Subject: Handle big data files
Posted by lucesmm on Mon, 02 Nov 2015 01:44:01 GMT
View Forum Message <> Reply to Message

Hello
I have been working with big data files (~18MB each)

And everytime I need to handle anything with the data, mostly plotting, takes a long time.
So I was trying to put them all in one file per year (the data is arrange by date)
But just running this takes a few hours.

```
PRO Save_1year
  GET_LUN, outlun
  Openw, outlun, 'name.txt'
  Year = ''
  READ, Year, PROMPT='Enter Year:'
  openw, outlun, 'C:data\year'+ Year+'.txt', /get_lun

  files=FILE_SEARCH('C:data'+Year+'*', COUNT=nfiles)
  Print, files
  data = []

  FOR i=0,nfiles-1 DO BEGIN
    filename=files[i]
    fileNumber = STRMID(filename,51,5)
    nlines = FILE_LINES(files[i])
    thisFile = files[i]

    ;line=fltarr(17,1)
    OpenR, inLun, thisFile, /Get_Lun
    SKIP_LUN, inLun, 1, /LINES
    while not EOF(inlun) do begin & $
      line=make_array(17,1, type=5)
    Readf, inLun, line, FORMAT='...'
    data=[[data],[line]]

  Endwhile
  Free_Lun, inLun

ENDFOR
PrintF, outLun, data, FORMAT= '...'
END
```

I was wondering if there is an easier way to handle this data, I don't know anything about SAVE files, may be this is easier?
Please help

Thank you

Luz Maria

Subject: Re: Handle big data files
Posted by Helder Marchetto on Mon, 02 Nov 2015 09:13:39 GMT
View Forum Message <> Reply to Message

Hi,
I think that this line is responsible for making things slow:

```
data=[[data],[line]]
```

If the array data gets to be lon, then it will take a long time to copy the previous data to a new variable and add one element...

You have two options:
1) only valid for IDL version >8.0. Use a list(). before the for use:
data = list()
then instead of data=[[data],[line]] use:
data->add, line
Then at the end:
PrintF, outLun, data->toArray(), FORMAT= '...'

2) it's more complicated, but general. Create the data array looooong, then fill it up. You could also actually guess it's length:

```
nData = 0l
FOR i=0,nfiles-1 DO BEGIN
   nlines = FILE_LINES(files[i])
   nData += nlines-1 ;one line you always disregard
ENDFOR
```

Now create data so that it is long enough:

```
myDataStructure = make_array(17,1, type=5)
data = replicate(myDataStructure, nData)
```

and in the cycle you fill up. You will also need a "fill-up" counter:

```
fillCounter = 0l
FOR...
   ...
   while...
      ...
      data[fillCounter] = line
      fillCounter++
   endwhile
   ...
endfor
```

I hope it helps...

Cheers,
Helder

Subject: Re: Handle big data files
Posted by lucesmm on Mon, 02 Nov 2015 15:02:10 GMT
View Forum Message <> Reply to Message

Hello Helder
Thank you for your help, I am implementing the first option you suggested. I am having trouble now because one of the columns is calendar format and I am getting this error
PRINTF: Value of Julian date is out of allowed range
:(is there an easy way to solve this, or shoudl I just keep date info in different colums?

HEre is what the data look like and my format:

2014-12-01T00:00:12.905
C(CYI, X ,CMOI02, X ,CDI02,X, CHI02,X, CMI02, X, CSF0)
I am using the same fomat in both reading and writing the data
is this correct?

Now I wanted to ask something else. I have a bunch of columns that I don't really need, is there a way to create a save file just with the date column and the one that I care? (I am guessing this is the easiest version of files to save big data because they are binary)

Thanks again

-Luz Maria
On Monday, November 2, 2015 at 1:13:43 AM UTC-8, Helder wrote:
> Hi,
> I think that this line is responsible for making things slow:
>
> data=[[data],[line]]
>
> If the array data gets to be lon, then it will take a long time to copy the previous data to a new variable and add one element...
>
> You have two options:
> 1) only valid for IDL version >8.0. Use a list(). before the for use:
> data = list()
> then instead of data=[[data],[line]] use:
> data->add, line
> Then at the end:
> PrintF, outLun, data->toArray(), FORMAT= '...'
>

