Package 'TSEtools'

January 8, 2026

Type Package

Title Manage Data from the Finance Markets

Version 1.0.0

Maintainer Ali Saeb <ali.saeb@gmail.com>

Description A set of tools designed to perform descriptive data analysis on assets, manage asset portfolios and capital allocation, and download, organize, and maintain data from the ``Tehran Stock Exchange'' and ``NOBITEX'' platforms.

License BSD_2_clause + file LICENSE

Encoding UTF-8

Imports xts, quantmod, quadprog, httr, rjson

NeedsCompilation no

Repository CRAN

Date/Publication 2026-01-08 12:40:02 UTC

RoxygenNote 7.3.3

Author Ali Saeb [aut, cre]

Contents

`cancel` *Cancel a group of orders.*

Description

Cancel a group of orders based on currency name or record time.

Usage

```
cancel(src, dst, hrs = 0, token)
```

Arguments

`src`	Source currency name ("btc", "eth", "bnb", etc).
`dst`	Destination currency name ("usdt" or "rls").
`hrs`	The number of hours after which orders will be canceled. If the value is less than one hour, it is specified as a decimal (e.g., 15 minutes is given by `hrs = 0.25`).
`token`	The token is string value obtain from user ID settings in the NOBITEX. If you are not registered on the website, please use the provided link.

Examples

```
## Not run:
tkn <- "CONTACT WITH AUTHOR"

# All orders recorded more than one hour ago are canceled.
cancel(" ", " ", hrs = 1, token = tkn)

## End(Not run)
```

capm *Capital assets pricing model including a risk-free asset.*

Description

Compute the capital assets pricing model including a risk-free asset.

Usage

```
capm(x, Rf = 0.2/270, sh = FALSE, eRtn = NULL)
```

Arguments

x	a numeric matrix of random returns per unit of price within some holding period.
Rf	the return of the risk free, i.e. has variance 0.
sh	a logical indicating whether shortsales on the risky securities are allowed. Default is FALSE.
eRtn	a value of expected returen of portofilo. The mean of whole data defualt.

Details

Let $\xi_1, \ldots, \xi_n$ be random asset returns and $w_1, \ldots, w_n$ the portfolio weights. The expected returns are $r_m = E\xi_m, m = 1, \ldots, n$. In addition to these risky investments, there is a risk-free asset (a bond or bank account) available, which has return r_0. Denoting the weights of w_0 for the risk-free asset. The return of portfolio given by

$$R_p = w^t r$$

where, $r = (r_1, \ldots, r_n)^t$.

Risk is measure by a deviation functional Σ. It is a variance-covariance of asset returns. The risk-free component w_0 ignore in the objective. So, the standard deviation of portfolio is given by $\sigma_p = w^t \Sigma w$.

To obtain the optimum value of $w_i, i = 1, \ldots, n$, we solve the following model:

$$\min w^t \Sigma w, \quad s.t: \quad w^t r + w_0 r_0 > \mu \ \ and \ \ \sum w_i + w_0 = 1$$

Note that, the portfolio weights may be negative (selling short is allowed). Market portfolio is named `MP` where, the risk free weight `w_0` is zero (see, the function of `prtf()`).

For any portfolio p,

$$E(R_p) = r_0 + \beta(p)(r_{MP} - r_0)$$

where, r_{MP} is return of market portfolio and $\beta(p)$ is the beta coefficient of the portfolio p. It is given by $\beta(p) = Cov(r_{MP}, r_p)/SD(r_{MP})$.

Value

wCAPM	weight of CAPM assets
wrF	weight of risk free assets
sd.capm	volatility of CAPM portfolio
rtn.capm	return of CAPM portfolio
beta	beta coefficient of portfolio

References

Pflug and Romisch (2007, ISBN: 9789812707406)

Examples

```
## Not run:
x <- rnorm(500,0.05,0.02)
y <- rnorm(500,0.01,0.03)
z<-cbind(x, y)
colnames(z) <- c("prt1","prt2")

capm( z, sh = FALSE, Rf= 0.2/270, eRtn=0.02 )

## End(Not run)
```

