Package: BLModel (via r-universe)

September 13, 2024

Title Black-Litterman Posterior Distribution

Version 1.0.2

Description Posterior distribution in the Black-Litterman model is computed from a prior distribution given in the form of a time series of asset returns and a continuous distribution of views provided by the user as an external function.

Depends R (>= 3.3.0)

License GNU General Public License version 3

Encoding UTF-8

LazyData true

Author Andrzej Palczewski [aut, cre], Jan Palczewski [aut], Alicja Gosiewska [ctb]

Maintainer Andrzej Palczewski <A.Palczewski@mimuw.edu.pl>

RoxygenNote 6.0.1

Suggests mvtnorm, testthat

NeedsCompilation no

Repository CRAN

Date/Publication 2017-03-29 06:38:36 UTC

Contents

BL_post_distr *Computes the Black-Litterman posterior distribution.*

Description

BL_post_distr computes posterior distribution in the Black-Litterman model starting from arbitrary prior distribution given as a discrete time series `dat` and using `views_distr` – submitted by the user distribution of views.

Usage

```
BL_post_distr (dat, returns_freq, prior_type = c("elliptic", NULL), market_portfolio,
SR, P, q, tau, risk = c("CVAR", "DCVAR", "LSAD", "MAD"),  alpha = NULL,
views_distr, views_cov_matrix_type = c("diag", "full"), cov_matrix = NULL)
```

Arguments

`dat` Time series of returns data; dat = cbind(rr, pk), where rr is an array (time series) of market asset returns, for n returns and k assets it is an array with $\dim(rr) = (n, k)$, pk is a vector of length n containing probabilities of returns.

`returns_freq` Frequency of data in time series `dat`; given as a number of data rows corresponding to the period of 1 year, i.e. 52 for weekly data or 12 for monthly data.

`prior_type` Type of distribution in time series `dat`; can be "elliptic" – rr is distributed according to (any) elliptical distribution, NULL – rr is distributed according to any non-elliptical distribution.

`market_portfolio` Market portfolio – benchmark (equilibrium) portfolio (for details see Palczewski&Palczewski).

`SR` Benchmark Sharpe ratio.

`P` "Pick" matrix in the Black-Litterman model (see Palczewski&Palczewski).

`q` Vector of investor's views on future returns in the Black-Litterman model (see Palczewski&Palczewski).

`tau` Confidence parameter in the Black-Litterman model.

`risk` Risk measure chosen for optimization; one of "CVAR", "DCVAR", "LSAD", "MAD", where "CVAR" – denotes Conditional Value-at-Risk (CVaR), "DCVAR" – denotes deviation CVaR, "LSAD" – denotes Lower Semi Absolute Deviation, "MAD" – denotes Mean Absolute Deviation.

`alpha` Value of alpha quantile in the definition of risk measures CVAR and DCVAR. Can be any number when risk measure is parameter free.

`views_distr` Distribution of views. An external function submitted by the user which computes densities of the distribution of views in given data points. It is assumed implicitly that this distribution is an elliptical distribution (but any other distribution type can be used provided calling to this function will preserve described below structure). Call to that function has to be of the following form `FUN(x,q,covmat,COF = NULL)`, where `x` is a data points matrix which collects in rows the coordinates of the points in which density is computed, `q` is a vector of investor's views, `covmat` is covariance matrix of the distribution and `COF` is a vector of additional parameters characterizing the distribution (if needed).

`views_cov_matrix_type`
: Type of the covariance matrix of the distribution of views; can be: "diag" – diagonal part of the covariance matrix is used; "full" – the complete covariance matrix is used; (for details see Palczewski&Palczewski).

`cov_matrix`
: Covariance matrix used for computation of market expected return (`RM`) from the formula `RM = SR * sqrt( t(w_m) * cov_matrix * w_m)` where `w_m` is market portfolio and `SR` – benchmark Sharpe ratio. When `cov_matrix` = NULL covariance matrix is computed from matrix rr in data set `dat`.

Value

`post_distr`
: a time series of data for posterior distribution; for a time series of length n and k assets it is a matrix $(n, k + 1)$, where columns (1:k) contain return vectors and the last column probabilities of returns.

References

Palczewski, J., Palczewski, A., Black-Litterman Model for Continuous Distributions (2016). Available at SSRN: https://ssrn.com/abstract=2744621.

Examples

```
library(mvtnorm)
k = 3
num =100
dat <-  cbind(rmvnorm (n=num, mean = rep(0,k), sigma=diag(k)), matrix(1/num,num,1))
# a data sample with num rows and (k+1) columns for k assets;
returns_freq = 52 # we assume that data frequency is 1 week
w_m <- rep(1/k,k) # benchmark portfolio, a vector of length k,
SR = 0.5 # Sharpe ratio
Pe <- diag(k) # we assume that views are "absolute views"
qe <- rep(0.05, k) # user's opinions on future returns (views)
tau = 0.02
BL_post_distr(dat, returns_freq, NULL, w_m, SR, Pe, qe, tau, risk = "MAD", alpha = 0,
views_distr = observ_normal, "diag", cov_matrix = NULL)
```

equilibrium_mean — *Solves the inverse optimization to mean-risk standard optimization problem to find equilibrium returns. The function is invoked by BL_post_distr and arguments are supplemented by BL_post_distr.*

Description

The function computes the vector of equilibrium returns implied by a market portfolio. The vector of means for the mean-risk optimization problem is found by inverse optimization.
The optimization problem is:
$\min F(w_m^T r)$
subject to
$w_m^T E(r) \geq RM$,
where
F is a risk measure – one from the list c("CVAR", "DCVAR", "LSAD", "MAD"),
r is a time series of market returns,
w_m is market portfolio,
RM is market expected return.

