

Nathan Martins

Métodos Numéricos Computacionais

09 November 2017

Relatório de Implementações

MÉTODOS IMPLEMENTADOS

- Euler Simples
- Euler Aprimorado
- Euler Inverso
- Runge-Kutta
- Adams-Bashforth (Ordens 1 a 6)
- Adams-Moulton (Ordens 1 a 6)

ALGORITMO BASE DE TODOS OS MÉTODOS

Todos as implementações seguem os seguintes passos:

1. Recebem a função $f(t, y)$
2. Recebem os pontos (t_i, y_i)
3. Recebem o tamanho do passo h
4. Recebem o número de passos n
5. Faz n repetições de um algoritmo particular
6. Abrem um plot da função e imprime os pontos no terminal

ALGORITMO PARTICULAR DE CADA MÉTODO

Para os exemplos a seguir a seguinte configuração será utilizada:

$$y(t) = 1 - t + 4y$$

$$y(0) = 1$$

$$h = .05$$

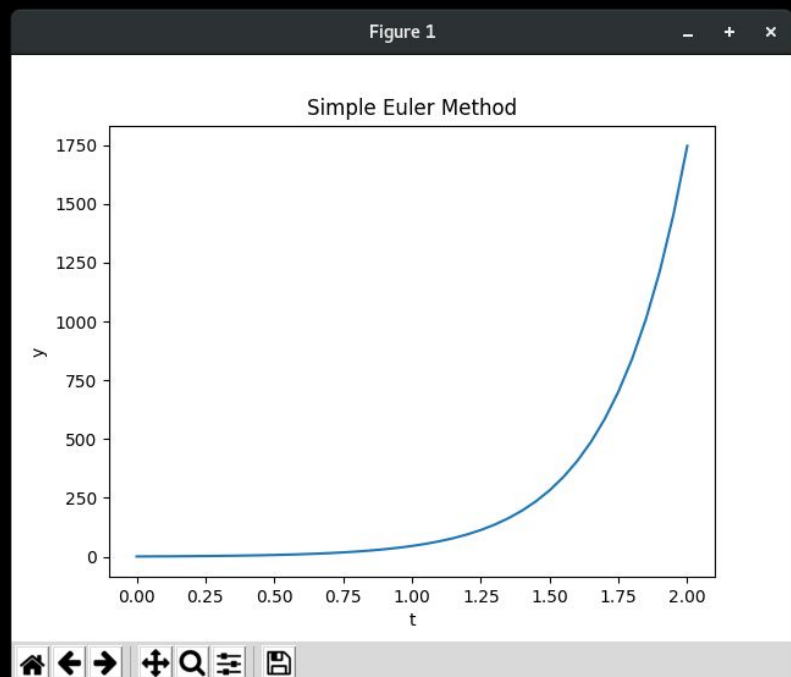
$$n = 40$$

Euler Simples

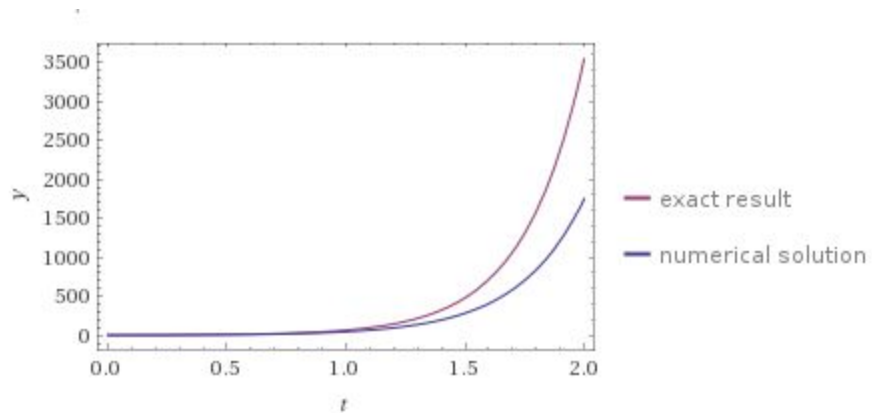
1. $k_1 = f(t, y)$
2. $y = y + h * k_1$
3. $t = t + h$

