Subject: Re: Comparing 2 arrays
Posted by ph le sager on Wed, 29 Aug 2007 17:32:39 GMT
View Forum Message <> Reply to Message

For somewhat similar reasons, I came up with the following function. I find myself using it in lot of routines to replace array subtraction. There is probably better way to do it, but it works fine, and I can tell then where the two arrays are not equal.

```
;+
; NAME:      FLOAT_DIFF
;
;          ...usefull only if you specify EPS > machine precision,
;          or use STRICT. Keep reading...
;
; PURPOSE: - to take into account the accuracy in number
;          when differentiating data.
;
;    In a computer, differences are zero if they are less than the
;    precision of the float (or double) representation. See idea in:
;
;    http://www.ibiblio.org/pub/languages/fortran/ch1-8.html#02
;
;    Here, the idea is to overwrite the machine precision. Basically,
;    DATA1 - DATA2 = 0 if less than the relative-EPS, where you can
;    define EPS. For example, if your data accuracy is 1% (which is a
;    lot less than the machine accuracy), you would type:
;
;            DIFF = FLOAT_DIFF(data1,data2,eps=0.01)
;
;    This is really useful to disregard insignificant differences
;    when comparing datasets.
;
;
; INPUTS: - data1, data2 : data to differentiate. If they are
;          not supplied(!), NAN is return. If they have different
;          size the smallest size of the two array is used, like for
;          a regular array difference.
;
;
; OUTPUT: - array same size as the smallest input data size
;            output  = array1 - array2
;
; CATEGORY:  math, testing
;
; CALLING SEQUENCE:
;
;         result = float_diff(array1, array2, eps=eps)
```

```
;
; KEYWORD PARAMETERS:
;
;       DOUBLE : set if everything is done with double precision
;
;       EPS : to redefine the floating (OR DOUBLE) point
;             precision. If not set, the machine precision for
;             floating (OR DOUBLE) is used, which will give the
;             same result as
;                   res = DATA1 - DATA2
;
;       STRICT: set if you want the precision to be specified in
;               absolute value. By default the
;               relative precision is used, that is difference is
;               compared to:
;
;                   EPS * MAX( ABS(VALUE1), ABS(VALUE2) )
;
;               Setting /Strict, differences are compared to EPS.
;
; LIMITATION -
;
; MODIFICATION HISTORY: phs, 11/08/06 - v1.0
;
;-

FUNCTION FLOAT_DIFF, DATA1,  DATA2,  $
          EPS=EPS, STRICT=STRICT, DOUBLE=DOUBLE, _extra=extra


  ;; on error return to caller
  on_error, 2


  ;; Basic inputs checking
  dbl = keyword_set(double)
  eps = n_elements(eps) eq 0 ? (machar(double=dbl)).eps : eps[0]

  if n_params() ne 2 then begin
    message, 'not enough input to FLOAT_DIFF... return NaN',  $
      /continue
    return, dbl ? !values.d_nan : !values.f_nan
  endif


  ;; Differences (zero if below accuracy)
  valid = keyword_set(strict) ? EPS : EPS*(abs(data1) > abs(data2))
  return, ( abs(data1-data2) gt valid ) * (data1-data2)
```

END