
Subject: Re: 2d min

Posted by [Kenneth P. Bowman](#) on Wed, 12 Jan 2011 22:45:18 GMT

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

In article

<e707ce03-19e9-4ee5-9976-aa4c2d6f3f7a@f35g2000vbl.googlegroups.com>,
Gray <graylikethecolor@gmail.com> wrote:

> I have a 3d array, NxNxM. What I would like is to find the minimum of
> each NxN slice, and note the index of the minimum in the slice. I can
> find my minimum by doing min(min(array,ind1,dim=1),dim=1,ind2), but
> I'm not sure how to turn those two index arrays into the indices that
> I need. Help...?

You could just loop over the M index and process each 2-D slice.
ARRAY_INDICES will get you the index within each slice.

Ken Bowman

Subject: Re: 2d min

Posted by [Gray](#) on Wed, 12 Jan 2011 23:06:47 GMT

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

On Jan 12, 5:45 pm, "Kenneth P. Bowman" <k-bow...@null.edu> wrote:

> In article
> < e707ce03-19e9-4ee5-9976-aa4c2d6f3...@f35g2000vbl.googlegroup s.com >,
>
> Gray <grayliketheco...@gmail.com> wrote:
>> I have a 3d array, NxNxM. What I would like is to find the minimum of
>> each NxN slice, and note the index of the minimum in the slice. I can
>> find my minimum by doing min(min(array,ind1,dim=1),dim=1,ind2), but
>> I'm not sure how to turn those two index arrays into the indices that
>> I need. Help...?
>
> You could just loop over the M index and process each 2-D slice.
> ARRAY_INDICES will get you the index within each slice.
>
> Ken Bowman
Yes, but that will be slow compared to a vectorized solution, is my guess.

Subject: Re: 2d min

Posted by [David Fanning](#) on Wed, 12 Jan 2011 23:44:20 GMT

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

Gray writes:

> Yes, but that will be slow compared to a vectorized solution, is my
> guess.

I learned from Matt Savoie that looking for vectorized solutions
before you needed to was a fool's errand. :-)

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: 2d min

Posted by [rogass](#) on Thu, 13 Jan 2011 07:18:17 GMT

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

On 12 Jan., 23:21, Gray <grayliketheco...@gmail.com> wrote:

> Hi all,

>

> I have a 3d array, NxNxM. What I would like is to find the minimum of
> each NxN slice, and note the index of the minimum in the slice. I can
> find my minimum by doing min(min(array,ind1,dim=1),dim=1,ind2), but
> I'm not sure how to turn those two index arrays into the indices that
> I need. Help...?

>

> Thanks!

>

> --Gray

Hi,

maybe I missed something, but why don't you use something like this:

```
IDL> a=randomn(seed,10,10,5)
```

```
IDL> min=min(a,dimension=3,ind)
```

```
IDL> help,min,ind
```

```
MIN          FLOAT    = Array[10, 10]
```

```
IND          LONG64   = Array[10, 10]
```

```
IDL> ind2=array_indices(size(a,/dimensions),ind,/dimensions)
```

```
IDL> help,ind2
```

```
IND2         LONG64   = Array[3, 100]
```

Is array_indices really to slow with the dimension keyword?

Cheers

CR

Subject: Re: 2d min

Posted by [Gray](#) on Thu, 13 Jan 2011 12:26:19 GMT

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

On Jan 13, 2:18 am, chris <rog...@googlemail.com> wrote:

> On 12 Jan., 23:21, Gray <grayliketheco...@gmail.com> wrote:

>

>> Hi all,

>

>> I have a 3d array, NxNxM. What I would like is to find the minimum of

>> each NxN slice, and note the index of the minimum in the slice. I can

>> find my minimum by doing min(min(array,ind1,dim=1),dim=1,ind2), but

>> I'm not sure how to turn those two index arrays into the indices that

>> I need. Help...?

>

>> Thanks!

>

>> --Gray

>

> Hi,

> maybe I missed something, but why don't you use something like this:

>

> IDL> a=randomn(seed,10,10,5)

> IDL> min=min(a,dimension=3,ind)

> IDL> help,min,ind

> MIN FLOAT = Array[10, 10]

> IND LONG64 = Array[10, 10]

> IDL> ind2=array_indices(size(a,/dimensions),ind,/dimensions)

> IDL> help,ind2

> IND2 LONG64 = Array[3, 100]

>

> Is array_indices really to slow with the dimension keyword?

