Package: dccmidas (via r-universe)

October 9, 2026

Type Package

Title DCC Models with GARCH and GARCH-MIDAS Specifications in the Univariate Step, RiskMetrics, Moving Covariance and Scalar and Diagonal BEKK Models

Version 0.1.3

Description Estimates a variety of Dynamic Conditional Correlation (DCC) models. More in detail, the 'dccmidas' package allows the estimation of the corrected DCC (cDCC) of Aielli (2013) <doi:10.1080/07350015.2013.771027>, the DCC-MIDAS of Colacito et al. (2011) <doi:10.1016/j.jeconom.2011.02.013>, the Asymmetric DCC of Cappiello et al. <doi:10.1093/jjfinec/nbl005>, and the Dynamic Equicorrelation (DECO) of Engle and Kelly (2012) <doi:10.1080/07350015.2011.652048>. 'dccmidas' offers the possibility of including standard GARCH <doi:10.1016/0304-4076(86)90063-1>, GARCH-MIDAS <doi:10.1162/REST_a_00300> and Double Asymmetric GARCH-MIDAS <doi:10.1016/j.econmod.2018.07.025> models in the univariate estimation. Moreover, also the scalar and diagonal BEKK <doi:10.1017/S0266466600009063> models can be estimated. Finally, the package calculates also the var-cov matrix under two non-parametric models: the Moving Covariance and the RiskMetrics specifications.

License GPL-3

LinkingTo Rcpp, RcppArmadillo

Encoding UTF-8

LazyData true

RdMacros Rdpack

Depends R (>= 4.0.0)

Imports maxLik (>= 1.3-8), rumidas (>= 0.1.1), rugarch (>= 1.4-4), roll (>= 1.1.4), xts (>= 0.12.0), Rdpack (>= 1.0.0), zoo (>= 1.8.8), stats (>= 4.0.2), utils (>= 4.0.2)

Suggests knitr, rmarkdown

Config/roxygen2/version 8.1.0
NeedsCompilation yes
Author Vincenzo Candila [aut, cre]
Maintainer Vincenzo Candila <vcandila@unisa.it>
Repository https://cran.r-universe.dev
Date/Publication 2026-09-09 15:20:10 UTC
RemoteUrl https://github.com/cran/dccmidas
RemoteRef HEAD
RemoteSha c6ae84e5df4cab784a839bb74ac691ac0d560b61

Contents

bekk_fit 2
cov_eval 4
dcc_fit 5
Det 9
ftse100 9
indpro 10
Inv 11
moving_cov 11
nasdaq 12
plot_dccmidas 13
riskmetrics_mat 14
sp500 15

Index **17**

`bekk_fit` *BEKK fit*

Description

Obtains the estimation the scalar and diagonal BEKK model

Usage

```
bekk_fit(r_t, model = "sBEKK", R = 100, out_of_sample = NULL)
```

Arguments

`r_t`	List of daily returns. At the moment, at most 5 assets can be considered
`model`	Valid choices are: 'sBEKK'(scalar BEKK) and 'dBEKK' (diagonal BEKK)
`R`	**optional** Number of random samples drawn from a Uniform distribution used to inizialize the log-likelihood. Equal to 100 by default
`out_of_sample`	**optional** A positive integer indicating the number of periods before the last to keep for out of sample forecasting

Details

Function bekk_fit implements the estimation of scalar and diagonal BEKK models. For details on BEKK models, see Engle and Kroner (1995)

Value

bekk_fit returns a list containing the following components:

- assets: Names of the assets considered.
- mat_coef: Matrix of estimated coefficients of the model, with the QML standard errors.
- obs: The number of daily observations used for the estimation.
- period: The period of the estimation.
- H_t: Conditional covariance matrix, reported as an array. It refers to the in-sample period.
- est_time: Time of estimation.
- llk: The value of the log-likelihood at the maximum.
- H_t_oos: Conditional covariance matrix, reported as an array, for the out-of-sample period, if the param 'out_of_sample' is used.
- Days: Days of the (in-)sample period.

References

Engle RF, Kroner KF (1995). "Multivariate simultaneous generalized ARCH." *Econometric theory*, **11**(1), 122–150. doi:10.1017/S0266466600009063.

