Hong Kong University of Science and Technology
COMP 4211: Machine Learning
Fall 2022

Problem Set
Due: 10 November 2022, Thursday, 11:59pm

Important Instructions:

- If an answer requires calculation or derivation, you are expected to show the steps as well in addition to the final answer.
- Your answers must be typewritten (not handwritten) properly for submission. You may use LaTeX (`http://www.latex-project.org/`), Word or other mathematical typesetting software to typeset your answers.
- Your submission must be done electronically in one PDF file via the Canvas course site.
- Late submission will be accepted but with penalty. The late penalty is deduction of one point (out of a maximum of 100 points) for every minute late after 11:59pm. Being late for a fraction of a minute is considered a full minute. For example, two points will be deducted if the submission time is 00:00:34.
- In case the bonus part you did for either one of the programming assignments allows you to submit this problem set late for up to 24 hours without grade penalty, the late submission policy above will no longer be applicable. In other words, no submission 24 hours after the original deadline will be accepted. Please also note that late submission should also be done in Canvas.
- While you may discuss with your classmates on general ideas about solving the problems, your submission should be based on your own independent effort.
- In case you seek help from any person or reference source, you should state it clearly in your submission. Failure to do so is considered plagiarism which will lead to appropriate disciplinary actions. Please refer to the regulations for student conduct and academic integrity on this webpage: `https://registry.hkust.edu.hk/resource-library/academic-standards`.

1. **Linear Regression** (13 points)

   (a) (10 points) Using the method illustrated in slides #6 and #7 of the Linear Regression notes (instead of that in slide #9 using multivariable calculus), derive the least squares estimate in slide #8 for the general case with $d \geq 1$.

   (b) (3 points) Suppose we want to use gradient descent to estimate the solution iteratively instead of the closed-form solution above. Derive the weight update rule for each of the weights.

2. **Logistic Regression** (14 points)

Consider a logistic regression model with K outputs. One way is to define the loss function as in slide #17 of the Logistic Regression notes using the softmax function.

Alternatively, instead of using the softmax function for the K outputs, we can treat each of the K outputs as corresponding to a binary classification problem.

(a) (6 points) Give a suitable loss function for the alternative formulation.

(b) (4 points) Describe one major difference between the two formulations in terms of the K output values.

(c) (4 points) In what way does the difference in part (b) reflect the difference in nature between the classification problems corresponding to the two different formulations?

3. **Feedforward Neural Networks** (14 points)

We consider the two-hidden-layer feedforward neural network discussed in class with its weight update rules summarized in slides #16 and #17 of the notes. Suppose all units in the two hidden layers use the hyberbolic tangent function as their activation functions, i.e.,

$$g(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}.$$

(a) (3 points) Derive and express the tanh activation function in terms of the sigmoid function σ.

(b) (3 points) Derive and express the first derivative of the tanh activation function in terms of the sigmoid function σ.

(c) (4 points) By referring to the weight update rules, explain why it is not good to initialize the weights to large magnitudes.

(d) (4 points) By referring to the weight update rules, explain why it is not good to initialize all the weights to zero.

4. **Convolutional Neural Networks** (20 points)

For each convolutional layer, let (n_i, n_o, h, w, s, p) be a concise way to represent its configuration which corresponds to n_i input channels, n_o output channels (a.k.a. feature maps), one convolution kernel of height h and width w for each feature map, a stride of s, and a padding p.

(a) Suppose we have a convolutional layer with configuration $(128, 64, 7, 7, 2, 2)$ and the image in each input channel is of size 227×227.

i. (3 points) What is the size of each feature map in the convolutional layer?

ii. (3 points) How many parameters are there in the convolutional layer?

(b) To reduce the number of parameters, we now add a new convolutional layer before the original one. The new convolutional layer has configuration $(128, 16, 1, 1, 1, 0)$ and the configuration of the original convolutional layer is modified to $(16, 64, 7, 7, 2, 2)$.

i. (3 points) What is the size of each feature map in the new convolutional layer?

ii. (3 points) How many parameters are there in the new convolutional layer?

iii. (3 points) What is the size of each feature map in the original convolutional layer after its configuration is modified?

iv. (3 points) How many parameters are there in the original convolutional layer after modification?

v. (2 points) By introducing an additional convolutional layer with 1×1 kernels, the number of parameters is reduced. What is the saving in percentage?

5. **Principal Component Analysis** (15 points)

We are given a two-dimensional dataset $\mathcal{S} \subset \mathbb{R}^2$ which consists of six data points:

$$\mathcal{S} = \{\mathbf{x}^{(\ell)}\}_{\ell=1}^{6} = \left\{(2,1)^\top, (3,5)^\top, (4,3)^\top, (5,6)^\top, (6,7)^\top, (7,8)^\top\right\}.$$

(a) (2 points) Calculate the mean vector $\boldsymbol{\mu}$ of $\mathcal{S}$.

(b) (2 points) Subtract the mean vector $\boldsymbol{\mu}$ from the given feature vectors $\mathbf{x}^{(\ell)}$ representing the six data points.

(c) (3 points) Calculate the covariance matrix $\boldsymbol{\Sigma}$ of $\mathcal{S}$.

(d) (5 points) Calculate the eigenvalues of the covariance matrix $\boldsymbol{\Sigma}$.

(e) (3 points) If we apply principal component analysis to project $\mathcal{S}$ to one dimension, calculate the resulting proportion of variance (PoV).

If you wish, you may complete parts (c) and (d) with the help of a Python-based software tool. Other than giving the answers, you should also write down the code to obtain them.

6. **Clustering – Partitional Clustering** (11 points)

We apply the k-medians clustering algorithm to a two-dimensional dataset $\mathcal{S}$ consisting of seven data points which can be represented by the coordinates in a Cartesian plane as follows:

p_1 :	$(0, 5)$
p_2 :	$(1, 3)$
p_3 :	$(2, 4)$
p_4 :	$(6, 2)$
p_5 :	$(7, 0)$
p_6 :	$(8, 3)$
p_7 :	$(9, 1)$

We want to form two clusters C_1 and C_2 and randomly initialize their medians to $m_1 = (4, 2)$ and $m_2 = (11, 3)$.

(a) (2 points) In the first iteration, which data points are assigned to C_1 and C_2?

(b) (2 points) In the first iteration, what are the medians after update?

(c) (2 points) In the second iteration, which data points are assigned to C_1 and C_2?

(d) (2 points) In the second iteration, what are the medians after update?

(e) (3 points) What are the final values of the medians after the algorithm stops?

7. **Clustering – Hierarchical Clustering** (13 points)

We apply agglomerative clustering to a two-dimensional dataset $\mathcal{S}$ using single-link proximity with the Euclidean distance measure. $\mathcal{S}$ consists of five data points which can be represented by the coordinates in a Cartesian plane as follows:

p_1 :	$(1,0)$
p_2 :	$(2,1)$
p_3 :	$(8,8)$
p_4 :	$(9,6)$
p_5 :	$(9,8)$

(a) (2 points) How many merging steps does the algorithm perform before stopping?

(b) (2 points) Which two points are chosen for merging in the first step and what is the distance between them? Let the cluster formed be denoted by C_1.

(c) (2 points) Which two points/clusters are chosen for merging in the second step and what is the distance between them? Let the cluster formed be denoted by C_2.

(d) (2 points) Which two points/clusters are chosen for merging in the third step and what is the distance between them? Let the cluster formed be denoted by C_3.

(e) (2 points) Continuing this process until the last merging step, what is the distance between the two points/clusters merged?

(f) (3 points) If we have to determine the number of clusters in the dataset, what is a natural choice? Explain your answer.