Package 'imf'

April 20, 2016

Title Set of functions for financial evaluation of Software Projects

Version 1.	.0																						
Date 2016	5-04-20																						
Author Eb	oer Schmit	Z																					
Maintaine	r Antoanr	ne Pont	es <aı< th=""><th>ntoa</th><th>anne</th><th>@ufr</th><th>rj.b</th><th>r></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></aı<>	ntoa	anne	@ufr	rj.b	r>															
Description	n R packe	age wi	ith a s	et of	fun	ctior	ns fo	r fii	nano	cial	eva	lua	tio	n o	f Sc	ftv	var	e F	ro	jec	t.		
License Lo	GPL (>= 2	.1)																					
URL http	os://gith	ub.co	m/ant	oanı	ne/i	fm																	
BugRepor	ts https:	//git	hub.c	om/	anto	ann	e/i	fm/	iss	ues	5												
NeedsCom	pilation 1	10																					
di dı	es documents documents of the second	e 																					1 2 3 4
ifi ni nj	ir fv pv	 						 			•	· ·			 								6 7 8 9
Index																							11
imf-pac	ckage		Set	off	unct	ions	for	fina	ınci	al e	valı	ıati	on	of	Sofi	'wc	ıre	Pr	oje	cts			

Description

Type Package

R packeage with a set of functions for financial evaluation of Software Project.

2 disc

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.

~~ An overview of how to use the package, including the most important functions ~~

Author(s)

Eber Schmitz

Maintainer: Antoanne Pontes <antoanne@ufrj.br>

References

~~ Literature or other references for background information ~~

See Also

~~ Optional links to other man pages, e.g. ~~

Examples

```
## examples here...
```

disc

~~function to do ... ~~

Description

~~ A concise (1-5 lines) description of what the function does. ~~

Usage

```
disc(r, n, bop = FALSE)
```

Arguments

```
r ~~Describe r here~~
n ~~Describe n here~~
bop ~~Describe bop here~~
```

Details

~~ If necessary, more details than the description above ~~

Value

~Describe the value returned If it is a LIST, use

```
comp1 Description of 'comp1' comp2 Description of 'comp2'
```

drawefs 3

Note

```
~~further notes~~
```

Author(s)

```
~~who you are~~
```

References

~put references to the literature/web site here ~

See Also

```
~~objects to See Also as help, ~~~
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

## The function is currently defined as
function (r, n, bop = FALSE)
{
    t < -1/(1 + r)
    e <- if (bop) {
       0:(n - 1)
    }
    else {
       1:n
    }
    return(t^e)</pre>
```

drawcfs

 $\sim\sim$ function to do ... $\sim\sim$

Description

```
~~ A concise (1-5 lines) description of what the function does. ~~
```

Usage

```
drawcfs(cfs)
```

Arguments

cfs

~~Describe cfs here~~

Details

~~ If necessary, more details than the description above ~~

4 fv

Value

```
~Describe the value returned If it is a LIST, use
```

```
comp1 Description of 'comp1'
comp2 Description of 'comp2'
```

Note

```
~~further notes~~
```

Author(s)

```
~~who you are~~
```

References

~put references to the literature/web site here ~

See Also

```
~~objects to See Also as help, ~~~
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
## The function is currently defined as
function (cfs)
{
   plot(cfs, xlab = "Periodo", ylab = "y")
    abline(h = c(0))
    title("Fluxo de Caixa")
    count <- 0
    for (i in cfs) {
        count <- count + 1
        if (i != 0) {
           arrows(count, 0, count, i, length = 0.1, angle = 20)
    }
  }
```

fν

 $\sim\sim$ function to do ... $\sim\sim$

Description

```
~~ A concise (1-5 lines) description of what the function does. ~~
```

Usage

```
fv(Pv, r, n)
```

fv 5

Arguments

```
Pv ~~Describe Pv here~~
r ~~Describe r here~~
n ~~Describe n here~~
```

Details

~~ If necessary, more details than the description above ~~

Value

```
~Describe the value returned If it is a LIST, use
```

```
comp1 Description of 'comp1'
comp2 Description of 'comp2'
```

Note

```
~~further notes~~
```

Author(s)

```
~~who you are~~
```

References

~put references to the literature/web site here ~

See Also

```
~~objects to See Also as help, ~~~
```

Examples

```
##--- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
## The function is currently defined as
function (Pv, r, n)
{
    return(Pv * (1 + (r/100))^n)
}
```