---

depth *Get the bid and ask prices with the cumulative volume*

Description

Get a cumulative volumes corresponding to the bid and ask's price

Usage

```
depth(src)
```

Arguments

src	List of the some crypto curencies which available in the function. BTCIRT, ETHIRT, LTCIRT, USDTIRT, XRPIRT, BCHIRT, BNBIRT, EOSIRT, XLMIRT, ETCIRT, TRXIRT, DOGEIRT, UNIIRT, DAIIRT, LINKIRT, DOTIRT, AAVEIRT, ADAIRT, SHIBIRT, FTMIRT, MATICIRT, AXSIRT, MANAIRT, SANDIRT, AVAXIRT, MKRIRT, GMTIRT, USDCIRT, BTCUSDT, ETHUSDT, LTCUSDT, XRPUSDT, BCHUSDT, BNBUSDT, EOSUSDT, XLMUSDT, ETCUSDT, TRXUSDT, PMNUSDT, DOGEUSDT, UNIUSDT, DAIUSDT, LINKUSDT, DOTUSDT, AAVEUSDT, ADAUSDT, SHIBUSDT, FTMUSDT, MATICUSDT, AXSUSDT, MANAUSDT, SANDUSDT, AVAXUSDT, MKRUSDT, GMTUSDT, USDCUSDT, CHZIRT, GRTIRT, CRVIRT, BANDUSDT, COMPUSDT, EGLDIRT, HBARUSDT, GALIRT, HBARIRT, WBTCUSDT, IMXIRT, WBTCIRT, ONEIRT,

GLMUSDT, ENSIRT, 1M_BTTIRT, SUSHIIRT, LDOIRT, ATOMUSDT, ZROIRT, STORJIRT, ANTIRT, AEVOUSDT, 100K_FLOKIIRT, RSRUSDT, API3USDT, GLMIRT, XMRIRT, ENSUSDT, OMIRT, RDNTIRT, MAGICUSDT, TIRT, ATOMIRT, NOTIRT, CVXIRT, XTZIRT, FILIRT, UMAIRT, 1B_BABYDOGEIRT, BANDIRT, SSVIRT, DAOIRT, BLURIRT, ONEUSDT, EGALAUSDT, GMXIRT, XTZUSDT, FLOWUSDT, GALUSDT, WIRT, CVCUSDT, NMRUSDT, SKLIRT, SNTIRT, BATUSDT, TRBUSDT, NMRIRT, RDNTUSDT, API3IRT, CVCIRT, WLDIRT, YFIUSDT, SOLIRT, TUSDT, QNTUSDT, IMXUSDT, AEVOIRT, GMXUSDT, ETHFIUSDT, QNTIRT, GRTUSDT, WLDUSDT, FETIRT, AGIXIRT, NOTUSDT, LPTIRT, SLPIRT, MEMEUSDT, SOLUSDT, BALUSDT, DAOUSDT, COMPIRT, MEMEIRT, TONUSDT, BATIRT, SNXIRT, TRBIRT, 1INCHUSDT, OMUSDT, RSRIRT, RNDRIRT, SLPUSDT, SSVUSDT, RNDRUSDT, AGLDIRT, NEARUSDT, WOOUSDT, YFIIRT, MDTIRT, CRVUSDT, MDTUSDT, EGLDUSDT, LRCIRT, LPTUSDT, BICOUSDT, 1M_PEPEIRT, BICOIRT, MAGICIRT, ETHFIIRT, ANTUSDT, 1INCHIRT, APEUSDT, 1M_NFTIRT, ARBIRT, LRCUSDT, WUSDT, BLURUSDT, CELRUSDT, DYDXIRT, CVXUSDT, BALIRT, TONIRT, 100K_FLOKIUSDT, JSTUSDT, ZROUSDT, ARBUSDT, APTIRT, 1M_NFTUSDT, CELRIRT, UMAUSDT, SKLUSDT, ZRXUSDT, AGLDUSDT, ALGOIRT, NEARIRT, APTUSDT, ZRXIRT, SUSHIUSDT, FETUSDT, ALGOUSDT, 1M_PEPEUSDT, MASKIRT, EGALAIRT, FLOWIRT, 1B_BABYDOGEUSDT, MASKUSDT, 1M_BTTUSDT, STORJUSDT, XMRUSDT, OMGIRT, SNTUSDT, APEIRT, FILUSDT, ENJUSDT, OMGUSDT, WOOIRT, CHZUSDT, ENJIRT, DYDXUSDT, AGIXUSDT, JSTIRT, LDOUSDT, SNXUSDT

Value

The values return as a data frame format. The names of the dimensions are
prc.ask: last 75 prices of the asks
vol.ask: a list of last 75 cumulative volumes corresponding to the ask's price.
prc.bid: last 75 prices of the bids
vol.bid: a list of last 75 cumulative volumes corresponding to the bid's price.