> 2) it's more complicated, but general. Create the data array loooong, then fill it up. You could also actually guess it's length:
> nData = 0l
> FOR i=0,nfiles-1 DO BEGIN
> nlines = FILE_LINES(files[i])
> nData += nlines-1 ;one line you always disregard
> ENDFOR
>
> Now create data so that it is long enough:
> myDataStructure = make_array(17,1, type=5)
> data = replicate(myDataStructure, nData)
>
> and in the cycle you fill up. You will also need a "fill-up" counter:
>
> fillCounter = 0l
> FOR...
> ...
> while...
> ...
> data[fillCounter] = line
> fillCounter++
> endwhile
> ...
> endfor
>
> I hope it helps...
>
> Cheers,
> Helder

Subject: Re: Handle big data files
Posted by Helder Marchetto on Mon, 02 Nov 2015 16:26:12 GMT
View Forum Message <> Reply to Message

On Monday, November 2, 2015 at 4:02:15 PM UTC+1, luc...@gmail.com wrote:
> Hello Helder
> Thank you for your help, I am implementing the first option you suggested. I am having trouble now because one of the columns is calendar format and I am getting this error
> PRINTF: Value of Julian date is out of allowed range
> :(is there an easy way to solve this, or shoudl I just keep date info in different colums?
>
> HEre is what the data look like and my format:
>
> 2014-12-01T00:00:12.905
> C(CYI, X ,CMOI02, X ,CDI02,X, CHI02,X, CMI02, X, CSF0)
> I am using the same fomat in both reading and writing the data
> is this correct?

>
> Now I wanted to ask something else. I have a bunch of columns that I don't really need, is there a way to create a save file just with the date column and the one that I care? (I am guessing this is the easiest version of files to save big data because they are binary)
>
> Thanks again
>
> -Luz Maria
> On Monday, November 2, 2015 at 1:13:43 AM UTC-8, Helder wrote:
>> Hi,
>> I think that this line is responsible for making things slow:
>>
>> data=[[data],[line]]
>>
>> If the array data gets to be lon, then it will take a long time to copy the previous data to a new variable and add one element...
>>
>> You have two options:
>> 1) only valid for IDL version >8.0. Use a list(). before the for use:
>> data = list()
>> then instead of data=[[data],[line]] use:
>> data->add, line
>> Then at the end:
>> PrintF, outLun, data->toArray(), FORMAT= '...'
>>
>> 2) it's more complicated, but general. Create the data array looooong, then fill it up. You could also actually guess it's length:
>> nData = 0l
>> FOR i=0,nfiles-1 DO BEGIN
>> nlines = FILE_LINES(files[i])
>> nData += nlines-1 ;one line you always disregard
>> ENDFOR
>>
>> Now create data so that it is long enough:
>> myDataStructure = make_array(17,1, type=5)
>> data = replicate(myDataStructure, nData)
>>
>> and in the cycle you fill up. You will also need a "fill-up" counter:
>>
>> fillCounter = 0l
>> FOR...
>> ...
>> while...
>> ...
>> data[fillCounter] = line
>> fillCounter++
>> endwhile
>> ...

>> endfor
>>
>> I hope it helps...
>>
>> Cheers,
>> Helder

Hi,
sorry, but I don't know much about dates and Julian in particular. You should either wait for somebody else to answer or repost with new subject.
I don't understand your second question what you mean by "just with the date column and the one that I care". Could you make an example, be more specific?

cheers,
Helder

Subject: Re: Handle big data files
Posted by lucesmm on Mon, 02 Nov 2015 16:36:00 GMT
View Forum Message <> Reply to Message

Hi again,
I worked aorunf the date issue, and now I have year in one column, month in other column, etc.
Also I re did my data with only the number I care about
newline=[line(1),line(2),line(3),line(4),line(5),line(6),lin e(10),line(19),line(20),line(21)]
So now this is how the data looks like
[year, month, day, hour, min, secs, data1, data2, data3, data4]
2003.00 1 1 0.0 0.0 44.2630 19.7620 0.173730 8.0 0.0

But I have another issue now...

When I do
PrintF, outLun, Newdata->toArray(), FORMAT='(...)' I only get the first column onto my array (this is what the file shows):
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003
2003 2003 2003 2003 2003 2003.000000 2003.000000 2003.000000 2003 2003

Subject: Re: Handle big data files
Posted by lucesmm on Mon, 02 Nov 2015 17:03:11 GMT

I solve it!
I need /TRANSPOSE keyword in the toArray()