Package ‘iemisc’

October 13, 2022

Title Irucka Embry's Miscellaneous Functions

Version 0.9.9

Author Irucka Embry [aut, cre], Felix Andrews [aut, ctb] (zoo code), Hans Werner Borchers [aut, ctb] (pracma code), Samit Basu [aut, ctb] (FreeMat code), David Bateman [aut, ctb] (GNU Octave code), Rik Wehbring [aut, ctb] (GNU Octave code)

Maintainer Irucka Embry `<iembry@ecoccs.com>`

Depends R (>= 3.0.0)

Imports zoo, pracma, iemiscdata, import, gsubfn (>= 0.7), fpCompare, units, CHNOSZ, testit, stringi, rgdal, sp, stats, data.table (>= 1.10.2)

Suggests install.load, ie2misc, Rcpp (>= 0.11.5), NISTunits, optiRum, rivr, knitr, rmarkdown, ramify

Description A collection of Irucka Embry's miscellaneous functions (Engineering Economics, Civil & Environmental/Water Resources Engineering, Geometry, Statistics, GNU Octave length functions, Trigonometric functions in degrees, etc.).

URL https://gitlab.com/iembry/iemisc

BugReports https://gitlab.com/iembry/iemisc/issues

License GPL (>= 3) | file LICENSE

Language en-US

Encoding UTF-8

LazyData true

NeedsCompilation no

RoxygenNote 7.1.1

Repository CRAN

Date/Publication 2020-08-02 01:30:02 UTC

R topics documented:

`acosd` *Inverse cosine (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of inverse cosine for each element of `x` in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
acosd(x)
```

Arguments

`x`	A numeric vector containing values in degrees

Value

The inverse cosine of each element of `x` in degrees.

Author(s)

David Bateman (GNU Octave acosd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 359.

Examples

```
library("iemisc")

# Examples from GNU Octave acosd
acosd (seq(0, 1, by = 0.1))
```

---

acotd *Inverse cotangent (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of inverse cotangent for each element of x in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
acotd(x)
```

Arguments

x	A numeric vector containing values in degrees

Value

The inverse cotangent of each element of x in degrees.

Author(s)

David Bateman (GNU Octave acotd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 359.

Examples

```
library("iemisc")

# Examples from GNU Octave acotd
acotd (seq(0, 90, by = 10))
```

`acscd` *Inverse cosecant (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of inverse cosecant for each element of `x` in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
acscd(x)
```

Arguments

`x`	A numeric vector containing values in degrees

Value

The inverse cosecant of each element of `x` in degrees.

Author(s)

David Bateman (GNU Octave acscd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 359.

Examples

```
library("iemisc")

# Examples from GNU Octave acscd
acscd (seq(0, 90, by = 10))
```

AgivenF — *Annual value given Future value (Engineering Economics)*

Description

Compute A given F

Usage

```
AgivenF(
  F,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)

AF(
  F,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`F`	numeric vector that contains the future value(s)
`n`	numeric vector that contains the period value(s)
`i`	numeric vector that contains the interest rate(s) as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

A is expressed as

$$A = F\left[\frac{i}{\left(1+i\right)^{n}-1}\right]$$

A the "uniform series amount (occurs at the end of each interest period)"

F the "future equivalent"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

AgivenF numeric vector that contains the annual value(s) rounded to 2 decimal places

AF data.frame of both n (0 to n) and the resulting annual values rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 135-136, 142, 164.

Examples

```
library("iemisc")
# Example for equation 4-12 from the Reference text (page 135-136)
AgivenF(309*10^6, 60, 0.5, "annual")
# the interest rate is 0.5% per month and n is 60 months

AF(309*10^6, 60, 0.5, "annual")
# the interest rate is 0.5% per month and n is 60 months
```

AgivenFcont — *Annual value given Future value [continuous] (Engineering Economics)*

Description

Compute A given F with interest compounded continuously

Usage

```
AgivenFcont(F, n, r)
```

Arguments

F	numeric vector that contains the future value(s)
n	numeric vector that contains the period value(s)
r	numeric vector that contains the continuously compounded nominal annual interest rate(s) as a percent

Details

A is expressed as

$$A = F\left[\frac{e^{r} - 1}{e^{rn} - 1}\right]$$

A the "annual equivalent amount (occurs at the end of each year)"

F the "future equivalent"

r the "nominal annual interest rate, compounded continuously"

n the "number of periods (years)"

Value

AgivenFcont numeric vector that contains the annual value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 169.

Examples

```
library("iemisc")
AgivenFcont(300, 2, 11) # 11% interest
```

AgivenG *Annual value given Gradient value (Engineering Economics)*

Description

Compute A given G

Usage

```
AgivenG(
  G,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`G`	numeric vector that contains the gradient value(s)
`n`	numeric vector that contains the period value(s)
`i`	numeric vector that contains the interest rate(s) as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

$$A = G\left[\frac{1}{i} - \frac{n}{(1+i)^n - 1}\right]$$

A the "uniform series amount (occurs at the end of each interest period)"

G the "uniform gradient amount"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

AgivenG numeric vector that contains the annual value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 142, 150, 152-154, 164, 166-167.

Examples

```
library("iemisc")
# Example 4-20 from the Reference text (pages 153-154)
  AgivenG(1000, 4, 15, "annual") # the interest rate is 15%


# Example 4-31 from the Reference text (pages 166-167)
 AgivenG(1000, 4, 20, "semiannual") # the nominal interest rate is 20% compounded semiannually
```

AgivenP *Annual value given Present value (Engineering Economics)*

Description

Compute A given P

Usage

```
AgivenP(
  P,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
```

```
)

AP(
  P,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

P	numeric vector that contains the present value(s)
n	numeric vector that contains the period value(s)
i	numeric vector that contains the interest rate(s) as a percent
frequency	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

A is expressed as

$$A = P\left[\frac{i\left(1+i\right)^{n}}{\left(1+i\right)^{n}-1}\right]$$

A the "uniform series amount (occurs at the end of each interest period)"

P the "present equivalent"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

AgivenP numeric vector that contains the annual value(s) rounded to 2 decimal places

AP data.frame of both n (0 to n) and the resulting annual values rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 136, 142, 164, 166.

Examples

```
library("iemisc")
# Example for equation 4-14 from the Reference text (page 136)
AgivenP(17000, 4, 1, "annual")
# the interest rate is 1% per month and n is 4 months

AP(17000, 4, 1, "annual")
# the interest rate is 1% per month and n is 4 months
```

```
# Example 4-30 from the Reference text (page 166)
AgivenP(10000, 5, 12, "month")
# the interest rate is 12% compounded monthly for 5 years

AP(10000, 5, 12, "month")
# the interest rate is 12% compounded monthly for 5 years
```

`AgivenPcont` *Annual value given Present value [continuous] (Engineering Economics)*

Description

Compute A given P with interest compounded continuously

Usage

```
AgivenPcont(P, n, r)
```

Arguments

`P`	numeric vector that contains the present value(s)
`n`	numeric vector that contains the period value(s)
`r`	numeric vector that contains the continuously compounded nominal annual interest rate(s) as a percent

Details

A is expressed as

$$A = P\left[\frac{e^{rn}\left(e^{r}-1\right)}{e^{rn}-1}\right]$$

A the "annual equivalent amount (occurs at the end of each year)"

P the "present equivalent"

r the "nominal annual interest rate, compounded continuously"

n the "number of periods (years)"

Value

AgivenPcont numeric vector that contains the annual value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 169-170.

Examples

```
library("iemisc")
# Example for equation 4-34 from the Reference text (page 170)
AgivenPcont(1000, 10, 20) # 20% interest
```

---

approxerror *Approximate error*

Description

This function computes the "approximate estimate of the error" ("percent relative error").

Usage

```
approxerror(pres, prev)
```

Arguments

`pres`	numeric vector that contains the "present approximation" value(s)
`prev`	numeric vector that contains the "previous approximation" value(s)

Details

Approximate error is expressed as

$$\varepsilon_a = \frac{present\ approximation - previous\ approximation}{present\ approximation} \cdot 100$$

ε_a the "approximate estimate of the error"

present approximation the "present approximation"

previous approximation the "previous approximation"

Value

approximate error, as a percent (%), as a numeric `vector`.

References

Steven C. Chapra, *Applied Numerical Methods with MATLAB for Engineers and Scientists*, Second Edition, Boston, Massachusetts: McGraw-Hill, 2008, page 82-84.

See Also

sgm for geometric mean, shm for harmonic mean, cv for coefficient of variation (CV), rms for root-mean-square (RMS), relerror for relative error, and ranges for sample range.

Examples

```
library("iemisc")
# Example 4.1 from the Reference text (page 84)
approxerror(1.5, 1) # answer as a percent (\%)
```

---

asecd *Inverse secant (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of inverse secant for each element of x in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
asecd(x)
```

Arguments

x	A numeric vector containing values in degrees

Value

The inverse secant of each element of x in degrees.

Author(s)

David Bateman (GNU Octave asecd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave asecd
asecd (seq(0, 90, by = 10))
```

---

asind *Inverse sine (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of inverse sine for each element of x in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
asind(x)
```

Arguments

x	A numeric vector containing values in degrees

Value

The inverse sine of each element of x in degrees.

Author(s)

David Bateman (GNU Octave asind), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 359.

Examples

```
library("iemisc")

# Examples from GNU Octave asind
asind (seq(0, 1, by = 0.1))
```

atan2d — *"Two-argument arc-tangent" (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of the "two-argument arc-tangent" for each element of (y, x) in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
atan2d(y, x)
```

Arguments

`y`	A numeric vector containing values in degrees
`x`	A numeric vector containing values in degrees

Value

The "two-argument arc-tangent" of each element of (y, x) in degrees. Note: "The arc-tangent of two arguments atan2(y, x) returns the angle between the x-axis and the vector from the origin to (x, y), i.e., for positive arguments atan2(y, x) == atan(y/x)." Source: `Trig` (base).

Author(s)

Rik Wehbring (GNU Octave atan2d), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave atan2d
atan2d (a <- seq(-1, 1, by = 0.1), b <- seq(1, -1, by = -0.1))
```

atand *Inverse tangent (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of inverse tangent for each element of x in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
atand(x)
```

Arguments

x	A numeric vector containing values in degrees

Value

The inverse tangent of each element of x in degrees.

Author(s)

David Bateman (GNU Octave atand), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 359.

Examples

```
library("iemisc")

# Examples from GNU Octave atand
atand (seq(0, 90, by = 10))
```

benefitcost *Benefit-Cost Ratio (Engineering Economics)*

Description

Compute the benefit-cost ratio between two alternatives

Usage

```
benefitcost(
  ic1,
  n1,
  ac1,
  ab1,
  i1,
  salvage1,
  ic2,
  n2,
  ac2,
  ab2,
  i2,
  salvage2,
  option1,
  option2,
  table = c("ptable", "rtable", "both")
)
```

Arguments

`ic1`	numeric vector that contains the initial cost for option 1
`n1`	numeric vector that contains the useful life (years) for option 1
`ac1`	numeric vector that contains the annual cost [operations & maintenance (O&M)] for option 1
`ab1`	numeric vector that contains the annual benefits for option 1
`i1`	numeric vector that contains the effective interest rate per period as a percent for option 1
`salvage1`	numeric vector that contains the salvage value for option 1
`ic2`	numeric vector that contains the initial cost for option 2
`n2`	numeric vector that contains the useful life (years) for option 2
`ac2`	numeric vector that contains the annual cost [operations & maintenance (O&M)] for option 2
`ab2`	numeric vector that contains the annual benefits for option 2
`i2`	numeric vector that contains the effective interest rate per period as a percent for option 2

`salvage2` numeric vector that contains the salvage value for option 2

`option1` character vector that contains the name of option for option 1

`option2` character vector that contains the name of option for option 2

`table` character vector that contains the table output format (ptable, rtable, or both)

Details

Benefit is expressed as

$$Benefit = AB \left[\frac{(1+i)^n - 1}{i\,(1+i)^n} \right]$$

Benefit the present equivalent benefit

AB the annual benefit

i the "effective interest rate" per year

n the number of years

Cost is expressed as

$$Cost = PC + OM \left[\frac{(1+i)^n - 1}{i\,(1+i)^n} \right] - S \left[\frac{1}{(1+i)^n} \right]$$

Cost the present equivalent cost

PC the present or initial cost

OM the annual operations & maintenance cost

S the salvage value

i the "effective interest rate" per year

n the number of years

Benefit-Cost ratio is expressed as

$$BC = \frac{B_2 - B_1}{C_2 - C_1} \geq 1$$

BC the present equivalent cost

B_1 the benefit for alternative 1

B_2 the benefit for alternative 2

C_1 the cost for alternative 1

C_2 the cost for alternative 2

Value

`data.table` with character vectors with the monetary values having thousands separator in a pretty table (ptable) & message with the best option, data.frame with numeric vectors without the thousands separator in regular table (rtable) & a message with the best option, or both options combined in a list

References

1. Michael R. Lindeburg, PE, *EIT Review Manual*, Belmont, California: Professional Publications, Inc., 1996, page 14-2, 14-4.
2. William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 133, 142, 442-443, 452-453.

