

Package ‘FinCal’

July 21, 2025

Title Time Value of Money, Time Series Analysis and Computational Finance

Description

Package for time value of money calculation, time series analysis and computational finance.

Version 0.6.3

Date 2016-07-27

Author Felix Yanhui Fan <nolanfyh@gmail.com>

Imports ggplot2, reshape2, RCurl

Maintainer Felix Yanhui Fan <nolanfyh@gmail.com>

License GPL (>= 2)

URL <http://felixfan.github.io/FinCal/>

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-07-29 06:40:54

Contents

bdf	3
bdf2mm	3
candlestickChart	4
cash.ratio	5
coefficient.variation	5
cogs	6
current.ratio	7
ddb	7
debt.ratio	8
diluted.EPS	9
discount.rate	10
ear	11
ear.continuous	11

ear2bey	12
ear2hpr	12
EIR	13
EPS	14
financial.leverage	15
fv	15
fv.annuity	16
fv.simple	17
fv.uneven	17
geometric.mean	18
get.ohlc.google	18
get.ohlc.yahoo	19
get.ohlcs.google	20
get.ohlcs.yahoo	20
gpm	21
harmonic.mean	22
hpr	22
hpr2bey	23
hpr2ear	23
hpr2mmy	24
irr	24
irr2	25
iss	26
lineChart	26
lineChartMult	27
lt.d2e	28
mmy2hpr	28
n.period	29
npm	30
npv	30
pmt	31
pv	32
pv.annuity	33
pv.perpetuity	33
pv.simple	34
pv.uneven	35
quick.ratio	35
r.continuous	36
r.norminal	36
r.perpetuity	37
sampling.error	38
SFRatio	38
Sharpe.ratio	39
slde	39
total.d2e	40
twrr	41
volumeChart	41
was	42

<i>bdy</i>	3
wpr	43
Index	44

bdy	<i>Computing bank discount yield (BDY) for a T-bill</i>
------------	---

Description

Computing bank discount yield (BDY) for a T-bill

Usage

`bdy(d, f, t)`

Arguments

d	the dollar discount, which is equal to the difference between the face value of the bill and the purchase price
f	the face value (par value) of the bill
t	number of days remaining until maturity

See Also

[bdy2mmy](#)

Examples

`bdy(d=1500, f=100000, t=120)`

bdy2mmy	<i>Computing money market yield (MMY) for a T-bill</i>
----------------	--

Description

Computing money market yield (MMY) for a T-bill

Usage

`bdy2mmy(bdy, t)`

Arguments

bdy	bank discount yield
t	number of days remaining until maturity

See Also[bdy](#)**Examples**

```
bdy2mmy(bdy=0.045, t=120)
```

candlestickChart

Technical analysts - Candlestick chart: show prices for each period as a continuous line. The box is clear if the closing price is higher than the opening price, or filled red if the closing is lower than the opening price.

Description

Technical analysts - Candlestick chart: show prices for each period as a continuous line. The box is clear if the closing price is higher than the opening price, or filled red if the closing is lower than the opening price.

Usage

```
candlestickChart(ohlc, start = NULL, end = NULL, main = "", ...)
```

Arguments

ohlc	output from <code>get.ohlc.yahoo</code> or <code>get.ohlc.google</code>
start	start date to plot, if not specified, all date in ohlc will be included
end	end date to plot
main	an overall title for the plot
...	Arguments to be passed to <code>ggplot</code>

See Also[get.ohlc.yahoo](#)[get.ohlc.google](#)**Examples**

```
# google <- get.ohlc.yahoo("GOOG", start="2013-07-01", end="2013-08-01"); candlestickChart(google)
# apple <- get.ohlc.google("AAPL", start="2013-07-01", end="2013-08-01"); candlestickChart(apple)
```

cash.ratio	<i>cash ratio – Liquidity ratios measure the firm's ability to satisfy its short-term obligations as they come due.</i>
------------	---

Description

cash ratio – Liquidity ratios measure the firm's ability to satisfy its short-term obligations as they come due.

Usage

```
cash.ratio(cash, ms, cl)
```

Arguments

cash	cash
ms	marketable securities
cl	current liabilities

See Also

[current.ratio](#)
[quick.ratio](#)

Examples

```
cash.ratio(cash=3000,ms=2000,cl=2000)
```

coefficient.variation	<i>Computing Coefficient of variation</i>
-----------------------	---

Description

Computing Coefficient of variation

Usage

```
coefficient.variation(sd, avg)
```

Arguments

sd	standard deviation
avg	average value

See Also

[Sharpe.ratio](#)

Examples

```
coefficient.variation(sd=0.15,avg=0.39)
```

cogs	<i>Cost of goods sold and ending inventory under three methods (FIFO,LIFO,Weighted average)</i>
------	---

Description

Cost of goods sold and ending inventory under three methods (FIFO,LIFO,Weighted average)

