T1 - Gauss e Cholesky

George Othon
NUSP 10349978

April 2020

1 Introdução

Este relatório tem como finalidade analisar os resultados obtidos para solução de um sistema linear Ax = b a partir dos algoritmos de Eliminação de Gauss (Com e sem pivoteamento) e Fatorização de Cholesky. Assim como, também calcularemos o determinante das matrizes.

A análise foi dividida em duas parte, onde na primeira utilizamos a matrix de Hilbert e na segunda um gerador de matrizes aleatórias.

Como métrica para a aproximação obtida pela implementação dos algoritmos foi usada a norma 2.

2 Análise do erro

Para verificar a proximidade entre o vetor solução x e o vetor aproximado $\hat{x}$ utilizamos a norma euclidiana aprenstada à seguir:

$$erro = \|x - \hat{x}\|_2 = \Big(\sum_{i=1}^{n}(x_i - \hat{x}_i)^2\Big)^{\frac{1}{2}}$$

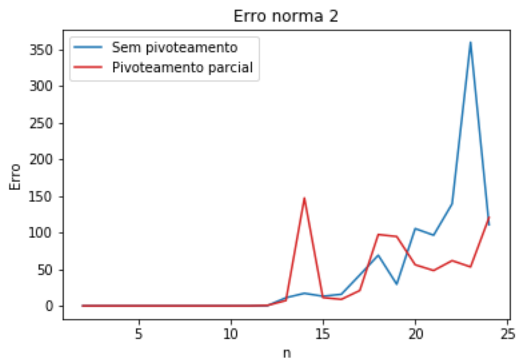
Com o vetor b_n é composto pela soma da das componentes da linha da matriz de A_{nxn}, temos que a solução é $x_i = 1,\ \forall i \in [1, n]$. Sendo assim, quanto mais proximo de zero for o **erro** melhor a implementação do algoritmo.

3 Parte I

Primeiramente, vamos comparar a implemetação da Eliminação de Gauss com pivoteamento e sem pivoteamento, para matrizes de Hilbert.

3.1 Erro

Testamos os dois algoritmos para matrizes quadradas nxn.



Podemos ver que o erro, para os dois métodos cresce bastante a partir de um certo momento. No pivoteamento parcial temos um outlier um pouco antes de chegar no n = 15, mas após n = 20 ele começa a crescer mais devagar do que no sem pivoteamento.

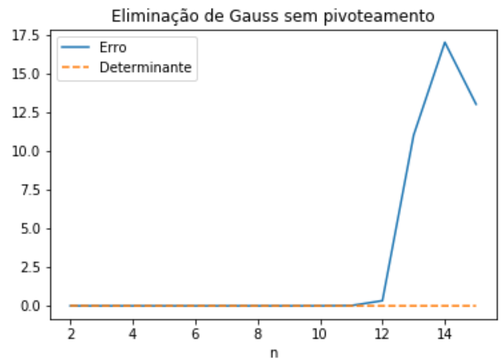
Vamos observar os resultado obtidos até n = 15.

n	Erro Sem pivoteamento	Erro Pivoteamento parcial
2	8.005932e-16	8.005932e-16
3	1.762180e-14	1.390751e-14
4	7.555036e-13	7.723508e-13
5	3.472697e-12	7.811368e-13
6	6.164844e-10	7.610330e-10
7	2.229098e-08	3.758696e-08
8	7.087262e-07	6.093073e-07
9	2.582603e-05	2.996273e-05
10	7.076327e-04	5.156807e-04
11	1.539228e-02	1.528513e-02
12	3.306751e-01	5.756644e-01
13	1.105270e+01	7.233662e+00
14	1.707028e+01	1.470169e+02
15	1.306001e+01	1.129886e+01

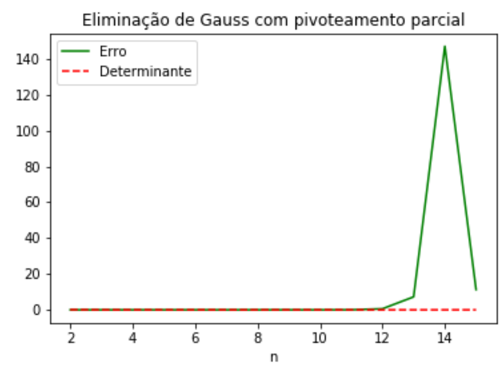
3.2 Determinante

Para comparar o erro e o valor do determinante foi necessário separar os casos, já que o determento fica muito próximo de zeros nos casos em estudo.