Examples

```
library("iemisc")
# Example from Lindeburg Reference text (page 14-4)
benefitcost(ic1 = 300000, n1 = 10, ac1 = 45000, ab1 = 150000, i1 = 10,
salvage1 = 0, ic2 = 400000, n2 = 10, ac2 = 35000, ab2 = 200000, i2 = 10,
salvage2 = 10000, option1 = "A", option2 = "B", table = "rtable")


# This is useful for saving the results as the named data.frame rtable
rtable <- benefitcost(ic1 = 300000, n1 = 10, ac1 = 45000, ab1 = 150000,
i1 = 10, salvage1 = 0, ic2 = 400000, n2 = 10, ac2 = 35000, ab2 = 200000,
i2 = 10, salvage2 = 10000, option1 = "A", option2 = "B", table = "rtable")

rtable


# This is useful for saving the results as the named data.frame ptable
ptable <- benefitcost(ic1 = 300000, n1 = 10, ac1 = 45000, ab1 = 150000,
i1 = 10, salvage1 = 0, ic2 = 400000, n2 = 10, ac2 = 35000, ab2 = 200000,
i2 = 10, salvage2 = 10000, option1 = "A", option2 = "B", table = "ptable")

ptable


# This is useful for saving the results as the named list of 2 data.frames
# called both
both <- benefitcost(ic1 = 300000, n1 = 10, ac1 = 45000, ab1 = 150000,
i1 = 10, salvage1 = 0, ic2 = 400000, n2 = 10, ac2 = 35000, ab2 = 200000,
i2 = 10, salvage2 = 10000, option1 = "A", option2 = "B", table = "both")


both


# Example 10-8 from the Sullivan Reference text (page 452-453)
project <- benefitcost(ic1 = 750000, n1 = 35, ac1 = 120000, ab1 = 245000,
i1 = 9, salvage1 = 0, ic2 = 625000, n2 = 25, ac2 = 110000, ab2 = 230000,
i2 = 9, salvage2 = 0, option1 = "Project I", option2 = "Project II",
table = "rtable")

project
```

CompIntPaid *Compound Interest Paid (Engineering Economics)*

Description

Computes the total amount paid at the end of n periods using compound interest

Usage

```
CompIntPaid(
  P,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`P`	numeric vector that contains the present value(s)
`n`	numeric vector that contains the period value(s)
`i`	numeric vector that contains the interest rate(s) as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

Compound Interest is expressed as

$$S_n = P\left(1 + i\right)^n$$

P the "principal amount (lent or borrowed)"

S_n the "total amount paid back"

i the "interest rate per interest period"

n the "number of interest periods"

Value

CompIntPaid numeric vector that contains the total amount paid at the end of n periods rounded to 2 decimal places

References

1. *SFPE Handbook of Fire Protection Engineering*. 3rd Edition, DiNenno, P. J.; Drysdale, D.; Beyler, C. L.; Walton, W. D., Editor(s), 5/93-104 p., 2002. Chapter 7; Section 5; NFPA HFPE-02. See http://fire.nist.gov/bfrlpubs//build02/art155.html.
2. William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 120.
3. Chinyere Onwubiko, *An Introduction to Engineering*, Mission, Kansas: Schroff Development Corporation, 1997, page 205-206.

Examples

```
library("iemisc")
# Compound Interest example from SFPE Reference text
CompIntPaid(100, 5, 10, frequency = "annual") # the interest rate is 10%
```

---

cosd — *Cosine (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of cosine for each element of `x` in degrees in a manner compatible with GNU Octave/MATLAB. Zero is returned for any "elements where (`x` - 90) / 180 is an integer." Source: Eaton.

Usage

```
cosd(x)
```

Arguments

x	A numeric vector containing values in degrees

Value

The cosine of each element of `x` in degrees. Zero for any "elements where (`x` - 90) / 180 is an integer."

Author(s)

David Bateman (GNU Octave cosd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave cosd
cosd(seq(0, 80, by = 10))

cosd(pi * seq(0, 80, by = 10) / 180)

cosd(c(0, 180, 360))

cosd(c(90, 270, 45))
```

---

cotd — *Cotangent (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of inverse secant for each element of x in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
cotd(x)
```

Arguments

x	A numeric vector containing values in degrees

Value

The inverse secant of each element of x in degrees.

Author(s)

David Bateman (GNU Octave cotd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave cotd
cotd (seq(0, 80, by = 10))

cotd (c(0, 180, 360))

cotd (c(90, 270))
```

---

cscd *Cosecant (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of cosecant for each element of `x` in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
cscd(x)
```

Arguments

x	A numeric vector containing values in degrees

Value

The cosecant of each element of `x` in degrees.

Author(s)

David Bateman (GNU Octave cscd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave cscd
cscd (seq(0, 90, by = 10))

cscd (c(0, 180, 360))

cscd (c(90, 270))
```

---

cv — *Coefficient of variation (CV)*

Description

This function computes the sample coefficient of variation (CV).

Usage

```
cv(x, na.rm = FALSE)
```

Arguments

x	numeric vector, matrix, data.frame, or data.table that contains the sample data points.
na.rm	logical vector that determines whether the missing values should be removed or not.

Details

CV is expressed as

$$\frac{s}{\bar{x}} \cdot 100$$

s the sample standard deviation

$\bar{x}$ the sample arithmetic mean

Value

coefficient of variation (CV), as a percent (%), as an R object: a numeric `vector` or a named numeric vector if using a named object (`matrix`, `data.frame`, or `data.table`). The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `cv(x, na.rm = TRUE)`.

References

1. Masoud Olia, Ph.D., P.E. and Contributing Authors, *Barron's FE (Fundamentals of Engineering Exam)*, 3rd Edition, Hauppauge, New York: Barron's Educational Series, Inc., 2015, page 84.

2. Irwin R. Miller, John E. Freund, and Richard Johnson, *Probability and Statistics for Engineers*, Fourth Edition, Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1990, page 25, 38.

See Also

`sgm` for geometric mean, `shm` for harmonic mean, `rms` for root-mean-square (RMS), `relerror` for relative error, `approxerror` for approximate error, and `ranges` for sample range.

Examples

```
library("iemisc")
library("data.table")

# Example 2.60 from Miller (page 38)
x <- c(14, 12, 21, 28, 30, 63, 29, 63, 55, 19, 20) # suspended solids in
     # parts per million (ppm)
cv(x)


# using a matrix of the numeric vector x
mat1 <- matrix(data = x, nrow = length(x), ncol = 1, byrow = FALSE,
        dimnames = list(c(rep("", length(x))), "Samples"))
cv(mat1)


# using a data.frame of the numeric vector x
df1 <- data.frame(x)
cv(df1)


# using a data.table of the numeric vector x
df2 <- data.table(x)
cv(df2)
```

EffInt — *Effective Interest rate (Engineering Economics)*

Description

Computes the effective interest rate given the nominal interest rate per period

Usage

```
EffInt(
  r,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`r`	numeric vector that contains the nominal interest rate(s) per period as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

i is expressed as

$$i = \left(1 + \frac{r}{n}\right)^n - 1$$

i the "effective interest rate per interest period"

r the "nominal interest rate"

n the "number of compounding periods per year"

Value

EffInt numeric vector that contains the effective interest rate rounded to 2 decimal places (this is the `i` used in the other Engineering Economics functions)

References

1. *SFPE Handbook of Fire Protection Engineering*. 3rd Edition, DiNenno, P. J.; Drysdale, D.; Beyler, C. L.; Walton, W. D., Editor(s), 5/93-104 p., 2002. Chapter 7; Section 5; NFPA HFPE-02. See http://fire.nist.gov/bfrlpubs//build02/art155.html.
2. William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 164-165.

Examples

```
library("iemisc")
# Example 4-28 from Sullivan Reference text (page 165)
EffInt(1.375, frequency = "month")
# the nominal interest rate per period (month) is 1.375%


# Example from SFPE Reference text
EffInt(18 / 12, frequency = "month")
# the nominal interest rate is 18% per year or 18% / 12 months
```

FgivenA — *Future value given Annual value (Engineering Economics)*

Description

Compute F given A

Usage

```
FgivenA(
  A,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)

FA(
  A,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`A`	numeric vector that contains the annual value(s)
`n`	numeric vector that contains the period value(s)
`i`	numeric vector that contains the interest rate(s) as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

F is expressed as

$$F = A \left[\frac{(1+i)^n - 1}{i} \right]$$

F the "future equivalent"

A the "uniform series amount (occurs at the end of each interest period)"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

FgivenA numeric vector that contains the future value(s) rounded to 2 decimal places

FA data.frame of both n (0 to n) and the resulting future values rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 131-132, 142, 164.

Examples

```
library("iemisc")
# Example 4-7 from the Reference text (page 131-132)
FgivenA(23000, 40, 6, "annual") # the interest rate is 6%

FA(23000, 40, 6, "annual") # the interest rate is 6%
```

`FgivenAcont` *Future value given Annual value [continuous] (Engineering Economics)*

Description

Compute F given A with interest compounded continuously

Usage

```
FgivenAcont(A, n, r)
```

Arguments

`A`	numeric vector that contains the annual value(s)
`n`	numeric vector that contains the period value(s)
`r`	numeric vector that contains the continuously compounded nominal annual interest rate(s) as a percent

Details

F is expressed as

$$F = A \left[\frac{e^{rn} - 1}{e^{r} - 1} \right]$$

F the "future equivalent"

A the "annual equivalent amount (occurs at the end of each year)"

r the "nominal annual interest rate, compounded continuously"

n the "number of periods (years)"

Value

FgivenAcont numeric vector that contains the future value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 169.

Examples

```
library("iemisc")
FgivenAcont(2100, 13, 7) # the interest rate is 7%
```

`FgivenP` *Future value given Present value (Engineering Economics)*

Description

Compute F given P

Usage

```
FgivenP(
  P,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)

FP(
  P,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`P`	numeric vector that contains the present value(s)
`n`	numeric vector that contains the period value(s)
`i`	numeric vector that contains the interest rate(s) as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

F is expressed as

$$F = P\left(1 + i\right)^n$$

F the "future equivalent"

P the "present equivalent"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

FgivenP numeric vector that contains the future value(s) rounded to 2 decimal places

FP data.frame of both n (0 to n) and the resulting future values rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 124, 142, 164-166.

Examples

```
library("iemisc")
# Example 4-3 from the Reference text (page 124)
FgivenP(8000, 4, 10, frequency = "annual") # the interest rate is 10%

FP(8000, 4, 10, frequency = "annual") # the interest rate is 10%


FgivenP(P = c(1000, 340, 23), n = c(12, 1.3, 3), i = c(10, 2, 0.3),
"annual")
# is is 10%, 2%, and 0.3%
# Can't use FP for this example


# Example 4-29 from the Reference text (page 165-166)
FgivenP(100, 10, 6, "quarter") # the interest rate is 6% per quarter

FP(100, 10, 6, "quarter") # the interest rate is 6% per quarter
```

`FgivenPcont` *Future value given Present value [continuous] (Engineering Economics)*

Description

Compute F given P with interest compounded continuously

Usage

```
FgivenPcont(P, n, r)
```

Arguments

`P`	numeric vector that contains the present value(s)
`n`	numeric vector that contains the period value(s)
`r`	numeric vector that contains the continuously compounded nominal annual interest rate(s) as a percent

Details

F is expressed as

$$F = Pe^{rn}$$

F the "future equivalent"

P the "present equivalent"

r the "nominal annual interest rate, compounded continuously"

n the "number of periods (years)"

Value

FgivenPcont numeric vector that contains the future value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 169-170.

Examples

```
library("iemisc")
# Example 4-33 from the Reference text (page 170)
FgivenPcont(10000, 2, 5) # the interest rate is 5%
```

---

iemisc *iemisc: Irucka Embry's miscellaneous functions*

Description

iemisc provides many useful functions. There are statistical analysis [RMS, coefficient of variation (CV), approximate and relative error, range, harmonic mean, geometric mean], engineering economics (benefit-cost, future value, present value, annual value, gradients, interest, periods, etc.), geometry (sphere volume and right triangle), civil & environmental/ water resources engineering (Air Stripper, Concrete Mix Design for Normal Strength & Structural Lightweight Concrete, Manning's n, Gauckler-Manning-Strickler equation for geometric cross-sections), a version of linear interpolation for use with NAs, & GNU Octave/MATLAB compatible size, numel, and length_octave functions.

---

igivenPFn *Interest rate given Future value, Number of periods, and Present value (Engineering Economics)*

Description

Compute i given F, n, and P

Usage

```
igivenPFn(P, F, n)
```

Arguments

P	numeric vector that contains the present value(s)
F	numeric vector that contains the future value(s)
n	numeric vector that contains the period value(s)

Details

i is expressed as

$$i = \sqrt[n]{\frac{F}{P}} - 1$$

i the "effective interest rate per interest period"

F the "future equivalent"

P the "present equivalent"

n the "number of interest periods

Value

i numeric vector that contains the effective interest rate as a percent rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 128-129, 142.

Examples

```
library("iemisc")
# Example for equation 4-6 from the Reference text (page 128)
igivenPFn(P = 500, F = 1000, n = 10)
```

`length_octave` *Length of R objects (GNU Octave/MATLAB compatible)*

Description

Obtain the length of R objects [arrays, matrices, and vectors (including lists)] in a manner compatible with GNU Octave/MATLAB. Some documentation from `length`.

Usage

```
length_octave(x)
```

Arguments

x An R object (array, matrix, vector)

Value

Return the length of the object x as an integer. "The length is 0 for empty objects, 1 for scalars (in R, a `vector` of `length` 1), and the number of elements (in R, the `length`) for vectors. For matrix objects, the length is the number of rows or columns, whichever is greater (this odd definition is used for compatibility with MATLAB)." Source: Eaton.

Author(s)

Irucka Embry, Samit Basu (FreeMat)

References

1. Samit Basu (2002-2006). FreeMat v4.0, http://freemat.sourceforge.net/.
2. John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 41.

See Also

`length`, `lengths`, `size`, `size`

Examples

```
library("iemisc")
import::from(pracma, ones)
# Example from pracma isempty
object1 <- matrix(0, 1, 0)
length_octave(object1)

object2 <- 2
length_octave(object2)

object3 <- 1:10
length_octave(object3)

object4 <- ones(3, 4)
length_octave(object4)

object5 <- "ss"
length_octave(object5)

object6 <- list(letters, b <- 2)
length_octave(object6)


## Not run:
```

```
# check against GNU Octave
library(RcppOctave) # requires Octave (>= 3.2.4) and its development files
o_source(text = "
object1 = [];
length(object1)

object2 = 2;
length(object2)

object3 = 1:10;
length(object3)

object4 = ones(3, 4);
length(object4)

object5 = 'ss';
length(object5)
")

