Decomposição QR para resolver Problemas de Quadrados Mínimos

Instituto de Matemática Pura e Aplicada

Dyckson Ternoski[1]
Marina Sayuri Vieira[2]
Monique Baptista Fragozo[3]
Otávio Dittrich Moreira[4]
Orientador: Prof. Dr. Abel Soares Siqueira[5]
Universidade Federal do Paraná
[1]dycksonternoski@hotmail.com , [2]marinasayuri16@gmail.com, [3]mbaptistafragozo@gmail.com, [4]otaviodittrich@gmail.com, [5]abelsiqueira@ufpr.br

DMAT - UFPR

Resumo

Com o objetivo de otimizar a resolução do problema de quadrados mínimos, neste trabalho implementamos na linguagem Julia a fatoração QR por Gram-Schmidt e por Rotação de Givens, comparando os dois métodos a partir da memória e tempo gastos.

1. O Problema de Quadrados Mínimos

Um dos problemas mais importantes em aplicações matemáticas é a resolução de sistemas lineares da forma:

$$Ax = b \tag{1}$$

onde $A \in \mathbb{R}^{n\times n}$ e $b \in \mathbb{R}^m$. Quando $m > n$, temos um sistema com mais equações do que incógnitas e (1) pode não ter solução. Nesse caso, a melhor solução é resolver o Problema de Quadrados Mínimos: $\min\limits_{x\in\mathbb{R}^n} \|Ax - b\|_2^2$.

Isso envolve resolver o sistema: $A^tAx = A^ty$, o qual possui solução única $x = (A^tA)^{-1}A^ty$ sempre que A tem vetores coluna L.I. Porém, a inversão de matrizes possui um gasto computacional alto. Para isso, utilizaremos a decomposição QR.

2. Decomposição QR

Definição 1. *A decomposição QR de uma matriz A é o par de matrizes Q e R, ortogonal e triangular superior, respectivamente, satisfazendo $A = QR$.*

Proposição 1. *Se Q é uma matriz ortogonal então Q preserva a norma.*
De fato,

$$\|Qv\|_2^2 = (Qv)^tQv = v^tQ^tQv = v^tv = \|v\|_2^2$$

2.1 Por Gram-Schmidt

Teorema 1. *Seja $A \in \mathbb{R}^{m\times n}$ com colunas L.I. Podemos usar o processo de Gram-Schmidt nas colunas $a_1, a_2, \ldots, a_n$ da matriz A e normalizá-las, obtendo elementos $q_1, q_2, \ldots, q_n$ para encontrar a decomposição QR, sendo: $Q = (q_1 q_2 \ldots q_n)$ e $R = Q^tA$.*

Abaixo o algoritmo para o cálculo da decomposição QR por Gram-Schmidt.

Algoritmo 1: QR POR GRAM-SCHMIDT

Entrada: $A \in \mathbb{R}^{m\times n}$ $A = (a_1, a_2, \ldots, a_n)$; $ntol = 1e-12$
Saída: Q,R `Decomposição QR de A`
início
  $q_1 \leftarrow \dfrac{a_1}{\|a_1\|}$
  para *i de 2 à m* **faça**
    $w_i \leftarrow a_i - \sum_{j=1}^{i-1}\langle q_j, a_i\rangle q_j$
    se $\|w_i\| \leq ntol$ **então**
      Erro: Norma de w_i muito próxima de 0
    fim
    $q_i \leftarrow \dfrac{w_i}{\|w_i\|}$
  fim
  $Q \leftarrow (\ q_1, q_2, \ldots, q_n\ )$
  $R = Q^tA$
  retorna Q, R
fim

2.2 Por Rotação de Givens

Teorema 2. *Se A é uma matriz que satisfaz $\mathrm{posto}(A) = n$, então existem matrizes de rotação $Q_1, Q_2, \ldots, Q_n$ tais que: $Q_1Q_2 \ldots Q_nA = R$, sendo R triangular superior. Portanto, se $Q = Q_n^tQ_{n-1}^t \cdots Q_1^t$, então: $A = QR$ é uma decomposição QR de A.*

A seguir o algoritmo desenvolvido para o cálculo da decomposição QR por Rotação de Givens.

Algoritmo 2: QR POR ROTAÇÕES DE GIVENS

Entrada: $A \in \mathbb{R}^{m\times n}$ $A = (a_1, a_2, \ldots, a_n)^t$
Saída: Q,R `Decomposição QR de A`
início
  $Q = (q_1, q_2, \ldots, q_n)^t = \text{Identidade } n \times n$
  para *j de 1 à n* **faça**
    para *i de j+1 à m* **faça**
      se $a_{ij} \neq 0$ **então**
        $c_{ij} \leftarrow a_{jj}/\sqrt{a_{jj}^2 + a_{ij}^2}$ `Supondo` $a_{ij}, a_{jj} \neq 0$
        $s_{ij} \leftarrow a_{ij}/\sqrt{a_{jj}^2 + a_{ij}^2}$
        $a_j,\ a_i \leftarrow a_jc_{ij} + a_is_{ij},\ a_j(-s_{ij}) + a_ic_{ij}$
        $q_j,\ q_i \leftarrow q_jc_{ij} + q_is_{ij},\ q_j(-s_{ij}) + q_ic_{ij}$
      fim
    fim
  fim
  $Q \leftarrow Q^t$
  $R \leftarrow A$
  retorna Q,R
fim

