Greville's Method For Preconditioning Least Squares Problems

X. Cui
Department of Informatics, School of Multidisciplinary Sciences, The Graduate University for Advanced Studies (Sokendai), 2-1-2, Hitotsubashi, Chiyoda-ku, Tokyo, Japan, 101-8430
K. Hayami
Principles of Informatics Research Division, National Institute of Informatics, 2-1-2, Hitotsubashi, Chiyodaku, Tokyo, Japan, 101-8430
J. Yin
Department of Mathematics, Tongji University, Shanghai, P.R. China

In this talk, we present a preconditioner for least squares problems $\min \|b - A_x\|_2$, where A can be matrices with any shape or rank. When A is rank deficient, our preconditioner will be rank deficient too. The preconditioner itself is a sparse approximation to the Moore-Penrose inverse of the coefficient matrix A.

Greville's method [1] is an old method for computing the Moore-Penrose inverse of a matrix A. We first write A in the following summation form,

$$A = \sum_{i=1}^{n} a_i e_i^T,$$

where a_i is the ith column of A, e_i is the ith column of an identity matrix of order m. Further define

$$A_i = \sum_{k=1}^{i} a_i e_i^T, \quad i = 1, \ldots, n,$$

and if we denote $A_0 = 0_{m\times n}$, then $A_i = A_{i-1} + a_i e_i^T$, $i = 1, \ldots, n$. Thus every A_i, $i = 1, \ldots, n$ is a rank-one update of A_{i-1}. Noticing that $A_0^\dagger = 0_{n\times m}$, we can use the following formula to compute the Moore-Penrose inverse of A_i, and in the end we obtain $A_n^\dagger$, which is $A^\dagger$.

$$A_i^\dagger = \begin{cases} A_{i-1}^\dagger + (e_i - A_{i-1}^\dagger a_i)((I - A_{i-1}A_{i-1}^\dagger)a_i)^\dagger & \text{if } a_i \notin \mathcal{R}(A_{i-1}) \\ A_{i-1}^\dagger + \frac{1}{\sigma_i}(e_i - A_{i-1}^\dagger a_i)(-A_{i-1}^\dagger a_i)^T A_{i-1}^\dagger & \text{if } a_i \in \mathcal{R}(A_{i-1}) \end{cases},$$

where $\sigma_i = 1 + \|A_{i-1}^\dagger a_i\|_2^2$. We can judge if $a_i \in \mathcal{R}(A_{i-1})$ or not by observing vector $u := (I - A_{i-1}A_{i-1}^\dagger)a_i$, since

$$a_i \notin \mathcal{R}(A_{i-1}) \Leftrightarrow u = (I - A_{i-1}A_{i-1}^\dagger)a_i = 0,$$
$$a_i \in \mathcal{R}(A_{i-1}) \Leftrightarrow u = (I - A_{i-1}A_{i-1}^\dagger)a_i = 0.$$

This method was proposed by Greville in the 1960s[1].

From Greville's method, a factorization for the Moore-Penrose inverse of A can be obtained. If we define vectors k_i, f_i and v_i as

$$
\begin{aligned}
k_i &= A_{i-1}^{\dagger} a_i, \\
u_i &= a_i - A_{i-1} k_i = (I - A_{i-1} A_{i-1}^{\dagger}) a_i, \\
\sigma_i &= 1 + \|k_i\|_2^2, \\
f_i &= \begin{cases} \|u_i\|_2^2 & \text{if } \ a_i \notin \mathcal{R}(A_{i-1}) \\ \sigma_i & \text{if } \ a_i \in \mathcal{R}(A_{i-1}) \end{cases}, \\
v_i &= \begin{cases} u_i & \text{if } \ a_i \notin \mathcal{R}(A_{i-1}) \\ (A_{i-1}^{\dagger})^T k_i & \text{if } \ a_i \in \mathcal{R}(A_{i-1}) \end{cases},
\end{aligned}
$$

we can express $A_i^{\dagger}$ in a unified form for general matrices as $A_i^{\dagger} = A_{i-1}^{\dagger} + \frac{1}{f_i}(e_i - k_i) v_i^T$, hence

$$
A^{\dagger} = \sum_{i=1}^{n} \frac{1}{f_i} (e_i - k_i) v_i^T.
$$

If we denote

$$
K = [k_1, \ldots, k_n], \quad V = [v_1, \ldots, v_n], \quad F = \mathrm{Diag}\{f_1, \ldots, f_n\},
$$

we obtain a matrix factorization of $A^{\dagger}$ as follows.

Theorem 1 *Let $A \in R^{m \times n}$ and $\mathrm{rank}(A) \leq \min\{m, n\}$. Using the above notations, the Moore-Penrose inverse of A has the following factorization*

$$
A^{\dagger} = (I - K) F^{-1} V^T.
$$

Here I is the identity matrix of order n, K is a strict upper triangular matrix, F is a diagonal matrix, whose diagonal elements are all positive.

If A is full column rank, then

$$
\begin{aligned}
V &= A(I - K) \\
A^{\dagger} &= (I - K) F^{-1} (I - K)^T A^T.
\end{aligned}
$$

We perform an incomplete version of Greville's method, so that we can construct an approximate Moore-Penrose inverse of A, maintaining the sparsity of the preconditioner and saving computations.

Algorithm 1 *Greville Preconditioning Algorithm*

```
set K = 0_{n×n}
for i = 1 : n
    u = a_i − A_{i−1} k_i
    if ||u|| is small
        f_i = ||u||_2^2
        v_i = u
    else
```

8. $\quad\quad f_i = \|k_i\|_2^2 + 1$
9. $\quad\quad v_i = (M_{i-1})^T k_i = \sum_{p=1}^{i-1} \frac{1}{f_p} v_p (e_p - k_p)^T k_i$
10. $\quad$ end if
11. $\quad$ for $j = i+1, \ldots, n$
12. $\quad\quad k_j = k_j + \frac{v_i^T a_j}{f_i}(e_i - k_i)$
13. $\quad\quad$ perform numerical droppings on k_j
14. $\quad$ end for
15. end for
16. $K = [k_1, \ldots, k_n]$, $F = \mathrm{Diag}\{f_1, \ldots, f_n\}$, $V = [v_1, \ldots, v_n]$ such that $A^\dagger \approx (I - K)F^{-1}V^T$.

Consider the least squares problem,

$$\min_{x \in R^n} \|b - Ax\|_2, \tag{1}$$

where $A \in R^{m \times n}$, $b \in R^m$. In [2], Hayami et al. proposed using use GMRES [3] to solve least squares problems by using some preconditioners. Using our preconditioner $M \in R^{n \times m}$ and we precondition (1) from the left, we can transform problem (1) to

$$\min_{x \in R^n} \|Mb - MAx\|_2.$$

On the other hand, we can also precondition problem (1) from the right and transform the problem (1) to

$$\min_{y \in R^m} \|b - AMy\|_2.$$

Numerical examples will be presented in the talk.

References

[1] T. N. E. Greville, Some applications of the pseudoinverse of a matrix. *SIAM Review*, Vol. 2, pp. 15–22, 1960.

[2] K. Hayami, J-F Yin, and T. Ito, GMRES methods for least squares problem. *National Institute of Informatics Technical Report*, NII-2007-09E, 2007.

[3] Y. Saad and M. H. Schultz, GMRES: A generalized minimal residual method for solving nonsymmetric linear systems. *SIAM J. Sci. Statist. Comput.*, Vol. 7, pp. 856–869, 1986.