors $y_i, i = 1, \ldots, n$ one from each unit which summarize all the data together in a single, long vector of length N.
id	An index for individuals or clusters.
tvec	A vector with the time indicator of individuals or clusters.
margmodel	Indicates the marginal model. Choices are “poisson” for Poisson, “bernoulli” for Bernoulli, and “nb1” , “nb2” for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).
corstr	Indicates the latent correlation structure of normal copula. Choices are “exch”, “ar”, and “unstr” for exchangeable, ar(1) and unstructured correlation structure, respectively.
link	The link function. Choices are “log” for the log link function, “logit” for the logit link function, and “probit” for the probit link function.

Details

The CL1 composite likelihood in Zhao and Joe (2005). That is the sum of bivariate marginal log-likelihoods.

`bcl.ord` is a variant of the code for ordinal (probit and logistic) regression.

Value

The negative bivariate composite likelihood for multivariate normal copula with Poisson or binary or negative binomial or ordinal regression.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Zhao, Y. and Joe, H. (2005) Composite likelihood estimation in multivariate data analysis. *The Canadian Journal of Statistics*, **33**, 335–356.

Cameron, A. C. and Trivedi, P. K. (1998) *Regression Analysis of Count Data.* Cambridge: Cambridge University Press.

See Also

`cl1`, `iee`

`childvisit` *Hospital Visit Data*

Description

The response vector consists of quarterly numbers of hospital visits over a one year period for a child age at four or younger, and three baseline covariates are age, sex, and maternal smoking status. The sample size of this study is $n = 73$ children.

Usage

```
data(childvisit)
```

Format

A data frame with 292 observations on the following 6 variables.

`id` The child ID.

`age` The age in months.

`sex` The sex indicator.

`matsmst` The maternal smoking status.

`quarter` The time indicator.

`hospvis` The response vector of quarterly numbers of hospital visits.

Source

Song P.X.K. (2007). Correlated Data Analysis: Modeling, Analytics, and Application. *Correlated Data Analysis: Modeling, Analytics, and Application.* Book Webpage/Supplementary Material. Springer, NY.

`cl1` *OPTIMIZATION ROUTINE FOR BIVARIATE COMPOSITE LIKELIHOOD FOR MVN COPULA*

Description

Optimization routine for bivariate composite likelihood for MVN copula.

Usage

```
cl1(b,gam,xdat,ydat,id,tvec,margmodel,corstr,link)
cl1.ord(b,gam,xdat,ydat,id,tvec,corstr,link)
```

Arguments

`b`	The regression coefficients.
`gam`	The uinivariate parameters that are not regression coefficients. That is the parameter γ of negative binomial distribution or the q-dimensional vector of the univariate cutpoints of ordinal model. γ is NULL for Poisson and binary regression.
`xdat`	$(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept (except for ordinal regression where there is no intercept). This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.
`ydat`	$(y_1, y_2, \ldots, y_n)^\top$, where the response data vectors y_i, $i = 1, \ldots, n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of data vectors $y_i, i = 1, \ldots, n$ one from each unit which summarize all the data together in a single, long vector of length N.
`id`	An index for individuals or clusters.
`tvec`	A vector with the time indicator of individuals or clusters.
`margmodel`	Indicates the marginal model. Choices are “poisson” for Poisson, “bernoulli” for Bernoulli, and “nb1” , “nb2” for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).
`corstr`	Indicates the latent correlation structure of normal copula. Choices are “exch”, “ar”, and “unstr” for exchangeable, ar(1) and unstructured correlation structure, respectively.
`link`	The link function. Choices are “log” for the log link function, “logit” for the logit link function, and “probit” for the probit link function.

Details

The CL1 composite likelihood method in Zhao and Joe (2005). The univariate parameters are estimated from the sum of univariate marginal log-likelihoods and then the dependence parameters are estimated from the sum of bivariate marginal log-likelihoods with the univariate parameters fixed from the first step.

Note that `bcl.ord` is a variant of the code for ordinal (probit and logistic) regression.

Value

A list containing the following components:

`minimum`	The negative value of the sum of bivariate marginal log-likelihoods at CL1 estimates.
`estimate`	The CL1 estimates.
`gradient`	The gradient at the estimated minimum of CL1.
`code`	An integer indicating why the optimization process terminated, same as in `nlm`.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Zhao, Y. and Joe, H. (2005) Composite likelihood estimation in multivariate data analysis. *The Canadian Journal of Statistics*, **33**, 335–356.

See Also

`bcl` `iee`

Examples

```
################################################################################
#                       NB1 regression for count data
################################################################################
################################################################################
#                       read and set up data set
################################################################################
data(childvisit)
# covariates
season1<-childvisit$q
season1[season1>1]<-0
xdat<-cbind(1,childvisit$sex,childvisit$age,childvisit$m,season1)
# response
ydat<-childvisit$hosp
#id
id<-childvisit$id
#time
```

```
tvec<-childvisit$q
################################################################################
#                       select the marginal model
################################################################################
margmodel="nb1"
################################################################################
#                       select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl1(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,margmodel,corstr)
cat("\nest.rho: CL1 estimates\n")
print(est.rho$e)
################################################################################
#                          Ordinal regression
################################################################################
################################################################################
#                       read and set up data set
################################################################################
## Not run:
data(arthritis)
nn=nrow(arthritis)
bas2<-bas3<-bas4<-bas5<-rep(0,nn)
bas2[arthritis$b==2]<-1
bas3[arthritis$b==3]<-1
bas4[arthritis$b==4]<-1
bas5[arthritis$b==5]<-1
t2<-t3<-rep(0,nn)
t2[arthritis$ti==3]<-1
t3[arthritis$ti==5]<-1
xdat=cbind(t2,t3,arthritis$trt,bas2,bas3,bas4,bas5,arthritis$age)
ydat=arthritis$y
id<-arthritis$id
#time
tvec<-arthritis$time
################################################################################
#                       select the link
################################################################################
link="logit"
################################################################################
#                       select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,ydat,link)
cat("\niest: IEE estimates\n")
```

```
print(c(i.est$reg,i.est$gam))
est.rho<-cl1.ord(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,corstr,link)
cat("\nest.rho: CL1 estimates\n")
print(est.rho$e)

## End(Not run)
```

`clic` *CL1 INFORMATION CRITERIA*

Description

Composite likelihood (CL1) information criteria.

Usage

```
clic1dePar(nbcl,r,b,gam,xdat,id,tvec,corstr,WtScMat,link,mvncmp)
clic(nbcl,r,b,gam,xdat,id,tvec,corstr,WtScMat,link,mvncmp)
```

Arguments

`nbcl`	The negative value of the sum of bivariate marginal log-likelihoods at CL1 estimates.
`r`	The CL1 estimates of the latent correlations.
`b`	The CL1 estimates of the regression coefficients.
`gam`	The CL1 estimates of the cutpoints.
`xdat`	$(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates. This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.
`id`	An index for individuals or clusters.
`tvec`	A vector with the time indicator of individuals or clusters.
`WtScMat`	A list containing the following components. omega: The array with the Ω_i, $i = 1, \ldots, n$ matrices; delta: The array with the Δ_i, $i = 1, \ldots, n$ matrices; X: The array with the X_i, $i = 1, \ldots, n$ matrices.
`corstr`	Indicates the latent correlation structure of normal copula. Choices are "exch", "ar", and "unstr" for exchangeable, ar(1) and unstructured correlation structure, respectively.
`link`	The link function. Choices are "logit" for the logit link function, and "probit" for the probit link function.
`mvncmp`	The method of computation of the MVN rectangle probabilities. Choices are 1 for `mvnapp` (faster), and 2 for `pmvnorm` (more accurate).

Details

First, consider the sum of univariate log-likelihoods

$$L_1 = \sum_{i=1}^{n}\sum_{j=1}^{d} \log f_1(y_{ij}; \nu_{ij}, \boldsymbol{\gamma}) = \sum_{i=1}^{n}\sum_{j=1}^{d} \ell_1(\nu_{ij}, \boldsymbol{\gamma},\, y_{ij}),$$

and then the sum of bivariate log-likelihoods

$$L_2 = \sum_{i=1}^{n}\sum_{j<k} \log f_2(y_{ij}, y_{ik}; \nu_{ij}, \nu_{ik}, \boldsymbol{\gamma}, \rho_{jk}) = \sum_{i=1}^{n}\sum_{j<k} \ell_2(\nu_{ij}, \nu_{ik}, \boldsymbol{\gamma}, \rho_{jk}; y_{ij}, y_{ik}),$$

where $\ell_2(\cdot) = \log f_2(\cdot)$ and

$$f_2(y_{ij}, y_{ik}; \nu_{ij}, \nu_{ik}, \boldsymbol{\gamma}, \rho_{jk}) = \int_{\Phi^{-1}[F_1(y_{ij}-1;\nu_{ij},\boldsymbol{\gamma})]}^{\Phi^{-1}[F_1(y_{ij};\nu_{ij},\boldsymbol{\gamma})]} \int_{\Phi^{-1}[F_1(y_{ik}-1;\nu_{ik}),\boldsymbol{\gamma}]}^{\Phi^{-1}[F_1(y_{ik};\nu_{ik},\boldsymbol{\gamma})]} \phi_2(z_j, z_d; \rho_{jk}) dz_j dz_k;$$

$\phi_2(\cdot;\rho)$ denotes the standard bivariate normal density with correlation ρ.

