

Trabalho Computacional de Otimização

Período 2022.2

O objetivo deste trabalho é analisar e implementar modelos de otimização para a minimização da função abaixo:

$$f(x, y) = x^2 + (e^x - y)^2$$
$$(x, y) \in \mathbb{R}^2$$

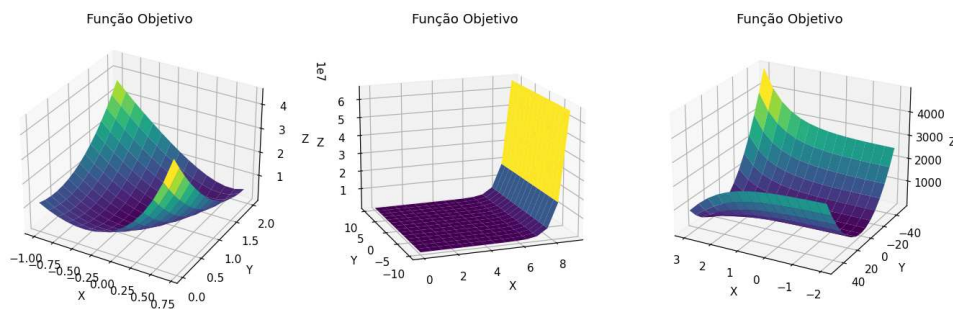
1. Estudo da Função

Como pode-se perceber no terceiro gráfico, a função não é convexa. Considerando que funções convexas possuem matriz hessiana **positiva definida** em todo seu domínio, nota-se que esse não é o caso aqui: a hessiana no ponto $(3, 40)$, por exemplo, é indefinida, pois tem autovalores positivos e negativos.

Fazendo $\det(\nabla^2 f(x, y) - \lambda I) = 0$ no ponto em questão obtemos $\lambda_1 = -34.8817$ e $\lambda_2 = 45.7539$, o que demonstra que a função não é convexa. [1] (https://www.wolframalpha.com/input?i=%282-1%29%282+%2B4e%5E%28x%29+-+2y%28e%5Ex%29+-+1%29+-+4*e%5E%28x%29+%3D+0%2C+x+%3D+3%2C+y+%3D+40%29).

A função possui apenas um ponto crítico em $(x, y) = (0, 1)$, pois esse é o único ponto que satisfaz $\nabla f(x, y) = 0$. Esse também é o ponto ótimo que minimiza a função.

A função dispara para valores de x muito positivos, se afastando muito rapidamente do ótimo nessa região.



2. Métodos e Critérios Utilizados

Os métodos de otimização implementados foram o Método de Newton, o Método do Gradiente, e os Métodos de Quase-Newton DFP e BFGS. Todos eles funcionam da mesma forma: é tomado um ponto (x_0, y_0) inicial e, enquanto um determinado critério de parada não for satisfeito, em cada iteração k o ponto atual (x_k, y_k) vai sofrer um deslocamento t_k em uma direção d_k , de forma que:

$$(x_{k+1}, y_{k+1}) = (x_k, y_k) + t_k d_k$$

O passo t_k foi obtido através da Busca de Armijo em todos os métodos, usando os parâmetros $\gamma = 0.8$ e $\eta = 0.25$.

A direção d_k depende do método utilizado:

- Método do Gradiente: $d_k = -\nabla f(x, y)$
- Método de Newton: $d_k = -(\nabla^2 f(x, y))^{-1} \nabla f(x, y)$
- Métodos Quase-Newton: $d_k = -H_k \nabla f(x, y)$
 - $H_0 = I$

Com exceção do Método de Newton, o critério de parada escolhido foi $\nabla f(x, y) \approx 0$, com tolerância de 1×10^{-6} , junto a um limite de iterações em que $N^\circ \text{ de Passos} \leq 500$.

No Método de Newton o critério de parada escolhido foi $\frac{1}{2} \lambda^2(x, y) \approx 0$, com tolerância de 1×10^{-6} , em que $\lambda(x, y) = (\nabla f(x, y)^T \nabla^2 f(x, y)^{-1} \nabla f(x, y))^{\frac{1}{2}}$ é chamado **Decremento de Newton** e é uma estimativa da distância entre o ponto atual e o ponto ótimo. Além disso, foi usado um limite de iterações em que $N^\circ \text{ de Passos} \leq 500$.

2. Teste e Análise com alguns Pontos Iniciais

Para testar os métodos e achar o mínimo da função, vamos analisar o comportamento dos modelos de otimização com cinco pontos iniciais distintos:

- $P_1 = (0, 0)$
- $P_2 = (0, 2)$
- $P_3 = (4, 10)$
- $P_4 = (7, 4)$
- $P_5 = (9, -2)$

2.1. Ponto $P_1(x, y) = (0, 0)$

O ponto $(0, 0)$ está bem próximo do ponto mínimo de $f(x, y)$, então a convergência é muito rápida no Método de Newton, de forma que ele teve a melhor performance.