Examples

```
require(xts)
# close to close daily log-returns
r_t_s<-diff(log(sp500['2010/2019'][,3]))
r_t_s[1]<-0
r_t_n<-diff(log(nasdaq['2010/2019'][,3]))
r_t_n[1]<-0
r_t_f<-diff(log(ftse100['2010/2019'][,3]))
r_t_f[1]<-0
db_m<-xts::merge.xts(r_t_s,r_t_n,r_t_f)
db_m<-db_m[complete.cases(db_m),]
colnames(db_m)<-c("S&P500","NASDAQ","FTSE100")
# list of returns
r_t<-list(db_m[,1],db_m[,2],db_m[,3])
bekk_est<-bekk_fit(r_t,model="sBEKK")
bekk_est$mat_coef
```

`cov_eval` *Var-cov matrix evaluation*

Description

Evaluates the estimated var-cov matrix H_t with respect to a covariance proxy, under different robust loss functions (Laurent et al. 2013). The losses considered are also used in Amendola et al. (2020).

Usage

```
cov_eval(H_t, cov_proxy = NULL, r_t = NULL, loss = "FROB")
```

Arguments

`H_t`	Estimated covariance matrix, formatted as array
`cov_proxy`	**optional** Covariance matrix, formatted as array
`r_t`	**optional** List of daily returns used to calculate H_t. If parameter 'cov_proxy' is not provided, then r_t must be included. In this case, a (noise) proxy will be automatically used
`loss`	Robust loss function to use. Valid choices are: "FROB" for Frobenius (by default), "SFROB" for Squared Frobenius, "EUCL" for Euclidean, "QLIKE" for QLIKE and "RMSE" for Root Mean Squared Errors

Value

The value of the loss for each t

References

Amendola A, Braione M, Candila V, Storti G (2020). "A Model Confidence Set approach to the combination of multivariate volatility forecasts." *International Journal of Forecasting*, **36**(3), 873 - 891. doi:10.1016/j.ijforecast.2019.10.001.

Laurent S, Rombouts JV, Violante F (2013). "On loss functions and ranking forecasting performances of multivariate volatility models." *Journal of Econometrics*, **173**(1), 1–10. doi:10.1016/j.jeconom.2012.08.004.

Examples

```
require(xts)
# close to close daily log-returns
r_t_s<-diff(log(sp500['2010/2019'][,3]))
r_t_s[1]<-0
r_t_n<-diff(log(nasdaq['2010/2019'][,3]))
r_t_n[1]<-0
r_t_f<-diff(log(ftse100['2010/2019'][,3]))
```

```
r_t_f[1]<-0
db_m<-xts::merge.xts(r_t_s,r_t_n,r_t_f)
db_m<-db_m[complete.cases(db_m),]
colnames(db_m)<-c("S&P500","NASDAQ","FTSE100")
# list of returns
r_t<-list(db_m[,1],db_m[,2],db_m[,3])
# estimation
K_c<-144
N_c<-36
cdcc_est<-dcc_fit(r_t,univ_model="sGARCH",distribution="norm",
corr_model="DCCMIDAS",N_c=N_c,K_c=K_c)
cov_eval(cdcc_est$H_t,r_t=r_t)[(K_c+1):dim(cdcc_est$H_t)[3]]
```

`dcc_fit` *DCC fit (first and second steps)*

Description

Obtains the estimation of a variety of DCC models, using as univariate models both GARCH and GARCH-MIDAS specifications.

Usage

```
dcc_fit(
  r_t,
  univ_model = "sGARCH",
  distribution = "norm",
  MV = NULL,
  K = NULL,
  corr_model = "cDCC",
  lag_fun = "Beta",
  N_c = NULL,
  K_c = NULL,
  out_of_sample = NULL
)
```

