Package 'StReg'

January 20, 2025

Type Package

Title Student's t Regression Models

Version 1.1

Date 2023-08-08

Description It contains functions to estimate multivariate Student's t dynamic and static regression models for given degrees of freedom and lag length. Users can also specify the trends and dummies of any kind in matrix form.
Poudyal, N., and Spanos, A. (2022) <doi:10.3390/econometrics10020017>.
Spanos, A. (1994) <http://www.jstor.org/stable/3532870>.

Depends R (>= 3.0.0)

Imports ADGofTest,numDeriv,MCMCpack,matlab,tseries

License GPL-2

NeedsCompilation no

Author Niraj Poudyal [aut, cre]

Maintainer Niraj Poudyal <niraj.poudyal@ku.edu.np>

Repository CRAN

Date/Publication 2023-08-17 06:42:41 UTC

Contents

StAR   *Student's t Autoregression (StAR)*

Description

Maximum likelihood estimation of Student's t autoregression model is the purpose of this function. It can be used to estimate the linear autoregressive function (conditional mean) and the quadratic autoskedastic function (conditional variance). Users can specify the model with deterministic variables such as trends and dummies in matrix form.

Usage

```
StAR(Data,Trend=1,lag=1,v=1,maxiter=1000,meth="BFGS",hes="FALSE",init="na")
```

Arguments

Data	A data vector with one column. Cannot be empty.
Trend	A matrix with columns representing deterministic variables like trends and dummies. If 1 (default), model with only constant intercept is estimated. If 0, the model is estimated without an intercept term.
lag	A positive integer (default value is 1) as lag length.
v	A scalar (default value is 1) greater than or equal to 1. Degrees of freedom parameter.
maxiter	Maximum number of iteration. Must be an integer bigger than 10.
meth	One of the optimization method from optim function (default value is BFGS). See details of optim function.
hes	Logical (default value is FALSE). If TRUE produces estimated hessian matrix and the standard errors of estimates.
init	If na (default), initial values for optimization are generated from a uniform distribution. A vector of initial values can also be used (not recommended). The length of the init vector must be equal to the number of parameters of the joint distribution.

Details

For the functional form of the autoregressive function and the autoskedastic function, see Spanos (1994) and Poudyal (2012).

Value

beta	coefficients of the autoregressive function including the coefficients of trends in matrix form with standard errors and p-values. If some of the standard errors are NA's, the StAR() function has to be run again.
var.coef	coefficients of the autoregressive function, standard errors and p-values if hes=TRUE.
like	maximum log likelihood value.

sigma	contemporary variance covariance matrix.
cvar	`(v/(v+lag*l-2))*sigma*cvar` is the fitted value of the autoskedastic function where `l` is the rank of `Data`
trend	estimated trend in the variables.
res	non-standardized residuals
fitted	fitted values of the autoregressive function.
init	estimates of the joint distribution parameters. It can be used as new initial value `init` in `StVAR()` to improve optimization further.
hes	estimated hessian matrix.
S	variance covariance matrix of the joint distribution.
R.squared	R-squared of the model.
F.stat	F-statistics of the model.
ad	Anderson-Darling test for Student's t distribution.

Author(s)

Niraj Poudyal <niraj.poudyal@ku.edu.np>

References

Poudyal, N. and Spanos, A. (2022), Model Validation and DSGE Modeling. Econometrics, 10 (2), 17.

Spanos, A. (1994), On Modeling Heteroskedasticity: the Student's t and Elliptical Linear Regression Models. Econometric Theory, 10: 286-315.

Examples

```
## StAR Model#####
## Random number seed
set.seed(4093)

## Creating trend variable.
t <- seq(1,50,1)

# Generating data on y.
y <-  0.004 + 0.0045*t - 0.09*t^2 + 50*rt(50,df=5)
T <- length(y)
# The trend matrix
Trend <- cbind(1,poly(t,2,raw=TRUE))

# Estimating the model
star <- StAR(y,lag=1,Trend=Trend,v=5,maxiter=2000)

# Generate arbitrary dates
dates <- seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "weeks")
```

```
## Plotting the variable y, its estimated trend and the fitted value.

oldpar <- par(mfcol=c(3,1))
matplot(dates[2:T],cbind(y[2:T],star$fitted,star$trend),xlab="Months",type='l',
lty=c(1,2,3),lwd=c(1,1,3),col=c("black","blue","black"),ylab=" ",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
legend("bottomleft",legend=c("data","trend","fitted values"),lty=c(1,2,3),lwd=c(1,1,3),
col=c("black","blue","black"),cex=.85)
hist(star$res,main="",xlab="") ## Histogram of y
matplot(dates[2:T],cbind(star$cvar),xlab="Months",type='l',lty=2,lwd=1,
ylab="fitted variance",xaxt="n")
axis.Date(1,at=seq(as.Date("2014/1/1"),as.Date("2016/1/1"),"months"),labels=TRUE)
par(oldpar)
```

StDLM   *Student's t Dynamic Linear Regression Model (StDLM)*

Description

Maximum likelihood estimation of Student's t dynamic linear regression model is the purpose of this function. It can be used to estimate the dynamic linear autoregressive function (conditional mean) and the quadratic autoskedastic function (conditional variance). Users can specify the model with deterministic variables such as trends and dummies in the matrix form.