## End(Not run)
```

`Manningcirc` *Circular cross-section using the Gauckler-Manning-Strickler equation*

Description

Manningcirc and Manningcircy solve for a missing variable for a circular cross-section. The `uniroot` function is used to obtain the missing parameter.

Usage

```
Manningcirc(
  Q = NULL,
  n = NULL,
  Sf = NULL,
  y = NULL,
  d = NULL,
  T = NULL,
  units = c("SI", "Eng")
)

Manningcircy(
  y = NULL,
  d = NULL,
  y_d = NULL,
  theta = NULL,
  Sf = NULL,
```

```
  Q = NULL,
  units = c("SI", "Eng")
)
```

Arguments

`Q`	numeric vector that contains the discharge value [m^3/s or ft^3/s], if known.
`n`	numeric vector that contains the Manning's roughness coefficient n, if known.
`Sf`	numeric vector that contains the bed slope (m/m or ft/ft), if known.
`y`	numeric vector that contains the flow depth (m or ft), if known.
`d`	numeric vector that contains the diameter value (m or ft), if known.
`T`	numeric vector that contains the temperature (degrees C or degrees Fahrenheit), if known.
`units`	character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English units (United States Customary System in the United States and Imperial Units in the United Kingdom)]
`y_d`	numeric vector that contains the filling ration (y/d), if known.
`theta`	numeric vector that contains the angle theta (radians), if known.

Details

The Manningcirc function solves for one missing variable in the Gauckler- Manning equation for a circular cross-section and uniform flow. The possible inputs are Q, n, Sf, y, and d. If y or d are not initially known, then Manningcircy can solve for y or d to use as input in the Manningcirc function.

The Manningcircy function solves for one missing variable in the Gauckler- Manning equation for a circular cross-section and uniform flow. The possible inputs are y, d, y_d (ratio of y/d), and theta.

Gauckler-Manning-Strickler equation is expressed as

$$V = \frac{K_n}{n} R^{\frac{2}{3}} \sqrt{S}$$

V the velocity (m/s or ft/s)

n Manning's roughness coefficient (dimensionless)

R the hydraulic radius (m or ft)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

This equation is also expressed as

$$Q = \frac{K_n}{n} \frac{A^{\frac{5}{3}}}{P^{\frac{2}{3}}} \sqrt{S}$$

Q the discharge [m^3/s or ft^3/s (cfs)] is VA

n Manning's roughness coefficient (dimensionless)

P the wetted perimeter of the channel (m or ft)

A the cross-sectional area (m^2 or ft^2)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

Other important equations regarding the circular cross-section follow:

$$R = \frac{A}{P}$$

R the hydraulic radius (m or ft)

A the cross-sectional area (m^2 or ft^2)

P the wetted perimeter of the channel (m or ft)

$$A = (\theta - \sin\theta)\,\frac{d^2}{8}$$

A the cross-sectional area (m^2 or ft^2)

d the diameter of the cross-section (m or ft)

θ see the equation defining this parameter

$$\theta = 2\arcsin\left[1 - 2\left(\frac{y}{d}\right)\right]$$

θ see the equation defining this parameter

y the flow depth (normal depth in this function) [m or ft]

d the diameter of the cross-section (m or ft)

$$d = 1.56\left[\frac{nQ}{K_n\sqrt{S}}\right]^{\frac{3}{8}}$$

d the initial diameter of the cross-section [m or ft]

Q the discharge [m^3/s or ft^3/s (cfs)] is VA

n Manning's roughness coefficient (dimensionless)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

Note: This will only provide the initial conduit diameter, check the design considerations to determine your next steps.

$$P = \frac{\theta d}{2}$$

P the wetted perimeter of the channel (m or ft)

θ see the equation defining this parameter

d the diameter of the cross-section (m or ft)

$$B = d \sin\left(\frac{\theta}{2}\right)$$

B the top width of the channel (m or ft)

θ see the equation defining this parameter

d the diameter of the cross-section (m or ft)

$$D = \frac{A}{B}$$

D the hydraulic depth (m or ft)

A the cross-sectional area (m^2 or ft^2)

B the top width of the channel (m or ft)

A rough turbulent zone check is performed on the water flowing in the channel using the Reynolds number (Re). The Re equation follows:

$$Re = \frac{\rho R V}{\mu}$$

Re Reynolds number (dimensionless)

ρ density (kg/m^3 or slug/ft^3)

R the hydraulic radius (m or ft)

V the velocity (m/s or ft/s)

μ dynamic viscosity (* 10^-3 kg/m*s or * 10^-5 lb*s/ft^2)

A critical flow check is performed on the water flowing in the channel using the Froude number (Fr). The Fr equation follows:

$$Fr = \frac{V}{\left(\sqrt{g * D}\right)}$$

Fr the Froude number (dimensionless)

V the velocity (m/s or ft/s)

g gravitational acceleration (m/s^2 or ft/sec^2)

D the hydraulic depth (m or ft)

Value

the missing parameter (Q, n, or Sf) & theta, area (A), wetted perimeter (P), velocity (V), top width (B), hydraulic radius (R), Reynolds number (Re), and Froude number (Fr) as a `list` for the Manningcirc function.

the missing parameter (d or y) & theta, area (A), wetted perimeter (P), top width (B), velocity (V), and hydraulic radius (R) as a `list` for the Manningcircy function.

Note

Assumptions: uniform flow, prismatic channel, and surface water temperature of 20 degrees Celsius (68 degrees Fahrenheit) at atmospheric pressure

Note: Units must be consistent

Source

r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See `https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot`.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 2, 8, 36, 102, 120, 123-125, 153-154.
2. Dan Moore, P.E., NRCS Water Quality and Quantity Technology Development Team, Portland Oregon, "Using Mannings Equation with Natural Streams", August 2011, `http://www.wcc.nrcs.usda.gov/ftpref/wntsc/H&H/xsec/manningsNaturally.pdf`.
3. Gilberto E. Urroz, Utah State University Civil and Environmental Engineering - OCW, CEE6510 - Numerical Methods in Civil Engineering, Spring 2006 (2006). Course 3. "Solving selected equations and systems of equations in hydraulics using Matlab", August/September 2004, `https://digitalcommons.usu.edu/ocw_cee/3`.
4. Tyler G. Hicks, P.E., *Civil Engineering Formulas: Pocket Guide*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2002, page 423, 425.
5. Wikimedia Foundation, Inc. Wikipedia, 26 November 2015, “Manning formula”, `https://en.wikipedia.org/wiki/Manning_formula`.
6. John C. Crittenden, R. Rhodes Trussell, David W. Hand, Kerry J. Howe, George Tchobanoglous, *MWH’s Water Treatment: Principles and Design*, Third Edition, Hoboken, New Jersey: John Wiley & Sons, Inc., 2012, page 1861-1862.
7. Andrew Chadwick, John Morfett and Martin Borthwick, *Hydraulics in Civil and Environmental Engineering*, Fourth Edition, New York City, New York: Spon Press, Inc., 2004, page 133.
8. Robert L. Mott and Joseph A. Untener, *Applied Fluid Mechanics*, Seventh Edition, New York City, New York: Pearson, 2015, page 376, 377-378, 392.
9. Wikimedia Foundation, Inc. Wikipedia, 17 March 2017, “Gravitational acceleration”, `https://en.wikipedia.org/wiki/Gravitational_acceleration`.
10. Wikimedia Foundation, Inc. Wikipedia, 29 May 2016, “Conversion of units”, `https://en.wikipedia.org/wiki/Conversion_of_units`.

See Also

`Manningtrap` for a trapezoidal cross-section, `Manningrect` for a rectangular cross-section, `Manningtri` for a triangular cross-section, and `Manningpara` for a parabolic cross-section.

Examples

```
library("iemisc")
library(iemiscdata)

# Practice Problem 14.12 from Mott (page 392)
y <- Manningcircy(y_d = 0.5, d = 6, units = "Eng")

# See npartfull in iemiscdata for the Manning's n table that the
# following example uses
# Use the normal Manning's n value for 1) Corrugated Metal, 2) Stormdrain.

data(npartfull)

# We are using the culvert as a stormdrain in this problem
nlocation <- grep("Stormdrain",
npartfull$"Type of Conduit and Description")

n <- npartfull[nlocation, 3] # 3 for column 3 - Normal n

Manningcirc(d = 6, Sf = 1 / 500, n = n, y = y$y, units = "Eng")
# d = 6 ft, Sf = 1 / 500 ft/ft, n = 0.024, y = 3 ft, units = "Eng"
# This will solve for Q since it is missing and Q will be in ft^3/s



# Example Problem 14.2 from Mott (page 377-378)
y <- Manningcircy(y_d = 0.5, d = 200/1000, units = "SI")

# See npartfull in iemiscdata for the Manning's n table that the
# following example uses
# Use the normal Manning's n value for 1) Clay, 2) Common drainage tile.

data(npartfull)

nlocation <- grep("Common drainage tile",
npartfull$"Type of Conduit and Description")

n <- npartfull[nlocation, 3] # 3 for column 3 - Normal n

Manningcirc(Sf = 1/1000, n = n, y = y$y, d = 200/1000, units = "SI")
# Sf = 1/1000 m/m, n = 0.013, y = 0.1 m, d = 200/1000 m, units = SI units
# This will solve for Q since it is missing and Q will be in m^3/s



# Example 4.1 from Sturm (page 124-125)
Manningcircy(y_d = 0.8, d = 2, units = "Eng")

y <- Manningcircy(y_d = 0.8, d = 2, units = "Eng")
# defines all list values within the object named y

y$y # gives the value of y
```

```
# Modified Exercise 4.1 from Sturm (page 153)
# Note: The Q in Exercise 4.1 is actually found using the Chezy equation,
# this is a modification of that problem
# See nchannel in iemiscdata for the Manning's n table that the
# following example uses
# Use the normal Manning's n value for 1) Natural streams - minor streams
# (top width at floodstage < 100 ft), 2) Mountain streams, no vegetation
# in channel, banks usually steep, trees and brush along banks submerged at
# high stages and 3) bottom: gravels, cobbles, and few boulders.

data(nchannel)

nlocation <- grep("bottom: gravels, cobbles, and few boulders",
nchannel$"Type of Channel and Description")

n <- nchannel[nlocation, 3] # 3 for column 3 - Normal n

Manningcirc(Sf = 0.002, n = n, y = y$y, d = 2, units = "Eng")
# Sf = 0.002 ft/ft, n = 0.04, y = 1.6 ft, d = 2 ft, units = English units
# This will solve for Q since it is missing and Q will be in ft^3/s



# Modified Exercise 4.5 from Sturm (page 154)
library(NISTunits)

ysi <- NISTftTOmeter(y$y)

dsi <- NISTftTOmeter(2)

Manningcirc(Sf = 0.022, n = 0.023, y = ysi, d = dsi, units = "SI")
# Sf = 0.022 m/m, n = 0.023, y = 0.48768 m, d = 0.6096 m, units = SI units
# This will solve for Q since it is missing and Q will be in m^3/s
```

`Manningpara` *Parabolic cross-section for the Gauckler-Manning-Strickler equation*

Description

This function solves for one missing variable in the Gauckler-Manning- Strickler equation for a parabolic cross-section and uniform flow. The `uniroot` function is used to obtain the missing parameter.

Usage

