EE 5184 Machine Learning Final Exam

Date: 2021/01/08

The paper is double-sided, 5 pages, consisting of 7 questions. Total 105 points.

In this exam, unless otherwise specified, we denote $\sigma(z) = \frac{1}{1+e^{-z}}$ as the sigmoid function.

1. (5 pts) ReLU Function

Show that

$$\int_{-\infty}^{0} \sigma(z+t)dt = \log(1+e^{z})$$

Hence ReLU function can be interpreted as the sum of infinitely many shifted sigmoid functions.

2. (10 pts) Principle Component Analysis (PCA)

Given m samples $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N \in \mathbb{R}^2$. Suppose

$$\frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i = \boldsymbol{0}, \qquad \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^T = \begin{bmatrix} 12.01 & 1.68 \\ 1.68 & 6.74 \end{bmatrix}$$

(1) **(3 pts)** Find $\frac{1}{N}\sum_{i=1}^{N}\|\boldsymbol{x}_i\|_2^2$.

(2) **(4 pts)** Find the first principle axis after performing PCA on this data set.

(3) **(3 pts)** Denote $\boldsymbol{u}_i$ as the projection of $\boldsymbol{x}_i$ to the first principle axis. Find $\frac{1}{N}\sum_{i=1}^{N}\|\boldsymbol{u}_i\|_2^2$.

3. (5 pts) Convolution Layer

A 6x6 input image of two channels is about to pass through a convolution layer which consists of one filter (kernel). The input image and the filter is illustrated as the following figure.

(1) **(3 pts)** Please show the result (referred as feature map Z) after the input image has passed through this convolution layer. Assume no padding and stride=1.

(2) **(2 pts)** Continuing (1), show the result after we apply 2x2 max-pool to the feature map Z.

Input image

Channel 1

1	-1	1	1	1	1
1	0	0	1	1	-1
-1	1	1	0	1	-1
1	1	-1	-1	0	-1
0	-1	0	1	0	-1
-1	1	1	1	-1	1

Channel 2

1	1	1	-1	1	0
-1	1	1	0	-1	-1
1	-1	-1	1	0	-1
-1	0	1	0	1	-1
0	0	0	0	1	1
0	0	-1	-1	1	-1

