Subject: Gradient of two dimensional field
Posted by Wilpert_Martin on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

Hi everybody,

we want to determine the electrical field from a given potential,
i.e. we have to calculate the gradient of a two dimensional array.

Has anybody a idl-pvwave procedure to do this task?

Thank you in advance
Martin Wilpert

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

Wilpert_Martin wrote:
>
> Hi everybody,
>
> we want to determine the electrical field from a given potential,
> i.e. we have to calculate the gradient of a two dimensional array.
>
> Has anybody a idl-pvwave procedure to do this task?
>
> Thank you in advance
> Martin Wilpert

I would think that the shift function (used twice)
could be used to do this.

--
Andrew F. Loughe |
afl@cdc.noaa.gov
University of Colorado, CIRES Box 449 |
http://cdc.noaa.gov/~afl
Boulder, CO 80309-0449 | phn:(303)492-0707
fax:(303)497-7013

"I do not feel obliged to believe that the same God who has endowed us
with
sense, reason, and intellect has intended us to forego their use."
-Galileo

Subject: Re: Gradient of two dimensional field
Posted by davidf on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

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

David Fanning, Ph.D.
Fanning Software Consulting
2642 Bradbury Court, Fort Collins, CO 80521
Phone: 970-221-0438 Fax: 970-221-4762
E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com

Subject: Re: Gradient of two dimensional field
Posted by stockwell on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

In article <yckohdg23jj.fsf@hmi.de> Wilpert_Martin <wilpert-m@hmi.de> writes:
> From: Wilpert_Martin <wilpert-m@hmi.de>
> Subject: Gradient of two dimensional field
> Date: 19 Feb 1997 16:56:00 +0100

> Hi everybody,
> we want to determine the electrical field from a given potential,
> i.e. we have to calculate the gradient of a two dimensional array.
> Has anybody a idl-pvwave procedure to do this task?
> Thank you in advance
> Martin Wilpert

The gradient is just (d/dx , d/dy)f(x,y)
so you can take the deriv() function on the rows and then the columns
"The DERIV function performs numerical differentiation using 3-point,

Lagrangian interpolation and returns the derivative."

so do this:

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;; array holds your potential function of size (len,len)
dx = fltarr(len,len)
dy = fltarr(len,len)
; my apologies for the for loop
for i = 0,len-1 do   dx(*,i) = deriv(array(0:len-1,i))
for i = 0,len-1 do   dy(i,*) = deriv(phase(i,0:len-1))
endfor
vel,dx,dy
end
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

dx holds the x component of the gradient
dy holds the y component of the gradient

cheers,
bob

=== =========
R. G. Stockwell | Space and Atmospheric Research Group
Phone: 519-679-2111 X6411 | University of Western Ontario
Fax: 519-661-3129 | London, Ontario, Canada
....
email: stockwell@uwo.ca
WWW: http://www.sar.physics.uwo.ca/~stockwell/time_series.html

" If I have not seen as far as other men, it's
because giants were standing on my shoulders"
=== =========

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 soulution 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
```

Subject: Re: Gradient of two dimensional field
Posted by bowman on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

In article <brian.jackel.106.19DE53E0@uwo.ca>, brian.jackel@uwo.ca (Brian Jackel) wrote:

