Escola Politécnica da USP
MAP3122 - Métodos Numéricos e Aplicações

Exercício Programa I

Paulo Henrique Diniz Fernandes
N°USP: 11257630

Professor Dr. Antoine Laurain

São Paulo
2021

Sumário

1 Introdução

Este relatório vem mostrar a solução e a problematização sobre o exercício programa da disciplina MAP3122 - Métodos Numéricos e Aplicações intitulado Tomografia Computadorizada. O problema proposto tem como objetivo apresentar ao alunado um problema altamente relevante em várias áreas do conhecimento que é o chamado problema inverso. Para resolvê-lo utilizaremos a linguagem de programação Python e suas respectivas bibliotecas permitidas.

2 Desenvolvimento

2.1 Exercício 1

O primeiro exercício consta somente de projeções horizontais e verticais. O valor n encontrado para cada medida é obtido do arquivo de formato .npy que nos retorna um vetor. No caso do exercício 1, o valor de n é metade do tamanho deste vetor. Em posse desse dado, conseguimos construir a matriz A e respectivamente a matriz requerida $A^TA+\delta$ I_{n^2} para determinado δ. Segue abaixo tabela com os doze valores de determinante dessa matriz requerida para os valores de δ solicitados.

Determinantes tomo1.py			
δ	n = 5	n =10	n =30
0	0	0	0
0.001	$3.912895627183303*10^{-42}$	$2.0037032417676077*10^{-224}$	0
0.01	$3.9731584436353565*10^{-26}$	$2.0373257919534943*10^{-143}$	0
0.1	$4.622562401610566*10^{-10}$	$2.4042564261302438*10^{-62}$	0

Tabela 1: Tabela dos determinantes do exercício 1.

A solução da equação $\left(A^TA + \delta I_{n^2}\right) f_\delta = A^Tp$ foi obtida no programa utilizando a decomposição LU. Ela nos dá as seguintes seguintes matrizes plotadas.

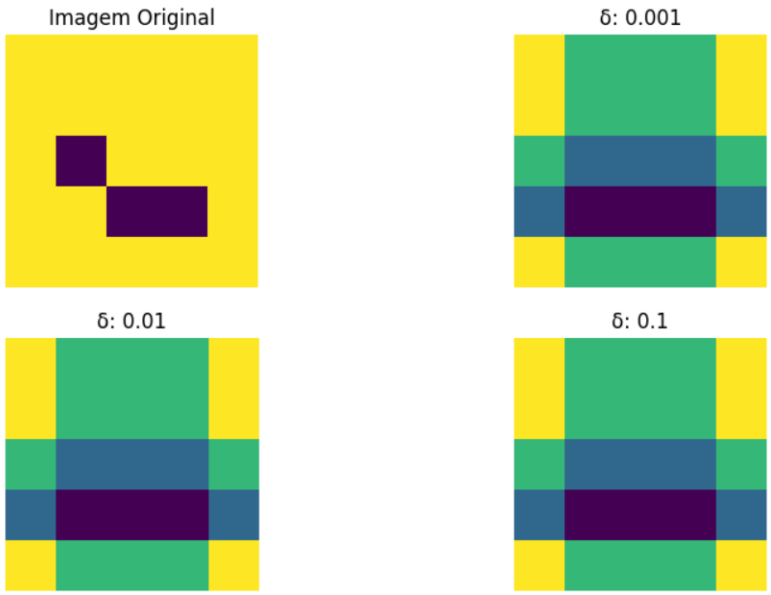

Figura 1: Imagens das matrizes-solução de im1 para respectivos δ.