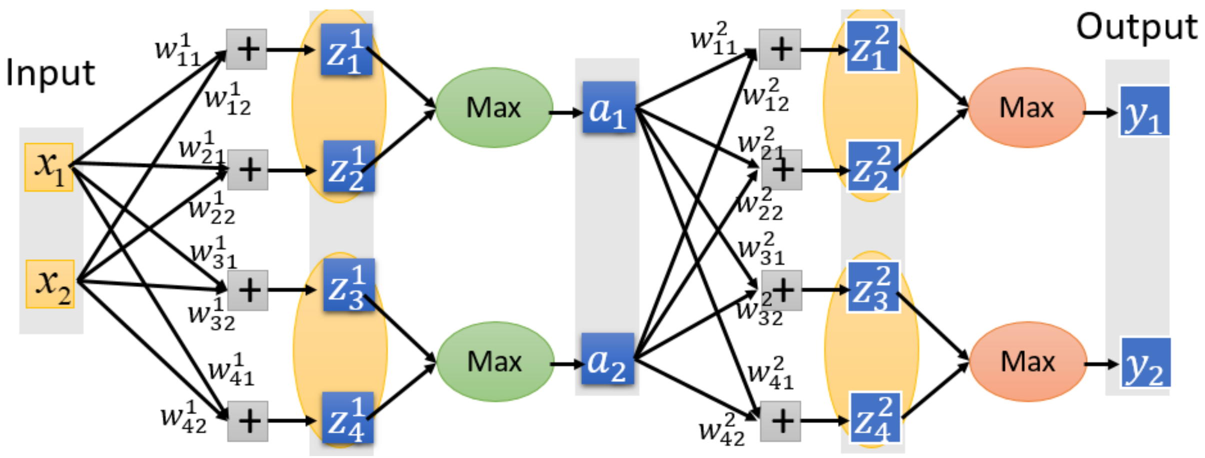
Filter

1	-1	-1
1	-1	1
0	1	0

1	-1	1
-1	1	1
0	-1	1

4. (20 pts) Maxout Network

Consider the following maxout network

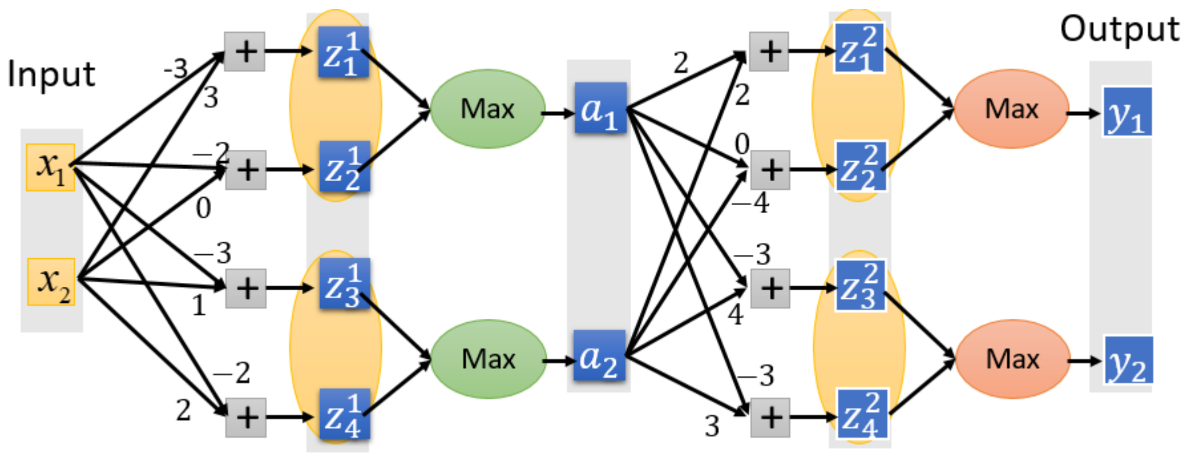


The above neural network can be represented as a function f_θ, namely

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = f_\theta\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right),$$

where parameter θ records all the weights w_{ij}^k.

(1) **(6 pts)** Suppose the weights are initialized as follows:



If $(x_1, x_2) = (1,10)$, please fill out the following table. No derivation required.

Variable	z_1^1	z_2^1	z_3^1	z_4^1	a_1	a_2	z_1^2	z_2^2	z_3^2	z_4^2	y_1	y_2
Value												

(2) **(6 pts)** Continuing (1), if the ground truth is $(\hat{y}_1, \hat{y}_2) = (-8, -1)$, and the L1-loss is adopted, namely

$$\mathrm{L}(\theta) = \left\| \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \end{bmatrix} - f_\theta\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) \right\|_1 = |y_1 - \hat{y}_1| + |y_2 - \hat{y}_2|$$

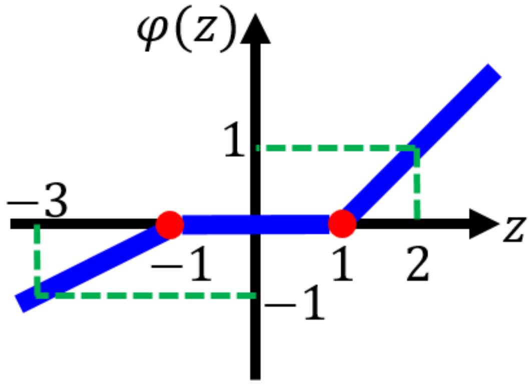
Please compute $\frac{\partial \mathrm{L}}{\partial w_{12}^1}$ and $\frac{\partial \mathrm{L}}{\partial w_{41}^2}$.

(3) **(5 pts)** Let $g_1(x), g_2(x), \ldots, g_k(x)$ be convex functions, prove that

$$\mathrm{h}(\mathrm{x}) = \max\big(g_1(x), g_2(x), \ldots, g_k(x)\big)$$

is also a convex function.

(4) **(3 pts)** Is it possible to implement the following piecewise-linear activation function φ with maxout network? If possible, please show how to implement; If impossible, please prove why it is impossible.



