

---

Subject: Check for duplicate locations

Posted by [Med Bennett](#) on Thu, 18 Feb 2016 16:48:34 GMT

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

---

I have X,Y,Z data for several thousand points that I need to check for duplicate locations. I cannot have duplicated locations in the sample data, as it breaks the kriging algorithm I am using. I've always used a brute force method of computing a distance function between each point and all subsequent points, and flagging any points for which the distance is zero, or some small threshold. This method is very slow for larger numbers of points, however. Does anyone have a method for doing this more efficiently? I've found simple methods for one-dimensional data, but not for points in 3-space.

Thanks!

---

---

Subject: Re: Check for duplicate locations

Posted by [Russell\[1\]](#) on Mon, 22 Feb 2016 19:36:10 GMT

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

---

On Thursday, February 18, 2016 at 11:48:38 AM UTC-5, Med Bennett wrote:

> I have X,Y,Z data for several thousand points that I need to check for duplicate locations. I cannot have duplicated locations in the sample data, as it breaks the kriging algorithm I am using. I've always used a brute force method of computing a distance function between each point and all subsequent points, and flagging any points for which the distance is zero, or some small threshold. This method is very slow for larger numbers of points, however. Does anyone have a method for doing this more efficiently? I've found simple methods for one-dimensional data, but not for points in 3-space.

>

> Thanks!

This is a tough answer to explain in just a few words....

Are the (x,y,z) values exactly the same? Either way, I would map them into a single coordinate (basically the inverse operation of array\_indices) and compute the histogram. Consider a 2-d example...

(x,y)=(1,2)

and the maximum value of x,y could be (Nx,Ny)=(100,100). Then you can combine (x,y) into single value:  $xy = x + nx * y$

Now use the histogram function on that new variable, and any bin in the histogram with more than a count of 2 has multiple entries. At that point you can do just about whatever to them.

---

---

Subject: Re: Check for duplicate locations

Posted by [Craig Markwardt](#) on Wed, 24 Feb 2016 14:14:41 GMT

On Thursday, February 18, 2016 at 11:48:38 AM UTC-5, Med Bennett wrote:

> I have X,Y,Z data for several thousand points that I need to check for duplicate locations. I cannot have duplicated locations in the sample data, as it breaks the kriging algorithm I am using. I've always used a brute force method of computing a distance function between each point and all subsequent points, and flagging any points for which the distance is zero, or some small threshold. This method is very slow for larger numbers of points, however. Does anyone have a method for doing this more efficiently? I've found simple methods for one-dimensional data, but not for points in 3-space.

>

> Thanks!

There's no simple answer to this. Usually you need to do some kind of filtering to make it a faster process. For example, if the spread in "Z" values in your 3-space is the greatest, sort by Z first, then you can limit the range over which you perform the 3D distance computation. But there are lots of potential gotchas when you do this! Another solution is to get a faster CPU!

Craig

---

---

Subject: Re: Check for duplicate locations

Posted by [Med Bennett](#) on Thu, 25 Feb 2016 20:35:34 GMT

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

---

Thanks for your replies - computationally, it's not really a problem, but I always like to find the most 'elegant' solution to a problem, and thought there might be a better way than brute force distance calculations. One issue is that sometimes the duplicate points differ by a small amount due to different rounding or storage, requiring use of a threshold. Thanks again!

---