Package ‘HDShOP’

January 20, 2025

Title High-Dimensional Shrinkage Optimal Portfolios

Version 0.1.5

Maintainer Dmitry Otryakhin <d.otryakhin.acad@protonmail.ch>

Author Taras Bodnar [aut] (<https://orcid.org/0000-0001-7855-8221>),
Solomiia Dmytriv [aut] (<https://orcid.org/0000-0003-1855-3044>),
Yarema Okhrin [aut] (<https://orcid.org/0000-0003-4704-5233>),
Dmitry Otryakhin [aut, cre] (<https://orcid.org/0000-0002-4700-7221>),
Nestor Parolya [aut] (<https://orcid.org/0000-0003-2147-2288>)

Description
Constructs shrinkage estimators of high-dimensional mean-variance portfolios and performs high-dimensional tests on optimality of a given portfolio. The techniques developed in Bodnar et al. (2018 <doi:10.1016/j.ejor.2017.09.028>, 2019 <doi:10.1109/TSP.2019.2929964>, 2020 <doi:10.1109/TSP.2020.3037369>, 2021 <doi:10.1080/07350015.2021.2004897>) are central to the package. They provide simple and feasible estimators and tests for optimal portfolio weights, which are applicable for 'large p and large n' situations where p is the portfolio dimension (number of stocks) and n is the sample size. The package also includes tools for constructing portfolios based on shrinkage estimators of the mean vector and covariance matrix as well as a new Bayesian estimator for the Markowitz efficient frontier recently developed by Bauder et al. (2021) <doi:10.1080/14697688.2020.1748214>.

License GPL-3

URL https://github.com/Otryakhin-Dmitry/global-minimum-variance-portfolio

BugReports
https://github.com/Otryakhin-Dmitry/global-minimum-variance-portfolio/issues

LazyData yes

Encoding UTF-8

Depends R (>= 3.5.0)

Imports Rdpack, lattice

Suggests ggplot2, testthat, EstimDiagnostics, MASS, corpcor, waldo

RdMacros Rdpack

RoxygenNote 7.3.1

NeedsCompilation no

Repository CRAN

Date/Publication 2024-03-25 05:50:02 UTC

Contents

`Class_MeanVar_portfolio` *S3 class MeanVar_portfolio*

Description

Class MeanVar_portfolio is designed to construct mean-variance portfolios with provided estimators of the mean vector, covariance matrix, and inverse covariance matrix. It includes the following elements:

Slots

Element	Description
call	the function call with which it was created
cov_mtrx	the sample covariance matrix of the asset returns
inv_cov_mtrx	the inverse of the sample covariance matrix
means	sample mean vector of the asset returns

weights	portfolio weights
Port_Var	portfolio variance
Port_mean_return	expected portfolio return
Sharpe	portfolio Sharpe ratio

See Also

summary.MeanVar_portfolio summary method for the class, new_MeanVar_portfolio class constructor, validate_MeanVar_portfolio class validator, MeanVar_portfolio class helper.

`CovarEstim` *Covariance matrix estimator*

Description

It is a function dispatcher for covariance matrix estimation. One can choose between traditional and shrinkage-based estimators.

Usage

```
CovarEstim(x, type = c("trad", "BGP14", "LW20"), ...)
```

Arguments

`x`	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
`type`	a character. The estimation method to be used.
`...`	arguments to pass to estimators

Details

The available estimation methods are:

Function	Paper	Type
`Sigma_sample_estimator`		traditional
`CovShrinkBGP14`	Bodnar et al 2014	BGP14
`nonlin_shrinkLW`	Ledoit & Wolf 2020	LW20

Value

an object of class matrix

Examples

```
n<-3e2 # number of realizations
p<-.5*n # number of assets

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

Mtrx_trad <- CovarEstim(x, type="trad")

TM <- matrix(0, p, p)
diag(TM) <- 1
Mtrx_bgp <- CovarEstim(x, type="BGP14", TM=TM)

Mtrx_lw <- CovarEstim(x, type="LW20")
```

CovShrinkBGP14 *Linear shrinkage estimator of the covariance matrix (Bodnar et al. 2014)*

Description

The optimal linear shrinkage estimator of the covariance matrix that minimizes the Frobenius norm:

$$\hat{\Sigma}_{OLSE} = \hat{\alpha} S + \hat{\beta} \Sigma_0,$$

where $\hat{\alpha}$ and $\hat{\beta}$ are optimal shrinkage intensities given in Eq. (4.3) and (4.4) of Bodnar et al. (2014). S is the sample covariance matrix (SCM, see `Sigma_sample_estimator`) and Σ_0 is a positive definite symmetric matrix used as the target matrix (TM), for example, $\frac{1}{p} I$.

