TRABALHO COMPUTACIONAL MAP2220

GUSTAVO SOARES GOMES, NUSP 12557438

SÃO PAULO, JANEIRO/2023

Questão 1

itens a) & b)

A implementação do método de Newton foi feita baseando-se no algoritmo dado no livro-texto da disciplina. Com respeito à Broyden, existem dois algoritmos distintos : o algoritmo de Broyden Good é aquele em que atualizamos a aproximação da própria jacobiana A a partir da Jacobiana da iteração anterior, já o algoritmo Broyden Bad é aquele que atualizamos a aproximação da inversa da jacobiana a partir da fórmula de Sherman-Morrison. O Good Broyden foi implementado com base em [1], já o Bad Broyden foi implementado com base no algoritmo do livro-texto.

Para todos os métodos, o grau de tolerância escolhido foi de 10^{-9}, o ponto inicial $X^{(0)} = (10, 10, 10, 10, 10)$ e o número máximo de iterações $N = 1000$. A taxa de convergência foi medida tomando-se a norma euclidiana do incremento em cada iteração do vetor x, neste caso, o autor atribui a variável y a esse incremento. Quanto mais próximo de zero essa norma, mais esperamos uma convergência para uma solução do sistema.

Na execução do Bad Broyden, não houve convergência no ponto inicial proposto, portanto também aplicamos todos os métodos com um ponto incial mais próximo da solução no qual houve convergência no Bad Broyden: $X^{(0)} = (10, 30, 10, 10, 10)$. Ademais, a tabela a seguir contem o tempo necessário e a quantidade de iterações necessárias para os algoritmos convergirem para a solução proposta no enunciado.

Método / $X^{(0)}$	$(10, 10, 10, 10, 10)$	$(10, 30, 10, 10, 10)$
Newton	12 iterações em 0.13s	12 iterações em 0.18s
Bad Broyden	Não converge	96 iterações em 1.12s
Good Broyden	317 iterações em 1.58s	95 iterações em 0.45s

item c)

Como vimos no livro-texto, a convergência do método de Newton é quadrática, enquanto que a convergência do método de Broyden é linear, portanto era esperado que o método de Newton chega-se na tolerância num número menor de iterações do que o método de Broyden começando de um mesmo ponto inicial, o que de fato ocorreu quando comparamos Newton tanto com o Bad como também com o Good Broyden.

Em relação aos métodos de Broyden iniciados em $(10, 30, 10, 10, 10)$, de acordo com [2] era esperado que ambos tivessem aproximadamente o mesmo número de iterações, o que de fato ocorreu. Como também vimos, a quantidade de computação exigida pelo método de Newton é superior ao do método de Broyden, já que em Newton temos que encontrar a jacobiana em cada iteração e ainda resolver um sistema linear que tem um custo de $O(n^3)$, portanto é esperado que o algoritmo de Newton leve mais tempo para fazer um mesmo número de iterações que o algoritmo de Broyden. Na nossa experimentação, tomando como base o ponto inicial $X^{(0)} = (10, 30, 10, 10, 10)$, a quantidade média de iterações feitas em 1 segundo em Newton foi de que $12/0.18 = 66.6$ iterações/s, em Bad Broyden foi de $96/1.12 = 85.71$ iterações/s e Good Broyden foi de $95/0.45 = 213.3$ iterações/s, o que comprova a teoria. Finalmente, podemos perceber pela tabela que os algoritmos de Broyden são muito mais sensíveis à mudança do ponto inicial do que o algortimo de Newton tanto em relação ao tempo como tempo ao número de iterações até a convergência.

Questão 2

Por Diferenças Finitas de Ordem 1 a) & b)

Em todos os métodos, ao invés de calcularmos explicitamente a jacobiana, nós iremos recorrer a fórmula de diferenças finitas de ordem 1 para tomar uma aproximação:

$$\frac{\partial f_i}{\partial x_j}(a) = \frac{f_i(a + h_j) - f_i(a)}{h}$$

onde h_j é o vetor nulo exceto na j-ésima posição aonde há h. Assim, as únicas alterações nos algoritmos foram calcular todas essas n^2 diferenças finitas para cada iteração no caso de Newton e apenas para a primeira iteração nos métodos de Broyden.