3. Quadrados mínimos via decomposição QR

Como a decomposição QR ajuda no Problema de Quadrados Mínimos?
Se A = QR, pela $\mathbf{Proposição 1}$, podemos resolver o Problema de Quadrados Mínimos de (A, b) como

$$\begin{aligned}\|Ax - b\|_2^2 &= \|QRx - b\|_2^2 \\ &= \|Q^t(QRx - b)\|_2^2 \\ &= \|Rx - Q^tb\|_2^2 \\ &= \|\hat{R}x - c\|_2^2 + \|d\|_2^2\end{aligned}$$

Onde

$$R = \begin{pmatrix}\hat{R} \\ 0\end{pmatrix} \quad Q^tb = \begin{pmatrix} c \\ d\end{pmatrix}$$

Assim, basta resolver o sistema triangular $\hat{R}x = c$.
A resolução de quadrados mínimos por decomposição QR é dada pelo algoritmo abaixo.

Algoritmo 3: RESOLUÇÃO DE QUADRADOS MÍNIMOS

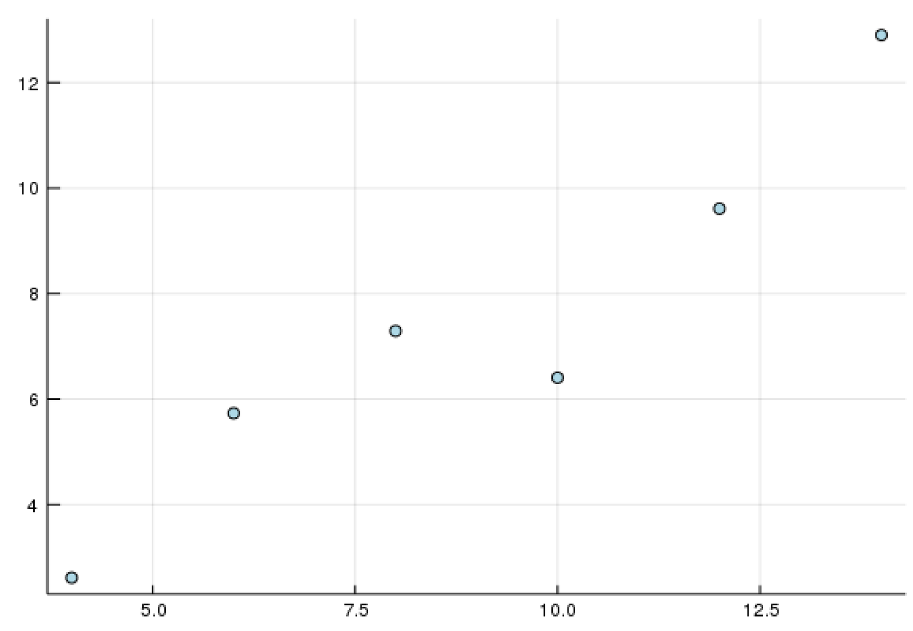
Entrada: $A \in \mathbb{R}^{n\times m}$, $b \in \mathbb{R}^{m\times 1}$
Saída: $\tilde{x} \in \mathbb{R}^{m\times 1}$ solução de $\min\limits_{x\in\mathbb{R}^m} \|Ax - b\|_2^2$
início
  $Q, R \leftarrow$ Decomposição QR de A
  $R_1 \leftarrow R[1:m, :]$
  $b \leftarrow Q^tb$
  $b_1 \leftarrow b[1:m]$
  $\tilde{x} \leftarrow R_1^{-1}b_1$
  retorna $\tilde{x}$
fim