Usage

```
CovShrinkBGP14(n, TM, SCM)
```

Arguments

`n`	sample size.
`TM`	the target matrix for the shrinkage estimator.
`SCM`	sample covariance matrix.

Value

a list containing an object of class matrix (S) and the estimated shrinkage intensities $\hat{\alpha}$ and $\hat{\beta}$.

References

Bodnar T, Gupta AK, Parolya N (2014). “On the strong convergence of the optimal linear shrinkage estimator for large dimensional covariance matrix.” *Journal of Multivariate Analysis*, **132**, 215–228.

Examples

```
# Parameter setting
n<-3e2
c<-0.7
p<-c*n
mu <- rep(0, p)
Sigma <- RandCovMtrx(p=p)

# Generating observations
X <- t(MASS::mvrnorm(n=n, mu=mu, Sigma=Sigma))

# Estimation
TM <- matrix(0, nrow=p, ncol=p)
diag(TM) <- 1/p
SCM <- Sigma_sample_estimator(X)
Sigma_shr <- CovShrinkBGP14(n=n, TM=TM, SCM=SCM)
Sigma_shr$S[1:6, 1:6]
```

`InvCovShrinkBGP16` *Linear shrinkage estimator of the inverse covariance matrix (Bodnar et al. 2016)*

Description

The optimal linear shrinkage estimator of the inverse covariance (precision) matrix that minimizes the Frobenius norm is given by:

$$\hat{\Pi}_{OLSE} = \hat{\alpha}\hat{\Pi} + \hat{\beta}\Pi_0,$$

where $\hat{\alpha}$ and $\hat{\beta}$ are optimal shrinkage intensities given in Eq. (4.4) and (4.5) of Bodnar et al. (2016). $\hat{\Pi}$ is the inverse of the sample covariance matrix (iSCM) and Π_0 is a positive definite symmetric matrix used as the target matrix (TM), for example, I.

Usage

```
InvCovShrinkBGP16(n, p, TM, iSCM)
```

Arguments

`n`	the number of observations
`p`	the number of variables (rows of the covariance matrix)
`TM`	the target matrix for the shrinkage estimator
`iSCM`	the inverse of the sample covariance matrix

Value

a list containing an object of class matrix (S) and the estimated shrinkage intensities $\hat{\alpha}$ and $\hat{\beta}$.

References

Bodnar T, Gupta AK, Parolya N (2016). "Direct shrinkage estimation of large dimensional precision matrix." *Journal of Multivariate Analysis*, **146**, 223–236.

Examples

```
# Parameter setting
n <- 3e2
c <- 0.7
p <- c*n
mu <- rep(0, p)
Sigma <- RandCovMtrx(p=p)

# Generating observations
X <- t(MASS::mvrnorm(n=n, mu=mu, Sigma=Sigma))

# Estimation
TM <- matrix(0, nrow=p, ncol=p)
diag(TM) <- 1
iSCM <- solve(Sigma_sample_estimator(X))
Sigma_shr <- InvCovShrinkBGP16(n=n, p=p, TM=TM, iSCM=iSCM)
Sigma_shr$S[1:6, 1:6]
```

MeanEstim — *Mean vector estimator*

Description

A user-friendly function for estimation of the mean vector. Essentially, it is a function dispatcher for estimation of the mean vector that chooses a method accordingly to the type argument.

Usage

```
MeanEstim(x, type, ...)
```

Arguments

`x`	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
`type`	a character. The estimation method to be used.
`...`	arguments to pass to estimators

Details

The available estimation methods for the mean are:

Function	Paper	Type
.rowMeans		trad

mean_bs	Jorion 1986	bs
mean_js	Jorion 1986	js
mean_bop19	Bodnar et al 2019	BOP19

Value

a numeric vector— a value of the specified estimator of the mean vector.

References

Jorion P (1986). "Bayes-Stein estimation for portfolio analysis." *Journal of Financial and Quantitative Analysis*, 279–292.

Bodnar T, Okhrin O, Parolya N (2019). "Optimal shrinkage estimator for high-dimensional mean vector." *Journal of Multivariate Analysis*, **170**, 63–79.