Examples

```
# Download daily data from last one month
## Not run:
depth("BTCUSDT")

## End(Not run)
```

`getNBTX` *Get the data of the crypto currency from NOBITEX*

Description

Download the history data of the crypto currencies from the Nobitex (website of "nobitex.ir").

Usage

```
getNBTX(src,frame,from,to)
```

Arguments

src	List of the some crypto curencies which available in the function. BTCIRT, ETHIRT, LTCIRT, USDTIRT, XRPIRT, BCHIRT, BNBIRT, EOSIRT, XLMIRT, ETCIRT, TRXIRT, DOGEIRT, UNIIRT, DAIIRT, LINKIRT, DOTIRT, AAVEIRT, ADAIRT, SHIBIRT, FTMIRT, MATICIRT, AXSIRT, MANAIRT, SANDIRT, AVAXIRT, MKRIRT, GMTIRT, USDCIRT, BTCUSDT, ETHUSDT, LTCUSDT, XRPUSDT, BCHUSDT, BNBUSDT, EOSUSDT, XLMUSDT, ETCUSDT, TRXUSDT, PMNUSDT, DOGEUSDT, UNIUSDT, DAIUSDT, LINKUSDT, DOTUSDT, AAVEUSDT, ADAUSDT, SHIBUSDT, FTMUSDT, MATICUSDT, AXSUSDT, MANAUSDT, SANDUSDT, AVAXUSDT, MKRUSDT, GMTUSDT, USDCUSDT, CHZIRT, GRTIRT, CRVIRT, BANDUSDT, COMPUSDT, EGLDIRT, HBARUSDT, GALIRT, HBARIRT, WBTCUSDT, IMXIRT, WBTCIRT, ONEIRT, GLMUSDT, ENSIRT, 1M_BTTIRT, SUSHIIRT, LDOIRT, ATOMUSDT, ZROIRT, STORJIRT, ANTIRT, AEVOUSDT, 100K_FLOKIIRT, RSRUSDT, API3USDT, GLMIRT, XMRIRT, ENSUSDT, OMIRT, RDNTIRT, MAGICUSDT, TIRT, ATOMIRT, NOTIRT, CVXIRT, XTZIRT, FILIRT, UMAIRT, 1B_BABYDOGEIRT, BANDIRT, SSVIRT, DAOIRT, BLURIRT, ONEUSDT, EGALAUSDT, GMXIRT, XTZUSDT, FLOWUSDT, GALUSDT, WIRT, CVCUSDT, NMRUSDT, SKLIRT, SNTIRT, BATUSDT, TRBUSDT, NMRIRT, RDNTUSDT, API3IRT, CVCIRT, WLDIRT, YFIUSDT, SOLIRT, TUSDT, QNTUSDT, IMXUSDT, AEVOIRT, GMXUSDT, ETHFIUSDT, QNTIRT, GRTUSDT, WLDUSDT, FETIRT, AGIXIRT, NOTUSDT, LPTIRT, SLPIRT, MEMEUSDT, SOLUSDT, BALUSDT, DAOUSDT, COMPIRT, MEMEIRT, TONUSDT, BATIRT, SNXIRT, TRBIRT, 1INCHUSDT, OMUSDT, RSRIRT, RNDRIRT, SLPUSDT, SSVUSDT, RNDRUSDT, AGLDIRT, NEARUSDT, WOOUSDT, YFIIRT, MDTIRT, CRVUSDT, MDTUSDT, EGLDUSDT, LRCIRT, LPTUSDT, BICOUSDT, 1M_PEPEIRT, BICOIRT, MAGICIRT, ETHFIIRT, ANTUSDT, 1INCHIRT, APEUSDT, 1M_NFTIRT, ARBIRT, LRCUSDT, WUSDT, BLURUSDT, CELRUSDT, DYDXIRT, CVXUSDT, BALIRT, TONIRT, 100K_FLOKIUSDT, JSTUSDT, ZROUSDT, ARBUSDT, APTIRT, 1M_NFTUSDT, CELRIRT, UMAUSDT, SKLUSDT, ZRXUSDT, AGLDUSDT, ALGOIRT, NEARIRT, APTUSDT, ZRXIRT, SUSHIUSDT, FETUSDT, ALGOUSDT, 1M_PEPEUSDT, MASKIRT, EGALAIRT, FLOWIRT, 1B_BABYDOGEUSDT, MASKUSDT, 1M_BTTUSDT, STORJUSDT, XMRUSDT, OMGIRT, SNTUSDT, APEIRT, FILUSDT, ENJUSDT, OMGUSDT, WOOIRT, CHZUSDT, ENJIRT, DYDXUSDT, AGIXUSDT, JSTIRT, LDOUSDT, SNXUSDT
frame	A character value shows the data frame. The frame is defined as "1" is one minute. "5" is 5 minutes. "15" is 15 minutes. "30" is 30 minutes. "60" is an hour. "180" is three hours. "240" is four hours. "360" is six hours.