Let $\mathbf{a}^\top = (\boldsymbol{\beta}^\top, \boldsymbol{\gamma}^\top)$ be the column vector of all $r = p + q$ univariate parameters. Differentiating L_1 with respect to $\mathbf{a}$ leads to the independent estimating equations or univariate composite score functions:

$$\mathbf{g}_1 = \mathbf{g}_1(\mathbf{a}) = \frac{\partial L_1}{\partial \mathbf{a}} = \sum_{i=1}^{n} \mathbf{X}_i^\top \mathbf{s}_i^{(1)}(\mathbf{a}) = \mathbf{0},$$

Differentiating L_2 with respect to $\mathbf{R} = \big(\rho_{jk}, 1 \le j < k \le d\big)$ leads to the bivariate composite score functions (Zhao and Joe, 2005):

$$\mathbf{g}_2 = \frac{\partial L_2}{\partial \mathbf{R}} = \sum_{i=1}^{n} \mathbf{s}_i^{(2)}(\mathbf{a}, \mathbf{R}) = \sum_{i=1}^{n} \Big(\mathbf{s}_{i,jk}^{(2)}(\mathbf{a}, \rho_{jk}), 1 \le j < k \le d\Big) = \mathbf{0},$$

where $\mathbf{s}_i^{(2)}(\mathbf{a}, \mathbf{R}) = \frac{\partial \sum_{j<k} \ell_2(\nu_{ij},\nu_{ik},\boldsymbol{\gamma},\rho_{jk};y_{ij},y_{ik})}{\partial \mathbf{R}}$ and $\mathbf{s}_{i,jk}^{(2)}(\boldsymbol{\gamma}, \rho_{jk}) = \frac{\partial \ell_2(\nu_{ij},\nu_{ik},\boldsymbol{\gamma},\rho_{jk};y_{ij},y_{ik})}{\partial \rho_{jk}}$.

The CL1 estimates $\widetilde{\mathbf{a}}$ and $\widetilde{\mathbf{R}}$ of the discretized MVN model are obtained by solving the above CL1 estimating functions.

The asymptotic covariance matrix for the estimator that solves them, also known as the inverse Godambe (Godambe, 1991) information matrix, is

$$\mathbf{V} = (-\mathbf{H}_\mathbf{g})^{-1} \mathbf{J}_\mathbf{g} (-\mathbf{H}_\mathbf{g}^\top)^{-1},$$

where $\mathbf{g} = (\mathbf{g}_1, \mathbf{g}_2)^\top$. First set $\boldsymbol{\theta} = (\mathbf{a}, \mathbf{R})^\top$, then

$$-\mathbf{H}_\mathbf{g} = E\Big(\frac{\partial \mathbf{g}}{\partial \boldsymbol{\theta}}\Big) = \begin{bmatrix} E\Big(\frac{\partial \mathbf{g}_1}{\partial \mathbf{a}}\Big) & E\Big(\frac{\partial \mathbf{g}_1}{\partial \mathbf{R}}\Big) \\ E\Big(\frac{\partial \mathbf{g}_2}{\partial \mathbf{a}}\Big) & E\Big(\frac{\partial \mathbf{g}_2}{\partial \mathbf{R}}\Big) \end{bmatrix} = \begin{bmatrix} -\mathbf{H}_{\mathbf{g}_1} & \mathbf{0} \\ -\mathbf{H}_{\mathbf{g}_{2,1}} & -\mathbf{H}_{\mathbf{g}_2} \end{bmatrix},$$

where $-\mathbf{H}_{\mathbf{g}_1} = \sum_i^n \mathbf{X}_i^\top \boldsymbol{\Delta}_i^{(1)} \mathbf{X}_i$, $-\mathbf{H}_{\mathbf{g}_{2,1}} = \sum_i^n \boldsymbol{\Delta}_i^{(2,1)} \mathbf{X}_i$, and $-\mathbf{H}_{\mathbf{g}_2} = \sum_i^n \boldsymbol{\Delta}_i^{(2,2)}$.

The covariance matrix $\mathbf{J}_\mathbf{g}$ of the composite score functions $\mathbf{g}$ is given as below

$$\mathbf{J}_\mathbf{g} = \mathrm{Cov}(\mathbf{g}) = \begin{bmatrix} \mathrm{Cov}(\mathbf{g}_1) & \mathrm{Cov}(\mathbf{g}_1, \mathbf{g}_2) \\ \mathrm{Cov}(\mathbf{g}_2, \mathbf{g}_1) & \mathrm{Cov}(\mathbf{g}_2) \end{bmatrix} = \begin{bmatrix} \mathbf{J}_\mathbf{g}^{(1)} & \mathbf{J}_\mathbf{g}^{(1,2)} \\ \mathbf{J}_\mathbf{g}^{(2,1)} & \mathbf{J}_\mathbf{g}^{(2)} \end{bmatrix} = \sum_i \begin{bmatrix} \mathbf{X}_i^\top \boldsymbol{\Omega}_i^{(1)} \mathbf{X}_i & \mathbf{X}_i^\top \boldsymbol{\Omega}_i^{(1,2)} \\ \boldsymbol{\Omega}_i^{(2,1)} \mathbf{X}_i & \boldsymbol{\Omega}_i^{(2)} \end{bmatrix},$$

where

$$\begin{bmatrix} \boldsymbol{\Omega}_i^{(1)} & \boldsymbol{\Omega}_i^{(1,2)} \\ \boldsymbol{\Omega}_i^{(2,1)} & \boldsymbol{\Omega}_i^{(2)} \end{bmatrix} = \begin{bmatrix} \mathrm{Cov}\Big(\mathbf{s}_i^{(1)}(\mathbf{a})\Big) & \mathrm{Cov}\Big(\mathbf{s}_i^{(1)}(\mathbf{a}), \mathbf{s}_i^{(2)}(\mathbf{a},\mathbf{R})\Big) \\ \mathrm{Cov}\Big(\mathbf{s}_i^{(2)}(\mathbf{a},\mathbf{R}), \mathbf{s}_i^{(1)}(\mathbf{a})\Big) & \mathrm{Cov}\Big(\mathbf{s}_i^{(2)}(\mathbf{a},\mathbf{R})\Big) \end{bmatrix}.$$

To this end, the composite AIC (Varin and Vidoni, 2005) and BIC (Gao and Song, 2011) criteria have the forms:

$$\mathrm{CL1AIC} = -2L_2 + 2\mathrm{tr}\Big(\mathbf{J_g}\mathbf{H_g}^{-1}\Big),$$

$$\mathrm{CL1BIC} = -2L_2 + \log(n)\mathrm{tr}\Big(\mathbf{J_g}\mathbf{H_g}^{-1}\Big).$$

Value

A list containing the following components:

`AIC` The CL1AIC.

`BIC` The CL1BIC.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>

References

Gao, X. and Song, P.X.K. (2011). Composite likelihood EM algorithm with applications to multivariate hidden Markov model. *Statistica Sinica* **21**, 165–185.

Godambe, V. P. (1991) *Estimating Functions*. Oxford: Oxford University Press

Nikoloulopoulos, A.K. (2015a) Correlation structure and variable selection in generalized estimating equations via composite likelihood information criteria. *Arxiv e-prints*.

Nikoloulopoulos, A.K. (2015b) Weighted scores estimating equations for longitudinal ordinal data. *Arxiv e-prints*.

Varin, C. and Vidoni, P. (2005). A note on composite likelihood inference and model selection. *Biometrika* **92**, 519–528.

Zhao, Y. and Joe, H. (2005) Composite likelihood estimation in multivariate data analysis. *The Canadian Journal of Statistics*, **33**, 335–356.

