Package: ARpLMEC (via r-universe)

August 29, 2024

Type Package

Title Censored Mixed-Effects Models with Different Correlation Structures

Version 2.4.1

Date 2022-06-02

Author Rommy C. Olivari, Kelin Zhong, Aldo M. Garay and Victor H. Lachos

Maintainer Rommy C. Olivari <rco1@de.ufpe.br>

Description Left, right or interval censored mixed-effects linear model with autoregressive errors of order p or DEC correlation structure using the type-EM algorithm. The error distribution can be Normal or t-Student. It provides the parameter estimates, the standard errors and prediction of future observations (available only for the normal case). Olivari et all (2021) <doi:10.1080/10543406.2020.1852246>.

Depends R (>= 2.14)

Imports numDeriv, stats, MASS, mnormt, tcltk, expm, relliptical, TruncatedNormal, LaplacesDemon

NeedsCompilation no

License GPL (>= 2)

RoxygenNote 7.2.0

Encoding UTF-8

Repository CRAN

Date/Publication 2022-06-26 23:50:02 UTC

Contents

ARpMMEC.est *Censored Mixed-Effects Models with Autoregressive Correlation Structure and DEC for Normal and t-Student Errors*

Description

This functino fits left, right or intervalar censored mixed-effects linear model, with autoregressive errors of order p, using the EM algorithm. It returns estimates, standard errors and prediction of future observations.

Usage

```
ARpMMEC.est(
  y,
  x,
  z,
  tt,
  cc,
  nj,
  struc = "UNC",
  order = 1,
  initial = NULL,
  nu.fixed = TRUE,
  typeModel = "Normal",
  cens.type = "left",
  LI = NULL,
  LS = NULL,
  MaxIter = 200,
  error = 1e-04,
  Prev = FALSE,
  step = NULL,
  isubj = NULL,
  xpre = NULL,
  zpre = NULL
)
```

Arguments

y	Vector 1 x n of censored responses, where n is the sum of the number of observations of each individual
x	Design matrix of the fixed effects of order n x s, corresponding to vector of fixed effects.
z	Design matrix of the random effects of ordern x b, corresponding to vector of random effects.
tt	Vector 1 x n with the time the measurements were made, where n is the total number of measurements for all individuals. Default it's considered regular times.

cc	Vector of censoring indicators of length n, where n is the total of observations. For each observation: 0 if non-censored, 1 if censored.
nj	Vector 1 x m with the number of observations for each subject, where m is the total number of individuals.
struc	UNC,ARp,DEC,SYM or DEC(AR) for uncorrelated ,autoregressive, DEC(phi1,phi2), DEC(phi1,phi2=1), DEC(DEC(phi1,phi2=1)) structure, respectively
order	Order of the autoregressive process. Must be a positive integer value.
initial	List with the initial values in the next orden: betas,sigma2,alphas,phi and nu. If it is not indicated it will be provided automatically. Default is NULL
nu.fixed	Logical. Should estimate the parameter "nu" for the t-student distribution?. If is False indicates the value in the list of initial values. Default is FALSE
typeModel	Normal for Normal distribution and Student for t-Student distribution. Default is Normal
cens.type	left for left censoring, right for right censoring and interval for intervalar censoring. Default is left
LI	Vector censoring lower limit indicator of length n. For each observation: 0 if non-censored, -inf if censored. It is only indicated for when cens.type is both. Default is NULL
LS	Vector censoring upper limit indicator of length n. For each observation: 0 if non-censored, inf if censored.It is only indicated for when cens.type is both. Default is NULL
MaxIter	The maximum number of iterations of the EM algorithm. Default is 200
error	The convergence maximum error. Default is 0.0001
Prev	Indicator of the prediction process. Available at the moment only for the typeModel=normal case. Default is FALSE
step	Number of steps for prediction. Default is NULL
isubj	Vector indicator of subject included in the prediction process. Default is NULL
xpre	Design matrix of the fixed effects to be predicted. Default is NULL.
zpre	Design matrix of the random effects to be predicted. Default is NULL.

