Title

stata.com

Mathematical functions

Contents   Functions   Video example   References   Also see

Contents

abs(x)	the absolute value of x
ceil(x)	the unique integer n such that $n-1 < x \leq n$; x (not ".") if x is missing, meaning that ceil(.a) = .a
cloglog(x)	the complementary log-log of x
comb(n,k)	the combinatorial function $n!/\{k!(n-k)!\}$
digamma(x)	the digamma() function, $d\ln\Gamma(x)/dx$
exp(x)	the exponential function e^x
expm1(x)	$e^x - 1$ with higher precision than exp(x) $-$ 1 for small values of $\lvert x\rvert$
floor(x)	the unique integer n such that $n \leq x < n+1$; x (not ".") if x is missing, meaning that floor(.a) = .a
int(x)	the integer obtained by truncating x toward 0 (thus, int(5.2) = 5 and int(-5.8) = -5); x (not ".") if x is missing, meaning that int(.a) = .a
invcloglog(x)	the inverse of the complementary log-log function of x
invlogit(x)	the inverse of the logit function of x
ln(x)	the natural logarithm, $\ln(x)$
ln1m(x)	the natural logarithm of $1-x$ with higher precision than ln($1-x$) for small values of $\lvert x\rvert$
ln1p(x)	the natural logarithm of $1+x$ with higher precision than ln($1+x$) for small values of $\lvert x\rvert$
lnfactorial(n)	the natural log of n factorial = $\ln(n!)$
lngamma(x)	$\ln\{\Gamma(x)\}$
log(x)	a synonym for ln(x)
log10(x)	the base-10 logarithm of x
log1m(x)	a synonym for ln1m(x)
log1p(x)	a synonym for ln1p(x)
logit(x)	the log of the odds ratio of x, logit(x) = $\ln\{x/(1-x)\}$
max(x_1,x_2,...,x_n)	the maximum value of $x_1, x_2, \ldots, x_n$
min(x_1,x_2,...,x_n)	the minimum value of $x_1, x_2, \ldots, x_n$
mod(x,y)	the modulus of x with respect to y
reldif(x,y)	the "relative" difference $\lvert x-y\rvert/(\lvert y\rvert+1)$; 0 if both arguments are the same type of extended missing value; *missing* if only one argument is missing or if the two arguments are two different types of *missing*

`round(`x`,`y`)` or `round(`x`)`	x rounded in units of y or x rounded to the nearest integer if the argument y is omitted; x (not "`.`") if x is missing (meaning that `round(.a)` = `.a` and that `round(.a,`y`)` = `.a` if y is not missing) and if y is missing, then "`.`" is returned
`sign(`x`)`	the sign of x: -1 if $x < 0$, 0 if $x = 0$, 1 if $x > 0$, or *missing* if x is missing
`sqrt(`x`)`	the square root of x
`sum(`x`)`	the running sum of x, treating missing values as zero
`trigamma(`x`)`	the second derivative of `lngamma(`x`)` = $d^2 \ln\Gamma(x)/dx^2$
`trunc(`x`)`	a synonym for `int(`x`)`

Functions

`abs(`x`)`

Description: the absolute value of x
Domain: $-$8e+307 to 8e+307
Range: 0 to 8e+307

`ceil(`x`)`

Description: the unique integer n such that $n - 1 < x \leq n$; x (not "`.`") if x is missing, meaning that `ceil(.a)` = `.a`

Also see `floor(`x`)`, `int(`x`)`, and `round(`x`)`.
Domain: $-$8e+307 to 8e+307
Range: integers in $-$8e+307 to 8e+307

`cloglog(`x`)`

Description: the complementary log-log of x

`cloglog(`x`)` $= \ln\{-\ln(1-x)\}$

Domain: 0 to 1
Range: $-$8e+307 to 8e+307

`comb(`n`,`k`)`

Description: the combinatorial function $n!/\{k!(n-k)!\}$
Domain n: integers 1 to 1e+305
Domain k: integers 0 to n
Range: 0 to 8e+307 or *missing*

`digamma(`x`)`

Description: the `digamma()` function, $d\ln\Gamma(x)/dx$

This is the derivative of `lngamma(`x`)`. The `digamma(`x`)` function is sometimes called the psi function, $\psi(x)$.
Domain: $-$1e+15 to 8e+307
Range: $-$8e+307 to 8e+307 or *missing*

`exp(`x`)`

Description: the exponential function e^x

This function is the inverse of `ln(`x`)`. To compute $e^x - 1$ with high precision for small values of $|x|$, use `expm1(`x`)`.

Domain: −8e+307 to 709

Range: 0 to 8e+307

`expm1(`x`)`

Description: $e^x - 1$ with higher precision than `exp(`x`)` $- 1$ for small values of $|x|$

Domain: −8e+307 to 709

Range: −1 to 8e+307

`floor(`x`)`

Description: the unique integer n such that $n \leq x < n + 1$; x (not "`.`") if x is missing, meaning that `floor(.a)` = `.a`

Also see `ceil(`x`)`, `int(`x`)`, and `round(`x`)`.