Para escolha do h, o livro - texto foi analisado: o menor h escolhido dentre exercicios e exemplos foi de 0.01, o que é um valor não tão grande para atrasar a convergência do método e também nem tão pequeno para exigir processamento da CPU, portanto $h = 0.01$ é o valor que usaremos nessa questão. Mantendo os mesmos pontos iniciais e mesma tolerância, obteve- se a seguinte tabela para os métodos:

	Diferenças Finitas de Ordem 1 com $h = 0.01$		
Método / $X^{(0)}$	$(10, 10, 10, 10, 10)$	$(10, 30, 10, 10, 10)$	$(1, 30, 1, 1, 1)$
Newton	16 iterações em 0.34s	17 iterações em 0.36s	13 iterações em 0.28s
Bad Broyden	Não converge	Converge errado	21 iterações em 0.35s
Good Broyden	Não converge	101 iterações em 0.22s	20 iterações em 0.06s

Durante a execução do método Bad Broyden no ponto inicial $(10, 30, 10, 10, 10)$ houve uma convergência para o seguinte vetor X com a taxa de convergência y associada:

$$X^{(174)} = (0.00310, 34.78909, -0.06518, 0.85991, 0.03700)$$

$$y = 2.35409307555002 \cdot 10^{-10}$$

Note que houve convergência para uma solução de fato, pois

$$F(X^{(174)}) = (-3.51864 \cdot 10^{-12}, 1.64187 \cdot 10^{-10}, 3.44701 \cdot 10^{-10}, -3.92264 \cdot 10^{-12}, 1.66866 \cdot 10^{-10})$$

Adotando a nossa tolerância, podemos assumir que $F(x)$ de fato está convergindo pra 0. Porém, como a terceira componente é negativa, estamos convergindo para uma solução que está fora do domínio especificado, isto porque provelmente começamos muito longe da solução ideal e o passo de $h = 0.01$ não é suficiente para garantir a convergência para a solução ideal. Existem então duas estratégias: ou aumentamos o valor do passo h ou começamos em um ponto inicial mais próximo da solução inicial. Para efeitos de comparação dos algoritmos, escolheu-se alterar o ponto inicial para $(1, 30, 1, 1, 1)$.

Por Diferenças Finitas de Ordem 2 a) & b)

Aqui, ao invés de uma diferença de ordem 1, usaremos a fórmula de três pontos centrada de ordem 2:

$$\frac{\partial f_i}{\partial x_j}(a) = \frac{f_i(a + h_j) - f_i(a - h_j)}{2h}$$

onde h_j é o vetor nulo exceto na j-ésima posição aonde há h. A tabela obtida para os métodos e seus respectivos pontos inicias é exibida a seguir.

	Diferenças Finitas de Ordem 2 com $h = 0.01$		
Método / $X^{(0)}$	$(10, 10, 10, 10, 10)$	$(10, 30, 10, 10, 10)$	$(1, 30, 1, 1, 1)$
Newton	12 iterações em 0.33s	12 iterações 0.33s	9 iterações em 0.25s
Bad Broyden	Não converge	Não converge	19 iterações em 0.21s
Good Broyden	Não converge	Converge errado	18 iterações em 0.06s

item c)

Como vimos na teoria, a aproximação de ordem 2 é mais precisa que a de ordem 1, portanto era esperado, em todos os métodos, que o número de iterações necessárias para alcançar tolerância em ordem 2 fosse menor ou igual comparado a quantidade necessária em ordem 1, o que de fato ocorreu se compararmos todos os métodos inciados no ponto $(1, 30, 1, 1, 1)$.

Pode ser observado pelas fórmulas de diferenças finitas que a quantidade de cálculos de f na aproximação de ordem 2 é a mesma do que a na aproximação de ordem 1, portanto é esperado que o tempo para realizar uma iteração de ordem 2 seja aproximadamente mesmo para realizar um iteração de ordem 1. Ainda tomando como referência o ponto inicial $(1, 30, 1, 1, 1)$ a quantidade média de iterações foi de $13/0.28 = 46.43$ iterações/s em Newton, de $21/0.24 = 87,5$ iterações/s em Bad Broyden e de $20/0.06 = 333,3$ iterações/s em Good Broyden usando ordem 1. Agora utilizando ordem 2, a quantidade média de iterações foi de $9/0.25 = 36$ iterações/s em Newton, de $19/0.21 = 90.48$ iterações/s em Bad Broyden e de $18/0.06 = 300$ iterações/s usando ordem 2, o que sustenta o resultado teórico.

Em relação aos métodos de Broyden, de acordo com [3] era esperado que esses dois métodos mantivessem aproximadamente o mesmo número de iterações, o que de fato aconteceu, tanto em ordem 1 ou como em ordem 2. Ademais, ainda de acordo com [3], era previsto que o Bad Broyden sofresse mais com a aproximação da Jacobiana no quesito tempo, o que de fato foi verificado nas duas aproximações.