"720" is twelve hours.
"D" is a day.
"2D" is two days.
"3D" is three days.

`from` "2025-12-23 13:13:57 +0330" The character string in the date format ("%Y-%m-%d %H:%M:%S") shows the start date of the time frame. Since, the data is time series format, the rules of `POSIXct` satisfied.

`to` The character string in the date format ("%Y-%m-%d %H:%M:%S") shows the end date of the time frame. Since, the data is time series format, the rules of `POSIXct` satisfied.

Details

A symbol of the crypto currency can be named and the values will be assigned the variable. Component data is an array with **xts** time series object. The names for dimensions are: Open, High, Low, Close, Volume. The value of symbol are appear in `Global Environment`.

Value

The output is an object of the "cryto currency" assigned the variable's name.

Examples

```
## Not run:
# Download daily data from last one month
x <- getNBTX("BTCUSDT", frame = "D", from = paste0(Sys.Date()-30, substr(Sys.time(), 11, 19))
                                    , to=Sys.time())


## End(Not run)
```

`getTSE` *Download Historical Dataset from Tehran Stock Exchange (TSE)*

Description

getTSE function is an easy way to download and organize the historical dataset from websites of TSE (website of "tsetmc"). Since, the raw data are not recorded in a standard format, we provide the function for organize the data in user friendly way. Daily data updated after 13:30 UTC. The market is closed on Friday and Wensday and national holidays.

Usage

```
getTSE(file, symbols=NA)
```

Arguments

`file`	the path and name of external file which include the name of symbol and company and a specific code.
`symbols`	a vector of character given the name of assets to download. The name of assets has to be included in the first column of `file`. The default value is NA which consider all assets at source file to download.

Details

The list of symbols or assets can be named, and the list names will be used as names for the variables. Component data is an array with **xts** time series object. The name for dimensions are: Open, High, Low, Close, Volume, Last. The value of symbols are appear in `Global Environment`.

Value

The output is an object of the "assets" name which get all the name of assets list to download.

Examples

```
## Not run:
fpath<-system.file("extdata", "sample.dat", package="TSEtools")
getTSE(fpath)

## End(Not run)
```

`hello` *Hello, World!*

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

lastPrft *Daily Profit/Loss*

Description

Shows a list of daily profit/loss with percentage and balance for the latest week or month.

Usage

```
lastPrft(period = "weekly", token)
```

Arguments

period	Shows a list of daily profit/loss for a week or month using `weekly` or `monthly`. Default is `weekly`.
token	The token is string value obtain from user ID settings in the NOBITEX. If you are not registered on the website, please use the provided link.

Value

report_date	Date index of the data frame.
total_profit	Total daily profit/loss value.
total_profit_percentage	Total daily profit/loss percentage.
total_balance	Total balance at the end of the day.

Examples

```
## Not run:
tkn <- "CONTACT WITH AUTHOR"
lastPrft("monthly", token = tkn)

## End(Not run)
```

lst *List of trades*

Description

This function lists all completed trades.

Usage

```
lst(n=500, token)
```

Arguments

n	An integer value specifying the number of trades to display, with a maximum of 500.
token	The token is string value obtain from user ID settings in the NOBITEX. If you are not registered on the website, please use the provided link.

Value

srcCurrency	Source name of the cryptocurrency
dstCurrency	Destination cryptocurrency; values are Tether or Rials.
timestamp	Trade date and time.
market	Market names
price	Trade price.
amount	Trade amount
total	Total trade value.
type	Trade type (buy or sell).
fee	Trade fee.
id	Trade id.
orderId	Trade order ID.

Examples

```
## Not run:
tkn <- "CONTACT WITH AUTHOR"
x <- lst(n = 500, token = tkn)

## End(Not run)
```

---

marketInfo *The information on the last trades*

Description

Obtain information about the most recent trades conducted in the cryptocurrency market. This data can provide insights into the market trends and help investors make informed decisions.

