Package ‘fAssets’

April 24, 2023

Title Rmetrics - Analysing and Modelling Financial Assets
Date 2023-04-20
Version 4023.85
Description A collection of functions to manage, to investigate and to analyze data sets of financial
assets from different points of view.
Depends R (>= 2.15.1), timeDate, timeSeries, fBasics
Imports fMultivar, robustbase, MASS, sn, ecodist, mvnormtest, energy,
grDevices, graphics, stats
Suggests methods, mnormt, RUnit
License GPL (>= 2)
URL https://r-forge.r-project.org/projects/rmetrics/
NeedsCompilation no
Author Diethelm Wuertz [aut],
    Tobias Setz [aut],
    Yohan Chalabi [aut],
    Stefan Theussl [aut, cre]<https://orcid.org/0000-0002-6523-4620>
Maintainer Stefan Theussl <Stefan.Theussl@R-project.org>
Repository CRAN
Date/Publication 2023-04-24 14:00:02 UTC

R topics documented:

fAssets-package ........................................... 2
assets-arrange ............................................. 6
assets-distance .......................................... 7
assets-lpm .................................................. 9
assets-meancov .......................................... 10
assets-modeling ........................................... 12
assets-outliers .......................................... 14
assets-selection ......................................... 15
assets-testing ............................................ 16
fAssets-package

Description

The Rmetrics fAssets package is a collection of functions to manage, to investigate and to analyze data sets of financial assets from different points of view.

Details

Package: fAssets
Type: Package
Date: 2014
License: GPL Version 2 or later
Copyright: (c) 1999-2014 Rmetrics Association
Repository: R-FORGE
URL: https://www.rmetrics.org

1 Introduction

The package fAssets was written to explore and investigate data sets of financial assets.

Included are functions to make the asset selection process easier, to robustify return and covariances for modeling portfolios, to test financial returns for multivariate normality, and to measure in a simple way performance and risk of funds and portfolios.

Beside this many functions for graphs and plots, and for a more sophisticated explorative data analysis are provided. They range from simple time series plots to more elaborated statistical chart tools: histogram, density, boxplots, and QQ plots; pairs, similarities, and covariance ellipses plots; star plots, and risk/reward graphs.
2 Assets Selection

The assets selection chapter contains functions which arrange assets from a data set according to different measures applying ideas from principal component analysis, from hierarchical clustering, or by a user defined statistical measure:

- `assetsArrange` Rearranges the columns in a data set of assets
- `pcaArrange` Returns PCA correlation ordered column names
- `hclustArrange` Returns hierarchical clustered column names
- `abcArrange` Returns assets sorted by column names
- `orderArrange` Returns assets ordered by column names
- `sampleArrange` Returns a re-sampled set of assets
- `statsArrange` Returns statistically rearranged column names

In addition we have summarized and bundled a set of distance measure functions to determine the similarity or dissimilarity of individual assets from a set of multivariate financial return series.

- `assetsDist` Computes the distances between assets
- `corDist` Returns correlation distance measure
- `kendallDist` Returns Kendall’s correlation distance measure
- `spearmanDist` Returns Spearman’s correlation distance measure
- `mutinfoDist` Returns mutual information distance measure
- `euclideanDist` Returns Euclidean distance measure
- `maximumDist` Returns maximum distance measure
- `manhattanDist` Returns Manhattan distance measure
- `canberraDist` Returns Canberra distance measure
- `binaryDist` Returns binary distance measure
- `minkowskiDist` Returns Minkowski distance measure
- `braycurtisDist` Returns Bray Curtis distance measure
- `mahalanobisDist` Returns Mahalanobis distance measure
- `jaccardDist` Returns Jaccard distance measure
- `sorensenDist` Returns Sorensen distance measure

A last group of functions allows to select assets by concepts from hierarchical or k-means clustering:

- `assetsSelect` Selects similar or dissimilar assets
- `.hclustSelect` Selects due to hierarchical clustering
- `.kmeansSelect` Selects due to k-means clustering

3 Assets Covariance Robustification

We provide several functions to compute robust measures for mean and/or covariance estimates which can be used for example in robustified Markowitz portfolio optimization.

- `assetsMeanCov` Estimates mean and variance for a set of assets
- `.covMeanCov` uses sample covariance estimation
An additional function allows to detect outliers from a PCA outlier analysis.

assetsOutliers Detects outliers in multivariate assets sets

4 Testing Assets for Normality

The multivariate Shapiro test and the E-Statistic Energy Test allow to test multivariate Normality of financial returns.

assetsTest Tests for multivariate Normal Assets
mvshapiroTest Multivariate Shapiro Test
mvenergyTest Multivariate E-Statistic (Energy) Test

5 Lower Partial Moments Measures

The computation of Lower partial moments is done by the following two functions:

assetsLPM Computes asymmetric lower partial moments
assetsSLPM Computes symmetric lower partial moments

6 Assets Time Series and Density Plot Functions

Dozens of tailored plot functions are included in the fAssets package. This makes it very easy to visualize properties and to perform an explorative data analysis. Starting from simple time series functions.

assetsReturnPlot Displays time series of individual assets
assetsCumulatedPlot Displays time series of individual assets
assetsSeriesPlot Displays time series of individual assets
we can also explore the distributional properties of the returns by histogram, density, boxplots, and QQ Plots:

- assetsHistPlot: Displays a histograms of a single asset
- assetsLogDensityPlot: Displays a pdf plot on logarithmic scale
- assetsHistPairsPlot: Displays a bivariate histogram plot
- assetsBoxPlot: Displays a standard box plot
- assetsBoxPercentilePlot: Displays a side-by-side box-percentage plot
- assetsQQNormPlot: Displays normal qq-plots of individual assets