Os outros métodos tiveram desempenho parecido entre si, com o Método BFGS tendo sido levemente mais rápido.

Todos os métodos foram capazes de convergir para o ponto ótimo $(x^*, y^*) = (0, 1)$.

Começando Metodo do Gradiente com Ponto Inicial $[[0]$
 $[0]]$, onde a F.O. vale 1.00000000

Passo Nº 1, Ponto Atual = $[-0.524288]$
 $[0.524288]]$, Valor da F.O.: 0.27945967

Passo Nº 2, Ponto Atual = $[-0.02844891]$
 $[0.59360123]]$, Valor da F.O.: 0.14395861

Passo Nº 3, Ponto Atual = $[-0.20633449]$
 $[0.79196597]]$, Valor da F.O.: 0.04304026

Passo Nº 4, Ponto Atual = $[-0.05169763]$
 $[0.80965654]]$, Valor da F.O.: 0.02226128

Passo Nº 5, Ponto Atual = $[-0.10491946]$
 $[0.90138035]]$, Valor da F.O.: 0.01100906

Passo Nº 6, Ponto Atual = $[-0.03557921]$
 $[0.90073593]]$, Valor da F.O.: 0.00540170

Passo Nº 7, Ponto Atual = $[-0.0572743]$
 $[0.95341898]]$, Valor da F.O.: 0.00336286

Passo Nº 8, Ponto Atual = $[-0.01411713]$
 $[0.9474657]]$, Valor da F.O.: 0.00168280

Passo Nº 9, Ponto Atual = $[-0.02975356]$
 $[0.97270778]]$, Valor da F.O.: 0.00088937

Passo Nº 10, Ponto Atual = $[-0.0089673]$
 $[0.97138195]]$, Valor da F.O.: 0.00046814

Passo Nº 11, Ponto Atual = $[-0.01587988]$
 $[0.98428653]]$, Valor da F.O.: 0.00025217

Passo Nº 12, Ponto Atual = $[-0.0054464]$
 $[0.98425967]]$, Valor da F.O.: 0.00013593

Passo Nº 13, Ponto Atual = $[-0.00859628]$
 $[0.9910156]]$, Valor da F.O.: 0.00007408

Passo Nº 14, Ponto Atual = $[-0.00323874]$
 $[0.9912941]]$, Valor da F.O.: 0.00004044

Passo Nº 15, Ponto Atual = $[-0.00469099]$
 $[0.99488049]]$, Valor da F.O.: 0.00002220

Passo Nº 16, Ponto Atual = $[-0.00120648]$
 $[0.99524053]]$, Valor da F.O.: 0.00001408

Passo Nº 17, Ponto Atual = $[-0.00274196]$
 $[0.99756949]]$, Valor da F.O.: 0.00000761

Passo Nº 18, Ponto Atual = $[-0.00074389]$
 $[0.99736784]]$, Valor da F.O.: 0.00000412

Passo Nº 19, Ponto Atual = $[-0.00149313]$
 $[0.99860552]]$, Valor da F.O.: 0.00000224

Passo Nº 20, Ponto Atual = $\begin{bmatrix} -0.00045077 \\ 0.9985416 \end{bmatrix}$, Valor da F.O.: 0.00000122

Passo Nº 21, Ponto Atual = $\begin{bmatrix} -0.00081548 \\ 0.99920203 \end{bmatrix}$, Valor da F.O.: 0.00000067

Passo Nº 22, Ponto Atual = $\begin{bmatrix} -0.0002698 \\ 0.99919077 \end{bmatrix}$, Valor da F.O.: 0.00000036

Passo Nº 23, Ponto Atual = $\begin{bmatrix} -0.00044643 \\ 0.99954432 \end{bmatrix}$, Valor da F.O.: 0.00000020

Passo Nº 24, Ponto Atual = $\begin{bmatrix} -0.00015998 \\ 0.99955044 \end{bmatrix}$, Valor da F.O.: 0.00000011

Passo Nº 25, Ponto Atual = $\begin{bmatrix} -0.00024489 \\ 0.99974023 \end{bmatrix}$, Valor da F.O.: 0.00000006

Passo Nº 26, Ponto Atual = $\begin{bmatrix} -0.00009417 \\ 0.99975 \end{bmatrix}$, Valor da F.O.: 0.00000003

Passo Nº 27, Ponto Atual = $\begin{bmatrix} -0.00014467 \\ 0.99987766 \end{bmatrix}$, Valor da F.O.: 0.00000002

Passo Nº 28, Ponto Atual = $\begin{bmatrix} -0.00003523 \\ 0.99986303 \end{bmatrix}$, Valor da F.O.: 0.00000001

Passo Nº 29, Ponto Atual = $\begin{bmatrix} -0.00007881 \\ 0.9999297 \end{bmatrix}$, Valor da F.O.: 0.00000001