Arguments

`r_t`	List of daily returns on which estimate a DCC model. Each daily return must be an 'xts' object. Note that the sample period of the returns should be the same. Otherwise, a merge is performed
`univ_model`	Specification of the univariate model. Valid choices are: some of the specifications used in the `rugarch` (ugarchspec) and `rumidas` (ugmfit) packages. More in detail, the models coming from `rugarch` are: model Valid models (currently implemented) are 'sGARCH', 'eGARCH', 'gjrGARCH', 'iGARCH', and 'csGARCH'. The models implemented from `rumidas` are: 'GM_skew','GM_noskew', 'DAGM_skew', and 'DAGM_noskew'

distribution	**optional** Distribution chosen for the univariate estimation. Valid choices are: "norm" (by default) and "std", respectively, for the Normal and Student's t distributions
MV	**optional** MIDAS variable to include in the univariate estimation, if the model specificied is a GARCH-MIDAS (GM, Engle et al. (2013)) or a Double Asymmetric GM (DAGM, Engle et al. (2013)). In the case of MIDAS-based models, please provide a list of the MIDAS variables obtained from the mv_into_mat function. If the same MV variable is used, then provide always a list, with the same (transformed) variable repeated
K	**optional** The number of lagged realization of MV variable to use, if 'univ_model' has a MIDAS term
corr_model	Correlation model used. Valid choices are: "cDCC" (the corrected DCC of Aielli (2013)), "aDCC" (the asymmetric DCC model of Cappiello et al. (2006)), "DECO" (Dynamic equicorrelation of Engle and Kelly (2012)), and "DCCMIDAS" (the DCC-MIDAS of Colacito et al. (2011)). By detault, it is "cDCC"
lag_fun	**optional**. Lag function to use. Valid choices are "Beta" (by default) and "Almon", for the Beta and Exponential Almon lag functions, respectively, if 'univ_model' has a MIDAS term and/or if 'corr_model' is "DCCMIDAS"
N_c	**optional** Number of (lagged) realizations to use for the standarized residuals forming the long-run correlation, if 'corr_model' is "DCCMIDAS"
K_c	**optional** Number of (lagged) realizations to use for the long-run correlation, if 'corr_model' is "DCCMIDAS"
out_of_sample	**optional** A positive integer indicating the number of periods before the last to keep for out of sample forecasting

Details

Function dcc_fit implements the two-steps estimation of the DCC models. In the first step, a variety of univariate models are considered. These models can be selected using for the parameter 'univ_model' one of the following choices: 'sGARCH' (standard GARCH of Bollerslev (1986)), 'eGARCH' of Nelson (1991), 'gjrGARCH' of Glosten et al. (1993), 'iGARCH' (Integrated GARCH of Engle and Bollerslev (1986)), 'csGARCH' (the Component GARCH of Engle and Lee (1999)), 'GM_noskew' and 'GM_skew' (the GARCH-MIDAS model of Engle et al. (2013), respectively, without and with the asymmetric term in the short-run component), and 'DAGM_noskew' and 'DAGM_skew' (the Double Asymmetric GARCH-MIDAS model of Amendola et al. (2019), respectively, without and with the asymmetric term in the short-run component).

Value

dcc_fit returns an object of class 'dccmidas'. The function summary.dccmidas can be used to print a summary of the results. Moreover, an object of class 'dccmidas' is a list containing the following components:

- assets: Names of the assets considered.
- model: Univariate model used in the first step.
- est_univ_model: List of matrixes of estimated coefficients of the univariate model, with the QML (Bollerslev and Wooldridge 1992) standard errors.

- corr_coef_mat: Matrix of estimated coefficients of the correlation model, with the QML standard errors.
- mult_model: Correlation model used in the second step.
- obs: The number of daily observations used for the estimation.
- period: The period of the (in-sample) estimation.
- H_t: Conditional covariance matrix, reported as an array.
- R_t: Conditional correlation matrix, reported as an array.
- R_t_bar: Conditional long-run correlation matrix, reported as an array, if the correlation matrix includes a MIDAS specification.
- H_t_oos: Conditional covariance matrix, reported as an array, for the out-of-sample period, if present.
- R_t_oos: Conditional correlation matrix, reported as an array, for the out-of-sample period, if present.
- R_t_bar_oos: Conditional long-run correlation matrix, reported as an array, if the correlation matrix includes a MIDAS specification, for the out-of-sample period, if present.
- est_time: Time of estimation.
- Days: Days of the (in-)sample period.
- llk: The value of the log-likelihood (for the second step) at the maximum.

References

Aielli GP (2013). "Dynamic conditional correlation: on properties and estimation." *Journal of Business & Economic Statistics*, **31**(3), 282–299. doi:10.1080/07350015.2013.771027.

