LAB 1: the bisection and secant methods

In this section you will learn how to implement and analyse the Bisection and Secant methods in MATLAB. Before you start, review the "Introduction to MATLAB" notes.

1 Programming the Bisection Method

Revise the lecture notes on the *Bisection Method.* Suppose we want to find a solution to $e^x - (2-x)^3 = 0$ in the interval $[0, 5]$ using Bisection.

- Define the function f as:
  `>> f = @(x)(exp(x) - (2-x).^3);`
- Taking $x_1 = 0$ and $x_2 = 5$, do 8 iterations of the Bisection method.
- Complete the table below. You may use that the solution is (approximately) $\tau = 0.7261444658054950$.

k	x_k	$\lvert\tau - x_k\rvert$
1		
2		
3		
4		
5		
6		
7		
8		

Implementing the Bisection method by hand is very tedious. Here is a program that will do it for you. You don't need to type it all in; you can download it from www.maths.nuigalway.ie/MA385/lab1/Bisection.m

```
3  clear; % Erase all stored variables
4  fprintf('\n\n---------\n Using Bisection\n');
5  % The function is
6  f = @(x)(exp(x) - (2-x).^3);
7  fprintf('Solving f=0 with the function\n');
8  disp(f);
9
10
11 tau = 0.72614446580549503614; % true solution
12 fprintf('The true solution is %12.8f\n', tau);
13
14 %% Our initial guesses are x_1=0 and x_2 =2;
15 x(1)=0;
16 fprintf('%2d | %14.8e | %9.3e \n', ...
17    1, x(1), abs(tau - x(1)));
18 x(2)=5;
19 fprintf('%2d | %14.8e | %9.3e \n', ...
20    2, x(2), abs(tau - x(2)));
21 for k=2:8
22    x(k+1) = (x(k-1)+x(k))/2;
23    if ( f(x(k+1))*f(x(k-1)) < 0)
24       x(k)=x(k-1);
25    end
26    fprintf('%2d | %14.8e | %9.3e\n', ...
27       k+1, x(k+1), abs(tau - x(k+1)));
28 end
```

Read the code carefully. If there is a line you do not understand, then ask a tutor, or look up the online help. For example, find out what that `clear` on Line 3 does by typing `>> doc clear`

Q1. Suppose we wanted an estimate x_k for τ so that $|\tau - x_k| \le 10^{-10}$.

(a) We know from theory that

$$|\tau - x_k| \le (\frac{1}{2})^{k-1}|b-a|.$$

Use this to estimate how many iterations are required in theory.

(b) Use the program above to find how many iterations are required in practice.

2 The Secant method

Recall the the Secant Method in: *Choose x_0 and x_1 so that there is a solution in $[x_0, x_1]$. Then define*

$$x_{k+1} = x_k - f(x_k)\frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})},$$

for $k = 1, 2, 3, \ldots$.

Q2 (a) Adapt the program above to implement the secant method.

(b) Use it to find a solution to $e^x - (2-x)^3 = 0$ in the interval $[0, 5]$.

(c) How many iterations are required to ensure that the error is less than 10^{-10}?

Q3 Recall that the *order of convergence* of a sequence $\{\varepsilon_0, \varepsilon_1, \varepsilon_2, \ldots\}$ is q if

$$\lim_{k\to\infty} \frac{\varepsilon_{k+1}}{\varepsilon_k^q} = \mu,$$

for some constant μ.

We would like to verify that $q = (1+\sqrt{5})/2 \approx 1.618$. This is difficult to do computationally because, after a relatively small number of iterations, the round-off error becomes significant. But we can still try!

Adapt the program above so that at each iteration it displays

$$\frac{|\tau - x_{k+1}|}{|\tau - x_k|}, \quad \frac{|\tau - x_{k+1}|}{|\tau - x_k|^{1.618}}, \quad \frac{|\tau - x_{k+1}|}{|\tau - x_k|^2},$$

and so deduce that the order of converges is greater than 1 (so better than bisection), less than 2, and roughly $(1+\sqrt{5})/2$.

3 To Finish

Before you leave the class upload your MATLAB code for the Q3 (only) to "**Lab 1**" in the "Assignments and Labs" section Blackboard. This file must include your name and ID number as comments. Include your answers to Q1–(b) and Q2–(c) as comments in that file.

Deadline: 5pm Friday, 4 October.

3.1 Extra

The bisection method is popular because it is robust: it will always work subject to minimal constraints. However, it is slow: if the Secant works, then it converges much more quickly. How can we combine these two algorithms to get a fast, robust method? Consider the following problem: *Solve*

$$1 - \frac{2}{x^2 - 2x + 2} = 0 \quad \text{on } [-10, 1].$$

You should find that the bisection method works (slowly) for this problem, but the Secant method will fail. So write a hybrid algorithm that switches between the bisection method and the secant method as appropriate.

Take care to document your code carefully, to show which algorithm is used when.

How many iterations are required?