```
Manningpara(
  Q = NULL,
  n = NULL,
  m = NULL,
  Sf = NULL,
  y = NULL,
  B1 = NULL,
  y1 = NULL,
  T = NULL,
  units = c("SI", "Eng")
)
```

Arguments

`Q`	numeric vector that contains the discharge value [m^3/s or ft^3/s], if known.
`n`	numeric vector that contains the Manning's roughness coefficient n, if known.
`m`	numeric vector that contains the "cross-sectional side slope of m:1 (horizontal:vertical)", if known.
`Sf`	numeric vector that contains the bed slope (m/m or ft/ft), if known.
`y`	numeric vector that contains the flow depth (m or ft), if known.
`B1`	numeric vector that contains the "bank-full width", if known.
`y1`	numeric vector that contains the "bank-full depth", if known.
`T`	numeric vector that contains the temperature (degrees C or degrees Fahrenheit), if known.
`units`	character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English units (United States Customary System in the United States and Imperial Units in the United Kingdom)]

Details

Gauckler-Manning-Strickler equation is expressed as

$$V = \frac{K_n}{n} R^{\frac{2}{3}} S^{\frac{1}{2}}$$

V the velocity (m/s or ft/s)

n Manning's roughness coefficient (dimensionless)

R the hydraulic radius (m or ft)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

This equation is also expressed as

$$Q = \frac{K_n}{n} \frac{A^{\frac{5}{3}}}{P^{\frac{2}{3}}} S^{\frac{1}{2}}$$

Q the discharge [m^3/s or ft^3/s (cfs)] is VA

n Manning's roughness coefficient (dimensionless)

P the wetted perimeter of the channel (m or ft)

A the cross-sectional area (m^2 or ft^2)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

Other important equations regarding the parabolic cross-section follow:

$$R = \frac{A}{P}$$

R the hydraulic radius (m or ft)

A the cross-sectional area (m^2 or ft^2)

P the wetted perimeter of the channel (m or ft)

$$A = \frac{2}{3}By$$

A the cross-sectional area (m^2 or ft^2)

y the flow depth (normal depth in this function) [m or ft]

B the top width of the channel (m or ft)

$$P = \left(\frac{B}{2}\right)\left[\sqrt{(1+x^2)} + \left(\frac{1}{x}\right)\ln\left(x + \sqrt{(1+x^2)}\right)\right]$$

P the wetted perimeter of the channel (m or ft)

B the top width of the channel (m or ft)

x 4y/b (dimensionless)

$$x = \frac{4y}{B}$$

x 4y/b (dimensionless)

B the top width of the channel (m or ft)

y the flow depth (normal depth in this function) [m or ft]

$$B = B_1\left(\sqrt{\frac{y}{y_1}}\right)$$

B the top width of the channel (m or ft)

y the flow depth (normal depth in this function) [m or ft]

B_1 the "bank-full width" (m or ft)

y_1 the "bank-full depth" (m or ft)

$$D = \frac{A}{B}$$

D the hydraulic depth (m or ft)

A the cross-sectional area (m^2 or ft^2)

B the top width of the channel (m or ft)

A rough turbulent zone check is performed on the water flowing in the channel using the Reynolds number (Re). The Re equation follows:

$$Re = \frac{\rho R V}{\mu}$$

Re Reynolds number (dimensionless)

ρ density (kg/m^3 or slug/ft^3)

R the hydraulic radius (m or ft)

V the velocity (m/s or ft/s)

μ dynamic viscosity (* 10^-3 kg/m*s or * 10^-5 lb*s/ft^2)

A critical flow check is performed on the water flowing in the channel using the Froude number (Fr). The Fr equation follows:

$$Fr = \frac{V}{(\sqrt{g * D})}$$

Fr the Froude number (dimensionless)

V the velocity (m/s or ft/s)

g gravitational acceleration (m/s^2 or ft/sec^2)

D the hydraulic depth (m or ft)

Value

the missing parameter (Q, n, m, Sf, B1, y1, or y) & area (A), wetted perimeter (P), velocity (V), top width (B), hydraulic radius (R), Reynolds number (Re), and Froude number (Fr) as a `list`.

Note

Assumptions: uniform flow, prismatic channel, and surface water temperature of 20 degrees Celsius (68 degrees Fahrenheit) at atmospheric pressure

Note: Units must be consistent

Source

r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 2, 8, 36, 102, 120, 153.
2. Dan Moore, P.E., NRCS Water Quality and Quantity Technology Development Team, Portland Oregon, "Using Mannings Equation with Natural Streams", August 2011, http://www.wcc.nrcs.usda.gov/ftpref/wntsc/H&H/xsec/manningsNaturally.pdf.
3. Gilberto E. Urroz, Utah State University Civil and Environmental Engineering - OCW, CEE6510 - Numerical Methods in Civil Engineering, Spring 2006 (2006). Course 3. "Solving selected equations and systems of equations in hydraulics using Matlab", August/September 2004, https://digitalcommons.usu.edu/ocw_cee/3.
4. Tyler G. Hicks, P.E., *Civil Engineering Formulas: Pocket Guide*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2002, page 423, 425.
5. Wikimedia Foundation, Inc. Wikipedia, 26 November 2015, "Manning formula", https://en.wikipedia.org/wiki/Manning_formula.
6. John C. Crittenden, R. Rhodes Trussell, David W. Hand, Kerry J. Howe, George Tchobanoglous, *MWH's Water Treatment: Principles and Design*, Third Edition, Hoboken, New Jersey: John Wiley & Sons, Inc., 2012, page 1861-1862.
7. Andrew Chadwick, John Morfett and Martin Borthwick, *Hydraulics in Civil and Environmental Engineering*, Fourth Edition, New York City, New York: Spon Press, Inc., 2004, page 133.
8. Robert L. Mott and Joseph A. Untener, *Applied Fluid Mechanics*, Seventh Edition, New York City, New York: Pearson, 2015, page 376.
9. Wikimedia Foundation, Inc. Wikipedia, 17 March 2017, "Gravitational acceleration", https://en.wikipedia.org/wiki/Gravitational_acceleration.
10. Wikimedia Foundation, Inc. Wikipedia, 29 May 2016, "Conversion of units", https://en.wikipedia.org/wiki/Conversion_of_units.

See Also

Manningtrap for a trapezoidal cross-section, Manningrect for a rectangular cross-section, Manningtri for a triangular cross-section, and Manningcirc for a circular cross-section.

Examples

```
library("iemisc")
# Exercise 4.3 from Sturm (page 153)
y <- Manningpara(Q = 12.0, B1 = 10, y1 = 2.0, Sf = 0.005, n = 0.05, units = "SI")
# defines all list values within the object named y
# Q = 12.0 m^3/s, B1 = 10 m, y1 = 2.0 m, Sf = 0.005 m/m, n = 0.05, units = SI units
# This will solve for y since it is missing and y will be in m

y$y # gives the value of y


# Modified Exercise 4.3 from Sturm (page 153)
Manningpara(y = y$y, B1 = 10, y1 = 2.0, Sf = 0.005, n = 0.05, units = "SI")
# y = 1.254427 m, B1 = 10 m, y1 = 2.0 m, Sf = 0.005 m/m, n = 0.05, units = SI units
```

```
# This will solve for Q since it is missing and Q will be in m^3/s
```

`Manningrect` *Rectangular cross-section for the Gauckler-Manning-Strickler equation*

Description

This function solves for one missing variable in the Gauckler-Manning- Strickler equation for a rectangular cross-section and uniform flow. The uniroot function is used to obtain the missing parameter.

Usage

```
Manningrect(
  Q = NULL,
  n = NULL,
  b = NULL,
  Sf = NULL,
  y = NULL,
  T = NULL,
  units = c("SI", "Eng")
)
```

Arguments

Q	numeric vector that contains the discharge value [m^3/s or ft^3/s], if known.
n	numeric vector that contains the Manning's roughness coefficient n, if known.
b	numeric vector that contains the bottom width, if known.
Sf	numeric vector that contains the bed slope (m/m or ft/ft), if known.
y	numeric vector that contains the flow depth (m or ft), if known.
T	numeric vector that contains the temperature (degrees C or degrees Fahrenheit), if known.
units	character vector that contains the system of units [options are SI for International System of Units and Eng for English units (United States Customary System in the United States and Imperial Units in the United Kingdom)]

Details

Gauckler-Manning-Strickler equation is expressed as

$$V = \frac{K_n}{n} R^{\frac{2}{3}} S^{\frac{1}{2}}$$

V the velocity (m/s or ft/s)

n Manning's roughness coefficient (dimensionless)

R the hydraulic radius (m or ft)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

This equation is also expressed as

$$Q = \frac{K_n}{n} \frac{A^{\frac{5}{3}}}{P^{\frac{2}{3}}} S^{\frac{1}{2}}$$

Q the discharge [m^3/s or ft^3/s (cfs)] is VA

n Manning's roughness coefficient (dimensionless)

P the wetted perimeter of the channel (m or ft)

A the cross-sectional area (m^2 or ft^2)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

Other important equations regarding the rectangular cross-section follow:

$$R = \frac{A}{P}$$

R the hydraulic radius (m or ft)

A the cross-sectional area (m^2 or ft^2)

P the wetted perimeter of the channel (m or ft)

$$A = by$$

A the cross-sectional area (m^2 or ft^2)

y the flow depth (normal depth in this function) [m or ft]

b the bottom width (m or ft)

$$P = b + 2y$$

P the wetted perimeter of the channel (m or ft)

y the flow depth (normal depth in this function) [m or ft]

b the bottom width (m or ft)

$$B = b$$

B the top width of the channel (m or ft)

b the bottom width (m or ft)

$$D = \frac{A}{B}$$

D the hydraulic depth (m or ft)

A the cross-sectional area (m^2 or ft^2)

B the top width of the channel (m or ft)

A rough turbulent zone check is performed on the water flowing in the channel using the Reynolds number (Re). The Re equation follows:

$$Re = \frac{\rho RV}{\mu}$$

Re Reynolds number (dimensionless)

ρ density (kg/m^3 or slug/ft^3)

R the hydraulic radius (m or ft)

V the velocity (m/s or ft/s)

μ dynamic viscosity (* 10^-3 kg/m*s or * 10^-5 lb*s/ft^2)

A critical flow check is performed on the water flowing in the channel using the Froude number (Fr). The Fr equation follows:

$$Fr = \frac{V}{\left(\sqrt{g * D}\right)}$$

Fr the Froude number (dimensionless)

V the velocity (m/s or ft/s)

g gravitational acceleration (m/s^2 or ft/sec^2)

D the hydraulic depth (m or ft)

Value

the missing parameter (Q, n, b, Sf, or y) & area (A), wetted perimeter (P), velocity (V), top width (B), hydraulic radius (R), Reynolds number (Re), and Froude number (Fr) as a `list`.

Note

Assumptions: uniform flow, prismatic channel, and surface water temperature of 20 degrees Celsius (68 degrees Fahrenheit) at atmospheric pressure

Note: Units must be consistent

Source

r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 2, 8, 36, 102, 120, 153-154.
2. Dan Moore, P.E., NRCS Water Quality and Quantity Technology Development Team, Portland Oregon, "Using Mannings Equation with Natural Streams", August 2011, http://www.wcc.nrcs.usda.gov/ftpref/wntsc/H&H/xsec/manningsNaturally.pdf.
3. Gilberto E. Urroz, Utah State University Civil and Environmental Engineering - OCW, CEE6510 - Numerical Methods in Civil Engineering, Spring 2006 (2006). Course 3. "Solving selected equations and systems of equations in hydraulics using Matlab", August/September 2004, https://digitalcommons.usu.edu/ocw_cee/3.
4. Tyler G. Hicks, P.E., *Civil Engineering Formulas: Pocket Guide*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2002, page 423, 425.
5. Wikimedia Foundation, Inc. Wikipedia, 26 November 2015, “Manning formula”, https://en.wikipedia.org/wiki/Manning_formula.
6. John C. Crittenden, R. Rhodes Trussell, David W. Hand, Kerry J. Howe, George Tchobanoglous, *MWH’s Water Treatment: Principles and Design*, Third Edition, Hoboken, New Jersey: John Wiley & Sons, Inc., 2012, page 1861-1862.
7. Andrew Chadwick, John Morfett and Martin Borthwick, *Hydraulics in Civil and Environmental Engineering*, Fourth Edition, New York City, New York: Spon Press, Inc., 2004, page 133.
8. Robert L. Mott and Joseph A. Untener, *Applied Fluid Mechanics*, Seventh Edition, New York City, New York: Pearson, 2015, page 376, 379-380.
9. Wikimedia Foundation, Inc. Wikipedia, 17 March 2017, “Gravitational acceleration”, https://en.wikipedia.org/wiki/Gravitational_acceleration.
10. Wikimedia Foundation, Inc. Wikipedia, 29 May 2016, “Conversion of units”, https://en.wikipedia.org/wiki/Conversion_of_units.

See Also

`Manningtrap` for a trapezoidal cross-section, `Manningtri` for a triangular cross-section, `Manningpara` for a parabolic cross-section, and `Manningcirc` for a circular cross-section.

Examples

```
library("iemisc")
library(iemiscdata)
# Example Problem 14.4 from Mott (page 379)
# See nchannel in iemiscdata for the Manning's n table that the following
# example uses
# Use the normal Manning's n value for 1) Natural streams - minor streams
# (top width at floodstage < 100 ft), 2) Lined or Constructed Channels,
# 3) Concrete, and 4) unfinished.

data(nchannel)

nlocation <- grep("unfinished", nchannel$"Type of Channel and Description")
```

```
n <- nchannel[nlocation, 3] # 3 for column 3 - Normal n

Manningrect(Q = 5.75, b = (4.50) ^ (3 / 8), Sf = 1.2/100, n = n, units =
"SI")
# Q = 5.75 m^3/s, b = (4.50) ^ (3 / 8) m, Sf = 1.2 percent m/m, n = 0.017,
# units = SI units
# This will solve for y since it is missing and y will be in m



# Example Problem 14.5 from Mott (page 379-380)
# See nchannel in iemiscdata for the Manning's n table that the following
# example uses
# Use the normal Manning's n value for 1) Natural streams - minor streams
# (top width at floodstage < 100 ft), 2) Lined or Constructed Channels,
# 3) Concrete, and 4) unfinished.

data(nchannel)

nlocation <- grep("unfinished", nchannel$"Type of Channel and Description")

n <- nchannel[nlocation, 3] # 3 for column 3 - Normal n

Manningrect(Q = 12, b = 2, Sf = 1.2/100, n = n, units = "SI")
# Q = 12 m^3/s, b = 2 m, Sf = 1.2 percent m/m, n = 0.017, units = SI
# units
# This will solve for y since it is missing and y will be in m
```

Manningtrap — *Trapezoidal cross-section for the Gauckler-Manning-Strickler equation*

Description

This function solves for one missing variable in the Gauckler-Manning- Strickler equation for a trapezoidal cross-section and uniform flow. The uniroot function is used to obtain the missing parameter.

Usage

```
Manningtrap(
  Q = NULL,
  n = NULL,
  m = NULL,
  m1 = NULL,
  m2 = NULL,
  Sf = NULL,
```

