

Numerical Technique(MA202)

Lab Assignment 7

March 6, 2020

Course Instructor: Dr. Jignesh S. Bhatt

Student and ID : Pushkar Patel (201851094)

1 Consider the nonlinear equations

1.1 $f(x) = 2.0 - x + \ln(x) = 0$

1.2 $f(x) = x^2 - 3x + 1 = 0$

Write a MATLAB function to solve the non-linear equations using Bisection method, Fixed point iteration method, Newton-Raphson method. Use `fzero()` and `fsolve()` MATLAB functions to verify your answers.

Matlab script to calculate solving non-linear equation $f(x) = 2.0 - x + \ln(x) = 0$

Table of Contents

.....	1
Function Calling	1
Bisection Method	1
Fixed Point Iterator;	2
NEWTON METHOD	3

```
clc;  
clear;  
close all;
```

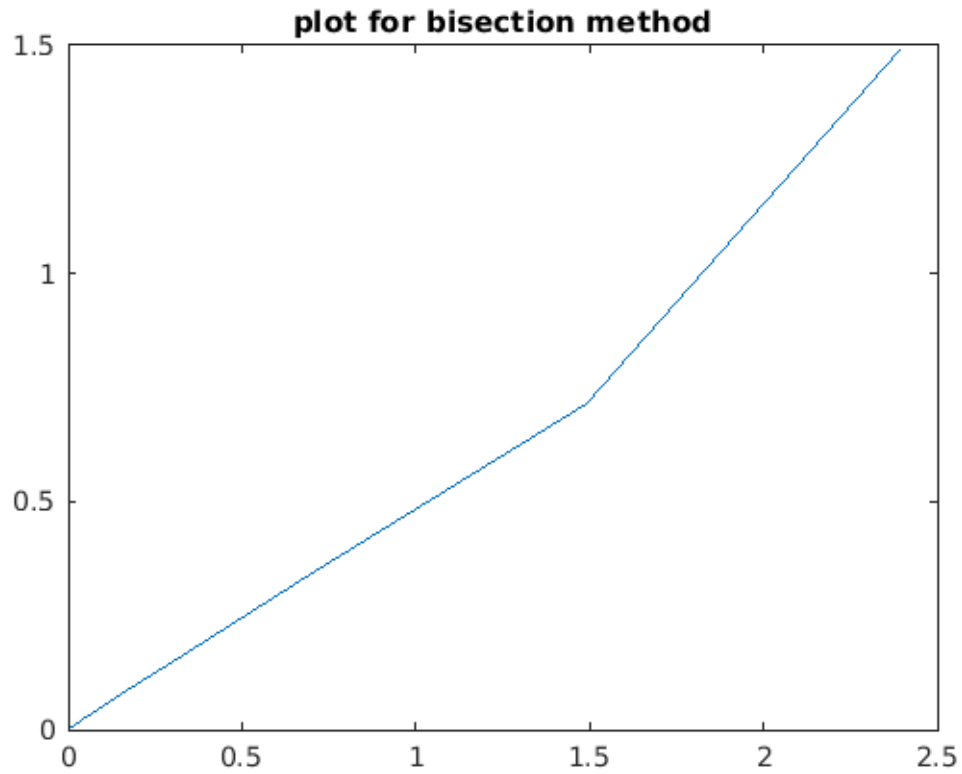
Function Calling

```
bisecting()  
fpt()  
newton()
```

Bisection Method

```
function bisecting()  
x1=1.0;  
x2=5.0;  
f1=finder(1,x1);  
f2=finder(1,x2);  
maxitr=50;  
tol=10^-6;  
if(f1*f2>0)  
    fprintf("Invalid guess !!");  
end  
err=zeros(1,maxitr);  
  
for itr=1:1:maxitr  
    m=(x1+x2)/2;  
    fm=finder(1,m);  
    err(itr)=abs(f2-f1);  
    if(err(itr)<tol)  
        break  
    end  
    if(f1*fm>0)  
        x1=m;
```

```
f1=fm;  
else  
    x2=m;  
    f2=fm;  
end  
  
end  
m  
figure(1)  
plot(err(1:itr-1),err(2:itr))  
title('plot for bisection method')  
end  
  
m =  
  
3.1462
```