Usage

```
marketInfo(src ,dst)
```

Arguments

src	List of the some crypto curencies which available in the function. btc, eth, ltc, usdt, xrp, bch, bnb, eos, xlm, etc, trx, pmn, doge, uni, dai, link, dot, aave, ada, shib, ftm, matic, axs, mana, sand, avax, mkr, gmt, atom, uma, w, rsr, wld, 1m_nft, flow, agld, ton, mask, snt, agix, algo, ssv, band, omg, comp, zrx, rdnt, imx, 1inch, mdt, sushi, bico, gmx, zro, bal, dao, gal, not, nmr, xmr, enj, apt, lrc, dydx, grt, near, cvx, 100k_floki, fil, sol, ldo, crv, aevo, qnt, om, woo, storj, ant, 1m_btt, magic, ape, rndr, hbar, lpt, glm, blur, wbtc, meme, ethfi, egala, arb, fet, skl, cvc, snx, jst, ens, trb, chz, xtz, slp, t, bat, celr, yfi, egld, one, usdc
dst	The destination currency can be chosen in two values: 'rls' (Rial) or 'usdt' (US. Dollar).

Value

The values return as format of a data frame. The colnames of the dimensions are
isClosed: Logit value. TRUE is market close and FALSE is market Open
bestSell: The first line price of bid list
bestBuy: The first line price of ask list
volumeSrc: The volume of the source (src) currency traded.
volumeDst: The volume of the distination (dst) currency traded.
latest: The last price traded.
mark: The mark price is a calculated by the bestSell, bestBuy and latest price.
dayLow: Lowest price in the day
dayHigh: Highes price in the day
dayOpen: Open price in the day
dayClose: Close price in the day
dayChange: Changes price in a day

Examples

```
## Not run:
# Download daily data from last one month

marketInfo("btc","rls")

## End(Not run)
```

`mcPrt` *Matrix Conversion in Portfolio*

Description

mcPrt function compound the vector of assets in matrix form to using the multivariate data analysis. The assets must be included the items of `Open`, `Close`, `High`, `Low` and `Volumn`. The function also calculate the rate of return and Sharpe ratio for portfolio selection.

Usage

```
mcPrt(asset, sub = "::", pstvRtn = FALSE, pr = "daily", Rf = 0.0)
```

Arguments

asset	a character vector of symbols.
sub	The character string in the form of "start date :: end date". The date format is "%Y-%M-%D". Since, the data is time series format, the rules of **xts** satisfied.
pstvRtn	logical flag to determine, if the symbols with mean of return (see, details) should be positive/negative. Missing value is taken as false. By setting pstvRtn=TRUE, only a check the symbols that the mean of return is not negative and symbols with negative returns will be vanished from the list of calculation.
pr	a character specifying the desired period time of return. Default value is daily. The values get "daily", "weekly" and "monthly".
Rf	an integer value to give risk free.

Details

The values of function contain Sharpe ratio and rate of return. Sharpe ratio is defined as:

$$SR = \frac{\bar{R} - R_f}{\sigma_R}$$

where $\bar{R}$ is the mean asset return, R_f is the risk free rate of return and σ_R is the standard deviation (volatility) of the asset.

Let X_t is the closing price of the asset at time t and its value after a period of length Δt, say $X_{t+\Delta t}$. The return over that period is defined as:

$$R_t = \frac{X_{t+\Delta t}}{X_t} - 1$$

Note that the return equation is equivalent to log return asset when the ratio value of the asset is close to 1 (see, Carmona 2003). To check the goodness of fit test on return value, maximum and minimum of prices, suggested to use the package of **gnFit**.

Value

The values of function involve close, return, max and min as a matrix and xts format. The columns are a value of assets which is ordered by date. The mean return, volatility and Sharpe ratio are assigned in the out. The plot of return vs volatility and also the graph of Sharpe ratio are outputs of function.

References

Carmona (2003, ISBN:0387202862)

Examples

```
## Not run:
fpath<-system.file("extdata", "sample.dat", package="TSEtools")
getTSE(fpath)
rtn<-mcPrt(asset0, sub="2016::", pstvRtn=TRUE, pr="weekly")$return

## End(Not run)
```

`orderADD` *Recorde an order's trade*

Description

This function is used to submit buy and sell orders on the Nobitex website

Usage

```
orderADD(typ, src, dst, amnt, prc, stp = NULL, stpL = NULL, exe = "limit", token)
```