7 Assets Dependency and Structure Plot Functions

Corellation and similarities are another source of information about the dependence structure of individual financial returns. The functions which help us to detect those properties in data sets of financial assets include:

- assetsPairsPlot: Displays pairs of scatterplots of assets
- assetsCorgramPlot: Displays pairwise correlations between assets
- assetsCorTestPlot: Displays and tests pairwise correlations
- assetsCorImagePlot: Displays an image plot of a correlations
- covEllipsesPlot: Displays a covariance ellipses plot
- assetsDendrogramPlot: Displays hierarchical clustering dendrogram
- assetsCorEigenPlot: Displays ratio of the largest two eigenvalues

Beside correlations und dependencies also risk/reward graphs give additional insight into the structure of assets.

- assetsRiskReturnPlot: Displays risk-return diagram of assets
- assetsNIGShapeTrianglePlot: Displays NIG Shape Triangle
- assetsTreePlot: Displays a minimum spanning tree of assets

Statistic visualized by star plots is a very appealing tool for characterization and classification of assets by eye:

- assetsStarsPlot: Draws segment/star diagrams of asset sets
- assetsBasicStatsPlot: Displays a segment plot of basic return stats
- assetsMomentsPlot: Displays a segment plot of distribution moments
- assetsBoxStatsPlot: Displays a segment plot of box plot statistics
- assetsNIGFitPlot: Displays a segment plot NIG parameter estimates

About Rmetrics:

The fAssets Rmetrics package is written for educational support in teaching "Computational Finance and Financial Engineering” and licensed under the GPL.
Rearranging Assets Columnwise

Description

Allows to rearrange a set of assets columnwise.

Usage

```r
assetsArrange(x, method = c("pca", "hclust", "abc"), ...)
pcaArrange(x, robust = FALSE, ...)  
hclustArrange(x, method = c("euclidean", "complete"), ...)  
abcArrange(x, ...)  
orderArrange(x, ...)  
sampleArrange(x, ...)  
statsArrange(x, FUN = colMeans, ...)
```

Arguments

- `x`: any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.
- `method`: a character string, which method should be applied to rearrange the assets? Either "pca" which arranges the columns by an eigenvalue decomposition, "hclust" which arranges the columns by hierarchical clustering, "abc" which arranges the columns alphabetically, "order" which arranges the columns by the order function, "sample" which arranges the columns randomly, or "stats" which arranges by an statistical strategy.
- `robust`: a logical flag. Should robust statistics applied?
- `FUN`: function name of the statistical function to be applied.
- `...`: optional arguments to be passed.

Value

A character vector with the rearranged assets names.

Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); 
*Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.
Examples

```r
## LPP -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC[, 1:3]
head(LPP)

## assetsArrange -
# Arrange Assets Columns:
assetsArrange(x=LPP, "pca")
assetsArrange(x=LPP, "hclust")
assetsArrange(x=LPP, "abc")

## Alternative Usage -
pcaArrange(x=LPP, robust=FALSE)
pcaArrange(x=LPP, robust=TRUE)
hclustArrange(x=LPP, method = c("euclidean", "complete"))
abcArrange(x=LPP)
orderArrange(x=LPP)
sampleArrange(x=LPP)
statsArrange(x=LPP, FUN=colMeans)
```

<table>
<thead>
<tr>
<th>assets-distance</th>
<th>Distance Measures</th>
</tr>
</thead>
</table>

Description

Allows to measure the distance or similarity between assets.

Usage

```r
assetsDist(x, method="cor", ...)
corDist(x)
kendallDist(x)
spearmanDist(x)

mutinfoDist(x, nbin=10)
euclideanDist(x)
maximumDist(x)
manhattanDist(x)
canberraDist(x)
binaryDist(x)
minkowskiDist(x)

braycurtisDist(x)
mahalanobisDist(x)
```
jaccardDist(x)
sorensenDist(x)

Arguments

- **x**: any rectangular time series object which can be converted by the function `dist()` into a distance object.
- **method**: a character string, the method from which to compute the distances. Allowed methods include `cor`, `kendall`, `spearman`, `mutinfo`, `euclidean`, `maximum`, `manhattan`, `canberra`, `binary`, `minkowski`, `braycurtis`, `mahalanobis`, `jaccard`, `difference`, or `sorensen`.
- **nbin**: an integer value, the number of bins, by default 10.
- **...**: optional argument to be passed the distance function.

Details

corDist, kendallDist, and spearmanDist call the base `cov` function from R.
mutinfoDist calls the function `mutinfo` from the contributed R package `bioDist`.
euclideanDist, maximumDist, manhattanDist, canberraDist, binaryDist, and minkowskiDist are functions build on top of R’s base package.
braycurtisDist, mahalanobisDist, jaccardDist, and sorensenDist call functions from the contributed R package `ecodist`.

Value

an object of class `dist`.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); *Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

```r
## LPP -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC[, 1:6]
head(LPP)

## Returns correlation distance measure
corDist(LPP)

## Returns kendalls correlation distance measure
kendallDist(LPP)

## Returns spearmans correlation distance measure
spearmanDist(LPP)
```
## Return mutual information distance measure
mutinfoDist(LPP)

## Return Euclidean distance measure
euclideanDist(LPP)

## Return maximum distance measure
maximumDist(LPP)

## Return Manhattan distance measure
manhattanDist(LPP)

## Return Canberra distance measure
canberraDist(LPP)

## Return binary distance measure
binaryDist(LPP)

## Return Minkowsky distance measure
minkowskiDist(LPP)

## Return Bray Curtis distance measure
braycurtisDist(LPP)

## Return Mahalanobis distance measure
# mahalanobisDist(LPP)

## Return Jaccard distance measure
jaccardDist(LPP)

## Return Sorensen distance measure
sorensenDist(LPP)

---

**assets-lpm**

*Computation of Lower Partial Moments of Asset Sets*

### Description

Computes lower partial moments from a time series of assets.

