
Subject: Re: Oooh...! It's harder than I thought!! To get average of each line using only meaningful data

Posted by [Ingo von Borstel](#) on Tue, 24 Jul 2007 08:32:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello,

> test00.txt
> 0 2 4 6 7 0
> 0 0 3 3 2 0
> 1 2 3 4 5 0
> 0 0 0 1 2 0
>
> I need to get the average of each line.
> For example, I have 4 meaningful data in the first line. In this case
> I want to have 4.75 (19/4) instead of 3.167 (19/6).
>
> To do this
> 1. I need to get the number of meaningful data, n, in each line first.
> 2. summation of each line, sum.
> 3. sum/n..., and then print to output.txt
> That's all I need. I believe I can handle step 2 and 3, but step 1 is
> harder than I thought...
>
> Please give me any suggestions. Thank you !!!

Assuming you have a fixed amount of data per line, it should work somewhere along these lines:

```
data_arr = dblarr(N)
OPEN, in_lun, 'test00.txt', /GET_LUN
OPENW, out_lun, 'output.txt', /GET_LUN
WHILE NOT EOF(in_lun) DO BEGIN
; read all your data, also the irrelevant, of one line and store them in
data_arr
; something like this:
data_strg = readf(in_lun,format='(a)'
data_strgs = strsplit(data_strg,' ',/EXTRACT)
n_datastrgs = N_ELEMENTS(data_strgs)
FOR i=0,n_datastrgs-1 DO data_arr[i] = DOUBLE(data_strgs[i])

; now get the relevant ones
where_notzero = WHERE(data_arr NE 0,n_notzero)
average = total(data_arr) / n_notzero

; write away your data to output.txt
PRINTF, out_lun, STRING(average,FORMAT='(f10.5)')
```

ENDWHILE
CLOSE, out_lun
CLOSE, in_lun

Best regards,
Ingo

--

Ingo von Borstel <newsgroups@planetmaker.de>
Public Key: <http://www.planetmaker.de/ingo.asc>

If you need an urgent reply, replace newsgroups by vgap.

Subject: Re: Oooh...! It's harder than I thought!! To get average of each line using only meaningful data

Posted by [weitkamp](#) on Tue, 24 Jul 2007 09:47:02 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Jul 24, 10:32 am, Ingo von Borstel <newsgro...@planetmaker.de>
wrote:

```
> where_notzero = WHERE(data_arr NE 0,n_notzero)
> average = total(data_arr) / n_notzero
```

Or simply,

```
average = TOTAL(data_arr) / TOTAL(data_arr NE 0).
```

Cheers,
Timm

Subject: Re: Oooh...! It's harder than I thought!! To get average of each line using only meaningful data

Posted by [Conor](#) on Tue, 24 Jul 2007 12:47:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Jul 24, 5:47 am, weik...@esrf.fr wrote:

```
> On Jul 24, 10:32 am, Ingo von Borstel <newsgro...@planetmaker.de>
> wrote:
>
>> where_notzero = WHERE(data_arr NE 0,n_notzero)
>> average = total(data_arr) / n_notzero
>
> Or simply,
>
```

```
> average = TOTAL(data_arr) / TOTAL(data_arr NE 0).  
>  
> Cheers,  
> Timm
```

It seems to me that the above two examples both find the average over the entire data array, not each row individually. To do that, I would do something like this:

```
totals = total(data_arr,1)/total(data_arr<1,1)
```

total(data_arr,1) returns a row vector where each element is the total of a single line of data. total(data_arr<1,1) does essentially the same thing, but first it imposes a ceiling on the array, so that no value is greater than 1. Then, when you total across a line, what you get is the total number of non-zero elements. So, you are dividing the sum of every line by the number of non-zero elements in every line (the average). The result is a row vector, not a column vector though. If that's a problem, you can always:

```
totals = reform(totals,1,numlines)
```

Also, I've assumed that data_arr is the same thing as in the above examples. I would make a slight change to the reading part of the above example though. I would change this line:

```
FOR i=0,n_datastrgs-1 DO data_arr[i] = DOUBLE(data_strgs[i])
```

To this:

```
data_arr = double(data_strgs)
```

There's no need for the for loop.

Subject: Re: Oooh...! It's harder than I thought!! To get average of each line using only meaningful data

Posted by edward.s.meinel@aero on Tue, 24 Jul 2007 15:40:26 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Jul 24, 8:47 am, Conor <cmanc...@gmail.com> wrote:

```
>  
> totals = total(data_arr,1)/total(data_arr<1,1)  
>  
> total(data_arr,1) returns a row vector where each element is the total  
> of a single line of data. total(data_arr<1,1) does essentially the  
> same thing, but first it imposes a ceiling on the array, so that no  
> value is greater than 1. Then, when you total across a line, what you
```

> get is the total number of non-zero elements.

Very clever, Conor! It just needs one small change: You have assumed that the total number of non-zero elements is, well, non-zero. If there are no non-zero elements, it will choke, so just do,

```
totals = total(data_arr,1)/(total(data_arr<1,1) > 1)
```

I use the `*/(* > 1)` trick quite frequently.

Ed

Subject: Re: Oooh...! It's harder than I thought!! To get average of each line using only meaningful data

Posted by [Conor](#) on Tue, 24 Jul 2007 15:52:27 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Jul 24, 11:40 am, "edward.s.mei...@aero.org" <mei...@aero.org> wrote:

> On Jul 24, 8:47 am, Conor <cmanc...@gmail.com> wrote:

>

>

>

>> totals = total(data_arr,1)/total(data_arr<1,1)

>

>> total(data_arr,1) returns a row vector where each element is the total
>> of a single line of data. total(data_arr<1,1) does essentially the
>> same thing, but first it imposes a ceiling on the array, so that no
>> value is greater than 1. Then, when you total across a line, what you
>> get is the total number of non-zero elements.

>

> Very clever, Conor! It just needs one small change: You have assumed
> that the total number of non-zero elements is, well, non-zero. If
> there are no non-zero elements, it will choke, so just do,

>

> totals = total(data_arr,1)/(total(data_arr<1,1) > 1)

>

> I use the `*/(* > 1)` trick quite frequently.

>

> Ed

Good call. I always forget to check for things like that :) On a related note, I've also assumed that the data is integer data. This would fail for floats (or, at least for any values between 0 and 1)
