Package: weightedCL (via r-universe)

October 10, 2024

Version 0.5

Date 2022-10-05

Title Efficient and Feasible Inference for High-Dimensional Normal Copula Regression Models

Author Aristidis K. Nikoloulopoulos [aut, cre]

Maintainer Aristidis K. Nikoloulopoulos <a.nikoloulopoulos@uea.ac.uk>

Depends R (>= 3.5.0), matlab,rootSolve,sure,MASS

Description Estimates high-dimensional multivariate normal copula regression models with the weighted composite likelihood estimating equations in Nikoloulopoulos (2022) <arXiv:2203.04619>. It provides autoregressive moving average correlation structures and binary, ordinal, Poisson, and negative binomial regressions.

License GPL (>= 3.5.0)

NeedsCompilation yes

Repository CRAN

Date/Publication 2022-10-10 05:50:02 UTC

Contents

`weightedCL-package` *Efficient and feasible inference for high-dimensional normal copula regression models*

Description

The weighted composite likelihood estimating equations for high-dimensional normal copula regression models in Nikoloulopoulos (2022).

Details

This package contains R functions to estimate high-dimensional MVN copula regression models with the WCL estimating equations in Nikoloulopoulos (2022). It provides ARMA(p, q) correlation structures and binary, ordinal, Poisson, and negative binomial (both NB1 and NB2 parametrizations) regressions.

Author(s)

Aristidis K. Nikoloulopoulos.

References

Nikoloulopoulos, A.K. (2022) Efficient and feasible inference for high-dimensional normal copula regression models. *Arxiv e-prints*, <arXiv:2203.04619>. https://arxiv.org/abs/2203.04619.

`cl` *COMPOSITE LIKELIHOOD ESTIMATION FOR MVN COPULA*

Description

Composite likelihood estimation for MVN copula.

Usage

```
cl(p,q,b,gam,xdat,ydat,margmodel,link)
cl.ord(p,q,b,gam,xdat,ydat,link)
```

Arguments

p	The order of the autoregressive component.
q	The order of the moving average component.
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`	The $d \times p$ matrix of covariates, where d is the length of the time-series and 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).
`ydat`	The d-dimensional vector of dicrete time series reponse, where d is the length of the series.
`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 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 `cl.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 composite likelihood 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>`

References

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

See Also

`wcl` `iee`

Examples

```
################################################################################
#                         NB2 regression for count time-series data
################################################################################
################################################################################
```

```
#                       read and set up data set
################################################################################
data(polio)
ydat <-polio
d=length(ydat)
tvec=1:length(ydat)
tvec1=tvec-73
xdat <- cbind(1, tvec1/1000, cos(2 * pi * tvec1 / 12), sin(2 * pi * tvec1 / 12),
                      cos(2 * pi * tvec1 / 6), sin(2 * pi * tvec1 / 6))
################################################################################
#                       select the marginal model
################################################################################
margmodel="nb2"
################################################################################
#                       select the  ARMA structure
################################################################################
p=2;q=1
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl(p=p,q=q,b=i.est$reg,gam=i.est$gam,
             xdat,ydat,margmodel,link)
cat("\nest.rho: CL estimates\n")
print(est.rho$e)
################################################################################
#                          Ordinal time-series regression
################################################################################
################################################################################
#                       read and set up data set
################################################################################
data(sleep)
ydat=sleep$sleep
bydat=oydat=ydat
bydat[ydat==4]=0
bydat[ydat<4]=1
oydat[ydat==4]=1
oydat[ydat<4]=2
oydat[ydat==2]=3
oydat[ydat==3]=4

x1=sleep$heartrate
x2=sleep$temperature
z1=(x1-mean(x1))/sd(x1)
z2=(x2-mean(x2))/sd(x2)
xdat=cbind(z1,z2)
################################################################################
#                       select the link
################################################################################
link="probit"
################################################################################
```

```
#                       select the ARMA structure
################################################################################
p=1;q=0
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,oydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl.ord(p=p,q=q,b=i.est$reg,gam=i.est$gam,
             xdat,oydat,link)
cat("\nest.rho: CL estimates\n")
print(est.rho$e)
```

---

godambe *INVERSE GODAMBE MATRIX*

Description

Asymptotic covariance matrix of the weighted composite likelihood estimates.

