Coursework 1

Mathematics for Machine Learning (CO-496)

1 Differentiation

a) Write f1(x) in the completed square form $(x-c)^T C(x-c)+c_0$.

Answer:

First we can simplify two equation to similar form

$$f_1(x) = x^T x + x^T Bx - a^T x + b^T x = x^T(B+I)x + (b^T - a^T)x$$

and

$$(x-\mathrm{c})^T C(x-c) + c_0 = x^T Cx - 2c^T Cx + c^T Cc + c_0$$

So

$$C = (B+I) = \begin{pmatrix} 4 & -1 \\ -1 & 4 \end{pmatrix}$$

Then

$$2c^T C = (b^T - a^T)$$

so we can get that

$$c = \begin{pmatrix} \frac{1}{6} \\ \frac{1}{6} \end{pmatrix}$$

Then because $c^T Cc + c_0 = 0$

So

$$c_0 = -\frac{1}{6}$$

The answer of this question is

$$C = (B+I) = \begin{pmatrix} 4 & -1 \\ -1 & 4 \end{pmatrix}$$

$$c = \begin{pmatrix} \frac{1}{6} \\ \frac{1}{6} \end{pmatrix}$$

$$c_0 = -\frac{1}{6}$$

b) Explain how you can tell that f1 has a minimum point. State the minimum value of f1 and find the input which achieves this minimum.

Answer:

Let $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$, then

$$f_1(x) = 4x_1^2 + 4x_2^2 - x_1 - x_2 - 2x_1x_2$$

After performing the second order derivative, we can get the Hessian matrix of $f_1(x)$

$$\mathrm{H} = \begin{pmatrix} 8 & -2 \\ -2 & 8 \end{pmatrix}$$

Then calculate the eigenvalues γ of $f_1(x)$

First, let

$$|H - \gamma I| = 0$$

We can get $\gamma = \begin{cases} 6 \\ 10 \end{cases}$

Because all eigenvalues are greater than 0, so H is positive definite, so we can confirm that f_1 has a minimum point.

To get the point achieving the minimum, we should first get the first order derivative of f_1,and let them equal to 0,

$$\frac{\partial f_1}{\partial x_1} = 8x_1 - 2x_2 - 1 = 0$$

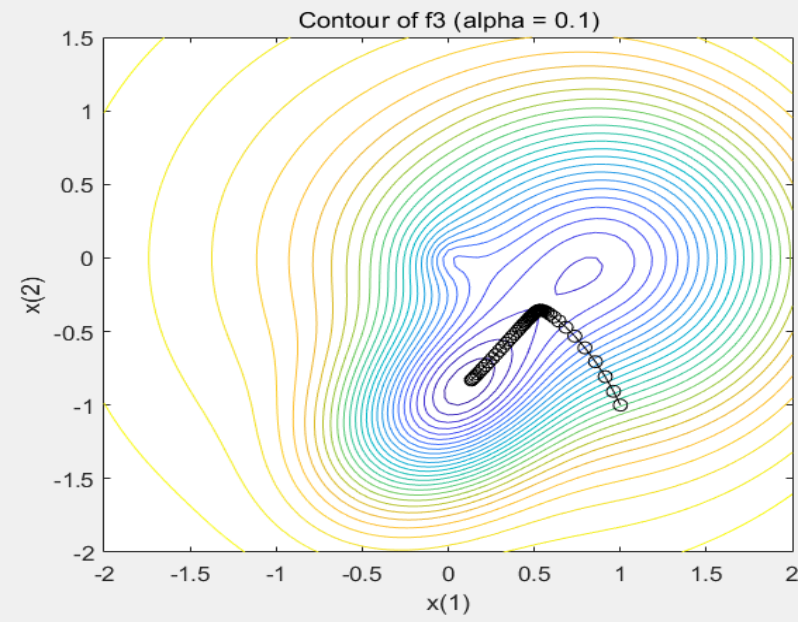
$$\frac{\partial f_1}{\partial x_2} = 8x_2 - 2x_1 - 1 = 0$$

So $x = \begin{pmatrix} \frac{1}{6} \\ \frac{1}{6} \end{pmatrix}$, at this point we can get the minimal $-\frac{1}{6}$

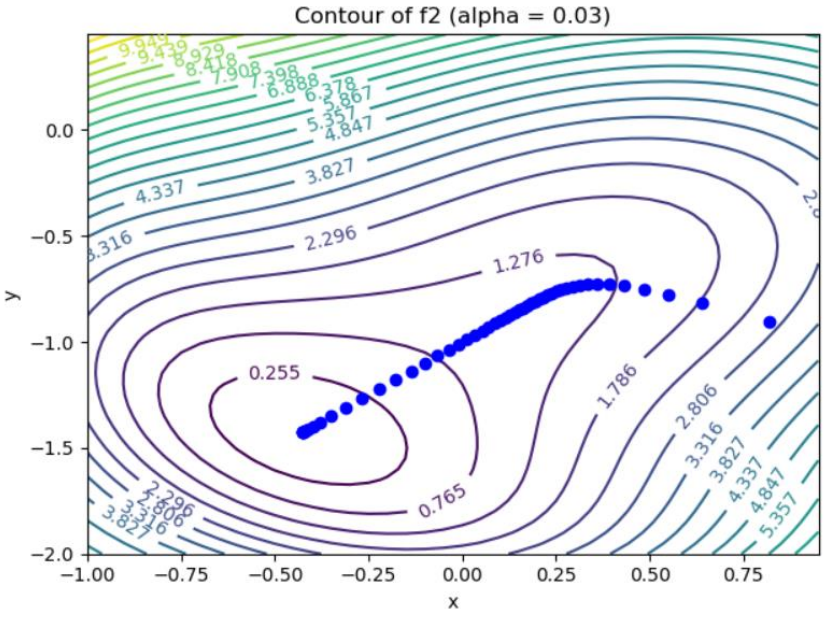
f) Use your gradients to implement a gradient descent algorithm with 50 iterations to find a local minimum for both f_2 and f_3. Show the steps of your algorithm on a contour plot of the function. Start from the point $(1, -1)$ and state the step size you used. Produce separate contour plots for the two functions, using first component of x on the x axis and the second on the y.

Answer:

I. For function f_2:



II. For function f_3:



g) For the two functions f_2 and f_3, discuss the qualitative differences you observe when performing gradient descent with step sizes varying between 0.1 and 1, again starting the point $(1, -1)$. Briefly describe also what happens in the two cases with grossly mis-specified step-sizes (i.e. greater than 1), with a reason to explain the difference in behavior.

Answer:

I. For function f_2:

This function has only one minima, and the performance of gradient descent varies with different step size α :

When $\alpha = 0.02$, the point cannot converge to the minima in 50 steps;

When $\alpha = 0.03$, the point can converge to the minima,

When $\alpha = 1$, the point cannot converge to the minima.

II. For function f_3:

This function has two minima, , and the performance of gradient descent also varies with different step size α :

When $\alpha = 0.05$, the point cannot converge to the minima in 50 steps;

When $\alpha = 0.1$, the point can converge to one minima,

When $\alpha = 0.3$, the point can converge to another minima,

When $\alpha = 1$, the point cannot converge to the minima.

III. Explanation:

When α is small, the point need many iterations to converge to the minima, so it can not reach the minima in only 50 iterations, and if α is too large, it starts oscillating, so it cannot reach the minima, either.