Rounding up, rounding down

By Rick Wicklin on The DO Loop | October 3, 2011

Topics | Programming Tips

AS has several ways to round a number to an integer. You can round a number up, round it down, o your data contain both positive and negative values, you can also round numbers toward zero, or a

The functions that perform rounding are the same in the DATA step as in the SAS/IML language:

- The FLOOR function rounds down.
- The CEIL function rounds up.
- The ROUND function rounds to the nearest integer.
- The INT function rounds towards zero.
- SAS does not have a built-in function that rounds away from zero, but you can combine the SIG function to round away from zero.

The following statements evaluate each function on a vector of numbers so that you can observe the

```
proc iml;
x={-3.5 -2.9 -1.1 1.1 2.9 3.5};
f = floor(x); /* towards -infinity */
c = ceil(x);  /* towards +infinity */
```

```
r = round(x); /* towards nearest integer */
t = int(x);   /* towards zero */
s = int(x) + sign(x); /* away from zero */

m = f//c//r//t//s; /* pack into matrix */
cNames = putn(x,"BEST4.");
rNames = {FLOOR CEIL ROUND INT "INT + SIGN"};
print m[c=cNames r=rNames label="Rounding Functions"];
```

Rounding Functions						
	-3.5	-2.9	-1.1	1.1	2.9	3.5
FLOOR	-4	-3	-2	1	2	3
CEIL	-3	-2	-1	2	3	4
ROUND	-4	-3	-1	1	3	4
INT	-3	-2	-1	1	2	3
INT + SIGN	-4	-3	-2	2	3	4

ounding to units that are not integers

ome SAS users do not realize that the ROUND function enables you to round a number to a certain
OUND function takes an optional second argument, which is the rounding unit. The following statem
earest hundred, the nearest ten, the nearest unit (the default), the nearest tenth, and the nearest hur

```
x = 123.456;
units = {100 10 1 0.1 0.01};
y = round(x, units);
cNames = putn(units,"BEST4.");
print y[c=cNames r="123.456" label="Rounding to Different Units"];
```

Rounding to Different Units					
	100	10	1	0.1	0.01
123.456	100	120	123	123.5	123.46

Although it is not usually done, you can also round a number to any unit you want. For example, roun
nearest third.

Tags

Share

ABOUT AUTHOR

Rick Wicklin

Distinguished Researcher in Computational Statistics

Rick Wicklin, PhD, is a distinguished researcher in computational statistics at SAS and is a IML and SAS/IML Studio. His areas of expertise include computational statistics, simulatio modern methods in statistical data analysis. Rick is author of the books *Statistical Progran* and *Simulating Data with SAS*.

16 COMMENTS

Chris Hemedinger on October 3, 2011 8:44 am

"...not usually done", you say -- hence the popular note, "31109 - Unusual uses for the ROUND it might show even better if we rounded up on the rating system.

Michael Richardson on July 22, 2013 10:24 am

One caveat to your "round away from zero" function: notice that integer values would be rounde is rounded to '4'). I doubt that would be the user's intention. Perhaps
s = ceil(abs(x)) # sign(x);
is a better choice.

Benjamin on November 16, 2016 11:48 am

Is there a way to always round down to different units? For example, I might want 199 to round

Rick Wicklin on November 16, 2016 1:02 pm

Yes. The ROUND function has a second optional argument that specifies the rounding u
rounds x to the nearest 100. To round DOWN, you'll need to subtract off half of the round
100) rounds x down to the nearest multiple of 100.

Somnath on March 29, 2017 5:45 am

Can you explain algorithm of round function?

Rick Wicklin on March 29, 2017 5:56 am

The documentation (which I link to in the article) contain a complete descript
the closest integer to x. ROUND(x, 0.1) returns the nearest "tenth" to x.

Chen on July 12, 2017 11:09 pm

Hi Rick, how could we ceil/floor a number to decimal points? For example, ceil 1.88 to 1.9 rathe

Rick Wicklin on July 13, 2017 6:15 am

To round, you can use round(1.88, 0.1). To floor or ceil), you can first multiply by the roun
then divide by the rounding factor. In your case, the answer is ceil(10*1.88) / 10. If you w
ceil(100*x)/100. For k decimal places: ceil(10**k * x)/ 10**k.

Leonid Batkhan on July 27, 2017 9:51 am

"SAS does not have a built-in function that rounds away from zero, but you can combine the SI
function to round away from zero". That "another function" could be truncn() - a user-defined fu
numbers. It is described in Truncating decimal numbers in SAS without rounding. One can add
for negative, +1 for positive, and 0 for zero) to the truncated number to achieve rounding away
truncn(x) + sign(x).

Erin Hodson on January 10, 2018 8:42 am

Hi Rick! I want to round all of the values of a variable up to the nearest 100. For example 135 tc
using this function?

Rick Wicklin on January 10, 2018 8:56 am

Sure. If you want to round to the nearest quantity, Q, use Q*ceil(x/Q). The following illustı

```
data A;
input x @@;
R = 100 * ceil( x / 100 );
datalines;
2 97 100 101 135 199 200 201
;
proc print; run;
```

Erin Hodson on January 10, 2018 9:30 am

Worked perfectly, thank you so much!!

varghese on April 2, 2019 4:12 am

Hi Rick,

If I want round down 195 to 100, 215 to 200,285 to 200 Which function I can use?

Rick Wicklin on April 2, 2019 5:49 am

To round down to the nearest 100, you need to
1. Divide your number by 100.
2. Use the FLOOR function to round down the ratio to the nearest integer
3. Multiply the result by 100 to restore the scale of the original number.
If x is your number, the SAS statement looks like this
y = 100 * floor(x/100); /* round down to nearest 100 */

Patrick O'Leary on May 23, 2019 1:25 pm

Hi Rick! I would like to round up to the nearest 0.003. Example : 0.051 would become 0.053. Than you!

Rick Wicklin on May 23, 2019 1:39 pm

The expression y = round(x, 0.003) rounds x to the nearest multiple of 0.003. However, tl itself because 0.051 = (17)(0.003). The next multiple is 0.054, so nothing will ever round

LEAVE A REPLY

Your Comment

Your Name

Your Email

Your Website