Usage

```
cogs(uinv, pinv, units, price, sinv, method = "FIFO")
```

Arguments

uinv	units of beginning inventory
pinv	price of beginning inventory
units	nx1 vector of inventory units. inventory purchased ordered by time (from first to last)
price	nx1 vector of inventory price. same order as units
sinv	units of sold inventory
method	inventory methods: FIFO (first in first out, permitted under both US and IFRS), LIFO (late in first out, US only), WAC (weighted average cost,US and IFRS)

Examples

```
cogs(uinv=2,pinv=2,units=c(3,5),price=c(3,5),sinv=7,method="FIFO")
cogs(uinv=2,pinv=2,units=c(3,5),price=c(3,5),sinv=7,method="LIFO")
cogs(uinv=2,pinv=2,units=c(3,5),price=c(3,5),sinv=7,method="WAC")
```

current.ratio	<i>current ratio – Liquidity ratios measure the firm's ability to satisfy its short-term obligations as they come due.</i>
---------------	--

Description

current ratio – Liquidity ratios measure the firm's ability to satisfy its short-term obligations as they come due.

Usage

```
current.ratio(ca, cl)
```

Arguments

ca	current assets
cl	current liabilities

See Also

[cash.ratio](#)
[quick.ratio](#)

Examples

```
current.ratio(ca=8000,cl=2000)
```

ddb	<i>Depreciation Expense Recognition – double-declining balance (DDB), the most common declining balance method, which applies two times the straight-line rate to the declining balance.</i>
-----	--

Description

Depreciation Expense Recognition – double-declining balance (DDB), the most common declining balance method, which applies two times the straight-line rate to the declining balance.

Usage

```
ddb(cost, rv, t)
```

Arguments

<code>cost</code>	cost of long-lived assets
<code>rv</code>	residual value of the long-lived assets at the end of its useful life. DDB does not explicitly use the asset's residual value in the calculations, but depreciation ends once the estimated residual value has been reached. If the asset is expected to have no residual value, the DB method will never fully depreciate it, so the DB method is typically changed to straight-line at some point in the asset's life.
<code>t</code>	length of the useful life

See Also[s1de](#)**Examples**

```
ddb(cost=1200,rv=200,t=5)
```

debt.ratio

debt ratio – Solvency ratios measure the firm's ability to satisfy its long-term obligations.

Description

debt ratio – Solvency ratios measure the firm's ability to satisfy its long-term obligations.

Usage

```
debt.ratio(td, ta)
```

Arguments

<code>td</code>	total debt
<code>ta</code>	total assets

See Also[total.d2e](#)[lt.d2e](#)[financial.leverage](#)**Examples**

```
debt.ratio(td=6000,ta=20000)
```

diluted.EPS	<i>diluted Earnings Per Share</i>
-------------	-----------------------------------

Description

diluted Earnings Per Share

Usage

```
diluted.EPS(ni, pd, cpd = 0, cdi = 0, tax = 0, w, cps = 0, cds = 0,  
iss = 0)
```

Arguments

ni	net income
pd	preferred dividends
cpd	dividends on convertible preferred stock
cdi	interest on convertible debt
tax	tax rate
w	weighted average number of common shares outstanding
cps	shares from conversion of convertible preferred stock
cds	shares from conversion of convertible debt
iss	shares issuable from stock options

See Also

[EPS](#)
[iss](#)
[was](#)

Examples

```
diluted.EPS(ni=115600,pd=10000,cdi=42000,tax=0.4,w=200000,cds=60000)  
diluted.EPS(ni=115600,pd=10000,cpd=10000,w=200000,cps=40000)  
diluted.EPS(ni=115600,pd=10000,w=200000,iss=2500)  
diluted.EPS(ni=115600,pd=10000,cpd=10000,cdi=42000,tax=0.4,w=200000,cps=40000,cds=60000,iss=2500)
```

discount.rate *Computing the rate of return for each period*

Description

Computing the rate of return for each period

Usage

```
discount.rate(n, pv, fv, pmt, type = 0)
```

Arguments

n	number of periods
pv	present value
fv	future value
pmt	payment per period
type	payments occur at the end of each period (type=0); payments occur at the beginning of each period (type=1)

See Also

[fv.simple](#)

[fv.annuity](#)

[fv](#)

[pv](#)

[pmt](#)

[n.period](#)

Examples

```
discount.rate(n=5,pv=0,fv=600,pmt=-100,type=0)
```

ear

Convert stated annual rate to the effective annual rate

Description

Convert stated annual rate to the effective annual rate

Usage

`ear(r, m)`

Arguments

<code>r</code>	stated annual rate
<code>m</code>	number of compounding periods per year

See Also

[ear.continuous](#)
[hpr2ear](#)
[ear2bey](#)
[ear2hpr](#)

