Subject: Matrix rank
Posted by Wox on Fri, 14 Dec 2007 14:16:29 GMT
View Forum Message <> Reply to Message

Hi IDLers,

Is there a routine available which calculates the rank of an (integer) matrix? Couldn't find it in the help and I would be surprised if it's not there. It's for knowing whether sets of linear equations have no solution, 1 solution or an infinite number of solutions.

Thanks.

Subject: Re: Matrix rank
Posted by d.poreh on Fri, 14 Dec 2007 18:17:42 GMT
View Forum Message <> Reply to Message

On Dec 14, 7:06 pm, Steve Eddins <Steve.Edd...@mathworks.com> wrote:
> Vince Hradil wrote:
>> On Dec 14, 9:42 am, Wox <nom...@hotmail.com> wrote:
>>> On Fri, 14 Dec 2007 06:35:11 -0800 (PST), Vince Hradil
>
>>> <hrad...@yahoo.com> wrote:
>>>> IDL can do SVD, can you get the rank from that? Look up SVDC in the
>>>> docs.
>>> I could do this, but maybe there's a better way?
>
>>> ; A: integers
>>> ; B: floats
>>> A = [[0,0,1], $
>>> [0,1,0], $
>>> [0,0,0]]
>>> B = [0.25,0.5,1]
>
>>> ; Decompose A
>>> SVDC, A, W, U, V
>>> ; Solve A.X=B
>>> X=SVSOL(U, W, V, B)
>
>>> ; Check
>>> B2=A##X
>>> ind=where(total(abs(A),1,/pres) ne 0)
>
>>> if array_equal(B[ind],B2[ind]) then print,X
>
>> Well, w contains the singular values, the number of these that are non-
>> zero will be the rank:

>> idx = where(w ne 0, rank)
>> print, rank
>> 2
>
> Since this is all in floating-point, it's appropriate to use a tolerance
> instead of comparing exactly with 0. See, for example, the algorithm
> used in the MATLAB rank function, which uses a tolerance based on the
> size of the matrix and the maximum singular value. It's described here:
>
> http://www.mathworks.com/access/helpdesk/help/techdoc/ref/ra nk.html
>
> I assume this is straightforward to express in IDL.
>
> ---
> Steve Eddinshttp://blogs.mathworks.com/steve/- Hide quoted text -
>
> - Show quoted text -

but in MATLAB:
tol = max(size(A)) * norm(A) * eps.
why?

Subject: Re: Matrix rank
Posted by Wox on Mon, 17 Dec 2007 10:53:42 GMT
View Forum Message <> Reply to Message

Thanks for your help guys. However I'm struggling with the problem I wanted to use this rank for. Maybe someone can help.

Suppose you have two subspaces of 3D space (e.g. [z,y+0.5,0] and [z,0,0]). Now I just want to check whether one is a subspace of the other.

My first idea was that if you could find a solution for
R1(3x3).X(3x1)+T1(3x1) = R2.X+T2
that one is a subspace of the other. However, [0.5,0,z] is not a subspace of [y,z,0] while it gives X=[0,0.5,0] as a solution.

The second (brute-force) idea is this:
h=histogram((total(abs(R1),1,/pres) eq 0)+$
(total(abs(R2),1,/pres) eq 0),min=0,max=2,binsize=1,rev=rev)

; h=0 => both fixed: check whether they are the same
...
; h=2 => both variable: check whether they are the same
...
; h=1 => one is variable: find solution

...

Is there a more elegant solution to this?

Thanks.
