Package ‘rumidas’

September 22, 2020

Title Univariate GARCH-MIDAS, Double-Asymmetric GARCH-MIDAS and MEM-MIDAS

Version 0.1.0

Description Adds the MIxing Data Sampling (MIDAS, Ghysels et al. (2007) <doi:10.1080/07474930600972467>) components within the GARCH (Engle et al. (2013) <doi:10.1162/REST_a_00300>) and MEM (Engle (2002) <doi:10.1002/jae.683>) frameworks, with the aim of predicting the volatility with additional low-frequency (that is, MIDAS) terms. The estimation takes place through simple functions, which provide in-sample and (if present) and out-of-sample evaluations. 'rumidas' also offers a summary tool, which synthesizes the main information of the estimated model. Finally, an option to generate one-step-ahead volatility forecasts automatically divides the whole period into a training and testing samples.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

RdMacros Rdpack

Depends R (>= 4.0.0), maxLik (>= 1.3-8)

Imports highfrequency (>= 0.6.5), roll (>= 1.1.4), xts (>= 0.12.0), tseries (>= 0.10.47), Rdpack (>= 1.0.0), lubridate (>= 1.7.9), zoo (>= 1.8.8), stats (>= 4.0.2), utils (>= 4.0.2)

Suggests knitr, rmarkdown

NeedsCompilation no

Author Vincenzo Candila [aut, cre]

Maintainer Vincenzo Candila <vincenzo.candila@uniroma1.it>

Repository CRAN

Date/Publication 2020-09-22 08:50:02 UTC

R topics documented:

`beta_function` *Beta function*

Description

Represents a tool able to accommodate various lag structures for the additional MIDAS variable observed each "low-frequency" period t. It can have a monotonically increasing, decreasing weighting scheme or a hump-shaped weighting scheme. The Beta function is:

$$\delta_k(\omega) = \frac{(k/K)^{\omega_1-1}(1-k/K)^{\omega_2-1}}{\sum_{j=1}^{K}(j/K)^{\omega_1-1}(1-j/K)^{\omega_2-1}}.$$

For additional details, see Ghysels et al. (2007).

Usage

```
beta_function(k, K, w1, w2)
```

Arguments

`k`	Lag of interest.
`K`	Number of (lagged) realizations to consider.
`w1, w2`	Parameters governing the weights of each k lag.

Value

The weights associated to each lag k, with $k = 1, \cdots, K$.

References

Ghysels E, Sinko A, Valkanov R (2007). "MIDAS regressions: Further results and new directions." *Econometric Reviews*, **26**(1), 53–90. doi: 10.1080/07474930600972467.

Examples

```
# suppose to have four lags:
# K<-5 # Note: the number of lags has to be increased by one
# w1<-1 # by setting w1=1, only a monotonically decreasing weighting scheme is allowed
#(more recent observations weigh more)
# w2<-5
beta_function(1:5,K=5,w1=1,w2=5)
```

`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)
```

mv_into_mat *MIDAS variable matrix transformation*

Description

Implements the trasformation of the MIDAS variable into a matrix, whose dimension is $(K+1)\times N$, where K is the number of lagged realizations to consider and N is the length of the variable x.

Usage

```
mv_into_mat(x, mv, K, type)
```

Arguments

x	Variable according to which the MIDAS term has to be aligned. It must be an 'xts' object.
mv	MIDAS variable, observed each period t. It must be an 'xts' object.
K	Number of (lagged) realizations of the MIDAS variable to consider.
type	The frequency of the period of observations for the MIDAS variable. It can be 'weekly', 'monthly' or 'quarterly'.

Value

The resulting matrix has as many rows as the number of lagged realizations (plus one) of the MIDAS variable to consider, and as many columns as the length of x.

Examples

```
# weekly frequency
# obtain weekly MIDAS variable after daily aggregation
# RV_weekly_sum<-apply.weekly(rv5^0.5,sum) #realized volatility
# then allocate correctly the information
# RV_weekly<-as.xts(coredata(RV_weekly_sum),seq(as.Date("2000-01-10"),
# by = "week", length.out = length(RV_weekly_sum)))
# use mv_into_mat (two cases, the second one does not work)
# mv_into_mat(sp500['2002/2003-12-26'],diff(RV_weekly['/2003-12']),K=4,"weekly")
# mv_into_mat(sp500['2002/2003-12-26'],diff(RV_weekly['/2005-12']),K=4,"weekly") #does not work