Passo Nº 30, Ponto Atual = $\begin{bmatrix} -0.00002158 \\ 0.99992412 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 31, Ponto Atual = $\begin{bmatrix} -0.00004302 \\ 0.99995971 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 32, Ponto Atual = $\begin{bmatrix} -0.00001304 \\ 0.99995792 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 33, Ponto Atual = $\begin{bmatrix} -0.00002353 \\ 0.99997695 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 34, Ponto Atual = $\begin{bmatrix} -0.0000078 \\ 0.99997664 \end{bmatrix}$, Valor da F.O.: 0.00000000

Finalizado em 34 passos, com Ponto Ótimo = $\begin{bmatrix} -0.0000078 \\ 0.99997664 \end{bmatrix}$ e Valor Ótimo da F.O.: 0.00000000

CPU times: total: 766 ms
Wall time: 776 ms

Out[5]:

```
(matrix( $\begin{bmatrix} -0.0000078 \\ 0.99997664 \end{bmatrix}$ ),  
3.03048029824233e-10,  
34,  
200)
```

Começando Metodo de Newton com Ponto Inicial $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$, onde a F.O. vale 1.00000000

Passo Nº 1, Ponto Atual = $\begin{bmatrix} -0. \\ 1. \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 2, Ponto Atual = $\begin{bmatrix} -0. \\ 1. \end{bmatrix}$, Valor da F.O.: 0.00000000

Finalizado em 2 passos, com Ponto Ótimo = $\begin{bmatrix} -0. \\ 1. \end{bmatrix}$ e Valor Ótimo da F.O.: 0.00000000
Obtidos 0 pontos onde a direção era de subida.

CPU times: total: 46.9 ms
Wall time: 50.9 ms

Out[6]:

```
(matrix( $\begin{bmatrix} -0. \\ 1. \end{bmatrix}$ ),  
      3.08148791101958e-33,  
      2,  
      2)
```

Começando Metodo DFP com Ponto Inicial $[[0]$
 $[0]]$, onde a F.O. vale 1.00000000

Passo Nº 1, Ponto Atual = $[-0.524288]$
 $[0.524288]]$, Valor da F.O.: 0.27945967

Passo Nº 2, Ponto Atual = $[-0.02844891]$
 $[0.59360123]]$, Valor da F.O.: 0.14395861

Passo Nº 3, Ponto Atual = $[-0.20633449]$
 $[0.79196597]]$, Valor da F.O.: 0.04304026

Passo Nº 4, Ponto Atual = $[-0.05169763]$
 $[0.80965654]]$, Valor da F.O.: 0.02226128

Passo Nº 5, Ponto Atual = $[-0.10491946]$
 $[0.90138035]]$, Valor da F.O.: 0.01100906

Passo Nº 6, Ponto Atual = $[-0.03557921]$
 $[0.90073593]]$, Valor da F.O.: 0.00540170

Passo Nº 7, Ponto Atual = $[-0.0572743]$
 $[0.95341898]]$, Valor da F.O.: 0.00336286

Passo Nº 8, Ponto Atual = $[-0.01411713]$
 $[0.9474657]]$, Valor da F.O.: 0.00168280

Passo Nº 9, Ponto Atual = $[-0.02975356]$
 $[0.97270778]]$, Valor da F.O.: 0.00088937

Passo Nº 10, Ponto Atual = $[-0.0089673]$
 $[0.97138195]]$, Valor da F.O.: 0.00046814

Passo Nº 11, Ponto Atual = $[-0.01587988]$
 $[0.98428653]]$, Valor da F.O.: 0.00025217

Passo Nº 12, Ponto Atual = $[-0.0054464]$
 $[0.98425967]]$, Valor da F.O.: 0.00013593

Passo Nº 13, Ponto Atual = $[-0.00859628]$
 $[0.9910156]]$, Valor da F.O.: 0.00007408

Passo Nº 14, Ponto Atual = $[-0.00323874]$
 $[0.9912941]]$, Valor da F.O.: 0.00004044

Passo Nº 15, Ponto Atual = $[-0.00469099]$
 $[0.99488049]]$, Valor da F.O.: 0.00002220

Passo Nº 16, Ponto Atual = $[-0.00120648]$
 $[0.99524053]]$, Valor da F.O.: 0.00001408

Passo Nº 17, Ponto Atual = $[-0.00274196]$
 $[0.99756949]]$, Valor da F.O.: 0.00000761

Passo Nº 18, Ponto Atual = $[-0.00074389]$
 $[0.99736784]]$, Valor da F.O.: 0.00000412

Passo Nº 19, Ponto Atual = $[-0.00149313]$
 $[0.99860552]]$, Valor da F.O.: 0.00000224

Passo Nº 20, Ponto Atual = $\begin{bmatrix} -0.00045077 \\ 0.9985416 \end{bmatrix}$, Valor da F.O.: 0.00000122