See Also

`solvewtsc`, `wtsc.wrapper`

Examples

```
################################################################################
#                           Binary regression
################################################################################
################################################################################
#                      read and set up the data set
################################################################################
data(childvisit)
# covariates
season1<-childvisit$q
season1[season1>1]<-0
xdat<-cbind(childvisit$sex,childvisit$age,childvisit$m,season1)
# response
ydat<-childvisit$hosp
ydat[ydat>0]=1
ydat=2-ydat
#id
id<-childvisit$id
#time
tvec<-childvisit$q
################################################################################
#                      select the link
################################################################################
link="logit"
################################################################################
#                      select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                      perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,ydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
# est.rho<-cl1.ord(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,corstr,link)
# cat("\nest.rho: CL1 estimates\n")
# print(est.rho$e)
# [1] 0.1961
# cat("\nest.rho: negative CL1 log-likelihood\n")
# print(est.rho$m)
# [1] 576.5246
################################################################################
#                      obtain the fixed weight matrices
################################################################################
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=0.1961,xdat,ydat,id,
                       tvec,corstr,link)
################################################################################
#                      obtain the CL1 information criteria
################################################################################
out<-clic1dePar(nbcl=576.5246,r=0.1961,i.est$r,i.est$g,xdat,id,tvec,corstr,WtScMat,link)
```

`exchmvn` *Exchangeable (positive) multivariate normal*

Description

Rectangle probability and derivatives of positive exchangeable multivariate normal

Usage

```
exchmvn(lb, ub, rh, mu=0, scale=1, eps = 1.e-06)
exchmvn.deriv.margin(lb, ub, rh, k, ksign, eps = 1.e-06)
exchmvn.deriv.rho(lb, ub, rh, eps = 1.e-06)
```

Arguments

`lb`	vector of lower limits of integral/probability
`ub`	vector of upper limits of integral/probability
`rh`	correlation, rho
`mu`	mean vector
`scale`	standard deviation
`eps`	tolerance for numerical integration
`k`	margin for which derivative is to be taken, that is, deriv of exchmvn(lb,ub,rh) with respect to lb[k] or ub[k]; use exchmvn.deriv.rh for deriv of exchmvn(lb,ub,rh) with respect to rho
`ksign`	=-1 for deriv of exchmvn(lb,ub,rh) with respect to lb[k] =+1 for deriv of exchmvn(lb,ub,rh) with respect to ub[k]

Value

rectangle probability or a derivative

Author(s)

Harry Joe <`harry.joe@ubc.ca`>

References

Kotz S and Johnson NL (1972). *Continuous Multivariate Distributions*. Wiley, New York, page 48.

Examples

```
# The tests here show clearly what the function parameters are.
# step size for numerical derivatives (accuracy of exchmvn etc about 1.e-6)
heps = 1.e-4

cat("case 1: m=3\n")
m=3
a=c(-1,-1,-1)
b=c(2,1.5,1)
rh=.6
pr=exchmvn(a,b,rh)
cat("pr=exchmvn(avec,bvec,rh)=",pr,"\n")
cat("derivative wrt rho\n")
rh2=rh+heps
pr2=exchmvn(a,b,rh2)
drh.numerical= (pr2-pr)/heps
drh.analytic= exchmvn.deriv.rho(a,b,rh)
cat("   numerical: ", drh.numerical, ", analytic: ", drh.analytic,"\n")

cat("derivative wrt a_k,b_k, k=1,...,",m,"\n")
for(k in 1:m)
{ cat("  k=", k, " lower\n")
  a2=a
  a2[k]=a[k]+heps
  pr2=exchmvn(a2,b,rh)
  da.numerical = (pr2-pr)/heps
  da.analytic= exchmvn.deriv.margin(a,b,rh,k,-1)
  cat("   numerical: ", da.numerical, ", analytic: ", da.analytic,"\n")
  cat("  k=", k, " upper\n")
  b2=b
  b2[k]=b[k]+heps
  pr2=exchmvn(a,b2,rh)
  db.numerical = (pr2-pr)/heps
  db.analytic= exchmvn.deriv.margin(a,b,rh,k,1)
  cat("   numerical: ", db.numerical, ", analytic: ", db.analytic,"\n")
}


cat("\ncase 2: m=5\n")
m=5
a=rep(-1,m)
b=c(2,1.5,1,1.5,2)
rh=.6
pr=exchmvn(a,b,rh)
cat("pr=exchmvn(avec,bvec,rh)=",pr,"\n")
cat("derivative wrt rho\n")
rh2=rh+heps
pr2=exchmvn(a,b,rh2)
drh.numerical= (pr2-pr)/heps
drh.analytic= exchmvn.deriv.rho(a,b,rh)
cat("   numerical: ", drh.numerical, ", analytic: ", drh.analytic,"\n")
```

```
cat("derivative wrt a_k,b_k, k=1,...,",m,"\n")
for(k in 1:m)
{ cat("  k=", k, " lower\n")
  a2=a
  a2[k]=a[k]+heps
  pr2=exchmvn(a2,b,rh)
  da.numerical = (pr2-pr)/heps
  da.analytic= exchmvn.deriv.margin(a,b,rh,k,-1)
  cat("   numerical: ", da.numerical, ", analytic: ", da.analytic,"\n")
  cat("  k=", k, " upper\n")
  b2=b
  b2[k]=b[k]+heps
  pr2=exchmvn(a,b2,rh)
  db.numerical = (pr2-pr)/heps
  db.analytic= exchmvn.deriv.margin(a,b,rh,k,1)
  cat("   numerical: ", db.numerical, ", analytic: ", db.analytic,"\n")
}
```

---

godambe *INVERSE GODAMBE MATRIX*

Description

Inverse Godambe matrix.

Usage

```
godambe(param,WtScMat,xdat,ydat,id,tvec,margmodel,link)
godambe.ord(param,WtScMat,xdat,ydat,id,tvec,link)
```

Arguments

param	The weighted scores estimates of regression and not regression parameters.
WtScMat	A list containing the following components. omega: The array with the Ω_i, $i = 1,\ldots,n$ matrices; delta: The array with the Δ_i, $i = 1,\ldots,n$ matrices; X: The array with the X_i, $i = 1,\ldots,n$ matrices.
xdat	$(\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1,\ldots,n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept (except for ordinal regression where there is no intercept). This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.
ydat	$(y_1,y_2,\ldots,y_n)^\top$, where the response data vectors y_i, $i = 1,\ldots,n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of

data vectors $y_i, i = 1, \dots, n$ one from each unit which summarize all the data together in a single, long vector of length N.

`id` An index for individuals or clusters.

`tvec` A vector with the time indicator of individuals or clusters.

`margmodel` Indicates the marginal model. Choices are "poisson" for Poisson, "bernoulli" for Bernoulli, and "nb1" , "nb2" for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).

`link` The link function. Choices are "log" for the log link function, "logit" for the logit link function, and "probit" for the probit link function.

Details

If the $W_{i,\text{working}}$ are assumed fixed for the second stage of solving the weighted scores equations

$$g_1 = g_1(a) = \sum_{i=1}^{n} X_i^T \, W_{i,\text{working}}^{-1} \, s_i(a) = 0,$$

the asymptotic covariance matrix of the solution $\widehat{a}_1$ is

$$V_1 = (-D_{g_1})^{-1} M_{g_1} (-D_{g_1}^T)^{-1}$$

with

$$-D_{g_1} = \sum_{i=1}^{n} X_i^T W_{i,\text{working}}^{-1} \Delta_i X_i,$$

$$M_{g_1} = \sum_{i=1}^{n} X_i^T W_{i,\text{working}}^{-1} \Omega_{i,\text{true}} (W_{i,\text{working}}^{-1})^T X_i,$$

where $\Omega_{i,\text{true}}$ is the true covariance matrix of $s_i(a)$. The inverse of V_1 is known as Godambe information matrix (Godambe, 1991).

Note that `godambe.ord` is a variant of the code for ordinal (probit and logistic) regression.

Value

The inverse Godambe matrix.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Godambe, V. P. (1991) *Estimating Functions*. Oxford: Oxford University Press

Nikoloulopoulos, A.K., Joe, H. and Chaganty, N.R. (2011) Weighted scores method for regression models with dependent data. *Biostatistics*, **12**, 653–665.

Nikoloulopoulos, A.K. (2015a) Correlation structure and variable selection in generalized estimating equations via composite likelihood information criteria. *Arxiv e-prints*.

Nikoloulopoulos, A.K. (2015b) Weighted scores estimating equations for longitudinal ordinal data. *Arxiv e-prints*.