4. Aplicação

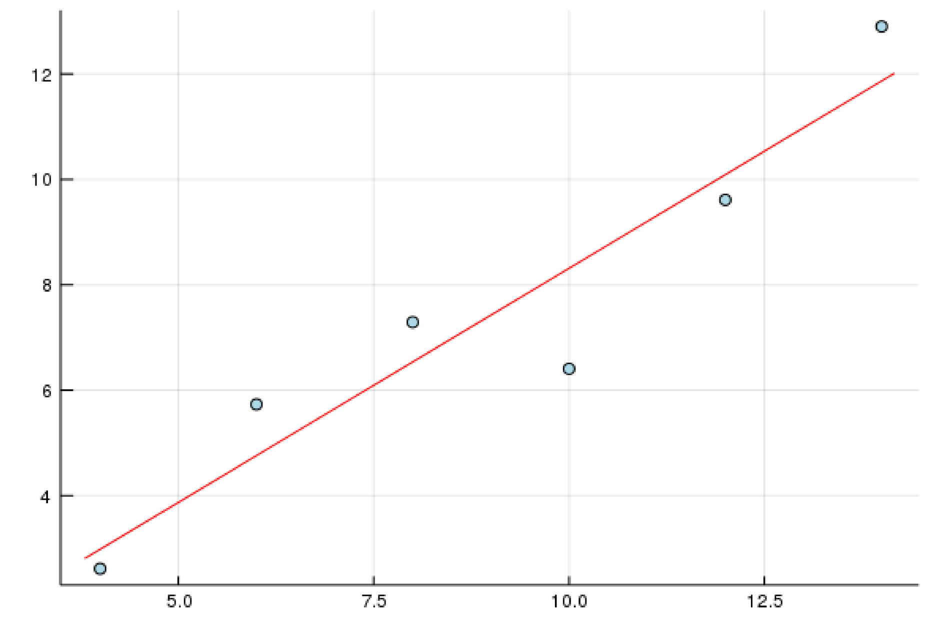
Vejamos uma aplicação na indústria de molas. Para obter a *constante da mola* k proveniente da Lei de Hooke $F = -kx$ são utilizadas medições da força aplicada à mola F e o deslocamento x obtido. A partir das medições, obtemos as seguintes relações:

$$\begin{aligned} 4.0k + p &= 2.614 \\ 6.0k + p &= 5.7309 \\ 8.0k + p &= 7.2909 \\ 10.0k + p &= 6.4058 \\ 12.0k + p &= 9.6112 \\ 14.0k + p &= 12.9009 \end{aligned}$$

Representadas no gráfico:



Aplicando o algoritmo de decomposição QR por rotação em A, obtemos k = 0.8884 e p = 0.5703. Ou seja, a reta que melhor aproxima esses pontos é: $F = \mathbf{0.0884}x - \mathbf{0.5703}$. Graficamente:



5. Exemplos

Exemplo 1: (Matrizes esparsas) Utilizamos a biblioteca SparseArrays para criar uma matriz $A \in \mathbb{R}^{10\times 10}$ esparsa arbitrária e um vetor $b \in \mathbb{R}^{10}$ também arbitrário, obtivemos:

	Tempo (s)	Alocações	Memória
Gram-Schmidt	0.0000150070	351 allocations	57.81 KiB
Rotação	0.0000021864	30 allocations	8.02 KiB

Note que o método das rotações de Givens é mais eficiente.

Exemplo 2: (Matrizes densas) Criando uma matriz densa arbitrária $A \in \mathbb{R}^{10\times 10}$ e $b \in \mathbb{R}^{10}$ um vetor arbitrário, obtivemos:

	Tempo (s)	Alocações	Memória
Gram-Schmidt	0.0000152320	351 allocations	57.81 KiB
Rotação	0.0000409430	910 allocations	145.52 KiB

Já neste exemplo, o método mais vantajoso é o processo de Gram-Schmidt.

6. Comparação dos métodos

Nos exemplos acima podemos ver uma grande diferença entre os valores obtidos, principalmente se tratando de memória utilizada e alocações, sendo que no exemplo 1, a rotação é muito melhor que Gram-Schmidt e no exemplo 2 temos o contrário.

Isso acontece porque para matrizes esparsas o método da rotação é de fato muito mais eficiente, pois quando o aplicamos, procuramos zerar elementos da matriz para encontrar a matriz $\mathbf{R}$ no formato triangular superior. Se a matriz já tem vários zeros (esparsa), então não precisamos zerá-los, portanto, o algoritmo da rotação pulará diversos passos.

Já para matrizes densas, como no exemplo 2, não há zeros na matriz $\mathbf{10} \times \mathbf{10}$, e portanto, para o método de rotação é necessário zerar diversas entradas, ao invés de " pular " algumas, como acontece com matrizes esparsas. Por conta disso, o método de Gram-Schmidt se torna mais vantajoso, já que utiliza menos memória e realiza menos alocações comparado ao método da rotação de Givens.

Concluímos que não podemos generalizar uma maneira mais eficiente de se calcular a fatoração QR, já que para o caso de matrizes esparsas, o preferível é utilizar as rotações de Givens e, para matrizes densas, o melhor é o processo de Gram-Schmidt.

Entretanto, é vantajoso usar métodos QR para resolver estes problemas, já que o custo computacional é bem menor do que ao utilizar o método tradicional (invertendo matrizes).

Referências

[1] ARAUJO, T. P. *Álgebra Linear: Teoria e Aplicações*, SBM, 2014.

[2] STRANGE, G. *Introduction to Linear Algebra* Wellesley, Cambridge Press, 2009.

[3] WATKINS, D. S. *Fundamentals of Matrix Computations*, Wiley-Interscience, 2002.

[4] STEWART, G. W. *Introduction to Matrix Computation*, Academic Press, 1973.

[5] AXLER, S. *Linear Algebra Done Right*, Springer, 2004.