Questão 3

Assim como foi feito anteriormente, tentaremos encontrar uma solução analítica para o sistema e comparar o resultados obtidos por ambos os métodos.

Nas duas primeiras equações podemos fazer uma bijeção entre as variáveis e senos e cossenos, como segue.

$$x_1 = sen\theta, x_2 = sen\phi, x_3 = cos\theta, x_4 = cos\phi$$

Já substituindo, as outras 4 equações restantes serão:

$$\begin{cases} (1) : x_5 \cdot cos^3\theta + x_6 \cdot cos^3\phi = 0 \\ (2) : x_5 \cdot sen^3\theta + x_6 \cdot sen^3\theta = 0 \\ (3) : x_5 \cdot sen\theta cos^2\theta + x_6 \cdot sen\phi cos^2\phi = 0 \\ (4) : x_5 \cdot sen^2\theta cos\theta + x_6 \cdot sen^2\theta cos\phi = 0 \end{cases}$$

Supondo que $sen^3\theta, cos^3\theta \neq 0$, isto é, $sen\theta, cos\theta \neq 0$, o sub-sistema formado por (1) e (2) resultará em:

$$x_6(sen^3\theta cos^3\phi - sen^3\phi cos^3\theta) = 0$$

1) $x_6 = 0$ e $sen^3\theta cos^3\phi - sen^3\phi cos^3\theta \neq 0$

Da equação (1) tiramos $x_5 = 0$ e os valores obtidos de x_5 e x_6 satisfazem (3) e (4), logo uma solução é:

$$\boxed{X = (sen\theta, sen\phi, cos\theta, cos\phi, 0, 0)}$$

com $\theta, \phi \in [0, 2\pi]$

2) $sen^3\theta cos^3\phi - sen^3\phi cos^3\theta = 0$ e $x_6 = \alpha \neq 0$

Supondo $cos\phi \neq 0$,

$$sen^3\theta cos^3\phi = sen^3\phi cos^3\theta \implies tg^3\theta = tg^3\phi \implies tg\theta = tg\phi \implies \phi = \theta + k\pi, k = 0, 1$$

A penultima passagem se dá pelo fato de a função $f(x) = x^3$ ser injetora. Usando o fato de que $sen(\theta + \pi) = -sen(\theta)$ e $cos(\theta + \pi) = -cos(\theta)$ em (3), chegamos que $x_5 = \alpha$, se $k = 1$ e $x_5 = -\alpha$, se $k = 0$. Além disso, em ambos os casos a eq. (4) é satisfeita, portanto outras duas soluções são

$$\boxed{X = (sen\theta, -sen\theta, cos\theta, -cos\theta, \alpha, \alpha)}$$

$$\boxed{X = (sen\theta, sen\theta, cos\theta, cos\theta, -\alpha, \alpha)}$$

com $\theta, \phi \in [0, 2\pi]$ e $\alpha \in \mathbb{R}^*$

Por Newton

Foram selecionados aleatoriamente 8 pontos no hipercubo de aresta 5 a fim de não demandar muito tempo para a convergência do método. A tabela a seguir contêm as soluções para quais os pontos iniciais convergiram e nos mostra que o método de Newton tende a convergir para soluções do tipo 1, independente do ponto inicial escolhido.

Ponto Inicial	Convergência
(-5, 2, -3, 1, 3, -3)	(-0.8575, 0.8944, -0.5145, 0.4472, 0.0, 0.0)
(-2, -5, -5, 2, -4, 3)	(-0.3714, -0.9285, -0.9285, 0.3714, 0.0, 0.0)
(-4, -3, -2, 2, -3, 1)	(-0.8944, -0.8321, -0.4472, 0.5547, 0.0, 0.0)
(-2, 3, -1, 5, 4, -1)	(-0.8941, 0.5145, -0.4478, 0.8575, 0.0, 0.0)
(0, 1, 2, -5, 1, -5)	(-0.0370, 0.1961, 0.9993, -0.9806, 0.0, 0.0)
(2, -3, -5, -1, 1, 3)	(0.3714, -0.9487, -0.9285, -0.3162, 0.0, 0.0)
(2, 3, 3, 0, -1, -5)	(0.5547, 0.8958, 0.8321, -0.4444, 0, 0.0)
(-5, 0, 0, 2, -4, -4)	Não houve convergência

Pela tabela também podemos ver que provavelmente não existe a bola de escolha de pontos iniciais que garante a convergência do método para uma solução. De fato, podemos refutar a hipótese 3 do Teorema 10.7 do livro-texto, pois ao transformarmos as equações na forma $G(x) = x$, vemos que $\frac{\partial g_4}{\partial x_4}(a) = 1, \forall a$, portanto não há garantia de convergência para uma solução e os resultados corroboram a essa hipótese, porém pode ser que ainda sim exista uma região de convergência.