# monthly frequency
# r_t<-sp500['2005/2010']
# mv_into_mat(r_t,diff(indpro),K=12,"monthly")

# quarterly frequency
# RV_quarterly_sum<-apply.quarterly(rv5,sum)
# RV_quarterly<-as.xts(coredata(RV_quarterly_sum),seq(as.Date("2000-04-01"),
# by = "quarter", length.out = length(RV_quarterly_sum)))
# mv_into_mat(sp500['2004/2010'],diff(RV_quarterly),K=10,"quarterly")
```

rv5 — *S&P 500 realized variance at 5-minutes*

Description

Daily data on the realized variance of the S&P 500 collected from the realized library of the Oxford-Man Institute (Heber et al. 2009). The realized variance has been calculated using intradaily intervals of five minutes (Andersen and Bollerslev 1998).

Usage

```
data(rv5)
```

Format

An object of class "xts".

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(rv5)
summary(rv5)
plot(rv5)
```

sp500 — *S&P 500 open-to-close daily log-returns*

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".

Source

Realized library of the Oxford-Man Institute

References

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)
plot(sp500)
```

---

summary.rumidas *Summary method for 'rumidas' class*

Description

Summary method for 'rumidas' class

Usage

```
## S3 method for class 'rumidas'
summary(object, ...)
```

Arguments

object	An object of class 'rumidas', that is the result of a call to ugmfit or umemfit.
...	Additional arguments affecting the summary produced.

Examples

```
# r_t<-sp500['2003/2010']
# real<-(rv5['2003/2010'])^0.5 # realized volatility
# fit<-umemfit(model="MEM",skew="NO",x=real)
# summary.rumidas(fit)
```

ugmfit — *Methods for obtaining (and evaluating) a variety of GARCH-MIDAS-based models*

Description

Estimates several GARCH-MIDAS-based models, according to two errors' conditional distributions: Normal and Student-t, and the presence of asymmetric terms in the short- and long-run components.

Usage