### Usage

assetsLPM(x, tau, a, ...)
assetsSLPM(x, tau, a, ...)

### Arguments

- **x**: any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`. 
tau the target return.
a the value of the moment.
... optional arguments to be passed.

Value
returns a list with two entries named mu and Sigma. The first denotes the vector of lower partial moments, and the second the co-LPM matrix. Note, that the output of this function can be used as data input for the portfolio functions to compute the LPM efficient frontier.

Author(s)
Diethelm Wuertz for the Rmetrics port.

References
Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples
```r
## LPP -
# Percentual Returns:
LPP <- 100 * as.timeSeries(data(LPP2005REC))[, 1:6]
colnames(LPP)
```

assets-mean cov                                     Estimation of Mean and Covariances of Asset Sets

Description
Estimates the mean and/or covariance matrix of a time series of assets by traditional and robust methods.

Usage
```r
assetsMeanCov(x,  
method = c("cov", "mve", "mcd", "MCD", "OGK", "nnve", "shrink", "bagged"),  
check = TRUE, force = TRUE, baggedR = 100, sigmamu = scaleTau2,  
alpha = 1/2, ...)

getCenterRob(object)  
getCovRob(object)
```
Arguments

`x` any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.

`method` a character string, which determines how to compute the covariance matrix. If `method="cov"` is selected then the standard covariance will be computed by R’s base function `cov`, if `method="shrink"` is selected then the covariance will be computed using the shrinkage approach as suggested in Schaefer and Strimmer [2005], if `method="bagged"` is selected then the covariance will be calculated from the bootstrap aggregated (bagged) version of the covariance estimator.

`check` a logical flag. Should the covariance matrix be tested to be positive definite? By default `TRUE`.

`force` a logical flag. Should the covariance matrix be forced to be positive definite? By default `TRUE`.

`baggedR` when `method="bagged"`, an integer value, the number of bootstrap replicates, by default 100.

`sigmamu` when `method="OGK"`, a function that computes univariate robust location and scale estimates. By default it should return a single numeric value containing the robust scale (standard deviation) estimate. When `mu.too` is true (the default), `sigmamu()` should return a numeric vector of length 2 containing robust location and scale estimates. See `scaleTau2`, `s_Qn`, `s_Sn`, `s_mad` or `s_IQR` for examples to be used as sigmamu argument. For details we refer to the help pages of the R-package `robustbase`.

`object` a list as returned by the function `assetsMeanCov`.

`alpha` when `method="MCD"`, a numeric parameter controlling the size of the subsets over which the determinant is minimized, i.e., `alpha*n` observations are used for computing the determinant. Allowed values are between 0.5 and 1 and the default is 0.5. For details we refer to the help pages of the R-package `robustbase`.

`...` optional arguments to be passed to the underlying estimators. For details we refer to the manual pages of the functions `cov.rob` for arguments "mve" and "mcd" in the R package `MASS`, to the functions `covMcd` and `covOGK` in the R package `robustbase`.

Value

`assetsMeanCov` returns a list with for entries named `center` `cov`, `mu` and `Sigma`. The list may have a character vector attributed with additional control parameters.

`getCenterRob` extracts the center from an object as returned by the function `assetsMeanCov`.

`getCovRob` extracts the covariance from an object as returned by the function `assetsMeanCov`.

Author(s)

Juliane Schaefer and Korbinian Strimmer for R’s corpcov package,
Diethelm Wuertz for the Rmetrics port.
References


Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); *Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

```r
## LPP -
LPP <- as.timeSeries(data(LPP2005REC))[, 1:6]
colnames(LPP)

## Sample Covariance Estimation:
assetsMeanCov(LPP)

## Shrinked Estimation:
shrink <- assetsMeanCov(LPP, "shrink")
shrink

## Extract Covariance Matrix:
getCovRob(shrink)
```

assets-modeling  

Modeling Multivariate Asset Sets

Description

Fitting and Simulating assets from multivariate asset sets based on modeling skew normal and related distributions.

Usage

```r
assetsFit(x, method = c("st", "sn", "sc"),
          title=NULL, description=NULL, fixed.df=NA, ...)

assetsSim(n, method = c("st", "sn", "sc"),
          model=list(beta=rep(0, 2), Omega=diag(2), alpha=rep(0, 2), nu=4),
          assetNames=NULL)
```

Arguments

- **x**: any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.
n | a numeric value which represents the number of random vectors to be drawn.
method | a character string with the names of the supported distributions: sn skew normal, st skew Student-t, and sc skew Cauchy
model | a list with the model parameters. beta a numeric vector, representing the location, Omega a symmetric positive-definite matrix (covariance matrix), alpha a numeric vector which regulates the skew of the density, nu a positive value representing the degrees of freedom.
fixed.df | a logical value, should the degrees of freedom fitted or held fixed?
title | an optional project title.
description | an option project description.
assetNames | a character vector with optional asset names.
... | optional arguments passed to the underlying functions.

Value
assetsFit returns the fitted parameters, assetsSim returns a simulated (return) series.

Author(s)
Diethelm Wuertz for the Rmetrics port.