Usage

```
godambe(b, gam, rh, p, q, xdat, margmodel, link = "log")
godambe.ord(b,gam,rh,p,q,xdat,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 autregressive and moving average parameters in high-dimensional normal copula regression models with an ARMA(p, q) correlation matrix.
`p`	The order of the autoregressive component.
`q`	The order of the moving average component.
`xdat`	The $d \times p$ matrix of covariates, where d is the length of the time-series and 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).
`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

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>

References

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

Nikoloulopoulos, A.K. (2022) Efficient and feasible inference for high-dimensional normal copula regression models. *Arxiv e-prints*, <arXiv:2203.04619>. https://arxiv.org/abs/2203.04619.

See Also

wcl, weightMat

Examples

```
################################################################################
#                       NB2 regression for count time-series data
################################################################################
################################################################################
#                       read and set up data set
################################################################################

data(polio)
ydat <-polio
d=length(ydat)
tvec=1:length(ydat)
tvec1=tvec-73
xdat <- cbind(1, tvec1/1000, cos(2 * pi * tvec1 / 12), sin(2 * pi * tvec1 / 12),
                      cos(2 * pi * tvec1 / 6), sin(2 * pi * tvec1 / 6))
################################################################################
#                       select the marginal model
################################################################################
margmodel="nb2"
################################################################################
#                       select the  ARMA structure
################################################################################
p=2;q=1
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
```

```
est.rho<-cl(p=p,q=q,b=i.est$reg,gam=i.est$gam,
             xdat,ydat,margmodel,link)
cat("\nest.rho: CL estimates\n")
print(est.rho$e)
################################################################################
#                       obtain the weight matrices
################################################################################
WtScMat<-weightMat(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
                   p=p,q=q,xdat,margmodel)
################################################################################
#            obtain the weighted composite likelihood estimates
################################################################################
est<-wcl(start=c(i.est$reg,i.est$gam),WtScMat,xdat,ydat,
              margmodel,link)
cat("est=parameter estimates\n")
print(est$r)
################################################################################
#                       obtain the inverse Godambe matrix
################################################################################
acov=godambe(b=est$r[-length(est$r)],gam=est$r[length(est$r)],
                rh=est.rho$e,p,q,xdat,margmodel)
cat("\nacov: inverse Godambe matrix\n")
print(acov)
################################################################################
#                          Ordinal time-series regression
################################################################################
################################################################################
#                       read and set up data set
################################################################################
data(sleep)
ydat=sleep$sleep
bydat=oydat=ydat
bydat[ydat==4]=0
bydat[ydat<4]=1
oydat[ydat==4]=1
oydat[ydat<4]=2
oydat[ydat==2]=3
oydat[ydat==3]=4

x1=sleep$heartrate
x2=sleep$temperature
z1=(x1-mean(x1))/sd(x1)
z2=(x2-mean(x2))/sd(x2)
xdat=cbind(z1,z2)
################################################################################
#                       select the link
################################################################################
link="probit"
################################################################################
#                       select the ARMA structure
################################################################################
p=1;q=0
################################################################################
```

```
#                        perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,oydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl.ord(p=p,q=q,b=i.est$reg,gam=i.est$gam,
             xdat,oydat,link)
cat("\nest.rho: CL estimates\n")
print(est.rho$e)
################################################################################
#                        obtain the weight matrices
################################################################################
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
                   p=p,q=q,xdat,link)
################################################################################
#            obtain the weighted composite likelihood estimates
################################################################################
est<-wcl.ord(start=c(i.est$reg,i.est$gam),WtScMat,
                  xdat,oydat,link)
cat("est=parameter estimates\n")
print(est$r)

