Subject: Re: REBIN needs explaining
Posted by JD Smith on Tue, 30 Nov 2004 23:56:14 GMT
View Forum Message <> Reply to Message

On Tue, 30 Nov 2004 16:32:03 -0700, David Fanning wrote:

> Hi Folks,
>
> Can someone (Craig?) explain this to me.
>
> Consider:
>
> v = [0, 10]
>
> I understand this:
>
> Print, rebin(v, 20, /Sample)
> 0 0 0 0 0 0 0 0 0
> 0 10 10 10 10 10 10 10 10
> 10 10
>
> What I don't understand is this:
>
> Print, rebin(v, 20)
> 0 1 2 3 4 5 6 7 8
> 9 10 10 10 10 10 10 10 10
> 10 10
>
> Aren't there an awful lot of 10s in there!? Why doesn't this come
> out like this:
>
> 0 0 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8
> 9 9 10 10
>
> I notice REBIN, CONGRID, and even CMCONGRID does this same thing. Why?

From the manual:

> When expanding an array, REBIN interpolates, it never
> extrapolates. Each of the n-1 intervals in the n-element input array
> produces m/n interpolates in the m-element output array. The last m/n
> points of the result are obtained by duplicating element n-1 of the
> input array because their interpolates would lie outside the input
> array.

So here it has just 1 interval producing 10 points, and 10 points are duplicated from the last value. The manual also recommends the solution:

If this effect is undesirable, use the INTERPOLATE function.

A good question is why they consider the need to avoid extrapolation
by duplicating elements, rather than just interpolating on all
internal intervals. I'm not sure for what problems the current
behavior would be desirable.

JD

Subject: Re: REBIN needs explaining
Posted by David Fanning on Wed, 01 Dec 2004 00:53:39 GMT
View Forum Message <> Reply to Message

JD Smith writes:

> So here it has just 1 interval producing 10 points, and 10 points are
> duplicated from the last value. The manual also recommends the
> solution:
>
> If this effect is undesirable, use the INTERPOLATE function.

Well, I *would* use the INTERPOLATE function if I could
understand it. :-(

The REBIN documentation says this:

****
; A four point vector:
A = [0, 10, 20, 30]

; Expand by a factor of 3:
B = REBIN(A, 12)

PRINT, B
IDL prints:

0 3 6 10 13 16 20 23 26 30 30 30

Note that the last element is repeated three times. If this effect is
undesirable, use the INTERPOLATE function. For example, to produce 12
equally spaced interpolates from the interval 0 to 30:

B = INTERPOLATE(A, 3./11. * FINDGEN(12))
PRINT, B
IDL prints:

0 2 5 8 10 13 16 19 21 24 27 30
****

But,

IDL> Print, 3./11. * FINDGEN(12))
0.000000 0.272727 0.545455 0.818182 1.09091
1.36364 1.63636 1.90909 2.18182 2.45455 2.72727
3.00000

How in the world to *those* numbers, when applied to A, produce
the result I get? Seems like magic to me. :-(

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: REBIN needs explaining
Posted by Chris Lee on Wed, 01 Dec 2004 08:23:49 GMT
View Forum Message <> Reply to Message

In article <MPG.1c16c3fd524557b998988c@news.frii.com>, "David Fanning"
<davidf@dfanning.com> wrote:

...
> INTERPOLATE(A, 3./11. * FINDGEN(12))
> PRINT, B
> IDL prints:
> 0 2 5 8 10 13 16 19 21 24 27 30
> ********** But,
>
> IDL> Print, 3./11. * FINDGEN(12))
> 0.000000 0.272727 0.545455 0.818182 1.09091
> 1.36364 1.63636 1.90909 2.18182 2.45455 2.72727
> 3.00000
> How in the world to *those* numbers, when applied to A, produce the
> result I get? Seems like magic to me. :-(Cheers,
> David
>

It looks like IDL is flooring the numbers (or just dropping the floating
point part) to make them INTs.

```
IDL> a=[0,10,20,30]
IDL> print, interpolate(a,findgen(12)*3/11.)
       0       2       5       8      10      13      16      19      21
      24      27      30

IDL> print, interpolate(a*1.0,findgen(12)*3/11.)
     0.00000     2.72727     5.45455     8.18182     10.9091     13.6364
     16.3636     19.0909     21.8182     24.5455     27.2727     30.0000

IDL> print, interpol(a, 12)
     0.00000     2.72727     5.45455     8.18182     10.9091     13.6364
     16.3636     19.0909     21.8182     24.5455     27.2727     30.0000
```

Chris.