References
Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples
```r
## LPP2005REC -
# Load Swiss Pension Fund Data as Percentual Returns:
LPP <- 100 * LPP2005REC[, 1:3]
head(LPP)

## assetsFit -
# Fit a Skew-Student-t Distribution:
fit <- assetsFit(LPP)
# Extract the Model:
model <- fit@fit$dp
# Show Model Slot:
print(model)

## assetsSim -
# Simulate set with same statistical properties:
set.seed(1953)
LPP.SIM <- assetsSim(n=nrow(LPP), model=model)
colnames(LPP.SIM) <- colnames(LPP)
head(LPP.SIM)
```
assets-outliers  

Detection of Outliers in Asset Sets

Description

Detects multivariate outliers in asset sets.

Usage

assetsOutliers(x, center, cov, ...)

Arguments

- **x**: an object of class `timeSeries`.
- **center**: a numeric vector, a (robust) estimate of the vector of means of the multivariate time series `x`.
- **cov**: a numeric matrix, a (robust) estimate of the covariance matrix of the multivariate time series `x`.
- **...**: optional arguments to be passed.

Value

returns a list with the following entries: the estimate for the location named center, the estimate for the covariance matrix named cov, the estimate for the correlation matrix named cor, the quantile named quantile, the outliers named outliers, and the time series named series.

Author(s)

Moritz Gschwandtner and Peter Filzmoser for the original R code from package "mvoutliers", Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); *Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

```r
## LPP -
LPP <- as.timeSeries(data(LPP2005REC))[, 1:6]
colnames(LPP)

## assetsOutliers -
assetsOutliers(LPP, colMeans(LPP), cov(LPP))
```
Selecting Assets from Multivariate Asset Sets

Description

Select assets from Multivariate Asset Sets based on clustering.

Usage

assetsSelect(x, method = c("hclust", "kmeans"), control = NULL, ...)

Arguments

x any rectangular time series object which can be converted by the function as.matrix() into a matrix object, e.g. like an object of class timeSeries, data.frame, or mts.

method a character string, which clustering method should be used? Either hclust for hierarchical clustering of dissimilarities, or kmeans for k-means clustering.

control a character string with two entries controlling the parameters used in the underlying cluster algorithms. If set to NULL, then default settings are taken: For hierarchical clustering this is method=c(measure="euclidean", method="complete"), and for kmeans clustering this is method=c(centers=3, algorithm="Hartigan-Wong").

... optional arguments to be passed. Note, for the k-means algorithm the number of centers has to be specified!

Details

The function assetsSelect calls the functions hclust or kmeans from R's "stats" package. hclust performs a hierarchical cluster analysis on the set of dissimilarities hclust(dist(t(x))) and kmeans performs a k-means clustering on the data matrix itself.

Note, the hierarchical clustering method has in addition a plot method.

Value

if use="hclust" was selected then the function returns a S3 object of class "hclust", otherwise if use="kmeans" was selected then the function returns an object of class "kmeans".

For details we refer to the help pages of hclust and kmeans.

Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.
Examples

```r
## LPP -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC
colnames(LPP)

## assetsSelect -
# Hierarchical Clustering:
hclust <- assetsSelect(LPP, "hclust")
plot(hclust)

## assetsSelect -
# kmeans Clustering:
assetsSelect(LPP, "kmeans", control =
c(c(centers = 3, algorithm = "Hartigan-Wong"))
```

assets-testing  Testing Normality of Multivariate Asset Sets

Description

Tests if the returns of a set of assets are normally distributed.

Usage

```r
assetsTest(x, method = c("shapiro", "energy"), Replicates = 99)
mvshapiroTest(x)
mveneryTest(x, Replicates = 99)
```

Arguments

- `x` any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.
- `method` a character string, which allows to select the test. If `method="shapiro"` then Shapiro's multivariate Normality test will be applied as implemented in R's contributed package `mvnormtest`. If `method="energy"` then the E-statistic (energy) for testing multivariate Normality will be used as proposed and implemented by Szekely and Rizzo [2005] using parametric bootstrap.
- `Replicates` an integer value, the number of bootstrap replicates, by default 100. This value is only used if `method="energy"`.  

Value

returns an object of class `htest`.
Author(s)

Diethelm Wuertz for this Rmetrics port.

References

Rizzo M.L. (2002); A New Rotation Invariant Goodness-of-Fit Test, PhD dissertation, Bowling Green State University.


Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

```r
## LPP -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC[, 1:6]
head(LPP)

## assetsTest -
# Multivariate Shapiro Test -
assetsTest(LPP, "shapiro")

## assetsTest -
# Multivariate Energy Test -
assetsTest(LPP, "energy")
```

Estimation of Mean and Covariances of Asset Sets

Description

Helper functions for estimating the mean and/or covariance matrix of a time series of assets by traditional and robust methods.

Usage

```r
.baggedMeanCov(x, baggedR = 100, ...)
.bayesSteinMeanCov(x, ...)
.cov.arw(x, center, cov, alpha = 0.025, pcrit = NULL)
.cov.nnve(datamat, k = 12, pnoise = 0.05, emconv = 0.001, bound = 1.5,
extension = TRUE, devsm = 0.01)
.cov.shrink(x, lambda, verbose = FALSE)
.donostahMeanCov(x, ...)
```
.ledoitWolfMeanCov(x, ...)  
.rmtMeanCov(x, ...)  
.studentMeanCov(x, ...)

