ECE-GY 6143: Introduction to Machine Learning
Final Exam , Spring 2021

Name: **ID:**

Answer ALL questions. Exam is closed book. No electronic aids. However, you are permitted two cheat sheets, two sides each sheet. Any content on the cheat sheet is permitted. Part marks are given. If you do not remember a particular python command or its syntax, use pseudo-code and state what syntax you are assuming.

Please use a separate page for each question. Clearly write the question numbers on top of the pages. When you submit, please order your pages by the question numbers.

Each student is required to open Zoom and turn on their video, making sure the camera captures your hands and your computer screen/keyboard. The whole exam will be recorded. To clearly capture the video of exam taking, it is recommended that: A student can use an external webcam connected to the computer, or use another device (smartphone/tablet/laptop with power plugged in). Adjust the position of the camera so that it clearly captures the keyboard, screen and both hands. During the exam, you should keep your video on all the time. If there is anything wrong with your Zoom connection, please reconnect ASAP. If you cannot, please email ASAP.

During the exam, if you need to use the restroom, please send a message using the chat function in Zoom, so we know you left.

At 1:30PM, submit a single PDF to Gradescope, under the assignment named "Final Exam". You can use Adobe Scan APP/iPhone Notes APP/etc to scan your answers as a PDF file. If you have never used your phone to scan a document, give it a try now.

DEADLINE for submission is in 1:45PM. Before you leave the exam, it's your responsibility to make sure all your answers are uploaded.If you have technical difficulty and cannot upload till 1:40PM, email a copy to your proctor and the professor.

Pei Liu: peiliu@nyu.edu
Ashutosh Srivastava: as12738@nyu.edu
Vishaal Ranjan: vr1089@nyu.edu
Rahul Petkar: rsp431@nyu.edu

Best of luck!

1. (20 points) *Maximum Margin Classifier and Support Vector Machine.* You are given four training samples.

i	1	2	3	4
x_{i1}	0	0	1	2
x_{i2}	0	2	-1	2
y_i	-1	1	-1	1

where x_{i1} and x_{i2} are the features and y_i is the label of the two classes.

(a) (10 points) Is it possible to use a linear function $x_{i1}+x_{i2}+b=0$ to perfectly separate the two classes for some parameter b? If it is possible, what is the value of b that maximize the margin? What is the maximum margin?

(b) (10 points) Consider an SVM classifier,

$$\widehat{y} = \begin{cases} 1 & \text{if } z > 0, \\ -1 & \text{if } z \leq 0, \end{cases} \qquad z = \sum_{i=1}^{N} \alpha_i y_i K(x, x_i) + b,$$

for the kernel function,

$$K(\mathbf{x}, \mathbf{x}_i) := \begin{cases} 1 & \text{if } \|\mathbf{x} - \mathbf{x}_i\|^2 \leq r^2, \\ 0 & \text{if } \|\mathbf{x} - \mathbf{x}_i\|^2 > r^2, \end{cases}$$

where $r > 0$ is some parameter and $\|\mathbf{w}\|^2$ denotes the squared norm $\|\mathbf{w}\|^2 = w_1^2 + w_2^2$. Find parameters r, α_i and b such that the SVM classifier makes no errors on the training data. Use at most two non-zero values of α_i. *(Hint: You should be able to find the parameters directly, instead of solving an optimization.)*

2. (20 points) *Neural Networks.* Consider a neural network, with $N_i = 2$ input units, $\mathbf{x} = (x_1, x_2)$, $N_h = 3$ hidden units and one output unit for binary classification:

$$z_j^{\text{H}} = \sum_{k=1}^{N_i} W_{jk}^{\text{H}} x_k + b_j^{\text{H}}, \quad u_j^{\text{H}} = \begin{cases} 1 & \text{if } z_j^{\text{H}} > 0 \\ -1 & \text{if } z_j^{\text{H}} \leq 0, \end{cases} \qquad j = 1, \ldots, N_h$$

$$z^{\text{O}} = \sum_{k=1}^{N_h} W_k^{\text{O}} u_k^{\text{H}} + b^{\text{O}}, \quad \widehat{y} = \begin{cases} 1 & \text{if } z^{\text{O}} > 0 \\ 0 & \text{if } z^{\text{O}} \leq 0. \end{cases}$$

(a) (10 points) Suppose that

$$\mathbf{W}^{\text{H}} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{b}^{\text{H}} = \begin{bmatrix} -2 \\ -1 \\ 1 \end{bmatrix}$$

For each hidden unit $j = 1, 2, 3$, draw the regions of inputs (x_1, x_2) where $u_j^{\text{H}} = 1$.