Examples

```
n<-3e2 # number of realizations
p<-.5*n # number of assets

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

Mean_trad <- MeanEstim(x, type="trad")

mu_0 <- rep(1/p, p)
Mean_BOP <- MeanEstim(x, type="BOP19", mu_0=mu_0)
```

MeanVar_portfolio *A helper function for MeanVar_portfolio*

Description

A user-friendly function making mean-variance portfolios for assets with customly computed covariance matrix and mean returns. The weights are computed in accordance with the formula

$$\hat{w}_{MV} = \frac{\hat{\Sigma}^{-1}1}{1'\hat{\Sigma}^{-1}1} + \gamma^{-1}\hat{Q}\hat{\mu},$$

where $\hat{\Sigma}$ is an estimator for the covariance matrix, $\hat{\mu}$ is an estimator for the mean vector, γ is the coefficient of risk aversion, and $\hat{Q}$ is given by

$$\hat{Q} = \hat{\Sigma}^{-1} - \frac{\hat{\Sigma}^{-1}11'\hat{\Sigma}^{-1}}{1'\hat{\Sigma}^{-1}1}.$$

The computation is made by new_MeanVar_portfolio and the result is validated by validate_MeanVar_portfolio.

Usage

```
MeanVar_portfolio(mean_vec, cov_mtrx, gamma)
```

Arguments

mean_vec	mean vector of asset returns provided in the form of a vector or a list.
cov_mtrx	the covariance matrix of asset returns. It could be a matrix or a data frame.
gamma	a numeric variable. Coefficient of risk aversion.

Value

Mean-variance portfolio in the form of object of S3 class MeanVar_portfolio.

Examples

```
n<-3e2 # number of realizations
p<-.5*n # number of assets
gamma<-1

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

# Simple MV portfolio
cov_mtrx <- Sigma_sample_estimator(x)
means <- rowMeans(x)

cust_port_simp <- MeanVar_portfolio(mean_vec=means,
                                    cov_mtrx=cov_mtrx, gamma=2)
str(cust_port_simp)
```

---

mean_bop19 *BOP shrinkage estimator*

Description

Shrinkage estimator of the high-dimensional mean vector as suggested in Bodnar et al. (2019). It uses the formula

$$\hat{\mu}_{BOP} = \hat{\alpha}\bar{x} + \hat{\beta}\mu_0,$$

where $\hat{\alpha}$ and $\hat{\beta}$ are shrinkage coefficients given by Eq.(6) and Eg.(7) of Bodnar et al. (2019) that minimize weighted quadratic loss for a given target vector μ_0 (shrinkage target). $\bar{x}$ stands for the sample mean vector.

Usage

```
mean_bop19(x, mu_0 = rep(1, p))
```

Arguments

x — a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.

mu_0 — a numeric vector. The target vector used in the construction of the shrinkage estimator.

Value

a numeric vector containing the shrinkage estimator of the mean vector

References

Bodnar T, Okhrin O, Parolya N (2019). "Optimal shrinkage estimator for high-dimensional mean vector." *Journal of Multivariate Analysis*, **170**, 63–79.

Examples

```
n<-7e2 # number of realizations
p<-.5*n # number of assets
x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)
mm <- mean_bop19(x=x, mu_0=rep(1,p))
```

---

mean_bs — *Bayes-Stein shrinkage estimator of the mean vector*

Description

Bayes-Stein shrinkage estimator of the mean vector as suggested in Jorion (1986). The estimator is given by

$$\hat{\mu}_{BS} = (1-\beta)\bar{x} + \beta Y_0 1,$$

where $\bar{x}$ is the sample mean vector, β and Y_0 are derived using Bayesian approach (see Eq.(14) and Eq.(17) in Jorion (1986)).

Usage

```
mean_bs(x)
```

Arguments

x — a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.

Value

a numeric vector containing the Bayes-Stein shrinkage estimator of the mean vector

References

Jorion P (1986). "Bayes-Stein estimation for portfolio analysis." *Journal of Financial and Quantitative Analysis*, 279–292.

Examples

```
n <- 7e2 # number of realizations
p <- .5*n # number of assets
x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)
mm <- mean_bs(x=x)
```

---

mean_js — *James-Stein shrinkage estimator of the mean vector*

Description

James-Stein shrinkage estimator of the mean vector as suggested in Jorion (1986). The estimator is given by

$$\hat{\mu}_{JS} = (1-\beta)\bar{x} + \beta Y_0 1,$$

where $\bar{x}$ is the sample mean vector, β is the shrinkage coefficient which minimizes a quadratic loss given by Eq.(11) in Jorion (1986). Y_0 is a prespecified value.