Fixed Point Iterator;

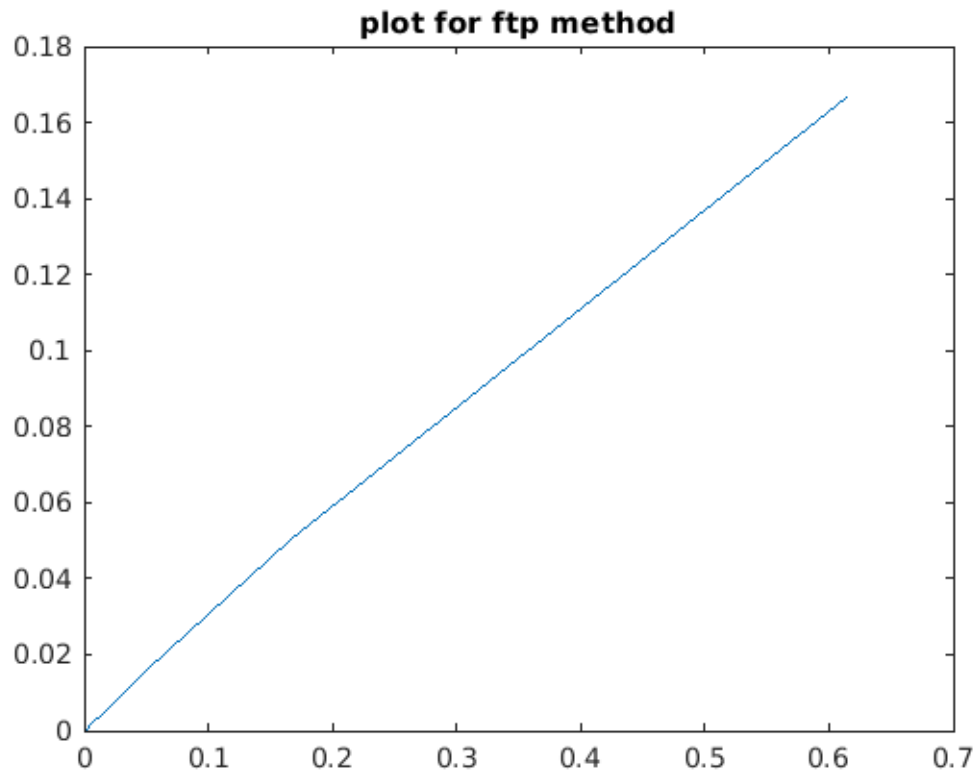
```
function fpt()  
x=4;  
xold=4;  
maxitr=50;  
errfp=zeros(1,maxitr);
```

```
tol=10^-6;

for itr=1:1:maxitr
    x=finder(2,xold);
    errfp(itr)=abs(x-xold);
    xold=x;
    if errfp(itr)<tol
        break
    end
end
x
figure(2)
plot(errfp(1:itr-1),errfp(2:itr))
title('plot for ftp method')
end

x =

    3.1462
```



