Subject: Re: What's HISTOGRAM doing now?
Posted by David Fanning on Fri, 22 Oct 2010 12:12:38 GMT
View Forum Message <> Reply to Message

Mrunmayee writes:

> Why are bins different? Output is unchaged for log bins when I am
> histo-ing alog10(data) with corresponding log bins. Also, instead of
> using BINSIZE, if I use NBINS in program, things work as they should.
> So what's histogram upto NOW? Where am I missing something?

The most common reason for histogram to produce
strange results is a data type mismatch
between the data and the value you pass in
for the binsize. The documentation mentions
they have to be the same data type, but this
is a strange requirement for IDL users, who
most of the time could care less about data
type. I would say most people don't know
this requirement exists.

I've never understood why Histogram can't
just do the data type conversion for you
and thus behave like a normal IDL command.
This was the most obnoxious thing I had to
deal with when I was creating the Histoplot
program.

Remnants of the struggle live on in the form
of the Convert_to_Type program in the Coyote
Library. :-)

Cheers,

David

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

Subject: Re: What's HISTOGRAM doing now?
Posted by Mrunmayee on Fri, 22 Oct 2010 14:21:46 GMT
View Forum Message <> Reply to Message

On Oct 22, 5:12 pm, David Fanning <n...@dfanning.com> wrote:
> Mrunmayee writes:
>> Why are bins different? Output is unchaged for log bins when I am
>> histo-ing alog10(data) with corresponding log bins. Also, instead of
>> using BINSIZE, if I use NBINS in program, things work as they should.
>> So what's histogram upto NOW? Where am I missing something?
>
> The most common reason for histogram to produce
> strange results is a data type mismatch
> between the data and the value you pass in
> for the binsize. The documentation mentions
> they have to be the same data type, but this
> is a strange requirement for IDL users, who
> most of the time could care less about data
> type.

Thanks, David.

In general, I try to pay attention to datatype matching, but I didn't know this requirement for histogram. In my tester code, though, data types *are* same. If I put help statement, to just check, I do get both - data and binsize - as float. So where is the discrepancy? For the time being I solved my problem using Nbins since it was giving me what I wanted. But I thought I understood histogram at least to *some* extent. Apparently, it can still baffle me.

Also, as a sidenote, why is it considered better than for loops? That is, how is it optimized? Or is it some water that I better not enter? I keep trying to convince people around me that IDL way is faster but I don't know whether it is faster than programming in other languages. Any pointer would be appreciated.

Of course, main question is the title of this topic anyway.