Usage

```
mean_js(x, Y_0 = 1)
```

Arguments

`x`	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
`Y_0`	a numeric variable. Shrinkage target coefficient.

Value

a numeric vector containing the James-Stein shrinkage estimator of the mean vector.

References

Jorion P (1986). "Bayes-Stein estimation for portfolio analysis." *Journal of Financial and Quantitative Analysis*, 279–292.

Examples

```
n<-7e2 # number of realizations
p<-.5*n # number of assets
x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)
mm <- mean_js(x=x, Y_0 = 1)
```

MVShrinkPortfolio *Shrinkage mean-variance portfolio*

Description

The main function for mean-variance (also known as expected utility) portfolio construction. It is a dispatcher using methods according to argument type, values of gamma and dimensionality of matrix x.

Usage

```
MVShrinkPortfolio(x, gamma, type = c("shrinkage", "traditional"), ...)
```

Arguments

x	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
gamma	a numeric variable. Coefficient of risk aversion.
type	a character. The type of methods to use to construct the portfolio.
...	arguments to pass to portfolio constructors

Details

The sample estimator of the mean-variance portfolio weights, which results in a traditional mean-variance portfolio, is calculated by

$$\hat{w}_{MV} = \frac{S^{-1}1}{1'S^{-1}1} + \gamma^{-1}\hat{Q}\bar{x},$$

where S^{-1} and $\bar{x}$ are the inverse of the sample covariance matrix and the sample mean vector of asset returns respectively, γ is the coefficient of risk aversion and $\hat{Q}$ is given by

$$\hat{Q} = S^{-1} - \frac{S^{-1}11'S^{-1}}{1'S^{-1}1}.$$

In the case when $p > n$, S^{-1} becomes S^{+}- Moore-Penrose inverse. The shrinkage estimator for the mean-variance portfolio weights in a high-dimensional setting is given by

$$\hat{w}_{ShMV} = \hat{\alpha}\hat{w}_{MV} + (1 - \hat{\alpha})b,$$

where $\hat{\alpha}$ is the estimated shrinkage intensity and b is a target vector with the sum of the elements equal to one.

In the case $\gamma \neq \infty$, $\hat{\alpha}$ is computed following Eq. (2.22) of Bodnar et al. (2023) for c<1 and following Eq. (2.29) of Bodnar et al. (2023) for c>1.

The case of a fully risk averse investor ($\gamma = \infty$) leads to the traditional global minimum variance (GMV) portfolio with the weights given by

$$\hat{w}_{GMV} = \frac{S^{-1}1}{1'S^{-1}1}.$$

The shrinkage estimator for the GMV portfolio is then calculated by

$$\hat{w}_{ShGMV} = \hat{\alpha}\hat{w}_{GMV} + (1-\hat{\alpha})b,$$

with $\hat{\alpha}$ given in Eq. (2.31) of Bodnar et al. (2018) for c<1 and in Eq. (2.33) of Bodnar et al. (2018) for c>1.

These estimation methods are available as separate functions employed by MVShrinkPortfolio accordingly to the following parameter configurations:

Function	Paper	Type	gamma	p/n
new_MV_portfolio_weights_BDOPS21	Bodnar et al. (2023)	shrinkage	< Inf	<1
new_MV_portfolio_weights_BDOPS21_pgn	Bodnar et al. (2023)	shrinkage	< Inf	>1
new_GMV_portfolio_weights_BDPS19	Bodnar et al. (2018)	shrinkage	Inf	<1
new_GMV_portfolio_weights_BDPS19_pgn	Bodnar et al. (2018)	shrinkage	Inf	>1
new_MV_portfolio_traditional		traditional	> 0	<1
new_MV_portfolio_traditional_pgn		traditional	> 0	>1

Value

A portfolio in the form of an object of class MeanVar_portfolio potentially with a subclass. See Class_MeanVar_portfolio for the details of the class.

References

Bodnar T, Okhrin Y, Parolya N (2023). “Optimal shrinkage-based portfolio selection in high dimensions.” *Journal of Business & Economic Statistics*, **41**, 140-156.

Bodnar T, Parolya N, Schmid W (2018). “Estimation of the global minimum variance portfolio in high dimensions.” *European Journal of Operational Research*, **266**(1), 371–390.