Passo Nº 21, Ponto Atual = $\begin{bmatrix} -0.00081548 \\ 0.99920203 \end{bmatrix}$, Valor da F.O.: 0.00000067

Passo Nº 22, Ponto Atual = $\begin{bmatrix} -0.0002698 \\ 0.99919077 \end{bmatrix}$, Valor da F.O.: 0.00000036

Passo Nº 23, Ponto Atual = $\begin{bmatrix} -0.00044643 \\ 0.99954432 \end{bmatrix}$, Valor da F.O.: 0.00000020

Passo Nº 24, Ponto Atual = $\begin{bmatrix} -0.00015998 \\ 0.99955044 \end{bmatrix}$, Valor da F.O.: 0.00000011

Passo Nº 25, Ponto Atual = $\begin{bmatrix} -0.00024489 \\ 0.99974023 \end{bmatrix}$, Valor da F.O.: 0.00000006

Passo Nº 26, Ponto Atual = $\begin{bmatrix} -0.00009417 \\ 0.99975 \end{bmatrix}$, Valor da F.O.: 0.00000003

Passo Nº 27, Ponto Atual = $\begin{bmatrix} -0.00014467 \\ 0.99987766 \end{bmatrix}$, Valor da F.O.: 0.00000002

Passo Nº 28, Ponto Atual = $\begin{bmatrix} -0.00003523 \\ 0.99986303 \end{bmatrix}$, Valor da F.O.: 0.00000001

Passo Nº 29, Ponto Atual = $\begin{bmatrix} -0.00007881 \\ 0.9999297 \end{bmatrix}$, Valor da F.O.: 0.00000001

Passo Nº 30, Ponto Atual = $\begin{bmatrix} -0.00002158 \\ 0.99992412 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 31, Ponto Atual = $\begin{bmatrix} -0.00004302 \\ 0.99995971 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 32, Ponto Atual = $\begin{bmatrix} -0.00001304 \\ 0.99995792 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 33, Ponto Atual = $\begin{bmatrix} -0.00002353 \\ 0.99997695 \end{bmatrix}$, Valor da F.O.: 0.00000000

Passo Nº 34, Ponto Atual = $\begin{bmatrix} -0.0000078 \\ 0.99997664 \end{bmatrix}$, Valor da F.O.: 0.00000000

Finalizado em 34 passos, com Ponto Ótimo = $\begin{bmatrix} -0.0000078 \\ 0.99997664 \end{bmatrix}$ e Valor Ótimo da F.O.: 0.00000000

CPU times: total: 703 ms
Wall time: 699 ms

Out[7]:

```
(matrix( $\begin{bmatrix} -0.0000078 \\ 0.99997664 \end{bmatrix}$ ),  
 3.03048029823905e-10,  
 34,  
 200)
```

Começando Metodo BFGS com Ponto Inicial $[[0]$
 $[0]]$, onde a F.O. vale 1.00000000

Passo Nº 1, Ponto Atual = $[-0.524288]$
 $[0.524288]]$, Valor da F.O.: 0.27945967

Passo Nº 2, Ponto Atual = $[-0.06785894]$
 $[0.63949637]]$, Valor da F.O.: 0.09156843

Passo Nº 3, Ponto Atual = $[-0.18600041]$
 $[0.85072927]]$, Valor da F.O.: 0.03501460

Passo Nº 4, Ponto Atual = $[-0.00803726]$
 $[0.88225844]]$, Valor da F.O.: 0.01210670

Passo Nº 5, Ponto Atual = $[-0.05903979]$
 $[0.94164859]]$, Valor da F.O.: 0.00348674

Passo Nº 6, Ponto Atual = $[-0.00539725]$
 $[0.95789828]]$, Valor da F.O.: 0.00137742

Passo Nº 7, Ponto Atual = $[-0.02052607]$
 $[0.9783388]]$, Valor da F.O.: 0.00042313

Passo Nº 8, Ponto Atual = $[-0.00246741]$
 $[0.98468968]]$, Valor da F.O.: 0.00017111

Passo Nº 9, Ponto Atual = $[-0.0085826]$
 $[0.99377416]]$, Valor da F.O.: 0.00007904

Passo Nº 10, Ponto Atual = $[-0.00105134]$
 $[0.99438333]]$, Valor da F.O.: 0.00002195

Passo Nº 11, Ponto Atual = $[-0.00307001]$
 $[0.9976571]]$, Valor da F.O.: 0.00000995

Passo Nº 12, Ponto Atual = $[-0.00042873]$
 $[0.99793247]]$, Valor da F.O.: 0.00000287

Passo Nº 13, Ponto Atual = $[-0.0011069]$
 $[0.99912099]]$, Valor da F.O.: 0.00000128

Passo Nº 14, Ponto Atual = $[-0.00017157]$
 $[0.99923779]]$, Valor da F.O.: 0.00000038