```
>>>  we want to determine the electrical field from a given potential,
>>>  i.e. we have to calculate the gradient of a two dimensional array.
>>>
>>>  Has anybody a idl-pvwave procedure to do this task?
>
>> I would think that the shift function (used twice)
>> could be used to do this.
>
>  Or even just
>
>   dx=  a(1:*,*) - a
>   dy=  a(*,1:*) - a
>
>  or
>
>   dx= a(1:n-1,*) - a(0:n-2,*)
>   dy= a(*,1:m-1) - a(*,0:m-2)
>
>  if "a" has dimensions of (n,m).  When doing it the first way IDL takes
```

> care of the different array sizes, with no perceptible performance hit.
> The second way is perhaps a bit easier to read. Is this what you (the
> original poster) were after?

You may want to use centered differences, i.e.

dzdx = (SHIFT(z,-1, 0) - SHIFT(z, 1, 0))/(2.0*dx)
dzdy = (SHIFT(z, 0,-1) - SHIFT(z, 0, 1))/(2.0*dy)

(I trust the compiler is smart enough to convert the division to
multiplication.)

Don't forget to fix the edges, i.e., use uncentered differences for the
normal component or whatever is appropriate for your problem.

Ken

--
Kenneth P. Bowman, Assoc. Prof. 409-862-4060
Department of Meteorology 409-862-4132 fax
Texas A&M University bowman@csrp.tamu.edu
College Station, TX 77843-3150

Subject: Re: Gradient of two dimensional field
Posted by brian.jackel on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

In article <330B44DD.647C@cdc.noaa.gov> Andy Loughe <afl@cdc.noaa.gov> writes:
> Wilpert_Martin wrote:

>> we want to determine the electrical field from a given potential,
>> i.e. we have to calculate the gradient of a two dimensional array.
>>
>> Has anybody a idl-pvwave procedure to do this task?

> I would think that the shift function (used twice)
> could be used to do this.

Or even just

dx= a(1:*,*) - a
dy= a(*,1:*) - a

or

```
 dx= a(1:n-1,*) - a(0:n-2,*)
 dy= a(*,1:m-1) - a(*,0:m-2)
```

if "a" has dimensions of (n,m). When doing it the first way IDL takes care of the different array sizes, with no perceptible performance hit. The second way is perhaps a bit easier to read. Is this what you (the original poster) were after?

Brian Jackel

Subject: Re: Gradient of two dimensional field
Posted by Mirko Vukovic on Wed, 19 Feb 1997 08:00:00 GMT
View Forum Message <> Reply to Message

Andy Loughe wrote:
>
> Wilpert_Martin wrote:
>>
>> Hi everybody,
>>
>> we want to determine the electrical field from a given potential,
>> i.e. we have to calculate the gradient of a two dimensional array.
>>
>> Has anybody a idl-pvwave procedure to do this task?
>>
>> Thank you in advance
>>   Martin Wilpert
>
> I would think that the shift function (used twice)
> could be used to do this.
>
> --
> Andrew F. Loughe |
> afl@cdc.noaa.gov
> University of Colorado, CIRES Box 449 |
> http://cdc.noaa.gov/~afl
> Boulder, CO 80309-0449 | phn:(303)492-0707
> fax:(303)497-7013
> -- ---------------
> "I do not feel obliged to believe that the same God who has endowed us
> with
> sense, reason, and intellect has intended us to forego their use."
> -Galileo
or, convolve with a kernel that will give you the gradient.
(we are talking two kernels, one for each direction)
--
Mirko Vukovic, Ph.D   3075 Hansen Way M/S K-109

Varian Associates Palo Alto, CA, 94304
415/424-4969 mirko.vukovic@varian.grc.com

Subject: Re: Gradient of two dimensional field
Posted by Andy Loughe on Mon, 03 Mar 1997 08:00:00 GMT
View Forum Message <> Reply to Message

Wilpert_Martin wrote:
>
> Hi everybody,
>
> we want to determine the electrical field from a given potential,
> i.e. we have to calculate the gradient of a two dimensional array.
>
> Has anybody a idl-pvwave procedure to do this task?
>
> Thank you in advance
> Martin Wilpert

The gradient of a 2-D field is a vector field, this function computes the
i,j vector components of the gradient, and returns the *magnitude* of that vector field. It performs one-sided differences at the lateral boundaries, which is an assumption that your field does not have a cyclical boundary condition. Let me know if it works for you!

```
;  Compute the magnitude of the vector 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 outter 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
```

--
Andrew F. Loughe |
afl@cdc.noaa.gov
University of Colorado, CIRES Box 449 |
http://cdc.noaa.gov/~afl
Boulder, CO 80309-0449 | phn:(303)492-0707
fax:(303)497-7013
 -- ----------
"I do not feel obliged to believe that the same God who has endowed us
with
sense, reason, and intellect has intended us to forego their use."
-Galileo