
Subject: Point inside/outside a polygon?

Posted by [Dennis Boccippio](#) on Fri, 10 Aug 2001 16:14:42 GMT

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

Hi all,

Does anyone know of IDL routines for determination of whether a point datum is inside/outside a closed irregular polygon? I'm told that some variants of this functionality are available in Matlab and Mathematica, was curious if anyone had implemented a solution for IDL. My usual kludge to this is polyfill/mask-based, which is probably a very inefficient approach...

Thanks,

Dennis Boccippio

Subject: Re: Point inside/outside a polygon?

Posted by [Med Bennett](#) on Fri, 10 Aug 2001 23:24:41 GMT

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

Here is what I ended up with when I had to solve this problem a few years back:

- 1) define a point that is outside of the polygon - easy enough.
- 2) define a line from the point in question to your outside point, say A-A'.
- 3) Count the number of times that this line crosses any of the line segments making up the polygon. If it's an even number, your point is outside the polygon. If it's odd, your point is inside the polygon.

It's easy to see why this works geometrically with a simple drawing, and it's pretty straightforward to code. David's method appears to be more computationally efficient, as it avoids loops, but I don't fully understand why it works. I'd be happy to send you my actual code if you'd like. The polyfillv method works also, but is better for points on a regular grid, and you have to convert the polygon coordinates to image coordinates to get it to work.

Med Bennett

Dennis Boccippio wrote:

> Hi all,
>

> Does anyone know of IDL routines for determination of whether a point
> datum is inside/outside a closed irregular polygon? I'm told that some
> variants of this functionality are available in Matlab and Mathematica,
> was curious if anyone had implemented a solution for IDL. My usual
> kludge to this is polyfill/mask-based, which is probably a very
> inefficient approach...
>
> Thanks,
>
> Dennis Boccippio

Subject: Re: Point inside/outside a polygon?

Posted by [Jim Pendleton](#) on Sat, 11 Aug 2001 00:40:00 GMT

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

"Dennis Boccippio" <djboccip@hotmail.com> wrote in message
news:djboccip-ECCC8D.11144210082001@news.mia.bellsouth.net.. .

> Hi all,
>
> Does anyone know of IDL routines for determination of whether a point
> datum is inside/outside a closed irregular polygon? I'm told that some
> variants of this functionality are available in Matlab and Mathematica,
> was curious if anyone had implemented a solution for IDL. My usual
> kludge to this is polyfill/mask-based, which is probably a very
> inefficient approach...
>
> Thanks,
>
> Dennis Boccippio
>

Hi Dennis,

The IDLanROI class has a method named "ContainsPoints" to perform this operation efficiently. It will let you know if a point lies exactly on an edge or vertex, or completely inside or outside the specified polygon. And it's also good in three dimensions. See the IDLanROIGroup class for extended functionality.

Jim P.

Subject: Re: Point inside/outside a polygon?

Posted by [Mark Fardal](#) on Sat, 11 Aug 2001 04:48:44 GMT

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

Hi Dennis,

you can also try

<http://www-astro.phast.umass.edu/~fardal/idl/pnpoly.pro>

If you do any comparison of methods (speed/accuracy/ease of use), the results would be interesting. Many of us have re-invented the wheel here, or at least re-implemented it. We should hold a race sometime...

Mark Fardal
UMass

Subject: Re: Point inside/outside a polygon?

Posted by [Andy Loughe](#) on Mon, 13 Aug 2001 15:12:55 GMT

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

Here is some ROI code I have collected over the years...

```
; Define and draw an oddly shaped polygon.  
x = [ .5, .7, .8, .8, .6, .5, .5]  
y = [ 0., .2, .1, .9, .7, .7, 0.]  
z = [ 0., 0., 0., 0., 0., 0., 0.]  
plot, x, y, /nodata, xrange=[0,1], yrange=[-.2,1]  
plots, x, y, thick=5
```

```
roi = Obj_New('IDLanROI', x, y, z)
```

```
; Point outside the polygon = 0 (CASE #1)  
x1 = .6 & y1=.05 & z1=0.0  
xyouts, x1, y1, '1', chars=1.25, align=.5  
print, roi->ContainsPoints(x1, y1, z1)
```

```
; Point inside the polygon = 1 (CASE #2)  
x1 = .78 & y1=.85 & z1=0.0  
xyouts, x1, y1, '2', chars=1.25, align=.5  
print, roi->ContainsPoints(x1, y1, z1)
```

```
; Point on an edge of the polygon = 2 (CASE #3)  
x1 = .8 & y1=.5 & z1=0.0  
xyouts, x1, y1, '3', chars=1.25, align=.5  
print, roi->ContainsPoints(x1, y1, z1)
```

```
; Point on a vertex of the polygon = 3 (CASE #4)  
x1 = .7 & y1=.2 & z1=0.0  
xyouts, x1, y1, '4', chars=1.2, align=.5  
print, roi->ContainsPoints(x1, y1, z1)
```

Obj_Destroy, roi

--

Andrew Loughe =====
NOAA/OAR/FSL/AD R/FS5 | email: loughe@fsl.noaa.gov
325 Broadway | www: www-ad.fsl.noaa.gov/users/loughe
Boulder, CO 80305-3328 | phone: 303-497-6211 fax: 303-497-6301

Subject: Re: Point inside/outside a polygon?

Posted by [James Kuyper](#) on Mon, 13 Aug 2001 16:07:39 GMT

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

Med Bennett wrote:

>
> Here is what I ended up with when I had to solve this problem a few years
> back:
>
> 1) define a point that is outside of the polygon - easy enough.
>
> 2) define a line from the point in question to your outside point, say
> A-A'.
>
> 3) Count the number of times that this line crosses any of the line
> segments making up the polygon. If it's an even number, your point is
> outside the polygon. If it's odd, your point is inside the polygon.
>
> It's easy to see why this works geometrically with a simple drawing, and
> it's pretty straightforward to code. David's method appears to be more
> computationally efficient, as it avoids loops, but I don't fully understand
> why it works. I'd be happy to send you my actual code if you'd like. The

For a point on the inside of a polygon, travelling around the polygon involves a net angular motion of 360 degrees around that point. For a point outside the polygon, the net angular change must be 0 degrees. It's easier to see that this is so, if you first consider a simple convex polygon, but it remains true even if it's not convex.
