

---

Subject: Re: Problem with REFORM and HISTOGRAM

Posted by [rogass](#) on Fri, 08 Oct 2010 22:33:34 GMT

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

---

On 8 Okt., 07:03, Mrunmayee <gaur...@gmail.com> wrote:

> My post did not get posted about 12 hours ago. I live in India and was  
> hoping the post would be up for gurus in western timezones to possibly  
> offer help. Now I will wait for yet another 24 hours! I always check  
> the box to "Send copy to self" which I DID get. And I also notice a  
> lot of spam has gone. I don't think my post contained anything  
> 'hotter' than rebin/reform/histogram. So, dear moderator, may I ask  
> how I offended this group? Anyway. Onto the problem:  
>  
> 1. I have 2 arrays of coordinates, x1arr of N1 size, x2arr of N2 size  
> and same for y1arr, y2arr, z1arr, z2arr. These are coordinates along 2  
> different lines and I need all the distances between points on one  
> line and points on another. I do it by:  
> x1arrReb = Rebin(x1arr, N1, N2)  
> x2arrReb = Rebin(Transpose(x2arr), N1, N2)  
> dx = x1arrReb - x2arrReb  
> Similarly obtain dy, dz. Then,  
> d = Sqrt(dx\*dx + dy\*dy + dz\*dz)  
>  
> 2. I need to bin these distances for further computation.  
> dhist = Histogram(d, nbins=8, locations=bins, reverse\_indices=ri)  
>  
> 3. Now I want to check, in each bin, which distance corresponds to  
> which x1 and which x2 coordinate. From  
> above, I can do following:  
>  
> x1arrRef = Reform(x1arrReb, N1\*N2) & x2arrRef = Reform(x2arrReb,  
> N1\*N2)  
>  
> Then use the Reverse\_Indices and get coordinates. Right?  
>  
> 4. Alternatively, I was trying this:  
> for i = 0,nbins-1  
> indices = ri[ ri[i]: ri[i+1]-1 ]  
> rowID = indices/N1 ; This is to get i,j values from the 1D indices.  
> colID = indices - rowID \* N1  
> x1coord = x1arr[rowID] ; This is the original 1-d N1-element  
> x1arr.  
> x2coord = x2arr[colID] ; ----- ditto ----- x2arr.  
> endfor  
>  
> Problem is, if this way, I get all the coordinates and re-calculate  
> distances to see if I get the distances in the bin 'i' back, I don't!  
> And I don't know where my understanding is wrong here. So can someone

> please point out? Am I book-keeping indices wrong?

Why don't you use the distance\_measure function? You can also just rebin the dist function into third dimension times its transpose and then ask for the values according to its indices ?

Cheers

Cr

---

Subject: Re: Problem with REFORM and HISTOGRAM  
Posted by [Mrunmayee](#) on Tue, 12 Oct 2010 12:05:29 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Sorry, I didn't realize that posts were on again.

I didn't know about this function, so thanks for pointing out. So for say 220 points on each line, there will be 220\*220 distances, this might be memory-prohibitive. With rebinning I mean, won't it?

On Oct 9, 3:33 am, chris <rog...@googlemail.com> wrote:

> On 8 Okt., 07:03, Mrunmayee <gaur...@gmail.com> wrote:

>  
>  
>

>> My post did not get posted about 12 hours ago. I live in India and was  
>> hoping the post would be up for gurus in western timezones to possibly  
>> offer help. Now I will wait for yet another 24 hours! I always check  
>> the box to "Send copy to self" which I DID get. And I also notice a  
>> lot of spam has gone. I don't think my post contained anything  
>> 'hotter' than rebin/reform/histogram. So, dear moderator, may I ask  
>> how I offended this group? Anyway. Onto the problem:

>

>> 1. I have 2 arrays of coordinates, x1arr of N1 size, x2arr of N2 size  
>> and same for y1arr, y2arr, z1arr, z2arr. These are coordinates along 2  
>> different lines and I need all the distances between points on one  
>> line and points on another. I do it by:

>> x1arrReb = Rebin(x1arr, N1, N2)  
>> x2arrReb = Rebin(Transpose(x2arr), N1, N2)  
>> dx = x1arrReb - x2arrReb  
>> Similarly obtain dy, dz. Then,  
>> d = Sqrt(dx\*dx + dy\*dy + dz\*dz)

>

>> 2. I need to bin these distances for further computation.  
>> dhist = Histogram(d, nbins=8, locations=bins, reverse\_indices=ri)