Arguments

- **x**: any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.  
- **baggedR**: when method="bagged", an integer value, the number of bootstrap replicates, by default 100.  
- **center**: specifies for a data set (n x p), the initial location estimator(1 x p).  
- **cov**: Initial scatter estimator (p x p).  
- **alpha**: Maximum thresholding proportion (optional scalar, default: alpha = 0.025).  
- **pcrit**: critical value for outlier probability (optional scalar, default values from simulations).  
- **datamat**: a matrix in which each row represents an observation or point and each column represents a variable.  
- **k**: desired number of nearest neighbors (default is 12).  
- **pnoise**: percent of added noise  
- **emconv**: convergence tolerance for EM.  
- **bound**: value used to identify surges in variance caused by outliers wrongly included as signal points (bound = 1.5 means a 50 percent increase).  
- **extension**: whether or not to continue after reaching the last chi-square distance. The default is to continue, which is indicated by setting `extension= TRUE`.  
- **devsm**: when extension = TRUE, the algorithm stops if the relative difference in variance is less than devsm (default is 0.01).  
- **lambda**: the correlation shrinkage intensity (range 0-1). If lambda is not specified (the default) it is estimated using an analytic formula from Schaefer and Strimmer (2005) - see details below. For lambda=0 the empirical correlations are recovered.  
- **verbose**: a logical indicating whether to print progress information to the stdout.  
- **...**: optional arguments to be passed to the underlying estimators. For details we refer to the manual pages of the functions `cov.rob` in the R package MASS, to the functions `covMcd` and `covOGK` in the R package robustbase.

Value

The functions return a list with elements containing the covariance and mean. The list may contain additional control parameters.
Bivariate Histogram Plots of Assets

Description
Displays bivariate histogram plots of assets returns.

Usage
assetsHistPairsPlot(x, bins = 30, method = c("square", "hex"), ...)

Arguments
x any rectangular time series object which can be converted by the function as.matrix() into a matrix object, e.g. like an object of class timeSeries, data.frame, or mts.
bins an integer value, the number of bins used for the bivariate histogram.
method a character string denoting which type of binning should be used, either "squared" or "hexagonal".
... optional arguments to be passed.

Author(s)
Diethelm Wuertz for the Rmetrics port.

References
Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples
## LPP2005REC
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC
head(LPP)

## assetsHistPairsPlot
# Create a bivariate Binning Plot: assetsHistPairsPlot
assetsHistPairsPlot(LPP[, c("LMI", "ALT")])

## assetsHistPairsPlot
# Now with hexagonal Bins:
assetsHistPairsPlot(LPP[, c("LMI", "ALT")], method = "hex")
gridd(col="red")
plot-boxplot Displays a Box Plot of Assets

Description

Displays standard box and box-percentile plots of assets.

Usage

assetsBoxPlot(x, col = "bisque", ...)  
assetsBoxPercentilePlot(x, col = "bisque", ...)

Arguments

x

any rectangular time series object which can be converted by the function as.matrix() into a matrix object, e.g. like an object of class timeSeries, data.frame, or mts.

col

a character string, defining the color to fill the boxes.

...

optional arguments to be passed.

Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

## LPP -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC[, 1:6]
head(LPP)

## assetsBoxPlot -
# Create a Boxplot: assetsBoxPlot -
assetsBoxPlot(LPP)

## assetsBoxPercentilePlot -
# Create a Box Percentile Plot: assetsBoxPercentilePlot -
assetsBoxPercentilePlot(LPP)
grid(NA, NULL, col="red")
plot-ellipses

Displays a Covariance Ellipses Plot

Description
Displays a covariance ellipses plot.

Usage

covEllipsesPlot(x = list(), ...)

Arguments

x
a list of at least two covariance matrices.

... optional arguments to be passed.

Details
This plot visualizes the difference between two or more covariance matrices. It is meant to compare different methods of covariance estimation.

References
Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

## LPP -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC[, 1:6]
head(LPP)

## assetsMeanCov -
# Compute Robust Covariance Matrix: assetsMeanCov -
Cov <- cov(LPP)
robustCov <- assetsMeanCov(LPP, "MCD")$Sigma

## covEllipsesPlot -
# Create Covariance Ellipse Plot:
covEllipsesPlot(list(Cov, robustCov))
plot-hist  

**Histogram Plots of Assets**

**Description**

Displays density of assets returns as a histogram and/or as log density plot.

**Usage**

```r
assetsHistPlot(x, col = "steelblue", skipZeros = FALSE, ...)
assetsLogDensityPlot(x, estimator = c("hubers", "sample", "both"),
                    labels = TRUE, ...)
```

**Arguments**

- `x`: any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.
- `skipZeros`: a logical, should zeros be skipped in the histogram plot of the return series?
- `col`: a character string, defining the color to fill the boxes.
- `estimator`: a character string naming the type of estimator to fit the mean and variance of the normal density. This may be either "huber", "sample", or "both".
- `labels`: a logical flag, if `TRUE` then default labels will be used, otherwise the plots will be displayed without labels and the user can add his own labels.
- `...`: optional arguments to be passed.

**Author(s)**

Diethelm Wuertz for the Rmetrics port.

**References**

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); *Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

**Examples**

```r
## LPP2005REC -
# Load Swiss Pension Fund Data:
x <- LPP2005REC
head(x)

## assetsHistPlot -
# Create Histogram Plot: assetsHistPlot -
# par(mfrow = c(2, 2))
assetsHistPlot(x[, 1:4])
```
## assetsLogDensityPlot

### Create Log Density Plot: assetsLogDensityPlot

```r
# par(mfrow = c(1, 1))
assetsLogDensityPlot(x[, "ALT"], estimator = "both")
```

---

**plot-mst**  
**Assets Tree Plot**

---

**Description**

Creates and displays a minimum spanning tree of assets.