Subject: Re: REBIN needs explaining
Posted by b.hinrichsen on Wed, 01 Dec 2004 08:38:14 GMT
View Forum Message <> Reply to Message

David Fanning <davidf@dfanning.com> wrote in message
news:<MPG.1c16c3fd524557b998988c@news.frii.com>...
> JD Smith writes:
>
>> So here it has just 1 interval producing 10 points, and 10 points are
>> duplicated from the last value. The manual also recommends the
>> solution:
>>
>> If this effect is undesirable, use the INTERPOLATE function.
>
> Well, I *would* use the INTERPOLATE function if I could
> understand it. :-(
>
> The REBIN documentation says this:
>
> **********
> ; A four point vector:
> A = [0, 10, 20, 30]
>
> ; Expand by a factor of 3:
> B = REBIN(A, 12)
>
> PRINT, B
> IDL prints:
>
> 0 3 6 10 13 16 20 23 26 30 30 30

>
> Note that the last element is repeated three times. If this effect is
> undesirable, use the INTERPOLATE function. For example, to produce 12
> equally spaced interpolates from the interval 0 to 30:
>
> B = INTERPOLATE(A, 3./11. * FINDGEN(12))
> PRINT, B
> IDL prints:
>
> 0 2 5 8 10 13 16 19 21 24 27 30
> **********
>
> But,
>
> IDL> Print, 3./11. * FINDGEN(12))
> 0.000000 0.272727 0.545455 0.818182 1.09091
> 1.36364 1.63636 1.90909 2.18182 2.45455 2.72727
> 3.00000
Interpolate considers these values as the new x-array which determines
the values of the y-array. The original x-array is implicitly
understood to be the index of A viz. [0,1,2,3].
>
> How in the world to *those* numbers, when applied to A, produce
> the result I get? Seems like magic to me. :-(
>
> Cheers,
>
> David

Greetings,

Bernd

Subject: Re: REBIN needs explaining
Posted by Paolo Grigis on Wed, 01 Dec 2004 08:45:40 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
> JD Smith writes:
>
>
>> So here it has just 1 interval producing 10 points, and 10 points are
>> duplicated from the last value. The manual also recommends the
>> solution:
>>
>> If this effect is undesirable, use the INTERPOLATE function.
>

>
> Well, I *would* use the INTERPOLATE function if I could
> understand it. :-(
>
> The REBIN documentation says this:
>
> ***
> ; A four point vector:
> A = [0, 10, 20, 30]
>
> ; Expand by a factor of 3:
> B = REBIN(A, 12)
>
> PRINT, B
> IDL prints:
>
> 0 3 6 10 13 16 20 23 26 30 30 30
>
> Note that the last element is repeated three times. If this effect is
> undesirable, use the INTERPOLATE function. For example, to produce 12
> equally spaced interpolates from the interval 0 to 30:
>
> B = INTERPOLATE(A, 3./11. * FINDGEN(12))
> PRINT, B
> IDL prints:
>
> 0 2 5 8 10 13 16 19 21 24 27 30
> ***
>
> But,
>
> IDL> Print, 3./11. * FINDGEN(12))
> 0.000000 0.272727 0.545455 0.818182 1.09091
> 1.36364 1.63636 1.90909 2.18182 2.45455 2.72727
> 3.00000
>
> How in the world to *those* numbers, when applied to A, produce
> the result I get? Seems like magic to me. :-(

They say, a picture is worth a thousand words, therefore the
following plot should be able to enlighten the meaning of INTERPOLATE
better than a lengthy explanation...

```
x=findgen(4)
a=[0.,10,20,30]
z=3./11. * FINDGEN(12)
plot,x,a,psym=-4,xrange=[-0.5,3.5],yrange=[-5,35]
oplot,z,interpolate(a,z),psym=1
```

Ciao,
Paolo
>
> Cheers,
>
> David
>

--

Paolo Grigis
ETHZ - Institute of Astronomy email: pgrigis@astro.phys.ethz.ch
Scheuchzerstrasse 7
ETH Zentrum phone: ++41 1 632 42 20
8092 Zurich fax : ++41 1 632 12 05
Switzerland http://www.astro.phys.ethz.ch/

Subject: Re: REBIN needs explaining
Posted by David Fanning on Wed, 01 Dec 2004 12:38:25 GMT
View Forum Message <> Reply to Message

b.hinrichsen@fkf.mpg.de writes:

> Interpolate considers these values as the new x-array which determines
> the values of the y-array. The original x-array is implicitly
> understood to be the index of A viz. [0,1,2,3].

Ah, the *X* values! Alles klar. Danke. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