Examples

```
ear(r=0.12,m=12)  
ear(0.04,365)
```

`ear.continuous`

Convert stated annual rate to the effective annual rate with continuous compounding

Description

Convert stated annual rate to the effective annual rate with continuous compounding

Usage

`ear.continuous(r)`

Arguments

<code>r</code>	stated annual rate
----------------	--------------------

See Also

[ear](#)
[r.nominal](#)

Examples

```
ear.continuous(r=0.1)
ear.continuous(0.03)
```

ear2bey

*bond-equivalent yield (BEY), 2 x the semiannual discount rate***Description**

bond-equivalent yield (BEY), 2 x the semiannual discount rate

Usage

```
ear2bey(ear)
```

Arguments

ear	effective annual rate
-----	-----------------------

See Also

[ear](#)

Examples

```
ear2bey(ear=0.08)
```

ear2hpr

*Computing HPR, the holding period return***Description**

Computing HPR, the holding period return

Usage

```
ear2hpr(ear, t)
```

Arguments

ear	effective annual rate
t	number of days remaining until maturity

See Also

[hpr2ear](#)
[ear](#)
[hpr](#)

Examples

```
ear2hpr(ear=0.05039, t=150)
```

EIR

*Equivalent/proportional Interest Rates***Description**

An interest rate to be applied n times p.a. can be converted to an equivalent rate to be applied p times p.a.

Usage

```
EIR(r, n = 1, p = 12, type = c("e", "p"))
```

Arguments

r	interest rate to be applied n times per year (r is annual rate!)
n	times that the interest rate r were compounded per year
p	times that the equivalent rate were compounded per year
type	equivalent interest rates ('e',default) or proportional interest rates ('p')

Examples

```
# monthly interest rat equivalent to 5% compounded per year
EIR(r=0.05,n=1,p=12)
```

```
# monthly interest rat equivalent to 5% compounded per half year
EIR(r=0.05,n=2,p=12)
```

```
# monthly interest rat equivalent to 5% compounded per quarter
EIR(r=0.05,n=4,p=12)
```

```
# annual interest rate equivalent to 5% compounded per month
EIR(r=0.05,n=12,p=1)
```

```
# this is equivalent to
ear(r=0.05,m=12)

# quarter interest rate equivalent to 5% compounded per year
EIR(r=0.05,n=1,p=4)

# quarter interest rate equivalent to 5% compounded per month
EIR(r=0.05,n=12,p=4)

# monthly proportional interest rate which is equivalent to a simple annual interest
EIR(r=0.05,p=12,type='p')
```

EPS*Basic Earnings Per Share***Description**

Basic Earnings Per Share

Usage

```
EPS(ni, pd, w)
```

Arguments

ni	net income
pd	preferred dividends
w	weighted average number of common shares outstanding

See Also

[diluted.EPS](#)

[was](#)

Examples

```
EPS(ni=10000,pd=1000,w=11000)
```

<code>financial.leverage</code>	<i>financial leverage – Solvency ratios measure the firm's ability to satisfy its long-term obligations.</i>
---------------------------------	--

Description

`financial leverage` – Solvency ratios measure the firm's ability to satisfy its long-term obligations.

Usage

```
financial.leverage(te, ta)
```

Arguments

<code>te</code>	total equity
<code>ta</code>	total assets

See Also

[total.d2e](#)
[lt.d2e](#)
[debt.ratio](#)

Examples

```
financial.leverage(te=16000, ta=20000)
```

<code>fv</code>	<i>Estimate future value (fv)</i>
-----------------	-----------------------------------

Description

Estimate future value (fv)

Usage

```
fv(r, n, pv = 0, pmt = 0, type = 0)
```

Arguments

<code>r</code>	discount rate, or the interest rate at which the amount will be compounded each period
<code>n</code>	number of periods
<code>pv</code>	present value
<code>pmt</code>	payment per period
<code>type</code>	payments occur at the end of each period (type=0); payments occur at the beginning of each period (type=1)

See Also

[fv.simple](#)
[fv.annuity](#)
[pv](#)
[pmt](#)
[n.period](#)
[discount.rate](#)

Examples

```
fv(0.07,10,1000,10)
```

fv.annuity

Estimate future value of an annuity

Description

Estimate future value of an annuity

Usage

```
fv.annuity(r, n, pmt, type = 0)
```

Arguments

r	discount rate, or the interest rate at which the amount will be compounded each period
n	number of periods
pmt	payment per period
type	payments occur at the end of each period (type=0); payments occur at the beginning of each period (type=1)

See Also

[fv](#)

Examples

```
fv.annuity(0.03,12,-1000)
```

```
fv.annuity(r=0.03,n=12,pmt=-1000,type=1)
```

fv.simple*Estimate future value (fv) of a single sum*

Description

Estimate future value (fv) of a single sum

Usage

```
fv.simple(r, n, pv)
```

Arguments

r	discount rate, or the interest rate at which the amount will be compounded each period
n	number of periods
pv	present value