**Usage**

```r
assetsTreePlot(x, labels = TRUE, title = TRUE, box = TRUE,
               method = "euclidian", seed = NULL, ...)
```

**Arguments**

- `x`: a multivariate `timeSeries` object.
- `labels`: a logical flag, if `TRUE` then default labels will be used, otherwise the plots will be displayed without labels and the user can add his own labels.
- `title`: a logical flag, should a default title be added? By default `TRUE`.
- `box`: a logical flag, should a box be added around the plot? By default `TRUE`.
- `method`: a character string, the method used to compute the distance matrix, see function `dist`.
- `seed`: an integer value setting the seed in the computation of the sample ranks.
- `...`: optional arguments to be passed.

**Author(s)**

Diethelm Wuertz for the Rmetrics port.

**References**

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); *Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.
Examples

```r
## LPP2005REC -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC[, 1:6]
head(LPP)

## assetsTreePlot(LPP) -
# Create Minimum Spanning Tree Graph: assetsTreePlot -
# par(mfrow = c(2, 2))
assetsTreePlot(LPP)
# new seeds ...
for (i in 1:3) assetsTreePlot(LPP)
```

plot-pairs

### Assets Pairs Plot

**Description**

Display several aspects of correlation between pairs of assets.

**Usage**

```r
assetsPairsPlot(x, ...)  
assetsCorgramPlot(x, 
                 method = c("pie", "shade"), ...)  
assetsCorTestPlot(x, ...)  
assetsCorImagePlot(x, labels = TRUE, show = c("cor", "test"), 
                    use = c("pearson", "kendall", "spearman"), abbreviate = 3, ...)  
```

**Arguments**

- `x`: any rectangular time series object which can be converted by the function as.matrix() into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.
- `labels`: a logical flag, if `TRUE` then default labels will be used, otherwise the plots will be displayed without labels and the user can add his own labels.
- `method`: a character string, the type of graph used in the lower panel.
- `show`: a character string, what should be presented, correlations or results from correlation tests?
- `use`: a character string indicating which correlation coefficient or covariance is to be computed. One of "pearson", the default, "kendall", or "spearman".
- `abbreviate`: allows to abbreviate strings to at least abbreviate characters, such that they remain unique, if they were.
- `...`: optional arguments to be passed.
Details

assetsPairsPlot
displays pairs of scatterplots of individual assets,

assetsCorgramPlot
displays correlations between assets,

assetsCorTestPlot
displays and tests pairwise correlations,

assetsCorImagePlot
displays an image plot of a correlation.

Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

## LPP2005REC -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC[, 1:6]
head(LPP)

## assetsPairsPlot -
# Create Pairs Plot:
assetsPairsPlot(LPP)

## assetsCorgramPlot -
# Create Corellogram Plot:
assetsCorgramPlot(LPP, method = "pie")
assetsCorgramPlot(LPP, method = "shade")

## assetsCorTestPlot -
# Create Correlation Test Plot:
assetsCorTestPlot(LPP)

## assetsCorImagePlot -
# Create Correlation Image Plot:
assetsCorImagePlot(LPP)
Normal Quantile-Quantile Plots

Description

Displays a normal quantile-quantile plot

Usage

assetsQQNormPlot(x, col = "steelblue", skipZeros = FALSE, ...)

Arguments

x 
any rectangular time series object which can be converted by the function as.matrix() into a matrix object, e.g. like an object of class timeSeries, data.frame, or mts.

col 
a character string, defining the color to fill the boxes.

skipZeros 
a logical, should zeros be skipped in the histogram plot of the return series?

... 
optional arguments to be passed.

Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

## LPP2005REC -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC
head(LPP)

## assetsQQNormPlot -
# Create normal Quantile-Quantile Plot:
# par(mfrow = c(2, 2))
assetsQQNormPlot(LPP[, 1:3])
Description

Displays risk plot from assets.

Usage

assetsRiskReturnPlot(x, col = "steelblue", percentage = FALSE, scale = 252,
                      labels = TRUE, add = TRUE, ...)

assetsNIGShapeTrianglePlot(x, labels = TRUE, col = "steelblue", ...)

Arguments

x
    any rectangular time series object which can be converted by the function
    as.matrix() into a matrix object, e.g. like an object of class timeSeries, data.frame,
    or mts.

col
    a character string, defining the color to fill the boxes.

percentage
    a logical flag. Are the returns given by log or percentual log returns?

scale
    an integer value, the scale, i.e number of days, in a year. Used by daily data
    sets.

labels
    a logical flag, if TRUE then default labels will be used, otherwise the plots will
    be displayed without labels and the user can add his own labels.

add
    a logical flag, defining the color to fill the boxes.

... optional arguments to be passed.

Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics
eBook, Rmetrics Association and Finance Online, Zurich.

Examples

## LPP2005REC -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC
head(LPP)

## assetsRiskReturnPlot -
# Create Risk/Return Plot:
# par(mfrow = c(2, 2))
assetsRiskReturnPlot(LPP)

## assetsNIGShapeTrianglePlot -
# Create NIG Shape Triangle Plot:
assetsNIGShapeTrianglePlot(LPP)

plot-series Displays Series Plots of Assets.

Description
Displays series from sets of assets.

Usage
assetsReturnPlot(x, col = "steelblue", ...)  
assetsCumulatedPlot(x, col = "steelblue", ...)  
assetsSeriesPlot(x, col = "steelblue", ...)

Arguments
x an object of class timeSeries.
col a character string, defining the color to fill the boxes.
... optional arguments to be passed.

