Subject: Re: Transpose(A)*P*A
Posted by Craig Markwardt on Sat, 11 Oct 2008 19:42:15 GMT
View Forum Message <> Reply to Message

"mapper4u6@gmail.com" <zjwang2u@gmail.com> writes:

> hello,
>
> I have a question about how to improve the computation speed when deal
> with non-linear equation
> A x = l, P is the weight for each
> row, P is M*M
> M*N N*1 M*1
>
> then I have to build normal matrix which is
> Transpose(A)*P*A x = Transpose(A)*P*l
> N*N N*1 N*1
>
> then x can be solved.

I'm going to channel our vice presidential candidate and answer a
different question.

It looks like you are trying to solve a least squares problem. It's
well documented that the normal equation method suffers from accuracy
problems, basically because you are squaring the A matrix, and thus
squaring the errors. Have you tried SVD or QR factorization?
Implemented correctly, the execution times should scale as,
Normal equation ~ N^2 * (M + N/3)
QR ~ N^2 * (2*M - 2*N/3)
SVD ~ N^2 * (2*M +11*N/3)
On its face, QR factorization will take longer (not more than double
the time though), but it is known to be more stable.

Craig

--
---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: cbmarkwardt+usenet@gmail.com
---------- ----------