################################################################################
#                        obtain the inverse Godambe matrix
################################################################################
acov=godambe.ord(b=est$r[1:2],gam=est$r[3:5],
                 rh=est.rho$e,p,q,xdat,link)
cat("\nacov: inverse Godambe matrix\n")
print(acov)
```

---

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	The $d \times p$ matrix of covariates, where d is the length of the time-series and 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).
ydat	The d-dimensional vector of dicrete time series reponse, where d is the length of the series.

`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>`

References

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

Examples

```
################################################################################
#                     NB2 regression for count time-series data
################################################################################
################################################################################
#                     read and set up data set
################################################################################
data(polio)
ydat <-polio
d=length(ydat)
tvec=1:length(ydat)
tvec1=tvec-73
xdat <- cbind(1, tvec1/1000, cos(2 * pi * tvec1 / 12), sin(2 * pi * tvec1 / 12),
                    cos(2 * pi * tvec1 / 6), sin(2 * pi * tvec1 / 6))
################################################################################
#                     select the marginal model
################################################################################
margmodel="nb2"

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

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(sleep)
ydat=sleep$sleep
bydat=oydat=ydat
bydat[ydat==4]=0
bydat[ydat<4]=1
oydat[ydat==4]=1
oydat[ydat<4]=2
oydat[ydat==2]=3
oydat[ydat==3]=4

x1=sleep$heartrate
x2=sleep$temperature
z1=(x1-mean(x1))/sd(x1)
z2=(x2-mean(x2))/sd(x2)
xdat=cbind(z1,z2)
################################################################################
#                       select the link
################################################################################
link="probit"
i.est<-iee.ord(xdat,ydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
```

---

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

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

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 $\gamma=(\alpha_1,\dots,\alpha_{K-1})$ 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.

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.

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)
```

---

pbvt *BIVARIATE NORMAL AND STUDENT CDFs WITH VECTORIZED INPUTS*

Description

Bivariate normal and Student cdfs with vectorized inputs

Usage

```
pbvt(z1,z2,param,icheck=FALSE)
```

Arguments

z1	scalar or vector of reals
z2	scalar or vector of reals
param	vector of length 2, or matrix with 2 columns; vectors and number of rows of matrix cannot be different if larger than 1; for param, first column is rho, second column is df.
icheck	TRUE if checks are made for proper inputs, default of FALSE

Value

cdf value(s)

References

Joe H (2014) *CopulaModel: Dependence Modeling with Copulas*. Software for book: *Dependence Modeling with Copulas*, Chapman & Hall/CRC, 2014.

Examples

```
cat("\n pbvt rho changing\n")
z1=.3; z2=.4; rho=seq(-.9,.9,.1); nu=2
param=cbind(rho,rep(nu,length(rho)))
out1=pbvt(z1,z2,param)
print(cbind(rho,out1))
cat("\n pbvt z1 changing\n")
z1=seq(-2,2,.4)
z2=.4; rho=.5; nu=2
out2=pbvt(z1,z2,c(rho,nu))
print(cbind(z1,out2))
```

polio *Polio cases in USA from Jan 1970 till Dec 1983*

Description

The data set contains the monthly number of cases of poliomyelitis in the United States between 1970 and 1983.

Usage

```
data(polio)
```

Format

The dataset consists of one variable of 168 monthly observations.

`polio` a numeric vector

Source

Zeger, S. A Regression Model for Time Series of Counts. *Biometrica*, **75**(4):621–629.

Examples

```
data(polio)
```

sleep *Infant sleep status data*

Description

The sleep data consist of sleep state measurements of a newborn infant together with his heart rate and temperature sampled every 30 seconds. The sleep states are classified as: (1) quiet sleep, (2) indeterminate sleep, (3) active sleep, (4) awake. The total number of observations is equal to 1024 and the objective is to predict the sleep state based on covariate information.

Usage

```
data(sleep)
```

Format

A data frame with 1024 observations on the following 3 variables:

`heartrate` Heart rate.

`sleep` An ordinal time series in the sense that the response increases from awake to active sleep, i.e., (4) < (1) < (2) < (3).

`temperature` Temperature

Source

Fokianos, K. and Kedem, B. (2003). Regression theory for categorical time series. *Statistical Science*, **18**(3):357–376.

