HW1

September 20, 2022

1 Question 1: The Secant Method to Solve a Scalar Equation

The Newton's method is an fast method to solve algebraic equations in the form $f(x) = 0$. However, it requirs the function $f(x)$ to be differentiable. So, for functions such as $f(x) = |x - e^{-x}| - 1$, or functions which dirivative is difficult to calculate, the method cannot be applied. The Secant method can be used instead, as it does not require the derivative information.

Unlike the Newton's method using the derivative at an initial guess x_0 to approximate the function $f(x)$, the secant method requires two initial guesses $x_0 \neq x_1$, then uses the secant line passing through $(x_0, f(x_0))$ and $(x_1, f(x_1))$ to approximate the function $f(x)$. That is, the zero of the secant line, denoted as x_2, is an improved approximation to the root of $f(x)$. The Secant line is

$$y - f(x_0) = \frac{f(x_1) - f(x_0)}{x_1 - x_0}(x - x_0),$$

The zero is thus (by letting $y = 0$)

$$x_2 = x_0 - \frac{f(x_0)}{f(x_1) - f(x_0)}(x_1 - x_0)$$

Like the Newton's method, this step is repeated by using x_1 and x_2 as the initial guesses and get x_3, etc. We may use one of the following stopping criterions to stop the iterative process. * $|x_1 - x_1| < \varepsilon$ for some predetermined error tolerance ε * $|f(x_2)| < \varepsilon$ for some predetermined error tolerance ε.

1. Write an equation solver that solves $f(x) = 0$ using the zSecant method. The solver is a function named **secant**, taking the following four arguments

- **f**, which is a function to solve
- **x0**, the first initial guess
- **x1**, the second initial guess
- **epsilon** the error tolerance to stop the iteration if $|f(x_2)|$ <epsilon

2. Use the solver that you wrote to solve $f(x) = |x - e^{-x}| - 1$, using the initial guesses $x_0 = 0.5$, $x_1 = 1$, and $\varepsilon = 10^{-6}$.

3. Use the **root_scalar** function provided in **scipy.optimize**, with its **secant** method and the same initial guesses and xtol=1e-6 to solve the same function. Compute the difference between this solution and your solution in part 2.

[46]:

```
#Import exp function from math library
from math import exp
```

```python
#This is method to define f(x)
def f(x):
    return abs(x-(exp(-x))) -1

#Question1
#This is a method to use a secant method
def secant(f,x0,x1,epsilon):
    f_x0 = f(x0)
    f_x1 = f(x1)
    while abs(f_x1) > epsilon:
        deno = float(f_x1 - f_x0)/(x1 - x0)
        x2 = x1 - float(f_x1)/ deno
        x0 = x1
        x1 = x2
        f_x0 = f_x1
        f_x1 = f(x1)
        return x2

#Question2
solution = secant(f,0.5,1.0,epsilon = 1.0e-6)
print("My answer is for question 2 is ",solution)

#Question 3
#import root_scalar function from the scipy.optimaze library
from scipy.optimize import root_scalar

solution2 = root_scalar(f, x0 = 0.5, x1 = 1, xtol = 1e-6, method = "secant")
print("My answer for question 3 is", solution2.root)

solution3 = float(solution - solution2.root )
#Find the difference
print("the difference between the two solutions is:", solution3)
```

```
My answer is for question 2 is  1.3499681422628547
My answer for question 3 is 1.2784645427608494
the difference between the two solutions is: 0.07150359950200524
```

```
[ ]:
```