

# ニュートン・ラフソン法による 1 変数方程式の解

青木繁伸

## 1 目的

ニュートン・ラフソン法により一変数関数  $f(x) = 0$  の解を求める。

## 2 使用法

```
from newton import newton
newton(fun, x1, delta=1e-5, epsilon=1e-14, max_rotation=100)
```

### 2.1 引数

fun	関数定義
x1	初期値
delta	数値微分するときの微小数値 (デフォルトは 1e-5)
epsilon	許容誤差 (デフォルトは 1e-14)
max_rotation	収束計算の上限回数 (デフォルトは 100 回)

### 2.2 戻り値の名前

求まった解もしくは求まらなかったときにはその由を表す文字列

## 3 使用例

```
import scipy as sp

def f1(x):
    return sp.sin(x)/x

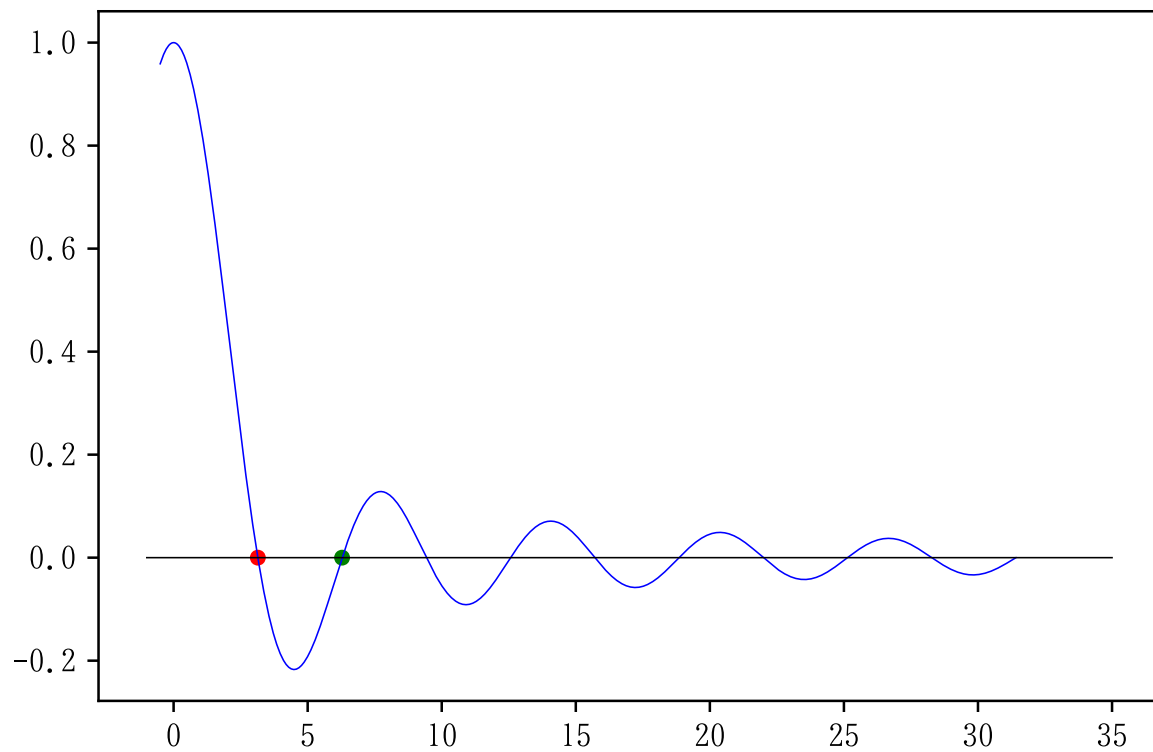
import matplotlib.pyplot as plt
x = sp.linspace(-0.5, 10*sp.pi, 1000)
y = f1(x)
plt.plot(x, y, linewidth=0.5, color="blue")
plt.plot([-1, 35], [0, 0], linewidth=0.5, color="black")

import sys
sys.path.append("statlib")
from newton import newton

a = newton(f1, 1)
```

```
plt.scatter(a, 0, s=16, c="red")  
  
a = newton(f1, 5)  
plt.scatter(a, 0, s=16, c="green")
```

<matplotlib.collections.PathCollection at 0x10fdeba20>



```
def f2(x):  
    return x**3-2  
  
a = newton(f2, 1)  
print(a, f2(a))
```

1.2599210498948732 0.0