Passo Nº 15, Ponto Atual = $[-0.00040039]$
 $[0.99967064]]$, Valor da F.O.: 0.00000017

Passo Nº 16, Ponto Atual = $[-0.00006792]$
 $[0.99971881]]$, Valor da F.O.: 0.00000005

Passo Nº 17, Ponto Atual = $[-0.00014506]$
 $[0.99987669]]$, Valor da F.O.: 0.00000002

Passo Nº 18, Ponto Atual = $[-0.00002668]$
 $[0.99989622]]$, Valor da F.O.: 0.00000001

Passo Nº 19, Ponto Atual = $[-0.00005261]$
 $[0.99995386]]$, Valor da F.O.: 0.00000000


```
Passo Nº 20, Ponto Atual = [[-0.00001042]
[ 0.99996169]], Valor da F.O.: 0.00000000
```

```
Passo Nº 21, Ponto Atual = [[-0.0000191 ]
[ 0.99998274]], Valor da F.O.: 0.00000000
```

```
Passo Nº 22, Ponto Atual = [[-0.00000029]
[ 0.99998663]], Valor da F.O.: 0.00000000
```

```
Passo Nº 23, Ponto Atual = [[-0.00000694]
[ 0.99999355]], Valor da F.O.: 0.00000000
```

```
Passo Nº 24, Ponto Atual = [[-0.00000022]
[ 0.99999508]], Valor da F.O.: 0.00000000
```

```
Finalizado em 24 passos, com Ponto Ótimo = [[-0.00000022]
[ 0.99999508]] e Valor Ótimo da F.O.: 0.00000000
```

```
CPU times: total: 625 ms
Wall time: 610 ms
```

Out[8]:

```
(matrix([[ -0.00000022],
[ 0.99999508]]),
2.21043830616147e-11,
24,
153)
```

2.2. Ponto $P_2(x, y) = (0, 2)$

O ponto $(0, 2)$ falha no método de Newton pois a matriz hessiana da função nesse ponto tem determinante zero, ou seja, é singular; portanto, não é possível obter o inverso da hessiana, parte fundamental do método.

Os outros métodos não tem esse problema, pois não se utilizam da hessiana, e convergiram rapidamente. O Método BFGS foi levemente mais rápido que os outros.

Com exceção do Método de Newton, os outros métodos foram capazes de convergir para o ponto ótimo $(x^*, y^*) = (0, 1)$.

Método do Gradiente

```
Ponto Inicial: (0, 2)
Valor Inicial da FO: 1.00000000
```

```
Ponto Ótimo: [[0.00000414]
[1.00001226]]
Valor da FO: 0.00000000
Nº de Passos: 40
Nº chamadas Armijo: 244
```

```
CPU times: total: 1.03 s
Wall time: 1.02 s
```

Começando Metodo de Newton com Ponto Inicial $\begin{bmatrix} 0 \\ 2 \end{bmatrix}$, onde a F.O. vale 1.00000000

Erro: A matriz Hessiana não é inversível.

```
-----  
LinAlgError                                Traceback (most recent call last)  
File <timed exec>:2, in <module>  
  
File ~\UFRJ 2022.2\Otimização\nlp_irrestrito.py:121, in OtimizacaoNLP.metodo  
_newton(self, verbose)  
    119 m_gradiente = np.matrix(self.gradiente_p(p_atual))  
    120 try:  
--> 121     direcao = - np.linalg.inv(m_hessiana)*m_gradiente # Passo de Ne  
wton  
    122 except np.linalg.LinAlgError:  
    123     if verbose:  
  
File <__array_function__ internals>:5, in inv(*args, **kwargs)  
  
File ~\anaconda3\lib\site-packages\numpy.linalg.linalg.py:545, in inv(a)  
    543 signature = 'D->D' if isComplexType(t) else 'd->d'  
    544 extobj = get_linalg_error_extobj(_raise_linalgerror_singular)  
--> 545 ainv = _umath_linalg.inv(a, signature=signature, extobj=extobj)  
    546 return wrap(ainv.astype(result_t, copy=False))  
  
File ~\anaconda3\lib\site-packages\numpy.linalg.linalg.py:88, in _raise_lina  
lgerror_singular(err, flag)  
    87 def _raise_linalgerror_singular(err, flag):  
--> 88     raise LinAlgError("Singular matrix")  
  
LinAlgError: Singular matrix
```

Método DFP

Ponto Inicial: (0, 2)

Valor Inicial da FO: 1.00000000

Ponto Ótimo: $\begin{bmatrix} 0.00000414 \\ 1.00001226 \end{bmatrix}$

Valor da FO: 0.00000000

Nº de Passos: 40

Nº chamadas Armijo: 244

CPU times: total: 1.14 s

Wall time: 1.15 s

Método BFGS

Ponto Inicial: (0, 2)

Valor Inicial da FO: 1.00000000

Ponto Ótimo: [[0.00000029]

[1.00000128]]