Usage

```
equilibrium_mean(dat, w_m, RM, risk = c("CVAR", "DCVAR", "LSAD", "MAD"),
  alpha = 0.95)
```

Arguments

`dat`	Time series of returns data; dat = cbind(rr, pk), where rr is an array (time series) of market asset returns, for n returns and k assets it is an array with $\dim(rr) = (n, k)$, pk is a vector of length n containing probabilities of returns.
`w_m`	Market portfolio.
`RM`	Market_expected_return.
`risk`	A risk measure, one from the list c("CVAR", "DCVAR", "LSAD", "MAD").
`alpha`	Value of alpha quantile in the definition of risk measures CVAR and DCVAR. Can be any number when risk measure is parameter free.

Value

`market_returns`	a vector of market returns obtain by inverse optimization; this is vector $E(r)$ from the description of this function.

References

Palczewski, J., Palczewski, A., Black-Litterman Model for Continuous Distributions (2016). Available at SSRN: https://ssrn.com/abstract=2744621.

Examples

```
# In normal usage all data are supplemented by function BL_post_distr.
library(mvtnorm)
k = 3
num =100
dat <-  cbind(rmvnorm (n=num, mean = rep(0,k), sigma=diag(k)), matrix(1/num,num,1))
# a data sample with num rows and (k+1) columns for k assets;
w_m <- rep(1/k,k) # market portfolio.
RM = 0.05 # market expected return.
equilibrium_mean (dat, w_m, RM, risk = "CVAR", alpha = 0.95)
```

`observ_normal` *Example of distribution of views – normal distribution*

Description

Function observ_normal computes density of normal distribution of views using the formula
$f(x) = c_k * \exp(-((x-q)^T * covmat^{-1} * (x-q))/2)$,
where c_k is a normalization constant (depends on the dimension of x and q).

Usage

```
observ_normal(x, q, covmat)
```

Arguments

`x`	Data points matrix which collects in rows coordinates of points in which distribution density is computed.
`q`	Vector of investor's views.
`covmat`	Covariance matrix of the distribution.

Value

function returns a vector of distribution densities in data points x.

References

Palczewski, J., Palczewski, A., Black-Litterman Model for Continuous Distributions (2016). Available at SSRN: https://ssrn.com/abstract=2744621.

Examples

```
k =3
observ_normal (x = matrix(c(rep(0.5,k),rep(0.2,k)),k,2), q = matrix(0,k,1),
               covmat = diag(k))
```

`observ_powerexp` *Example of distribution of views – power exponential distribution*

Description

Function observ_powerexp computes density of power exponential distribution of views using the formula
$f(x) = c_k * \exp(-((x-q)^T * \Sigma^{-1} * (x-q))^\beta / 2)$,
where c_k is a normalization constant (depends on the dimension of x and q) and Σ is the dispersion matrix.

Usage

```
observ_powerexp(x, q, covmat, beta = 0.6)
```

Arguments

`x`	Data points matrix which collects in rows coordinates of points in which distribution density is computed.
`q`	Vector of investor's views.
`covmat`	Covariance matrix of the distribution; dispersion matrix Σ is computed from `covmat`.
`beta`	Shape parameter of the power exponential distribution.

Value

function returns a vector of distribution densities in data points x.

References

Gomez, E., Gomez-Villegas, M., Marin, J., A multivariate generalization of the power exponential family of distributions. Commun. Statist. Theory Methods, 27 (1998), 589–600. DOI: 10.1080/03610929808832115

Examples

```
k =3
observ_powerexp (x = matrix(c(rep(0.5,k),rep(0.2,k)),k,2), q = matrix(0,k,1),
               covmat = diag(k), beta = 0.6)
```

observ_ts *Example of distribution of views – Student t-distribution*

Description

Function observ_ts computes density of Student t-distribution of views using the formula
$f(x) = c_k * (1 + (x - q)^T * \Sigma^{-1} * (x - q)/df)^{(-(df+k)/2)}$,
where c_k is a normalization constant (depends on the dimension of x and q) and Σ is the dispersion matrix.

Usage

```
observ_ts(x, q, covmat, df = 5)
```

Arguments

x	Data points matrix which collects in rows coordinates of points in which distribution density is computed.
q	Vector of investor's views.
covmat	Covariance matrix of the distribution; dispersion matrix Σ is computed from `covmat`.
df	Number of degrees of freedom of Students t-distribution.

Value

function returns a vector of observation distribution densities in data points x.

References

Kotz, S., Nadarajah, S., Multivariate t Distributions and Their Applications. Cambridge University Press, 2004.

Examples

```
k =3
observ_ts (x = matrix(c(rep(0.5,k),rep(0.2,k)),k,2), q = matrix(0,k,1), covmat = diag(k),
         df=5)
```

Index