Author(s)
Diethelm Wuertz for the Rmetrics port.

References
Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); Portfolio Optimization with R/Rmetrics, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples
## LPP2005REC -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC
head(LPP)

## assetsReturnPlot -
# Create Return Series Plot:
# par(mfrow = c(3, 2))
assetsReturnPlot(LPP[, 1:3])

## assetsCumulatedPlot -
# Create Cumulated Price/Index Plot:
assetsCumulatedPlot(LPP[, "LPP40"], col = "red")

## assetsSeriesPlot
# Create Time Series Plot:
assetsSeriesPlot(LPP[, c("LMI", "ALT")],
  col = c("orange", "brown"))

---

## plot-similarity
### Assets Similarity Plots

**Description**

Displays plots of similarities and dissimilarities between data sets of assets.

**Usage**

assetsDendrogramPlot(x, labels = TRUE, title = TRUE, box = TRUE,
  method = c(dist = "euclidian", clust = "complete"), ...)

assetsCorEigenPlot(x, labels = TRUE, title = TRUE, box = TRUE,
  method = c("pearson", "kendall", "spearman"), ...)

**Arguments**

- **box**
  a logical flag, should a box be added around the plot? By default TRUE.
- **labels**
  a logical flag, if TRUE then default labels will be used, otherwise the plots will be displayed without labels and the user can add his own labels.
- **method**
  [assetsCorgramPlot] - a character string, the type of graph used in the lower panel, for the function assetsCorgramPlot a character string, the method used to compute the correlation matrix.
  [assetsTreePlot] - a character string, the method used to compute the distance matrix, see function dist.
- **title**
  a logical flag, should a default title be added? By default TRUE.
- **x**
  any rectangular time series object which can be converted by the function as.matrix() into a matrix object, e.g. like an object of class timeSeries, data.frame, or mts.
- **...**
  optional arguments to be passed.

**Details**

- assetsDendrogramPlot displays a hierarchical clustering dendrogram,
- assetsCorEigenPlot displays ratio plot of the largest two eigenvalues.
Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); *Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

```r
## LPP2005REC -
# Load Swiss Pension Fund Data:
LPP <- LPP2005REC
head(LPP)

## assetsDendrogramPlot -
# Display a Dendrogram Plot:
assetsDendrogramPlot(LPP)

## assetsCorEigenPlot -
# Display a Correlation Eigenvalue Ratio Plot:
assetsCorEigenPlot(LPP)
```

Description

Displays star plots to compare assets sets.

Usage

```r
assetsStarsPlot(x, method = c("segments", "stars"), locOffset = c(0, 0),
keyOffset = c(0, 0), ...)

assetsBoxStatsPlot(x, par = TRUE, oma = c(0,0,0,0), mar = c(4, 4, 4, 4),
keyOffset = c(-0.65, -0.50), main = "Assets Statistics",
title = "Assets", titlePosition = c(3, 3.65),
description = "Box Plot Statistics", descriptionPosition = c(3, 3.50), ...)

assetsBasicStatsPlot(x, par = TRUE, oma = c(0,0,0,0), mar = c(4, 4, 4, 4),
keyOffset = c(-0.65, -0.50), main = "Assets Statistics",
title = "Assets", titlePosition = c(3, 3.65),
description = "Basic Returns Statistics", descriptionPosition = c(3, 3.50), ...)

assetsMomentsPlot(x, par = TRUE, oma = c(0,0,0,0), mar = c(4, 4, 4, 4),
keyOffset = c(-0.65, -0.50), main = "Assets Statistics",
title = "Assets", titlePosition = c(3, 3.65),
```

plot-stars

*Stars Plots of Assets.*
Plotting functions for assets statistics:

```r
assetsNIGFitPlot(x, par = TRUE, oma = c(0,0,0,0), mar = c(4, 4, 4, 4),
    keyOffset = c(-0.65, -0.50), main = "Assets Statistics",
    title = "Assets", titlePosition = c(3, 3.65),
    description = "NIG Parameters", descriptionPosition = c(3, 3.50), ...)
```

**Arguments**

- `description` a description string.
- `descriptionPosition` the position of the description string.
- `method` a character string from which to select the plot method. Either a "star" or a "segment" plot.
- `keyOffset` a numeric vector of length two, specifying an offset in the legend with respect to x and y direction.
- `locOffset` a numeric vector of length two, specifying an offset in the location of the stars/circles with respect to x and y direction.
- `main` to set the main title.
- `mar` to set the number of lines of margin to be specified on the four sides of the plot. The default is `c(5,4,4,2)+0.1`.
- `oma` to set the size of the outer margins in lines of text.
- `par` a logical flag. Should be `internal_par()` setting be used?
- `title` a character string, the plot title.
- `titlePosition` the position of the title string.
- `x` any rectangular time series object which can be converted by the function `as.matrix()` into a matrix object, e.g. like an object of class `timeSeries`, `data.frame`, or `mts`.
- `...` optional arguments to be passed.

**Details**

- `assetsStarsPlot` draws segment or star diagrams of data sets.
- `assetsBasicStatsPlot` displays a segment plot of box plot statistics.
- `assetsMomentsPlot` displays a segment plot of distribution moments.
- `assetsBoxStatsPlot` displays a segment plot of box plot statistics.
- `assetsNIGFitPlot` displays a segment plot NIG parameter estimates.
Author(s)

Diethelm Wuertz for the Rmetrics port.

References

Wuertz, D., Chalabi, Y., Chen W., Ellis A. (2009); *Portfolio Optimization with R/Rmetrics*, Rmetrics eBook, Rmetrics Association and Finance Online, Zurich.