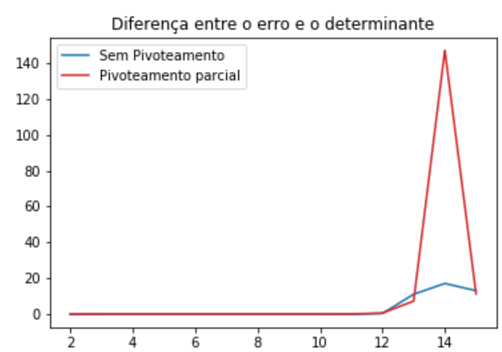
Para a eliminação sem pivoteamento:



E para a eliminação com pivoteamento:



Podemos perceber que até certo ponto, nos dois métodos, o erro fica próximo ao determinante. Para melhor observar foi feita a diferença entre o erro e o determinante.



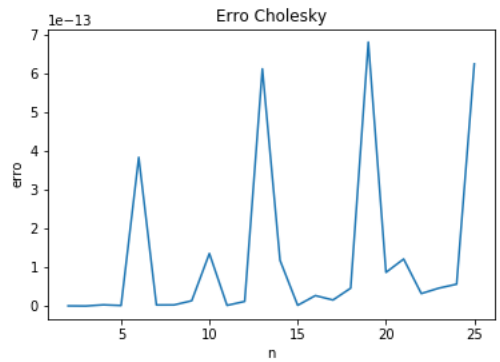
Podemos perceber então que até aproximadamente n = 12, o erro fica bastante próximo ao determinante, nos dois casos estudados.

4 Parte II

Na segunda parte utilizei um gerador de matrizes A_{nxn} obtida através de uma multiplicação entre uma matrix M_{nxn}, onde cada elemento m_{ij} é um número escolhido aleatóriamente do intervalo [-10,10].

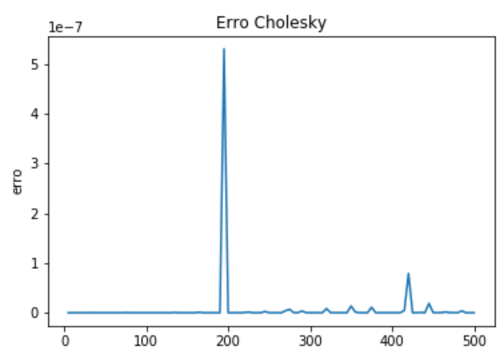
4.1 Erro

Para a decomposição de Cholesky com substiuições direta e reversa obtivemos o seguiinte erro.

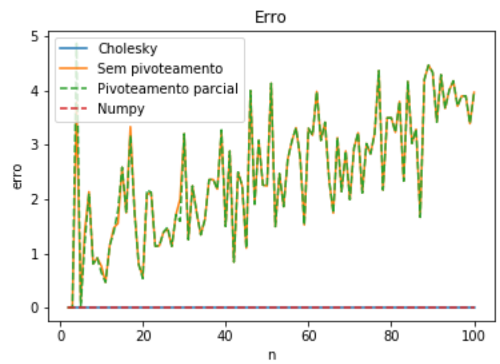


Pode-se perceber que o erro varia bastante, mas é sempre próximo á zero, até n = 25 o erro é menor ou igual a $7x10^{-13}$.

