Subject: HISTOGRAM and the Razor's Edge.
Posted by timrobishaw on Thu, 12 Jun 2003 06:44:32 GMT
View Forum Message <> Reply to Message

So I've been reading up on HISTOGRAM and how it is optimized to be wicked fast.
Also, I'm discovering the fancy things one can do with reverse indices.
I've even read JD Smith's exegesis on the topic:
"HISTOGRAM: The Breathless Horror and Disgust"

So, just when I thought it was safe to start using HISTOGRAM with the frequency that Californians use LIKE, I was brought this scary result by the guy on the other side of my wall (his name is Tiberius):

```
IDL> print,  histogram([-5.50,-5.45,-5.40,-5.35,-5.30,-5.25],min=-5.50,bi nsize=0.05)
       1           2           0           2           0           1
```

Wait a minute, this should be a uniform distribution!

Now, I admit that Tiberius contrived this example with the intent to cause harm to HISTOGRAM by placing the values at the exact boundaries of each bin. But, we didn't EXPECT it to break! Honest. After scratching our heads for a bit we realized that HISTOGRAM is conducting its internal affairs by subtracting off MIN from the data and then dividing by BINSIZE:

```
IDL> print, ([-5.50,-5.45,-5.40,-5.35,-5.30,-5.25]-(-5.50))/0.05,
format='(f21.19)'
0.0000000000000000000
1.0000038146972656250
1.9999980926513671875
3.0000019073486328125
3.9999961853027343750
5.0000000000000000000
```

Well, there ya go. It's a roundoff error problem that results from trying to balance the values on a razor's edge... the subtraction and division knock a few values off balance. But, the result is still WRONG and I just don't know enough about roundoff error to know if this is an insurmountable problem (but my educated guess is: yes.) Is there some clever way to make HISTOGRAM behave properly in such situations?

Thanks a bunch -Tim.

========== ==========
"Nothing shocks me. I'm a scientist." - Indiana Jones

Tim Robishaw UC Berkeley Astronomy