See Also[fv](#)**Examples**

```
fv.simple(0.08,10,-300)
```

```
fv.simple(r=0.04,n=20,pv=-50000)
```

fv.uneven*Computing the future value of an uneven cash flow series*

Description

Computing the future value of an uneven cash flow series

Usage

```
fv.uneven(r, cf)
```

Arguments

r	stated annual rate
cf	uneven cash flow

See Also`fv.simple`**Examples**

```
fv.uneven(r=0.1, cf=c(-1000, -500, 0, 4000, 3500, 2000))
```

`geometric.mean`*Geometric mean return***Description**

Geometric mean return

Usage`geometric.mean(r)`**Arguments**

<code>r</code>	returns over multiple periods
----------------	-------------------------------

Examples

```
geometric.mean(r=c(-0.0934, 0.2345, 0.0892))
```

`get.ohlc.google`*Download stock prices from Google Finance (open, high, low, close, volume)***Description**

Download stock prices from Google Finance (open, high, low, close, volume)

Usage

```
get.ohlc.google(symbol, start = "2013-01-01", end = "today")
```

Arguments

<code>symbol</code>	symbol of stock, e.g. AAPL, GOOG, SPX
<code>start</code>	start date, e.g., 2013-07-31
<code>end</code>	end date, e.g., 2013-08-06

See Also

[get.ohlc.yahoo](#)
[get.ohlcs.google](#)

Examples

```
# get.ohlc.google(symbol="AAPL")
# get.ohlc.google(symbol="AAPL",start="2013-08-01")
# get.ohlc.google(symbol="AAPL",start="2013-07-01",end="2013-08-01")
```

get.ohlc.yahoo

Download stock prices from Yahoo Finance (open, high, low, close, volume, adjusted)

Description

Download stock prices from Yahoo Finance (open, high, low, close, volume, adjusted)

Usage

```
get.ohlc.yahoo(symbol, start = "firstDay", end = "today", freq = "d")
```

Arguments

symbol	symbol of stock, e.g. AAPL, GOOG, SPX
start	start date, e.g., 2013-07-31
end	end date, e.g., 2013-08-06
freq	time interval, e.g., d:daily, w:weekly, m:monthly

See Also

[get.ohlcs.yahoo](#)
[get.ohlc.google](#)

Examples

```
# get.ohlc.yahoo(symbol="AAPL")
# get.ohlc.yahoo(symbol="AAPL",start="2013-08-01",freq="d")
# get.ohlc.yahoo(symbol="AAPL",start="2013-07-01",end="2013-08-01",freq="w")
```

`get.ohlcs.google`

Batch download stock prices from Google Finance (open, high, low, close, volume)

Description

Batch download stock prices from Google Finance (open, high, low, close, volume)

Usage

```
get.ohlcs.google(symbols, start = "2013-01-01", end = "today")
```

Arguments

<code>symbols</code>	symbols of stock, e.g. AAPL, GOOG, SPX
<code>start</code>	start date, e.g., 2013-07-31
<code>end</code>	end date, e.g., 2013-08-06

See Also

[get.ohlc.google](#)
[get.ohlcs.yahoo](#)

Examples

```
# get.ohlcs.google(symbols=c("AAPL","GOOG","SPY"))
# get.ohlcs.google(symbols=c("AAPL","GOOG","SPY"),start="2013-01-01")
# get.ohlcs.google(symbols=c("AAPL","GOOG","SPY"),start="2013-01-01",end="2013-07-31")
```

`get.ohlcs.yahoo`

Batch download stock prices from Yahoo Finance (open, high, low, close, volume, adjusted)

Description

Batch download stock prices from Yahoo Finance (open, high, low, close, volume, adjusted)

Usage

```
get.ohlcs.yahoo(symbols, start = "firstDay", end = "today", freq = "d")
```

Arguments

symbols	symbols of stock, e.g. AAPL, GOOG, SPX
start	start date, e.g., 2013-07-31
end	end date, e.g., 2013-08-06
freq	time interval, e.g., d:daily, w:weekly, m:monthly

See Also

[get.ohlc.yahoo](#)
[get.ohlcs.google](#)

Examples

```
# get.ohlcs.yahoo(symbols=c("AAPL", "GOOG", "SPY"), freq="d")
# get.ohlcs.yahoo(symbols=c("AAPL", "GOOG", "SPY"), start="2013-01-01", freq="m")
```

gpm*gross profit margin – Evaluate a company's financial performance*

Description

gross profit margin – Evaluate a company's financial performance

Usage

```
gpm(gp, rv)
```

Arguments

gp	gross profit, equal to revenue minus cost of goods sold (cogs)
rv	revenue (sales)

See Also

[npm](#)

