Subject: Re: A better way to find a dip (local minimum with certain conditions)
Posted by rogass on Wed, 16 Dec 2009 21:39:31 GMT

Hi David,
as i understood the following might work (untested):

```
FUNCTION finddip,array,minim,range=range
range=keyword_set(range)? 0>range<n_elements(array) : 4
  minimum = (min(array-minim,wheremin,/nan))[0] ; [0]-> to get only
the first match
if wheremin ne -1 then begin
 ar = array[-1>(wheremin-range):(wheremin+range)<(n_elements
(array)-1)]
 if max(((arr=ar[sort(ar)]-ar))) eq min(arr) then return,wheremin
endif else return,-1
end
```

Maybe it works and maybe that the principle behind gives you a hint
for solving the problem

Regards

CR

Subject: Re: A better way to find a dip (local minimum with certain conditions)
Posted by Giorgio on Wed, 16 Dec 2009 23:58:47 GMT

On Dec 16, 1:39 pm, chris <rog...@googlemail.com> wrote:
> Hi David,
> as i understood the following might work (untested):
>
> FUNCTION finddip,array,minim,range=range
> range=keyword_set(range)? 0>range<n_elements(array) : 4
>   minimum   =   (min(array-minim,wheremin,/nan))[0] ; [0]-> to get only
> the first match
> if wheremin ne -1 then begin
>   ar   =   array[-1>(wheremin-range):(wheremin+range)<(n_elements
> (array)-1)]
>   if max(((arr=ar[sort(ar)]-ar))) eq min(arr) then return,wheremin
> endif else return,-1
> end
>
> Maybe it works and maybe that the principle behind gives you a hint
> for solving the problem
>

> Regards
>
> CR

Hi,
 Following CR, this is a small program to find all the local minima.
 I did not test it yet.

```
function findip, array,minim
  range = 4
  nel = n_elements(array)
  ; look for candidates
  candidates = where(array GT minim)
  mins = candidates
  ; look which of them is a local minima
  FOR i =0, n_elements(candidates)-1 do begin
    localmin = where(array[((candidates[i]-range)>0):((candidates[i]
+range)<(nel-1) GT array[candidates[i]])
    IF localmin GE 0 THEN mins[i] = 0
  endfor
  return, where(mins NE 0)
end
```

Subject: Re: A better way to find a dip (local minimum with certain conditions)
Posted by DavidPS on Mon, 21 Dec 2009 18:44:33 GMT

Thanks to all!! I've seen all your suggestions and you have me
pointed me through the way I wanted.
Finally, what I've got from Jim P and James version and adding stuff
here and there is:

```
;==================== CODE========================
FUNCTION finddip5,array,minim=minim,range=range

 range=KEYWORD_SET(range) ? 1>range<(n_elements(array)/2-1) : 4

 ; Find the negative gradients.  We're only interested in the sign of
the gradient, not the magnitude.
 c1 = fix(convol(array, [-1, 1]) GT 0)

 ; Find the positive gradients, and invert the sign.
 c2 = -fix(convol(array, [1, -1]) GT 0)

 ; Combine the two Filter for four negative gradients, a zero, and
four positive gradients.
 minima = where(fix(convol(c1+c2, [replicate(-1,range+1),replicate
```

```
(1,range)]) eq 2*range+1))

 ;Look for those greater than minim if set.
 IF (minima[0] NE -1) AND (KEYWORD_SET(minim)) THEN BEGIN
   lab = where(array[minima] ge minim,nlab)
   minima=(nlab GT 0) ? minima[lab] : -1
 ENDIF
 RETURN,minima
END
;================= END CODE=====================
```

Which is very clever, scalable and a lot faster than what I had.
However... I still have a small problem, which I've been trying to
solve using the same method... but I have not found the way to do it.

In my first example I was interested also on those dips where on the
decreasing or increasing part of it there's repeated values. For
example consider these arrays:
[-5,-4,18,12,12,3,1,-0.5,7,11,13,30,29,5] or
[-5,-4,18,16,12,3,1,-0.5,7,11,11,30,29,5]
or a mix of both:
[-5,-4,18,12,12,3,1,-0.5,7,11,11,30,29,5]

I've managed to get when there's on one side or another of the dip,
but not in both... making ones in C1 and C1 everything bigger than
-0.0001 and then making 1 the values gt 2 on c1+c2.
does someone have any clue?

Thank you to everybody again!!

David