Examples

```
n<-3e2 # number of realizations
gamma<-1

# The case p<n

p<-.5*n # number of assets
b<-rep(1/p,p)

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

test <- MVShrinkPortfolio(x=x, gamma=gamma,
                          type='shrinkage', b=b, beta = 0.05)
str(test)

test <- MVShrinkPortfolio(x=x, gamma=Inf,
                          type='shrinkage', b=b, beta = 0.05)
str(test)
```

```
test <- MVShrinkPortfolio(x=x, gamma=gamma, type='traditional')
str(test)

# The case p>n

p<-1.2*n # Re-define the number of assets
b<-rep(1/p,p)

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

test <- MVShrinkPortfolio(x=x, gamma=gamma, type='shrinkage',
                          b=b, beta = 0.05)
str(test)

test <- MVShrinkPortfolio(x=x, gamma=Inf, type='shrinkage',
                          b=b, beta = 0.05)
str(test)
```

---

new_GMV_portfolio_weights_BDPS19 *Constructor of GMV portfolio object.*

Description

Constructor of global minimum variance portfolio. new_GMV_portfolio_weights_BDPS19 is for the case p<n, while new_GMV_portfolio_weights_BDPS19_pgn is for p>n, where p is the number of assets and n is the number of observations. For more details of the method, see MVShrinkPortfolio.

Usage

```
new_GMV_portfolio_weights_BDPS19(x, b, beta)

new_GMV_portfolio_weights_BDPS19_pgn(x, b, beta)
```

Arguments

x	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
b	a numeric vector. 1-beta is the confidence level of the symmetric confidence interval, constructed for each weight.
beta	a numeric variable. The confidence level for weight intervals.

Value

an object of class MeanVar_portfolio with subclass GMV_portfolio_weights_BDPS19.

Element	Description
call	the function call with which it was created
cov_mtrx	the sample covariance matrix of the asset returns
inv_cov_mtrx	the inverse of the sample covariance matrix
means	sample mean vector estimate of the asset returns
w_GMVP	sample estimator of portfolio weights
weights	shrinkage estimator of portfolio weights
alpha	shrinkage intensity for the weights
Port_Var	portfolio variance
Port_mean_return	expected portfolio return
Sharpe	portfolio Sharpe ratio
weight_intervals	A data frame, see details

weight_intervals contains a shrinkage estimator of portfolio weights, asymptotic confidence intervals for the true portfolio weights, the value of test statistic and the p-value of the test on the equality of the weight of each individual asset to zero (see Section 4.3 of Bodnar et al. 2023). weight_intervals is only computed when p<n.

References

Bodnar T, Dmytriv S, Parolya N, Schmid W (2019). "Tests for the weights of the global minimum variance portfolio in a high-dimensional setting." *IEEE Transactions on Signal Processing*, **67**(17), 4479–4493.

Bodnar T, Parolya N, Schmid W (2018). "Estimation of the global minimum variance portfolio in high dimensions." *European Journal of Operational Research*, **266**(1), 371–390.

Bodnar T, Dette H, Parolya N, Thorsén E (2023). "Corrigendum to "Sampling Distributions of Optimal Portfolio Weights and Characteristics in Low and Large Dimensions."." *Random Matrices: Theory and Applications*, **12**, 2392001. doi:10.1142/S2010326323920016.

Examples

```
# c<1

n <- 3e2 # number of realizations
p <- .5*n # number of assets
b <- rep(1/p,p)

# Assets with a diagonal covariance matrix
x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

test <- new_GMV_portfolio_weights_BDPS19(x=x, b=b, beta=0.05)
str(test)

# Assets with a non-diagonal covariance matrix
Mtrx <- RandCovMtrx(p=p)
x <- t(MASS::mvrnorm(n=n , mu=rep(0,p), Sigma=Mtrx))

test <- new_GMV_portfolio_weights_BDPS19(x=x, b=b, beta=0.05)
```

```
summary(test)

# c>1

p <- 1.3*n # number of assets
b <- rep(1/p,p)

# Assets with a diagonal covariance matrix
x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

test <- new_GMV_portfolio_weights_BDPS19_pgn(x=x, b=b, beta=0.05)
str(test)
```

---

new_MeanVar_portfolio *A constructor for class MeanVar_portfolio*

Description

A light-weight constructor of objects of S3 class MeanVar_portfolio. This function is for development purposes. A helper function equipped with error messages and allowing more flexible input is `MeanVar_portfolio`.

Usage

```
new_MeanVar_portfolio(mean_vec, cov_mtrx, gamma)
```

Arguments

`mean_vec`	mean vector of asset returns
`cov_mtrx`	the covariance matrix of asset returns
`gamma`	a numeric variable. Coefficient of risk aversion.