Examples

```
gpm(gp=1000, rv=20000)
```

`harmonic.mean` *harmonic mean, average price*

Description

harmonic mean, average price

Usage

`harmonic.mean(p)`

Arguments

`p` price over multiple periods

Examples

`harmonic.mean(p=c(8,9,10))`

`hpr` *Computing HPR, the holding period return*

Description

Computing HPR, the holding period return

Usage

`hpr(ev, bv, cfr = 0)`

Arguments

<code>ev</code>	ending value
<code>bv</code>	beginning value
<code>cfr</code>	cash flow received

See Also

[twrr](#)
[hpr2ear](#)
[hpr2mmy](#)

Examples

`hpr(ev=33,bv=30,cfr=0.5)`

hpr2bey	<i>bond-equivalent yield (BEY), 2 x the semiannual discount rate</i>
---------	--

Description

bond-equivalent yield (BEY), 2 x the semiannual discount rate

Usage

`hpr2bey(hpr, t)`

Arguments

hpr	holding period return
t	number of month remaining until maturity

See Also

[hpr](#)

Examples

`hpr2bey(hpr=0.02, t=3)`

hpr2ear	<i>Convert holding period return to the effective annual rate</i>
---------	---

Description

Convert holding period return to the effective annual rate

Usage

`hpr2ear(hpr, t)`

Arguments

hpr	holding period return
t	number of days remaining until maturity

See Also

[ear](#)
[hpr](#)
[ear2hpr](#)

Examples

```
hpr2ear(hpr=0.015228, t=120)
```

hpr2mmmy

Computing money market yield (MMY) for a T-bill

Description

Computing money market yield (MMY) for a T-bill

Usage

```
hpr2mmmy(hpr, t)
```

Arguments

hpr	holding period return
t	number of days remaining until maturity

See Also

[hpr](#)

[mmmy2hpr](#)

Examples

```
hpr2mmmy(hpr=0.01523, t=120)
```

irr

Computing IRR, the internal rate of return

Description

Computing IRR, the internal rate of return

Usage

```
irr(cf)
```

Arguments

cf	cash flow, the first cash flow is the initial outlay
----	--

See Also[pv.uneven](#)[npv](#)**Examples**

```
# irr(cf=c(-5, 1.6, 2.4, 2.8))
```

irr2*Computing IRR, the internal rate of return*

Description

This function is the same as `irr` but can calculate negative value. This function may take a very long time. You can use larger cutoff and larger step to get a less precision `irr` first. Then based on the result, change from and to, to narrow down the interval, and use a smaller step to get a more precision `irr`.

Usage

```
irr2(cf, cutoff = 0.1, from = -1, to = 10, step = 1e-06)
```

Arguments

<code>cf</code>	cash flow, the first cash flow is the initial outlay
<code>cutoff</code>	threshold to take <code>npv</code> as zero
<code>from</code>	smallest <code>irr</code> to try
<code>to</code>	largest <code>irr</code> to try
<code>step</code>	increment of the <code>irr</code>

See Also[irr](#)**Examples**

```
# irr2(cf=c(-5, 1.6, 2.4, 2.8))
# irr2(cf=c(-200, 50, 60, -70, 30, 20))
```

iss	<i>calculate the net increase in common shares from the potential exercise of stock options or warrants</i>
------------	---

Description

calculate the net increase in common shares from the potential exercise of stock options or warrants

Usage

```
iss(amp, ep, n)
```

Arguments

amp	average market price over the year
ep	exercise price of the options or warrants
n	number of common shares that the options and warrants can be converted into

See Also

[diluted.EPS](#)

Examples

```
iss(amp=20, ep=15, n=10000)
```

lineChart	<i>Technical analysts - Line charts: show prices for each period as a continuous line</i>
------------------	---

Description

Technical analysts - Line charts: show prices for each period as a continuous line

Usage

```
lineChart(ohlc, y = "close", main = "", ...)
```

Arguments

ohlc	output from get.ohlc.yahoo or get.ohlc.google
y	y coordinates: close, open, high, low or adjusted (yahoo data only)
main	an overall title for the plot
...	Arguments to be passed to ggplot

See Also

[get.ohlc.yahoo](#)
[get.ohlc.google](#)

Examples

```
# google <- get.ohlc.yahoo("GOOG"); lineChart(google)
# apple <- get.ohlc.google("AAPL"); lineChart(apple)
```

lineChartMult

Technical analysts - Line charts: show prices for each period as a continuous line for multiple stocks

Description

Technical analysts - Line charts: show prices for each period as a continuous line for multiple stocks

Usage

```
lineChartMult(ohlcs, y = "close", main = "", ...)
```

Arguments

ohlcs	output from <code>get.ohlc.yahoo.mult</code> or <code>get.ohlc.google.mult</code>
y	y coordinates: close, open, high, low or adjusted (yahoo data only)
main	an overall title for the plot
...	Arguments to be passed to <code>ggplot</code>

See Also

[get.ohlcs.yahoo](#)
[get.ohlcs.google](#)
[lineChart](#)

Examples

```
# googapple <- get.ohlcs.yahoo(c("GOOG", "AAPL"), start="2013-01-01");
# lineChartMult(googapple)
# googapple <- get.ohlcs.google(c("GOOG", "AAPL"), start="2013-01-01");
# lineChartMult(googapple)
```

lt.d2e

long-term debt-to-equity – Solvency ratios measure the firm's ability to satisfy its long-term obligations.