See Also

wtsc, solvewtsc, weightMat, wtsc.wrapper

Examples

```
################################################################################
#                            Poisson regression
################################################################################
################################################################################
#                       read and set up the data set
################################################################################
data(childvisit)
# covariates
season1<-childvisit$q
season1[season1>1]<-0
xdat<-cbind(1,childvisit$sex,childvisit$age,childvisit$m,season1)
# response
ydat<-childvisit$hosp
#id
id<-childvisit$id
#time
tvec<-childvisit$q
################################################################################
#                       select the marginal model
################################################################################
margmodel="poisson"
################################################################################
#                       select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl1(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,margmodel,corstr)
cat("\nest.rho: CL1 estimates\n")
print(est.rho$e)
################################################################################
#                       obtain the fixed weight matrices
################################################################################
WtScMat<-weightMat(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
xdat,ydat,id,tvec,margmodel,corstr)
################################################################################
#                       obtain the weighted scores estimates
################################################################################
# solve the nonlinear system of equations
ws<-solvewtsc(start=c(i.est$reg,i.est$gam),WtScMat,xdat,ydat,id,
tvec,margmodel,link)
cat("ws=parameter estimates\n")
print(ws$r)
```

```
################################################################################
#                      obtain the inverse Godambe matrix
################################################################################
acov<-godambe(ws$r,WtScMat,xdat,ydat,id,tvec,margmodel)
cat("\nacov: inverse Godambe matrix with W based on first-stage wt
matrices\n")
print(acov)
################################################################################
#                         Ordinal regression
################################################################################
################################################################################
#                      read and set up data set
################################################################################
## Not run:
data(arthritis)
nn=nrow(arthritis)
bas2<-bas3<-bas4<-bas5<-rep(0,nn)
bas2[arthritis$b==2]<-1
bas3[arthritis$b==3]<-1
bas4[arthritis$b==4]<-1
bas5[arthritis$b==5]<-1
t2<-t3<-rep(0,nn)
t2[arthritis$ti==3]<-1
t3[arthritis$ti==5]<-1
xdat=cbind(t2,t3,arthritis$trt,bas2,bas3,bas4,bas5,arthritis$age)
ydat=arthritis$y
id<-arthritis$id
#time
tvec<-arthritis$time
################################################################################
#                      select the link
################################################################################
link="probit"
################################################################################
#                      select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                      perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,ydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl1.ord(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,corstr,link)
cat("\nest.rho: CL1 estimates\n")
print(est.rho$e)
################################################################################
#                      obtain the fixed weight matrices
################################################################################
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,xdat,ydat,id,
tvec,corstr,link)
################################################################################
#                      obtain the weighted scores estimates
```

```
################################################################################
# solve the nonlinear system of equations
ws<-solvewtsc.ord(start=c(i.est$reg,i.est$gam),WtScMat,xdat,ydat,id,
tvec,link)
cat("ws=parameter estimates\n")
print(ws$r)
################################################################################
#                       obtain the inverse Godambe matrix
################################################################################
acov<-godambe.ord(ws$r,WtScMat,xdat,ydat,id,tvec,link)
cat("\nacov: inverse Godambe matrix with W based on first-stage wt
matrices\n")
print(acov)

## End(Not run)
```

---

iee *INDEPENDENT ESTIMATING EQUATIONS FOR BINARY AND COUNT REGRESSION*

Description

Independent estimating equations for binary and count regression.

Usage

```
iee(xdat,ydat,margmodel,link)
```

Arguments

`xdat`	$(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept. This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.
`ydat`	$(y_1, y_2, \ldots, y_n)^\top$, where the response data vectors y_i, $i = 1, \ldots, n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of data vectors $y_i, i = 1, \ldots, n$ one from each unit which summarize all the data together in a single, long vector of length N.
`margmodel`	Indicates the marginal model. Choices are "poisson" for Poisson, "bernoulli" for Bernoulli, and "nb1" , "nb2" for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).
`link`	The link function. Choices are "log" for the log link function, "logit" for the logit link function, and "probit" for the probit link function.

Details

The univariate parameters are estimated from the sum of univariate marginal log-likelihoods.

Value

A list containing the following components:

coef	The vector with the estimated regression parameters.
gam	The vector with the estimated parameters that are not regression parameters. This is NULL for Poisson and binary regression.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Cameron, A. C. and Trivedi, P. K. (1998) *Regression Analysis of Count Data.* Cambridge: Cambridge University Press.

See Also

marglik

Examples

```
################################################################################
#                       read and set up data set
################################################################################
data(toenail)
# covariates
xdat<-cbind(1,toenail$treat,toenail$time,toenail$treat*toenail$time)
# response
ydat<-toenail$y
#id
id<-toenail$id
#time
tvec<-toenail$time
################################################################################
#                       select the marginal model
################################################################################
margmodel="bernoulli"
################################################################################
#                       perform the IEE method
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
```

iee.ord *Maximum Likelihood for Ordinal Model*

Description

Maximum Likelihood for Ordinal Probit and Logit: Newton-Raphson minimization of negative log-likelihood.

Usage

```
iee.ord(x,y,link,iprint=0,maxiter=20,toler=1.e-6)
```

Arguments

`x`	vector or matrix of explanatory variables. Each row corresponds to an observation and each column to a variable. The number of rows of x should equal the number of data values in y, and there should be fewer columns than rows. Missing values are not allowed.
`y`	numeric vector containing the ordinal response. The values must be in the range 1,2,..., number of categories. Missing values are not allowed.
`link`	The link function.Choices are “logit” for the logit link function, and “probit” for the probit link function.
`iprint`	logical indicator, default is FALSE, for whether the iterations for numerical maximum likelihood should be printed.
`maxiter`	maximum number of Newton-Raphson iterations, default = 20.
`toler`	tolerance for convergence in Newton-Raphson iterations, default = 1.e-6.

Details

The ordinal probit model is similar to the ordinal logit model. The parameter estimate of ordinal logit are roughly 1.8 to 2 times those of ordinal probit.

Value

list of MLE of parameters and their associated standard errors, in the order cutpt1,...,cutpt(number of categ-1),b1,...b(number of covariates).

`negloglik`	value of negative log-likelihood, evaluated at MLE
`gam`	MLE of ordered cutpoint parameters
`reg`	MLE of regression parameters
`cov`	estimated covariance matrix of the parameters

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Anderson, J.A. and Pemberton, J.D. (1985). The grouped continuous model for multivariate ordered categorical variables and covariate adjustment. *Biometrics*, **41**, 875–885.

Examples

```
################################################################################
#                         Ordinal regression
################################################################################
################################################################################
#                      read and set up data set
################################################################################
data(arthritis)
nn=nrow(arthritis)
bas2<-bas3<-bas4<-bas5<-rep(0,nn)
bas2[arthritis$b==2]<-1
bas3[arthritis$b==3]<-1
bas4[arthritis$b==4]<-1
bas5[arthritis$b==5]<-1
t2<-t3<-rep(0,nn)
t2[arthritis$ti==3]<-1
t3[arthritis$ti==5]<-1
xdat=cbind(t2,t3,arthritis$trt,bas2,bas3,bas4,bas5,arthritis$age)
ydat=arthritis$y
################################################################################
#                      select the link
################################################################################
link="probit"
################################################################################
i.est<- iee.ord(xdat,ydat,link)
print(i.est)
```

---

marglik — *NEGATIVE LOG-LIKELIHOOD ASSUMING INDEPEDENCE WITHIN CLUSTERS*

Description

Negative log-likelihood assuming independence within clusters.

Usage

```
marglik(param,xdat,ydat,margmodel,link)
```

Arguments

param	The vector of regression and not regression parameters.

`xdat`	$(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept. This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^{n} j_i$ is the total number of observations from all units.
`ydat`	$(y_1, y_2, \ldots, y_n)^\top$, where the response data vectors y_i, $i = 1, \ldots, n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^{n} j_i$. The ydat are the collection of data vectors $y_i, i = 1, \ldots, n$ one from each unit which summarize all the data together in a single, long vector of length N.
`margmodel`	Indicates the marginal model. Choices are "poisson" for Poisson, "bernoulli" for Bernoulli, and "nb1" , "nb2" for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).
`link`	The link function. Choices are "log" for the log link function, "logit" for the logit link function, and "probit" for the probit link function.

Details

The negative sum of univariate marginal log-likelihoods.

Value

Minus log-likelihood assuming independence.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Cameron, A. C. and Trivedi, P. K. (1998) *Regression Analysis of Count Data.* Cambridge: Cambridge University Press.