Exemplo:

```
Running Simple Euler Method
y(0.0500) = 1.2500000
y(0.1000) = 1.5475000
y(0.1500) = 1.9020000
y(0.2000) = 2.3249000
y(0.2500) = 2.8298800
y(0.3000) = 3.4333560
y(0.3500) = 4.1550272
y(0.4000) = 5.0185326
y(0.4500) = 6.0522392
y(0.5000) = 7.2901870
y(0.5500) = 8.7732244
y(0.6000) = 10.5503693
y(0.6500) = 12.6804431
y(0.7000) = 15.2340318
y(0.7500) = 18.2958381
y(0.8000) = 21.9675057
y(0.8500) = 26.3710069
y(0.9000) = 31.6527083
y(0.9500) = 37.9882499
y(1.0000) = 45.5883999
y(1.0500) = 54.7060799
y(1.1000) = 65.6447959
y(1.1500) = 78.7687550
y(1.2000) = 94.5150061
y(1.2500) = 113.4080073
y(1.3000) = 136.0771087
y(1.3500) = 163.2775305
y(1.4000) = 195.9155366
y(1.4500) = 235.0786439
y(1.5000) = 282.0718726
y(1.5500) = 338.4612472
y(1.6000) = 406.1259966
y(1.6500) = 487.3211959
y(1.7000) = 584.7529351
y(1.7500) = 701.6685221
y(1.8000) = 841.9647265
y(1.8500) = 1010.3176719
y(1.9000) = 1212.3387062
y(1.9500) = 1454.7614475
y(2.0000) = 1745.6662370
```



A título de comparação: [Fonte \(Wolfram Alpha\)](#)



Euler Inverso

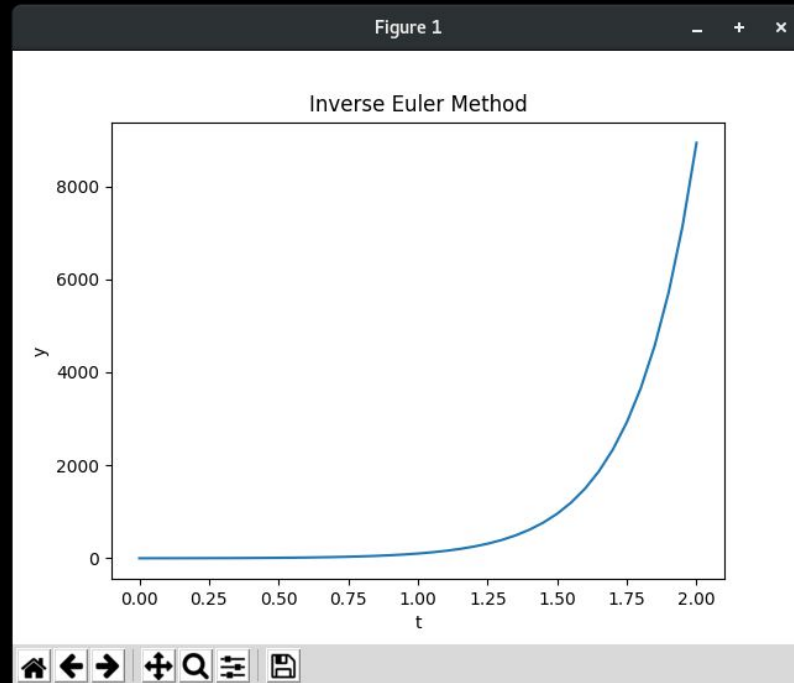
1. $y_{n+1} = y_n + h * f(t_{n+1}, y_{n+1})$
2. $t = t + h$