Examples

```r
## LPP2005REC - # Load Swiss Pension Fund Data:
LPP <- LPP2005REC
head(LPP)

## assetsBasicStatsPlot - # Create a basic Stats Plot: assetsBasicStatsPlot -
## par(mfrow = c(1, 1))
assetsBasicStatsPlot(LPP, title = "", description = "")

## assetsMomentsPlot - # Create a Moments Plot: assetsMomentsPlot -
assetsMomentsPlot(LPP, title = "", description = "")

## assetsBoxStatsPlot - # Create a Box Stats Plot: assetsBoxStatsPlot -
assetsBoxStatsPlot(LPP, title = "", description = "")

## assetsNIGFitPlot - # Create a NIG Fit Plot: assetsNIGFitPlot -
assetsNIGFitPlot(LPP[, 7:9], title = "", description = "")
```
Index

* **models**
  - assets-arrange, 6
  - assets-lpm, 9
  - assets-meancov, 10
  - assets-modeling, 12
  - assets-outliers, 14
  - assets-selection, 15
  - assets-testing, 16
  - fAssets-package, 2
  - plot-binning, 19
  - plot-boxplot, 20
  - plot-ellipses, 21
  - plot-hist, 22
  - plot-mst, 23
  - plot-pairs, 24
  - plot-qqplot, 26
  - plot-risk, 27
  - plot-series, 28
  - plot-similarity, 29
  - plot-stars, 30

* **package**
  - fAssets-package, 2

* **stats**
  - assets-distance, 7
  - baggedMeanCov (builtin), 17
  - bayesSteinMeanCov (builtin), 17
  - cov.arw (builtin), 17
  - cov.nnve (builtin), 17
  - cov.shrink (builtin), 17
  - donostahMeanCov (builtin), 17
  - ledoitWolfMeanCov (builtin), 17
  - rmtMeanCov (builtin), 17
  - studentMeanCov (builtin), 17

  abcArrange (assets-arrange), 6
  assets-arrange, 6
  assets-distance, 7
  assets-lpm, 9
  assets-meancov, 10
  assets-modeling, 12
  assets-outliers, 14
  assets-selection, 15
  assets-testing, 16
  assetsArrange (assets-arrange), 6
  assetsBasicStatsPlot (plot-stars), 30
  assetsBoxPercentilePlot (plot-boxplot), 20
  assetsBoxPlot (plot-boxplot), 20
  assetsBoxStatsPlot (plot-stars), 30
  assetsCorEigenPlot (plot-similarity), 29
  assetsCorgramPlot (plot-pairs), 24
  assetsCorImagePlot (plot-pairs), 24
  assetsCorTestPlot (plot-pairs), 24
  assetsCumulatedPlot (plot-series), 28
  assetsDendrogramPlot (plot-similarity), 29
  assetsDist (assets-distance), 7
  assetsFit (assets-modeling), 12
  assetsHistPairsPlot (plot-binning), 19
  assetsHistPlot (plot-hist), 22
  assetsLogDensityPlot (plot-hist), 22
  assetsLPM (assets-lpm), 9
  assetsMeanCov (assets-meancov), 10
  assetsMomentsPlot (plot-stars), 30
  assetsNIGFitPlot (plot-stars), 30
  assetsNIGShapeTrianglePlot (plot-risk), 27
  assetsOutliers (assets-outliers), 14
  assetsPairsPlot (plot-pairs), 24
  assetsQQNormPlot (plot-qqplot), 26
  assetsReturnPlot (plot-series), 28
  assetsRiskReturnPlot (plot-risk), 27
  assetsSelect (assets-selection), 15
  assetsSeriesPlot (plot-series), 28
  assetsSim (assets-modeling), 12
  assetsSLPM (assets-lpm), 9
  assetsStarsPlot (plot-stars), 30
  assetsTest (assets-testing), 16
  assetsTreePlot (plot-mst), 23
binaryDist (assets-distance), 7
binningPlot (plot-binning), 19
boxPlot (plot-boxplot), 20
braycurtisDist (assets-distance), 7
builtIn, 17
canberraDist (assets-distance), 7
corDist (assets-distance), 7
covEllipsesPlot (plot-ellipses), 21
euclideanDist (assets-distance), 7
fAssets (fAssets-package), 2
fAssets-package, 2
getCenterRob (assets-meancov), 10
getCovRob (assets-meancov), 10
hclustArrange (assets-arrange), 6
histPlot (plot-hist), 22
jaccardDist (assets-distance), 7
kendallDist (assets-distance), 7
mahalanobisDist (assets-distance), 7
manhattanDist (assets-distance), 7
maximumDist (assets-distance), 7
minkowskiDist (assets-distance), 7
mutinfoDist (assets-distance), 7
mvenergyTest (assets-testing), 16
mvshapiroTest (assets-testing), 16
orderArrange (assets-arrange), 6
pairsPlot (plot-pairs), 24
pcaArrange (assets-arrange), 6
plot-binning, 19
plot-boxplot, 20
plot-ellipses, 21
plot-hist, 22
plot-mst, 23
plot-pairs, 24
plot-qqplot, 26
plot-risk, 27
plot-series, 28
plot-similarity, 29
plot-stars, 30
sampleArrange (assets-arrange), 6
seriesPlot (plot-series), 28
seriesPlots (plot-risk), 27
similarityPlot (plot-similarity), 29
sorensenDist (assets-distance), 7
spearmanDist (assets-distance), 7
starsPlot (plot-stars), 30
statsArrange (assets-arrange), 6
treePlot (plot-mst), 23