NEWTON METHOD

```
function newton()
x=4;
```

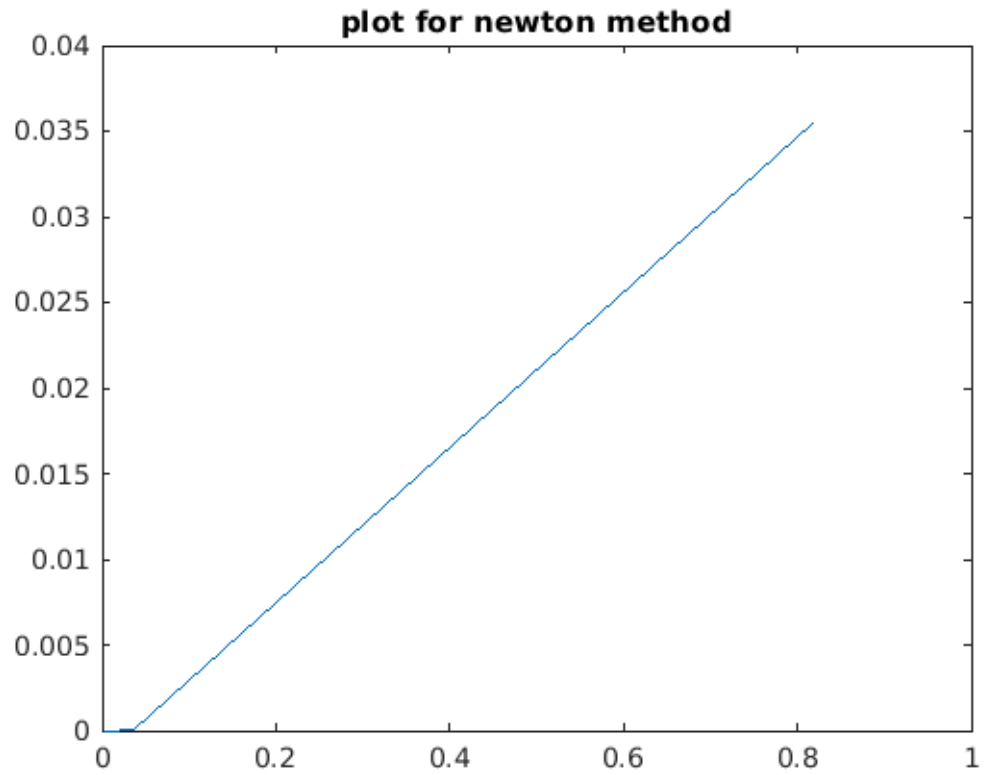
```
maxitr=50;
errnt=zeros(1,maxitr);
tol=10^-6;

for itr=1:1:maxitr
    fx=finder(1,x);
    dfx=finder(3,x);
    xnew=x-fx/dfx;
    errnt(itr)=abs(xnew-x);
    if(errnt(itr)<tol)
        break
    end
    x=xnew;

end
xnew
figure(3)
plot(errnt(1:itr-1),errnt(2:itr))
title('plot for newton method')
end
function fval=finder(arg,xval)
    if arg==1
        fval=2-xval+log(xval);
    elseif arg==2
        fval=2+log(xval);
    else
        fval=1/(xval)-1;
    end
end

xnew =

    3.1462
```



Published with MATLAB® R2019b

Matlab script to calculate solving non-linear equation $f(x) = x^2 - 3x + 1 = 0$

Table of Contents

.....	1
Function Calling	1
Bisection Method	1
Fixed Point Iterator;	2
NEWTON METHOD	3

```
clc;  
clear;  
close all;
```

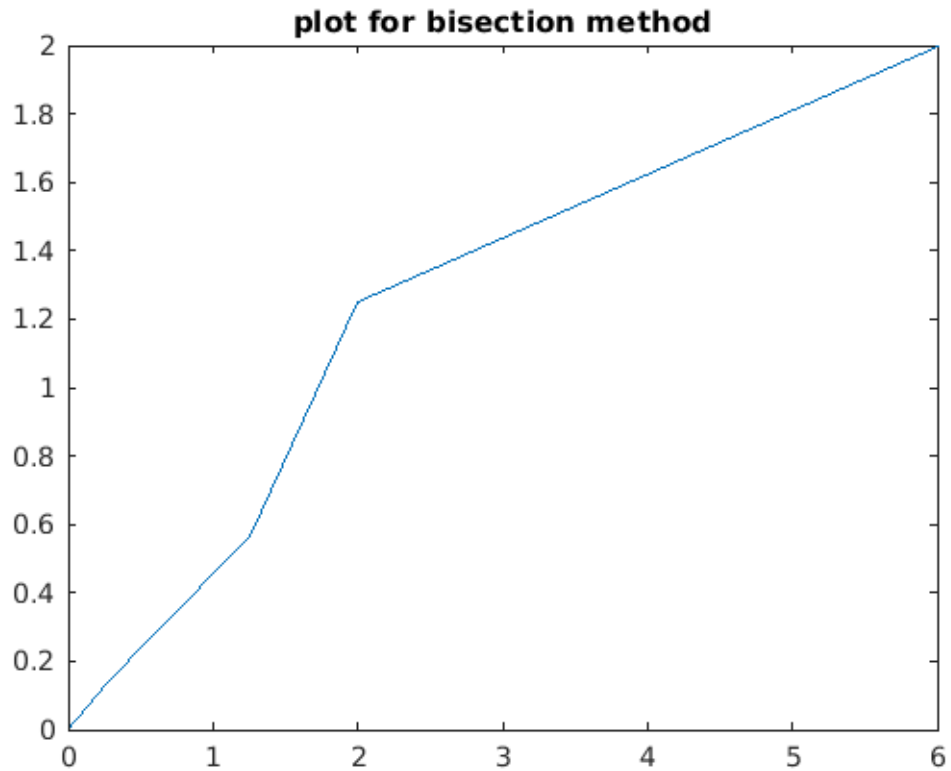
Function Calling

```
bisecting()  
fpt()  
newton()
```

