Subject: histogram function question
Posted by Dae-Kyu Shin on Fri, 06 Mar 2015 14:03:01 GMT
View Forum Message <> Reply to Message

hi

here is a example code

```
   data = [0, 0.3, 1.3d]
   h1 = HISTOGRAM(data, nbin=3, min=0, max=1, loc=loc1)
   h2 = HISTOGRAM(data, binsize=0.5d, min=0, max=1, loc=loc2)

print, h1
print, h2

h1 =  2         0         1
h2 =  2         0         0
```

h1 and h2 is not equal!!
is it correct??

thanks

Subject: Re: histogram function question
Posted by Heinz Stege on Fri, 06 Mar 2015 14:22:59 GMT
View Forum Message <> Reply to Message

No, this is not correct. This is a bug.

Cheers, Heinz

Subject: Re: histogram function question
Posted by Dae-Kyu Shin on Fri, 06 Mar 2015 14:24:51 GMT
View Forum Message <> Reply to Message

> hi
>
> here is a example code
>
>
>     data = [0, 0.3, 1.3d]

>     h1 = HISTOGRAM(data, nbin=3, min=0, max=1, loc=loc1)
>     h2 = HISTOGRAM(data, binsize=0.5d, min=0, max=1, loc=loc2)
>
>  print, h1
>  print, h2
>
>  h1 =  2         0         1
>  h2 =  2         0         0
>
>
>  h1 and h2 is not equal!!
>  is it correct??
>
>
>  thanks
IDL verssion : 8.4

---

Subject: Re: histogram function question
Posted by David Fanning on Fri, 06 Mar 2015 14:26:39 GMT
View Forum Message <> Reply to Message

Dae-Kyu Shin writes:

>
>  hi
>
>  here is a example code
>
>
>     data = [0, 0.3, 1.3d]
>     h1 = HISTOGRAM(data, nbin=3, min=0, max=1, loc=loc1)
>     h2 = HISTOGRAM(data, binsize=0.5d, min=0, max=1, loc=loc2)
>
>  print, h1
>  print, h2
>
>  h1 =  2         0         1
>  h2 =  2         0         0
>
>
>  h1 and h2 is not equal!!
>  is it correct??

I don't think so:

IDL> data = [0, 0.3, 1.3d]
IDL> h1 = cghistogram(data, nbin=3, min=0.0, max=1.0 ,loc=loc1)

```
IDL> h2 = cghistogram(data, binsize=0.5d, min=0.0, max=1.0 ,loc=loc2)
IDL> print, h1
       2       0       0
IDL> print, h2
       2       0       0
```

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

---

Subject: Re: histogram function question
Posted by Matthew Argall on Fri, 06 Mar 2015 14:27:36 GMT
View Forum Message <> Reply to Message

There is a note in the documentation

Note: The data type of the value specified for MIN should match the data type of the input array. Since MIN is converted to the data type of the input array, specifying mismatched data types may produce undesired results.

In the first case, 0.3 and 1.3 are truncated to 0 and 1, respectively.

http://exelisvis.com/docs/HISTOGRAM.html#H_835179117_677182

---

Subject: Re: histogram function question
Posted by Matthew Argall on Fri, 06 Mar 2015 14:31:28 GMT
View Forum Message <> Reply to Message

> Note: The data type of the value specified for MIN should match the data type of the input array. Since MIN is converted to the data type of the input array, specifying mismatched data types may produce undesired results.
>
> In the first case, 0.3 and 1.3 are truncated to 0 and 1, respectively.

Maybe I should learn how to read, too :-p The MIN, MAX, and BINSIZE values are converted to the input type, not the other way around.

However, the behavior is as if the input array is converted to integer type...

---

Subject: Re: histogram function question
Posted by Helder Marchetto on Fri, 06 Mar 2015 15:15:57 GMT
View Forum Message <> Reply to Message

On Friday, March 6, 2015 at 3:31:31 PM UTC+1, Matthew Argall wrote:
>> Note: The data type of the value specified for MIN should match the data type of the input array. Since MIN is converted to the data type of the input array, specifying mismatched data types may produce undesired results.
>>
>> In the first case, 0.3 and 1.3 are truncated to 0 and 1, respectively.
>
> Maybe I should learn how to read, too :-p The MIN, MAX, and BINSIZE values are converted to the input type, not the other way around.
>
> However, the behavior is as if the input array is converted to integer type...

cgHistogram works because it calculates the histogram() *always* using binsize. When nbins is present instead of binsize, cgHistogram calculates binsize (after taking care of the types).