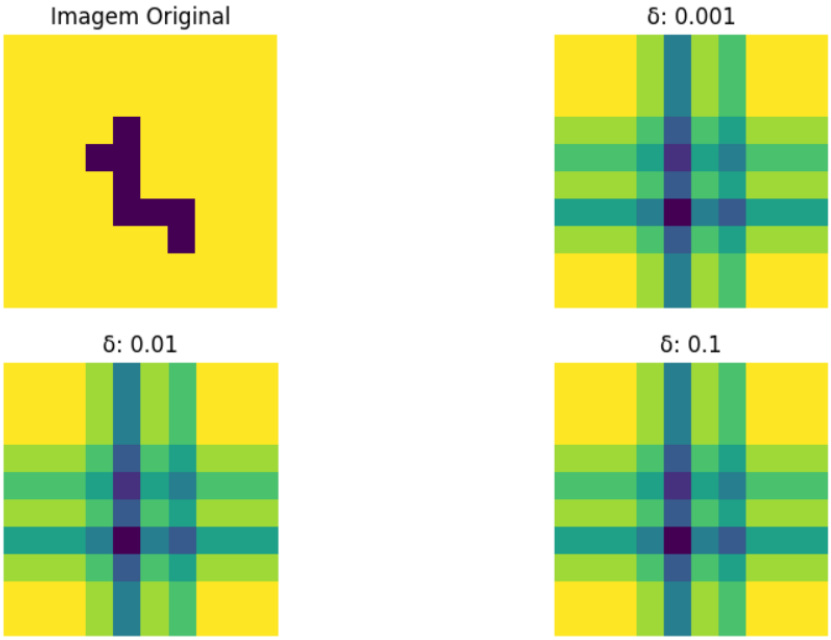

Figura 2: Imagens das matrizes-solução de im2 para respectivos δ.

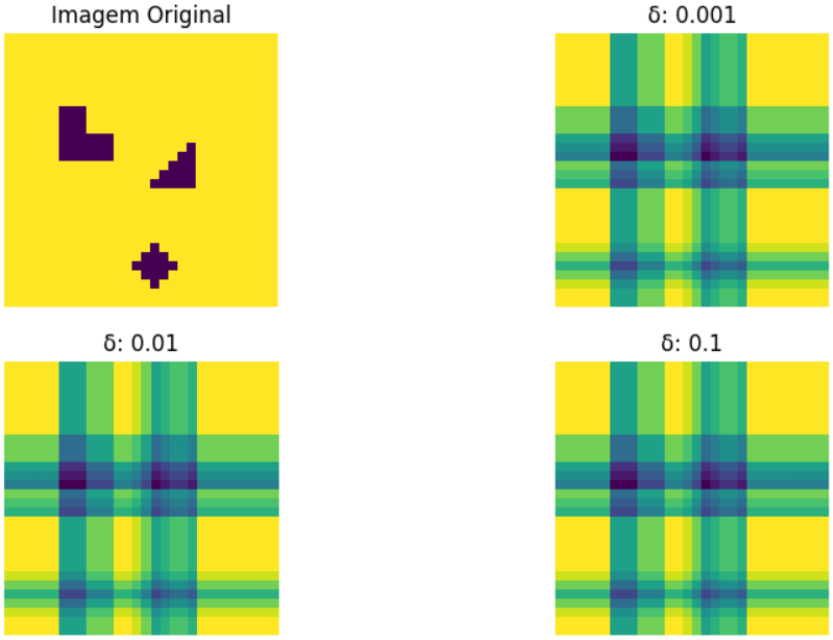

Figura 3: Imagens das matrizes-solução de im3 para respectivos δ.

Conforme observado nas três figuras é possível ver que os valores de δ não alteram visualmente as imagens, mais pra frente quando calcularmos o erro poderemos discutir se essas imagens são todas iguais.

2.2 Exercício 2

O exercício dois conta com mais projeções, aqui será introduzida as projeções diagonais para melhorar o resultado da saída.

Da mesma forma que no exercício 1, o n é encontrado a partir do arquivo .npy que retorna um vetor. Diferentemente do exercício 1, o valor de n aqui é dado sabendo que o vetor tem tamanho $(6n - 2)$ e não $2n$ como anteriormente. Em posse desse dado n, construi-se a matriz A e a matriz requerida $A^T A + \delta\ \mathrm{I}_{n^2}$ para determinado δ. Segue abaixo tabela com os doze valores de determinantes dessa matriz requerida para os valores de δ solicitados.

Determinantes tomo2.py			
δ	n = 5	n =10	n =30
0	$3.412514887920707*10^{-52}$	0	0
0.001	0.8733266215324784	$1.2573406037931072*10^{-107}$	0
0.01	9364.345078323937	$1.4015508634694351*10^{-58}$	0
0.1	184072192.9951481	$4.0440174074925725*10^{-09}$	0

Tabela 2: Tabela dos determinantes do exercício 2.