Description

long-term debt-to-equity – Solvency ratios measure the firm's ability to satisfy its long-term obligations.

Usage

```
lt.d2e(ltd, te)
```

Arguments

ltd	long-term debt
te	total equity

See Also

[total.d2e](#)
[debt.ratio](#)
[financial.leverage](#)

Examples

```
lt.d2e(ltd=8000, te=20000)
```

mmy2hpr

Computing HPR, the holding period return

Description

Computing HPR, the holding period return

Usage

```
mmy2hpr(mmy, t)
```

Arguments

mmy	money market yield
t	number of days remaining until maturity

See Also

[bdy2mmy](#)
[hpr2mmy](#)
[hpr](#)

Examples

```
mmy2hpr(mmy=0.04898, t=150)
```

n.period	<i>Estimate the number of periods</i>
----------	---------------------------------------

Description

Estimate the number of periods

Usage

```
n.period(r, pv, fv, pmt, type = 0)
```

Arguments

r	discount rate, or the interest rate at which the amount will be compounded each period
pv	present value
fv	future value
pmt	payment per period
type	payments occur at the end of each period (type=0); payments occur at the beginning of each period (type=1)

See Also

[pv](#)
[fv](#)
[pmt](#)
[discount.rate](#)

Examples

```
n.period(0.1, -10000, 60000000, -50000, 0)  
n.period(r=0.1, pv=-10000, fv=60000000, pmt=-50000, type=1)
```

npm*net profit margin – Evaluate a company's financial performance***Description**

net profit margin – Evaluate a company's financial performance

Usage

```
npm(ni, rv)
```

Arguments

ni net income

rv revenue (sales)

See Also

[gpm](#)

Examples

```
npm(ni=8000, rv=20000)
```

npv*Computing NPV, the PV of the cash flows less the initial (time = 0) outlay***Description**

Computing NPV, the PV of the cash flows less the initial (time = 0) outlay

Usage

```
npv(r, cf)
```

Arguments

r discount rate, or the interest rate at which the amount will be compounded each period

cf cash flow, the first cash flow is the initial outlay

See Also[pv.simple](#)[pv.uneven](#)[irr](#)**Examples**

```
npv(r=0.12, cf=c(-5, 1.6, 2.4, 2.8))
```

pmt*Estimate period payment*

Description

Estimate period payment

Usage

```
pmt(r, n, pv, fv, type = 0)
```

Arguments

r	discount rate, or the interest rate at which the amount will be compounded each period
n	number of periods
pv	present value
fv	future value
type	payments occur at the end of each period (type=0); payments occur at the beginning of each period (type=1)

See Also[pv](#)[fv](#)[n.period](#)**Examples**

```
pmt(0.08,10,-1000,10)
```

```
pmt(r=0.08,n=10,pv=-1000,fv=0)
```

```
pmt(0.08,10,-1000,10,1)
```

pv	<i>Estimate present value (pv)</i>
----	------------------------------------

Description

Estimate present value (pv)

Usage

```
pv(r, n, fv = 0, pmt = 0, type = 0)
```

Arguments

r	discount rate, or the interest rate at which the amount will be compounded each period
n	number of periods
fv	future value
pmt	payment per period
type	payments occur at the end of each period (type=0); payments occur at the beginning of each period (type=1)

See Also

[pv.simple](#)
[pv.annuity](#)
[fv](#)
[pmt](#)
[n.period](#)
[discount.rate](#)

Examples

```
pv(0.07,10,1000,10)
```

```
pv(r=0.05,n=20,fv=1000,pmt=10,type=1)
```

<code>pv.annuity</code>	<i>Estimate present value (pv) of an annuity</i>
-------------------------	--

Description

Estimate present value (pv) of an annuity

Usage

```
pv.annuity(r, n, pmt, type = 0)
```

Arguments

<code>r</code>	discount rate, or the interest rate at which the amount will be compounded each period
<code>n</code>	number of periods
<code>pmt</code>	payment per period
<code>type</code>	payments occur at the end of each period (type=0); payments occur at the beginning of each period (type=1)

See Also

[pv](#)

Examples

```
pv.annuity(0.03,12,1000)
```

```
pv.annuity(r=0.0425,n=3,pmt=30000)
```

<code>pv.perpetuity</code>	<i>Estimate present value of a perpetuity</i>
----------------------------	---

