

---

Subject: Patch to userlib/deriv.pro to add MAX\_VALUE keyword

Posted by [rfinch](#) on Fri, 24 Dec 1993 17:22:08 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Following context diff, when run through patch, will add the MAX\_VALUE keyword to the userlib function deriv.pro. However, I still believe it would be better if IDL had the fundamental notion of missing values built-in to their basic mathematics; IEEE NaN might be a good choice. If IDL understood NaN's, you wouldn't even need keywords for missing values or max\_value (presuming that most usage of MAX\_VALUE really is for missing values).

\*\*\* deriv.pro Fri Dec 24 09:19:35 1993

--- deriv.pro~ Fri Dec 24 09:20:03 1993

\*\*\*\*\*

\*\*\* 1,6 \*\*\*\*

! Function Deriv, X, Y, max\_value=max\_value

;

; NAME:

; DERIV

--- 1,6 ----

! Function Deriv, X, Y

;

; NAME:

; DERIV

\*\*\*\*\*

\*\*\* 13,20 \*\*\*\*

; Numerical analysis.

;

; CALLING SEQUENCE:

! ; Dy = Deriv(Y, max\_value=mv) ;Dy(i)/di, point spacing = 1.

! ; Dy = Deriv(X, Y, max\_value=mv) ;Dy/Dx, unequal point spacing.

;

; INPUTS:

; Y: Variable to be differentiated.

--- 13,20 ----

; Numerical analysis.

;

; CALLING SEQUENCE:

! ; Dy = Deriv(Y) ;Dy(i)/di, point spacing = 1.

! ; Dy = Deriv(X, Y) ;Dy/Dx, unequal point spacing.

;

; INPUTS:

; Y: Variable to be differentiated.

\*\*\*\*\*

\*\*\* 22,30 \*\*\*\*

; spacing for Y (i.e., X(i) = i) is assumed.

;

```

; OPTIONAL INPUT PARAMETERS:
! ;max_value: Maximum value; original values greater than this and
! ; their neighbors will be restored to their original value,
! ; not the first derivative.
;
;
; OUTPUTS:
; Returns the derivative.
--- 22,28 ----
; spacing for Y (i.e., X(i) = i) is assumed.
;
;
; OPTIONAL INPUT PARAMETERS:
! ; As above.
;
;
; OUTPUTS:
; Returns the derivative.
*****
*** 51,78 ****
n = n_elements(x)
if n lt 3 then message, 'Parameters must have at least 3 points'

- count=0
if n_params(0) ge 2 then begin
  if n ne n_elements(y) then message, 'Vectors must have same size'
-   IF KEYWORD_SET(max_value) THEN BEGIN
-     mv_ndx = where(y GT max_value OR $
-               shift(y, -1) GT max_value OR $
-               shift(y, +1) GT max_value, count)
-   ENDIF
  d = float(shift(y,-1) - shift(y,1))/(shift(x,-1) - shift(x,1))
  d(0) = (-3.0*y(0) + 4.0*y(1) - y(2))/(x(2)-x(0))
  d(n-1) = (3.*y(n-1) - 4.*y(n-2) + y(n-3))/(x(n-1)-x(n-3))
-   IF count GT 0 THEN d(mv_ndx) = y(mv_ndx)
  end else begin
-   IF KEYWORD_SET(max_value) THEN BEGIN
-     mv_ndx = where(x GT max_value OR $
-               shift(x, -1) GT max_value OR $
-               shift(x, +1) GT max_value, count)
-   ENDIF
  d = (shift(x,-1) - shift(x,1))/2.
  d(0) = (-3.0*x(0) + 4.0*x(1) - x(2))/2.
  d(n-1) = (3.*x(n-1) - 4.*x(n-2) + x(n-3))/2.
-   IF count GT 0 THEN d(mv_ndx) = x(mv_ndx)
  endelse
  return, d
end
--- 49,63 ----
--
Ralph Finch 916-653-8268 voice

```

---

Subject: Patch #2 to userlib/deriv.pro to add MAX\_VALUE keyword

Posted by [rfinch](#) on Tue, 04 Jan 1994 17:56:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

Second try. First attempt didn't work for certain locations of where the max values were. Maybe this will ;-)