Usage

```
StDLM(y,X,Trend=1,lag=1,v=1,maxiter=1000,meth="BFGS",hes="FALSE",init="na")
```

Arguments

y	A vector representing dependent variable. Cannot be empty.
X	A data matrix whose columns represent exogenous variables. Cannot be empty.
Trend	A matrix with columns representing deterministic variables like trends and dummies. If 1 (default), model with only constant intercept is estimated. If 0, the model is estimated without an intercept term.
lag	A non-negative integer (default value is 1) as lag length. If lag=0, a static model is estimated.
v	A scalar (default value is 1) greater than or equal to 1. Degrees of freedom parameter.
maxiter	Maximum number of iteration. Must be an integer bigger than 10.
meth	One of the optimization method from `optim` function (default value is BFGS). See details of `optim` function.
hes	Logical (default value is FALSE). If TRUE produces estimated hessian matrix and the standard errors of estimates.
init	If na (default), initial values for optimization are generated from a uniform distribution. A vector of initial values can also be used (not recommended). The length of the init vector must be equal to the number of parameters of the joint distribution.

Details

For the functional form of the autoregressive function and the autoskedastic function, see Spanos (1994) and Poudyal (2012).

Value

beta	coefficients of the dynamic linear regression model including the coefficients of trends in matrix form with standard errors and p-values. If some of the standard errors are NA's, the StVAR() function has to be run again.
var.coef	coefficients of the autoskedastic function, standard errors and p-values if if hes=TRUE
like	maximum log likelihood value.
sigma	contemporary variance covariance matrix.
cvar	(v/(v+lag*l-2))*sigma*cvar is the fitted value of the autoskedastic function where l is the rank of Data
trend	estimated trend in the variable y.
res	non-standardized residuals
fitted	fitted values of the autoregressive function.
init	estimates of the joint distribution parameters. It can be used as new initial value init in StVAR() to improve optimization further.
S	variance covariance matrix of the joint distribution.
R.squared	R-squared of the model.
F.stat	F-statistics of the model.
ad	Anderson-Darling test for Student's t distribution.

Author(s)

Niraj Poudyal <niraj.poudyal@ku.edu.np>

References

Poudyal, N. and Spanos, A. (2022), Model Validation and DSGE Modeling. Econometrics, 10 (2), 17.

Spanos, A. (1994), On Modeling Heteroskedasticity: the Student's t and Elliptical Linear Regression Models. Econometric Theory, 10: 286-315.

Examples

```
## StDLM Model#####
## Random number seed
set.seed(7504)

## Creating trend variable.
t <- seq(1,100,1)
ut <- rt(100,df=5)
# Generating data on y, x and z.
```

```
y <-  0.004 + 0.0045*t - 0.09*t^2 + 0.001*t^3 + 50*ut
x <-  0.05 - 0.005*t + 0.09*t^2 - 0.001*t^3 + 40*ut
z <-  0.08 - 0.006*t + 0.08*t^2 - 0.001*t^3 + 30*ut

# The trend matrix
Trend <- cbind(1,poly(t,3,raw=TRUE))

# Estimating the model
stdlrm <- StDLM(y,cbind(x,z),lag=1,Trend=Trend,v=5,maxiter=2000)

# Generate arbitrary dates
dates <- seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "weeks")

## Plotting the variable y, its estimated trend and the fitted value.
lag <- 1
oldpar <- par(mfcol=c(3,1))
matplot(dates[(lag+1):length(y)],cbind(y[(lag+1):length(y)],stdlrm$fit,stdlrm$trend),
xlab="Months",type='l',lty=c(1,2,3),lwd=c(1,1,3),col=c("black","blue","black"),ylab=" ",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
legend("bottomleft",legend=c("data","trend","fitted values"),lty=c(1,2,3),lwd=c(1,1,3),
col=c("black","blue","black"),cex=.85)
hist(stdlrm$res,main="",xlab="") ## Histogram of y
matplot(dates[2:length(y)],cbind(stdlrm$cvar),xlab="Months",type='l',lty=2,lwd=1,
ylab="fitted variance",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
par(oldpar)
```

StLM   *Student's t Linear Regression Model (StLM)*

Description

Maximum likelihood estimation of Student's t linear regression model is the purpose of this function. It can be used to estimate the linear regression function (conditional mean) and the quadratic skedastic function (conditional variance). Users can specify the model with deterministic variables such as trends and dummies in the matrix form.

