Subject: Smooth()
Posted by Richard French on Thu, 14 Dec 2000 05:28:57 GMT
View Forum Message <> Reply to Message

Out of curiosity, can anyone think of a good reason that the
smooth() function bombs when you ask it to smooth something by 1?

Ex:

```
print,smooth([1.,3.,4.,2.],3)
   1.00000      2.66667      3.00000      2.00000
print,smooth([1.,3.,4.,2.],1)
% SMOOTH: Width must be > 2 and smaller than array dimensions: <INT
(       1)>
% Execution halted at:  $MAIN$
```

I have lots of instances where the amount of smoothing I want to
do is a variable, and it seems silly to have to do my own checking
to see if I actually need any smoothing or not.
It certainly makes sense to me that if the
smoothing width is only one bin, the smooth() function should just
be a no-op and return the input array. This is what the REBIN()
function does, after all - it does not complain if you tell it you
want the array to be rebinned to its actual size!

I've had to construct a 'mysmooth()' function that checks to see
if the number of points by which to smooth things is less than 2,
in which case I just return the calling array.

Are there other functions out there that you can think of that
don't have a sensible default evalution for limiting cases like this?
I would love to see this one changed.

Dick French