Arguments

`typ`	`"buy"` is used for placing buy orders, while `"sell"` is used for placing sell orders
`src`	Cryptocurrency sources are defined by lowercase names in the function (e.g., "btc", "eth", "bnb", "paxg").
`dst`	For the Rials or USD system, the destination currency must be set to either `"rls"` or `"usdt"`.
`amnt`	Amount of cryptocurrency to trade
`prc`	Price of cryptocurrency to trade. The trade price ("prc") depends on the exe's option.
`stp`	If using stop-loss or OCO, define the stop price here. Default is NULL. When the market price reaches "stp", "prc" is activated and recorded in the order list. For a "buy" position, "stp" must be greater than "prc"; for a "sell"" position, "stp" must be less than `prc`.
`stpL`	If using "oco" then stop limit price must be defined. Defualt is NULL. While "stp" is active, "stpl" is recorded as the trade price; otherwise, "prc" is used.
`exe`	Three common ways to configure a trade: regular trading, stop-loss trading, or One-Cancels-the-Other (OCO) trading. Regular trading: The `limit` will be recorded in `prc` as the trade price. The `market` will be recorded in `prc` as the nearest trade price to the latest bid or ask, depending on the buyer or seller. Stop loss trading: `stop_limit` and `stop_market` are two trade types that depend on price of `stp`. OCO: Using `oco` requires defining two price: `stp` and `stpL`.
`token`	The token is string value obtain from user ID settings in the NOBITEX. If you are not registered on the website, please use the provided link.

Value

Returns a list: "ok" if the order is recorded successfully on the website; otherwise, "failed".

Examples

```
## Not run:
tkn <- "CONTACT WITH AUTHOR"
x <- orderADD(typ = "sell", src = "usdt", dst = "rls", amnt = 5, prc = 1600000,
              + exe = "limit", token = tkn)

x <- orderADD(typ = "sell", src = "usdt", dst = "rls", amnt = 5, prc = 1400000,
              + stp = 1300000, exe = "stop_limit", token = tkn)

x <- orderADD(typ = "sell", src = "usdt", dst = "rls", amnt = 5, prc = 1400000,
              + stp = 1300000, stpL = 1350000, exe = "oco", token = tkn)

## End(Not run)
```

---

orderBook *Get the bid and ask prices with the volume*

Description

Get the bid and ask prices with the volumes in orders

Usage

```
orderBook(src)
```

Arguments

src	List of the some crypto curencies which available in the function. BTCIRT, ETHIRT, LTCIRT, USDTIRT, XRPIRT, BCHIRT, BNBIRT, EOSIRT, XLMIRT, ETCIRT, TRXIRT, DOGEIRT, UNIIRT, DAIIRT, LINKIRT, DOTIRT, AAVEIRT, ADAIRT, SHIBIRT, FTMIRT, MATICIRT, AXSIRT, MANAIRT, SANDIRT, AVAXIRT, MKRIRT, GMTIRT, USDCIRT, BTCUSDT, ETHUSDT, LTCUSDT, XRPUSDT, BCHUSDT, BNBUSDT, EOSUSDT, XLMUSDT, ETCUSDT, TRXUSDT, PMNUSDT, DOGEUSDT, UNIUSDT, DAIUSDT, LINKUSDT, DOTUSDT, AAVEUSDT, ADAUSDT, SHIBUSDT, FTMUSDT, MATICUSDT, AXSUSDT, MANAUSDT, SANDUSDT, AVAXUSDT, MKRUSDT, GMTUSDT, USDCUSDT, CHZIRT, GRTIRT, CRVIRT, BANDUSDT, COMPUSDT, EGLDIRT, HBARUSDT, GALIRT, HBARIRT, WBTCUSDT, IMXIRT, WBTCIRT, ONEIRT, GLMUSDT, ENSIRT, 1M_BTTIRT, SUSHIIRT, LDOIRT, ATOMUSDT, ZROIRT, STORJIRT, ANTIRT, AEVOUSDT, 100K_FLOKIIRT, RSRUSDT, API3USDT, GLMIRT, XMRIRT, ENSUSDT, OMIRT, RDNTIRT, MAGICUSDT, TIRT, ATOMIRT, NOTIRT, CVXIRT, XTZIRT, FILIRT, UMAIRT, 1B_BABYDOGEIRT, BANDIRT, SSVIRT, DAOIRT, BLURIRT, ONEUSDT, EGALAUSDT, GMXIRT, XTZUSDT, FLOWUSDT, GALUSDT, WIRT, CVCUSDT, NMRUSDT, SKLIRT,

