Ally Racho
CMPSC456
Homework 1

Homework 1

1. $f(x_1, x_2) = e^{x_1 - 3x_2} + e^{x_1 + 3x_2} + e^{-x_1}$

gradient: $\nabla f(x_1, x_2) = \begin{pmatrix} e^{x_1 - 3x_2} + e^{x_1 + 3x_2} - e^{-x_1} \\ -3e^{x_1 - 3x_2} + 3e^{x_1 + 3x_2} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

hessian: $\nabla^2 f(x_1, x_2) = \begin{pmatrix} e^{x_1 - 3x_2} + e^{x_1 + 3x_2} + e^{-x_1} & -3e^{x_1 - 3x_2} + 3e^{x_1 + 3x_2} \\ -3e^{x_1 - 3x_2} + 3e^{x_1 + 3x_2} & 9e^{x_1 - 3x_2} + 9e^{x_1 + 3x_2} \end{pmatrix}$

$= \begin{pmatrix} 3 & 0 \\ 0 & 18 \end{pmatrix}$

newton step: $x_1 = x_0 - [\nabla^2 f(x_0)]^{-1} \nabla f(x_0) = \begin{pmatrix} 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 3 & 0 \\ 0 & 18 \end{pmatrix}^{-1} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -1/3 \\ 0 \end{pmatrix}$

After 1 iteration we find the approx to be $\begin{pmatrix} -1/3 \\ 0 \end{pmatrix}$

2. We find we need 4 iterations before convergence for Newton's method but 7 iterations for modified version. This implies Newton's method has a faster convergence.

Below are screenshots of my code used in finding the answer to problem 2.

Driver file:

```
%% Question 2

x0 = [-5 -4]';
tol = 10^-9;

% Newton's Method
ans1 = newton(x0, @f1, tol);

% Modified Newton's Method
ans2 = newtonmod(x0, @f1, tol);
```

Equation inputs:

```
function [f,G,H] = f(x)

f = [x(1)^2 - x(2)^2 + 2*x(2); 2*x(1) + x(2)^2 - 6];

G = [0;0];
G(1) = x(1)^2 - x(2)^2 + 2*x(2); %f1
G(2) = 2*x(1) + x(2)^2 - 6; %f2

%jacobian matrix
H(1,1) = 2*x(1);
H(1,2) = -2*x(2)+2;
H(2,1) = 2;
H(2, 2) = 2*x(2);

end
```

Newton's method function:

```
function x= newton(x0,f,tol)
dx=1.0;
while dx>tol
    [F,G,H]=feval(f,x0);
    x1 = x0 - H\G;
    dx=norm(x1-x0);
    x0=x1;
    disp(x0');
end
x=x0;
```

Output:

```
  -4.8667   -3.9667

  -4.8642   -3.9659

  -4.8642   -3.9659

  -4.8642   -3.9659
```

Modified Newton's method function:

```
function x= newtonmod(x0,f,tol)
dx=1.0;
[F,G,H]=feval(f,x0); %get jacobian at x0 and do not recalculate

while dx>tol
    [F,G]=feval(f,x0);
    x1 = x0 -  H\G;
    dx=norm(x1-x0);
    x0=x1;
    disp(x0');
end
x=x0;
```

Output:

```
-4.8667   -3.9667

-4.8643   -3.9659

-4.8642   -3.9659

-4.8642   -3.9659

-4.8642   -3.9659

-4.8642   -3.9659

-4.8642   -3.9659
```