Value

Mean-variance portfolio in the form of object of S3 class MeanVar_portfolio.

Examples

```
n<-3e2 # number of realizations
p<-.5*n # number of assets
gamma<-1

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

# Simple MV portfolio
cov_mtrx <- Sigma_sample_estimator(x)
means <- rowMeans(x)
```

```
cust_port_simp <- new_MeanVar_portfolio(mean_vec=means,
                                        cov_mtrx=cov_mtrx, gamma=2)
str(cust_port_simp)

# Portfolio with Bayes-Stein shrunk means
# and a Ledoit and Wolf estimator for covariance matrix
TM <- matrix(0, p, p)
diag(TM) <- 1
cov_mtrx <- CovarEstim(x, type="LW20", TM=TM)
means <- mean_bs(x)

cust_port_BS_LW <- new_MeanVar_portfolio(mean_vec=means$means,
                                         cov_mtrx=cov_mtrx, gamma=2)
str(cust_port_BS_LW)
```

---

new_MV_portfolio_traditional
Traditional mean-variance portfolio

Description

Mean-variance portfolios with the traditional (sample) estimators for the mean vector and the covariance matrix of asset returns. For more details of the method, see MVShrinkPortfolio. new_MV_portfolio_traditional is for the case p<n, while new_MV_portfolio_traditional_pgn is for p>n, where p is the number of assets and n is the number of observations.

Usage

```
new_MV_portfolio_traditional(x, gamma)

new_MV_portfolio_traditional_pgn(x, gamma)
```

Arguments

x	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
gamma	a numeric variable. Coefficient of risk aversion.

Value

an object of class MeanVar_portfolio

Element	Description
call	the function call with which it was created
cov_mtrx	the sample covariance matrix of asset returns
inv_cov_mtrx	the inverse of the sample covariance matrix
means	sample mean estimator of the asset returns
W_mv_hat	sample estimator of portfolio weights

Port_Var	portfolio variance
Port_mean_return	expected portfolio return
Sharpe	portfolio Sharpe ratio

Examples

```
n <- 3e2 # number of realizations
p <- .5*n # number of assets
gamma <- 1

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

test <- new_MV_portfolio_traditional(x=x, gamma=gamma)
str(test)
```

`new_MV_portfolio_weights_BDOPS21`
Constructor of MV portfolio object

Description

Constructor of mean-variance shrinkage portfolios. new_MV_portfolio_weights_BDOPS21 is for the case p<n, while new_MV_portfolio_weights_BDOPS21_pgn is for p>n, where p is the number of assets and n is the number of observations. For more details of the method, see `MVShrinkPortfolio`.

Usage

```
new_MV_portfolio_weights_BDOPS21(x, gamma, b, beta)

new_MV_portfolio_weights_BDOPS21_pgn(x, gamma, b, beta)
```

Arguments

`x`	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
`gamma`	a numeric variable. Coefficient of risk aversion.
`b`	a numeric variable. 1-beta is the confidence level of the symmetric confidence interval, constructed for each weight.
`beta`	a numeric variable. The confidence level for weight intervals.

Value

an object of class MeanVar_portfolio with subclass MV_portfolio_weights_BDOPS21.

Element	Description
call	the function call with which it was created

cov_mtrx	the sample covariance matrix of the asset returns
inv_cov_mtrx	the inverse of the sample covariance matrix
means	sample mean vector of the asset returns
W_mv_hat	sample estimator of the portfolio weights
weights	shrinkage estimator of the portfolio weights
alpha	shrinkage intensity for the weights
Port_Var	portfolio variance
Port_mean_return	expected portfolio return
Sharpe	portfolio Sharpe ratio
weight_intervals	A data frame, see details

weight_intervals contains a shrinkage estimator of portfolio weights, asymptotic confidence intervals for the true portfolio weights, value of the test statistic and the p-value of the test on the equality of the weight of each individual asset to zero (see Section 4.3 of Bodnar et al. 2023) weight_intervals is only computed when p<n.

References

Bodnar T, Dmytriv S, Okhrin Y, Parolya N, Schmid W (2021). "Statistical Inference for the Expected Utility Portfolio in High Dimensions." *IEEE Transactions on Signal Processing*, **69**, 1-14.

Bodnar T, Dette H, Parolya N, Thorsén E (2023). "Corrigendum to "Sampling Distributions of Optimal Portfolio Weights and Characteristics in Low and Large Dimensions."." *Random Matrices: Theory and Applications*, **12**, 2392001. doi:10.1142/S2010326323920016.

