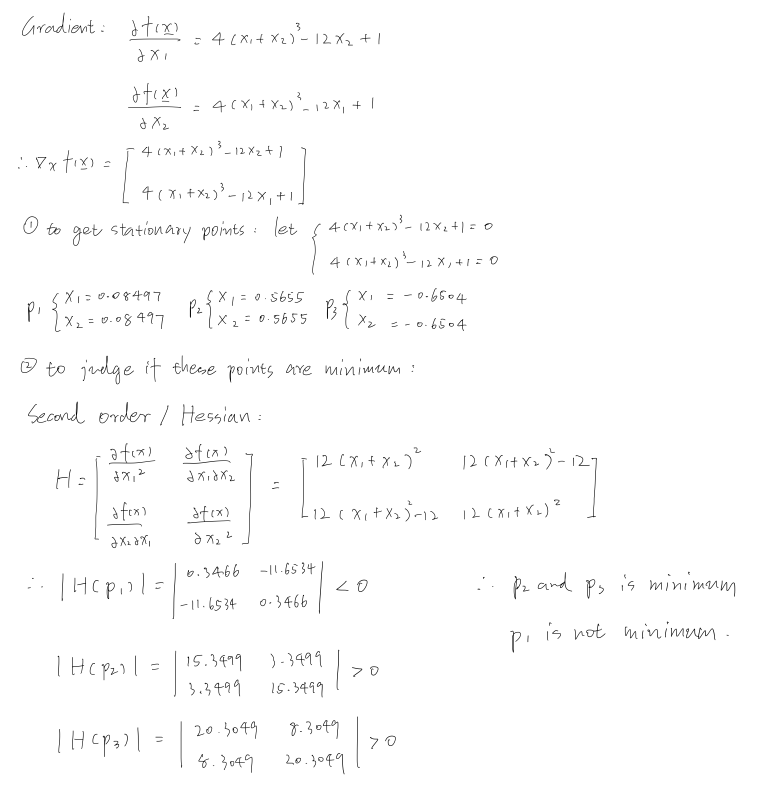
**ECE 595 Homework 3 Due: Feb. 28, 4 PM**

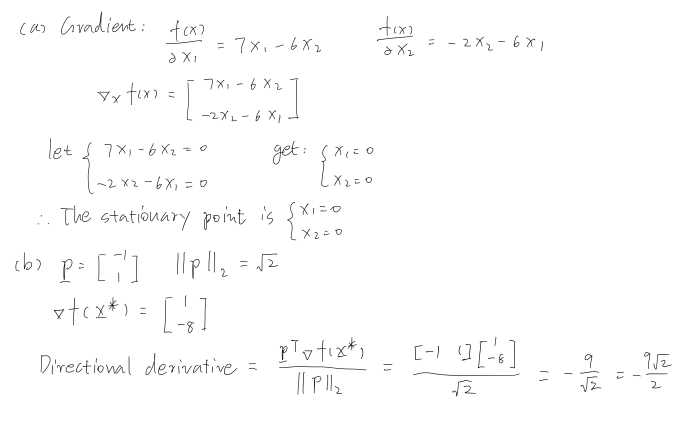
1. For the following function of two variables, find the stationary points, i.e., where the gradient is zero.



Which of the stationary points, if any, is the minimum?