Domain: −8e+307 to 8e+307

Range: integers in −8e+307 to 8e+307

`int(`x`)`

Description: the integer obtained by truncating x toward 0 (thus, `int(5.2)` = 5 and `int(-5.8)` = −5); x (not "`.`") if x is missing, meaning that `int(.a)` = `.a`

One way to obtain the closest integer to x is `int(`x`+sign(`x`)/2)`, which simplifies to `int(`x`+0.5)` for $x \geq 0$. However, use of the `round()` function is preferred. Also see `round(`x`)`, `ceil(`x`)`, and `floor(`x`)`.

Domain: −8e+307 to 8e+307

Range: integers in −8e+307 to 8e+307

`invcloglog(`x`)`

Description: the inverse of the complementary log-log function of x

`invcloglog(`x`)` $= 1 - \exp\{-\exp(x)\}$

Domain: −8e+307 to 8e+307

Range: 0 to 1 or *missing*

`invlogit(`x`)`

Description: the inverse of the logit function of x

`invlogit(`x`)` $= \exp(x)/\{1 + \exp(x)\}$

Domain: −8e+307 to 8e+307

Range: 0 to 1 or *missing*

`ln(`x`)`

Description: the natural logarithm, $\ln(x)$

This function is the inverse of `exp(`x`)`. The logarithm of x in base b can be calculated via $\log_b(x) = \log_a(x)/\log_a(b)$. Hence,
$\log_5(x)$ = `ln(`x`)/ln(5)` = `log(`x`)/log(5)` = `log10(`x`)/log10(5)`
$\log_2(x)$ = `ln(`x`)/ln(2)` = `log(`x`)/log(2)` = `log10(`x`)/log10(2)`

You can calculate $\log_b(x)$ by using the formula that best suits your needs. To compute `ln(`$1-x$`)` and `ln(`$1+x$`)` with high precision for small values of $|x|$, use `ln1m(`x`)` and `ln1p(`x`)`, respectively.

Domain: 1e–323 to 8e+307

Range: −744 to 709

`ln1m(`x`)`

Description: the natural logarithm of $1-x$ with higher precision than `ln(`$1-x$`)` for small values of $|x|$

Domain: −8e+307 to 1 − `c(epsdouble)`

Range: −37 to 709

`ln1p(`x`)`

Description: the natural logarithm of $1+x$ with higher precision than `ln(`$1+x$`)` for small values of $|x|$

Domain: −1 + `c(epsdouble)` to 8e+307

Range: −37 to 709

`lnfactorial(`n`)`

Description: the natural log of n factorial = $\ln(n!)$

To calculate $n!$, use `round(exp(lnfactorial(`n`)),1)` to ensure that the result is an integer. Logs of factorials are generally more useful than the factorials themselves because of overflow problems.

Domain: integers 0 to 1e+305

Range: 0 to 8e+307

`lngamma(`x`)`

Description: $\ln\{\Gamma(x)\}$

Here the gamma function, $\Gamma(x)$, is defined by $\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt$. For integer values of $x > 0$, this is $\ln((x-1)!)$.

`lngamma(`x`)` for $x < 0$ returns a number such that `exp(lngamma(`x`))` is equal to the absolute value of the gamma function, $\Gamma(x)$. That is, `lngamma(`x`)` always returns a real (not complex) result.

Domain: −2,147,483,648 to 1e+305 (excluding negative integers)

Range: −8e+307 to 8e+307

`log(`x`)`

Description: a synonym for `ln(`x`)`

`log10(`x`)`
Description: the base-10 logarithm of x
Domain: 1e–323 to 8e+307
Range: -323 to 308

`log1m(`x`)`
Description: a synonym for `ln1m(`x`)`

`log1p(`x`)`
Description: a synonym for `ln1p(`x`)`

`logit(`x`)`
Description: the log of the odds ratio of x, `logit(`x`)` $= \ln\{x/(1-x)\}$
Domain: 0 to 1 (exclusive)
Range: -8e+307 to 8e+307 or *missing*

`max(`x_1`,`x_2`,...,`x_n`)`
Description: the maximum value of $x_1, x_2, \ldots, x_n$

Unless all arguments are *missing*, missing values are ignored.
`max(2,10,.,7)` = `10`
`max(.,.,.)` = `.`
Domain x_1: -8e+307 to 8e+307 or *missing*
Domain x_2: -8e+307 to 8e+307 or *missing*
...
Domain x_n: -8e+307 to 8e+307 or *missing*
Range: -8e+307 to 8e+307 or *missing*