Examples

```
# c<1

# Assets with a diagonal covariance matrix

n <- 3e2 # number of realizations
p <- .5*n # number of assets
b <- rep(1/p,p)
gamma <- 1

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

test <- new_MV_portfolio_weights_BDOPS21(x=x, gamma=gamma, b=b, beta=0.05)
summary(test)

# Assets with a non-diagonal covariance matrix

Mtrx <- RandCovMtrx(p=p)
x <- t(MASS::mvrnorm(n=n , mu=rep(0,p), Sigma=Mtrx))

test <- new_MV_portfolio_weights_BDOPS21(x=x, gamma=gamma, b=b, beta=0.05)
str(test)
```

```
# c>1

n <-2e2 # number of realizations
p <-1.2*n # number of assets
b <-rep(1/p,p)
x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

test <- new_MV_portfolio_weights_BDOPS21_pgn(x=x, gamma=gamma,
                                             b=b, beta=0.05)
summary(test)

# Assets with a non-diagonal covariance matrix
```

`nonlin_shrinkLW` *nonlinear shrinkage estimator of the covariance matrix of Ledoit and Wolf (2020)*

Description

The nonlinear shrinkage estimator of the covariance matrix, that minimizes the minimum variance loss functions as defined in Eq (2.1) of Ledoit and Wolf (2020).

Usage

```
nonlin_shrinkLW(x)
```

Arguments

x	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.

Value

an object of class matrix

References

Ledoit O, Wolf M (2020). “Analytical nonlinear shrinkage of large-dimensional covariance matrices.” *Annals of Statistics*, **48**(5), 3043–3065.

Examples

```
n<-3e2
c<-0.7
p<-c*n
mu <- rep(0, p)
Sigma <- RandCovMtrx(p=p)

X <- t(MASS::mvrnorm(n=n, mu=mu, Sigma=Sigma))
Sigma_shr <- nonlin_shrinkLW(X)
```

`plot_frontier` *Plot the Bayesian efficient frontier (Bauder et al. 2021) and the provided portfolios.*

Description

The plotted Bayesian efficient frontier is provided by Eq. (8) in Bauder et al. (2021). It is the set of optimal portfolios obtained by employing the posterior predictive distribution on the asset returns. This efficient frontier can be used to assess the mean-variance efficiency of various estimators of the portfolio weights. The standard deviation of the portfolio return is plotted in the x-axis and the mean portfolio return in the y-axis. The portfolios with the weights w are added to the plot by computing $\sqrt{\mathrm{w}'\mathrm{S}\mathrm{w}}$ and $\mathrm{w}'\bar{\mathrm{x}}$.

Usage

```
plot_frontier(x, weights.eff = rep(1/nrow(x), length = nrow(x)))
```

Arguments

`x`	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
`weights.eff`	matrix of portfolio weights. Each column contains p values of the weights for a given portfolio. Default: equally weighted portfolio.

Value

a ggplot object

References

Bauder D, Bodnar T, Parolya N, Schmid W (2021). “Bayesian mean–variance analysis: optimal portfolio selection under parameter uncertainty.” *Quantitative Finance*, **21**(2), 221–242.

Examples

```
p <- 150
n <- 300
gamma <- 10
mu <- seq(0.2,-0.2, length.out=p)
Sigma <- RandCovMtrx(p=p)

x <- t(MASS::mvrnorm(n=n , mu=mu, Sigma=Sigma))

EW_port <- rep(1/p, length=p)
MV_shr_port <- new_MV_portfolio_weights_BDOPS21(x=x, gamma=gamma,
                                                b=EW_port, beta=0.05)$weights
GMV_shr_port <- new_MV_portfolio_weights_BDOPS21(x=x, gamma=Inf, b=EW_port,
                                                 beta=0.05)$weights
```

```
MV_trad_port <- new_MV_portfolio_traditional(x=x, gamma=gamma)$weights
GMV_trad_port <- new_MV_portfolio_traditional(x=x, gamma=Inf)$weights

weights.eff <- cbind(EW_port, MV_shr_port, GMV_shr_port,
                     MV_trad_port, GMV_trad_port)
colnames(weights.eff) <- c("EW", "MV_shr", "GMV_shr", "MV_trad", "GMV_trad")


Fplot <- plot_frontier(x, weights.eff)
Fplot
```

RandCovMtrx *Covariance matrix generator*

Description

Generates a covariance matrix from Wishart distribution with given eigenvalues or with exponentially decreasing eigenvalues. Useful for examples and tests when an arbitrary covariance matrix is needed.