5. (20 pts) Gradient Boosting

Consider the binary classification problem, where we are given training data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ with $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \{\pm 1\}$. Let $\mathcal{F}$ be a collection of binary classifiers, each mapping from $\mathbb{R}^d$ to $\{\pm 1\}$. Given number of epochs $\mathrm{T} \in \mathbb{N}$, suppose we want to find the function

$$\mathrm{g}(\boldsymbol{x}) = \sum_{t=1}^{T} \alpha_t f_t(\boldsymbol{x})$$

Where $f_t \in \mathcal{F}$ and $\alpha_t \in \mathbb{R}$ for all $t = 1, \dots, \mathrm{T}$, by which the aggregated classifier is given by

$$\mathrm{h}(\boldsymbol{x}) = \begin{cases} 1 & \text{, if } g(\boldsymbol{x}) > 0 \\ -1 & \text{, if } g(\boldsymbol{x}) \le 0 \end{cases}$$

Please apply gradient boosting to show how the functions f_t and coefficients α_t are computed with an aim to minimize the following loss function

$$\mathrm{L}(g) = \sum_{i=1}^{N} \log\left(1 + e^{-y_i g(\boldsymbol{x}_i)}\right)$$

6. (20 pts) Expectation Maximization Interpretation behind Semi-Supervised Learning

Given N samples $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N \in \mathbb{R}^m$ as well as their labels $y_1, \dots, y_N \in \{0, 1, \dots, K\}$. Consider the generative model where each sample $\boldsymbol{x}_i$ is generated independently according to Gaussian mixture model *that depends on the label* y_i, as represented by random variable

$$X_i \sim \begin{cases} \sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) & \text{, if } y_i = 0 \\ \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) & \text{, if } y_i = k \neq 0 \end{cases}$$

where $\pi_1 + \cdots + \pi_K = 1$, and $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, with probability density function

$$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^m |\boldsymbol{\Sigma}|}} exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right)$$

We would like to apply Expectation Maximization algorithm to find the maximum likelihood estimation of parameters $\theta = \{(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$.

(1) **(16 pts)** Please write down the E-step and M-step and show that the parameters are updated from $\theta^{(t)} = \left\{\left(\pi_k^{(t)}, \boldsymbol{\mu}_k^{(t)}, \boldsymbol{\Sigma}_k^{(t)}\right)\right\}_{k=1}^K$ to $\theta^{(t+1)} = \left\{\left(\pi_k^{(t+1)}, \boldsymbol{\mu}_k^{(t+1)}, \boldsymbol{\Sigma}_k^{(t+1)}\right)\right\}_{k=1}^K$ in the following form:

$$\pi_k^{(t+1)} = \frac{N_k + \sum_{i:y_i=0} \delta_{ik}^{(t)}}{N}$$

$$\mu_k^{(t+1)} = \frac{\sum_{i:y_i=k} \boldsymbol{x}_i + \sum_{i:y_i=0} \delta_{ik}^{(t)} \boldsymbol{x}_i}{N_k + \sum_{i:y_i=0} \delta_{ik}^{(t)}}$$

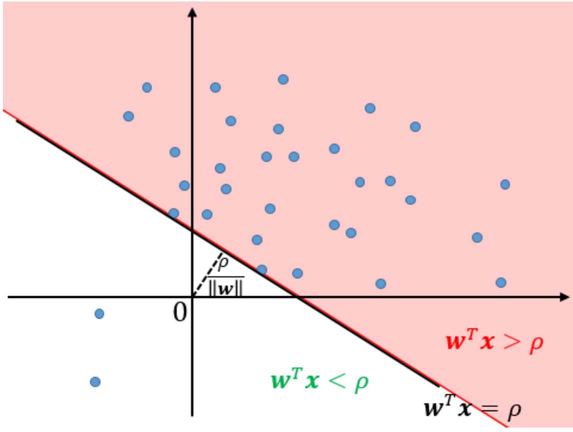
$$\boldsymbol{\Sigma}_k^{(t+1)} = \frac{\sum_{i:y_i=k}\left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(t+1)}\right)\left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(t+1)}\right)^T + \sum_{i:y_i=0} \delta_{ik}^{(t)}\left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(t+1)}\right)\left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(t+1)}\right)^T}{N_k + \sum_{i:y_i=0} \delta_{ik}^{(t)}}$$

where $N_k = \sum_{i:y_i=k} 1$ is the number of samples in class k. Please show your derivations.

(2) **(4 pts)** What is the closed form expression of $\delta_{ik}^{(t)}$? Please show your derivations.

7. (25 pts) One-Class Support Vector Machine

Given unlabeled training data $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N \in \mathbb{R}^d$, the goal of *one-class SVM* is to find the "smallest" half space $S = \{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{w}^T\boldsymbol{x} \geq \rho\}$ that contains the most data points, as illustrated below.