>

> Cheers

>

> CR

This gets me a NxN array of minima... I want a vector of minima of length M (so the minimum in each plane).

Subject: Re: 2d min

Posted by [Gray](#) on Thu, 13 Jan 2011 13:11:50 GMT

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

On Jan 13, 7:26 am, Gray <grayliketheco...@gmail.com> wrote:

> On Jan 13, 2:18 am, chris <rog...@googlemail.com> wrote:

>

>

>

>

>

>

>

>

>

>> On 12 Jan., 23:21, Gray <grayliketheco...@gmail.com> wrote:

>

>>> Hi all,

>

>>> I have a 3d array, NxNxM. What I would like is to find the minimum of

>>> each NxN slice, and note the index of the minimum in the slice. I can

>>> find my minimum by doing min(min(array,ind1,dim=1),dim=1,ind2), but

>>> I'm not sure how to turn those two index arrays into the indices that

>>> I need. Help...?

>

>>> Thanks!

>

>>> --Gray

>

>> Hi,

>> maybe I missed something, but why don't you use something like this:

>

>> IDL> a=randomn(seed,10,10,5)

>> IDL> min=min(a,dimension=3,ind)

>> IDL> help,min,ind

>> MIN FLOAT = Array[10, 10]

>> IND LONG64 = Array[10, 10]

>> IDL> ind2=array_indices(size(a,/dimensions),ind,/dimensions)

>> IDL> help,ind2

>> IND2 LONG64 = Array[3, 100]

>

>> Is array_indices really to slow with the dimension keyword?

>

>> Cheers

>

>> CR

>

> This gets me a NxN array of minima... I want a vector of minima of

> length M (so the minimum in each plane).

OK, for anyone else, here's what you have to do.

```
IDL> a = randomu(seed,10,10,100)
IDL> minima = min(min(a,ind1,dim=1),ind2,dim=2)
IDL> ind1 -= rebin(indgen(1,100)*10^2.,10,100)
IDL> ind2 -= (indgen(100)*10)
IDL> xind = ind1[ind2,indgen(100)] mod 10
IDL> yind = ind2
```

You have to use ind2 to find the right elements of ind1. Since you get 1d indices from min, you need to subtract off N^2 or N to talk about each plane individually. The reason I wanted to vectorize is that my actual M is $\sim 20k$.

Subject: Re: 2d min

Posted by [rogass](#) on Thu, 13 Jan 2011 13:57:35 GMT

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

On 13 Jan., 14:11, Gray <grayliketheco...@gmail.com> wrote:

> On Jan 13, 7:26 am, Gray <grayliketheco...@gmail.com> wrote:

>

>

>

>

>

>

>

>

>

>> On Jan 13, 2:18 am, chris <rog...@googlemail.com> wrote:

>

>>> On 12 Jan., 23:21, Gray <grayliketheco...@gmail.com> wrote:

>

>>>> Hi all,

>

>>>> I have a 3d array, $N \times N \times M$. What I would like is to find the minimum of

>>>> each $N \times N$ slice, and note the index of the minimum in the slice. I can

>>>> find my minimum by doing `min(min(array,ind1,dim=1),dim=1,ind2)`, but

>>>> I'm not sure how to turn those two index arrays into the indices that

>>>> I need. Help...?

>

>>>> Thanks!

>

>>>> --Gray

>

>>> Hi,

```

>>> maybe I missed something, but why don't you use something like this:
>
>>> IDL> a=randomn(seed,10,10,5)
>>> IDL> min=min(a,dimension=3,ind)
>>> IDL> help,min,ind
>>> MIN          FLOAT    = Array[10, 10]
>>> IND          LONG64   = Array[10, 10]
>>> IDL> ind2=array_indices(size(a,/dimensions),ind,/dimensions)
>>> IDL> help,ind2
>>> IND2         LONG64   = Array[3, 100]
>
>>> Is array_indices really to slow with the dimension keyword?
>
>>> Cheers
>
>>> CR
>
>> This gets me a NxN array of minima... I want a vector of minima of
>> length M (so the minimum in each plane).
>
> OK, for anyone else, here's what you have to do.
>
> IDL> a = randomu(seed,10,10,100)
> IDL> minima = min(min(a,ind1,dim=1),ind2,dim=2)
> IDL> ind1 -= rebin(indgen(1,100)*10^2.,10,100)
> IDL> ind2 -= (indgen(100)*10)
> IDL> xind = ind1[ind2,indgen(100)] mod 10
> IDL> yind = ind2
>
> You have to use ind2 to find the right elements of ind1. Since you
> get 1d indices from min, you need to subtract off N^2 or N to talk
> about each plane individually. The reason I wanted to vectorize is
> that my actual M is ~20k.