Description

Estimate present value of a perpetuity

Usage

```
pv.perpetuity(r, pmt, g = 0, type = 0)
```

Arguments

<code>r</code>	discount rate, or the interest rate at which the amount will be compounded each period
<code>pmt</code>	payment per period
<code>g</code>	growth rate of perpetuity
<code>type</code>	payments occur at the end of each period (<code>type=0</code>); payments occur at the beginning of each period (<code>type=1</code>)

See Also

[`r.perpetuity`](#)

Examples

```
pv.perpetuity(r=0.1,pmt=1000,g=0.02)
```

```
pv.perpetuity(r=0.1,pmt=1000,type=1)
```

```
pv.perpetuity(r=0.1,pmt=1000)
```

pv.simple

Estimate present value (pv) of a single sum

Description

Estimate present value (pv) of a single sum

Usage

```
pv.simple(r, n, fv)
```

Arguments

<code>r</code>	discount rate, or the interest rate at which the amount will be compounded each period
<code>n</code>	number of periods
<code>fv</code>	future value

See Also

[`pv`](#)

Examples

```
pv.simple(0.07,10,100)
```

```
pv.simple(r=0.03,n=3,fv=1000)
```

`pv.uneven`

Computing the present value of an uneven cash flow series

Description

Computing the present value of an uneven cash flow series

Usage

`pv.uneven(r, cf)`

Arguments

<code>r</code>	discount rate, or the interest rate at which the amount will be compounded each period
<code>cf</code>	uneven cash flow

See Also

[pv.simple](#)
[npv](#)

Examples

`pv.uneven(r=0.1, cf=c(-1000, -500, 0, 4000, 3500, 2000))`

`quick.ratio`

quick ratio – Liquidity ratios measure the firm's ability to satisfy its short-term obligations as they come due.

Description

quick ratio – Liquidity ratios measure the firm's ability to satisfy its short-term obligations as they come due.

Usage

`quick.ratio(cash, ms, rc, cl)`

Arguments

<code>cash</code>	cash
<code>ms</code>	marketable securities
<code>rc</code>	receivables
<code>cl</code>	current liabilities

See Also

[current.ratio](#)
[cash.ratio](#)

Examples

```
quick.ratio(cash=3000,ms=2000,rc=1000,cl=2000)
```

r.continuous

Convert a given nominal rate to a continuous compounded rate

Description

Convert a given nominal rate to a continuous compounded rate

Usage

```
r.continuous(r, m)
```

Arguments

r	nominal rate
m	number of times compounded each year

See Also

[r.norminal](#)

Examples

```
r.continuous(0.03,4)
```

r.norminal

Convert a given continuous compounded rate to a nominal rate

Description

Convert a given continuous compounded rate to a nominal rate

Usage

```
r.norminal(rc, m)
```

Arguments

rc	continuous compounded rate
m	number of desired times compounded each year

See Also

[r.continuous](#)
[ear.continuous](#)

Examples

```
r.norminal(0.03,1)  
r.norminal(0.03,4)
```

r.perpetuity *Rate of return for a perpetuity*

Description

Rate of return for a perpetuity

Usage

```
r.perpetuity(pmt, pv)
```

Arguments

pmt	payment per period
pv	present value

See Also

[pv.perpetuity](#)

Examples

```
r.perpetuity(pmt=4.5,pv=-75)
```

`sampling.error` *Computing Sampling error*

Description

Computing Sampling error

Usage

```
sampling.error(sm, mu)
```

Arguments

<code>sm</code>	sample mean
<code>mu</code>	population mean

Examples

```
sampling.error(sm=0.45, mu=0.5)
```

`SFRatio` *Computing Roy's safety-first ratio*

Description

Computing Roy's safety-first ratio

Usage

```
SFRatio(rp, rl, sd)
```

Arguments

<code>rp</code>	portfolio return
<code>rl</code>	threshold level return
<code>sd</code>	standard deviation of portfolio retwns

See Also

[Sharpe.ratio](#)

Examples

```
SFRatio(rp=0.09, rl=0.03, sd=0.12)
```

Sharpe.ratio *Computing Sharpe Ratio*

Description

Computing Sharpe Ratio

Usage

```
Sharpe.ratio(rp, rf, sd)
```

Arguments

rp	portfolio return
rf	risk-free return
sd	standard deviation of portfolio retwns

See Also

[coefficient.variation](#)
[SFRatio](#)

Examples

```
Sharpe.ratio(rp=0.038, rf=0.015, sd=0.07)
```

slde *Depreciation Expense Recognition – Straight-line depreciation (SL) allocates an equal amount of depreciation each year over the asset's useful life*

Description

Depreciation Expense Recognition – Straight-line depreciation (SL) allocates an equal amount of depreciation each year over the asset's useful life

Usage

```
slde(cost, rv, t)
```

Arguments

cost	cost of long-lived assets
rv	residual value of the long-lived assets at the end of its useful life
t	length of the useful life

See Also[ddb](#)**Examples**

```
slde(cost=1200, rv=200, t=5)
```

total.d2e

total debt-to-equity – Solvency ratios measure the firm's ability to satisfy its long-term obligations.