(b) (10 points) Find a vector of output weights $\mathbf{W}^{\text{O}}$ and bias b^{O} such that:

- $\mathbf{x} = (0, 0)$, $(2, 0)$, $(1, 2)$ are classified as $\widehat{y} = 0$; and
- $\mathbf{x} = (1, 0.5)$ is classified as $\widehat{y} = 1$.

3. (15 points) *Backpropagation.* Consider the model with D-dimensional inputs $\mathbf{x} = (x_1, \ldots, x_D)$ one-dimensional outputs $\widehat{y}$, given by

$$\widehat{y} = \sum_{\ell=1}^{L} \mathrm{e}^{B_\ell z_\ell},$$

$$z_\ell = \sum_{j=1}^{D} x_j A_{j\ell}, \quad l = 1, \ldots, L,$$

with parameters $A_{j\ell}$ and B_ℓ. The loss function is $J = \sum_{i=1}^{N} (y_i - \widehat{y}_i)^2$, where the number of training sample is N.

(a) (5 points) Draw the computation graph between the data, parameters and loss function.

(b) (5 points) Show how to compute $\partial J / \partial A_{j\ell}$ from $\partial J / \partial z_{i\ell}$.

(c) (5 points) Show how to compute $\partial J / \partial B_\ell$.

4. (15 points) *Convolutional neural network*. Answer the following questions for a CNN that processes RGB color images with the following input size and two layers.

- `Input`: 256×256 RGB image
- `Conv1`: 2D convolution, $K = 9 \times 9$ filters, $N_o = 12$ output channels, valid mode.
- `MaxPool1`: 2D max pooling, $K = 4 \times 4$ pool size, horizontal and vertical stride $s = 2$.

(a) (5 points) What are the dimensions of the input and the outputs for the first two layers for a mini-batch of 100 images.

(b) (5 points) How many parameters are there in the `Conv1` and `MaxPool1` layers?

(c) (5 points) Describe a kernel that can detect the difference of the image in the R and G channel, but ignores the B channel (Assume channel index R=1, G=2 and B=3).

5. (15 points) *PCA*. Considering the following data matrix, where your data sample is in two dimensional space.

$$X = \begin{bmatrix} 4 & 1 \\ 2 & 3 \\ 5 & 4 \\ 1 & 0 \end{bmatrix}$$

We want to represent the data in one dimensional space using PCA.

(a) (5 points) Calculate the covariance matrix. Show your work.

(b) (5 points) Assume the covariance matrix has eigenvector $v_1 = [1 \quad 1]^T$ with eigenvalue $\lambda_1 = 16$ and $v_2 = [1 \quad -1]^T$ with eigenvalue $\lambda_2 = 4$. Find the projection of $x_1 = [4 \quad 1]^T$ in the direction of maximum variance.

(c) (5 points) Calculate the total variance of all the data sample in the direction of maximum variance.

6. *K-means.* You are given three data samples:

i	1	2	3
x_{i1}	-1	1	4
x_{i2}	0	0	4

(a) (8 points) We want to have $K = 2$ cluster. After K-mean finishes on the above training data, what are the centers of the two clusters? What is the equation for the boundary of the two clusters?

(b) (7 points) Suppose you want to use clustering for outlier detection. You find cluster means $\boldsymbol{\mu}_i$, $i = 1, \ldots, K$ on the training data. Then, given a new data $\mathbf{x}$ and a threshold t, you declare $\mathbf{x}$ an outlier if $\|\mathbf{x} - \boldsymbol{\mu}_i\| \geq t$ for all i. Complete the following function to implement the outlier detection on a matrix of data `X`. The output is `out[i]=1` if the sample `X[i,:]` is an outlier, and `out[i]=0` otherwise. You must specify the other inputs of your function. Avoid for loops for full credit.

```
def outlier_detect(X, ...):
    ...
    return out
```