SNTIRT, BATUSDT, TRBUSDT, NMRIRT, RDNTUSDT, API3IRT, CVCIRT, WLDIRT, YFIUSDT, SOLIRT, TUSDT, QNTUSDT, IMXUSDT, AEVOIRT, GMXUSDT, ETHFIUSDT, QNTIRT, GRTUSDT, WLDUSDT, FETIRT, AGIXIRT, NOTUSDT, LPTIRT, SLPIRT, MEMEUSDT, SOLUSDT, BALUSDT, DAOUSDT, COMPIRT, MEMEIRT, TONUSDT, BATIRT, SNXIRT, TRBIRT, 1INCHUSDT, OMUSDT, RSRIRT, RNDRIRT, SLPUSDT, SSVUSDT, RNDRUSDT, AGLDIRT, NEARUSDT, WOOUSDT, YFIIRT, MDTIRT, CRVUSDT, MDTUSDT, EGLDUSDT, LRCIRT, LPTUSDT, BICOUSDT, 1M_PEPEIRT, BICOIRT, MAGICIRT, ETHFIIRT, ANTUSDT, 1INCHIRT, APEUSDT, 1M_NFTIRT, ARBIRT, LRCUSDT, WUSDT, BLURUSDT, CELRUSDT, DYDXIRT, CVXUSDT, BALIRT, TONIRT, 100K_FLOKIUSDT, JSTUSDT, ZROUSDT, ARBUSDT, APTIRT, 1M_NFTUSDT, CELRIRT, UMAUSDT, SKLUSDT, ZRXUSDT, AGLDUSDT, ALGOIRT, NEARIRT, APTUSDT, ZRXIRT, SUSHIUSDT, FETUSDT, ALGOUSDT, 1M_PEPEUSDT, MASKIRT, EGALAIRT, FLOWIRT, 1B_BABYDOGEUSDT, MASKUSDT, 1M_BTTUSDT, STORJUSDT, XMRUSDT, OMGIRT, SNTUSDT, APEIRT, FILUSDT, ENJUSDT, OMGUSDT, WOOIRT, CHZUSDT, ENJIRT, DYDXUSDT, AGIXUSDT, JSTIRT, LDOUSDT, SNXUSDT

Value

The values return as a data frame format. The names of the dimensions are
`prc.ask`: last 24 prices of the asks
`vol.ask`: a list of last 24 volumes corresponding to the ask's price.
`prc.bid`: last 24 prices of the bids
`vol.bid`: a list of last 24 volumes corresponding to the bid's price.
`lst.prc`: last trade's price

Examples

```
## Not run:
# Download bid and ask market price

orderBook("BTCUSDT")

## End(Not run)
```

`orderList` *Order List*

Description

The order list shows all registered orders waiting to be traded.

Usage

```
orderList(token)
```

Arguments

token	The token is string value obtain from user ID settings in the NOBITEX. If you are not registered on the website, please use the provided link.

Value

The value returned:

type	Order type: "sell" or "buy".
execution	Order execution type can be "Limit", "Market", "StopMarket", or "StopLimit"
tradeType	Order trade type is "Spot" or "Margin"
srcCurrency	Source currency name ("Bitcoin", "Tether", etc).
dstCurrency	Destination currency name (USDT or RLS).
price	Order price
amount	Order amount
totalOrderPrice	Total order price.
matchedAmount	Matched amount.
unmatchedAmount	Remaining amount for trade.
id	Id number of order.
status	Order status: New, "Active", "Inactive", "Done", or "Canceled".
fee	Order fee.
created_at	Created time of order.
market	Market name of order.

Examples

```
## Not run:
tkn <- "CONTACT WITH AUTHOR"
x <- orderList(token = tkn)

## End(Not run)
```

---

prtf *Design the Portfolio of assets*

Description

Compute the efficient frontier function for some selected risk functionals in a portfolio optimization setting.

Usage

```
prtf (x, Rf = 0.0, sh = FALSE ,eRtn = NULL)
```

Arguments

x	a numeric matrix of random returns per unit of price within some holding period.
sh	a logical indicating whether shortsales on the risky securities are allowed. Default is FALSE.
Rf	the return of the risk free, i.e. has variance 0.
eRtn	a value of expected returen of portofilo. The mean of whole data defualt.

Details

Let $\xi_1, \ldots, \xi_n$ be random asset returns and $w_1, \ldots, w_n$ the portfolio weights. The expected returns are $r_m = E\xi_m, m = 1, \ldots, n$. In addition to these risky investments, there is a risk-free asset (a bond or bank account) available, which has return r_0. Denoting the weights of w_0 for the risk-free asset. The return of portfolio given by

$$R_p = w^t r$$

where, $r = (r_1, \ldots, r_n)^t$.

