Proceeding Series of the Brazilian Society of Applied and Computational Mathematics, v. 6, n. 1, 2018.

Trabalho apresentado no XXXVII CNMAC, S.J. dos Campos - SP, 2017.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Geometria do AINV

Moisés Ceni[1]
PPG-EM, UERJ, Departamento de Matemática e Desenho, CAp, UERJ, Rio de Janeiro, RJ
Luiz Mariano Carvalho[2]
Departamento de Matemática Aplicada, IME, UERJ, Rio de Janeiro, RJ
Michael Souza[3]
Departamento de Estatística e Matemática Aplicada, UFC, Ceará, CE

1 Introdução

Em trabalho apresentado no CNMAC2016, provamos que o algoritmo AINV [1] (Approximate INVerse), com adaptações mínimas, pode ser utilizado para calcular a inversa de uma matriz simétrica e positivo definida (SPD) em bloco. Dentre as vantagens de usar matrizes em bloco destacam-se a ampliação do AINV para uma classe maior de matrizes, a estabilidade e o ganho de performance pelo melhor gerenciamento das memórias rápidas do computador.

Para A, uma matriz SPD, o AINV pode ser descrito, em notação inspirada em Python e no LaTeX, pelo Algoritmo 1.

Algoritmo 1 AINV

```
Z=I
for i in range(k):
   p_i = (a_i)^T z_i
   for j in range(i+1,k):
       v_j^{(i-1)} =  (a_i)^T z_j^{(i-1)}# índice superior representa a iteração
   for j in range (i+1,k):
       z_j^{(i)}= z_j^{(i-1)}-z_i^{(i-1)} (p_i)^(-1)v_j^{(i-1)}
D = Diag(p_1, p_2,...,p_k)
```

Onde a_i^T representa a i-ésima linha de A e "Diag" constrói uma matriz diagonal a partir dos elementos p_i. Esse algoritmo calcula a inversa exata de A. O objetivo principal é conseguir uma aproximação suficientemente boa e barata da inversa de A que possa ser usada de forma multiplicativa como um precondicionador, por exemplo, de um método iterativo de Krylov. Nessa contribuição, apresentamos uma demonstração geométrica da construção do AINV em bloco, que também vale para matrizes escalares. Nessa proposta,

[1]moisesceni@gmail.com
[2]luizmc@ime.uerj.br
[3]souza.michael@gmail.com

devido à limitação de espaço, apresentaremos apenas uma parte da demonstração, relativa à construção de uma matriz bloco diagonal como resultado do Algoritmo 1. O restante da demonstração, com a formação de pivôs aproximados, $\tilde{p}_i$, não-singulares, será apresentada no Congresso.

2 AINV para matrizes em bloco

Um vetor-bloco é uma matriz $v = \begin{pmatrix} v_{[1]} & v_{[2]} & \dots & v_{[k]} \end{pmatrix}^T$, $n \times m$, com n divisível por m, onde cada $v_{[s]} \in \mathbb{R}^{m \times m}$, para $s = 1, \dots, k$. Diremos que dois vetores-bloco, u e v, são ortogonais se $v^T u = 0 \in R^{m \times m}$. Usando essas definições, o Algoritmo 1 pode ser alterado para matrizes em bloco, bastando substituir os vetores por vetores-bloco. Para o algoritmo na versão em blocos, apresentamos dois resultados, supondo que o pivôs, p_i, não são singulares.

Teorema 2.1. *Sejam $z_{[j]}$ e $a_{[j]}$ as j-ésimas colunas-bloco (vetores-bloco), respectivamente, das matrizes em bloco Z e A, formadas ao fim do Algoritmo 1, em sua versão em bloco. Então, $z_{[j]}^T a_{[i]} = 0, \ i < j$.*

Demonstração. Para não sobrecarregar a notação, não utilizaremos os índices superiores relativos às iterações. Fazendo indução em i. Primeiro, provamos que $z_{[j]}^T a_{[1]} = 0, j \geq 2$. De fato,

$$a_{[1]}^T z_{[j]} = a_{[1]}^T (z_{[j]} - z_{[1]} (a_{[1]}^T z_{[1]})^{-1} a_{[1]}^T z_{[j]}) = 0.$$

Suponha que $z_{[j]}^T a_{[n]} = 0$ para $n < i < j$. Então, na i-ésima iteração

$$a_{[i]}^T z_{[j]} = a_{[i]}^T (z_{[j]} - z_{[i]} (a_{[i]}^T z_{[i]})^{-1} a_{[i]}^T z_{[j]}) = 0.$$

□

Teorema 2.2. *Dada uma matriz A SPD, o AINV por blocos produz uma matriz triangular superior por blocos Z e uma matriz diagonal por blocos D tal que $Z^T A Z = D$.*

Demonstração. Devido ao Teorema 2.1, temos que o produto matricial AZ é triangular inferior por blocos. Como Z é triangular superior por blocos, Z^T é triangular inferior por blocos. Portanto, $Z^T AZ$ é triangular inferior por blocos.

Agora, como AZ é triangular inferior por blocos e A é matriz simétrica, $(AZ)^T = Z^T A$ é triangular superior por blocos. Portanto, $Z^T AZ$ é triangular superior por blocos, provando que, na realidade, trata-se de uma matriz diagonal por blocos. □

A discussão acima prova que, se o AINV vai até o final, caso não haja nenhum pivô, p_i, singular, então $Z^T AZ = D$ é uma matriz diagonal por blocos. Faltando demonstrar, e que será objeto do pôster no Congresso, que os pivôs realmente não são singulares.

Referências

[1] M. Benzi, C. D. Meyer and M. Tuma. A Sparse Approximate Inverse Preconditioner for the Conjugate Gradiente Method, *Soc. Ind. Appl. Math.*, 17:1135-1149, 1996.

