
Subject: Re: Gradient of two dimensional field
Posted by [Andy Loughe](#) on Wed, 19 Feb 1997 08:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

```
>
> Andy Loughe <afl@cdc.noaa.gov> writes in response to Wilpert Martin:
>
>>> we want to determine the electrical field from a given potential,
>>> i.e. we have to calculate the gradient of a two dimensional array.
>
>> I would think that the shift function (used twice)
>> could be used to do this.
>
> Andy, do you think you could you give those of us who are
> wondering about this just a small example of what you
> mean? Thanks!
>
> David
```

A simple solution was posted already, but I think it accomplished the task via one-sided differences. To use two-sided differences one may wish to use the shift function twice for each vector component, then use shift again to handle the pesky boundaries (assuming non-cyclic boundary values), then compute the "magnitude" of the gradient, and return this value.

Here is a solution that I have not had time to debug.
If improvements need to be made, please let me know...

```
; Compute the vector magnitude of the gradient.
;
;
; Andrew F. Loughe (afl@cdc.noaa.gov)
;
```

```
function grad, data, x, y
```

```
sz = size(data) & im = sz(1) & jm = sz(2)
```

```
if ( N_params() eq 0 ) then message, ' grad_data = grad(data, x, y)'
if ( sz(0) ne 2 ) then message, 'Input data must be 2-D'
if ( N_elements(x) eq 0 ) then x = indgen(im) + 1
if ( N_elements(y) eq 0 ) then y = jm - indgen(jm)
```

```
; Begin here
```

```

x2 = x # replicate(1, jm)
y2 = replicate(1,jm) # y

; i-component
if (x(1) gt x(0)) then dx = shift(x2, -1, 0) - shift(x2, 1, 0)
if (x(1) lt x(0)) then dx = shift(x2, 1, 0) - shift(x2, -1, 0)
grad_x = ( shift(data, -1, 0) - shift(data, 1, 0) ) / dx

; j-component
if (y(1) lt y(0)) then dy = shift(y2, 0, 1) - shift(y2, 0, -1)
if (y(1) gt y(0)) then dy = shift(y2, 0, -1) - shift(y2, 0, 1)
grad_y = ( shift(data, 0, 1) - shift(data, 0, -1) ) / dy

; But for non-cyclic boundary values we still have a problem...
; Take care of the outer rows and columns
grad_y(*,jm-1) = ( data(*,jm-1) - data(*,jm-2) ) / $
                  ( y2(*,jm-1) - y2(*,jm-2) )
grad_y(*,0)    = ( data(*,1) - data(*,0) ) / ( y2(*,1) - y2(*,0) )

grad_x(im-1,*) = ( data(im-1,*) - data(im-2,*) ) / $
                  ( x2(im-1,*) - x2(im-2,*) )
grad_x(0,*)    = ( data(1,*) - data(0,*) ) / ( x2(1,*) - x2(0,*) )

grad = sqrt(grad_x^2 + grad_y^2)

return, grad

end

```