Running Inverse Euler Method

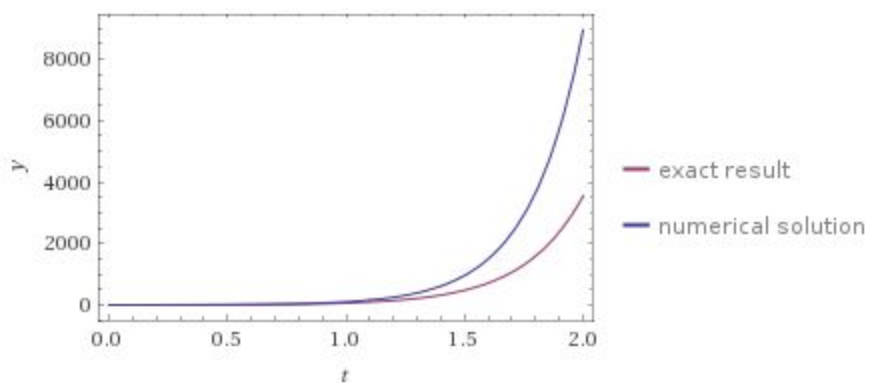
```

y(0.0500) = 1.3093750
y(0.1000) = 1.6929688
y(0.1500) = 2.1693359
y(0.2000) = 2.7616699
y(0.2500) = 3.4989624
y(0.3000) = 4.4174530
y(0.3500) = 5.5624413
y(0.4000) = 6.9905516
y(0.4500) = 8.7725645
y(0.5000) = 10.9969556
y(0.5500) = 13.7743195
y(0.6000) = 17.2428993
y(0.6500) = 21.5754992
y(0.7000) = 26.9881240
y(0.7500) = 33.7507799
y(0.8000) = 42.2009749
y(0.8500) = 52.7605937
y(0.9000) = 65.9569921
y(0.9500) = 82.4493651
y(1.0000) = 103.0617064
y(1.0500) = 128.8240080
y(1.1000) = 161.0237600
y(1.1500) = 201.2703250
y(1.2000) = 251.5754062
y(1.2500) = 314.4536328
y(1.3000) = 393.0482910
y(1.3500) = 491.2884887
y(1.4000) = 614.0856109
y(1.4500) = 767.5788886
y(1.5000) = 959.4423607
y(1.5500) = 1199.2685759
y(1.6000) = 1499.0482199
y(1.6500) = 1873.7696499
y(1.7000) = 2342.1683124
y(1.7500) = 2927.6635155
y(1.8000) = 3659.5293943
y(1.8500) = 4574.3586179
y(1.9000) = 5717.8920224
y(1.9500) = 7147.3056530
y(2.0000) = 8934.0695662

```



Comparando: [Fonte](#)



Euler Aprimorado (Heun)

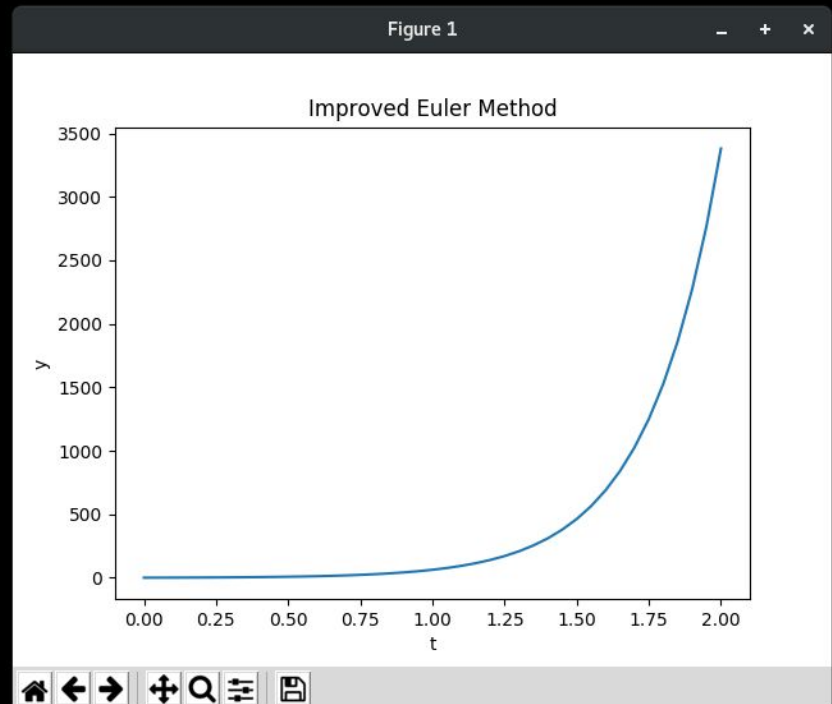
1. $y_{n+1} = y_n + \frac{f_n + f(t_n + h, y_n + hf_n)}{2}h$
2. $t = t + h$

