Package ‘xdcclarge’

July 21, 2025

Type Package

Title Estimating a (c)DCC-GARCH Model in Large Dimensions

Version 0.1.0

Description Functions for Estimating a (c)DCC-GARCH Model in large dimensions based on a publication by Engle et,al (2017) <doi:10.1080/07350015.2017.1345683> and Nakagawa et,al (2018) <doi:10.3390/ijfs6020052>.
This estimation method is consist of composite likelihood method by Pakel et al. (2014) <http://paneldataconference2015.ceu.hu/Program/Cavit-Pakel.pdf>
and (Non-)linear shrinkage estimation of covariance matrices by Ledoit and Wolf (2004,2015,2016). (<doi:10.1016/S0047-259X(03)00096-4>, <doi:10.1214/12-AOS989>, <doi:10.1016/j.jmva.2015.04.006>).

License GPL (>= 2)

Author Kei Nakagawa [aut, cre] (ORCID: <https://orcid.org/0000-0001-5046-8128>),
Mitsuyoshi Imamura [aut]

Maintainer Kei Nakagawa <kei.nak.0315@gmail.com>

LazyData TRUE

Imports stats, Rcpp(>= 0.10.6), nlshrink

Depends R(>= 3.0.2)

LinkingTo Rcpp(>= 0.10.6),RcppArmadillo(>= 0.2.34)

Suggests rugarch(>= 1.0.0)

RoxygenNote 6.0.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-07-12 18:50:03 UTC

Contents

`cdcc_correlations` *This function get the correlation matrix (Rt) of estimated cDCC-GARCH model.*

Description

This function get the correlation matrix (Rt) of estimated cDCC-GARCH model.

Usage

```
cdcc_correlations(param, stdresids, uncR, ts)
```

Arguments

`param`	cDCC-GARCH parameters(alpha,beta)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)
`ts`	ts how many time series are you taking

Value

the correlation matrix (Rt) of estimated cDCC-GARCH model (T by N^2)

Note

Rt are vectorized values of the conditional correlation matrix(Rt) until time t(ts) for each row.

`cdcc_estimation` *This function estimates the parameters(alpha,beta) and time-varying correlation matrices(Rt) of cDCC-GARCH model.*

Description

This function estimates the parameters(alpha,beta) and time-varying correlation matrices(Rt) of cDCC-GARCH model.

Usage

```
cdcc_estimation(ini.para = c(0.05, 0.93), ht, residuals, method = c("COV",
  "LS", "NLS"), ts = 1)
```

Arguments

`ini.para`	initial cDCC-GARCH parameters(alpha,beta) of optimization
`ht`	matrix of conditional variance vectors
`residuals`	matrix of residual(de-mean) returns
`method`	shrinkage method of unconditional correlation matrix(Cov:sample,LS:Linear Shrinkage,NLS:NonLinear Shrinkage)
`ts`	ts how many time series are you taking(dufalut:1 latest value)

Value

time-varying correlations(Rt) and the result of estimation

Note

Rt are vectorized values of the conditional correlation matrix(Rt) until time t(ts) for each row.

Examples

```
library(rugarch)
library(xdcclarge)
#load data
data(us_stocks)
n<-3
Rtn<-log(us_stocks[-1,1:n]/us_stocks[-nrow(us_stocks),1:n])

# Step 1:GARCH Parameter Estimation with rugarch
spec = ugarchspec()
mspec = multispec( replicate(spec, n = n) )
fitlist = multifit(multispec = mspec, data = Rtn)
ht<-sigma(fitlist)^2
residuals<-residuals(fitlist)

# Step 2:cDCC-GARCH Parameter Estimation with xdcclarge
```

```
  cDCC<-cdcc_estimation(ini.para=c(0.05,0.93) ,ht ,residuals)
  #Time varying correlation matrix Rt at time t
  (Rt<-matrix(cDCC$cdcc_Rt,n,n))

  ## Not run:
  #If you want Rt at time t-s,then
  s<-10
  cDCC<-cdcc_estimation(ini.para=c(0.05,0.93) ,ht ,residuals,ts = s)
  matrix(cDCC$cdcc_Rt[s,],n,n)

## End(Not run)
```

`cdcc_gradient` *This functions calculates numerical gradient of log-likelihood of cDCC-GARCH model.*

Description

This functions calculates numerical gradient of log-likelihood of cDCC-GARCH model.