Bisection Method

```
function bisecting()  
x1=2.0;  
x2=4.0;  
f1=finder(1,x1);  
f2=finder(1,x2);  
maxitr=50;  
tol=10^-6;  
if(f1*f2>0)  
    fprintf("Invalid guess !!");  
end  
err=zeros(1,maxitr);  
  
for itr=1:1:maxitr  
    m=(x1+x2)/2;  
    fm=finder(1,m);  
    err(itr)=abs(f2-f1);  
    if(err(itr)<tol)  
        break  
    end  
    if(f1*fm>0)  
        x1=m;
```

```
f1=fm;  
else  
    x2=m;  
    f2=fm;  
end  
  
end  
m  
figure(1)  
plot(err(1:itr-1),err(2:itr))  
title('plot for bisection method')  
end  
  
m =  
  
2.6180
```

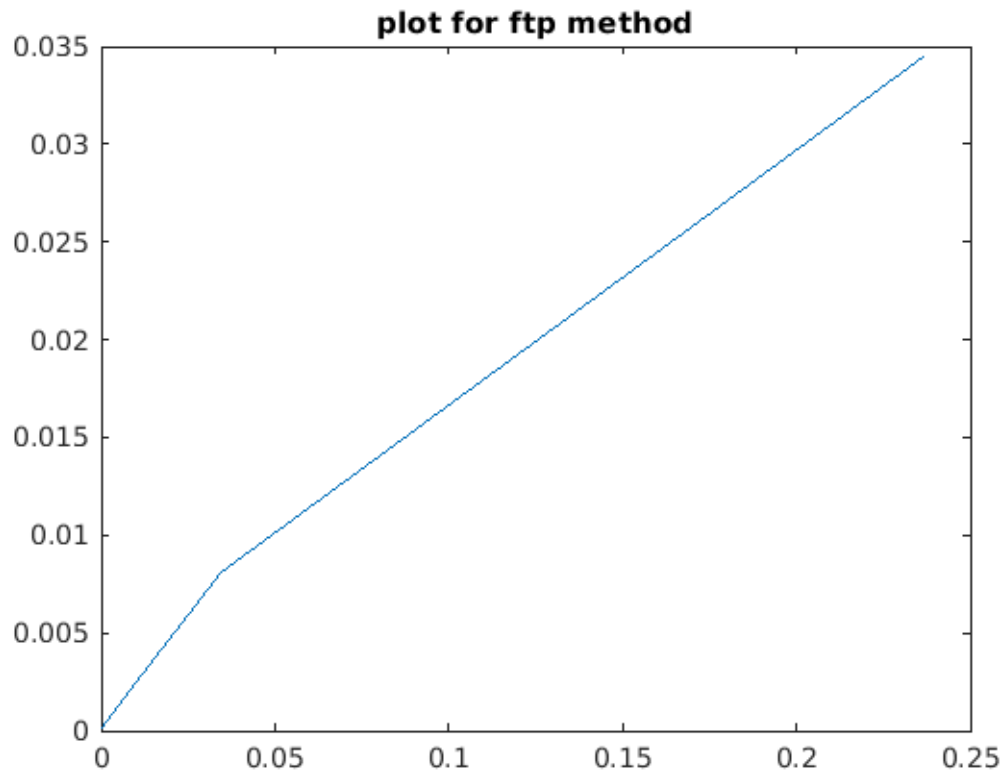


Fixed Point Iterator;

```
function fpt()  
x=0.1;  
xold=x;  
maxitr=50;  
errfp=zeros(1,maxitr);
```



```
tol=10^-6;  
  
for itr=1:1:maxitr  
    x=finder(2,xold);  
    errfp(itr)=abs(x-xold);  