This patch is for the original deriv.pro, NOT the previous patch.

```
*** deriv.pro 1994/01/04 17:46:55
```

```
--- deriv.pro~ 1993/12/24 15:29:48
```

```
*****
```

```
*** 1,6 ***
```

```
! ; $Id: deriv.pro,v 1.0 1993/12/24 15:29:48 rfinch Exp $
```

```
! Function Deriv, X, Y
```

```
  ;+
```

```
  ; NAME:
```

```
  ; DERIV
```

```
--- 1,6 ---
```

```
! ; $Id: deriv.pro,v 1.0.1.6 1994/01/04 17:46:55 rfinch Exp $
```

```
! Function Deriv, X, Y, max_value=max_value
```

```
  ;+
```

```
  ; NAME:
```

```
  ; DERIV
```

```
*****
```

```
*** 13,20 ***
```

```
  ; Numerical analysis.
```

```
  ;
```

```
  ; CALLING SEQUENCE:
```

```
! ; Dy = Deriv(Y) ;Dy(i)/di, point spacing = 1.
```

```
! ; Dy = Deriv(X, Y) ;Dy/Dx, unequal point spacing.
```

```
  ;
```

```
  ; INPUTS:
```

```
  ; Y: Variable to be differentiated.
```

```
--- 13,20 ---
```

```
  ; Numerical analysis.
```

```
  ;
```

```
  ; CALLING SEQUENCE:
```

```
! ; Dy = Deriv(Y, max_value=mv) ;Dy(i)/di, point spacing = 1.
```

```
! ; Dy = Deriv(X, Y, max_value=mv) ;Dy/Dx, unequal point spacing.
```

```
  ;
```

```
  ; INPUTS:
```

```

; Y: Variable to be differentiated.
*****
*** 22,28 ***
;   spacing for Y (i.e., X(i) = i) is assumed.
;
;
; OPTIONAL INPUT PARAMETERS:
! ; As above.
;
;
; OUTPUTS:
; Returns the derivative.
--- 22,30 ----
;   spacing for Y (i.e., X(i) = i) is assumed.
;
;
; OPTIONAL INPUT PARAMETERS:
! ; max_value: Maximum value; original values greater than this and
! ;   their neighbors will be restored to the first large value
! ;   found, not the first derivative.
;
;
; OUTPUTS:
; Returns the derivative.
*****
*** 49,63 ***
--- 51,88 ----
n = n_elements(x)
if n lt 3 then message, 'Parameters must have at least 3 points'

+ count=0
if n_params(0) ge 2 then begin
if n ne n_elements(y) then message, 'Vectors must have same size'
+   IF KEYWORD_SET(max_value) THEN BEGIN
+     mv_ndx = where(y GT max_value, count)
+     IF count GT 0 THEN BEGIN
+       big_value = y(mv_ndx(0))
+       mv_ndx=[mv_ndx-1, mv_ndx+1]
+       mv_ndx = mv_ndx(where(mv_ndx GT 0 AND mv_ndx LT n))
+       IF y(2) GT max_value THEN mv_ndx = [mv_ndx, 0]
+       IF y(n-3) GT max_value THEN mv_ndx = [mv_ndx, n-1]
+     ENDIF
+   ENDIF
d = float(shift(y,-1) - shift(y,1))/(shift(x,-1) - shift(x,1))
d(0) = (-3.0*y(0) + 4.0*y(1) - y(2))/(x(2)-x(0))
d(n-1) = (3.*y(n-1) - 4.*y(n-2) + y(n-3))/(x(n-1)-x(n-3))
+   IF count GT 0 THEN d(mv_ndx) = big_value
end else begin
+   IF KEYWORD_SET(max_value) THEN BEGIN
+     mv_ndx = where(x GT max_value, count)
+     IF count GT 0 THEN BEGIN
+       big_value = x(mv_ndx(0))

```

```

+      mv_ndx=[mv_ndx-1, mv_ndx+1]
+      mv_ndx = mv_ndx(where(mv_ndx GT 0 AND mv_ndx LT n))
+      IF x(2) GT max_value THEN mv_ndx = [mv_ndx, 0]
+      IF x(n-3) GT max_value THEN mv_ndx = [mv_ndx, n-1]
+      ENDIF
+      ENDIF
d = (shift(x,-1) - shift(x,1))/2.
d(0) = (-3.0*x(0) + 4.0*x(1) - x(2))/2.
d(n-1) = (3.*x(n-1) - 4.*x(n-2) + x(n-3))/2.
+      IF count GT 0 THEN d(mv_ndx) = big_value
      endelse
      return, d
      end
--

```

Ralph Finch 916-653-8268 voice

rfinch@venice.water.ca.gov 916-653-6077 fax

Any opinions expressed are my own; they do not represent the DWR

---