`min(`x_1`,`x_2`,...,`x_n`)`
Description: the minimum value of $x_1, x_2, \ldots, x_n$

Unless all arguments are *missing*, missing values are ignored.
`min(2,10,.,7)` = `2`
`min(.,.,.)` = `.`
Domain x_1: -8e+307 to 8e+307 or *missing*
Domain x_2: -8e+307 to 8e+307 or *missing*
...
Domain x_n: -8e+307 to 8e+307 or *missing*
Range: -8e+307 to 8e+307 or *missing*

`mod(`x`,`y`)`

Description: the modulus of x with respect to y

$$\texttt{mod}(x,y) = x - y\ \texttt{floor}(x/y)$$

$\texttt{mod}(x\texttt{,0}) = \texttt{.}$

Domain x: $-$8e+307 to 8e+307

Domain y: 0 to 8e+307

Range: 0 to 8e+307

`reldif(`x`,`y`)`

Description: the "relative" difference $|x - y|/(|y| + 1)$; `0` if both arguments are the same type of extended missing value; *missing* if only one argument is missing or if the two arguments are two different types of *missing*

Domain x: $-$8e+307 to 8e+307 or *missing*

Domain y: $-$8e+307 to 8e+307 or *missing*

Range: 0 to 8e+307 or *missing*

`round(`x`,`y`)` or `round(`x`)`

Description: x rounded in units of y or x rounded to the nearest integer if the argument y is omitted; x (not "`.`") if x is missing (meaning that `round(.a)` = `.a` and that `round(.a,`y`)` = `.a` if y is not missing) and if y is missing, then "`.`" is returned

For $y = 1$, or with y omitted, this amounts to the closest integer to x; `round(5.2,1)` is 5, as is `round(4.8,1)`; `round(-5.2,1)` is -5, as is `round(-4.8,1)`. The rounding definition is generalized for $y \neq 1$. With $y = 0.01$, for instance, x is rounded to two decimal places; `round(sqrt(2),.01)` is 1.41. y may also be larger than 1; `round(28,5)` is 30, which is 28 rounded to the closest multiple of 5. For $y = 0$, the function is defined as returning x unmodified.

For values of x exactly at midpoints, where it may not be clear whether to round up or down, x is always rounded up to the larger value. For example, `round(4.5)` is 5 and `round(-4.5)` is -4. Note that rounding a number is based on the floating-point number representation of the number instead of the number itself. So `round()` is sensitive to representation errors and precision limits. For example, 0.15 has no exact floating-point number representation. Therefore, `round(0.15,0.1)` is 0.1 instead of 0.2. See [U] **13.12 Precision and problems therein** for details.

Also see `int(`x`)`, `ceil(`x`)`, and `floor(`x`)`.

Domain x: $-$8e+307 to 8e+307

Domain y: $-$8e+307 to 8e+307

Range: $-$8e+307 to 8e+307

`sign(`x`)`

Description: the sign of x: -1 if $x < 0$, 0 if $x = 0$, 1 if $x > 0$, or *missing* if x is missing

Domain: $-$8e+307 to 8e+307 or *missing*

Range: -1, 0, 1 or *missing*

`sqrt(`x`)`
Description: the square root of x
Domain: 0 to 8e+307
Range: 0 to 1e+154

`sum(`x`)`
Description: the running sum of x, treating missing values as zero

For example, following the command `generate y=sum(x)`, the jth observation on `y` contains the sum of the first through jth observations on `x`. See [D] **egen** for an alternative sum function, `total()`, that produces a constant equal to the overall sum.
Domain: all real numbers or *missing*
Range: −8e+307 to 8e+307 (excluding *missing*)

`trigamma(`x`)`
Description: the second derivative of `lngamma(`x`)` = $d^2 \ln\Gamma(x)/dx^2$

The `trigamma()` function is the derivative of `digamma(`x`)`.
Domain: −1e+15 to 8e+307
Range: 0 to 8e+307 or *missing*

`trunc(`x`)`
Description: a synonym for `int(`x`)`

Video example

How to round a continuous variable

References

Abramowitz, M., and I. A. Stegun, ed. 1964. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Washington, DC: National Bureau of Standards.

Cox, N. J. 2003. Stata tip 2: Building with floors and ceilings. *Stata Journal* 3: 446–447.

——. 2007. Stata tip 43: Remainders, selections, sequences, extractions: Uses of the modulus. *Stata Journal* 7: 143–145.

——. 2018. Speaking Stata: From rounding to binning. *Stata Journal* 18: 741–754.

Oldham, K. B., J. C. Myland, and J. Spanier. 2009. *An Atlas of Functions*. 2nd ed. New York: Springer.

Also see

[FN] **Functions by category**

[D] **egen** — Extensions to generate

[D] **generate** — Create or change contents of variable

[M-4] **Intro** — Categorical guide to Mata functions

[U] **13.3 Functions**