Description

total debt-to-equity – Solvency ratios measure the firm's ability to satisfy its long-term obligations.

Usage

```
total.d2e(td, te)
```

Arguments

td	total debt
te	total equity

See Also

[total.d2e](#)
[debt.ratio](#)
[financial.leverage](#)

Examples

```
total.d2e(td=6000, te=20000)
```

twrr*Computing TWRR, the time-weighted rate of return*

Description

Computing TWRR, the time-weighted rate of return

Usage

```
twrr(ev, bv, cfr)
```

Arguments

ev	ordered ending value list
bv	ordered beginning value list
cfr	ordered cash flow received list

See Also

[hpr](#)

Examples

```
twrr(ev=c(120,260),bv=c(100,240),cfr=c(2,4))
```

volumeChart*Technical analysts - Volume charts: show each period's volume as a vertical line*

Description

Technical analysts - Volume charts: show each period's volume as a vertical line

Usage

```
volumeChart(ohlc, main = "", ...)
```

Arguments

ohlc	output from get.ohlc.yahoo or get.ohlc.google
main	an overall title for the plot
...	Arguments to be passed to ggplot

See Also

[get.ohlc.yahoo](#)
[get.ohlc.google](#)

Examples

```
# google <- get.ohlc.yahoo("GOOG");
# volumeChart(google)
# apple <- get.ohlc.google("AAPL");
# volumeChart(apple)
```

was

calculate weighted average shares – weighted average number of common shares

Description

calculate weighted average shares – weighted average number of common shares

Usage

```
was(ns, nm)
```

Arguments

ns	n x 1 vector vector of number of shares
nm	n x 1 vector vector of number of months relate to ns

See Also

[EPS](#)
[diluted.EPS](#)

Examples

```
s=c(10000,2000);m=c(12,6);was(ns=s,nm=m)

s=c(11000,4400,-3000);m=c(12,9,4);was(ns=s,nm=m)
```

wpr

Weighted mean as a portfolio return

Description

Weighted mean as a portfolio return

Usage

`wpr(r, w)`

Arguments

<code>r</code>	returns of the individual assets in the portfolio
<code>w</code>	corresponding weights associated with each of the individual assets

Examples

`wpr(r=c(0.12, 0.07, 0.03), w=c(0.5, 0.4, 0.1))`

Index

bdy, 3, 4
bdy2mmy, 3, 3, 29

candlestickChart, 4
cash.ratio, 5, 7, 36
coefficient.variation, 5, 39
cogs, 6
current.ratio, 5, 7, 36

ddb, 7, 40
debt.ratio, 8, 15, 28, 40
diluted.EPS, 9, 14, 26, 42
discount.rate, 10, 16, 29, 32

ear, 11, 12, 13, 23
ear.continuous, 11, 11, 37
ear2bey, 11, 12
ear2hpr, 11, 12, 23
EIR, 13
EPS, 9, 14, 42

financial.leverage, 8, 15, 28, 40
fv, 10, 15, 16, 17, 29, 31, 32
fv.annuity, 10, 16, 16
fv.simple, 10, 16, 17, 18
fv.uneven, 17

geometric.mean, 18
get.ohlc.google, 4, 18, 19, 20, 27, 42
get.ohlc.yahoo, 4, 19, 19, 21, 27, 42
get.ohlcs.google, 19, 20, 21, 27
get.ohlcs.yahoo, 19, 20, 20, 27
gpm, 21, 30

harmonic.mean, 22
hpr, 13, 22, 23, 24, 29, 41
hpr2bey, 23
hpr2ear, 11, 13, 22, 23
hpr2mmy, 22, 24, 29

IRR, 24, 25, 31

IRR2, 25
iss, 9, 26

lineChart, 26, 27
lineChartMult, 27
lt.d2e, 8, 15, 28

mmy2hpr, 24, 28

n.period, 10, 16, 29, 31, 32
npm, 21, 30
npv, 25, 30, 35

pmt, 10, 16, 29, 31, 32
pv, 10, 16, 29, 31, 32, 33, 34
pv.annuity, 32, 33
pv.perpetuity, 33, 37
pv.simple, 31, 32, 34, 35
pv.uneven, 25, 31, 35

quick.ratio, 5, 7, 35

r.continuous, 36, 37
r.norminal, 12, 36, 36
r.perpetuity, 34, 37

sampling.error, 38
SFRatio, 38, 39
Sharpe.ratio, 6, 38, 39
slde, 8, 39

total.d2e, 8, 15, 28, 40, 40
twrr, 22, 41

volumeChart, 41

was, 9, 14, 42
wpr, 43