Usage

```
RandCovMtrx(p = 200, eigenvalues = 0.1 * exp(5 * seq_len(p)/p))
```

Arguments

p	dimension of the covariance matrix
eigenvalues	the vector of positive eigenvalues

Details

This function generates a symmetric positive definite covariance matrix with given eigenvalues. The eigenvalues can be specified explicitly. Or, by default, they are generated with exponential decay.

Value

covariance matrix

Examples

```
p<-1e1
# A non-diagonal covariance matrix
Mtrx <- RandCovMtrx(p=p)
Mtrx
```

`Sigma_sample_estimator` *Sample covariance matrix*

Description

It computes the sample covariance of matrix S as follows:

$$S = \frac{1}{n-1} \sum_{j=1}^{n} (x_j - \bar{x})(x_j - \bar{x})', \quad \bar{x} = \frac{1}{n} \sum_{j=1}^{n} x_j,$$

where x_j is the j-th column of the data matrix x.

Usage

```
Sigma_sample_estimator(x)
```

Arguments

`x`	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.

Value

Sample covariance estimation

Examples

```
p<-5 # number of assets
n<-1e1 # number of realizations

x <-matrix(data = rnorm(n*p), nrow = p, ncol = n)
Sigma_sample_estimator(x)
```

`SP_daily_asset_returns` *Daily log-returns of selected constituents S&P500.*

Description

Daily log-returns of selected constituents of S&P500 in percents. The data are sampled in business time, i.e., weekends and holidays are omitted.

Usage

```
SP_daily_asset_returns
```

Format

a matrix with the first column containing the data and company names as column labels.

Source

Yahoo finance

`test_MVSP` *Test for mean-variance portfolio weights*

Description

A high-dimensional asymptotic test on the mean-variance efficiency of a given portfolio with the weights w_0. The tested hypotheses are

$$H_0 : w_{MV} = w_0 \quad vs \quad H_1 : w_{MV} \neq w_0.$$

The test statistic is based on the shrinkage estimator of mean-variance portfolio weights (see Eq.(44) of Bodnar et al. 2021).

Usage

```
test_MVSP(gamma, x, w_0, beta = 0.05)
```

Arguments

`gamma`	a numeric variable. Coefficient of risk aversion.
`x`	a p by n matrix or a data frame of asset returns. Rows represent different assets, columns – observations.
`w_0`	a numeric vector of tested weights.
`beta`	a significance level for the test.

Details

Note: when gamma == Inf, we get the test for the weights of the global minimum variance portfolio as in Theorem 2 of Bodnar et al. (2019).

Value

Element	Description
alpha_hat	the estimated shrinkage intensity
alpha_sd	the standard deviation of the shrinkage intensity
alpha_lower	the lower bound for the shrinkage intensity
alpha_upper	the upper bound for the shrinkage intensity
T_alpha	the value of the test statistic
p_value	the p-value for the test

References

Bodnar T, Dmytriv S, Okhrin Y, Parolya N, Schmid W (2021). "Statistical Inference for the Expected Utility Portfolio in High Dimensions." *IEEE Transactions on Signal Processing*, **69**, 1-14.

Bodnar T, Dmytriv S, Parolya N, Schmid W (2019). "Tests for the weights of the global minimum variance portfolio in a high-dimensional setting." *IEEE Transactions on Signal Processing*, **67**(17), 4479–4493.

Examples

```
n<-3e2 # number of realizations
p<-.5*n # number of assets
b<-rep(1/p,p)
gamma<-1

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

T_alpha <- test_MVSP(gamma=gamma, x=x, w_0=b, beta=0.05)
T_alpha
```

---

validate_MeanVar_portfolio

A validator for objects of class MeanVar_portfolio

Description

A validator for objects of class MeanVar_portfolio

Usage

```
validate_MeanVar_portfolio(w)
```

Arguments

w	Object of class MeanVar_portfolio.

Value

If the object passes all the checks, then w itself is returned, otherwise an error is thrown.

Examples

```
n<-3e2 # number of realizations
p<-.5*n # number of assets
gamma<-1

x <- matrix(data = rnorm(n*p), nrow = p, ncol = n)

# Simple MV portfolio
cov_mtrx <- Sigma_sample_estimator(x)
means <- rowMeans(x)

cust_port_simp <- new_MeanVar_portfolio(mean_vec=means,
                                        cov_mtrx=cov_mtrx, gamma=2)
str(validate_MeanVar_portfolio(cust_port_simp))
```

Index