```
  y = NULL,
  b = NULL,
  T = NULL,
  units = c("SI", "Eng"),
  type = c("symmetrical", "non-symmetrical")
)
```

Arguments

`Q`	numeric vector that contains the discharge value [m^3/s or ft^3/s], if known.
`n`	numeric vector that contains the Manning's roughness coefficient n, if known.
`m`	numeric vector that contains the "cross-sectional side slope of m:1 (horizontal:vertical)", if known.
`m1`	numeric vector that contains the "cross-sectional side slope of m1:1 (horizontal:vertical)", if known.
`m2`	numeric vector that contains the "cross-sectional side slope of m2:1 (horizontal:vertical)", if known.
`Sf`	numeric vector that contains the bed slope (m/m or ft/ft), if known.
`y`	numeric vector that contains the flow depth (m or ft), if known.
`b`	numeric vector that contains the bottom width, if known.
`T`	numeric vector that contains the temperature (degrees C or degrees Fahrenheit), if known.
`units`	character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English units (United States Customary System in the United States and Imperial Units in the United Kingdom)]
`type`	character vector that contains the type of trapezoid (symmetrical or non-symmetrical). The symmetrical trapezoid uses `m` while the non- symmetrical trapezoid uses `m1` and `m2`.

Details

Gauckler-Manning-Strickler equation is expressed as

$$V = \frac{K_n}{n} R^{\frac{2}{3}} S^{\frac{1}{2}}$$

V the velocity (m/s or ft/s)

n Manning's roughness coefficient (dimensionless)

R the hydraulic radius (m or ft)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

This equation is also expressed as

$$Q = \frac{K_n}{n} \frac{A^{\frac{5}{3}}}{P^{\frac{2}{3}}} S^{\frac{1}{2}}$$

Q the discharge [m^3/s or ft^3/s (cfs)] is VA

n Manning's roughness coefficient (dimensionless)

P the wetted perimeter of the channel (m or ft)

A the cross-sectional area (m^2 or ft^2)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

Other important equations regarding the trapezoidal cross-section follow:

$$R = \frac{A}{P}$$

R the hydraulic radius (m or ft)

A the cross-sectional area (m^2 or ft^2)

P the wetted perimeter of the channel (m or ft)

$$A = y\left(b + my\right)$$

A the cross-sectional area (m^2 or ft^2)

y the flow depth (normal depth in this function) [m or ft]

m the horizontal side slope

b the bottom width (m or ft)

$$P = b + 2y\sqrt{\left(1 + m^2\right)}$$

P the wetted perimeter of the channel (m or ft)

y the flow depth (normal depth in this function) [m or ft]

m the horizontal side slope

b the bottom width (m or ft)

$$B = b + 2my$$

B the top width of the channel (m or ft)

y the flow depth (normal depth in this function) [m or ft]

m the horizontal side slope

b the bottom width (m or ft)

$$D = \frac{A}{B}$$

D the hydraulic depth (m or ft)

A the cross-sectional area (m^2 or ft^2)

B the top width of the channel (m or ft)

A rough turbulent zone check is performed on the water flowing in the channel using the Reynolds number (Re). The Re equation follows:

$$Re = \frac{\rho RV}{\mu}$$

Re Reynolds number (dimensionless)

ρ density (kg/m^3 or slug/ft^3)

R the hydraulic radius (m or ft)

V the velocity (m/s or ft/s)

μ dynamic viscosity (* 10^-3 kg/m*s or * 10^-5 lb*s/ft^2)

A critical flow check is performed on the water flowing in the channel using the Froude number (Fr). The Fr equation follows:

$$Fr = \frac{V}{(\sqrt{g * D})}$$

Fr the Froude number (dimensionless)

V the velocity (m/s or ft/s)

g gravitational acceleration (m/s^2 or ft/sec^2)

D the hydraulic depth (m or ft)

Value

the missing parameter (Q, n, b, m, Sf, or y) & area (A), wetted perimeter (P), velocity (V), top width (B), hydraulic depth (D), hydraulic radius (R), Reynolds number (Re), and Froude number (Fr) as a `list`.

Note

Assumptions: uniform flow, prismatic channel, and surface water temperature of 20 degrees Celsius (68 degrees Fahrenheit) at atmospheric pressure

Note: Units must be consistent

Source

r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 2, 8, 36, 102, 120, 153.
2. Dan Moore, P.E., NRCS Water Quality and Quantity Technology Development Team, Portland Oregon, "Using Mannings Equation with Natural Streams", August 2011, http://www.wcc.nrcs.usda.gov/ftpref/wntsc/H&H/xsec/manningsNaturally.pdf.
3. Gilberto E. Urroz, Utah State University Civil and Environmental Engineering - OCW, CEE6510 - Numerical Methods in Civil Engineering, Spring 2006 (2006). Course 3. "Solving selected equations and systems of equations in hydraulics using Matlab", August/September 2004, https://digitalcommons.usu.edu/ocw_cee/3.
4. Tyler G. Hicks, P.E., *Civil Engineering Formulas: Pocket Guide*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2002, page 423, 425.
5. Andrew Chadwick, John Morfett, and Martin Borthwick, *Hydraulics in Civil and Environmental Engineering*, Fourth Edition, New York City, New York: Spon Press, 2004, pages 132-133.
6. Wikimedia Foundation, Inc. Wikipedia, 26 November 2015, “Manning formula”, https://en.wikipedia.org/wiki/Manning_formula.
7. John C. Crittenden, R. Rhodes Trussell, David W. Hand, Kerry J. Howe, George Tchobanoglous, *MWH’s Water Treatment: Principles and Design*, Third Edition, Hoboken, New Jersey: John Wiley & Sons, Inc., 2012, page 1861-1862.
8. Robert L. Mott and Joseph A. Untener, *Applied Fluid Mechanics*, Seventh Edition, New York City, New York: Pearson, 2015, page 376, 392.
9. Wikimedia Foundation, Inc. Wikipedia, 17 March 2017, “Gravitational acceleration”, https://en.wikipedia.org/wiki/Gravitational_acceleration.
10. Wikimedia Foundation, Inc. Wikipedia, 29 May 2016, “Conversion of units”, https://en.wikipedia.org/wiki/Conversion_of_units.

See Also

Manningrect for a rectangular cross-section, Manningtri for a triangular cross-section, Manningpara for a parabolic cross-section, and Manningcirc for a circular cross-section.

Examples

```
library("iemisc")
library(iemiscdata)
# Exercise 4.1 from Sturm (page 153)

Manningtrap(Q = 3000, b = 40, m = 3, Sf = 0.002, n = 0.025, units = "Eng")
# Q = 3000 cfs, b = 40 ft, m = 3, Sf = 0.002 ft/ft, n = 0.025,
# units = English units
# This will solve for y since it is missing and y will be in ft



# Practice Problem 14.19 from Mott (page 392)
# See nchannel in iemiscdata for the Manning's n table that the following
```

```
# example uses
# Use the minimum Manning's n value for 1) Natural streams - minor streams
# (top width at floodstage < 100 ft), 2) Lined or Constructed Channels,
# 3) Concrete and 4) float finish.

data(nchannel)

nlocation <- grep("float finish",
nchannel$"Type of Channel and Description")

n <- nchannel[nlocation, 3][1] # 3 for column 3 - Normal n

Manningtrap(y = 1.5, b = 3, m = 3/2, Sf = 0.1/100, n = n, units = "SI")
# y = 1.5 m, b = 3 m, m = 3/2, Sf = 0.1/100 m/m, n = 0.023, units = SI
# units
# This will solve for Q since it is missing and Q will be in m^3/s
```

`Manningtri` *Triangular cross-section for the Gauckler-Manning-Strickler equation*

Description

This function solves for one missing variable in the Gauckler-Manning- Strickler equation for a triangular cross-section and uniform flow. The `uniroot` function is used to obtain the missing parameter.

Usage

```
Manningtri(
  Q = NULL,
  n = NULL,
  m = NULL,
  Sf = NULL,
  y = NULL,
  T = NULL,
  units = c("SI", "Eng")
)
```

Arguments

`Q`	numeric vector that contains the discharge value [m^3/s or ft^3/s], if known.
`n`	numeric vector that contains the Manning's roughness coefficient n, if known.
`m`	numeric vector that contains the "cross-sectional side slope of m:1 (horizontal:vertical)", if known.
`Sf`	numeric vector that contains the bed slope (m/m or ft/ft), if known.

`y` numeric vector that contains the flow depth (m or ft), if known.

`T` numeric vector that contains the temperature (degrees C or degrees Fahrenheit), if known.

`units` character vector that contains the system of units [options are `SI` for International System of Units and `Eng` for English units (United States Customary System in the United States and Imperial Units in the United Kingdom)]

Details

Gauckler-Manning-Strickler equation is expressed as

$$V = \frac{K_n}{n} R^{\frac{2}{3}} S^{\frac{1}{2}}$$

V the velocity (m/s or ft/s)

n Manning's roughness coefficient (dimensionless)

R the hydraulic radius (m or ft)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

This equation is also expressed as

$$Q = \frac{K_n}{n} \frac{A^{\frac{5}{3}}}{P^{\frac{2}{3}}} S^{\frac{1}{2}}$$

Q the discharge [m^3/s or ft^3/s (cfs)] is VA

n Manning's roughness coefficient (dimensionless)

P the wetted perimeter of the channel (m or ft)

A the cross-sectional area (m^2 or ft^2)

S the slope of the channel bed (m/m or ft/ft)

K_n the conversion constant – 1.0 for SI and 3.2808399 ^ (1 / 3) for English units – m^(1/3)/s or ft^(1/3)/s

Other important equations regarding the triangular cross-section follow:

$$R = \frac{A}{P}$$

R the hydraulic radius (m or ft)

A the cross-sectional area (m^2 or ft^2)

P the wetted perimeter of the channel (m or ft)

$$A = my^2$$

A the cross-sectional area (m^2 or ft^2)

y the flow depth (normal depth in this function) [m or ft]

m the horizontal side slope

$$P = 2y\sqrt{(1 + m^2)}$$

P the wetted perimeter of the channel (m or ft)

y the flow depth (normal depth in this function) [m or ft]

m the horizontal side slope

$$B = 2my$$

B the top width of the channel (m or ft)

y the flow depth (normal depth in this function) [m or ft]

m the horizontal side slope

$$D = \frac{A}{B}$$

D the hydraulic depth (m or ft)

A the cross-sectional area (m^2 or ft^2)

B the top width of the channel (m or ft)

A rough turbulent zone check is performed on the water flowing in the channel using the Reynolds number (Re). The Re equation follows:

$$Re = \frac{\rho R V}{\mu}$$

Re Reynolds number (dimensionless)

ρ density (kg/m^3 or slug/ft^3)

R the hydraulic radius (m or ft)

V the velocity (m/s or ft/s)

μ dynamic viscosity (* 10^-3 kg/m*s or * 10^-5 lb*s/ft^2)

A critical flow check is performed on the water flowing in the channel using the Froude number (Fr). The Fr equation follows:

$$Fr = \frac{V}{(\sqrt{g * D})}$$

Fr the Froude number (dimensionless)

V the velocity (m/s or ft/s)

g gravitational acceleration (m/s^2 or ft/sec^2)

D the hydraulic depth (m or ft)

Value

the missing parameter (Q, n, m, Sf, or y) & area (A), wetted perimeter (P), velocity (V), top width (B), hydraulic radius (R), Reynolds number (Re), and Froude number (Fr) as a `list`.

Note

Assumptions: uniform flow, prismatic channel, and surface water temperature of 20 degrees Celsius (68 degrees Fahrenheit) at atmospheric pressure

Note: Units must be consistent

Source

r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See `https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot`.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 2, 8, 36, 102, 120, 153-154.
2. Dan Moore, P.E., NRCS Water Quality and Quantity Technology Development Team, Portland Oregon, "Using Mannings Equation with Natural Streams", August 2011, `http://www.wcc.nrcs.usda.gov/ftpref/wntsc/H&H/xsec/manningsNaturally.pdf`.
3. Gilberto E. Urroz, Utah State University Civil and Environmental Engineering - OCW, CEE6510 - Numerical Methods in Civil Engineering, Spring 2006 (2006). Course 3. "Solving selected equations and systems of equations in hydraulics using Matlab", August/September 2004, `https://digitalcommons.usu.edu/ocw_cee/3`.
4. Tyler G. Hicks, P.E., *Civil Engineering Formulas: Pocket Guide*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2002, page 423, 425.
5. Wikimedia Foundation, Inc. Wikipedia, 26 November 2015, “Manning formula”, `https://en.wikipedia.org/wiki/Manning_formula`.
6. John C. Crittenden, R. Rhodes Trussell, David W. Hand, Kerry J. Howe, George Tchobanoglous, *MWH’s Water Treatment: Principles and Design*, Third Edition, Hoboken, New Jersey: John Wiley & Sons, Inc., 2012, page 1861-1862.
7. Andrew Chadwick, John Morfett and Martin Borthwick, *Hydraulics in Civil and Environmental Engineering*, Fourth Edition, New York City, New York: Spon Press, Inc., 2004, page 133.
8. Robert L. Mott and Joseph A. Untener, *Applied Fluid Mechanics*, Seventh Edition, New York City, New York: Pearson, 2015, page 376, 393.
9. Wikimedia Foundation, Inc. Wikipedia, 17 March 2017, “Gravitational acceleration”, `https://en.wikipedia.org/wiki/Gravitational_acceleration`.
10. Wikimedia Foundation, Inc. Wikipedia, 29 May 2016, “Conversion of units”, `https://en.wikipedia.org/wiki/Conversion_of_units`.

See Also

`Manningtrap` for a trapezoidal cross-section, `Manningrect` for a rectangular cross-section, `Manningpara` for a parabolic cross-section, and `Manningcirc` for a circular cross-section.

Examples

```
library("iemisc")
library(iemiscdata)
# Practice Problem 14.41 from Mott (page 393)
# See nchannel in iemiscdata for the Manning's n table that the
# following example uses
# Use the normal Manning's n value for 1) Natural streams - minor streams
# (top width at floodstage < 100 ft), 2) Excavated or Dredged Channels, 3)
# Earth, straight, and uniform, 4) clean, recently completed.

data(nchannel)

nlocation <- grep("clean, recently completed",
nchannel$"Type of Channel and Description")

n <- nchannel[nlocation, 3] # 3 for column 3 - Normal n

Manningtri(Q = 0.68, m = 1.5, Sf = 0.0023, n = n, units = "Eng")
# Q = 0.68 cfs, m = 1.5, Sf = 0.002 ft/ft, n = 0.05, units = English units
# This will solve for y since it is missing and y will be in ft



# Modified Exercise 4.1 from Sturm (page 153)
Manningtri(Q = 3000, m = 3, Sf = 0.002, n = 0.025, units = "Eng")
# Q = 3000 cfs, m = 3, Sf = 0.002 ft/ft, n = 0.025, units = English units
# This will solve for y since it is missing and y will be in ft



