Subject: Re: data inside a circle
Posted by Rob Dimeo on Fri, 10 Jun 2005 22:46:58 GMT
View Forum Message <> Reply to Message

You might want to check out the following:

http://www.dfanning.com/code_tips/slowloops.html

angela wrote:
> Hello,
> I have a list of x,y coordinates and I am trying to make circles of
> different centers and radii and find how many of the data points are
> inside in the different circles. Do you happen to know an elegant way
> to do that?
> I have tried to do that using FOR loops but my data points are about
> 10,000 and the number of circles are about 160,000, so it takes about
> 2-3 hours for my program to run. I was wondering if you know of a
> faster way to do that.
>
> Thank you,
> Angela

Subject: Re: data inside a circle
Posted by Rick Towler on Sat, 11 Jun 2005 00:12:53 GMT
View Forum Message <> Reply to Message

In addition of Rob's suggestion, you could cull much of your data set on each pass before checking if it is within your circle. Determine the X/Y bounds of your circle and then extract that subset of data and operate on that.

I'm walking out the door so I can't provide any pseudo code but hopefully this will give you an idea or two.

-Rick

angela wrote:
> Hello,
> I have a list of x,y coordinates and I am trying to make circles of
> different centers and radii and find how many of the data points are
> inside in the different circles. Do you happen to know an elegant way
> to do that?
> I have tried to do that using FOR loops but my data points are about
> 10,000 and the number of circles are about 160,000, so it takes about
> 2-3 hours for my program to run. I was wondering if you know of a
> faster way to do that.

>
> Thank you,
> Angela
>

---

Subject: Re: data inside a circle
Posted by Michael Wallace on Mon, 13 Jun 2005 15:20:03 GMT
View Forum Message <> Reply to Message

> In addition of Rob's suggestion, you could cull much of your data set on
> each pass before checking if it is within your circle. Determine the
> X/Y bounds of your circle and then extract that subset of data and
> operate on that.
>
> I'm walking out the door so I can't provide any pseudo code but
> hopefully this will give you an idea or two.

I was thinking the same thing. For each circle, find the bounding box of the circle and then extract a subset of data points that are within the bounding box. Now test if a given point of the subset is within the circle or not. I don't know what kind of ratio you have between points inside/outside a given circle, but if there are few points inside and many outside, you should see a big improvement.

-Mike

---

Subject: Re: data inside a circle
Posted by angela on Mon, 13 Jun 2005 16:10:04 GMT
View Forum Message <> Reply to Message

Thank you. I was wondering if there is a function or procedure in IDL that will find the coordinates of the bounding box for a circle and/or ellipse.

Angela

---

Subject: Re: data inside a circle
Posted by Rick Towler on Mon, 13 Jun 2005 22:16:54 GMT
View Forum Message <> Reply to Message

Rick Towler wrote:
> In addition of Rob's suggestion, you could cull much of your data set on
> each pass before checking if it is within your circle. Determine the

> X/Y bounds of your circle and then extract that subset of data and
> operate on that.

On my ride home I realized that your data is probably not regularly gridded but is an unordered list which would make this approach much less effective. You could squeeze a bit more performance by doing a gross check on the data with a before actually calculating if the point falls in the circle. I use a separating axis check which given the bounding box and your point progressively tests if the point falls on the "right" side of the four sides of your bounding box. You can google for separating axis test or I can provide an example. But in the end you're still looping thru every element which is going to be slow.

A better option would be to first process the data to order it in some way then cull the data. This way you'll be able to extract the indices to the data which fall within your bounding box which you can use to extract your sub-array.

A simple approach would be to use sorting. First sort your entire array by X (only have to do this once). Then use the separating axis test (or you can use two calls to where and AND the results) to extract all values of X and their associated Y's w/in the bounding box. Then sort that subset by Y and extract a subset of the subset of data which intersects the bounding box in Y.

You could also apply HISTOGRAM to both your X and Y values independently and then use the reverse indices to get at the subset of your data that fall within the bins that are intersect your bounding box. You would have to do the same 2 step approach as the sorting method. Check out JD's histogram tutorial on David Fanning's website (www.dfanning.com) to help bring you up to speed on the HISTOGRAM function.

Lastly, you can do it in C. I have done a lot of this in 3d "the hard way" by looping thru my vertices, performing the separating axis test, and then calculating whatever I am calculating. Probably the fastest solution of the all but there's a bit of a learning curve. If this is something you will be doing *a lot* of then it might be the best way. I could probably provide some code to get you started.

-Rick

> angela wrote:
>
>> Hello,
>> I have a list of x,y coordinates and I am trying to make circles of
>> different centers and radii and find how many of the data points are

>> inside in the different circles. Do you happen to know an elegant way
>> to do that?
>> I have tried to do that using FOR loops but my data points are about
>> 10,000 and the number of circles are about 160,000, so it takes about
>> 2-3 hours for my program to run. I was wondering if you know of a
>> faster way to do that.
>>
>> Thank you,
>> Angela
>>

---