```

Ok, and this one?

```

t0=systime(1)
a=randomu(seed,100l,100l,10000l)
mins=min(reform(a,100l*100l,10000l,/over),ind,dimension=1)
a=reform(a,100l,100l,10000l,/over) ;if you need this...
help,ind,a &print,systime(1)-t0

```

```

IND          LONG64   = Array[10000]
A            FLOAT    = Array[100, 100, 10000]
1.9180000

```

Maybe you can also use Thread Pool Keywords together with min.

Cheers

Subject: Re: 2d min
Posted by [Gray](#) on Thu, 13 Jan 2011 17:07:21 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Jan 13, 8:57 am, chris <rog...@googlemail.com> wrote:
> On 13 Jan., 14:11, Gray <grayliketheco...@gmail.com> wrote:
>
>
>
>
>
>
>
>
>
>> On Jan 13, 7:26 am, Gray <grayliketheco...@gmail.com> wrote:
>
>>> On Jan 13, 2:18 am, chris <rog...@googlemail.com> wrote:
>
>>>> On 12 Jan., 23:21, Gray <grayliketheco...@gmail.com> wrote:
>
>>>> > Hi all,
>
>>>> > I have a 3d array, NxNxM. What I would like is to find the minimum of
>>>> > each NxN slice, and note the index of the minimum in the slice. I can
>>>> > find my minimum by doing min(min(array,ind1,dim=1),dim=1,ind2), but
>>>> > I'm not sure how to turn those two index arrays into the indices that
>>>> > I need. Help...?
>
>>>> > Thanks!
>
>>>> > --Gray
>
>>>> Hi,
>>>> maybe I missed something, but why don't you use something like this:
>
>>>> IDL> a=randomn(seed,10,10,5)
>>>> IDL> min=min(a,dimension=3,ind)
>>>> IDL> help,min,ind
>>>> MIN FLOAT = Array[10, 10]
>>>> IND LONG64 = Array[10, 10]
>>>> IDL> ind2=array_indices(size(a,/dimensions),ind,/dimensions)
>>>> IDL> help,ind2
>>>> IND2 LONG64 = Array[3, 100]
>

```

>>>> Is array_indices really to slow with the dimension keyword?
>
>>>> Cheers
>
>>>> CR
>
>>> This gets me a NxN array of minima... I want a vector of minima of
>>> length M (so the minimum in each plane).
>
>> OK, for anyone else, here's what you have to do.
>
>> IDL> a = randomu(seed,10,10,100)
>> IDL> minima = min(min(a,ind1,dim=1),ind2,dim=2)
>> IDL> ind1 -= rebin(indgen(1,100)*10^2.,10,100)
>> IDL> ind2 -= (indgen(100)*10)
>> IDL> xind = ind1[ind2,indgen(100)] mod 10
>> IDL> yind = ind2
>
>> You have to use ind2 to find the right elements of ind1. Since you
>> get 1d indices from min, you need to subtract off N^2 or N to talk
>> about each plane individually. The reason I wanted to vectorize is
>> that my actual M is ~20k.
>
> Ok, and this one?
>
> t0=sysime(1)
> a=randomu(seed,100l,100l,10000l)
> mins=min(reform(a,100l*100l,10000l,/over),ind,dimension=1)
> a=reform(a,100l,100l,10000l,/over) ;if you need this...
> help,ind,a &print,sysime(1)-t0
>
> IND          LONG64  = Array[10000]
> A            FLOAT   = Array[100, 100, 10000]
>      1.9180000
> Maybe you can also use Thread Pool Keywords together with min.
>
> Cheers
> CR

```

Ooh, reforming so that the first 2 dimensions become 1 that I can min over... I didn't think about that possibility! Either way works, I guess.