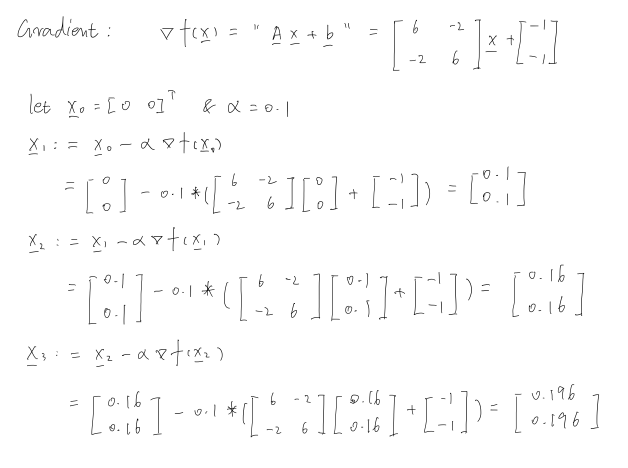


1. For the function , find (a) the stationary points, and (b) the gradient at  in the direction of .



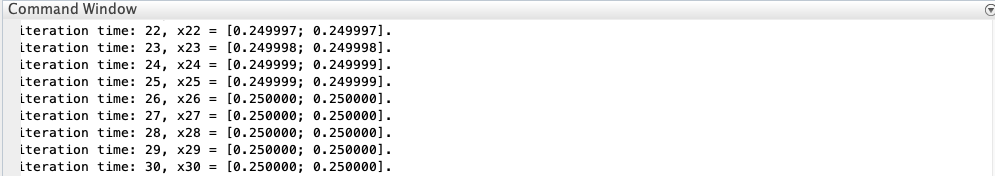
1. Find the minimum point for the function, ,

using the steepest descent algorithm with the initial guess of  and a learning rate of 0.1. Show a few iterations by hand calculations. Use MATLAB to verify your calculations. Show your code and final results.

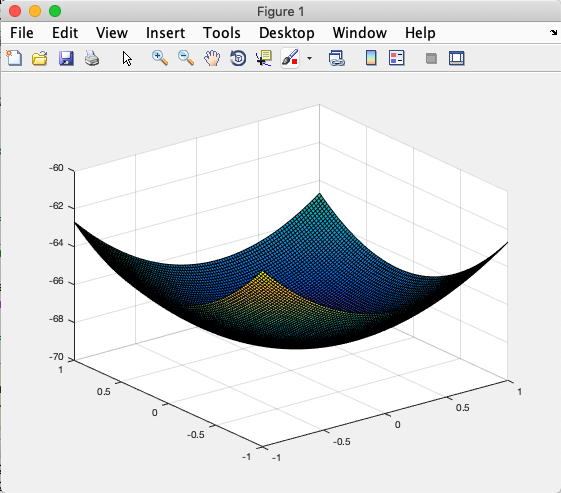


For MATLAB coding, I got the result like:



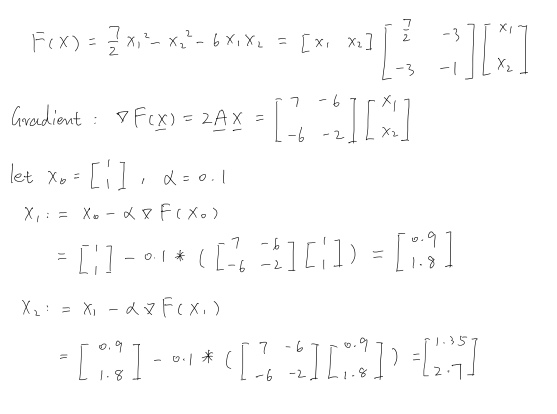


The function will converge after around 26 iterations, also I plot the function to check if there is a global minimum.

