Subject: Re: Help with Histogram...
Posted by peter.albert@gmx.de on Wed, 28 Sep 2005 08:26:05 GMT
View Forum Message <> Reply to Message

Hi again,

assuming that the above assumption is correct, I could propose
something to start with. It's just a start, as there is still one
for-loop, but you can get rid of the ugly where statements:

let's first create some test data:

```
;I'll use 8 cloud layers, each 300 meters thick:
base = findgen(8) * 1000.
top = base + 300.

;Here is the profile variable:
profile = intarr(60)

; And here come the height levels, just an arbitrary set
; of numbers, unevenly spaced.
; Mind that this array has 61 elements, as you need one upper
; and lower boundary for each layer in "profile"

height_level = [0., exp((findgen(60)-30) / 15.)] * 1300.

; I'll use interpol for finding the correct indices, so we need
; the abscissa values for "height_levels":

x = indgen(61)

; Now, with interpol, we can find those  layers of "height_level"
; which will be completely covered by each combination
; of cloud base and top value; using ceil and floor
; is equivalent to the "...GE base[0] AND ... LE top[0] ..."
; in your example:

base_idx = ceil(interpol(x, height_level, base))
top_idx = floor(interpol(x, height_level, top))

; Fine so far, but now I still need a for-loop for filling
; "profile":

for i = 0, 7 do $
  if base_idx[i] le top_idx[i] then $
    profile[base_idx[i]:top_idx[i]] = 1
```

Note that the "interpol" way also works if "base" and "top" are not simple 8-element vectors but rather arrays of size(number_of_profiles, 8). In that case, "base_idx" and "top_idx" are of the same size, but here the for-loops becomes even more uglier.

Regards,

Peter