Subject: Histogram and bin sizes
Posted by jeffnettles4870 on Wed, 20 Feb 2008 19:20:40 GMT
View Forum Message <> Reply to Message

I've always wondered why you have to use a constant bin size with HISTOGRAM(). To quote J.D.'s famous tutorial: "a histogram represents nothing more than a fancy way to count." Doesn't an imposed constant bin size imply that this is the only way it's ok to count? I can think of several reasons i wouldn't want to do this - I used logarithmic bin sizes in my dissertation, for example (now i'm hoping someone isn't going to answer this post saying i screwed up in my dissertation :-)). And besides, Excel lets you use arbitrary bin sizes....and if Excel lets you do it, it has to be ok, right???? ;-)

Jeff

Subject: Re: Histogram and bin sizes
Posted by Conor on Fri, 22 Feb 2008 13:51:48 GMT
View Forum Message <> Reply to Message

On Feb 21, 5:54 pm, "Kenneth P. Bowman" <k-bow...@null.edu> wrote:
> In article
> < f6219865-59f4-4bf8-8718-67884c9df...@64g2000hsw.googlegroups .com >,
>
>
>
> Conor <cmanc...@gmail.com> wrote:
>> Arbitrary bin sizes should be pretty easy to program. You just need
>> to map your data points appropriately. For instance if you had the
>> data set:
>
>> x = randomu(seed,100)
>
>> and you wanted bins from:
>> [0-.1,.1-.3,.3-.35,.35-.8,.8-1]
>
>> you might do something like this:
>
>> x = randomu(seed,100)
>> bins = [[0,.1], [.1,.3], [.3,.35], [.35,.8], [.8,1]]
>> newx = fltarr(n_elements(x))
>> for i=0,n_elements(bins[0,*])-1 do begin
>> w = where(x ge bins[0,i] and x lt bins[1,i], c)
>> if c gt 0 then newx[w] = i+.5
>> endfor
>
>> hist = histogram(newx,binsize=1.0,min=0)

>> plothist,newx
>
> This will work, but will be extremely slow because you test every value
> in the input array once for every bin.
>
> The VALUE_LOCATE approach will be much faster, particularly for large
> numbers of bins, as it does a binary search.
>
> Ken Bowman

Oh fancy! I like it.