Valor da FO: 0.00000000

Nº de Passos: 28

Nº chamadas Armijo: 186

CPU times: total: 812 ms

Wall time: 832 ms

2.3. Ponto $P_3(x, y) = (4, 10)$

No ponto (4, 10), observa-se que o Método do Gradiente é extremamente lento. O valor da Função Objetivo decresce, mas muito devagar, extrapolando o número de passos permitindo antes de atingir a convergência no ponto mínimo. Isso acontece porque os autovalores da matriz hessiana $\nabla^2 f(x, y)$ nesse ponto são tais que $\lambda_1 \ll \lambda_2$, situação em que as curvas de nível de $f(x, y)$ são elipsóides excêntricos, ou seja, curvas muito achatadas, o que faz com que a convergência ocorra de forma muito lenta. De fato, $\lambda_1 \approx 0.89$ e $\lambda_2 \approx 10835.00$. [2] (https://www.wolframalpha.com/input?i=%282-l%29%282+%2B+4e%5E%282x%29+-+2y%28e%5Ex%29+-+l%29+-+4*e%5E%282x%29+%3D+0%2C+x+%3D+4%2C+y+%3D+10%29).

O mesmo problema acontece com os Métodos DFP e BFGS, que também sofrem de convergência muito lenta e extrapolam o número máximo de passos.

O Método de Newton, por outro lado, tem um ótimo desempenho, convergindo para o ponto ótimo rapidamente.

Apenas o Método de Newton foi capaz de convergir para o ponto ótimo $(x^*, y^*) = (0, 1)$.

Método do Gradiente

Ponto Inicial: (4, 10)

Valor Inicial da FO: 2004.99498638

Ponto Ótimo: [[1.67470292]

[5.71446538]]

Valor da FO: 2.94695199

Nº de Passos: 500

Nº chamadas Armijo: 9840

CPU times: total: 40.9 s

Wall time: 41.1 s

Método de Newton

Ponto Inicial: (4, 10)
Valor Inicial da F0: 2004.99498638

Ponto Ótimo: [[0.]
[1.]]
Valor da F0: 0.00000000
Nº de Passos: 18
Nº chamadas Armijo: 45

CPU times: total: 438 ms
Wall time: 459 ms

Método DFP

Ponto Inicial: (4, 10)
Valor Inicial da F0: 2004.99498638

Ponto Ótimo: [[1.67470292]
[5.71446538]]
Valor da F0: 2.94695199
Nº de Passos: 500
Nº chamadas Armijo: 9840

CPU times: total: 39.6 s
Wall time: 39.7 s

Método BFGS

Ponto Inicial: (4, 10)
Valor Inicial da F0: 2004.99498638

Ponto Ótimo: [[2.07547003]
[8.27880296]]
Valor da F0: 4.40399357
Nº de Passos: 500
Nº chamadas Armijo: 13357

CPU times: total: 53.7 s
Wall time: 54.6 s

2.4. Ponto $P_4(x, y) = (7, 4)$

O ponto (7, 4) está localizado onde a função $f(x, y)$ dispara, onde x é muito positivo. Nesse ponto, a função objetivo vale mais de 1×10^6 e está bem longe do ótimo.

Assim, apesar do Método de Newton ter tido o melhor desempenho com relação a tempo de execução na CPU, nota-se que foram necessários bem mais passos do que antes para a convergência.

O Método BFGS se saiu consideravelmente pior que os outros, precisando de mais de 300 passos até a convergência e demorando para convergir, visto o número muito grande de chamadas de Armijo.

Apesar disso, todos os métodos foram capazes de convergir para o ponto ótimo $(x^*, y^*) = (0, 1)$.

Método do Gradiente

Ponto Inicial: (7, 4)

Valor Inicial da FO: 1193896.21889735

Ponto Ótimo: [[0.00000167]
[1.00000498]]

Valor da FO: 0.00000000

Nº de Passos: 78

Nº chamadas Armijo: 753

CPU times: total: 2.84 s

Wall time: 2.89 s

Método de Newton

Ponto Inicial: (7, 4)

Valor Inicial da FO: 1193896.21889735

Ponto Ótimo: [[0.]
[1.]]

Valor da FO: 0.00000000

Nº de Passos: 85

Nº chamadas Armijo: 243

CPU times: total: 2 s

Wall time: 2.01 s

Método DFP

Ponto Inicial: (7, 4)

Valor Inicial da FO: 1193896.21889735

Ponto Ótimo: [[0.00000167]
[1.00000498]]

Valor da FO: 0.00000000

Nº de Passos: 78

Nº chamadas Armijo: 753

CPU times: total: 2.73 s

Wall time: 2.74 s

Método BFGS

Ponto Inicial: (7, 4)

Valor Inicial da FO: 1193896.21889735

Ponto Ótimo: [[0.00000007]
[0.99999948]]

Valor da FO: 0.00000000

Nº de Passos: 371

Nº chamadas Armijo: 8499

CPU times: total: 25.9 s

Wall time: 26.1 s

2.5. Ponto $P_5(x, y) = (9, -2)$

O ponto $(9, -2)$ está localizado onde a função $f(x, y)$ está crescendo ainda mais rapidamente que antes. Nesse ponto, a função objetivo vale mais de 65×10^6 e está muito longe do ótimo. Isso agravou a situação vista no ponto anterior.

