Subject: Re: interpolation question
Posted by news.verizon.net on Mon, 24 Apr 2006 19:47:50 GMT
View Forum Message <> Reply to Message

> I am so wonder that why IDL has no simple function like MATLAB's
> 'interp2'.

I agree with you that IDL appears deficient to MATLAB in providing an easy and consistent set of interpolation routines. First of all, while IDL has interpol.pro for 1d interpolation, there is no equivalent function for 2-d interpolation. (bilinear.pro and interpolate require you to supply indicies). And while interpol.pro provides several interpolation methods, it doesn't include the simplest (nearest neighbor), though this would be easy to add (as in http://idlastro.gsfc.nasa.gov/ftp/pro/math/linterp.pro). It makes much more sense to have functions interp1d and interp2d, each with a variety of interpolation methods available.

> Another problem is for value_locate. Some suggestions
> mentioned to use value_locate. Here is a example to show my problem.
>
> IDL> vec = [2.0, 5.0, 8.0, 10.0]
> IDL> print, vec
> 2.00000 5.00000 8.00000 10.0000
> IDL> loc = VALUE_LOCATE(vec, [0.0, 4.5, 5.0, 6.0, 12. ])
DL> print, loc
-1 0 1 1 3

VALUE_LOCATE is doing what it says it does -- returning a value j such that
vec[j] < x < vec[j+1]. I don't think anyone suggested that VALUE_LOCATE can give you the answer by itself, but both JD Smith and the archive posting from David Fanning (http://tinyurl.com/r9t5s) showed how you could use VALUE_LOCATE to get the index of the nearest value. In this case

loc = round(loc+(x-vec[loc]) / (vec[loc+1]-vec[loc]))

(Actually one should first check that 0 < loc < N_Elements(vec)-1, as in
http://idlastro.gsfc.nasa.gov/ftp/pro/math/tabinv.pro )

--Wayne