Value

returns list of class “ARpMMEC”:

FixEffect	Data frame with: estimate, standar errors and confidence intervals of the fixed effects.
Sigma2	Data frame with: estimate, standar errors and confidence intervals of the variance of the white noise process.
Phi	Data frame with: estimate, standar errors and confidence intervals of the autoregressive parameters.
RandEffect	Data frame with: estimate, standar errors and confidence intervals of the random effects.
nu	the parameter "nu" for the t-student distribution

Est	Vector of parameters estimate (fixed Effects, sigma2, phi, random effects).
SE	Vector of the standard errors of (fixed Effects, sigma2, phi, random effects).
Residual	Vector of the marginal residuals.
loglik	Log-likelihood value.
AIC	Akaike information criterion.
BIC	Bayesian information criterion.
AICc	Corrected Akaike information criterion.
iter	Number of iterations until convergence.
Yfit	Vector "y" fitted
MI	Information matrix
Prev	Predicted values (if xpre and zpre is not NULL).
time	Processing time.
others	The first and second moments of the random effect and vector Y

References

Olivari, R. C., Garay, A. M., Lachos, V. H., & Matos, L. A. (2021). Mixed-effects models for censored data with autoregressive errors. Journal of Biopharmaceutical Statistics, 31(3), 273-294. doi:10.1080/10543406.2020.1852246

Examples

```
## Not run:
p.cens   = 0.1
m           = 10
D = matrix(c(0.049,0.001,0.001,0.002),2,2)
sigma2 = 0.30
phi    = 0.6
beta   = c(1,2,1)
nj=rep(4,10)
tt=rep(1:4,length(nj))
x<-matrix(runif(sum(nj)*length(beta),-1,1),sum(nj),length(beta))
z<-matrix(runif(sum(nj)*dim(D)[1],-1,1),sum(nj),dim(D)[1])
data=ARpMMEC.sim(m,x,z,tt,nj,beta,sigma2,D,phi,struc="ARp",typeModel="Normal",p.cens=p.cens)

teste1=ARpMMEC.est(data$y_cc,x,z,tt,data$cc,nj,struc="ARp",order=1,typeModel="Normal",MaxIter = 2)
teste2=ARpMMEC.est(data$y_cc,x,z,tt,data$cc,nj,struc="ARp",order=1,typeModel="Student",MaxIter = 2)

xx=matrix(runif(6*length(beta),-1,1),6,length(beta))
zz=matrix(runif(6*dim(D)[1],-1,1),6,dim(D)[1])
isubj=c(1,4,5)
teste3=ARpMMEC.est(data$y_cc,x,z,tt,data$cc,nj,struc="ARp",order=1,typeModel="Normal",
                   MaxIter = 2,Prev=TRUE,step=2,isubj=isubj,xpre=xx,zpre=zz)
teste3$Prev


## End(Not run)
```

ARpMMEC.sim *Generating Censored Autoregressive Dataset with Mixed Effects, for normal distribution.*

Description

This function simulates a censored response variable with autoregressive errors of order p, with mixed effect and a established censoring rate. This function returns the censoring vector and censored response vector.

Usage

```
ARpMMEC.sim(
  m,
  x = NULL,
  z = NULL,
  tt = NULL,
  nj,
  beta,
  sigmae,
  D,
  phi,
  struc = "ARp",
  order = 1,
  typeModel = "Normal",
  p.cens = NULL,
  n.cens = NULL,
  cens.type = "left",
  nu = NULL
)
```

Arguments

m	Number of individuals
x	Design matrix of the fixed effects of order n x s, corresponding to vector of fixed effects.
z	Design matrix of the random effects of ordern x b, corresponding to vector of random effects.
tt	Vector 1 x n with the time the measurements were made, where n is the total number of measurements for all individuals.
nj	Vector 1 x m with the number of observations for each subject, where m is the total number of individuals.
beta	Vector of values fixed effects.
sigmae	It's the value for sigma.
D	Covariance Matrix for the random effects.

phi	Vector of length Arp, of values for autoregressive parameters.
struc	Correlation structure. This must be one of UNC,ARp,DEC,SYM or DEC(AR).
order	Order of the autoregressive process. Must be a positive integer value.
typeModel	Normal for Normal distribution and Student for t-Student distribution. Default is Normal
p.cens	Censoring percentage for the process. Default is NULL
n.cens	Censoring level for the process. Default is NULL
cens.type	left for left censoring, right for right censoring and interval for intervalar censoring. Default is left
nu	degrees of freedom for t-Student distibution (nu > 0, maybe non-integer).

Value

returns list:

cc	Vector of censoring indicators.
y_cc	Vector of responses censoring.

Examples

```
## Not run:
p.cens   = 0.1
m           = 10
D = matrix(c(0.049,0.001,0.001,0.002),2,2)
sigma2 = 0.30
phi    = 0.6
beta   = c(1,2,1)
nj=rep(4,10)
tt=rep(1:4,length(nj))
x<-matrix(runif(sum(nj)*length(beta),-1,1),sum(nj),length(beta))
z<-matrix(runif(sum(nj)*dim(D)[1],-1,1),sum(nj),dim(D)[1])
data=ARpMMEC.sim(m,x,z,tt,nj,beta,sigma2,D,phi,struc="ARp",typeModel="Normal",p.cens=p.cens)
 y<-data$y_cc
 cc<-data$cc

## End(Not run)
```

Index