As far as the example goes, the result is the same also when the type of min, max and binsize are the same as the input data.

I find the result a bit strange, because the bin locations are not what I would expect them to be! I would have expected the bin locations to be:
nbins=3 -> [0-0.3, 0.3-0.6 and 0.6-1.0]
Because histogram should distribute 3 bins evenly between 0.0 and 1.0.

binsize=0.5 -> [0.0-0.5 and 0.5-1.0] (for only 2 bins... not 3)
Because the bins sizes should be starting at 0.0 and have a size of 0.5.

So I would have expected the following two results:
2, 0, 0
and
2, 0

Maybe this is just a problem with my understanding of the histogram function. I'll read about it over weekend and maybe I will have an idea of what's happening...
I guess, that generally speaking, the histogram function is wyginwye (well known acronym for "what you get is not what you expect").

```
IDL> data = [0, 0.3, 1.3d]
IDL> print, HISTOGRAM(data, nbin=3, min=0d, max=1d, loc=loc1), HISTOGRAM(data,
binsize=0.5d, min=0d, max=1d, loc=loc2)
IDL> print, loc1, loc2
           2           0           1
           2           0           0
      0.00000000  0.50000000   1.0000000
      0.00000000  0.50000000   1.0000000
```

I'll stick to cgHistogram. At least it's consistent!

h

---

Subject: Re: histogram function question
Posted by Burch on Fri, 06 Mar 2015 16:08:09 GMT
View Forum Message <> Reply to Message

On Friday, March 6, 2015 at 8:03:05 AM UTC-6, Dae-Kyu Shin wrote:
> hi
>
> here is a example code
>
>
>  data = [0, 0.3, 1.3d]
>  h1 = HISTOGRAM(data, nbin=3, min=0, max=1, loc=loc1)
>  h2 = HISTOGRAM(data, binsize=0.5d, min=0, max=1, loc=loc2)
>
> print, h1
> print, h2
>
> h1 = 2 0 1
> h2 = 2 0 0
>
>
> h1 and h2 is not equal!!
> is it correct??
>
>
> thanks

There is a note in the documentation for the MAX keyword:

"Note: If NBINS is specified, the value for MAX will be adjusted to NBINS*BINSIZE + MIN. This ensures that the last bin has the same width as the other bins."

So it seems for h1 BINSIZE is calculated to be

(max - min)/(nbins-1) = 0.5

and then max is adjusted to

nbins*binsize + min = 1.5,

whereas in h2 there is no *internal* adjustment to MAX.

-Jeff

---

Subject: Re: histogram function question
Posted by Helder Marchetto on Fri, 06 Mar 2015 16:28:50 GMT
View Forum Message <> Reply to Message

That makes sense. Thanks for the explanation.
(Why does one go through the min, nbins and binsize keyword documentation only to find out that the max keyword makes the difference is a mystery).
Cheers
Helder

Subject: Re: histogram function question
Posted by Heinz Stege on Fri, 06 Mar 2015 16:43:55 GMT
View Forum Message <> Reply to Message

On Fri, 6 Mar 2015 08:08:09 -0800 (PST), Jeff B wrote:

> whereas in h2 there is no *internal* adjustment to MAX.
>
In other words: The first bin contains all values
  {x; 0.0 <= x < 0.5}
The second bin:
  {x; 0.5 <= x < 1.0}
The third bin:
  {x; 1.0 <= x < 1.5 AND x <= MAX)}

I may be wrong in my quick judgement, calling this a bug. However I
think, that it is an "inartfully behaviour".

Cheers, Heinz

Subject: Re: histogram function question
Posted by Fabzi on Fri, 06 Mar 2015 17:48:28 GMT
View Forum Message <> Reply to Message

On 06.03.2015 17:43, Heinz Stege wrote:
> I may be wrong in my quick judgement, calling this a bug. However I
> think, that it is an "inartfully behaviour".

agreed. If the "loc" values are the same, then the binsize is the same,
then the histogram should be the same.

Fabien