Subject: Merits of different ways of 'extending' arrays
Posted by Andy Sayer on Thu, 29 Aug 2013 15:44:51 GMT
View Forum Message <> Reply to Message

Hi all,

I am writing some code where I am loading a whole bunch of files one by one, querying them from valid data, and putting valid data from each file into an array (for later use). I don't know ahead of time how many files there will be, or how many valid data points there will be in a file.

The way I have written my code so far is like this:

```
var_1_arr=[!values.f_nan]
var_2_arr=[!values.f_nan]
var_3_arr=[!values.f_nan]

f=file_search( [path and identifier to files],count=nfiles)

for i=0l,nfiles-1 do begin

  [load contents of file f[i] into a structure]

  is_valid=where(blah blah,n_valid)

  if n_valid gt 0 then begin
    var_1_arr=[var_1_arr,f.var_1[is_valid]]
    var_2_arr=[var_2_arr,f.var_2[is_valid]]
    var_3_arr=[var_3_arr,f.var_3[is_valid]]

  endif

endfor
```

So, hopefully you get the idea. I only have a small subset of the test data to work with at the moment (the rest is a few months off).

It occurs to me that I could code it something like this:

```
max_points=1.e7

var_1_arr=fltarr(max_points)
var_1_arr(*)=!values.f_nan
var_2_arr=var_1_arr
var_3_arr=var_1_arr

f=file_search( [path and identifier to files],count=nfiles)

ctr=0l
```

```
for i=0l,nfiles-1 do begin

  [load contents of file f[i] into a structure]

  is_valid=where(blah blah,n_valid)

  if n_valid gt 0 then begin
    var_1_arr[ctr:ctr+n_valid]=f.var_1[is_valid]
    var_2_arr[ctr:ctr+n_valid]=f.var_2[is_valid]
    var_3_arr[ctr:ctr+n_valid]=f.var_3[is_valid]
    ctr=ctr+n_valid
  endif

endfor
```

This has the drawback that I have to know in advance the maximum number of data points I could have (but I can set max_points to some arbitrary high number to be safe). Does anyone know whether any one method is better/less memory-intensive than the other, when it comes to largeish data volumes (tens of millions of points)? I only have a few percent of the final data so far, so am interested in the likely merits of each method. Google didn't help but perhaps I was using the wrong search keywords.

In case relevant, this is IDL 7.1.1. or 8.2.2.

Thanks,

Andy