```
ugmfit(
  model,
  skew,
  distribution,
  daily_ret,
  mv_m,
  K,
  out_of_sample = NULL,
  vol_proxy = NULL
)
```

Arguments

`model`	Model to estimate. Valid choices are: "GM" for GARCH-MIDAS, "DAGM" for Double Asymmetric GARCH-MIDAS.
`skew`	The skewness parameter to include in the short-run equation. Valid choices are: "YES" and "NO".
`distribution`	The conditional density to use for the innovations. At the moment, valid choices are "norm" and "std", for the Normal and Student-t distribution, respectively.
`daily_ret`	Daily returns, which must be an "xts" object.
`mv_m`	MIDAS variable already transformed into a matrix, through `mv_into_mat` function.
`K`	Number of (lagged) realizations of the MIDAS variable to consider.
`out_of_sample`	**optional**. A positive integer indicating the number of periods before the last to keep for out of sample forecasting.
`vol_proxy`	**optional**. If present, the vol_proxy is the volatilty proxy used for the in-sample and out-of-sample (again, if present) evaluation. It could be the realized variance. If it left unspecified, vol_proxy is replaced by the squared daily returns.

Details

Function `ugmfit` implements the estimation and evaluation of the GARCH-MIDAS-based models, with and without the asymmetric term linked to negative lagged daily returns, according to two distributions for the error term. The general framework assumes that:

$$r_{i,t} = \sqrt{\tau_t \times g_{i,t}} \epsilon_{i,t},$$

where

- $r_{i,t}$ is the daily return for the i-th day ($i = 1, \ldots, N_t$) of the period t (for example, a week, a month or a quarter; $t = 1, \ldots, T$);
- τ_t is the long-run component, varying each period t;
- $g_{i,t}$ is the short-run term, varying each day i of the period t;
- $\epsilon_{i,t}$ is an iid error term which has a zero mean and unit variance.

The short-run component of the GARCH-MIDAS (parameter "model" set to "GM") and DAGM, when the parameter "skew" is "YES", is:

$$g_{i,t} = (1 - \alpha - \gamma/2 - \beta) + \left(\alpha + \gamma \cdot I_{(r_{i-1,t}<0)}\right) \frac{(r_{i-1,t})^2}{\tau_t} + \beta g_{i-1,t},$$

where $I_{(\cdot)}$ is an indicator function. When, for both the models, the parameter "skew" is set to "NO", γ disappears. The long-run component of the GARCH-MIDAS is:

$$\tau_t = \exp\left\{ m + \theta \sum_{j=1}^{K} \delta_j(\omega) X_{t-j} \right\},$$

where X_t is the MIDAS term. The long-run component of the DAGM model is:

$$\tau_t = \exp\left(m + \theta^+ \sum_{k=1}^{K} \delta_k(\omega)^+ X_{t-k} I_{(X_{t-k} \geq 0)} + \theta^- \sum_{k=1}^{K} \delta_k(\omega)^- X_{t-k} I_{(X_{t-k}<0)} \right)$$

Value

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

- model: The model used for the estimation.
- rob_coef_mat: The matrix of estimated coefficients, with the QML standard errors. For details, see: Bollerslev and Wooldridge (1992).
- obs: The number of daily observations used for the (in-sample) estimation.
- period: The period of the in-sample estimation.
- loglik: The value of the log-likelihood at the maximum.
- inf_criteria: The AIC and BIC information criteria.
- loss_in_s: The in-sample MSE and QLIKE averages, calculated considering the distance with respect to the volatility proxy (if provided) or the squared daily returns.

- est_in_s: The in-sample predicted volatility, that is: $\sqrt{\hat{\tau}_t \times \hat{g}_{i,t}}$.
- est_lr_in_s: The in-sample predicted long-run component of the dependent variable.
- loss_oos: The out-of-sample MSE and QLIKE averages, calculated considering the distance with respect to the volatility proxy (if provided) or the squared daily returns.
- est_oos: The out-of-sample predicted dependent variable, that is: $\sqrt{\hat{\tau}_t \times \hat{g}_{i,t}}$.
- est_lr_oos: The out-of-sample predicted long-run component of the dependent variable.

See Also

mv_into_mat.

Examples

```
# estimate a GARH-MIDAS model, without the skewness parameter
# r_t<-sp500['2008']
# mv_m<-mv_into_mat(r_t,diff(indpro),K=12,"monthly")
# fit<-ugmfit(model="GM",skew="NO",distribution="norm",r_t,mv_m,K=12)
# fit
# summary.rumidas(fit)
# names(fit)

# to see the estimated coefficients with the QML standard errors:
# fit$rob_coef_mat

# estimate a DAGM model, with the skewness parameter,
# including the volatility proxy (realized variance), and
# leaving the last 100 observations for the out-of-sample evaluation
# r_t<-sp500['2002/2020']
# mv_m<-mv_into_mat(r_t,diff(indpro),K=12,"monthly")
# fit_2<-ugmfit(model="DAGM",skew="YES",distribution="norm",r_t,
# mv_m,K=12,vol_proxy=rv5['2002/2020'],out_of_sample=100)
# fit_2
# summary.rumidas(fit_2)
```

---

umemfit *Methods for obtaining (and evaluating) a variety of MEM(-MIDAS)-based models*

Description

Estimates several MEM-MIDAS-based models.

Usage