Amendola A, Candila V, Gallo GM (2019). "On the asymmetric impact of macro–variables on volatility." *Economic Modelling*, **76**, 135–152. doi:10.1016/j.econmod.2018.07.025.

Bollerslev T (1986). "Generalized autoregressive conditional heteroskedasticity." *Journal of Econometrics*, **31**(3), 307–327. doi:10.1016/03044076(86)900631.

Bollerslev T, Wooldridge JM (1992). "Quasi-maximum likelihood estimation and inference in dynamic models with time-varying covariances." *Econometric Reviews*, **11**, 143–172. doi:10.1080/07474939208800229.

Cappiello L, Engle RF, Sheppard K (2006). "Asymmetric dynamics in the correlations of global equity and bond returns." *Journal of Financial Econometrics*, **4**(4), 537–572. doi:10.1093/jjfinec/nbl005.

Colacito R, Engle RF, Ghysels E (2011). "A component model for dynamic correlations." *Journal of Econometrics*, **164**(1), 45–59. doi:10.1016/j.jeconom.2011.02.013.

Engle R, Kelly B (2012). "Dynamic equicorrelation." *Journal of Business & Economic Statistics*, **30**(2), 212–228. doi:10.1080/07350015.2011.652048.

Engle RF, Bollerslev T (1986). "Modelling the persistence of conditional variances." *Econometric*

Reviews, **5**(1), 1–50. doi:10.1080/07474938608800095.

Engle RF, Ghysels E, Sohn B (2013). "Stock market volatility and macroeconomic fundamentals." *Review of Economics and Statistics*, **95**(3), 776–797. doi:10.1162/REST_a_00300.

Engle RF, Lee GJ (1999). "A Long-run and Short-run Component Model of Stock Return Volatility." In Engle RF, White H (eds.), *Cointegration, Causality, and Forecasting: A Festschrift in Honor of Clive W. J. Granger*, 475–497. Oxford University Press, Oxford.

Glosten LR, Jagannathan R, Runkle DE (1993). "On the relation between the expected value and the volatility of the nominal excess return on stocks." *The Journal of Finance*, **48**(5), 1779–1801. doi:10.1111/j.15406261.1993.tb05128.x.

Nelson DB (1991). "Conditional heteroskedasticity in asset returns: A new approach." *Econometrica*, **59**(2), 347–370. doi:10.2307/2938260.

Examples

```
# This example is computationally intensive because it estimates
# three univariate GARCH-MIDAS models followed by a DCC-MIDAS model
require(xts)
# daily log-returns
# close to close daily log-returns
r_t_s<-diff(log(sp500['2018/2019'][,3]))
r_t_s[1]<-0
r_t_n<-diff(log(nasdaq['2018/2019'][,3]))
r_t_n[1]<-0
r_t_f<-diff(log(ftse100['2018/2019'][,3]))
r_t_f[1]<-0
db_m<-xts::merge.xts(r_t_s,r_t_n,r_t_f)
db_m<-db_m[complete.cases(db_m),]
colnames(db_m)<-c("S&P500","NASDAQ","FTSE100")
# list of returns
r_t<-list(db_m[,1],db_m[,2],db_m[,3])
# MV transformation (same MV for all the stocks)
require(rumidas)
mv_m<-mv_into_mat(r_t[[1]],diff(indpro),K=24,"monthly")
# list of MV
MV<-list(mv_m,mv_m,mv_m)
# estimation
K_c<-36
N_c<-18
set.seed(123)
dccmidas_est<-dcc_fit(r_t,univ_model="GM_noskew",distribution="norm",
MV=MV,K=24,corr_model="DCCMIDAS",N_c=N_c,K_c=K_c)
dccmidas_est
summary.dccmidas(dccmidas_est)
```

Det *Matrix determinant*

Description

Calculates the determinant of a numeric matrix.

Usage

```
Det(x)
```

Arguments

x	a numeric matrix

Value

The determinant of x.

Examples

```
x<-matrix(sample(1:25,25,replace=TRUE),ncol=5)
Det(x)
```

ftse100 *FTSE 100 data*

Description

Daily data on FTSE 100 collected from the realized library of the Oxford-Man Institute (Heber et al. 2009).

Usage

```
data(ftse100)
```

Format

An object of class "xts".

