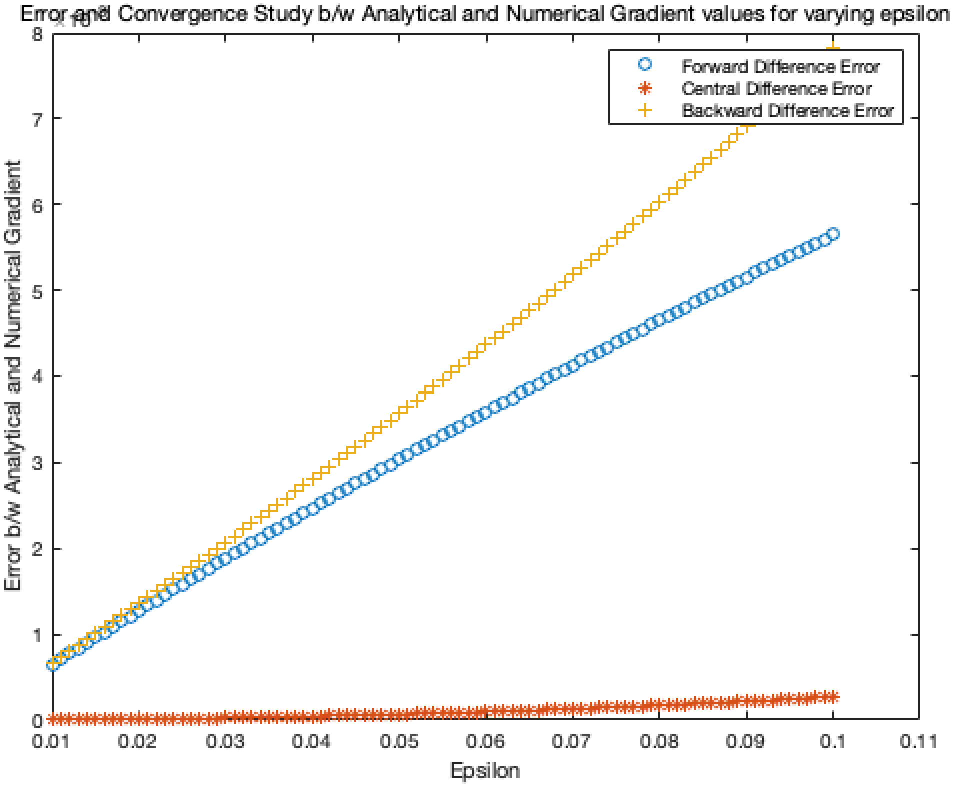
```
function [] = Gradient()
clear all;
close all;
disp('========================================');
disp('     GRADIENT by Finite Differences Assignment 1 Q2     ');
disp('     AML 771: Design Optimization and Decision Theory   ');
disp('     Student: Debjit Hore        ');
disp('========================================');
% computes the gradient of a function supplied by user in #Subroutine
 Getfun#
% using forward, backward and central differences, and compares this
 with
% analytical gradient provided by the user in #Subroutine Gradient#

      N = 2;              % N: Dimension of the space
      epsilon = [0.01:0.001:0.1];
%     ===== READ POINT (Initial point)
      X = [0.5, 1.5]';
      [F0] = GETFUN (X);
% ----- GRADIENTS
      [G] = GRAD (X);
      for J=1:length(epsilon)
%----- FORWARD DIFFERENCE EVALUATION
      for I = 1:N
      A = X(I);
      DX = epsilon(J);      % DX: Purturbation of X
      if X(I) ~= 0.
        DX =  epsilon(J) * abs(X(I));
      end
      X(I) = X(I) + DX;
      [F] = GETFUN (X);
      X(I) = A;
      GF(J,I) = (F - F0) / DX;          % Gradient of F calculation
      end
% ----- BACKWARD DIFFERENCE EVALUATION
      for I = 1: N
      A = X(I);
      DX =  epsilon(J);
        if X(I) ~= 0.
          DX =  epsilon(J) * abs(X(I));
      end
      X(I) = X(I) - DX;
      [F] = GETFUN (X);
      X(I) = A;
      GB(J, I) = (F0 - F) / DX;
      end
% ----- CENTRAL DIFFERENCE EVALUATION
      for I = 1: N
        A = X(I);
      DX =  epsilon(J);
      if X(I) ~= 0.
        DX =  epsilon(J) * abs(X(I));
      end
      X(I) = X(I) - .5 * DX;            %purturbing back
      [F1] = GETFUN (X);
      X(I) = A;
      X(I) = X(I) + .5 * DX;            %purturbing forward
      [F] = GETFUN (X);
      X(I) = A;
      GC(J,I) = (F - F1) / DX;
      end
      end
     GF_Error= sqrt((GF(:,1)-G(1)).^2+(GF(:,2)-G(2)).^2);
     GB_Error= sqrt((GB(:,1)-G(1)).^2+(GB(:,2)-G(2)).^2);
     GC_Error= sqrt((GC(:,1)-G(1)).^2+(GC(:,2)-G(2)).^2);
     disp(' ')
     disp('  Analytical Gradient  Forw. Diff.  Backw. Diff.  Central
 Diff.')
    for I = 1: N
        disp(sprintf(' %15.4E %15.4E %15.4E %15.4E\n',G(I), GF(1,I),
 GB(1,I), GC(1,I)));
    end
     figure;
     plot(epsilon, GF_Error, 'o');
     hold on;
     plot(epsilon, GC_Error,  '*');
     plot(epsilon, GB_Error,  '+');
     hold off;
     xlabel('Epsilon');
     ylabel('Error b/w Analytical and Numerical Gradient');
     legend('Forward Difference Error', 'Central Difference
 Error', 'Backward Difference Error');
     title('Error and Convergence Study b/w Analytical and Numerical
 Gradient values for varying epsilon', 'FontWeight',"normal");
    %KINDLY NOTE THAT THE ERRORS IN THIS HAVE A POWER OF 10E-3 THAT'S
 ONLY
    %VISIBLE IF THE GRAPH IS OPENED IN FULL SCREEN MODE.
      function [F] = GETFUN(X)
% ===== DEFINE FUNCTION
      F= (4*X(2)^2-X(1)*X(2))/(10000*(X(2)*X(1)^3-X(1)^4));

      function [G] = GRAD(X)
% ===== GRADIENTS
      G(1) = ((- 40000*X(1)^3 + 30000*X(2)*X(1)^2)*(- 4*X(2)^2
 + X(1)*X(2)))/(- 10000*X(1)^4 + 10000*X(2)*X(1)^3)^2 - X(2)/(-
 10000*X(1)^4 + 10000*X(2)*X(1)^3);
      G(2) = (10000*X(1)^3*(- 4*X(2)^2 + X(1)*X(2)))/(- 10000*X(1)^4
 + 10000*X(2)*X(1)^3)^2 - (X(1) - 8*X(2))/(- 10000*X(1)^4 +
 10000*X(2)*X(1)^3);

========================================
     GRADIENT by Finite Differences Assignment 1 Q2
     AML 771: Design Optimization and Decision Theory
     Student: Debjit Hore
========================================
  Analytical Gradient  Forw. Diff.  Backw. Diff.  Central Diff.
     -3.4200E-02      -3.3553E-02    -3.4868E-02     -3.4203E-02

      2.6000E-03       2.6089E-03     2.5909E-03      2.6000E-03
```



Published with MATLAB® R2021a