# Modified Exercise 4.5 from Sturm (page 154)
Manningtri(Q = 950, m = 2, Sf = 0.022, n = 0.023, units = "SI")
# Q = 950 m^3/s, m = 2, Sf = 0.022 m/m, n = 0.023, units = SI units
# This will solve for y since it is missing and y will be in m
```

---

n *Manning's n for natural channels*

Description

This function computes Manning's n for natural channels.

Usage

```
n(nb = NULL, n1 = NULL, n2 = NULL, n3 = NULL, n4 = NULL, m = NULL)
```

Arguments

nb	numeric vector that contains "the base value for a straight, uniform channel", if needed
n1	numeric vector that contains "correction for surface irregularities", if needed
n2	numeric vector that contains "correction for variations in the shape and size of the cross section", if needed
n3	numeric vector that contains "correction for obstructions", if needed
n4	numeric vector that contains "correction for vegetation and flow conditions", if needed
m	numeric vector that contains "correction factor for channel meandering", if needed

Details

"Roughness values for channels and flood plains should be determined separately. The composition, physical shape, and vegetation of a flood plain can be quite different from those of a channel." Source: USGS.

The equation to find Manning's n for natural channels is expressed as

$$n = (n_b + n_1 + n_2 + n_3 + n_4)\, m$$

n Manning's n

n_b "the base value for a straight, uniform channel"

n_1 "correction for surface irregularities"

n_2 "correction for variations in the shape and size of the cross section"

n_3 "correction for obstructions"

n_4 "correction for vegetation and flow conditions"

m "correction factor for channel meandering"

Source: Sturm page 114.

Value

n as Manning's n for a natural channel as a numeric vector.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 114.
2. Guide for Selecting Manning's Roughness Coefficients for Natural Channels and Flood Plains, United States Geological Survey Water-supply Paper 2339 Metric Version
3. George J. Arcement, Jr., and Verne R. Schneider, United States Geological Survey Water-Supply Paper 2339, "Guide for Selecting Manning's Roughness Coefficients for Natural Channels and Flood Plains", 1989, http://pubs.usgs.gov/wsp/2339/report.pdf.

See Also

nc1 for Horton method for composite Manning's n, nc2 for Einstein and Banks method for composite Manning's n, nc3 for Lotter method for composite Manning's n, and nc4 for Krishnamurthy and Christensen method for composite Manning's n.

Examples

```
library("iemisc")
# Example from Table 4. from the USGS Reference text page 35
n(nb = 0.025, n4 = 0.005, m = 1.00)
```

---

na.interp1 *na.interp1*

Description

This function combines pracma's `interp1` constant interpolation method with zoo's `na.approx` linear interpolation method. Here, `x = x` rather than `x = index(object)` in na.approx. Here, `y = y` rather than `y = object` in na.approx. Also, here, `xi` is used instead of `xout` in na.approx. The Arguments list was obtained from both interp1 and na.approx.

Usage

```
na.interp1(x, y, xi = x, ..., na.rm = TRUE, maxgap = Inf)
```

Arguments

`x`	Numeric vector; points on the x-axis; at least two points required; will be sorted if necessary.
`y`	Numeric vector; values of the assumed underlying function; `x` and `y` must be of the same length.
`xi`	Numeric vector; points at which to compute the interpolation; all points must lie between `min(x)` and `max(x)`.
`...`	further arguments passed to methods. The `n` argument of `approx` is currently not supported.
`na.rm`	logical. If the result of the (`spline`) interpolation still results in `NA`s, should these be removed?
`maxgap`	maximum number of consecutive `NA`s to fill. Any longer gaps will be left unchanged. Note that all methods listed above can accept `maxgap` as it is ultimately passed to the default method.

Value

Numeric vector representing values at points `xi`.

Author(s)

Hans Werner Borchers (pracma interp1), Felix Andrews (zoo na.approx), Irucka Embry

Source

1. zoo's na.approx.R - modified on Fri Aug 6 00:26:22 2010 UTC by felix. See https://r-forge.r-project.org/scm/viewvc.php/pkg/zoo/R/na.approx.R?view=markup&revision=781&root=zoo.
2. pracma interp1 function definition - R package pracma created and maintained by Hans Werner Borchers. See interp1.

See Also

na.approx, interp1

Examples

```
library("iemisc")
library("data.table")

# zoo time series example
zoo1 <- structure(c(1.6, 1.7, 1.7, 1.7, 1.7, 1.7, 1.6, 1.7, 1.7, 1.7,
1.7, 1.7, 2, 2.1, 2.1, NA, NA, 2.1, 2.1, NA, 2.3, NA, 2, 2.1), .Dim = c(12L,
2L), .Dimnames = list(NULL, c("V1", "V2")), index = structure(c(1395242100,
1395243000, 1395243900, 1395244800, 1395245700, 1395256500, 1395257400,
1395258300, 1395259200, 1395260100, 1395261000, 1395261900), class =
c("POSIXct", "POSIXt"), tzone = "GMT"), class = "zoo")

zoo1 <- as.data.frame(zoo1) # to data.frame from zoo
zoo1[, "Time"] <- as.POSIXct(rownames(zoo1)) # create column named Time as a
# POSIXct class
zoo1 <- setDT(zoo1) # create data.table out of data.frame
setcolorder(zoo1, c(3, 1, 2)) # set the column order as the 3rd column
# followed by the 2nd and 1st columns
zoo1 <- setDF(zoo1) # return to data.frame

rowsinterps1 <- which(is.na(zoo1$V2 == TRUE))
# index of rows of zoo1 that have NA (to be interpolated)
xi <- as.numeric(zoo1[which(is.na(zoo1$V2 == TRUE)), 1])
# the Date-Times for V2 to be interpolated in numeric format
interps1 <- na.interp1(as.numeric(zoo1$Time), zoo1$V2, xi = xi,
na.rm = FALSE, maxgap = 1)
# the interpolated values where only gap sizes of 1 are filled
zoo1[rowsinterps1, 3] <- interps1
# replace the NAs in V2 with the interpolated V2 values
zoo1



# data frame time series example
df1 <- structure(list(Time = structure(c(1395242100, 1395243000, 1395243900,
```

```
 1395244800, 1395245700, 1395256500, 1395257400, 1395258300, 1395259200,
 1395260100, 1395261000, 1395261900), class = c("POSIXct", "POSIXt"),
 tzone = "GMT"), V1 = c(1.6, 1.7, 1.7, 1.7, 1.7, 1.7, 1.6, 1.7, 1.7, 1.7,
 1.7, 1.7), V2 = c(2, 2.1, 2.1, NA, NA, 2.1, 2.1, NA, 2.3, NA, 2, 2.1)),
 .Names = c("Time", "V1", "V2"), row.names = c(NA, -12L),
 class = "data.frame")

rowsinterps1 <- which(is.na(df1$V2 == TRUE))
# index of rows of df1 that have NA (to be interpolated)
xi <- as.numeric(df1[which(is.na(df1$V2 == TRUE)), 1])
# the Date-Times for V2 to be interpolated in numeric format
interps1 <- na.interp1(as.numeric(df1$Time), df1$V2, xi = xi,
 na.rm = FALSE, maxgap = 1)
# the interpolated values where only gap sizes of 1 are filled
df1[rowsinterps1, 3] <- interps1
# replace the NAs in V2 with the interpolated V2 values
df1



# data.table time series example
dt1 <- structure(list(Time = structure(c(1395242100, 1395243000, 1395243900,
 1395244800, 1395245700, 1395256500, 1395257400, 1395258300, 1395259200,
 1395260100, 1395261000, 1395261900), class = c("POSIXct", "POSIXt"),
 tzone = "GMT"), V1 = c(1.6, 1.7, 1.7, 1.7, 1.7, 1.7, 1.6, 1.7, 1.7, 1.7,
 1.7, 1.7), V2 = c(2, 2.1, 2.1, NA, NA, 2.1, 2.1, NA, 2.3, NA, 2, 2.1)),
 .Names = c("Time", "V1", "V2"), row.names = c(NA, -12L), class =
 c("data.table", "data.frame"), sorted = "Time")

rowsinterps2 <- which(is.na(dt1[, 3, with = FALSE] == TRUE))
# index of rows of x that have NA (to be interpolated)
xi <- as.numeric(dt1[rowsinterps2, Time])
# the Date-Times for V2 to be interpolated in numeric format
interps2 <- dt1[, na.interp1(as.numeric(Time), V2, xi = xi,
 na.rm = FALSE, maxgap = 1)]
# the interpolated values where only gap sizes of 1 are filled
dt1[rowsinterps2, `:=` (V2 = interps2)]
# replace the NAs in V2 with the interpolated V2 values
dt1
```

---

nc1 — *Horton method for composite Manning's n*

Description

This function computes the composite Manning's n using the Horton method.

Usage

```
nc1(P, n)
```

Arguments

P	numeric vector that contains "wetted perimeter of any section i"
n	numeric vector that contains "Manning's n of any section i"

Details

"A composite value of Manning's n for a single channel; that is, for the main channel only of a compound channel or a canal with laterally varying roughness." Source: Sturm page 118.

The equation to find Manning's composite n using the Horton method is

$$n_c = \left[\frac{\sum\limits_{i=1}^{N} P_i n_i^{\frac{3}{2}}}{P}\right]^{\frac{2}{3}}$$

n_c Manning's composite n

P "wetted perimeter of the entire cross section"

P_i "wetted perimeter of any section i"

n_i "Manning's n of any section i"

N "total number of sections into which the wetted perimeter is divided"

Source: Sturm page 118.

Value

numeric vector that contains nc1 as Manning's composite n.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 118.
2. Dan Moore, P.E., NRCS Water Quality and Quantity Technology Development Team, Portland Oregon, "Using Mannings Equation with Natural Streams", August 2011, `http://www.wcc.nrcs.usda.gov/ftpref/wntsc/H&H/xsec/manningsNaturally.pdf`.

See Also

`n` for Manning's n for natural channels, `nc2` for Einstein and Banks method for composite Manning's n, `nc3` for Lotter method for composite Manning's n, and `nc4` for Krishnamurthy and Christensen method for composite Manning's n.

Examples

```
library("iemisc")
# Example from the Moore Reference text
nc1(n = c(0.05, 0.035, 0.05, 0.04), P = c(22.22, 34.78, 2.00, 6.08))
```

---

nc2 *Einstein and Banks method for composite Manning's n*

Description

This function computes the composite Manning's n using the Einstein and Banks method.

Usage

```
nc2(P, n)
```

Arguments

P	numeric vector that contains "wetted perimeter of any section i"
n	numeric vector that contains "Manning's n of any section i"

Details

"A composite value of Manning's n for a single channel; that is, for the main channel only of a compound channel or a canal with laterally varying roughness." Source: Sturm page 118.

The equation to find Manning's composite n using the Einstein and Banks method is

$$n_c = \left[\frac{\sum\limits_{i=1}^{N} P_i n_i^2}{P}\right]^{\frac{1}{2}}$$

n_c Manning's composite n

P "wetted perimeter of the entire cross section"

P_i "wetted perimeter of any section i"

n_i "Manning's n of any section i"

N "total number of sections into which the wetted perimeter is divided"

Source: Sturm page 118.

Value

numeric vector that contains nc2 as Manning's composite n.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 118-119.
2. Dan Moore, P.E., NRCS Water Quality and Quantity Technology Development Team, Portland Oregon, "Using Mannings Equation with Natural Streams", August 2011, http://www.wcc.nrcs.usda.gov/ftpref/wntsc/H&H/xsec/manningsNaturally.pdf.

See Also

n for Manning's n for natural channels, nc1 for Horton method for composite Manning's n, nc3 for Lotter method for composite Manning's n, and nc4 for Krishnamurthy and Christensen method for composite Manning's n.

Examples

```
library("iemisc")
# Example from the Moore Reference text
nc2(n = c(0.05, 0.035, 0.05, 0.04), P = c(22.22, 34.78, 2.00, 6.08))
```

---

nc3 *Lotter method for composite Manning's n*

Description

This function computes the composite Manning's n using the Lotter method.

Usage

```
nc3(P, n, R)
```

Arguments

P	numeric vector that contains "wetted perimeter of any section i"
n	numeric vector that contains "Manning's n of any section i"
R	numeric vector that contains "hydraulic radius of any section i"

Details

"A composite value of Manning's n for a single channel; that is, for the main channel only of a compound channel or a canal with laterally varying roughness."

The equation to find Manning's composite n using the Lotter method is

$$n_c = \frac{PR^{\frac{5}{3}}}{\sum\limits_{i=1}^{N} \frac{P_i R_i^{\frac{5}{3}}}{n_i}}$$

n_c Manning's composite n

P "wetted perimeter of the entire cross section"

R "hydraulic radius of the entire cross section"

P_i "wetted perimeter of any section i"

R_i "hydraulic radius of any section i"

n_i "Manning's n of any section i"

N "total number of sections into which the wetted perimeter and hydraulic radius are divided"

Value

numeric vector that contains nc3 as Manning's composite n.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 118-119.

See Also

`n` for Manning's n for natural channels, `nc1` for Horton method for composite Manning's n, `nc2` for Einstein and Banks method for composite Manning's n, and `nc4` for Krishnamurthy and Christensen method for composite Manning's n.

Examples

```
library("iemisc")
nc3(n = c(0.0024, 0.035), P = c(23.65, 36.08), R = c(2.02, 6.23))
```

`nc4` *Krishnamurthy and Christensen method for composite Manning's n*

Description

This function computes the composite Manning's n using the Krishnamurthy and Christensen method.

Usage