Examples

```
data(sleep)
```

`wcl` *SOLVING THE WEIGHTED COMPOSITE LIKELIHOOD ESTIMATING EQUATIONS WITH INPUTS THE WEIGHT MATRICES AND DATA*

Description

Solving the weighted composite likelihood estimating equations with inputs the weight matrices and data.

Usage

```
wcl(start,WtScMat,xdat,ydat,margmodel,link)
wcl.ord(start,WtScMat,xdat,ydat,link)
```

Arguments

`start`	A starting value of the vector of regression and not regression parameters. The composite likelihood estimates of regression and not regression parameters is a good starting value.
`WtScMat`	A list containing the following components. omega: the matrix $\mathbf{\Omega}^{(1)}$; delta: the matrix $\mathbf{\Delta}^{(1)}$; X: the matrix $\mathbf{X}$.
`xdat`	The $d \times p$ matrix of covariates, where d is the length of the time-series and 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).
`ydat`	The d-dimensional vector of dicrete time series reponse, where d is the length of the series.
`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 estimates $\hat{\mathbf{a}}$ of the univariate parameters solving the weighted composite likelihood estimating equations.

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

Value

A list containing the following components:

root	The weighted composite likelihood estimates.
f.root	The value of the weighted composite likelihood estimating equations 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>

References

Nikoloulopoulos, A.K. (2022) Efficient and feasible inference for high-dimensional normal copula regression models. *Arxiv e-prints*, <arXiv:2203.04619>. https://arxiv.org/abs/2203.04619.

See Also

weightMat, godambe

Examples

```
################################################################################
#                       NB2 regression for count time-series data
################################################################################
################################################################################
#                       read and set up data set
################################################################################
data(polio)
ydat <-polio
d=length(ydat)
tvec=1:length(ydat)
tvec1=tvec-73
xdat <- cbind(1, tvec1/1000, cos(2 * pi * tvec1 / 12), sin(2 * pi * tvec1 / 12),
                      cos(2 * pi * tvec1 / 6), sin(2 * pi * tvec1 / 6))
################################################################################
#                       select the marginal model
################################################################################
margmodel="nb2"
################################################################################
#                       select the  ARMA structure
################################################################################
p=2;q=1
################################################################################
#                       perform CL1 estimation
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
```

```
print(c(i.est$reg,i.est$gam))
est.rho<-cl(p=p,q=q,b=i.est$reg,gam=i.est$gam,
            xdat,ydat,margmodel,link)
cat("\nest.rho: CL estimates\n")
print(est.rho$e)
################################################################################
#                      obtain the weight matrices
################################################################################
WtScMat<-weightMat(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
                   p=p,q=q,xdat,margmodel)
################################################################################
#           obtain the weighted composite likelihood estimates
################################################################################
est<-wcl(start=c(i.est$reg,i.est$gam),WtScMat,xdat,ydat,
             margmodel,link)
cat("est=parameter estimates\n")
print(est$r)
################################################################################
#                         Ordinal time-series regression
################################################################################
################################################################################
#                      read and set up data set
################################################################################
data(sleep)
ydat=sleep$sleep
bydat=oydat=ydat
bydat[ydat==4]=0
bydat[ydat<4]=1
oydat[ydat==4]=1
oydat[ydat<4]=2
oydat[ydat==2]=3
oydat[ydat==3]=4

x1=sleep$heartrate
x2=sleep$temperature
z1=(x1-mean(x1))/sd(x1)
z2=(x2-mean(x2))/sd(x2)
xdat=cbind(z1,z2)
################################################################################
#                      select the link
################################################################################
link="probit"
################################################################################
#                      select the ARMA structure
################################################################################
p=1;q=0
################################################################################
#                      perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,oydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl.ord(p=p,q=q,b=i.est$reg,gam=i.est$gam,
```

```
            xdat,oydat,link)
cat("\nest.rho: CL estimates\n")
print(est.rho$e)
################################################################################
#                     obtain the weight matrices
################################################################################
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
                 p=p,q=q,xdat,link)
################################################################################
#          obtain the weighted composite likelihood estimates
################################################################################
est<-wcl.ord(start=c(i.est$reg,i.est$gam),WtScMat,
                xdat,oydat,link)
cat("est=parameter estimates\n")
print(est$r)
```

`weightMat` *WEIGHT MATRICES FOR THE WEIGHTED COMPOSITE LIKELIHOOD ESTIMATING EQUATIONS*

Description

Weight matrices for the weighted composite likelhood estimating equations.