Usage

```
StLM(y,X,Trend=1,v=1,maxiter=1000,meth="BFGS",hes="FALSE",init="na")
```

Arguments

y	A vector representing dependent variable. Cannot be empty.
X	A data matrix whose columns represent exogenous variables. Cannot be empty.
Trend	A matrix with columns representing deterministic variables like trends and dummies. If 1 (default), model with only constant intercept is estimated. If 0, the model is estimated without an intercept term.

v	A scalar (default value is 1) greater than or equal to 1. Degrees of freedom parameter.
maxiter	Maximum number of iteration. Must be an integer bigger than 10.
meth	One of the optimization method from optim function (default value is BFGS). See details of optim function.
hes	Logical (default value is FALSE). If TRUE produces estimated hessian matrix and the standard errors of estimates.
init	If na (default), initial values for optimization are generated from a uniform distribution. A vector of initial values can also be used (not recommended). The length of the init vector must be equal to the number of parameters of the joint distribution.

Details

For the functional form of the regression function and the skedastic function, see Spanos (1994) and Poudyal (2012).

Value

beta	coefficients of the regression function including the coefficients of trends in matrix form with standard errors and p-values. If some of the standard errors are NA's, the StLM() function has to be run again.
var.coef	coefficients of the skedastic function, standard errors and p-values if if hes=TRUE
like	maximum log likelihood value.
sigma	contemporary variance covariance matrix.
cvar	(v/(v+l-2))*sigma*cvar is the fitted value of the skedastic function where l is the rank of Data
trend	estimated trend in the variable y.
res	non-standardized residuals
fitted	fitted values of the regression function.
init	estimates of the joint distribution parameters. It can be used as new initial value init in StVAR() to improve optimization further.
S	variance covariance matrix of the joint distribution.
R.squared	R-squared of the model.
F.stat	F-statistics of the model.
ad	Anderson-Darling test for Student's t distribution.

Author(s)

Niraj Poudyal <niraj.poudyal@ku.edu.p>

References

Poudyal, N. and Spanos, A. (2022), Model Validation and DSGE Modeling. Econometrics, 10 (2), 17.

Spanos, A. (1994), On Modeling Heteroskedasticity: the Student's t and Elliptical Linear Regression Models. Econometric Theory, 10: 286-315.

Examples

```
## stlm Model#####
## Random number seed
set.seed(7504)

## Creating trend variable.
t <- seq(1,50,1)

# Generating data on y, x and z.
x <-  rt(50,df=5) ; set.seed(7505)
z <- 5*rt(50,df=5) ; set.seed(7506)

y <-  4 + 0.045*t + 2*x + 5*z + rt(50,25,df=5)

# The trend matrix
Trend <- cbind(1,poly(t,1,raw=FALSE))
colnames(Trend) <- c("const","t")

# Estimating the model
stlrm <- StLM(y,cbind(x,z),Trend=Trend,v=5,maxiter=2000)

# Generate arbitrary dates
dates <- seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "weeks")

## Plotting the variable y, its estimated trend and the fitted value.
oldpar <- par(mfcol=c(3,1))
matplot(dates[1:length(y)],cbind(y[1:length(y)],stlrm$fit,stlrm$trend),xlab="Months",
type='l',lty=c(1,2,3),lwd=c(1,1,3),col=c("black","blue","black"),ylab=" ",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
legend("bottomleft",legend=c("data","trend","fitted values"),lty=c(1,2,3),lwd=c(1,1,3),
col=c("black","blue","black"),cex=.85)
hist(stlrm$res,main="",xlab="") ## Histogram of y
matplot(dates[1:length(y)],cbind(stlrm$cvar),xlab="Months",type='l',lty=2,lwd=1,
ylab="fitted variance",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
par(oldpar)
```

StVAR   *Student's t Vector Autoregression (StVAR)*

Description

Maximum likelihood estimation of Student's t vector autoregression (VAR) model is the purpose of this function. It can be used to estimate the linear autoregressive function (conditional mean) and the quadratic autoskedastic function (conditional variance). Users can specify the model with deterministic variables such as trends and dummies in the matrix form.

