METHODS FOR FINDING ROOTS OF EQUATIONS

Xinsijian Liu
SUNY Oswego, Oswego, NY
National Central University, Jhongli, Taiwan

Introduction

- Very exciting to join in this research program. These 40 days studied in NCU, I basically learned four methods of finding roots of equations. Learned how to use basic MATLAB function and learned how to input MATLAB code. I got to find out the roots of equations with each different methods.

Methods

- Week 2: The Newton Method
- Week 3: The Bisection Method
- Week 4: The False Position Method
- Week 5: The Secant Method

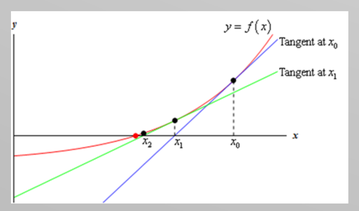
Comparision of Four Methods

The Newton Method

A process for approximating the roots of an equation by replacing the curve representing the equation by its tangent and finding the intersection of the tangent with the x-axis and iterating this process.

The process is repeated as:

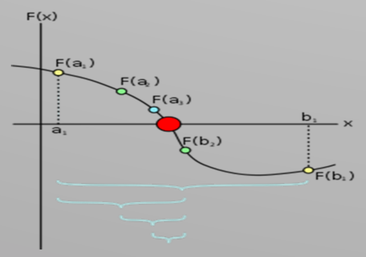
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$



The Bisection Method

The bisection method in mathematics is a root-finding method which repeatedly bisects an interval and then selects a subinterval in which a root must lie for further processing.

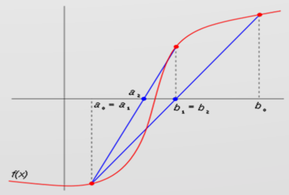
Given a function f (x) continuous on an interval [a,b] and f (a) * f (b) < 0
Do
c = (a+b)/2
if f (a) * f (c) < 0 then b = c
else a = c
while (none of the convergence criteria C1, C2 or C3 is satisfied)



The False Position Method

The false position method or regula falsi method is a term for problem-solving methods in arithmetic, algebra, and calculus. In simple terms, these methods begin by attempting to evaluate a problem using test ("false") values for the variables, and then adjust the values accordingly.

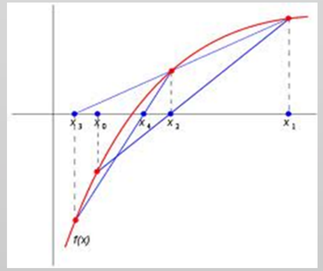
$$c = \frac{a*f(b) - b*f(a)}{f(b) - f(a)}$$



The Secant Method

Secant Method: An iterative method for finding a root of the nonlinear equation f(x) = 0. It is given by the formula:

$$x_{i+1} = x_i - \frac{f(x_i) * (x_i - x_{i-1})}{f(x_i) - f(x_{i-1})} \qquad i = 1,2,3...$$



Example: market equilibrium

The demand equation for the "Tempus" quartz wristwatch is given by

p = 50e^(-0.1(x + 1)^2)

Where x is the quantity demanded per week and p is the unit wholesale price in dollars. National Importers, the supplier of the watches, will make x units available in the market if the unit wholesale price is

p = 10 + 5x^2

Dollars. Find the equilibrium quantity and price

The Newton method method

```
x0 = 5
for i = 1 : 100
    x = x0 - (10 + 5*x0^2 - 50*exp(-0.1*(x0 + 1)^2))/(10*x0 - 50*exp(-0.1*(x0 + 1)^2)*(-0.2*(x0 + 1)));
    x0 = x
    if abs(x0 - 1.688978454087906) < 10^-5
        break
    end
end
disp (x0)
```

The Bisection method

The False Position Method

```
format long
f = @(x) 10 + 5*x^2 - 50*exp(-0.1*(x + 1)^2)
a = 0; b = 5;
step_size = Inf;
e_step = 10^(-6);
n = 0;
while (step_size > e_step)
    c = b - (f(b)*(b-a)/(f(b)-f(a)))
    if f(a)*f(c)<0
        step_size = b - c;
        b = c;
    else
        step_size = c - a;
        a = c;
    end
    n = n +1;
end
fprintf('the approximate root after %d iteration is: %12.5f\n', n, c);
```

The Secant Method

```
f=inline('10 + 5*x^2 - 50*exp(-0.1*(x + 1)^2)');
x(1)=input('enter first guess:');
x(2)=input('enter second guess:');
n=input('enter tolerance:');
iteration=0;
for i = 3:1000
    x(i)=x(i-1)-(f(x(i-1)))*((x(i-1)-x(i-2))/(f(x(i-1))-f(x(i-2))));
    iteration=iteration+1;
    if abs((x(i)-x(i-1))/x(i))*100<n
        root=x(i);
        iteration=iteration;
        break
    end
end
display(x(i))
```

Recording data by Excel

The Newton Method					The Bisection Method			
n(iteration)	root	error	accuracy		n(iteration)	root	error	
1	2.412174552	0.723196098	1.878336		1	2.5	0.811021924	
2	1.736764429	0.047785975	1.994422		2	1.25	0.438978076	
3	1.689268821	0.000290367			3	1.875	0.186021924	
4	1.688978465	1.10308E-08			4	1.5625	0.126478076	
…					…			
4	1.688978454				22	1.688978076		
The False Position Method					The Secant Method			
n(iteration)	root	error	accuracy		n(iteration)	root	error	accuracy
1	1.04342644	0.64555356	1.071463		1	1.04342644	0.645552014	1.77097
2	1.501307531	0.187672469	1.016961		2	1.501307531	0.187670923	1.670322
3	1.639031001	0.049948999			3	1.710026458	0.021048004	
4	1.675981228	0.012998772			4	1.688433833	0.000544621	
…					…			
12	1.68898				7	1.688978454		

Conclusion

By solving the equation, I got to know different iteration between each one. I recorded data from those four methods. And I found out the best method which is the Newton Method to solve the question.