Usage

```
weightMat(b,gam,rh,p,q,xdat,margmodel,link)
weightMat.ord(b,gam,rh,p,q,xdat,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 autregressive and moving average parameters in high-dimensional normal copula regression models with an ARMA(p,q) correlation matrix.
`p`	The order of the autoregressive component.
`q`	The order of the moving average component.
`xdat`	The $d \times p$ matrix of covariates, where d is the length of the time-series and 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).
`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 matrices that form the weight matrices $\mathbf{W}^{(1)}$ of the weighted composite likelihood estimating equations in Nikoloulopoulos et al. (2022). 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 $\mathbf{\Omega}^{(1)}$ matrix.
`delta`	The $\mathbf{\Delta}^{(1)}$ matrix.
`X`	The $\mathbf{X}$ matrix.

Author(s)

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

References

Nikoloulopoulos, A.K. (2022) Efficient and feasible inference for high-dimensional normal copula regression models. *Arxiv e-prints*, <arXiv:2203.04619>. https://arxiv.org/abs/2203.04619.

See Also

`wcl`, `godambe`,

Examples

```
################################################################################
#                        NB2 regression for count time-series data
################################################################################
################################################################################
#                        read and set up data set
################################################################################
data(polio)
ydat <-polio
d=length(ydat)
tvec=1:length(ydat)
tvec1=tvec-73
xdat <- cbind(1, tvec1/1000, cos(2 * pi * tvec1 / 12), sin(2 * pi * tvec1 / 12),
                      cos(2 * pi * tvec1 / 6), sin(2 * pi * tvec1 / 6))
################################################################################
#                        select the marginal model
################################################################################
margmodel="nb2"
################################################################################
#                        select the  ARMA structure
################################################################################
p=2;q=1
################################################################################
#                        perform CL1 estimation
```

```
################################################################################
i.est<-iee(xdat,ydat,margmodel)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl(p=p,q=q,b=i.est$reg,gam=i.est$gam,
            xdat,ydat,margmodel,link)
cat("\nest.rho: CL estimates\n")
print(est.rho$e)
################################################################################
#                      obtain the weight matrices
################################################################################
WtScMat<-weightMat(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
                  p=p,q=q,xdat,margmodel)
################################################################################
#                         Ordinal time-series regression
################################################################################
################################################################################
#                      read and set up data set
################################################################################
data(sleep)
ydat=sleep$sleep
bydat=oydat=ydat
bydat[ydat==4]=0
bydat[ydat<4]=1
oydat[ydat==4]=1
oydat[ydat<4]=2
oydat[ydat==2]=3
oydat[ydat==3]=4

x1=sleep$heartrate
x2=sleep$temperature
z1=(x1-mean(x1))/sd(x1)
z2=(x2-mean(x2))/sd(x2)
xdat=cbind(z1,z2)
################################################################################
#                      select the link
################################################################################
link="probit"
################################################################################
#                      select the ARMA structure
################################################################################
p=1;q=0
################################################################################
#                      perform CL1 estimation
################################################################################
i.est<-iee.ord(xdat,oydat,link)
cat("\niest: IEE estimates\n")
print(c(i.est$reg,i.est$gam))
est.rho<-cl.ord(p=p,q=q,b=i.est$reg,gam=i.est$gam,
            xdat,oydat,link)
cat("\nest.rho: CL1 estimates\n")
print(est.rho$e)
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,xdat,ydat,id,tvec,corstr,link)
```

```
################################################################################
#                       obtain the weight matrices
################################################################################
WtScMat<-weightMat.ord(b=i.est$reg,gam=i.est$gam,rh=est.rho$e,
                   p=p,q=q,xdat,link)
```

Index