```
nc4(P, n, y)
```

Arguments

`P`	numeric vector that contains "wetted perimeter of any section i"
`n`	numeric vector that contains "Manning's n of any section i"
`y`	numeric vector that contains "flow depth in the ith section"

Details

"A composite value of Manning's n for a single channel; that is, for the main channel only of a compound channel or a canal with laterally varying roughness."

The equation to find Manning's composite n using the Krishnamurthy and Christensen method is

$$\ln n_c = \frac{\sum\limits_{i=1}^{N} P_i y_i^{\frac{3}{2}} \ln n_i}{\sum\limits_{i=1}^{N} P_i y_i^{\frac{3}{2}}}$$

n_c Manning's composite n

P_i "wetted perimeter of any section i"

y_i "flow depth in the ith section"

n_i "Manning's n of any section i"

N "total number of sections into which the wetted perimeter and hydraulic radius are divided"

Value

numeric vector that contains nc4 as Manning's composite n.

References

1. Terry W. Sturm, *Open Channel Hydraulics*, 2nd Edition, New York City, New York: The McGraw-Hill Companies, Inc., 2010, page 118-119.

See Also

`n` for Manning's n for natural channels, `nc1` for Horton method for composite Manning's n, `nc2` for Einstein and Banks method for composite Manning's n, and `nc3` for Lotter method for composite Manning's n.

Examples

```
library("iemisc")
nc4(n = c(0.0024, 0.035), P = c(23.65, 36.08), y = c(10.23, 7.38))
```

`ngivenPFi` *To Find i Given F, n, and P (Engineering Economics)*

Description

Compute n given P, F, and i

Usage

```
ngivenPFi(P, F, i)
```

Arguments

`P`	numeric vector that contains the present value(s)
`F`	numeric vector that contains the future value(s)
`i`	numeric vector that contains the interest rate(s) as a percent

Details

n is expressed as

$$n = \frac{\log\left(\frac{F}{P}\right)}{\log\left(1 + i\right)}$$

n the "number of interest periods"

F the "future equivalent"

P the "present equivalent"

i the "effective interest rate per interest period"

Value

n numeric vector that contains the period value(s)

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 129, 142.

Examples

```
library("iemisc")
# Example for equation 4-7 from the Reference text (page
ngivenPFi(P = 500, F = 1000, i = 15)
```

numel *Number of elements (GNU Octave/MATLAB compatible)*

Description

Obtain the number of elements of R objects [arrays, matrices, and vectors (including lists)] in a manner compatible with GNU Octave/MATLAB. Some documentation from `length`.

Usage

```
numel(x, ...)
```

Arguments

`x`	An R object (array, matrix, vector)
`...`	R objects (indices idx1, idx2, ...)

Value

"Return the number of elements in the R object x. Optionally, if indices idx1, idx2, ... are supplied, return the number of elements that would result from the indexing a(idx1, idx2, ...)." Source: Eaton page 41.

Author(s)

Irucka Embry, Samit Basu (FreeMat)

Source

1. r - Add a Column to a Dataframe From a List of Values - Stack Overflow answered by Matthew Plourde on Jun 21 2012. See https://stackoverflow.com/questions/11130037/add-a-column-to-a-dataframe 11130178.
2. r - Why does is.vector() return TRUE for list? - Stack Overflow answered by Andrie on May 17 2011. See https://stackoverflow.com/questions/6032772/why-does-is-vector-return-true-for-list/ 6032909.

References

1. Samit Basu (2002-2006). FreeMat v4.0, http://freemat.sourceforge.net/.
2. John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 41.

See Also

`numel`, `numel`, `size`, `length`

Examples

```
library("iemisc")
import::from(pracma, ones)
xx <- list(1:26, 1:10)
numel(xx)

# Examples from GNU Octave numel
a <- 1
b <- ones(2, 3)
numel(a, b)

a <- 2
b <- ones(2, 3)
c <- ones(3, 4)
numel(a, b)
numel(a, b, c)

f <- matrix(c(10, 12, 23, 21, 62, 93), nrow = 2, ncol = 3, byrow = TRUE)
g <- c(2, 4)
numel(f, g)


## Not run:
# check against GNU Octave
library(RcppOctave) # requires Octave (>= 3.2.4) and its development files
o_source(text = "
xx = {1:26, 1:10}

\
a = 1;
b = ones(2, 3);
numel(a, b)

a = 2;
b = ones(2, 3);
c = ones(3, 4);
numel(a, b)
numel(a, b, c)

f = [10 12 23; 21 62 93];
g = [2 4];
numel(f, g)
")

## End(Not run)
```

PgivenA *Present value given Annual value (Engineering Economics)*

Description

Compute P given A

Usage

```
PgivenA(
  A,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)

PA(
  A,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`A`	numeric vector that contains the annual value(s)
`n`	numeric vector that contains the period value(s)
`i`	numeric vector that contains the interest rate(s) as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

P is expressed as

$$P = A \left[\frac{\left(1 + i\right)^n - 1}{i \left(1 + i\right)^n}\right]$$

P the "present equivalent"

A the "uniform series amount (occurs at the end of each interest period)"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

PgivenA numeric vector that contains the present value(s) rounded to 2 decimal places

PA data.frame of both n (0 to n) and the resulting present values rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 133-134, 142, 164.

Examples

```
library("iemisc")
# Example 4-9 from the Reference text (page 133-134)
PgivenA(20000, 5, 15, "annual") # the interest rate is 15%

PA(20000, 5, 15, "annual") # the interest rate is 15%
```

PgivenA1 *Present value for geometric gradient series (Engineering Economics)*

Description

Compute P given A1

Usage

```
PgivenA1(A1, i, f, n)
```

Arguments

A1	numeric vector that contains the initial annual value(s)
i	numeric vector that contains the interest rate(s) as a percent
f	numeric vector that contains the average interest rate value(s) as a percent per period
n	numeric vector that contains the period value(s)

Details

P is expressed as

$$P = \frac{A_1 \left[1 - \left(1 + i\right)^{-n} \left(1 + f\right)^{n}\right]}{i - f}, \, where \; f \neq i$$

or

$$P = A_1 n \left(1 + i\right)^{-1}, \, where \; f = i$$

P "the present equivalent of the geometric gradient series"

A_1 "the initial cash flow in that occurs at the end of period one"

i the "interest rate per period"

f the "average rate each period"

n the "number of interest periods"

Note: "f can be positive or negative"

Value

PgivenA1 numeric vector that contains the present value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 156-159.

Examples

```
library("iemisc")
# Example 4-23 from the Reference text (page 158-159)
PgivenA1(1000, 25, 20, 4) # i is 25% and f is 20%


# Example 4-24 from the Reference text (page 159)
PgivenA1(1000, 25, -20, 4) # i is 25% and f is -20%
```

PgivenAcont — *Present value given Annual value [continuous] (Engineering Economics)*

Description

Compute P given A with interest compounded continuously

Usage

```
PgivenAcont(A, n, r)
```

Arguments

`A`	numeric vector that contains the annual value(s)
`n`	numeric vector that contains the period value(s)
`r`	numeric vector that contains the continuously compounded nominal annual interest rate(s) as a percent

Details

P is expressed as

$$P = A\left[\frac{e^{rn} - 1}{e^{rn}\left(e^{r} - 1\right)}\right]$$

P the "present equivalent"

A the "annual equivalent amount (occurs at the end of each year)"

r the "nominal annual interest rate, compounded continuously"

n the "number of periods (years)"

Value

PgivenAcont numeric vector that contains the present value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 169.

Examples

```
library("iemisc")
  PgivenAcont(2000, 3, 12) # the interest rate is 12%
```

PgivenF *Present value given Future value (Engineering Economics)*

Description

Compute P given F

Usage

```
PgivenF(
  F,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)

PF(
  F,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

F	numeric vector that contains the future value(s)
n	numeric vector that contains the period value(s)
i	numeric vector that contains the interest rate(s) as a percent
frequency	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

P is expressed as

$$P = F \left[\frac{1}{\left(1 + i\right)^n} \right]$$

P the "present equivalent"

F the "future equivalent"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

PgivenF numeric vector that contains the present value(s) rounded to 2 decimal places

PF data.frame of both n (0 to n) and the resulting present values rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 128, 142, 164.

Examples

```
library("iemisc")
# Example 4-4 from the Reference text (page 128)
PgivenF(10000, 6, 8, "annual") # the interest rate is 8%

PF(10000, 6, 8, "annual") # the interest rate is 8%
```

`PgivenFcont` *Present value given Future value [continuous] (Engineering Economics)*

Description

Compute P given F with interest compounded continuously

Usage

```
PgivenFcont(F, n, r)
```

Arguments

F	numeric vector that contains the future value(s)
n	numeric vector that contains the period value(s)
r	numeric vector that contains the continuously compounded nominal annual interest rate(s) as a percent

Details

P is expressed as

$$P = Fe^{-rn}$$

P the "present equivalent"

F the "future equivalent"

r the "nominal annual interest rate, compounded continuously"

n the "number of periods (years)"

Value

PgivenFcont numeric vector that contains the present value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 169.

Examples

```
library("iemisc")
PgivenFcont(1000, 9, 7) # the interest rate is 7%
```

PgivenFivary *"Present equivalent of a series of future cash flows subject to varying interest rates" (Engineering Economics)*

Description

Compute P given F and i that varies

Usage

```
PgivenFivary(Fn, ik, k)
```

Arguments

`Fn`	numeric vector that contains the future value(s) at the end of a period n
`ik`	numeric vector that contains the effective interest rate(s) per period as a percent for the kth period
`k`	numeric vector that contains the kth period values

Details

P is expressed as

$$P = \frac{F_n}{\prod\limits_{k=1}^{n} (1 + i_k)}$$

P the "present equivalent"

F_n the "future cash flows subject to varying interest rates"

i_k the "interest rate for the kth period"

k the "number of interest periods"

Value

PgivenFivary numeric vector that contains the present value(s)

Source

1. r - Add a Column to a Dataframe From a List of Values - Stack Overflow answered by Matthew Plourde on Jun 21 2012. See https://stackoverflow.com/questions/11130037/add-a-column-to-a-datafram 11130178.
2. r - Why does is.vector() return TRUE for list? - Stack Overflow answered by Andrie on May 17 2011. See https://stackoverflow.com/questions/6032772/why-does-is-vector-return-true-for-list/ 6032909.

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 142, 162.

Examples

```
library("iemisc")
# Example for equation 4-31 from the Reference text (page 162)
PgivenFivary(Fn = 1000, ik = c(10, 12, 13, 10), k = 1)
# i1 is 10%, i2 is 12%, i3 is 14%, and i4 is 10% & k = 1 year
```

`PgivenG` *Present value given Gradient value (Engineering Economics)*

Description

Compute P given G

Usage

```
PgivenG(
  G,
  n,
  i,
  frequency = c("annual", "semiannual", "quarter", "bimonth", "month", "daily")
)
```

Arguments

`G`	numeric vector that contains the gradient value(s)
`n`	numeric vector that contains the period value(s)
`i`	numeric vector that contains the interest rate(s) as a percent
`frequency`	character vector that contains the frequency used to obtain the number of periods [annual (1), semiannual (2), quarter (4), bimonth (6), month (12), daily (365)]

Details

$$P = G\left\{\frac{1}{i}\left[\frac{\left(1+i\right)^n - 1}{i\left(1+i\right)^n} - \frac{n}{\left(1+i\right)^n}\right]\right\}$$

P the "present equivalent"

G the "uniform gradient amount"

i the "effective interest rate per interest period"

n the "number of interest periods"

Value

PgivenG numeric vector that contains the present value(s) rounded to 2 decimal places

References

William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 142, 150, 152-154.

Examples

```
library("iemisc")
# Example 4-20 from the Reference text (pages 153-154)
PgivenG(1000, 4, 15, "annual") # the interest rate is 15%
```

`ranges` *Sample range*

Description

This function computes the sample range.

Usage

```
ranges(x, na.rm = FALSE, finite = FALSE)
```

Arguments

`x`	numeric vector
`na.rm`	logical vector that determines whether the missing values should be removed or not.
`finite`	logical vector that determines whether non-finite values should be removed or not.

Details

"The range is the difference between the largest number and the smallest number in the set." Source: Onwubiko page 176.

The following statements are from `range`:

"If na.rm is `FALSE`, `NA` and `NaN` values in any of the arguments will cause `NA` values to be returned, otherwise `NA` values are ignored."

"If finite is `TRUE`, the minimum and maximum of all finite values is computed, i.e., `finite` = `TRUE` includes `na.rm` = `TRUE`."

Value

ranges as the difference between the maximum and minimum values in `x` as a numeric `vector`. Unlike the `range`, ranges can't take character vectors as arguments, only numeric vectors.

References

Chinyere Onwubiko, *An Introduction to Engineering*, Mission, Kansas: Schroff Development Corporation, 1997, page 176.

See Also

`sgm` for geometric mean, `shm` for harmonic mean, `cv` for coefficient of variation (CV), `rms` for root-mean-square (RMS), `relerror` for relative error, and `approxerror` for approximate error.

Examples

```
library("iemisc")
require(stats)
set.seed(100) # makes the example reproducible
x <- rnorm(100)
ranges(x)
```

`relerror` *Relative error*

Description

This function computes the relative error.

Usage

```
relerror(xt, xa)
```

Arguments

`xt`	numeric vector that contains the true value(s)
`xa`	numeric vector that contains the approximate value(s)

Details

Relative error is expressed as

$$\varepsilon_t = \frac{true\ value - approximation}{true\ value} \cdot 100$$

ε_t the "true percent relative error"

true value the true value

approximation the approximate value

Value

relative error, as a percent (%), as a numeric `vector`.

References

Steven C. Chapra, *Applied Numerical Methods with MATLAB for Engineers and Scientists*, Second Edition, Boston, Massachusetts: McGraw-Hill, 2008, page 82-83.

See Also

`sgm` for geometric mean, `shm` for harmonic mean, `cv` for coefficient of variation (CV), `rms` for root-mean-square (RMS), `approxerror` for approximate error, and `ranges` for sample range.

Examples

```
library("iemisc")
# Example 4.1 from the Reference text (page 83)
relerror(1.648721, 1.5) # answer as a percent (\%)
```

`righttri` *Right triangle calculations*

Description

This function computes the missing length (must have at least 2 sides) and the interior angles (degrees) of a right triangle.

Usage