Por Good Broyden

Novamente foram selecionados 8 pontos aleatorios dentro do hipercubo de aresta 5. Após a análise da tabela que vem a seguir, conclui-se que este método apresentou uma variedade maior de resultados, com soluções de todos os tipos, com prevalência do tipo 1.

Ponto Inicial	Convergência
(1, 2, 1, 5, 0, 1)	(0.3714, 1.0, 0.9285, 0.0, 0.0, 0.0)
(-3, -2, 4, -3, -2, -1)	(-0.8212, 0.8212, 0.5706, -0.5706, 0.5592, 0.5592)
(-1, 4, -3, -1, 4, 4)	Não houve convergência
(-5, 5, -2, 5, -1, 5)	Não houve convergência
(2, 2, -3, -3, 3, 3)	Não houve convergência
(-4, 4, 4, 4, 1, 5)	(0.8468, 0.8468, -0.5319, -0.5319, -3.2093, 3.2093)
(0, 0, -5, -3, -2, -1)	(0.0, -0.9285, -1.0, -0.3714, 0.0, 0.0)
(0, 3, 2, 2, 1, -3)	Não houve convergência

Apesar da maior diversidade de soluções, o risco de não convergência é bem maior se comparado ao Método de Newton, resultado que esperavamos e que foi discutido nas questões anteriores.

Por Bad Broyden

Novamente foram selecionados 8 pontos aleatórios do hipercubo de aresta 5. Após a análise da tabela que vêm a seguir, percebe-se que o método converge prioritamente para soluções do tipo 1, assim como o método de Newton.

Ponto Inicial	Convergência
(-5,-1,-5,0,-5,4)	(-0.7071, -1.0,-0.7071, 0.0, 0.0, 0.0)
(-5, 2, 3, 2, 5, 1)	(-0.8575, 0.7071, 0.5145, 0.7071, 0.0, 0.0)
(1, -5, 2, 4, -1, -5)	(0.4472, -0.7809, 0.8944, 0.6247, 0.0, 0.0)
(-1, -5, -2, 3, 4, -1)	(-0.4472, -0.8575, -0.8944, 0.5145, 0.0, 0.0)
(2, 4, -4, 1, -5, 2)	(0.4472, 0.9701, -0.8944, 0.2425, 0.0, 0.0)
(3, -3, -1, 4, -2, 1)	(0.9487, -0.6000, -0.3162, 0.8000, 0.0, 0.0)
(0, 3, 5, -3, 0, -2)	Não houve convergência
(1, 5, 2, 1, 3, -1)	Não houve convergência

Conclusão

O método de Newton nos fornece apenas soluções do tipo 1, porém com uma taxa de convergência grande. O Bad Broyden também fornece apenas soluções do tipo 1, porém com uma taxa de convergência menor quando comparado a Newton. Finalmente, o Good Broyden nos oferece uma diversidade maior de soluções, porém com uma taxa relativamente baixa de convergência.
Tendo isso em vista, é recomendado que o leitor que deseja encontrar uma quantidade grande de soluções num curto período de tempo utilize Newton, enquanto que o leitor interessado em obter o maior conjunto possivel de soluções utilize Good Broyden com número de iterações grande para obter uma quantidade de soluções razoável.

Código Fonte

Sobre acesso

O código fonte em Jupyter Notebook pode ser acessado em [4] por meio do Google Colab, caso queira abrir o arquivo em um editor de sua preferência instalado em máquina, basta baixar o arquivo .ipynb enviado no email junto com este relatório e abrir em seu aplicativo.

Sobre edição

Cada método é feito em uma célula, caso desejar alterar algum parâmetro (ponto inicial, passo, tolerância, etc.) clique sobre a célula e substitua o parâmetro em questão.

Bibliografia

[1] - https://nickcdryan.com/2017/09/16/broydens-method-in-python/;

[2] - Dachao Lin, Haishan Ye, Zhihua Zhang. Explicit Superlinear Convergence Rates of Broyden's Methods in Nonlinear Equations;

[3] - Eric Kvaalen. A faster Broyden method;

[4] - https://colab.research.google.com/drive/1AyGTu5_oEKuafK69Xch2MdBS0nO-qBOf?usp=sharing