Details

ftse100 includes the open price (open_price), the realized variance (rv5), and the close price (close_price). The realized variance has been calculated using intradaily intervals of five minutes (Andersen and Bollerslev 1998).

Source

Realized library of the Oxford-Man Institute

References

Andersen TG, Bollerslev T (1998). "Answering the Skeptics: Yes, Standard Volatility Models do Provide Accurate Forecasts." *International Economic Review*, **39**, 885–905. doi:10.2307/2527343.

Heber G, Lunde A, Shephard N, Sheppard K (2009). "OMI's realised library, version 0.1." Oxford–Man Institute, University of Oxford.

Examples

```
head(ftse100)
summary(ftse100)
```

---

indpro *Monthly U.S. Industrial Production*

Description

Monthly data on the U.S. Industrial Production index (IP, index 2012=100, seasonally adjusted) collected from the Federal Reserve Economic Data (FRED) archive. The IP has been used as MIDAS term in different contributions (see, for instance, Engle et al. (2013), Conrad and Loch (2015), and Amendola et al. (2017)).

Usage

```
data(indpro)
```

Format

An object of class `"xts"`.

Source

Archive of the Federal Reserve Economic Data (FRED)

References

Amendola A, Candila V, Scognamillo A (2017). "On the influence of US monetary policy on crude oil price volatility." *Empirical Economics*, **52**(1), 155–178. doi:10.1007/s0018101610695.

Conrad C, Loch K (2015). "Anticipating Long-Term Stock Market Volatility." *Journal of Applied Econometrics*, **30**(7), 1090–1114. doi:10.1002/jae.2404.

Engle RF, Ghysels E, Sohn B (2013). "Stock market volatility and macroeconomic fundamentals." *Review of Economics and Statistics*, **95**(3), 776–797. doi:10.1162/REST_a_00300.

Examples

```
head(indpro)
summary(indpro)
plot(indpro)
```

Inv *Inverse of a matrix*

Description

Calculates the inverse of a numeric matrix

Usage

```
Inv(x)
```

Arguments

x	a numeric matrix

Value

The inverse of x.

Examples

```
x<-matrix(sample(1:25,25,replace=TRUE),ncol=5)
Inv(x)
```

---

moving_cov *Moving Covariance model*

Description

Obtains the matrix H_t, under the Moving Covariance model.

Usage

```
moving_cov(r_t, V = 22)
```

Arguments

r_t	List of daily returns
V	Length of the rolling window adopted. By default, V is 22

Value

A list with the H_t matrix, for each t.

Examples

```
require(xts)
# close to close daily log-returns
r_t_s<-diff(log(sp500['2010/2019'][,3]))
r_t_s[1]<-0
r_t_n<-diff(log(nasdaq['2010/2019'][,3]))
r_t_n[1]<-0
r_t_f<-diff(log(ftse100['2010/2019'][,3]))
r_t_f[1]<-0
db_m<-merge.xts(r_t_s,r_t_n,r_t_f)
db_m<-db_m[complete.cases(db_m),]
colnames(db_m)<-c("S&P500","NASDAQ","FTSE100")
# list of returns
r_t<-list(db_m[,1],db_m[,2],db_m[,3])
MC<-moving_cov(r_t,V=60)
```

---

nasdaq *NASDAQ data*

Description

Daily data on NASDAQ collected from the realized library of the Oxford-Man Institute (Heber et al. 2009).

Usage

```
data(nasdaq)
```

Format

An object of class "xts".

Details

nasdaq includes the open price (open_price), the realized variance (rv5), and the close price (close_price). The realized variance has been calculated using intradaily intervals of five minutes (Andersen and Bollerslev 1998).

Source

Realized library of the Oxford-Man Institute

References

Andersen TG, Bollerslev T (1998). "Answering the Skeptics: Yes, Standard Volatility Models do Provide Accurate Forecasts." *International Economic Review*, **39**, 885–905. doi:10.2307/2527343.

Heber G, Lunde A, Shephard N, Sheppard K (2009). "OMI's realised library, version 0.1." Oxford–Man Institute, University of Oxford.

Examples

```
head(nasdaq)
summary(nasdaq)
```

`plot_dccmidas` *Plot method for 'dccmidas' class*

Description

Plots of the conditional volatilities on the main diagonal and of the conditional correlations on the extra-diagonal elements.