Usage

```
cdcc_gradient(param, ht, residuals, stdresids, uncR, d = 1e-05)
```

Arguments

`param`	cDCC-GARCH parameters(alpha,beta)
`ht`	matrix of conditional variance vectors (T by N)
`residuals`	matrix of residual(de-mean) returns (T by N)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)
`d`	(log-lik(x+d) - log-lik(x))/d

Value

numerical gradient of log-likelihood of cDCC-GARCH model(vector)

cdcc_loglikelihood *This function calculates log-likelihood of cDCC-GARCH model.*

Description

This function calculates log-likelihood of cDCC-GARCH model.

Usage

```
cdcc_loglikelihood(param, ht, residuals, stdresids, uncR)
```

Arguments

`param`	cDCC-GARCH parameters(alpha,beta)
`ht`	matrix of conditional variance vectors (T by N)
`residuals`	matrix of residual(de-mean) returns (T by N)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)

Value

log-likelihood of cDCC-GARCH model (scaler)

cdcc_optim *This function optimizes log-likelihood of cDCC-GARCH model.*

Description

This function optimizes log-likelihood of cDCC-GARCH model.

Usage

```
cdcc_optim(param, ht, residuals, stdresids, uncR)
```

Arguments

`param`	cDCC-GARCH parameters(alpha,beta)
`ht`	matrix of conditional variance vectors (T by N)
`residuals`	matrix of residual(de-mean) returns (T by N)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)

Value

results of optimization

`dcc_correlations` *This function get the correlation matrix (Rt) of estimated DCC-GARCH model.*

Description

This function get the correlation matrix (Rt) of estimated DCC-GARCH model.

Usage

```
dcc_correlations(param, stdresids, uncR, ts)
```

Arguments

`param`	DCC-GARCH parameters(alpha,beta)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)
`ts`	ts how many time series are you taking

Value

the correlation matrix (Rt) of estimated DCC-GARCH model (T by N^2)

Note

Rt are vectorized values of the conditional correlation matrix(Rt) until time t(ts) for each row.

`dcc_estimation` *This function estimates the parameters(alpha,beta) and time-varying correlation matrices(Rt) of DCC-GARCH model.*

Description

This function estimates the parameters(alpha,beta) and time-varying correlation matrices(Rt) of DCC-GARCH model.

Usage

```
dcc_estimation(ini.para = c(0.05, 0.93), ht, residuals, method = c("COV",
  "LS", "NLS"), ts = 1)
```

Arguments

`ini.para`	initial DCC-GARCH parameters(alpha,beta) of optimization
`ht`	matrix of conditional variance vectors
`residuals`	matrix of residual(de-mean) returns
`method`	shrinkage method of unconditional correlation matrix(Cov:sample,LS:Linear Shrinkage,NLS:NonLinear Shrinkage)
`ts`	ts how many time series are you taking(dufalut:1 latest value)

Value

time-varying correlations(Rt) and the result of estimation

Note

Rt are vectorized values of the conditional correlation matrix(Rt) until time t(ts) for each row.

Examples

```
  library(rugarch)
  library(xdcclarge)
  #load data
  data(us_stocks)
  n<-3
  Rtn<-log(us_stocks[-1,1:n]/us_stocks[-nrow(us_stocks),1:n])

  # Step 1:GARCH Parameter Estimation with rugarch
  spec = ugarchspec()
  mspec = multispec( replicate(spec, n = n) )
  fitlist = multifit(multispec = mspec, data = Rtn)
  ht<-sigma(fitlist)^2
  residuals<-residuals(fitlist)

  # Step 2:DCC-GARCH Parameter Estimation with xdcclarge
  DCC<-dcc_estimation(ini.para=c(0.05,0.93) ,ht ,residuals)
  #Time varying correlation matrix Rt at time t
  (Rt<-matrix(DCC$dcc_Rt,n,n))

  ## Not run:
  #If you want Rt at time t-s,then
  s<-10
  DCC<-dcc_estimation(ini.para=c(0.05,0.93) ,ht ,residuals,ts = s)
  matrix(DCC$cdcc_Rt[s,],n,n)

## End(Not run)
```

`dcc_gradient`	*This functions calculates numerical gradient of log-likelihood of DCC-GARCH model.*

Description

This functions calculates numerical gradient of log-likelihood of DCC-GARCH model.