See Also

`iee`

`margmodel` *DENSITY AND CDF OF THE UNIVARIATE MARGINAL DISTRIBUTION*

Description

Density and cdf of the univariate marginal distribution.

Usage

```
dmargmodel(y,mu,gam,invgam,margmodel)
pmargmodel(y,mu,gam,invgam,margmodel)
dmargmodel.ord(y,mu,gam,link)
pmargmodel.ord(y,mu,gam,link)
```

Arguments

`y`	Vector of (non-negative integer) quantiles.
`mu`	The parameter μ of the univariate distribution.
`gam`	The parameter(s) γ that are not regression parameters. γ is NULL for Poisson and Bernoulli distribution.
`invgam`	The inverse of parameter γ of negative binomial distribution.
`margmodel`	Indicates the marginal model. Choices are "poisson" for Poisson, "bernoulli" for Bernoulli, and "nb1" , "nb2" for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998). See details.
`link`	The link function. Choices are "logit" for the logit link function, and "probit" for the probit link function.

Details

Negative binomial distribution NB(τ, ξ) allows for overdispersion and its probability mass function (pmf) is given by

$$f(y;\tau,\xi) = \frac{\Gamma(\tau+y)}{\Gamma(\tau)\; y!} \frac{\xi^y}{(1+\xi)^{\tau+y}}, \quad y\; = 0,1,2,\ldots,\tau>0,\; \xi>0,$$

with mean $\mu = \tau\,\xi = \exp(\beta^T x)$ and variance $\tau\,\xi\,(1+\xi)$.

Cameron and Trivedi (1998) present the NBk parametrization where $\tau = \mu^{2-k}\gamma^{-1}$ and $\xi = \mu^{k-1}\gamma$, $1 \le k \le 2$. In this function we use the NB1 parametrization $(\tau = \mu\gamma^{-1},\; \xi = \gamma)$, and the NB2 parametrization $(\tau = \gamma^{-1},\; \xi = \mu\gamma)$; the latter is the same as in Lawless (1987).

`margmodel.ord` is a variant of the code for ordinal (probit and logistic) model. In this case, the response Y is assumed to have density

$$f_1(y;\nu,\gamma) = F(\alpha_y + \nu) - F(\alpha_{y-1} + \nu),$$

where $\nu = x\beta$ is a function of x and the p-dimensional regression vector β, and $\gamma = (\alpha_1, \ldots, \alpha_{K-1})$ is the q-dimensional vector of the univariate cutpoints ($q = K - 1$). Note that F normal leads to the probit model and F logistic leads to the cumulative logit model for ordinal response.

Value

The density and cdf of the univariate distribution.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Cameron, A. C. and Trivedi, P. K. (1998) *Regression Analysis of Count Data.* Cambridge: Cambridge University Press.

Lawless, J. F. (1987) Negative binomial and mixed Poisson regression. *The Canadian Journal of Statistics*, **15**, 209–225.

Nikoloulopoulos, A.K. (2015b) Weighted scores estimating equations for longitudinal ordinal data. *Arxiv e-prints*.

Examples

```
y<-3
gam<-2.5
invgam<-1/2.5
mu<-0.5
margmodel<-"nb2"
dmargmodel(y,mu,gam,invgam,margmodel)
pmargmodel(y,mu,gam,invgam,margmodel)
link="probit"
dmargmodel.ord(y,mu,gam,link)
pmargmodel.ord(y,mu,gam,link)
```

`mvn.deriv` *Derivatives of Multivariate Normal Rectangle Probabilities*

Description

Derivatives of Multivariate Normal Rectangle Probabilities based on Approximations

Usage

```
mvn.deriv.margin(lb,ub,mu,sigma,k,ksign,type=1,eps=1.e-05,nsim=0)
mvn.deriv.rho(lb,ub,mu,sigma,j1,k1,type=1,eps=1.e-05,nsim=0)
```

Arguments

lb	vector of lower limits of integral/probability
ub	vector of upper limits of integral/probability
mu	mean vector
sigma	covariance matrix, it is assumed to be positive-definite
type	indicator, type=1 refers to the first order approximation, type=2 is the second order approximation.
eps	accuracy/tolerance for bivariate marginal rectangle probabilities
nsim	an optional integer if random permutations are used in the approximation for dimension >=6; nsim=2000 recommended for dim>=6
k	margin for which derivative is to be taken, that is, deriv of mvnapp(lb,ub,mu,sigma) with respect to lb[k] or ub[k];
ksign	=-1 for deriv of mvnapp(lb,ub,mu,sigma) with respect to lb[k] =+1 for deriv of mvnapp(lb,ub,mu,sigma) with respect to ub[k]
j1	correlation for which derivative is to be taken, that is, deriv of mvnapp(lb,ub,mu,sigma) with respect to rho[j1,k1], where rho is a correlation corresponding to sigma
k1	See above explanation with j1

Value

derivative with respect to margin lb[k], ub[k], or correlation rho[j1][k1] corresponding to sigmamatrix

Author(s)

Harry Joe <harry.joe@ubc.ca>

References

Joe, H (1995). Approximations to multivariate normal rectangle probabilities based on conditional expectations. *Journal of American Statistical Association*, **90**, 957–964.

See Also

mvnapp

Examples

```
# step size for numerical derivatives (accuracy of mvnapp etc may be about 1.e-4 to 1.e-5)
heps = 1.e-3

cat("compare numerical and analytical derivatives based on mvnapp\n")

cat("\ncase 1: dim=3\n");
m=3
mu=rep(0,m)
a=c(0,0,0)
```

```
b=c(1,1.5,2)
rr=matrix(c(1,.3,.3,.3,1,.4,.3,.4,1),m,m)
pr=mvnapp(a,b,mu,rr)$pr
# not checking ifail returned from mvnapp
cat("pr=mvnapp(avec,bvec,mu=0,sigma=corrmat)=",pr,"\n")

cat("derivative wrt a_k,b_k, k=1,...,",m,"\n")
for(k in 1:m)
{ cat("  k=", k, " lower\n")
  a2=a
  a2[k]=a[k]+heps
  pr2=mvnapp(a2,b,mu,rr)$pr
  da.numerical = (pr2-pr)/heps
  da.analytic= mvn.deriv.margin(a,b,mu,rr,k,-1)$deriv
  cat("   numerical: ", da.numerical, ", analytic: ", da.analytic,"\n")
  cat("  k=", k, " upper\n")
  b2=b
  b2[k]=b[k]+heps
  pr2=mvnapp(a,b2,mu,rr)$pr
  db.numerical = (pr2-pr)/heps
  db.analytic= mvn.deriv.margin(a,b,mu,rr,k,1)$deriv
  cat("   numerical: ", db.numerical, ", analytic: ", db.analytic,"\n")
}

cat("derivative wrt rho(j,k)\n")
for(j in 1:(m-1))
{ for(k in (j+1):m)
  { cat("  (j,k)=", j,k,"\n")
    rr2=rr
    rr2[j,k]=rr[j,k]+heps
    rr2[k,j]=rr[k,j]+heps
    pr2=mvnapp(a,b,mu,rr2)$pr
    drh.numerical = (pr2-pr)/heps
    drh.analytic= mvn.deriv.rho(a,b,mu,rr2,j,k)$deriv
    cat("   numerical: ", drh.numerical, ", analytic: ", drh.analytic,"\n")
  }
}

#============================================

cat("\ncase 2: dim=5\n");
m=5
mu=rep(0,m)
a=c(0,0,0,-1,-1)
b=c(1,1.5,2,2,2)
rr=matrix(c(1,.3,.3,.3,.4, .3,1,.4,.4,.4,  .3,.4,1,.4,.4,
  .3,.4,.4,1,.4, .4,.4,.4,.4,1),m,m)
pr=mvnapp(a,b,mu,rr)$pr
# not checking ifail returned from mvnapp
cat("pr=mvnapp(avec,bvec,mu=0,sigma=corrmat)=",pr,"\n")

cat("derivative wrt a_k,b_k, k=1,...,",m,"\n")
for(k in 1:m)
```

```
{ cat("  k=", k, " lower\n")
  a2=a
  a2[k]=a[k]+heps
  pr2=mvnapp(a2,b,mu,rr)$pr
  da.numerical = (pr2-pr)/heps
  da.analytic= mvn.deriv.margin(a,b,mu,rr,k,-1)$deriv
  cat("   numerical: ", da.numerical, ", analytic: ", da.analytic,"\n")
  cat("  k=", k, " upper\n")
  b2=b
  b2[k]=b[k]+heps
  pr2=mvnapp(a,b2,mu,rr)$pr
  db.numerical = (pr2-pr)/heps
  db.analytic= mvn.deriv.margin(a,b,mu,rr,k,1)$deriv
  cat("   numerical: ", db.numerical, ", analytic: ", db.analytic,"\n")
}


cat("derivative wrt rho(j,k): first order approx\n")
for(j in 1:(m-1))
{ for(k in (j+1):m)
  { cat("  (j,k)=", j,k,"\n")
    rr2=rr
    rr2[j,k]=rr[j,k]+heps
    rr2[k,j]=rr[k,j]+heps
    pr2=mvnapp(a,b,mu,rr2)$pr
    drh.numerical = (pr2-pr)/heps
    drh.analytic= mvn.deriv.rho(a,b,mu,rr2,j,k)$deriv
    cat("   numerical: ", drh.numerical, ", analytic: ", drh.analytic,"\n")
  }
}

cat("\nsecond order approx\n")
pr=mvnapp(a,b,mu,rr,type=2)$pr
cat("pr=mvnapp(avec,bvec,mu=0,sigma=corrmat,type=2)=",pr,"\n")

cat("derivative wrt a_k,b_k, k=1,...,",m,"\n")
for(k in 1:m)
{ cat("  k=", k, " lower\n")
  a2=a
  a2[k]=a[k]+heps
  pr2=mvnapp(a2,b,mu,rr,type=2)$pr
  da.numerical = (pr2-pr)/heps
  da.analytic= mvn.deriv.margin(a,b,mu,rr,k,-1,type=2)$deriv
  cat("   numerical: ", da.numerical, ", analytic: ", da.analytic,"\n")
  cat("  k=", k, " upper\n")
  b2=b
  b2[k]=b[k]+heps
  pr2=mvnapp(a,b2,mu,rr,type=2)$pr
  db.numerical = (pr2-pr)/heps
  db.analytic= mvn.deriv.margin(a,b,mu,rr,k,1,type=2)$deriv
  cat("   numerical: ", db.numerical, ", analytic: ", db.analytic,"\n")
}
```

```
cat("derivative wrt rho(j,k): second order approx\n")
for(j in 1:(m-1))
{ for(k in (j+1):m)
  { cat("  (j,k)=", j,k,"\n")
    rr2=rr
    rr2[j,k]=rr[j,k]+heps
    rr2[k,j]=rr[k,j]+heps
    pr2=mvnapp(a,b,mu,rr2,type=2)$pr
    drh.numerical = (pr2-pr)/heps
    drh.analytic= mvn.deriv.rho(a,b,mu,rr2,j,k,type=2)$deriv
    cat("   numerical: ", drh.numerical, ", analytic: ", drh.analytic,"\n")
  }
}
```

