Subject: Re: Missing Data
Posted by Peter Mason on Thu, 18 Jul 1996 07:00:00 GMT
View Forum Message <> Reply to Message

>> I have to plot data gaps. I know that there is a plotm routine in the
>> user library that will plot missing data as long as there is a default
>> value, however, my data has not default values. I know where the
>> missing data occurs. For example, if I have an array of x(10) I know
>> that I do not want to connect the points between x(3:4) and x(7:8). Is
>> there any way to plot that with the existing functions or with simple
>> modifications to a user routine? I can think of a couple of ways to do
>> that but those ways won't be practical with large data sets.
>
>
> Does this example work for you?
>
> x = indgen(10) ; Create a data array.
> x(2) = -99. & x(7) = -99. ; Assign missing data values.
> x (where (x lt -50)) = 999. ; Change missing data points to large values.
> plot, x, max=500. ; Use the max_value keyword.

Alternatively, you could use the special NaN ("not-a-number") functionality
which was introduced (for most platforms?) in IDL version 4. A number of
IDL routines, including PLOT and PLOTS, automatically ignore data with NaN
"values". Also, NaN is not specific to IDL; it's "value" is defined in
the IEEE floating point standard, and so can be recognised by other
applications which are aware of IEEE "denormals" (NaN and infinity).

You can get at NaN via the !VALUES system variable:
!VALUES.F_NAN for float
!VALUES.D_NAN for double

If DAT is a FLOAT data array and INX is an index into DAT giving the bad
values, you can set the bad values to NaN with:
DAT(INX) = !VALUES.F_NAN
Or, considering the latest posting on "Problems with the IDL TIME_TEST"
(=> system variables seem to be accessed slower than general ones), it
might be quicker to use:
NAN = !VALUES.F_NAN & DAT(INX) = NAN
if you have a lot of bad data.

Peter Mason