Optimization and Machine Learning, Fall 2023
Homework 5
(Due Thursday, Jan 11 at 11:59pm (CST))

1. [10 points] [Deep Learning Model]

   (a) Consider a 2D convolution layer. Suppose the input size is $4 \times 64 \times 64 \times$ (channel, width, height) and we use **ten** 3×3 (width, height) kernels with 4 channels input and 4 channels output to convolve with it. Set stride $= 1$ and pad $= 1$. What is the output size? Let the bias for each kernel be a scalar, how many parameters do we have in this layer? [5 points]

   (b) The convolution layer is followed by a max pooling layer with 2×2 (width, height) filter and stride $= 2$. What is the output size of the pooling layer? How many parameters do we have in the pooling layer? [5 points]

   (a) Since the input image is $4 \times 64 \times 64$, so $W = 64, H = 64$.
   And since stride $S = 1$, pad $P = 1$, kernel size $F = 3$,
   so the output size is $W_{conv} = \frac{W + 2P - F}{S} + 1 = 64$, $H_{conv} = \frac{H + 2P - F}{S} + 1 = 64$.
   And since we have 10 kernels, so the output size is $10 \times 64 \times 64$.

   For each convolution kernel, we have $4 \times 3 \times 3 = 36$ parameters.
   And since we have 10 kernels, each kernel has a bias, which is 1 parameter.
   So the total number of parameters is $10 \times (4 \times 3 \times 3 + 1) = 370$.

   So above all, the output size is $10 \times 64 \times 64$, and the total number of parameters is 370.

   (b) Since the output size of the convolution layer is $10 \times 64 \times 64$.
   And for the pooling layer, the filter size is $F' = 2$, stride $S' = 2$,
   so the output size is $W_{pooling} = \frac{W_{conv} - F'}{S'} + 1 = 32$, $H_{pooling} = \frac{H_{conv} - F'}{S'} + 1 = 32$.
   So the output size is $10 \times 32 \times 32$.

   And since the pooling layer is a max pooling layer, so there is no parameter in this layer.

   So above all, the output size is $10 \times 32 \times 32$, and the total number of parameters is 0.

2. [10 points] Use the k-means++ algorithm and Euclidean distance to cluster the 8 data points into $K = 3$ clusters. The coordinates of the data points are:

$$x^{(1)} = (2, 8),\ x^{(2)} = (2, 5),\ x^{(3)} = (1, 2),\ x^{(4)} = (5, 8),$$
$$x^{(5)} = (7, 3),\ x^{(6)} = (6, 4),\ x^{(7)} = (8, 4),\ x^{(8)} = (4, 7).$$

Suppose that initially the first cluster centers is $x^{(1)}$.
To ensure consistent results, please use random numbers in the order shown in the table below. When selecting a center, arrange it in ascending order of sequence number. For example, when the normalized weights of 5 nodes are 0.2, 0.1, 0.3, 0.3, and 0.1, if the random number is 0.3, the selected node is the third one. Note that you don't necessarily need to use all of them.

0.6	0.2	0.5	0.9	0.3

(a) Perform the k-means++ algorithm to initialize other centers and report the coordinates of the resulting centroids. [3 points]

(b) Calculate the loss function

$$Q(r, c) = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{K} r_{ij} ||x^{(i)} - c_j||^2, \tag{1}$$

where $r_{ij} = 1$ if $x^{(i)}$ belongs to the j-th cluster and 0 otherwise. [2 points]

(c) How many more iterations are needed to converge? [3 points] Calculate the loss after it converged. [2 points]

(a) We can calculate the other points' Euclidean distance to $x^{(1)}$ is $D(x^{(i)})$, and the probability of selecting $x^{(i)}$ as the next center is $p(x^{(i)})$, which is proportional to $D(x^{(i)})^2$.
So the $D^2(x^{(i)})$ and $p(x^{(i)})$ are shown in the table below.

point	$x^{(2)}$	$x^{(3)}$	$x^{(4)}$	$x^{(5)}$	$x^{(6)}$	$x^{(7)}$	$x^{(8)}$
$D^2(x^{(i)})$	9	37	9	50	32	52	5
$p(x^{(i)})$	0.05	0.19	0.05	0.26	0.16	0.27	0.03

We randomly sample a point. The random number is 0.6, and since $\sum_{i=2}^{5} p(x^{(i)}) = 0.55 < 0.6$,

$\sum_{i=2}^{6} p(x^{(i)}) = 0.71 > 0.6$, so we choose $x^{(6)}$ as the second class center.

2. Then, we need to choose the third center.
Suppose that for the i-th point $x^{(i)}$, the Euclidean distance for it to $x^{(1)}$ is $D_1(x^{(i)})$, the Euclidean distance for it to $x^{(6)}$ is $D_2(x^{(i)})$.
So the Euclidean distance to the closest center $D(x^{(i)}) = \min(D_1(x^{(i)}), D_2(x^{(i)}))$. So the $D_1^2(x^{(i)}), D_2^2(x^{(i)}), D^2(x^{(i)})$ and $p(x^{(i)})$ are shown in the table below.

point	$x^{(2)}$	$x^{(3)}$	$x^{(4)}$	$x^{(5)}$	$x^{(7)}$	$x^{(8)}$
$D_1^2(x^{(i)})$	9	37	9	50	52	5
$D_2^2(x^{(i)})$	17	29	17	2	4	13
$D^2(x^{(i)})$	9	29	9	2	4	5
$p(x^{(i)})$	0.16	0.50	0.16	0.03	0.07	0.09

We randomly sample a point. The random number is 0.2, and since $p(x^{(2)}) = 0.16 < 0.2$,
$p(x^{(2)}) + p(x^{(3)}) = 0.76 > 0.2$, so we choose $x^{(3)}$ as the third class center.

So above all, the centroids of the second and third cluster is $x^{(6)} = (6, 4)$ and $x^{(3)} = (1, 2)$.

(b)

(c)

3. [10 points] Name 2 deep generation networks. [2 points] Briefly describe the training procedure of a GAN model.(What's the objective function? How to update the parameters in each stage?) [8 points]