Da mesma forma que no exercício 1, a solução da equação $\left(A^T A + \delta I_{n^2}\right) f_\delta = A^T p$ nos dá as seguintes matrizes plotadas.

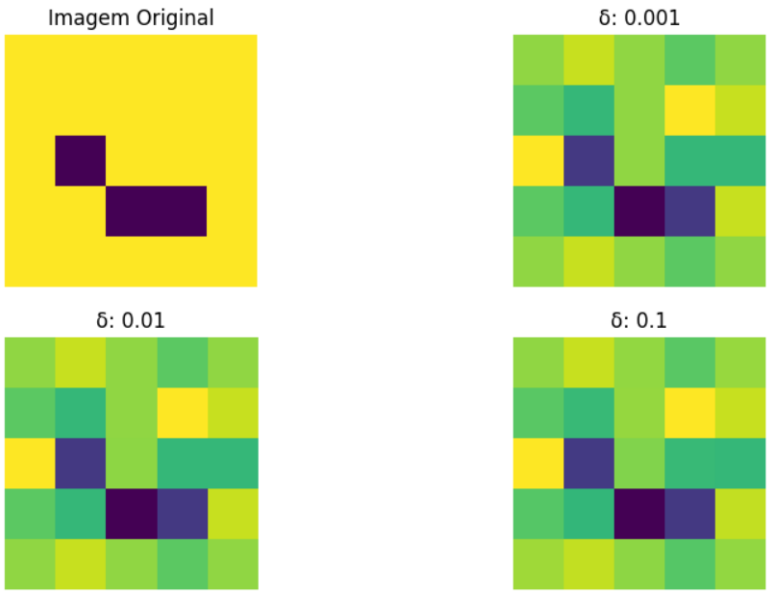

Figura 4: Imagens das matrizes-solução de im1 para respectivos δ do exercício 2.

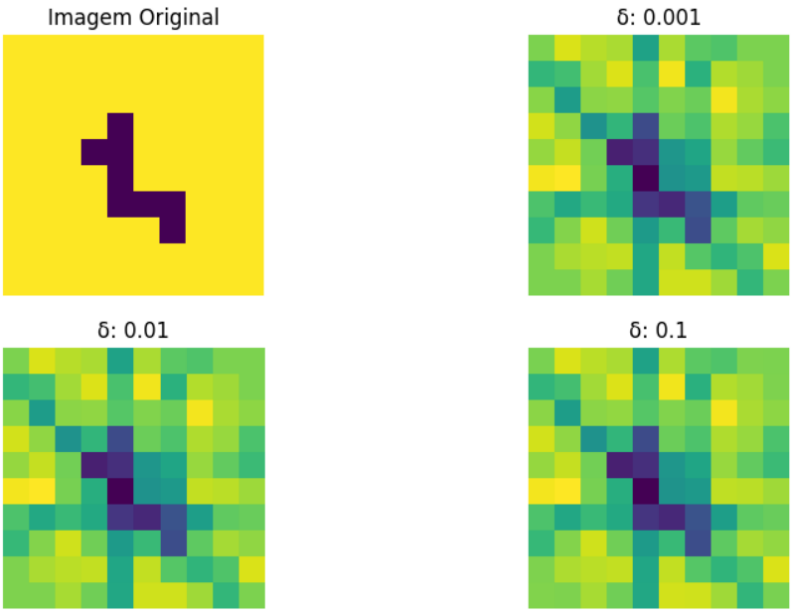

Figura 5: Imagens das matrizes-solução de im2 para respectivos δ do exercício 2.

