
Subject: Re: Matrix algebra and index order, A # B vs A ## B
Posted by [David Fanning](#) on Mon, 26 Mar 2012 14:35:57 GMT
[View Forum Message](#) <> [Reply to Message](#)

Mats Löfdahl writes:

> It's that bad? :o)

Well, it gets better after a couple of beers. :-)

>
> One thing that had me wondering is the documentation for Craig Markwardt's qrfac routine:
>
>
> ; Given an MxN matrix A (M>N), the procedure QRFAC computes the QR
> ; decomposition (factorization) of A. This factorization is useful
> ; in least squares applications solving the equation, A # x = B.
> ; Together with the procedure QRSOLV, this equation can be solved in
> ; a least squares sense.
> ;
> ; The QR factorization produces two matrices, Q and R, such that
> ;
> ; A = Q ## R
> ;
> ; where Q is orthogonal such that TRANSPOSE(Q)##Q equals the identity
> ; matrix, and R is upper triangular.

See, this is the real problem. Is Craig explaining the linear algebra to us, or is he giving us the IDL rendition of the linear algebra? Or, did Craig write the documentation after the fact and is really trying to forget what he *actually* did in the program? All I know is that anytime I read the word "matrix" I can tell I'm about to get a headache.

> The ## operator for the matrix-matrix multiplications but # for
> matrix-vector multiplication! But then I thought this might be
> IDL 1D arrays being interpreted as row vectors so x # A is
> actually just another way of writing A ## transpose(x). And
> the former would be more efficient. Am I on the right
> track here...?

Probably. But I can't be bothered. I've got to go
find some aspirin. :-(

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.idlcoyote.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")