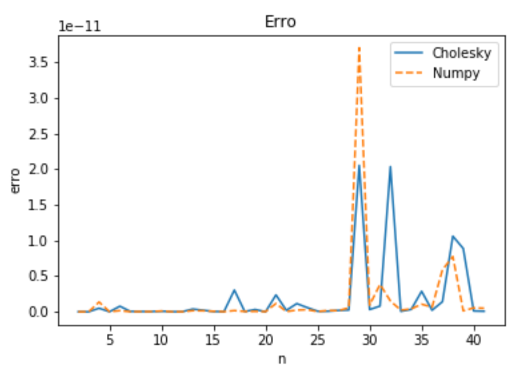
Apesar do crescimento do erro, devido ao maior número de operações para resolver o sistema, o método de Cholesky ainda tem bons resultados para matrizes grandes.



Quando comparamos os dois algoritmos da Parte I, a implementação de cholesky e a função do numpy temos dificuldade em fazer a análise já que existe uma grande discrepância entre os métodos para a mesma matriz.



Podemos observar um erro muito maior na Eliminação de Gauss, e o método do Numpy e de Cholesky ficam próximo de zero.

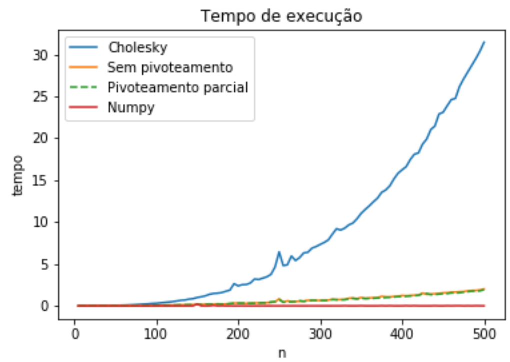


Temos então um erro bastante próximo por nossa implementação de Cholesky e o método linagl.solve.

4.2 Tempo

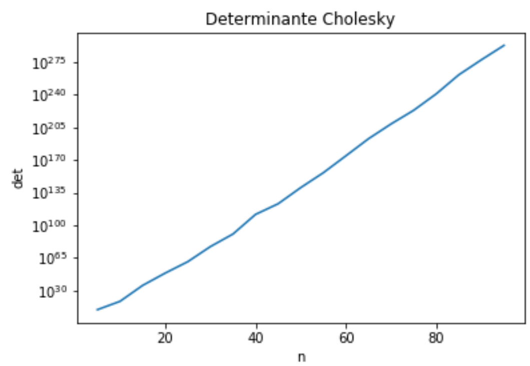
Em relação ao tempo, o método de cholesky se apresentou eficiente para n's baixos.

Embora o código de Cholesky tenha um crescimento exponecial maior que os outros códigos, ainda se mostra bastante eficaz até aproximadament n = 40, onde fica próximo aos outros métodos.



4.3 Determinante

A análise do determinante foi a mais complexa, já que o determinante tem uma variação muito grande com o aumento do n.



Foi utilizado um gráfico com escala logarítmica para analisar o crescimento do determinante. Logo, analisando o gráfico, percebemos uma elevação de forma exponencial.

Conclusão

Com base no que foi apresentado, para as matrizes de Hilbert, o erro foi baixo para valores pequenos de n, fazendo com que os algoritmos de Eliminação funcionassem bem para n até 12.

Tivemos bons resultados com os métodos de Eliminação de Gauss e com a Fatorização de Cholesky nas matrizes geradas aleatóriamente, apontando um erro muito menor para Cholesky do que paras as eliminações de Gauss com e sem pivoteamento. Já ao examinar o tempo computacional, o método de Cholesky tem um crescimento exponencial muito maior que a eliminação. O determinante teve um crescimento exponencial, a escala logarítmica nos ajudou na vizualização da evolução do determinante, onde, nessa escala, nos permite observar uma reta, tornando mais fácil a interpretação da tendência do determinante com o aumento de n.

A Fatorização de Cholesky foi o método escolhido como mais eficaz, já que seu erro para n's alto se manteve extremamente próximo de zero, mesmo apresentando um tempo maior, com um crescimento muito mais rápido se mostrou o melhor para resolver sistemas lineares.