Running Improved Euler Method

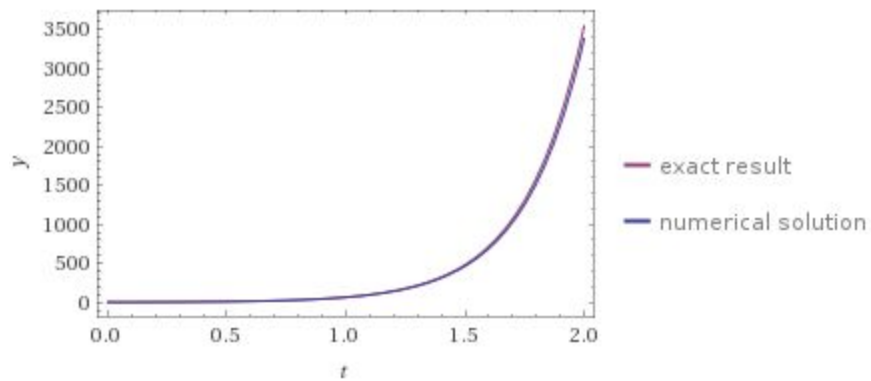
```

y(0.0500) = 1.2737500
y(0.1000) = 1.6049750
y(0.1500) = 2.0063195
y(0.2000) = 2.4932098
y(0.2500) = 3.0844659
y(0.3000) = 3.8030485
y(0.3500) = 4.6769691
y(0.4000) = 5.7404023
y(0.4500) = 7.0350408
y(0.5000) = 8.6117498
y(0.5500) = 10.5325848
y(0.6000) = 12.8732534
y(0.6500) = 15.7261192
y(0.7000) = 19.2038654
y(0.7500) = 23.4439658
y(0.8000) = 28.6141382
y(0.8500) = 34.9189986
y(0.9000) = 42.6081783
y(0.9500) = 51.9862276
y(1.0000) = 63.4246976
y(1.0500) = 77.3768811
y(1.1000) = 94.3957950
y(1.1500) = 115.1561199
y(1.2000) = 140.4809662
y(1.2500) = 171.3745288
y(1.3000) = 209.0619251
y(1.3500) = 255.0377987
y(1.4000) = 311.1256144
y(1.4500) = 379.5499995
y(1.5000) = 463.0249994
y(1.5500) = 564.8617493
y(1.6000) = 689.0998341
y(1.6500) = 840.6675476
y(1.7000) = 1025.5774081
y(1.7500) = 1251.1646879
y(1.8000) = 1526.3784192
y(1.8500) = 1862.1364215
y(1.9000) = 2271.7584342
y(1.9500) = 2771.4945397
y(2.0000) = 3381.1698384

```



Comparação: [Fonte](#)