6 ifir

ifir

~~ Function to calculate inflation free interest rate. ~~

Description

```
~~ A concise (1-5 lines) description of what the function does. ~~
```

Usage

```
ifir(r, f)
```

Arguments

```
r ~~Describe r here~~
f ~~Describe f here~~
```

Details

~~ If necessary, more details than the description above ~~

Value

~Describe the value returned If it is a LIST, use

```
comp1 Description of 'comp1' comp2 Description of 'comp2'
```

Note

```
~~further notes~~
```

Author(s)

```
~~who you are~~
```

References

~put references to the literature/web site here ~

See Also

```
~~objects to See Also as help, ~~~
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
## The function is currently defined as
function (r, f)
{
    return((r - f)/(1 + f))
}
```

nfv 7

nfv ~~function to do ... ~~

Description

~~ A concise (1-5 lines) description of what the function does. ~~

Usage

```
nfv(cfs, r, bop = TRUE)
```

Arguments

```
cfs ~~Describe cfs here~~
r ~~Describe r here~~
bop ~~Describe bop here~~
```

Details

~~ If necessary, more details than the description above ~~

Value

~Describe the value returned If it is a LIST, use

comp1 Description of 'comp1'
comp2 Description of 'comp2'

Note

```
~~further notes~~
```

Author(s)

```
~~who you are~~
```

References

~put references to the literature/web site here ~

See Also

```
~~objects to See Also as help, ~~~
```

8 npv

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

## The function is currently defined as
function (cfs, r, bop = TRUE)
{
    e <- if (bop) {
        0:(length(cfs) - 1)
    }
    else {
        1:(length(cfs))
    }
    tax <- (1 + (r/100))^e
    return(cfs * tax)
}</pre>
```

npv

~~function to do ... ~~

Description

~~ A concise (1-5 lines) description of what the function does. ~~

Usage

```
npv(cfs, r, bop = TRUE)
```

Arguments

```
cfs ~~Describe cfs here~~
r ~~Describe r here~~
bop ~~Describe bop here~~
```

Details

~~ If necessary, more details than the description above ~~

Value

~Describe the value returned If it is a LIST, use

```
comp1 Description of 'comp1'
comp2 Description of 'comp2'
```

Note

```
~~further notes~~
```

Author(s)

```
~~who you are~~
```

pv 9

References

~put references to the literature/web site here ~

See Also

```
~~objects to See Also as help, ~~~
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.

## The function is currently defined as
function (cfs, r, bop = TRUE)
{
    e <- if (bop) {
        0:(length(cfs) - 1)
    }
    else {
        1:(length(cfs))
    }
    tax <- (1/((1 + (r/100))^e))
    return(cfs * tax)
}</pre>
```

pν

~~function to do ... ~~

Description

~~ A concise (1-5 lines) description of what the function does. ~~

Usage

```
pv(Fv, r, n)
```

Arguments

```
Fv ~~Describe Fv here~~
r ~~Describe r here~~
n ~~Describe n here~~
```

Details

~~ If necessary, more details than the description above ~~

Value

~Describe the value returned If it is a LIST, use

```
comp1 Description of 'comp1' comp2 Description of 'comp2'
```

10 pv

Note

```
~~further notes~~
```

Author(s)

```
~~who you are~~
```

References

~put references to the literature/web site here ~

See Also

```
~~objects to See Also as help, ~~~
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==> Define data, use random,
##--or do help(data=index) for the standard data sets.
## The function is currently defined as
function (Fv, r, n)
{
    return(Fv/(1 + (r/100))^n)
}
```

Index

```
*Topic \textasciitildekwd1
     disc, 2
     drawcfs, 3
     fv, 4
     ifir, 6
     nfv, 7
     npv, 8
     pv, 9
*Topic \textasciitildekwd2
     \operatorname{disc}, 2
     drawcfs, 3
     fv, 4
     ifir, 6
     nfv, 7
     npv, 8
     pv, 9
*Topic package
     imf-package, 1
disc, 2
drawcfs, 3
fv, 4
help, 3–7, 9, 10
ifir, 6
imf(imf-package), 1
imf-package, 1
nfv, 7
npv, 8
pν, <mark>9</mark>
```