Usage

```
plot_dccmidas(
  x,
  K_c = NULL,
  vol_col = "black",
  long_run_col = "red",
  cex_axis = 0.75,
  LWD = 2,
  asset_sub = NULL
)
```

Arguments

`x`	An object of class 'dccmidas', that is the result of a call to `dcc_fit`.
`K_c`	**optional** Number of (lagged) realizations to use for the long-run correlation, , if 'corr_model' is "DCCMIDAS"
`vol_col`	**optional** Color of the volatility and correlation plots. "black" by default
`long_run_col`	**optional** Color of the long-run correlation plots, if present. "red" by default
`cex_axis`	**optional** Size of the x-axis. Default to 0.75
`LWD`	**optional** Width of the plotted lines. Default to 2
`asset_sub`	**optional** Numeric vector of selected assets to consider for the plot. NULL by default

Value

No return value, called for side effects

Examples

```
require(xts)
# close to close daily log-returns
r_t_s<-diff(log(sp500['2010/2019'][,3]))
r_t_s[1]<-0
r_t_n<-diff(log(nasdaq['2010/2019'][,3]))
r_t_n[1]<-0
r_t_f<-diff(log(ftse100['2010/2019'][,3]))
r_t_f[1]<-0
db_m<-xts::merge.xts(r_t_s,r_t_n,r_t_f)
db_m<-db_m[complete.cases(db_m),]
colnames(db_m)<-c("S&P500","NASDAQ","FTSE100")
# list of returns
r_t<-list(db_m[,1],db_m[,2],db_m[,3])
# estimation
K_c<-144
N_c<-36
cdcc_est<-dcc_fit(r_t,univ_model="sGARCH",distribution="norm",
corr_model="DCCMIDAS",N_c=N_c,K_c=K_c)
plot_dccmidas(cdcc_est,K_c=144)
```

---

riskmetrics_mat *RiskMetrics model*

Description

Obtains the matrix H_t, under the RiskMetrics model.

Usage

```
riskmetrics_mat(r_t, lambda = 0.94)
```

Arguments

r_t	List of daily returns
lambda	**optional** Decay parameter. Default to 0.94

Value

A list with the H_t matrix, for each t.

Examples

```
# close to close daily log-returns
r_t_s <- diff(log(sp500['2010/2019'][,3]))
r_t_s[1] <- 0

r_t_n <- diff(log(nasdaq['2010/2019'][,3]))
r_t_n[1] <- 0

r_t_f <- diff(log(ftse100['2010/2019'][,3]))
r_t_f[1] <- 0

db_m <- xts::merge.xts(r_t_s, r_t_n, r_t_f)
db_m <- db_m[complete.cases(db_m),]
colnames(db_m) <- c("S&P500", "NASDAQ", "FTSE100")

# list of returns
r_t <- list(db_m[,1], db_m[,2], db_m[,3])
RM <- riskmetrics_mat(r_t)
```

---

sp500 *S&P 500 data*

Description

Daily data on S&P 500 collected from the realized library of the Oxford-Man Institute (Heber et al. 2009).

Usage

```
data(sp500)
```

Format

An object of class "xts".

Details

sp500 includes the open price (open_price), the realized variance (rv5), and the close price (close_price). The realized variance has been calculated using intradaily intervals of five minutes (Andersen and Bollerslev 1998).

Source

Realized library of the Oxford-Man Institute

References

Andersen TG, Bollerslev T (1998). "Answering the Skeptics: Yes, Standard Volatility Models do Provide Accurate Forecasts." *International Economic Review*, **39**, 885–905. doi:10.2307/2527343.

Heber G, Lunde A, Shephard N, Sheppard K (2009). "OMI's realised library, version 0.1." Oxford–Man Institute, University of Oxford.

Examples

```
head(sp500)
summary(sp500)
```

Index

∗ **datasets**
ftse100, 9
indpro, 10
nasdaq, 12
sp500, 15

bekk_fit, 2

cov_eval, 4

dcc_fit, 5, *13*
Det, 9

ftse100, 9

indpro, 10
Inv, 11

moving_cov, 11
mv_into_mat, *6*

nasdaq, 12

plot_dccmidas, 13

riskmetrics_mat, 14

sp500, 15
summary.dccmidas, *6*

ugarchspec, *5*
ugmfit, *5*