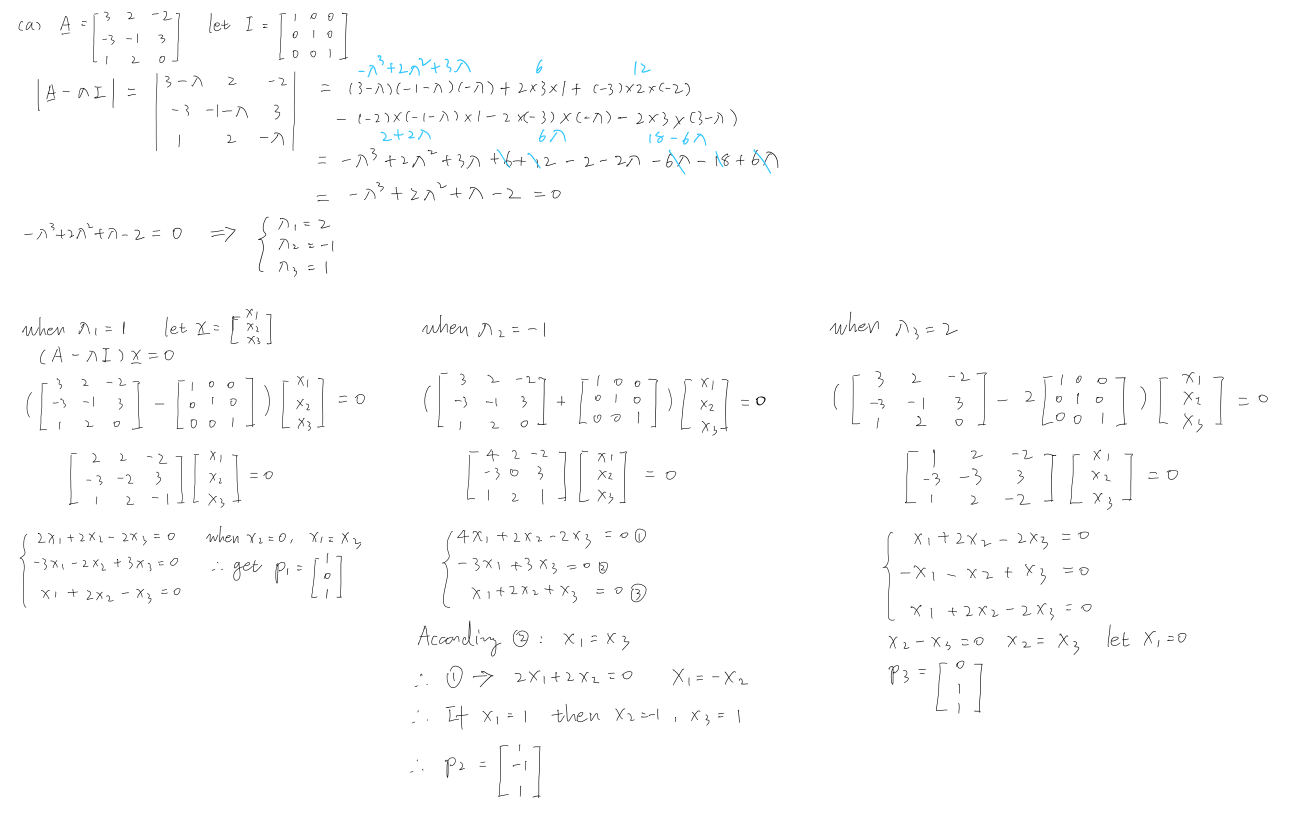


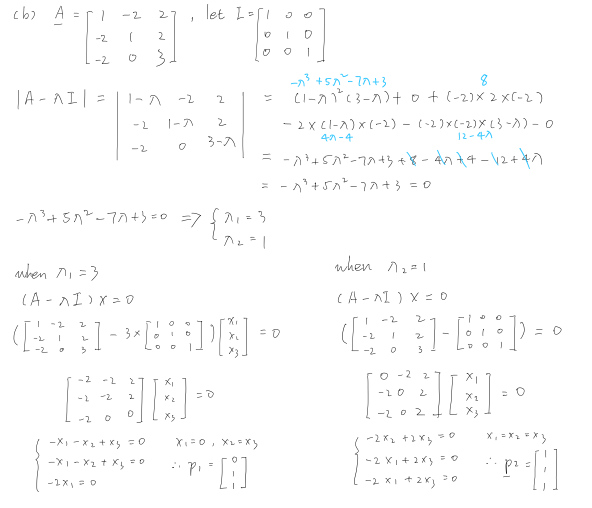
The shape of the function is a perfect bowl shape and its global minimum is at the point [0.25 0.25]**T**.