    xold=x;  
    if errfp(itr)<tol  
        break  
    end  
end  
x  
figure(2)  
plot(errfp(1:itr-1),errfp(2:itr))  
title('plot for ftp method')  
end  
  
x =  
  
    0.3820
```



NEWTON METHOD

```
function newton()  
x=4;
```

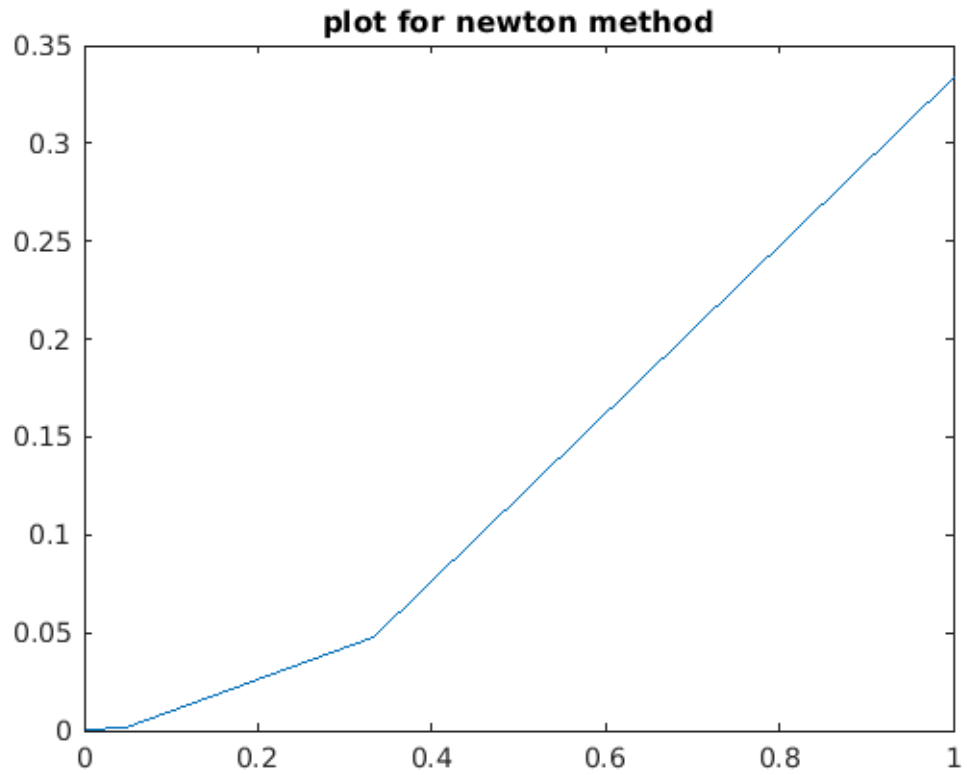
```
maxitr=50;
errnt=zeros(1,maxitr);
tol=10^-6;

for itr=1:1:maxitr
    fx=finder(1,x);
    dfx=finder(3,x);
    xnew=x-fx/dfx;
    errnt(itr)=abs(xnew-x);
    if(errnt(itr)<tol)
        break
    end
    x=xnew;

end
xnew
figure(3)
plot(errnt(1:itr-1),errnt(2:itr))
title('plot for newton method')
end
function fval=finder(arg,xval)
    if arg==1
        fval=xval^2-3*xval+1;
    elseif arg==2
        fval=(xval^2+1)/3;
    else
        fval=2*xval-3;
    end
end

xnew =

    2.6180
```



Published with MATLAB® R2019b