Usage

```
StVAR(Data,Trend=1,lag=1,v=1,maxiter=1000,meth="BFGS",hes="FALSE",init="na")
```

Arguments

Data	A data matrix with at least two columns. Cannot be empty.
Trend	A matrix with columns representing deterministic variables like trends and dummies. If 1 (default), model with only constant intercept is estimated. If 0, the model is estimated without the intercept term.
lag	A positive integer (default value is 1) as lag length.
v	A scalar (default value is 1) greater than or equal to 1. Degrees of freedom parameter.
maxiter	Maximum number of iteration. Must be an integer bigger than 10.
meth	One of the optimization method from `optim` function (default value is BFGS). See details of `optim` function.
hes	Logical (default value is FALSE). If TRUE produces estimated hessian matrix and the standard errors of estimates.
init	If na (default), initial values for optimization are generated from a uniform distribution. A vector of initial values can also be used (not recommended). The length of the init vector must be equal to the number of parameters of the joint distribution.

Details

For the functional form of the autoregressive function and the autoskedastic function, see Spanos (1994) and Poudyal (2012).

Value

beta	coefficients of the autoregressive function including the coefficients of trends in matrix form with standard errors and p-values. If some of the standard errors are NA's, the `StVAR()` function has to be run again.
var.coef	coefficients of the autoskedastic (conditional variance) function, standard errors and p-values.
like	maximum log likelihood value.
sigma	contemporary variance-covariance matrix.
cvar	`(v/(v+lag*l-2))*sigma*cvar` is the fitted value of the autoskedastic function where `l` is the rank of `Data`.

trends	estimated trends in the variables.
res	non-standardized residuals.
fitted	fitted values of the autoregressive function.
init	estimates of the joint distribution parameters. It can be used as new initial value init in StVAR() to improve optimization further.
hes	the estimated hessian matrix if hes=TRUE.
S	variance covariance matrix of the joint distribution.
R.squared	R-squared of each equation of the model.
F.stat	F-statistics of each equation of the model.
ad	Anderson-Darling test for Student's t distribution.

Author(s)

Niraj Poudyal <niraj.poudyal@ku.edu.np>

References

Poudyal, N. and Spanos, A. (2022), Model Validation and DSGE Modeling. Econometrics, 10 (2), 17.

Spanos, A. (1994), On Modeling Heteroskedasticity: the Student's t and Elliptical Linear Regression Models. Econometric Theory, 10: 286-315.

Examples

```
## StVAR Model#####
 ## Random number seed
 set.seed(7504)

 ## Creating trend variable.
 t <- seq(1,50,1)
 y <- x <- vector(length=50)
 y[1] <- x[1] <- 0
 # Generating data on y and x.
 ut <- rt(50,df=5)
 for(i in 2:50)
 {
 y[i] <-  0.004 + 0.0045*t[i] - 0.9*y[i-1] + .2*x[i-1] + ut[i]
 x[i] <-  0.05 - 0.005*t[i] + 0.8*x[i-1] + 1*y[i-1] + ut[i]
 }

 # The trend matrix

 Trend <- cbind(1,t)

 # Estimating the model
 stvar <- StVAR(cbind(y,x),lag=1,Trend=Trend,v=6,maxiter=2000,hes=TRUE)

 # Generate arbitrary dates
```

```
dates <- seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "weeks")

## Plotting the variable y, its estimated trend and the fitted value.
oldpar <- par(mfcol=c(3,1))
matplot(dates[2:length(y)],cbind(y[2:length(y)],stvar$fit[,1],stvar$trend[,1]),xlab="Months",
type='l',lty=c(1,2,3), lwd=c(1,1,3),col=c("black","blue","black"),ylab=" ",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
legend("bottomleft",legend=c("data","trend","fitted values"),lty=c(1,2,3),lwd=c(1,1,3),
col=c("black","blue","black"),cex=.85)
hist(stvar$res[,1],main="",xlab="") ## Histogram of y
matplot(dates[2:length(y)],cbind(stvar$cvar),xlab="Months",type='l',lty=2,lwd=1,
ylab="fitted variance",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
par(oldpar)

## Plotting the variable x, its estimated trend and the fitted value.
oldpar <- par(mfcol=c(3,1))
matplot(dates[2:length(x)],cbind(x[2:length(x)],stvar$fit[,2],stvar$trend[,2]),
xlab="Months",type='l',lty=c(1,2,3),lwd=c(1,1,3),col=c("black","blue","black"),ylab=" ",xaxt="n")
axis.Date(1, at = seq(as.Date("2014/1/1"), as.Date("2016/1/1"), "months"),labels=TRUE)
legend("bottomleft",legend=c("data","trend","fitted values"),lty=c(1,2,3),lwd=c(1,1,3),
col=c("black","blue","black"),cex=.85)
hist(stvar$res[,2],main="",xlab="") ## Histogram of x
matplot(dates[2:length(x)],cbind(stvar$cvar),xlab="Months",type='l',lty=2,lwd=1,
ylab="fitted variance",xaxt="n")
axis.Date(1,at=seq(as.Date("2014/1/1"),as.Date("2016/1/1"),"months"),labels=TRUE)
par(oldpar)
```

Index