Runge-Kutta

$$k_1 = f(t_n, y_n),$$

$$k_2 = f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1\right),$$

$$k_3 = f\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2\right),$$

$$k_4 = f(t_n + h, y_n + hk_3).$$

$$y_{n+1} = y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4),$$

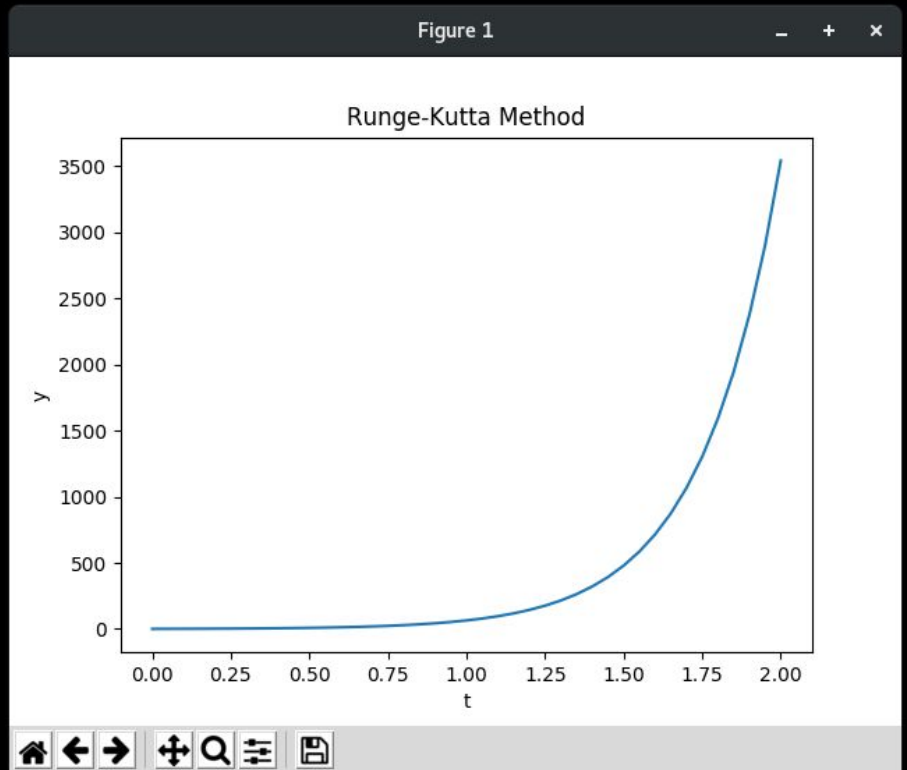
$$t_{n+1} = t_n + h$$

Exemplo:

```

Running Runge-Kutta
y(0.0500) = 1.2754125
y(0.1000) = 1.6090338
y(0.1500) = 2.0137514
y(0.2000) = 2.5053060
y(0.2500) = 3.1029232
y(0.3000) = 3.8300854
y(0.3500) = 4.7154738
y(0.4000) = 5.7941197
y(0.4500) = 7.1088104
y(0.5000) = 8.7118060
y(0.5500) = 10.6669373
y(0.6000) = 13.0521672
y(0.6500) = 15.9627196
y(0.7000) = 19.5149007
y(0.7500) = 23.8507672
y(0.8000) = 29.1438270
y(0.8500) = 35.6060029
y(0.9000) = 43.4961369
y(0.9500) = 53.1303791
y(1.0000) = 64.8948750
y(1.0500) = 79.2612628
y(1.1000) = 96.8056014
y(1.1500) = 118.2314891
y(1.2000) = 144.3983008
y(1.2500) = 176.3556771
y(1.3000) = 215.3856490
y(1.3500) = 263.0540892
y(1.4000) = 321.2735545
y(1.4500) = 392.3800420
y(1.5000) = 479.2267383
y(1.5500) = 585.2985256
y(1.6000) = 714.8518392
y(1.6500) = 873.0854889
y(1.7000) = 1066.3493012
y(1.7500) = 1302.3989539
y(1.8000) = 1590.7072323
y(1.8500) = 1942.8441961
y(1.9000) = 2372.9415161
y(1.9500) = 2898.2596152
y(2.0000) = 3539.8803741

```



Comparação: [Fonte](#)