Assim, o Método de Newton não se sai tão bem, precisando de mais de 270 passos para a convergência.

O Método BFGS teve um desempenho bem ruim, estourando o número máximo de passos e conseguindo convergir para um ponto próximo, mas não ótimo. Cada passo teve um alto custo de tempo devido a muitas chamadas de Armijo.

Já o Método do Gradiente e o Método DFP se saem muito bem, convergindo em poucos passos para o padrão desses métodos.

Além do Método BFGS, os outros métodos foram capazes de convergir para o ponto ótimo $(x^*, y^*) = (0, 1)$. O Método BFGS conseguiu um ponto próximo ao ótimo, mas não ideal.

Método do Gradiente

Ponto Inicial: $(9, -2)$

Valor Inicial da FO: 65692466.47304080

Ponto Ótimo: $\begin{bmatrix} -0.00000587 \\ 0.99998236 \end{bmatrix}$

Valor da FO: 0.00000000

Nº de Passos: 41

Nº chamadas Armijo: 466

CPU times: total: 1.41 s

Wall time: 1.4 s

Método de Newton

Ponto Inicial: $(9, -2)$

Valor Inicial da FO: 65692466.47304080

Ponto Ótimo: $\begin{bmatrix} 0. \\ 1. \end{bmatrix}$

Valor da FO: 0.00000000

Nº de Passos: 278

Nº chamadas Armijo: 814

CPU times: total: 6.62 s

Wall time: 6.66 s

Método DFP

Ponto Inicial: (9, -2)
Valor Inicial da FO: 65692466.47304080

Ponto Ótimo: $\begin{bmatrix} -0.00000587 \\ 0.99998236 \end{bmatrix}$
Valor da FO: 0.00000000
Nº de Passos: 41
Nº chamadas Armijo: 466

CPU times: total: 1.33 s
Wall time: 1.32 s

Método BFGS

Ponto Inicial: (9, -2)
Valor Inicial da FO: 65692466.47304080

Ponto Ótimo: $\begin{bmatrix} 0.01104176 \\ 1.34986751 \end{bmatrix}$
Valor da FO: 0.11488335
Nº de Passos: 500
Nº chamadas Armijo: 18693

CPU times: total: 56.1 s
Wall time: 57.1 s

2.6. Ponto extra $P_*(x, y) = (30, -40)$

O ponto $(30, -40)$ falha no método de Newton, convergindo para um ponto muito distante do ótimo. Observa-se que, ao fim do primeiro passo do método, obtém-se aproximadamente o ponto $(30, 1.068 \times 10^{13})$. Nesse ponto, temos que $\nabla f(x, y)^T d > 0$, o que significa que a direção é de subida, e portanto o método falha.

O Método do Gradiente, no entanto, consegue convergir normalmente.

Obs: Os Métodos BFGS e DFP não funcionaram, pois os valores obtidos durante os cálculos extrapolaram os limites que os vetores numpy conseguem suportar.

Método do Gradiente

Ponto Inicial: (30, -40)
Valor Inicial da FO: 11420073898242300000000000.00000000

Ponto Ótimo: $\begin{bmatrix} -0.0000008 \\ 0.99999743 \end{bmatrix}$
Valor da FO: 0.00000000
Nº de Passos: 58
Nº chamadas Armijo: 2632

CPU times: total: 7.47 s
Wall time: 7.67 s

Começando Metodo de Newton com Ponto Inicial $\begin{bmatrix} 30 \\ -40 \end{bmatrix}$, onde a F.O. vale 11420073898242300000000000.00000000

Passo Nº 1, Ponto Atual = $\begin{bmatrix} 3.00000000e+01 \\ 1.06864746e+13 \end{bmatrix}$, Valor da F.O.: 900.00000563

Warning: A direção de deslocamento é de subida.

Passo Nº 2, Ponto Atual = $\begin{bmatrix} 3.00000035e+01 \\ 1.06865124e+13 \end{bmatrix}$, Valor da F.O.: 5397.52528433

Finalizado em 2 passos, com Ponto Ótimo = $\begin{bmatrix} 3.00000035e+01 \\ 1.06865124e+13 \end{bmatrix}$ e Valor Ótimo da F.O.: 5397.52528433
Obtidos 1 pontos onde a direção era de subida.

Out[31]:

```
(matrix( $\begin{bmatrix} 3.00000035e+01, \\ 1.06865124e+13 \end{bmatrix}$ ),  
5397.52528433472,  
2,  
46)
```

3. Tabela com os Resultados

Abaixo estão as tabelas com os resultados obtidos considerando os pontos iniciais estudados para cada método.