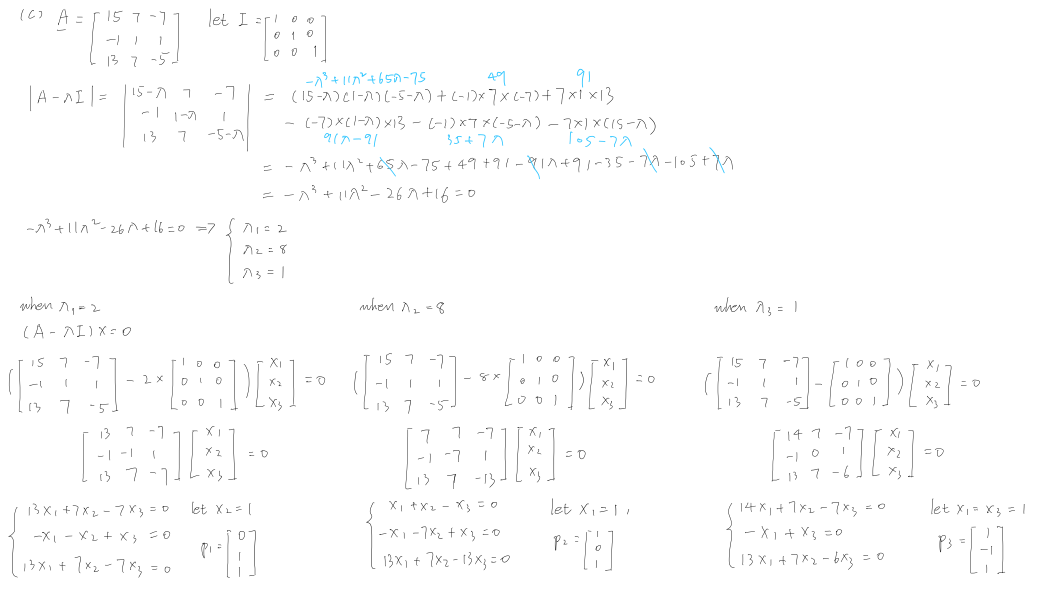
1. For the function , perform two iterations of the **steepest descent** algorithm, starting at the initial guess of .



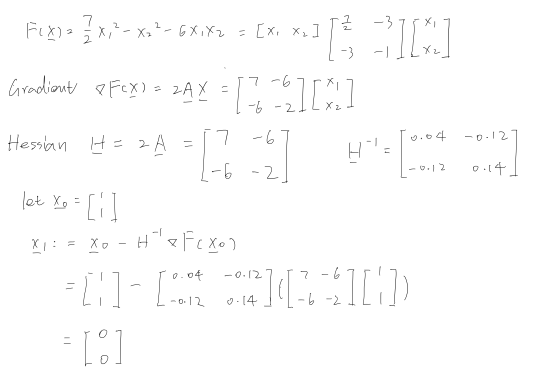
1. Determine the eigenvalues and eigenvectors for the following matrices.
2.  (b)  (c) 







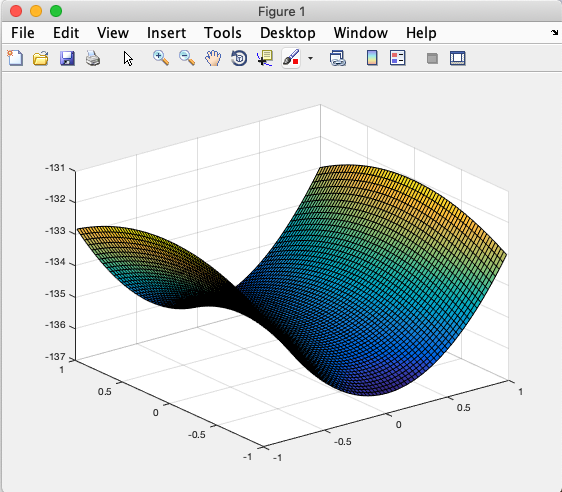
1. For the function given in Problem 4, show an iteration of **Newton’s method** using the initial guess of  . Complete the solution in MATLAB. Compare the two algorithms for this problem.



