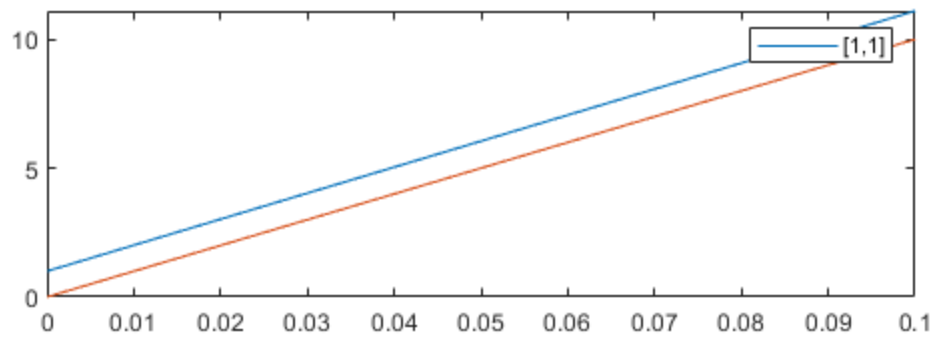
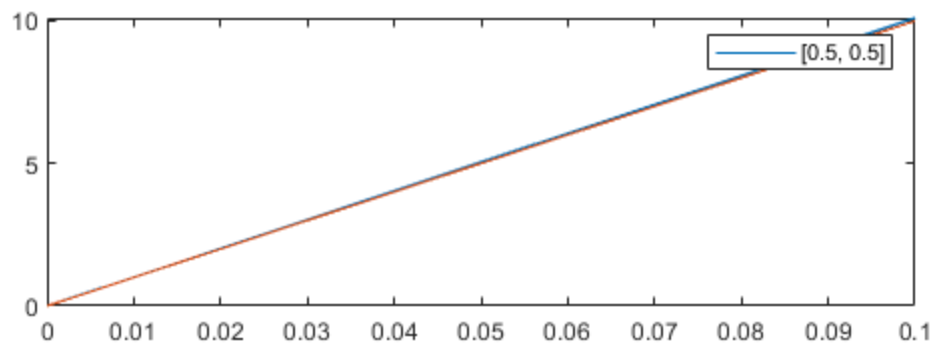

Calculating derivative with forward-difference scheme

```
close all;
clear;
clc;
f = @(x) 100*(x(2)-x(1))^2 + (1-x(1))^2;
grad_1 = @(x) -200*(x(2)-x(1)) - 2*(1-x(1));
grad_2 = @(x) 200*(x(2)-x(1));
e1 = [1; 0];
e2 = [0; 1];

epsilon = 0.00001;
x_first = [0.5; 0.5];
x_second = [1; 1];
true_grad_x_first = zeros(2, 3);
grad_approx_x_first = zeros(2, 3);
true_grad_x_second = zeros(2, 3);
grad_approx_x_second = zeros(2, 3);
true_grad_x2 = zeros(2, 3);
grad_approx_x2 = zeros(2, 3);
epsilons = zeros(1, 3);
for k=1:3
    epsilons(k) = epsilon;
    del_x1 = (f(x_first+epsilon*e1)-f(x_first))/epsilon;
    del_x2 = (f(x_first+epsilon*e2)-f(x_first))/epsilon;
    grad_approx_x_first(:,k) = [del_x1;del_x2];
    true_grad_x_first(:,k) = [grad_1(x_first); grad_2(x_first)];
    del_x1 = (f(x_second+epsilon*e1)-f(x_second))/epsilon;
    del_x2 = (f(x_second+epsilon*e2)-f(x_second))/epsilon;
    grad_approx_x_second(:,k) = [del_x1;del_x2];
    true_grad_x_second(:,k) = [grad_1(x_first); grad_2(x_first)];
    epsilon = epsilon*100;
end

error_first = abs(true_grad_x_first-grad_approx_x_first);
error_second = abs(true_grad_x_second-grad_approx_x_second);
figure();
subplot(211);
plot(epsilons, error_first(1,:));
hold on;
plot(epsilons, error_first(2, :));
legend(' [0.5, 0.5] ');
subplot(212);
plot(epsilons, error_second(1,:));
hold on;
plot(epsilons, error_second(2, :));
legend(' [1,1] ')
```



Published with MATLAB® R2019b