Out[33]:

Resultados Computacionais pelo Método de Newton							
	Ponto Inicial	Nº de Passos	Nº Ch. Armijo	Ponto Ótimo	Valor Ótimo	Erro de Aproximação	Tempo de CPU
0	[0, 0]	2	2	$\begin{bmatrix} -0.000000, \\ 1.000000 \end{bmatrix}$	0.000000	3.081488E-33	0.031250
1	[0, 2]	NA	NA	NA	NA	NA	NA
2	[4, 10]	18	45	$\begin{bmatrix} 0.000000, \\ 1.000000 \end{bmatrix}$	0.000000	1.593902E-22	0.421875
3	[7, 4]	85	243	$\begin{bmatrix} 0.000000, \\ 1.000000 \end{bmatrix}$	0.000000	2.272360E-22	2.187500
4	[9, -2]	278	814	$\begin{bmatrix} 0.000000, \\ 1.000000 \end{bmatrix}$	0.000000	6.474633E-25	7.218750

Out[34]:

Resultados Computacionais pelo Método do Gradiente

	Ponto Inicial	Nº de Passos	Nº Ch. Armijo	Ponto Ótimo	Valor Ótimo	Erro de Aproximação	Tempo de CPU
0	[0, 0]	34	200	[-0.000008, 0.999977]	0.000000	3.030480E-10	0.656250
1	[0, 2]	40	244	[0.000004, 1.000012]	0.000000	8.306732E-11	0.953125
2	[4, 10]	500	9840	[1.674703, 5.714465]	2.946952	2.946952E+0	40.296875
3	[7, 4]	78	753	[0.000002, 1.000005]	0.000000	1.376175E-11	2.984375
4	[9, -2]	41	466	[-0.000006, 0.999982]	0.000000	1.729456E-10	1.281250

Out[35]:

Resultados Computacionais pelo Método DFP

	Ponto Inicial	Nº de Passos	Nº Ch. Armijo	Ponto Ótimo	Valor Ótimo	Erro de Aproximação	Tempo de CPU
0	[0, 0]	34	200	[-0.000008, 0.999977]	0.000000	3.030480E-10	0.750000
1	[0, 2]	40	244	[0.000004, 1.000012]	0.000000	8.306732E-11	1.000000
2	[4, 10]	500	9840	[1.674703, 5.714465]	2.946952	2.946952E+0	41.281250
3	[7, 4]	78	753	[0.000002, 1.000005]	0.000000	1.376175E-11	3.390625
4	[9, -2]	41	466	[-0.000006, 0.999982]	0.000000	1.729457E-10	1.531250

Out[36]:

Resultados Computacionais pelo Método BFGS

	Ponto Inicial	Nº de Passos	Nº Ch. Armijo	Ponto Ótimo	Valor Ótimo	Erro de Aproximação	Tempo de CPU
0	[0, 0]	24	153	[-0.000000, 0.999995]	0.000000	2.210438E-11	0.593750
1	[0, 2]	28	186	[0.000000, 1.000001]	0.000000	1.055432E-12	0.703125
2	[4, 10]	500	13357	[2.075470, 8.278803]	4.403994	4.403994E+0	53.734375
3	[7, 4]	371	8499	[0.000000, 0.999999]	0.000000	3.548834E-13	26.015625
4	[9, -2]	500	18693	[0.011042, 1.349868]	0.114883	1.148834E-1	54.453125

4. Conclusões

Com isso, é possível concluir que:

1. O Método de Newton funciona de forma excelente para pontos iniciais próximos ao ponto ótimo. No entanto, para pontos muito distantes do ótimo seu tempo de convergência pode aumentar consideravelmente.
2. O Método de Newton possui limitações: ele não funciona quando a matriz hessiana no ponto estudado é singular; além disso, ele não garante que a direção será de descida. Em ambos os casos, o método falha.
3. O Método do Gradiente fica extremamente lento quando os autovalores λ_1 e λ_2 da hessiana estão muito distantes entre si.
4. O Método BFGS parece não ser muito eficiente em regiões onde a função objetivo está crescendo muito rápido, tendo um desempenho muito pior que o do Método de Newton nessas situações. Nesses casos, ele faz muitas chamadas de Armijo a cada iteração, o que torna o passo t_k muito pequeno e a convergência bem devagar. Esse número alto de chamadas de Armijo é um fator que também aumenta o tempo de execução.
5. Mesmo em situações não ideais, o Método de Newton parece não precisar de muitas chamadas de Armijo, o que, apesar do fato da operação de inversão de matriz que faz parte do método ser custosa, torna as iterações do Método de Newton na prática visivelmente mais rápidas que as dos outros métodos grande parte das vezes.
6. O Método DFP teve um desempenho extremamente semelhante ao Método do Gradiente em todos os pontos.