Risk is measure by a deviation functional Σ. It is a variance-covariance of asset returns. The risk-free component w_0 ignore in the objective. So, the standard deviation of portfolio is given by $\sigma_p = w^t \Sigma w$.

To obtain the optimum value of $w_i, i = 1, \ldots, n$, we solve the following model:

$$\min w^t \Sigma w \;\; s.t: \;\; w^t r > \mu \;\; and \;\; \sum w_i = 1.$$

where, μ is a constant value. Note that, the portfolio weights may be negative (selling short is allowed).

Value

The minimum weights show with MIN which is the portfolio with the minimum volatility. Market portfolio is given by MP where, the risk free weight w_0 is zero. MP is the tangency point between the market line and efficient frountier curve. A list containing the following components:

prt	list the name of assests in the portfolio
obs.p	return and volatiliy of overall portfolio
vol	volatility of portfolio
rtn	return of portfolio
w	weigths of portfolio

References

Pflug and Romisch (2007, ISBN: 9789812707406)

See Also

portfolio.optimization, portfolio.optim

Examples

```
## Not run:
x <- rnorm(500,0.05,0.02)
y <- rnorm(500,0.01,0.03)
z<-cbind(x, y)
colnames(z) <- c("prt1","prt2")

prtf (z, sh = FALSE)

## End(Not run)
```

`tradeInfo` *The information on the last trades*

Description

Obtain information about the most recent trades conducted in the cryptocurrency market. This data can provide insights into the market trends and help investors make informed decisions.

Usage

```
tradeInfo(src)
```

Arguments

`src`	List of the some crypto curencies which available in the function. BTCIRT, ETHIRT, LTCIRT, USDTIRT, XRPIRT, BCHIRT, BNBIRT, EOSIRT, XLMIRT, ETCIRT, TRXIRT, DOGEIRT, UNIIRT, DAIIRT, LINKIRT, DOTIRT, AAVEIRT, ADAIRT, SHIBIRT, FTMIRT, MATICIRT, AXSIRT, MANAIRT, SANDIRT, AVAXIRT, MKRIRT, GMTIRT, USDCIRT, BTCUSDT, ETHUSDT, LTCUSDT, XRPUSDT, BCHUSDT, BNBUSDT, EOSUSDT, XLMUSDT, ETCUSDT, TRXUSDT, PMNUSDT, DOGEUSDT, UNIUSDT, DAIUSDT, LINKUSDT, DOTUSDT, AAVEUSDT, ADAUSDT, SHIBUSDT, FTMUSDT, MATICUSDT, AXSUSDT, MANAUSDT, SANDUSDT, AVAXUSDT, MKRUSDT, GMTUSDT, USDCUSDT

Value

The values return as a data frame format. The names of the dimensions are
price: List the prices at which the trade was done. volume: List the volume corresponding the price at which the trade was done. type: Type of trade: 'sell' means selecting the price from the bid, and 'buy' refers to the ask price from which you choose.

Examples

```
# Download daily data from last one month
## Not run:
tradeInfo("BTCUSDT")

## End(Not run)
```

---

updateStatus *Cancel an order.*

Description

Cancel the specified order using the order ID.

Usage

```
updateStatus(order, st = "canceled", token)
```

Arguments

order	The order ID can be obtained by using the order list function.
st	By default, the specified order is canceled.
token	The token is string value obtain from user ID settings in the NOBITEX. If you are not registered on the website, please use the provided link.

Examples

```
## Not run:
tkn <- "CONTACT WITH AUTHOR"
x <- orderList(token = tkn)

# Ex. consider the first order recorded,
ID <- x[1]$id

# SELECT SPECIFY ID order from the list
updateStatus( ID, st = "canceled", token = tkn)

## End(Not run)
```

walletLST *List of Wallet*

Description

The wallet list includes the balance and blocked value of each asset.

Usage

```
walletLST(token)
```

Arguments

token	The token is string value obtain from user ID settings in the NOBITEX. If you are not registered on the website, please use the provided link.

Value

id	Cryptocurrency name and ID.
balance	The balance value is available for orders.
blocked	The blocked value represents funds locked in orders.

Examples

```
## Not run:
tkn <- "CONTACT WITH AUTHOR"
x <- walletLST(token = tkn)

## End(Not run)
```

Index