`mvnapp` *MVN Rectangle Probabilities*

Description

Approximation to multivariate normal rectangle probabilities using methods in Joe (1995, JASA)

Usage

```
mvnapp(lb,ub,mu,sigma,type=1,eps=1.e-05,nsim=0)
```

Arguments

`lb`	vector of lower limits of integral/probability
`ub`	vector of upper limits of integral/probability
`mu`	mean vector
`sigma`	covariance matrix, it is assumed to be positive-definite
`type`	indicator, type=1 refers to the first order approximation, type=2 is the second order approximation.
`eps`	accuracy/tolerance for bivariate marginal rectangle probabilities
`nsim`	an optional integer if random permutations are used in the approximation for dimension >=6; nsim=2000 recommended for dim>=6

Value

`prob`	rectangle probability with approximation
`esterr`	indicator of accuracy in the approximation
`ifail`	= 0 if no problems >= 1 if problems from using Schervish's code in dimensions 2 to 4.

Author(s)

Harry Joe <`harry.joe@ubc.ca`>

References

Joe, H (1995). Approximations to multivariate normal rectangle probabilities based on conditional expectations. *Journal of American Statistical Association*, **90**, 957–964.

Examples

```
m<-2
rh<-0.5
a<-c(-1,-1)
b<-c(1,1)
mu<-rep(0,m)
s<-matrix(c(1,rh,rh,1),2,2)
print(mvnapp(a,b,mu,s))
print(mvnapp(a,b,mu,s,type=2))
print(mvnapp(a,b,mu,s,type=2,nsim=3))

m<-3
rh<-0.3
a<-c(-1,-1,-2)
b<-c(1,1,.5)
mu<-rep(0,m)
s<-matrix(c(1,.5,.6,.5,1,.7,.6,.7,1),3,3)
print(mvnapp(a,b,mu,s))
print(mvnapp(a,b,mu,s,type=2))
print(mvnapp(a,b,mu,s,type=2,nsim=3))

m<-4
rh<- -0.1
a<-c(-1,-2.5,-2,-1.5)
b<-c(1.68,1.11,.5,.25)
mu<-rep(0,m)
s<-matrix(c(1,.5,.3,.4,.5,1,.5,.4,.3,.5,1,.4,.4,.4,.4,1),m,m)
print(mvnapp(a,b,mu,s))
print(mvnapp(a,b,mu,s,type=2))
print(mvnapp(a,b,mu,s,type=2,nsim=3))

m<-5
rh<-.4
a<-rep(-1,m)
b<-rep(2,m)
mu<-rep(0,m)
s<-matrix(c(1,rh,rh,rh,rh,rh,1,rh,rh,rh,rh,rh,1,rh,rh,rh,rh,rh,1,
       rh,rh,rh,rh,rh,1),m,m)
print(mvnapp(a,b,mu,s))
print(mvnapp(a,b,mu,s,type=2))
print(mvnapp(a,b,mu,s,type=2,nsim=3))

m<-6
a<-c(-1,-1,-1,-1.5,-1,-2)
b<-rep(7,m)
mu<-rep(0,m)
s<-matrix(c(1,rh,rh,rh,rh,rh,rh,1,rh,rh,rh,rh,rh,rh,1,rh,rh,rh,rh,rh,rh,1,
```

```
        rh,rh,rh,rh,rh,rh,1,rh,rh,rh,rh,rh,rh,1),m,m)
print(mvnapp(a,b,mu,s))
print(mvnapp(a,b,mu,s,type=2))
print(mvnapp(a,b,mu,s,type=2,nsim=3))
```

`scoreCov` *COVARIANCE MATRIX OF THE UNIVARIATE SCORES*

Description

Covariance matrix of the univariates scores.

Usage

```
scoreCov(scnu,scgam,pmf,index,margmodel)
scoreCov.ord(scgam,pmf,index)
```

Arguments

`scnu`	The matrix of the score functions with respect to ν.
`scgam`	The matrix of the score functions with respect to γ.
`pmf`	The matrix of rectangle probabilities.
`index`	The bivariate pair.
`margmodel`	Indicates the marginal model. Choices are “poisson” for Poisson, “bernoulli” for Bernoulli, and “nb1” , “nb2” for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).

Details

The covariance matrix Ω_i of $s_i(a)$ computed from the fitted discretized MVN model with estimated parameters $\tilde{a}$, $\tilde{R}$.

Note that `scoreCov.ord` is a variant of the code for ordinal (probit and logistic) regression.

Value

Covariance matrix of the univariates scores Ω_i.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

See Also

approxbvncdf

solvewtsc — *SOLVING THE WEIGHTED SCORES EQUATIONS WITH INPUTS OF THE WEIGHT MATRICES AND THE DATA*

Description

Solving the weighted scores equations with inputs of the weight matrices and the data.

Usage

```
solvewtsc(start,WtScMat,xdat,ydat,id,tvec,margmodel,link)
solvewtsc.ord(start,WtScMat,xdat,ydat,id,tvec,link)
```

Arguments

`start` A starting value of the vector of regression and not regression parameters. The CL1 estimates of regression and not regression parameters is a good starting value.

`WtScMat` A list containing the following components. omega: The array with the Ω_i, $i = 1,\ldots,n$ matrices; delta: The array with the Δ_i, $i = 1,\ldots,n$ matrices; X: The array with the X_i, $i = 1,\ldots,n$ matrices.

`xdat` $(\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1,\ldots,n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept (except for ordinal regression where there is no intercept). This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.

`ydat` $(y_1,y_2,\ldots,y_n)^\top$, where the response data vectors y_i, $i = 1,\ldots,n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of data vectors $y_i, i = 1,\ldots,n$ one from each unit which summarize all the data together in a single, long vector of length N.

`id` An index for individuals or clusters.

`tvec` A vector with the time indicator of individuals or clusters.

`margmodel` Indicates the marginal model. Choices are “poisson” for Poisson, “bernoulli” for Bernoulli, and “nb1” , “nb2” for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).