Usage

```
dcc_gradient(param, ht, residuals, stdresids, uncR, d = 1e-05)
```

Arguments

`param`	DCC-GARCH parameters(alpha,beta)
`ht`	matrix of conditional variance vectors (T by N)
`residuals`	matrix of residual(de-mean) returns (T by N)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)
`d`	(log-lik(x+d) - log-lik(x))/d

Value

numerical gradient of log-likelihood of DCC-GARCH model (vector)

`dcc_loglikelihood`	*This function calculates log-likelihood of DCC-GARCH model.*

Description

This function calculates log-likelihood of DCC-GARCH model.

Usage

```
dcc_loglikelihood(param, ht, residuals, stdresids, uncR)
```

Arguments

`param`	DCC-GARCH parameters(alpha,beta)
`ht`	matrix of conditional variance vectors (T by N)
`residuals`	matrix of residual(de-mean) returns (T by N)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)

Value

log-likelihood of DCC-GARCH model (scaler)

`dcc_optim` *This function optimizes log-likelihood of DCC-GARCH model.*

Description

This function optimizes log-likelihood of DCC-GARCH model.

Usage

```
dcc_optim(param, ht, residuals, stdresids, uncR)
```

Arguments

`param`	DCC-GARCH parameters(alpha,beta)
`ht`	matrix of conditional variance vectors (T by N)
`residuals`	matrix of residual(de-mean) returns (T by N)
`stdresids`	matrix of standrdized(De-GARCH) residual returns (T by N)
`uncR`	unconditional correlation matrix of stdresids (N by N)

Value

results of optimization

`us_stocks` *the closing price data of us stocks in SP500 index from 2006-03-31 to 2014-03-31 from yahoo finance.*

Description

the closing price data of us stocks in SP500 index from 2006-03-31 to 2014-03-31 from yahoo finance.

Format

A data frame with 2013 rows and 460 variables:

Source

Yahoo finance

`xdcclarge` *Package*

Description

Functions for Estimating a (c)DCC-GARCH Model in large dimensions based on a publication by Engle et,al (2017) and Nakagawa et,al (2018). This estimation method is consist of composite likelihood method by Pakel et al. (2014) and (Non-)linear shrinkage estimation of covariance matrices by Ledoit and Wolf (2004,2015,2016).

Details

To estimate the covariance matrix in financial time series, it is necessary consider two important aspects: the cross section and the time series. With regard to the cross section, we have the difficulty of correcting the biases of the sample covariance matrix eigenvalues in a large number of time series. With regard to the time series aspect, we have to account for volatility clustering and time-varying correlations. This package is implemented the improved estimation of the covariance matrix based on the following publications:

- Aielli, Gian Piero. (2013). Dynamic conditional correlation: on properties and estimation. Journal of Business & Economic Statistics 31: 282-99. <doi:10.1080/07350015.2013.771027>
- Engle, Robert F. (2002). Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. Journal of Business & Economic Statistics 20: 339-50. <doi:10.1198/073500102288618487>
- Engle, Robert F, Olivier Ledoit, and Michael Wolf. (2017). Large dynamic covariance matrices. Journal of Business & Economic Statistics, 1-13. <doi:10.1080/07350015.2017.1345683>
- Kei Nakagawa, Mitsuyoshi Imamura and Kenichi Yoshida. (2018). Risk-Based Portfolios with Large Dynamic Covariance Matrices. International Journal of Financial Studies, 6(2), 1-14. <doi:10.3390/ijfs6020052>
- Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. Journal of Multivariate Analysis, 88(2). <doi:10.1016/S0047-259X(03)00096-4>
- Ledoit, O. and Wolf, M. (2012). Nonlinear shrinkage estimation of large-dimensional covariance matrices. Annals of Statistics, 40(2). <doi:10.1214/12-AOS989>
- Ledoit, O. and Wolf, M. (2015). Spectrum estimation: a unified framework for covariance matrix estimation and PCA in large dimensions. Journal of Multivariate Analysis, 139(2). <doi:10.1016/j.jmva.2015.04.006>
- Pakel, Cavit and Shephard, Neil and Sheppard, Kevin and Engle, Robert F. (2014). Fitting vast dimensional time-varying covariance models. Technical report <http://paneldataconference2015.ceu.hu/Program/Cavit-Pakel.pdf>

Index