Solving an Equation using Newton-Raphson Method

we obtain a better approximation

Continue in this way .

If x_n is the current estimate, then the next estimate x_{n+1} is given by

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

Steps to find root using Newton's Method:

1) Check if the given function is differentiable or not. If the function is not differentiable, Newton's method cannot be applied.
2) Find the first derivative $f'(x)$ of the given function $f(x)$.
3) Take an initial guess root of the function, say $x_{1.}$
4) Use Newton's iteration formula to get new better approximate of the root, say x_2 $x_2 = x_1 - f(x_1)/f'(x_1)$
5) Repeat the process for x_3, x_4… till the actual root of the function is obtained, fulfilling the tolerance of error.

Ex: using Newton-Raphson Method for **$f(x) = x^3 - x - 1$**

Solution:

Given function: **$x^3 - x - 1 = 0$**, is differentiable.

The first derivative of f(x) is $f'(x) = 3x^2 - 1$

Lets determine the guess value.

$f(1) = 1 - 1 - 1 = -1$ and $f(2) = 8 - 2 - 1 = 5$

Therefore, the root lies in the interval [1, 2]. So, assume $x_1 = 1.5$ as the initial guess root of the function $f(x) = x^3 - x - 1$.

Now,

$f(1.5) = 1.5^3 - 1.5 - 1 = 0.875$

$f'(1.5) = 3 * 1.5^2 - 1 = 5.750$

Using Newton's iteration formula:

$x_2 = x_1 - f(x_1)/f'(x_1) = 1.5 - 0.875/5.750 = 1.34782600$

n	x_n	$f(x_n)$
1	1.34782608696	0.100682173091
2	1.32520039895	0.002058361917
3	1.32471817400	0.000000924378
4	1.32471795724	0.000000000000
5	0.000000000000	

```
% Program Code of Newton-Raphson Method in MATLAB

a=input('Enter the function in the form of variable x:','s');
x(1)=input('Enter Initial Guess:');
error=input('Enter allowed Error:');
f=inline(a)
dif=diff(sym(a));
d=inline(dif);
for i=1:100
x(i+1)=x(i)-((f(x(i))/d(x(i))));
err(i)=abs((x(i+1)-x(i))/x(i));
if err(i)<error
break
end
end
root=x(i)
```