Subject: Re: read_ascii for more than one file
Posted by Conor on Thu, 23 Aug 2007 16:56:27 GMT
View Forum Message <> Reply to Message

On Aug 23, 12:24 pm, b...@uni-mainz.de wrote:
> Hello,
>
> i'd like to read arrays (3 columns, 2072758 lines) from more than one
> file do get some kind of data(k) where k (k=146) is the number of
> files and data is the array.
>
> For one file it works quite well with:
>
> data = read_ascii(file) & data = data.(0)
> g = data[0, *]
> r = data[1,*]
> nir = data[2,*]
>
> For this code i do not need to declare data, g, r and nir
>
> But i do not know how i can manage it for k files (without getting
> error messages).
>
> Britta

Well, why not just create a gigantic 3-d array and fill the array one
by one?

```
everything = fltarr(3,2072758,146)

for i=0,146-1 do begin
  file = 'file' + string(i+1,format='(i0)')
  data = read_ascii(file) & data = data.(0)
  everything[*,*,i] = data
endfor
```

In this case, the program expects the files to be named something
like: 'file1', 'file2', ... 'file146'. Obviously, you would have to
adjust the names accordingly. Worse comes to worse you could always
specify each one individually in a string array and extract the
filename as you loop through.

Subject: Re: read_ascii for more than one file
Posted by Jean H. on Thu, 23 Aug 2007 17:00:28 GMT
View Forum Message <> Reply to Message

bmey@uni-mainz.de wrote:
> Hello,
>
> i'd like to read arrays (3 columns, 2072758 lines) from more than one
> file do get some kind of data(k) where k (k=146) is the number of
> files and data is the array.
>
> For one file it works quite well with:
>
> data = read_ascii(file) & data = data.(0)
> g = data[0, *]
> r = data[1,*]
> nir = data[2,*]
>
> For this code i do not need to declare data, g, r and nir
>
> But i do not know how i can manage it for k files (without getting
> error messages).
>
>
> Britta

If you want everything appended to the end of the R G NIR arrays, you can do something like this, for the 2nd files and later:

r = [[r], [data[1,*]]]
g = [[g], [data[0,*]]]
nir = [[nir], [data[2,*]]]

Otherwise, you can declare r g nir as arrays of pointer (size = k), each of them pointing to your data... you can easily fill them up in a loop, when reading your files.

Jean

Subject: Re: read_ascii for more than one file
Posted by britta.mey on Fri, 24 Aug 2007 06:33:59 GMT
View Forum Message <> Reply to Message

On Aug 23, 6:56 pm, Conor <cmanc...@gmail.com> wrote:
> On Aug 23, 12:24 pm, b...@uni-mainz.de wrote:
>
>
>
>> Hello,
>
>> i'd like to read arrays (3 columns, 2072758 lines) from more than one

>> file do get some kind of data(k) where k (k=146) is the number of
>> files and data is the array.
>
>> For one file it works quite well with:
>
>> data = read_ascii(file) & data = data.(0)
>> g = data[0, *]
>> r = data[1,*]
>> nir = data[2,*]
>
>> For this code i do not need to declare data, g, r and nir
>
>> But i do not know how i can manage it for k files (without getting
>> error messages).
>
>> Britta
>
> Well, why not just create a gigantic 3-d array and fill the array one
> by one?
>
> everything = fltarr(3,2072758,146)
>
> for i=0,146-1 do begin
> file = 'file' + string(i+1,format='(i0)')
> data = read_ascii(file) & data = data.(0)
> everything[*,*,i] = data
> endfor
>
> In this case, the program expects the files to be named something
> like: 'file1', 'file2', ... 'file146'. Obviously, you would have to
> adjust the names accordingly. Worse comes to worse you could always
> specify each one individually in a string array and extract the
> filename as you loop through.

Hello,

when i try this suggestion i get the error message: "%Array has too
many elements." at the line "everything = fltarr(3,2072758,146)".

Is there a possibility to fix this error?

Britta

Subject: Re: read_ascii for more than one file

Posted by Maarten[1] on Fri, 24 Aug 2007 08:06:03 GMT
View Forum Message <> Reply to Message

On Aug 24, 6:33 am, britta....@gmail.com wrote:
> On Aug 23, 6:56 pm, Conor <cmanc...@gmail.com> wrote:
>
>
>
>> On Aug 23, 12:24 pm, b...@uni-mainz.de wrote:
>
>>> Hello,
>
>>> i'd like to read arrays (3 columns, 2072758 lines) from more than one
>>> file do get some kind of data(k) where k (k=146) is the number of
>>> files and data is the array.
>
>>> For one file it works quite well with:
>
>>> data = read_ascii(file) & data = data.(0)
>>> g = data[0, *]
>>> r = data[1,*]
>>> nir = data[2,*]
>
>>> For this code i do not need to declare data, g, r and nir
>
>>> But i do not know how i can manage it for k files (without getting
>>> error messages).
>
>>> Britta
>
>> Well, why not just create a gigantic 3-d array and fill the array one
>> by one?
>
>> everything = fltarr(3,2072758,146)
>
>> for i=0,146-1 do begin
>> file = 'file' + string(i+1,format='(i0)')
>> data = read_ascii(file) & data = data.(0)
>> everything[*,*,i] = data
>> endfor
>
>> In this case, the program expects the files to be named something
>> like: 'file1', 'file2', ... 'file146'. Obviously, you would have to
>> adjust the names accordingly. Worse comes to worse you could always
>> specify each one individually in a string array and extract the
>> filename as you loop through.
>
> Hello,
>

> when i try this suggestion i get the error message: "%Array has too
> many elements." at the line "everything = fltarr(3,2072758,146)".
>
> Is there a possibility to fix this error?

That is a flt array with about 900 million points, or just under 4 GB
of memory. I'd say: get more memory and use the 64 bit version of IDL.
Or rethink your analysis, and perform it in a way that does not
require all data in memory at teh same time. I'd vote for the latter,
as it is probably faster as well.

Maarten
