

Machine Learning Exercise Sheet 6

Optimization

Group_369

Fan XUE – fan98.xue@tum.de

Xing ZHOU – xing.zhou@tum.de

Jianzhe LIU – jianzhe.liu@tum.de

December 1, 2021

Problem 4

- a) This statement is false, which can be proved by a counterexample. For that, we need to adopt an inference that the second derivation of a convex function is non-negative. We take $f(x) = x^2$ and $g(x) = -x$, both of which are convex functions, since $\frac{d^2 f(x)}{dx^2} = 2 \geq 0$ and $\frac{d^2 g(x)}{dx^2} = 0 \geq 0$. Then we take the composition of the two functions as $h(x) = g(f(x))$, and we have:

$$\frac{d^2}{dx^2} h(x) = -2 \leq 0.$$

That means $h(x) = g(f(x))$ is non-convex.

- b) This statement is true. We prove it with the definition of convex function. We take $h(x) = g(f(x))$, $\forall x_1, x_2 \in \mathbb{R}$ and $t \in (0, 1)$, since $f(x)$ is convex we have

$$f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2).$$

Since $g(x)$ is non-decreasing and convex, we have

$$g(f(tx_1 + (1-t)x_2)) \leq g(tf(x_1) + (1-t)f(x_2)) \quad (1)$$

$$g(tf(x_1) + (1-t)f(x_2)) \leq tg(f(x_1)) + (1-t)g(f(x_2)). \quad (2)$$

Combine (1) and (2), we have

$$g(f(tx_1 + (1-t)x_2)) \leq tg(f(x_1)) + (1-t)g(f(x_2))$$

$$\Updownarrow$$

$$h(tx_1 + (1-t)x_2) \leq th(x_1) + (1-t)h(x_2),$$

which proves $h(x) = g(f(x))$ is convex.

Problem 5

a) By observing the given function f :

$$f(x_1, x_2) = 0.5x_1^2 + x_2^2 + 2x_1 + x_2 + \cos(\sin(\sqrt{\pi}))$$

we can find that it consists of several convex functions. Which means that when we separately find all the x_n that minimize its own function, we find the \mathbf{x}^* that minimizes the function f .

The gradient of f can be computed by followings:

$$\begin{pmatrix} \frac{\partial f(x_1, x_2)}{\partial x_1} \\ \frac{\partial f(x_1, x_2)}{\partial x_2} \end{pmatrix} = \begin{pmatrix} x_1 + 2 \\ 2x_2 + 1 \end{pmatrix} = 0$$

So minimizer \mathbf{x}^* is:

$$\mathbf{x}^* = \begin{pmatrix} -2 \\ \frac{1}{2} \end{pmatrix}$$

b) 2 steps of gradient descent with leaning rate $\tau = 1$ can be performed by following procedure:

$$\text{step 1: } \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix} = \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \end{pmatrix} - \tau \nabla_{\mathbf{x}} f(\mathbf{x}^{(0)}) = \begin{pmatrix} -2 \\ -1 \end{pmatrix}$$

$$\text{step 2: } \begin{pmatrix} x_1^{(2)} \\ x_2^{(2)} \end{pmatrix} = \begin{pmatrix} x_1^{(1)} \\ x_2^{(1)} \end{pmatrix} - \tau \nabla_{\mathbf{x}} f(\mathbf{x}^{(1)}) = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

Here $\nabla f(\mathbf{x})$ is obtained from **question a)** above, where

$$\frac{\partial f(x_1, x_2)}{\partial x_1} = x_1 + 2$$

$$\frac{\partial f(x_1, x_2)}{\partial x_2} = 2x_2 + 1$$

c) we can try computing two more iteration:

$$\begin{pmatrix} x_1^{(3)} \\ x_2^{(3)} \end{pmatrix} = \begin{pmatrix} -2 \\ 0 \end{pmatrix} - 1 \begin{pmatrix} -2 + 2 \\ 0 + 1 \end{pmatrix} = \begin{pmatrix} -2 \\ -1 \end{pmatrix}$$

$$\begin{pmatrix} x_1^{(4)} \\ x_2^{(4)} \end{pmatrix} = \begin{pmatrix} -2 \\ -1 \end{pmatrix} - 1 \begin{pmatrix} -2 + 2 \\ -2 + 1 \end{pmatrix} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

Obviously, it's an endless loop.....Yet we already know that minimizer \mathbf{x}^* is: $\begin{pmatrix} -2 \\ \frac{1}{2} \end{pmatrix}$.

So the answer is, this gradient descent procedure can never converge to the true minimizer \mathbf{x}^* .

To solve this problem, we must change the learning rate. In our case, it has to decrease to get to the minimizer.

Problem 6

The result of the programming task is appended at the end of the file.

Problem 7

- a) This region S is not a convex region. Because we can find points whose line in between is outside region S . As shown in figure 1, red, green and blue lines are all in this situation.

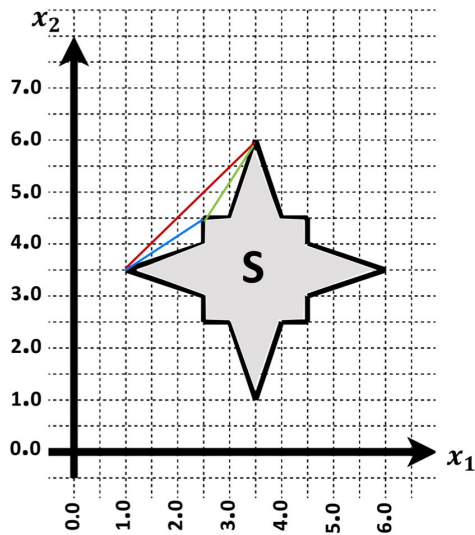


Figure 1: Figure 1

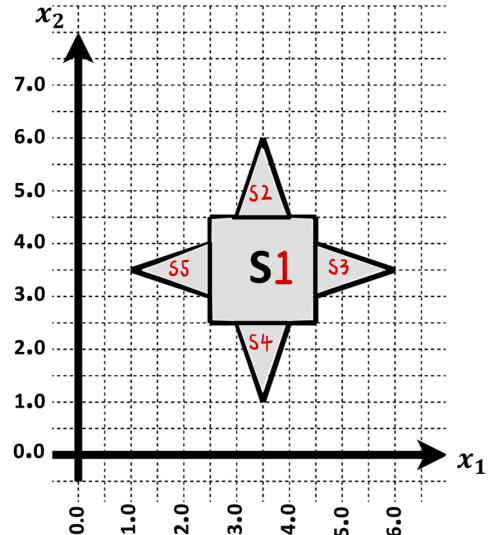


Figure 2: Figure 2

- b) It's obvious that Function F is meaningful only if it's region is convex. So to give this **ConvOpt** algorithm a qualified input, we need to divide the original region S into 5 small regions, which are all convex regions(as shown in figure2).

This way, we can use this **ConvOpt** algorithm to get 5 different results, which are respectively the minimum in their region. Then we choose the minimum among this 5 values, it is the minimum of whole region S .