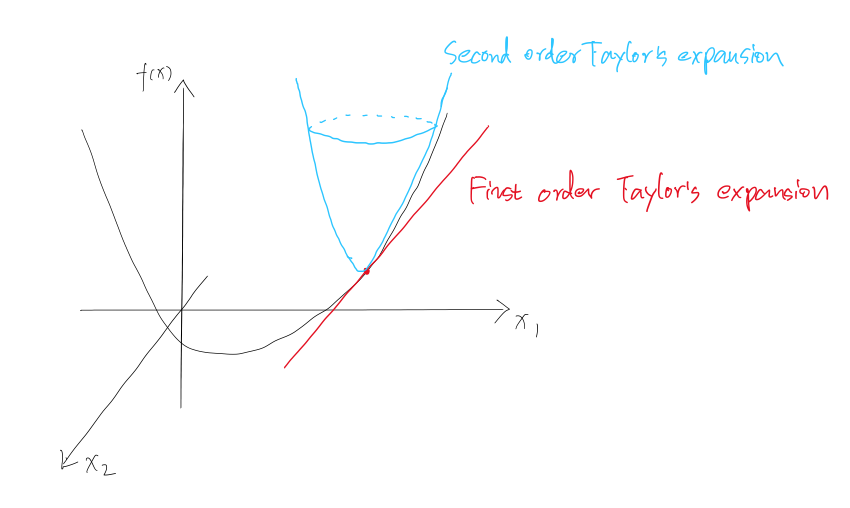
For MATLAB coding:



Also, I plot the function and find that the point [0 0]T is a saddle point.



To compare the steepest descent and Newton’s method, in my opinion, according to Taylor’s series, steepest descent uses a straight line to fit the curve, while Newton’s method is using a curve to fit the curve.



Therefore, the Newton’s method is much faster than steepest descent.

Code:

Question\_3.m

%% Machine Learning Homework 3 Question 3

% Author: Xinrun Zhang

% Time: 02/28/2019 13:42

% =====================================================================

%% initializting

% gradient = A \* x[i] + b;

A = [6 -2; -2 6];

b = [-1; -1];

% initial guess x, learning rate alpha and iteration

x = [0; 0];

alpha = 0.1;

itr = 30;

% =====================================================================

%% steepest descent

for i = 1:itr

    x = x - alpha \* (A \* x + b);

    fprintf('iteration time: %d, x%d = [%f; %f].\n',i,i,x(1),x(2));

end

% =====================================================================

%% plot

[x1, x2] = meshgrid(-1:0.02:1);

z = 3 \* (x1.^2) + 3 \* (x2.^2) - 2 \* x1 \* x2 - x1 - x2;

surf(x1, x2, z);

% =====================================================================

Question\_4.m

%% Machine Learning Homework 3 Question 6

% Author: Xinrun Zhang

% Time: 02/28/2019 14:03

% =====================================================================

%% initializting

% gradient = 2 \* A \* x

% Hessian = 2 \* A

A = [7/2 -3; -3 1];

H = 2 \* A;

H\_inv = (H)^-1;

% initial guess x

x = [1; 1];

% =====================================================================

%% Newton's Method

x = x - H\_inv \* (2 \* A \* x);

fprintf('x found by Newtons Method is: [%f; %f]\n',x(1), x(2));

% =====================================================================

%% plot

[x1, x2] = meshgrid(-1:0.03:1);

z = (7/2) \* (x1.^2) - (x2.^2) - 6 \* x1 \* x2;

surf(x1, x2, z);

% =====================================================================