`link` The link function. Choices are “log” for the log link function, “logit” for the logit link function, and “probit” for the probit link function.

Details

Obtain robust estimates $\hat{a}$ of the univariate parameters solving the weighted scores equation, $g_1 = g_1(a) = \sum_{i=1}^n X_i^T W_{i,\text{working}}^{-1} s_i(a) = 0$, where $W_{i,\text{working}}^{-1} = \Delta_i \Omega_{i,\text{working}}^{-1} = \Delta_i(\tilde{a})\Omega_i(\tilde{a},\tilde{R})^{-1}$ is based on the covariance matrix of $s_i(a)$ computed from the fitted discretized MVN model with estimated parameters $\tilde{a}, \tilde{R}$. A reliable non-linear system solver is used.

Note that `solvewtsc.ord` is a variant of the code for ordinal (probit and logistic) regression.

Value

A list containing the following components:

`root`	The weighted scores estimates.
`f.root`	The value of the wtsc function evaluated at the root.
`iter`	The number of iterations used.
`estim.precis`	The estimated precision for root.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Nikoloulopoulos, A.K., Joe, H. and Chaganty, N.R. (2011) Weighted scores method for regression models with dependent data. *Biostatistics*, **12**, 653–665.

Nikoloulopoulos, A.K. (2015a) Correlation structure and variable selection in generalized estimating equations via composite likelihood information criteria. *Arxiv e-prints*.

Nikoloulopoulos, A.K. (2015b) Weighted scores estimating equations for longitudinal ordinal data. *Arxiv e-prints*.

See Also

`wtsc`, `weightMat`, `godambe`, `wtsc.wrapper`

Examples

```
################################################################################
#                            NB2 regression
################################################################################
################################################################################
#                       read and set up the data set
################################################################################
data(childvisit)
# covariates
season1<-childvisit$q
season1[season1>1]<-0
xdat<-cbind(1,childvisit$sex,childvisit$age,childvisit$m,season1)
# response
ydat<-childvisit$hosp
```

```
#id
id<-childvisit$id
#time
tvec<-childvisit$q
################################################################################
#                       select the marginal model
################################################################################
margmodel="nb2"
################################################################################
#                       select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl1(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,margmodel,corstr,link)
cat("\nest.rho: CL1 estimates\n")
print(est.rho$e)
################################################################################
#                       obtain the fixed weight matrices
################################################################################
WtScMat<-weightMat(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
xdat,ydat,id,tvec,margmodel,corstr)
################################################################################
#                       obtain the weighted scores estimates
################################################################################
# solve the nonlinear system of equations
ws<-solvewtsc(start=c(i.est$reg,i.est$gam),WtScMat,xdat,ydat,id,
tvec,margmodel,link)
cat("ws=parameter estimates\n")
print(ws$r)
################################################################################
#                          Ordinal regression
################################################################################
################################################################################
#                       read and set up data set
################################################################################
## Not run:
data(arthritis)
nn=nrow(arthritis)
bas2<-bas3<-bas4<-bas5<-rep(0,nn)
bas2[arthritis$b==2]<-1
bas3[arthritis$b==3]<-1
bas4[arthritis$b==4]<-1
bas5[arthritis$b==5]<-1
t2<-t3<-rep(0,nn)
t2[arthritis$ti==3]<-1
t3[arthritis$ti==5]<-1
xdat=cbind(t2,t3,arthritis$trt,bas2,bas3,bas4,bas5,arthritis$age)
ydat=arthritis$y
```

```
id<-arthritis$id
#time
tvec<-arthritis$time
################################################################################
#                       select the link
################################################################################
link="probit"
################################################################################
#                       select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,ydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl1.ord(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,corstr,link)
cat("\nest.rho: CL1 estimates\n")
print(est.rho$e)
################################################################################
#                       obtain the fixed weight matrices
################################################################################
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,xdat,ydat,id,
tvec,corstr,link)
################################################################################
#                       obtain the weighted scores estimates
################################################################################
# solve the nonlinear system of equations
ws<-solvewtsc.ord(start=c(i.est$reg,i.est$gam),WtScMat,xdat,ydat,id,
tvec,link)
cat("ws=parameter estimates\n")
print(ws$r)

## End(Not run)
```

`toenail` *The toenail infection data*

Description

The data consist of up to seven binary observations on each of 294 subjects who had been randomly assigned to one of two treatment groups. The observations, taken at regularly scheduled time points, are coded as 1 if the subject's infection was severe and 0 otherwise. The interest is to investigate if the two treatments differ and if the percentage of severe infections decreased over time.

Usage

```
data(toenail)
```

Format

A data frame with 1568 observations on the following 4 variables.

`idnum` The index for individuals.

`treatn` The treatment binary covariate.

`y` The subject's infection binary response.

`time` The time indicator.

Source

Molenberghs, G., Verbeke, G., 2005. *Models for Discrete Longitudinal Data*. Springer, NY.

`weightMat` *WEIGHT MATRICES FOR THE ESTIMATING EQUATIONS*

Description

Weight matrices for the estimating equations.

Usage

```
weightMat(b,gam,rh,xdat,ydat,id,tvec,margmodel,corstr,link)
weightMat.ord(b,gam,rh,xdat,ydat,id,tvec,corstr,link)
```

Arguments

`b` The regression coefficients.

`gam` The uinivariate parameters that are not regression coefficients. That is the parameter γ of negative binomial distribution or the q-dimensional vector of the univariate cutpoints of ordinal model. γ is NULL for Poisson and binary regression.

`rh` The vector of normal copula parameters.

`xdat` $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept (except for ordinal regression where there is no intercept). This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.

`ydat` $(y_1, y_2, \ldots, y_n)^\top$, where the reponse data vectors $y_i, i = 1, \ldots, n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of data vectors $y_i, i = 1, \ldots, n$ one from each unit which summarize all the data together in a single, long vector of length N.

`id`	An index for individuals or clusters.
`tvec`	A vector with the time indicator of individuals or clusters.
`margmodel`	Indicates the marginal model. Choices are "poisson" for Poisson, "bernoulli" for Bernoulli, and "nb1" , "nb2" for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).
`corstr`	Indicates the latent correlation structure of normal copula. Choices are "exch", "ar", and "unstr" for exchangeable, ar(1) and unstrucutred correlation structure, respectively.
`link`	The link function. Choices are "log" for the log link function, "logit" for the logit link function, and "probit" for the probit link function.

Details

The fixed weight matrices $W_{i,\text{working}}$ based on a working discretized MVN, of the weighted scores equations in Nikoloulopoulos et al. (2011)

$$g_1 = g_1(a) = \sum_{i=1}^{n} X_i^T \, W_{i,\text{working}}^{-1} \, s_i(a) = 0,$$

where $W_{i,\text{working}}^{-1} = \Delta_i \Omega_{i,\text{working}}^{-1} = \Delta_i(\tilde{a})\Omega_i(\tilde{a}, \tilde{R})^{-1}$ is based on the covariance matrix of $s_i(a)$ computed from the fitted discretized MVN model with estimated parameters $\tilde{a}$, $\tilde{R}$.

Note that `weightMat.ord` is a variant of the code for ordinal (probit and logistic) regression.

Value

A list containing the following components:

`omega`	The array with the $\Omega_i,\ i = 1, \ldots, n$ matrices.
`delta`	The array with the $\Delta_i,\ i = 1, \ldots, n$ matrices.
`X`	The array with the $X_i,\ i = 1, \ldots, n$ matrices.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Nikoloulopoulos, A.K., Joe, H. and Chaganty, N.R. (2011) Weighted scores method for regression models with dependent data. *Biostatistics*, **12**, 653–665.

Nikoloulopoulos, A.K. (2015a) Correlation structure and variable selection in generalized estimating equations via composite likelihood information criteria. *Arxiv e-prints*.

Nikoloulopoulos, A.K. (2015b) Weighted scores estimating equations for longitudinal ordinal data. *Arxiv e-prints*.