Formally speaking, with a given hyper-parameter $0 < \nu < 1$, the *one-class SVM* aims to solve the following optimization problem

$$\begin{aligned}
&\text{minimize} && \frac{1}{2}\|\boldsymbol{w}\|^2 + \frac{1}{\nu N}\sum_{i=1}^{N} \xi_i - \rho \\
&\text{subject to} && \begin{matrix} \boldsymbol{w}^T\boldsymbol{x}_i \geq \rho - \xi_i \\ \xi_i \geq 0 \end{matrix}, \quad \forall i = 1, \dots, N \\
&\text{variables} && \boldsymbol{w} \in \mathbb{R}^d, \rho \in \mathbb{R}, \xi_1, \dots, \xi_N \in \mathbb{R}
\end{aligned}$$

Which can be rewritten in the form of **primal problem**

$$\begin{aligned}
&\text{minimize} && f(\mathbf{w}, \rho, \boldsymbol{\xi}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \frac{1}{\nu N}\sum_{i=1}^{N} \xi_i - \rho \\
&\text{subject to} && \begin{matrix} g_{1,i}(\mathbf{w}, \rho, \boldsymbol{\xi}) = \rho - \xi_i - \boldsymbol{w}^T\boldsymbol{x}_i \leq 0 \\ g_{2,i}(\mathbf{w}, \rho, \boldsymbol{\xi}) = -\xi_i \leq 0 \end{matrix}, \quad \forall i = 1, \dots, N \\
&\text{variables} && \boldsymbol{w} \in \mathbb{R}^d, \rho \in \mathbb{R}, \xi_1, \dots, \xi_N \in \mathbb{R}
\end{aligned}$$

(1) **(7 pts)** Show that the **dual problem** of *one-class SVM* can be written as

$$\text{maximize} \quad \theta(\boldsymbol{\alpha}) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j \boldsymbol{x}_i^T \boldsymbol{x}_j$$

$$\text{subject to} \quad \sum_{i=1}^{N} \alpha_i = 1$$

$$\text{variables} \quad 0 \le \alpha_i \le \frac{1}{\nu N}, \ \mathrm{i} = 1, \dots, \mathrm{N}$$

Let $(\overline{\boldsymbol{w}}, \bar{b}, \bar{\boldsymbol{\xi}})$ be a primal optimal solution, $\overline{\boldsymbol{\alpha}}$ be a dual optimal solution.

(2) **(8 pts)** Which of the following statements are true? Please justify your answers.

i. $\mathrm{f}(\overline{\boldsymbol{w}}, \bar{\rho}, \bar{\boldsymbol{\xi}}) = \theta(\overline{\boldsymbol{\alpha}})$.

ii. If $\overline{\boldsymbol{w}}^T \boldsymbol{x}_i < \bar{\rho}$, then $\bar{\alpha}_i = 1/(\nu \mathrm{N})$.

iii. If $\overline{\boldsymbol{w}}^T \boldsymbol{x}_i > \bar{\rho}$, then $\bar{\alpha}_i = 0$.

iv. $\bar{\xi}_i = max(\bar{\rho} - \overline{\boldsymbol{w}}^T \boldsymbol{x}_i, 0)$.

(3) **(5 pts)** Show that the optimal bias $\bar{\rho}$ can be expressed in terms of the optimal weighting vector $\overline{\boldsymbol{w}}$ as

$$\bar{\rho} = \underset{\rho \in \mathbb{R}}{\operatorname{argmin}} \left(\frac{1}{\nu \mathrm{N}} \sum_{i=1}^{N} max(\rho - \overline{\boldsymbol{w}}^T \boldsymbol{x}_i, 0) - \rho \right)$$

(Hint: Rewrite the primal problem as an unconstrained optimization problem over variables $\boldsymbol{w}, \rho$*)*

(4) **(5 pts)** Suppose $\mathrm{k} - 1 < \nu \mathrm{N} < \mathrm{k}$, for which k is a positive integer, then how many among the N data points $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$ satisfy $\overline{\boldsymbol{w}}^T \boldsymbol{x} \ge \bar{\rho}$? In other words, please compute $\sum_{i:\ \overline{\boldsymbol{w}}^T \boldsymbol{x}_i \ge \bar{\rho}} 1$. Please show your derivations.

(Hint: Solve the minimization problem in (3) by taking upper-derivatives)