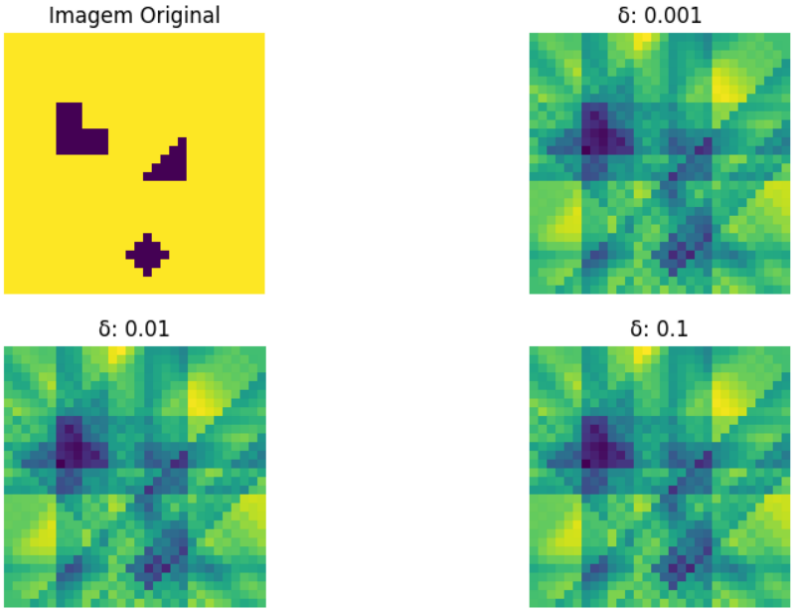

Figura 6: Imagens das matrizes-solução de im3 para respectivos δ do exercício 2.

Vale a pena salientar que o determinante de $n = 5$ vai para infinito com o aumento de δ, mas o ponto principal é o notório aumento da semelhança das imagens projetadas com relação à imagem original.

2.3 Cálculo dos erros

Podemos calcular o erro com a seguinte equação:

$$\text{erro } = 100 \times \frac{\sqrt{\sum_{j=1}^{n^2}\left(f_j - f_j^*\right)^2}}{\sqrt{\sum_{j=1}^{n^2}\left(f_j^*\right)^2}} \tag{1}$$

Onde f^* refere-se à imagem original e f à reconstrução. Fazendo para o exercício 1 e o exercício 2, temos as tabelas a seguir.

Erro (%) tomo1.py			
δ	n = 5	n =10	n =30
0.001	28.284273085507287	22.936586107309825	21.909202450174238
0.01	28.284454671170355	22.936641593513546	21.909208669330692
0.1	28.302233967099973	22.942134278329753	21.909828610806784

Tabela 3: Tabela dos erros do exercício 1.

Erro (%) tomo2.py			
δ	n = 5	n =10	n =30
0.001	13.484012598141238	16.293043864572827	19.923008257242127
0.01	13.485504889550997	16.293128086953203	19.9230145174292
0.1	13.611143403518271	16.30117486474399	19.923614298433897

Tabela 4: Tabela dos erros do exercício 2.

Note-se que os valores estão exatamente como saíram da variável. Isto acontece porque no programa original (tomo1.py e tomo2.py) foram instroduzidos partes no código para que esses valores fossem escritos num arquivo texto de saída (txt), facilitando assim a obtênção desses dados e por conseguinte a formação dessas tabelas.

Há de se destacar que o aumento de n fez com que no exercicío 1 os valores do erro diminuíssem, o que acontece de maneira inversa no exercício 2, o erro aumenta com o aumento de n. Esse comportamento é explicado pela diferença dos tamanhos de p. No caso um, o p tem tamanho $2n$ enquanto a solução f tem n^2 e o caso dois, o p tem tamanho $6n - 2$ e a solução f continua no tamanho de n^2. Isso evidencia que para uma mesma quantidade de soluções o segundo caso precisa de mais medições e, por consequência, o aumento de n diminui a nitidez de suas imagens.

Outro ponto importante é que nas imagens que foram mostradas acima, para determinado n, aparenta-se que as imagens são iguais independentemente do δ, mas analisando esses erros podemos constatar que as imagens não são iguais, elas são ligeiramente diferentes, mas nossos olhos não conseguem distinguir tão pouca diferença.

3 Conclusão

Conforme exposto no desenvolvimento deste projeto, pode-se concluir que o projeto teve êxito em seus propósitos. Pode-se citar como destaques: o exercício de programa uma linguagem muito difundida e consolidada (Python), programar partes importantes da disciplina (Solução por LU, por exemplo) e apresentar um problema real que é bastante utilizado tanto no mercado como na pesquisa.

4 Bibliografia

Referências

[1] Site da disciplina com o respectivo problema. https://edisciplinas.usp.br/pluginfile.php/5877514/mod_resource/content/6/EP1_V4.pdf. Acessado em 15/02/2021.

[2] LU Decomposition for Solving Linear Equations. https://courses.physics.illinois.edu/cs357/sp2020/notes/ref-9-linsys.html. Acessado em 15/02/2021.