```
righttri(a = NULL, b = NULL, c = NULL)
```

Arguments

a	numeric vector that contains the known side a, if known
b	numeric vector that contains the known side b, if known
c	numeric vector that contains the known side c (hypotenuse), if known

Details

Side `a` is the side adjacent to angle B and opposite angle A. Side `b` is the side adjacent to angle A and opposite angle B. Side c (hypotenuse) is opposite the right angle (angle C).

This function makes the following calculations:

1. the length of the missing side using the Pythagorean theorem,
2. the area of the right triangle,
3. the altitude of the right triangle,
4. the angle associated with the side named a (degrees),
5. the angle associated with the side named b (degrees), and
6. the angle associated with the side named c (degrees).

Value

list of known sides a, b, and c & the interior angles A, B, and C (right angle), in degrees, if and only if the given sides create a right triangle.

Source

1. r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot.
2. r - switch() statement usage - Stack Overflow answered by Tommy on Oct 19 2011 and edited by Tommy on Mar 6 2012. See https://stackoverflow.com/questions/7825501/switch-statement-usage.
3. Using Switch Statement in R - Stack Overflow answered by Gavin Simpson on Jul 25 2013. See https://stackoverflow.com/questions/17847034/using-switch-statement-in-r.

References

1. r - Better error message for stopifnot? - Stack Overflow answered by Andrie on Dec 1 2011. See https://stackoverflow.com/questions/8343509/better-error-message-for-stopifnot.
2. Masoud Olia, Ph.D., P.E. and Contributing Authors, *Barron's FE (Fundamentals of Engineering Exam)*, 3rd Edition, Hauppauge, New York: Barron's Educational Series, Inc., 2015, page 44-45.
3. Wikimedia Foundation, Inc. Wikipedia, 28 December 2015, "Pythagorean theorem", https://en.wikipedia.org/wiki/Pythagorean_theorem.
4. Wikimedia Foundation, Inc. Wikipedia, 26 November 2015, "Radian", https://en.wikipedia.org/wiki/Radian.
5. Wikimedia Foundation, Inc. Wikipedia, 9 December 2015, "Right triangle", https://en.wikipedia.org/wiki/Right_triangle.

Examples

```
library("iemisc")
## Not run:
righttri(0, 2) # a = 0, b = 2

righttri(1, 2) # a = 1, b = 2

righttri(a = 5, c = 10)

righttri(a = 3, c = 5)

righttri(a = 5, c = 10)

## End(Not run)
```

`rms` *Root-mean-square*

Description

This function computes the sample root-mean-square (RMS).

Usage

```
rms(x, na.rm = FALSE)
```

Arguments

`x`	numeric vector that contains the sample data points.
`na.rm`	logical vector that determines whether the missing values should be removed or not.

Details

RMS is expressed as

$$x_{rms} = \sqrt{\frac{\sum\limits_{i=1}^{n} x_i^2}{n}}$$

x_rms the sample harmonic mean

x the values in a sample

n the number of values

Value

sample root-mean-square as a numeric vector. The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `rms(x, na.rm = TRUE)`.

References

Masoud Olia, Ph.D., P.E. and Contributing Authors, *Barron's FE (Fundamentals of Engineering Exam)*, 3rd Edition, Hauppauge, New York: Barron's Educational Series, Inc., 2015, page 84.

See Also

`sgm` for geometric mean, `shm` for harmonic mean, `cv` for coefficient of variation (CV), `relerror` for relative error, `approxerror` for approximate error, and `ranges` for sample range.

Examples

```
library("iemisc")
samp <- c(0.5, 100, 1000.25, 345, 0.0213, 0, 45, 99, 23, 11, 1, 89, 0, 34,
        65, 98, 3)
rms(samp)
```

---

secd *Secant (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of secant for each element of `x` in degrees in a manner compatible with GNU Octave/MATLAB.

Usage

```
secd(x)
```

Arguments

`x`	A numeric vector containing values in degrees

Value

The secant of each element of `x` in degrees.

Author(s)

David Bateman (GNU Octave secd), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave secd
secd (seq(0, 80, by = 10))

secd (c(0, 180, 360))

secd (c(90, 270))
```

sgm *Geometric mean*

Description

This function computes the sample geometric mean.

Usage

```
sgm(x, na.rm = FALSE)
```

Arguments

x	numeric vector that contains the sample data points (any negative values will be ignored).
na.rm	logical vector that determines whether the missing values should be removed or not.

Details

Geometric mean is expressed as

$$\bar{x}_g = (x_1 x_2 \cdots x_n)^{\frac{1}{n}}$$

$\bar{x}_g$ the sample geometric mean

x the values in a sample

n the number of positive values

"The geometric mean is used in averaging values that represent a rate of change. It is the positive nth root of the product of the n values."

Value

sample geometric mean as a numeric vector. The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `sgm(x, na.rm = TRUE)`.

References

Nathabandu T. Kottegoda and Renzo Rosso, *Statistics, Probability, and Reliability for Civil and Environmental Engineers*, New York City, New York: The McGraw-Hill Companies, Inc., 1997, page 13.

See Also

`mean` for arithmetic mean

`shm` for harmonic mean, `cv` for coefficient of variation (CV), `relerror` for relative error, `approxerror` for approximate error, `rms` for root-mean-square (RMS), and `ranges` for sample range.

Examples

```
library("iemisc")
# Example 1.13 from Kottegoda (page 13)
city_pop <- c(230000, 310000)
sgm(city_pop)

# Compare the geometric mean to the arithmetic mean
mean(city_pop)
```

---

shm — *Harmonic mean*

Description

This function computes the sample harmonic mean.

Usage

```
shm(x, na.rm = FALSE)
```

Arguments

`x`	numeric vector that contains the sample data points.
`na.rm`	logical vector that determines whether the missing values should be removed or not.

Details

Harmonic mean is expressed as

$$\bar{x}_h = \frac{1}{\left(\frac{1}{n}\right)\left[\left(\frac{1}{x_1}\right) + \left(\frac{1}{x_2}\right) + \cdots + \left(\frac{1}{x_n}\right)\right]}$$

$\bar{x}_h$ the sample harmonic mean

$\boldsymbol{x}$ the values in a sample

$\boldsymbol{n}$ the number of values

"The harmonic mean is the reciprocal of the mean of the reciprocals. It is applied in situations where the reciprocal of a variable is averaged."

Value

sample harmonic mean as a numeric vector. The default choice is that any NA values will be kept (`na.rm = FALSE`). This can be changed by specifying `na.rm = TRUE`, such as `shm(x, na.rm = TRUE)`.

References

Nathabandu T. Kottegoda and Renzo Rosso, *Statistics, Probability, and Reliability for Civil and Environmental Engineers*, New York City, New York: The McGraw-Hill Companies, Inc., 1997, page 13.

See Also

`mean` for arithmetic mean

`sgm` for geometric mean, `cv` for coefficient of variation (CV), `relerror` for relative error, `approxerror` for approximate error, `rms` for root-mean-square (RMS), and `ranges` for sample range.

Examples

```
library("iemisc")
library("data.table")

# Example 1.12 from Kottegoda (page 13)
x <- c(0.20, 0.24, 0.16) # stream velocities in m/s
shm(x)

# using a matrix of the numeric vector x
mat1 <- matrix(data = x, nrow = length(x), ncol = 1, byrow = FALSE,
        dimnames = list(c(rep("", length(x))), "Velocities"))
shm(mat1)


# using a data.frame of the numeric vector x
df1 <- data.frame(x)
shm(df1)


# using a data.table of the numeric vector x
df2 <- data.table(x)
shm(df2)
```

SimpIntPaid *Simple Interest Paid (Engineering Economics)*

Description

Computes the total amount paid at the end of n periods using simple interest

Usage

```
SimpIntPaid(P, n, i)
```

Arguments

P	numeric vector that contains the present value(s)
n	numeric vector that contains the period value(s)
i	numeric vector that contains the interest rate(s) as whole number or decimal

Details

Simple Interest is expressed as

$$I = Pni$$

$$S_n = P + I$$

or

$$S_n = P\left(1 + ni\right)$$

P the "principal amount (lent or borrowed)"

S_n the "total amount paid back"

I the "simple interest"

i the "interest rate per interest period"

n the "number of interest periods"

Value

SimpIntPaid numeric vector that contains the total amount paid at the end of n periods rounded to 2 decimal places

References

1. Chinyere Onwubiko, *An Introduction to Engineering*, Mission, Kansas: Schroff Development Corporation, 1997, page 205-206.
2. William G. Sullivan, Elin M. Wicks, and C. Patrick Koelling, *Engineering Economy*, Fourteenth Edition, Upper Saddle River, New Jersey: Pearson/Prentice Hall, 2009, page 116.

Examples

```
library("iemisc")
# Example for equation 4-1 from the Sullivan Reference text (page 116)
SimpIntPaid(1000, 3, 10) # the interest rate is 10%
```

`sind` *Sine (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of sine for each element of `x` in degrees in a manner compatible with GNU Octave/MATLAB. Zero is returned for any "elements where `x` / 180 is an integer." Source: Eaton.

Usage

```
sind(x)
```

Arguments

`x`	A numeric vector containing values in degrees

Value

The sine of each element of `x` in degrees. Zero for any "elements where `x` / 180 is an integer."

Author(s)

David Bateman (GNU Octave sind), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave sind
sind(seq(10, 90, by = 10))

sind(c(0, 180, 360))

sind(c(90, 270))
```

size — Size of R objects (GNU Octave/MATLAB compatible)

Description

Provides the dimensions of R objects in a manner compatible with GNU Octave/MATLAB. This function is the same as `size`, except this `size` can find the size of character vectors too. Some documentation from `size`.

Usage

```
size(x, k)
```

Arguments

`x`	An R object (array, vector, or matrix)
`k`	integer specifying a particular dimension

Value

"Return the number of rows and columns of the object x as a numeric vector. If given a second argument, `size` will return the size of the corresponding dimension." Source: Eaton.

Author(s)

Hans Werner Borchers (pracma size), Irucka Embry

Source

pracma size function definition - R package pracma created and maintained by Hans Werner Borchers. See `size`.

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL http://www.gnu.org/software/octave/doc/interpreter/. Page 42.

See Also

`dim`, `size`

Examples

```
library("iemisc")
library(gsubfn)


# Examples from GNU Octave size
object1 <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 3, ncol = 2, byrow = TRUE)
size(object1)


list[nr, nc] <- size(matrix(c(1, 2, 3, 4, 5, 6), nrow = 3, ncol = 2,
                byrow = TRUE))

size(matrix(c(1, 2, 3, 4, 5, 6), nrow = 3, ncol = 2, byrow = TRUE), 2)

# Examples from pracma size
size(1:8)

size(matrix(1:8, 2, 4))

size(matrix(1:8, 2, 4), 2)

size(matrix(1:8, 2, 4), 3)

ss <- "object"
size(ss)


## Not run:
# check against GNU Octave
library(RcppOctave) # requires Octave (>= 3.2.4) and its development files
o_source(text = "
\
object1 = [1, 2; 3, 4; 5, 6];
size(object1)

[nr, nc] = size([1, 2; 3, 4; 5, 6])

size([1, 2; 3, 4; 5, 6], 2)

\
size(1:8)

object2 = [1 3 5 7; 2 4 6 8];

size(object2)

size(object2, 2)

size(object2, 3)

ss = 'object';
```

```
size(ss)
")

## End(Not run)
```

---

tand *Tangent (in degrees) [GNU Octave/MATLAB compatible]*

Description

Calculates the value of tangent for each element of `x` in degrees in a manner compatible with GNU Octave/MATLAB. Zero is returned for any "elements where `x` / 180 is an integer and `Inf` for elements where (`x` - 90) / 180 is an integer." Source: Eaton.

Usage

```
tand(x)
```

Arguments

`x`	A numeric vector containing values in degrees

Value

The tangent of each element of `x` in degrees. Zero for any "elements where `x` / 180 is an integer and `Inf` for elements where (`x` - 90) / 180 is an integer."

Author(s)

David Bateman (GNU Octave tand), Irucka Embry

References

John W. Eaton, David Bateman, and Søren Hauberg (2009). *GNU Octave version 3.0.1 manual: a high-level interactive language for numerical computations*. CreateSpace Independent Publishing Platform. ISBN 1441413006, URL `http://www.gnu.org/software/octave/doc/interpreter/`. Page 358.

Examples

```
library("iemisc")

# Examples from GNU Octave tand
tand(seq(10, 80, by = 10))

tand(c(0, 180, 360))
```

```
tand(c(90, 270))
```

---

volsphere *Sphere volume*

Description

This function computes the volume of a sphere using a given radius.

Usage

```
volsphere(r)
```

Arguments

`r`	numeric vector, matrix, data.frame, or data.table that contains the radius of a sphere.

Details

The radius of a sphere is "the integral of the surface area of a sphere."

Volume of a sphere is expressed as

$$V = \frac{4}{3}\pi r^3$$

V the volume of a sphere

r the radius of a sphere

Value

volume of a sphere (as L^3 units) as an R object: a numeric `vector` or a named numeric vector if using a named object (`matrix`, `data.frame`, or `data.table`).

References

Wikimedia Foundation, Inc. Wikipedia, 30 December 2015, "Volume", https://en.wikipedia.org/wiki/Volume.

Examples

```
library("iemisc")
library("data.table")

volsphere(3) # in

volsphere(4.5) # in


x <- c(3, 4, 0.2, 12, 34, 7.5) # cm
volsphere(x)


# using a matrix of the numeric vector x
mat1 <- matrix(data = x, nrow = length(x), ncol = 1, byrow = FALSE,
       dimnames = list(c(rep("", length(x))), "Radius"))
volsphere(mat1)


# using a data.frame of the numeric vector x
df1 <- data.frame(x)
volsphere(df1)


# using a data.table of the numeric vector x
df2 <- data.table(x)
volsphere(df2)
```

Index