See Also

`wtsc`, `solvewtsc`, `godambe`, `wtsc.wrapper`

Examples

```
################################################################################
#                             binary regression
################################################################################
################################################################################
#                       read and set up the data set
################################################################################
data(toenail)
xdat<-cbind(1,toenail$treat,toenail$time,toenail$treat*toenail$time)
# response
ydat<-toenail$y
#id
id<-toenail$id
#time
tvec<-toenail$time
################################################################################
#                       select the marginal model
################################################################################
margmodel="bernoulli"
link="probit"
################################################################################
#                       select the  correlation structure
################################################################################
corstr="ar"
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee(xdat,ydat,margmodel,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
# est.rho<-cl1(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,margmodel,corstr,link)
# cat("\nest.rho: CL1 estimates\n")
# print(est.rho$e)
# [1] 0.8941659
################################################################################
#                       obtain the fixed weight matrices
################################################################################
WtScMat<-weightMat(b=i.est$reg,gam=i.est$gam,rh=0.8941659,
xdat,ydat,id,tvec,margmodel,corstr,link)
################################################################################
#                          Ordinal regression
################################################################################
################################################################################
#                       read and set up data set
################################################################################
## Not run:
data(arthritis)
nn=nrow(arthritis)
bas2<-bas3<-bas4<-bas5<-rep(0,nn)
bas2[arthritis$b==2]<-1
bas3[arthritis$b==3]<-1
bas4[arthritis$b==4]<-1
```

```
bas5[arthritis$b==5]<-1
t2<-t3<-rep(0,nn)
t2[arthritis$ti==3]<-1
t3[arthritis$ti==5]<-1
xdat=cbind(t2,t3,arthritis$trt,bas2,bas3,bas4,bas5,arthritis$age)
ydat=arthritis$y
id<-arthritis$id
#time
tvec<-arthritis$time
################################################################################
#                       select the link
################################################################################
link="probit"
################################################################################
#                       select the  correlation structure
################################################################################
corstr="exch"
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,ydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl1.ord(b=i.est$reg,gam=i.est$gam,xdat,ydat,id,tvec,corstr,link)
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,xdat,ydat,id,tvec,corstr,link)

## End(Not run)
```

`wtsc` *THE WEIGHTED SCORES EQUATIONS WITH INPUTS OF THE WEIGHT MATRICES AND THE DATA*

Description

The weighted scores equations with inputs of the weight matrices and the data.

Usage

```
wtsc(param,WtScMat,xdat,ydat,id,tvec,margmodel,link)
wtsc.ord(param,WtScMat,xdat,ydat,id,tvec,link)
```

Arguments

`param`	The vector of regression and not regression parameters.
`WtScMat`	A list containing the following components. omega: The array with the Ω_i, $i = 1, \ldots, n$ matrices; delta: The array with the Δ_i, $i = 1, \ldots, n$ matrices; X: The array with the X_i, $i = 1, \ldots, n$ matrices.

xdat — $(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept (except for ordinal regression where there is no intercept). This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.

ydat — $(y_1, y_2, \ldots, y_n)^\top$, where the response data vectors y_i, $i = 1, \ldots, n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of data vectors $y_i, i = 1, \ldots, n$ one from each unit which summarize all the data together in a single, long vector of length N.

id — An index for individuals or clusters.

tvec — A vector with the time indicator of individuals or clusters.

margmodel — Indicates the marginal model. Choices are “poisson” for Poisson, “bernoulli” for Bernoulli, and “nb1” , “nb2” for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).

link — The link function. Choices are “log” for the log link function, “logit” for the logit link function, and “probit” for the probit link function.

Details

The weighted scores estimating equations, with $W_{i,\text{working}}$ based on a working discretized MVN, have the form:

$$g_1 = g_1(a) = \sum_{i=1}^n X_i^T \, W_{i,\text{working}}^{-1} \, s_i(a) = 0,$$

where $W_{i,\text{working}}^{-1} = \Delta_i \Omega_{i,\text{working}}^{-1} = \Delta_i(\tilde{a})\Omega_i(\tilde{a}, \tilde{R})^{-1}$ is based on the covariance matrix of $s_i(a)$ computed from the fitted discretized MVN model with estimated parameters $\tilde{a}$, $\tilde{R}$.

Note that `wtsc.ord` is a variant of the code for ordinal (probit and logistic) regression.

Value

The weighted scores equations.

References

Nikoloulopoulos, A.K., Joe, H. and Chaganty, N.R. (2011) Weighted scores method for regression models with dependent data. *Biostatistics*, **12**, 653–665.

Nikoloulopoulos, A.K. (2015a) Correlation structure and variable selection in generalized estimating equations via composite likelihood information criteria. *Arxiv e-prints*.

Nikoloulopoulos, A.K. (2015b) Weighted scores estimating equations for longitudinal ordinal data. *Arxiv e-prints*.

See Also

`solvewtsc`, `weightMat`, `godambe`, `wtsc.wrapper` wtsc.wrapper *THE WEIGHTED SCORES METHOD WITH INPUTS OF THE DATA*

Description

The weighted scores method with inputs of the data.

Usage

```
wtsc.wrapper(xdat,ydat,id,tvec,margmodel,corstr,link,iprint=FALSE)
wtsc.ord.wrapper(xdat,ydat,id,tvec,corstr,link,iprint=FALSE)
```

Arguments

xdat	$(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^\top$, where the matrix $\mathbf{x}_i$, $i = 1, \ldots, n$ for a given unit will depend on the times of observation for that unit (j_i) and will have number of rows j_i, each row corresponding to one of the j_i elements of y_i and p columns where p is the number of covariates including the unit first column to account for the intercept (except for ordinal regression where there is no intercept). This xdat matrix is of dimension $(N \times p)$, where $N = \sum_{i=1}^n j_i$ is the total number of observations from all units.
ydat	$(y_1, y_2, \ldots, y_n)^\top$, where the response data vectors y_i, $i = 1, \ldots, n$ are of possibly different lengths for different units. In particular, we now have that y_i is $(j_i \times 1)$, where j_i is the number of observations on unit i. The total number of observations from all units is $N = \sum_{i=1}^n j_i$. The ydat are the collection of data vectors $y_i, i = 1, \ldots, n$ one from each unit which summarize all the data together in a single, long vector of length N.
id	An index for individuals or clusters.
tvec	A vector with the time indicator of individuals or clusters.
margmodel	Indicates the marginal model. Choices are “poisson” for Poisson, “bernoulli” for Bernoulli, and “nb1” , “nb2” for the NB1 and NB2 parametrization of negative binomial in Cameron and Trivedi (1998).
corstr	Indicates the latent correlation structure of normal copula. Choices are “exch”, “ar”, and “unstr” for exchangeable, ar(1) and unstructured correlation structure, respectively.
link	The link function. Choices are “log” for the log link function, “logit” for the logit link function, and “probit” for the probit link function.
iprint	Indicates printing of some intermediate results, default FALSE

Details

This wrapper functions handles all the steps to obtain the weighted scores estimates and standard errors.

Value

A list containing the following components:

IEEest	The estimates of the regression and not regression parameters ignoring dependence.
CL1est	The vector with the CL1 estimated dependence parameters (latent correlations).
CL1lik	The value of the sum of bivariate marginal log-likelihoods at CL1 estimates.
WSest	The weighted score estimates of the regression and not regression parameters.
asympcov	The estimated weighted scores asymptotic covariance matrix.

Author(s)

Aristidis K. Nikoloulopoulos <A.Nikoloulopoulos@uea.ac.uk>
Harry Joe <harry.joe@ubc.ca>

References

Nikoloulopoulos, A.K., Joe, H. and Chaganty, N.R. (2011) Weighted scores method for regression models with dependent data. *Biostatistics*, **12**, 653–665.

Nikoloulopoulos, A.K. (2015a) Correlation structure and variable selection in generalized estimating equations via composite likelihood information criteria. *Arxiv e-prints*.

Nikoloulopoulos, A.K. (2015b) Weighted scores estimating equations for longitudinal ordinal data. *Arxiv e-prints*.

See Also

wtsc, solvewtsc, weightMat

Examples

```
################################################################################
#                      read and set up the data set
################################################################################
data(childvisit)
# covariates
season1<-childvisit$q
season1[season1>1]<-0
xdat<-cbind(1,childvisit$sex,childvisit$age,childvisit$m,season1)
# response
ydat<-childvisit$hosp
#id
id<-childvisit$id
#time
tvec<-childvisit$q
################################################################################
out<-wtsc.wrapper(xdat,ydat,id,tvec,margmodel="nb1",corstr="ar",iprint=TRUE)
## Not run:
################################################################################
```

```
#                        transform to binary responses                         #
################################################################################
y2<-ydat
y2[ydat>0]<-1
################################################################################
out<-wtsc.wrapper(xdat,y2,id,tvec,margmodel="bernoulli",link="probit",
corstr="exch",iprint=TRUE)
################################################################################
#                        via the code for ordinal                             #
################################################################################
out<-wtsc.ord.wrapper(xdat[,-1],2-y2,id,tvec,link="probit",
corstr="exch",iprint=TRUE)

## End(Not run)
```

Index