>

```

>> 3. Now I want to check, in each bin, which distance corresponds to
>> which x1 and which x2 coordinate. From
>> above, I can do following:
>
>> x1arrRef = Reform(x1arrReb, N1*N2) & x2arrRef = Reform(x2arrReb,
>> N1*N2)
>
>> Then use the Reverse_Indices and get coordinates. Right?
>
>> 4. Alternatively, I was trying this:
>> for i = 0,nbins-1
>>   indices = ri[ ri[i]: ri[i+1]-1 ]
>>   rowID = indices/N1 ; This is to get i,j values from the 1D indices.
>>   colID = indices - rowID * N1
>>   x1coord = x1arr[rowID] ; This is the original 1-d N1-element
>>   x1arr.
>>   x2coord = x2arr[colID] ; ----- ditto ----- x2arr.
>>   endfor
>
>> Problem is, if this way, I get all the coordinates and re-calculate
>> distances to see if I get the distances in the bin 'i' back, I don't!
>> And I don't know where my understanding is wrong here. So can someone
>> please point out? Am I book-keeping indices wrong?
>
> Why don't you use the distance_measure function? You can also just
> rebin the dist function into third dimension times its transpose and
> then ask for the values according to its indices ?
>
> Cheers
>
> Cr

```

---

Subject: Re: Problem with REFORM and HISTOGRAM

Posted by [rogass](#) on Tue, 12 Oct 2010 12:33:03 GMT

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

---

On 12 Okt., 14:05, Mrunmayee <gaur...@gmail.com> wrote:

```

> Sorry, I didn't realize that posts were on again.
>
> I didn't know about this function, so thanks for pointing out. So for
> say 220 points on each line, there will be 220*220 distances, this
> might be memory-prohibitive. With rebinning I mean, won't it?
>
> On Oct 9, 3:33 am, chris <rog...@googlemail.com> wrote:
>
>
>
>

```

```

>> On 8 Okt., 07:03, Mrunmayee <gaur...@gmail.com> wrote:
>
>>> My post did not get posted about 12 hours ago. I live in India and was
>>> hoping the post would be up for gurus in western timezones to possibly
>>> offer help. Now I will wait for yet another 24 hours! I always check
>>> the box to "Send copy to self" which I DID get. And I also notice a
>>> lot of spam has gone. I don't think my post contained anything
>>> 'hotter' than rebin/reform/histogram. So, dear moderator, may I ask
>>> how I offended this group? Anyway. Onto the problem:
>
>>> 1. I have 2 arrays of coordinates, x1arr of N1 size, x2arr of N2 size
>>> and same for y1arr, y2arr, z1arr, z2arr. These are coordinates along 2
>>> different lines and I need all the distances between points on one
>>> line and points on another. I do it by:
>>> x1arrReb = Rebin(x1arr, N1, N2)
>>> x2arrReb = Rebin(Transpose(x2arr), N1, N2)
>>> dx = x1arrReb - x2arrReb
>>> Similarly obtain dy, dz. Then,
>>> d = Sqrt(dx*dx + dy*dy + dz*dz)
>
>>> 2. I need to bin these distances for further computation.
>>> dhist = Histogram(d, nbins=8, locations=bins, reverse_indices=ri)
>
>>> 3. Now I want to check, in each bin, which distance corresponds to
>>> which x1 and which x2 coordinate. From
>>> above, I can do following:
>
>>> x1arrRef = Reform(x1arrReb, N1*N2) & x2arrRef = Reform(x2arrReb,
>>> N1*N2)
>
>>> Then use the Reverse_Indices and get coordinates. Right?
>
>>> 4. Alternatively, I was trying this:
>>> for i = 0,nbins-1
>>> indices = ri[ ri[i]: ri[i+1]-1 ]
>>> rowID = indices/N1 ; This is to get i,j values from the 1D indices.
>>> colID = indices - rowID * N1
>>> x1coord = x1arr[rowID] ; This is the original 1-d N1-element
>>> x1arr.
>>> x2coord = x2arr[colID] ; ----- ditto ----- x2arr.
>>> endfor
>
>>> Problem is, if this way, I get all the coordinates and re-calculate
>>> distances to see if I get the distances in the bin 'i' back, I don't!
>>> And I don't know where my understanding is wrong here. So can someone
>>> please point out? Am I book-keeping indices wrong?
>
>> Why don't you use the distance_measure function? You can also just

```

```
>> rebin the dist function into third dimension times its transpose and  
>> then ask for the values according to its indices  ?  
>  
>> Cheers  
>  
>> Cr
```

If you don' use the /MATRIX flag, the output will be pairwise.

Cheers

CR

---