```
umemfit(
  model,
  skew,
  x,
  daily_ret = NULL,
  mv_m = NULL,
  K = NULL,
  out_of_sample = NULL
)
```

Arguments

`model`	Model to estimate. Valid choices are: "MEMMIDAS" for MEM-MIDAS, "MEM" for base MEM.
`skew`	The skewness parameter linked to lagged daily returns. Valid choices are: "YES" and "NO".
`x`	Dependent variable to predict. Usually the realized volatility. It must be positive and "xts" object.
`daily_ret`	**optional**. Daily returns, which must be an "xts" object. NULL by default.
`mv_m`	**optional**. MIDAS variable already transformed into a matrix, through `mv_into_mat` function. NULL by default.
`K`	**optional**. Number of (lagged) realizations of the MIDAS variable to consider. NULL 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 `umemfit` implements the estimation and evaluation of the MEM and MEM-MIDAS models, with and without the asymmetric term linked to negative lagged daily returns. The general framework assumes that:

$$x_{i,t} = \mu_{i,t}\epsilon_{i,t} = \tau_t\xi_{i,t}\epsilon_{i,t},$$

where

- $x_{i,t}$ is a time series coming from a non-negative discrete time process for the i-th day ($i = 1, \ldots, N_t$) of the period t (for example, a week, a month or a quarter; $t = 1, \ldots, T$);
- τ_t is the long-run component, determining the average level of the conditional mean, varying each period t;
- $\xi_{i,t}$ is a factor centered around one, labelled as the short-run term, which plays the role of dumping or amplifying $\tau_{i,t}$;
- $\epsilon_{i,t}$ is an iid error term which, conditionally on the information set, has a unit mean, an unknown variance, and a probability density function defined over a non-negative support.

The short-run component of the MEM-MIDAS is:

$$\xi_{i,t} = (1 - \alpha - \gamma/2 - \beta) + \left(\alpha + \gamma \cdot I_{(r_{i-1,t}<0)}\right) \frac{x_{i-1,t}}{\tau_t} + \beta \xi_{i-1,t},$$

where $I_{(\cdot)}$ is an indicator function and $r_{i,t}$ is the daily return of the day i of the period t. The long-run component of the MEM-MIDAS is:

$$\tau_t = \exp\left\{ m + \theta \sum_{k=1}^{K} \delta_k(\omega) X_{t-k} \right\},$$

where X_t is the MIDAS term. When the "skew" parameter is set to "NO", γ disappears. The MEM model does not have the difference between the long- and short-run components. Therefore, it directly evolves according to $\mu_{i,t}$. When the "skew" parameter is present:

$$\mu_{i,t} = (1 - \alpha - \gamma_1/2 - \beta)\, \mu + (\alpha + \gamma I_{(r_{i-1,t}<0)}) x_{i-1,t} + \beta \mu_{i-1,t},$$

where $\mu = E(x_{i,t})$. When the "skew" parameter is set to "NO", in the previous equation γ cancels.

Value

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

- model: The model used for the estimation.
- rob_coef_mat: The matrix of estimated coefficients, with the QML standard errors. For details, see: Bollerslev and Wooldridge (1992), Engle and Gallo (2006), and Amendola et al. (2020).
- obs: The number of daily observations used for the (in-sample) estimation.
- period: The period of the in-sample estimation.
- loglik: The value of the log-likelihood at the maximum.
- inf_criteria: The AIC and BIC information criteria.
- loss_in_s: The in-sample MSE and QLIKE averages, calculated considering the distance with respect to the dependent variable.
- est_in_s: The in-sample predicted dependent variable.
- est_lr_in_s: The in-sample predicted long-run component (if present) of the dependent variable.
- loss_oos: The out-of-sample MSE and QLIKE averages, calculated considering the distance with respect to the dependent variable.
- est_oos: The out-of-sample predicted dependent variable.
- est_lr_oos: The out-of-sample predicted long-run component (if present) of the dependent variable.

See Also

`mv_into_mat`.

Examples

```
# estimate the base MEM, without the asymmetric term linked to negative lagged returns
real<-(rv5['2003/2010'])^0.5 # realized volatility
fit<-umemfit(model="MEM",skew="NO",x=real)
fit
summary.rumidas(fit)
# to see the estimated coefficients with the QML standard errors:
fit$rob_coef_mat

# All the other elements of fit are:
names(fit)

# estimate the MEM-MIDAS, with the asymmetric term linked to negative lagged returns,
# leaving the last 200 observations for the out-of-sample analysis
# r_t<-sp500['2003/2010']
# real<-(rv5['2003/2010'])^0.5 # realized volatility
# mv_m<-mv_into_mat(real,diff(indpro),K=12,"monthly")
# fit_2<-umemfit(model="MEMMIDAS",skew="YES",x=real,daily_ret=r_t,mv_m=mv_m,K=12,out_of_sample=200)
# fit_2
# summary.rumidas(fit_2)
# to see the estimated coefficients with the QML standard errors:
# fit_2$rob_coef_mat
```

Index