Introducing to Machine Learning (CS 419M) **Midterm Exam**
Computer Science and Engineering **20-02-2023 Friday, Total time 2 hours**
Indian Institute of Technology Bombay

Roll Number: _____

Instructions

1. It is an OPEN BOOK examination.
2. This paper has four questions. The maximum marks is 30.
3. Write your name and roll number on the top of the question paper.
4. All your answers must be in the place provided. Answers written elsewhere will NOT be graded.
5. There will be partial credits for subjective questions, if you have made substantial progress towards the answer. However there will be NO credit for rough work.
6. Do not attach the rough work with the question paper
7. Total time for the examination is 2 hours.

Signature: _____

1. **1.a** Consider the dataset $\{(\mathbf{x}_i, y_i)\}_{i\in\mathcal{D}}$ having N samples where each $\mathbf{x}_i \in \mathbb{R}^d$ is sampled from a non-degenerate continuous albeit unknown distribution. Consider

$$X = \begin{bmatrix} \mathbf{x}_1^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix} \tag{1}$$

Note that $\mathbf{x}_i^T$ is a row vector. Assuming $d < N$, what is the probability that $X^T X$ is invertible?

1.a	/1	

Answer: - 1

1.b Which of the following loss functions would *always* lead to differentiable gradients during gradient descent? Assume that $|f(\mathbf{x})|$ and $|y|$ is finite for all points. (More than one answer can be correct)

i. Mean-Squared Error: $\sum_i (y_i - f(\mathbf{x}_i))^2$
ii. Mean Absolute Error: $\sum_i |y_i - f(\mathbf{x}_i)|$
iii. Log-Cosh Loss: $\sum_i \log((e^{(y_i - f(\mathbf{x}_i))} + e^{-(y_i - f(\mathbf{x}_i))})/2)$
iv. Inverse Loss: $\sum_i -\frac{1}{(y_i - f(\mathbf{x}_i))^2}$

1.b	/1	

Answer:- i,iii

1.c A student was training a regression model on a provided dataset. The dataset was split into training data and testing data. The model achieved a very high accuracy on the training data but performed very poorly on the testing data. What could be some possible corrections the student can make before re-training? (Select all that apply)

i. Decreasing the regularisation of the model parameters
ii. Reduce number of model parameters
iii. Increase number of model parameters
iv. Introducing K-fold cross-validation
v. Training for more epochs.

1.c	/2	

Answer:- ii,iv

1.d Consider the same setup as is in the next question. Now, let us assume that by mistake, we multiply the trained parameter vector $\mathbf{w}$ with 2 during test. Then, under which approach the set of predicted points must not change?

i. A1
ii. A2
iii. A1 and A2
iv. Both A1 and A2 will change.

1.d	/1	

Answer: ii

1.e Consider a binary classification task, where we trained a linear support vector machine without bias. Now we follow two approaches to predict the class of an instance.

A1 Class $y = 1$ if $\mathbf{w}^\top \mathbf{x} > 1$ and $y = -1$, otherwise

A2 Sort all instances based on the value of $\mathbf{w}^\top \mathbf{x}$ in the decreasing order and return top 10% points from the top.

Define $S(A1)$ and $S(A2)$ to be the instances predicted as $y = +1$ using A1 and A2 respectively. Assume $S(A1)$ is non empty. Mark the correct option.

i. $S(A1) \cap S(A2)$ will be always non-empty.

ii. $S(A1) \cap S(A2)$ will be always non-empty only if the classes are separable.

iii. $S(A1) \cap S(A2)$ will be always non-empty if the classes are separable.

iv. None of the above

1.e /1

Answer: iii

1.f Suppose that we generate examples (x, y) with $x \in \mathbb{R}$ so that $y = \sin(\exp(x)) + \epsilon$ where ϵ is drawn from $\mathcal{N}(0, 1)$. Next, we use a subset of data S to fit two models $M1$: $y = w\,x + b$ and $M2$: $y = \cos(\exp(w\,x) + b)$ using simple linear regression. Which of the following statements are false?

i. If we increase the size of S, then bias of the model $M1$ will decrease

ii. If we increase the size of S, then bias of the model $M2$ will decrease

iii. Given a fixed S, the bias of $M1$ may be lower than $M2$

iv. Variance will reduce for both $M1$ and $M2$ with increase of the size of S

1.f /3

Answer:- i,iii

1.g Assume a 2 class classification problem with input features $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^d$ and the class labels $y \in \mathcal{Y} = \{-1, +1\}$. We have three kinds of dataset namely (i) training data $\{(\mathbf{x}_i, y_i)\}_{i \in D_{\text{Train}}}$, (ii) validation data $\{(\mathbf{x}_i, y_i)\}_{i \in D_{\text{Valid}}}$ and (iii) test data $\{(\mathbf{x}_i, y_i)\}_{i \in D_{\text{Test}}}$ with $|D_{\text{Valid}}| << |D_{\text{Train}}|$.

We are given that the percentage of samples with label $+1$ in $D_{\text{Train}}, D_{\text{Valid}}, D_{\text{Test}}$ is $\eta_{\text{Train}}, \eta_{\text{Valid}}, \eta_{\text{Test}}$ respectively. Which of the following are favourable, assuming we want to fit a constant model, i.e. $\forall x, h(x) = c$? (Select all that apply)

i. $|\eta_{\text{Train}} - \eta_{\text{Test}}| > |\eta_{\text{Valid}} - \eta_{\text{Test}}|$

ii. $|\eta_{\text{Train}} - \eta_{\text{Test}}| < |\eta_{\text{Valid}} - \eta_{\text{Test}}|$

iii. $(\eta_{\text{Valid}} - 0.5)(\eta_{\text{Test}} - 0.5) < 0$

iv. $(\eta_{\text{Valid}} - 0.5)(\eta_{\text{Test}} - 0.5) > 0$

1.g /1

Answer:- i,iv

1.h Consider a binary classification problem to be solved using a probablistic method called logistic regression. Here, we have

$$\Pr(y = +1|\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^\top \mathbf{x}}} \tag{2}$$

wherein we use a decision rule such that if $\Pr(y = +1|\mathbf{x}) > \tau$, then we classify it in class 1, else in class 0, for some cutoff τ. Let $\hat{\mathbf{w}}, \hat{\tau}$ be the parameters that maximise accuracy on training data (computed using maximum likelihood estimation) and $\mathbf{w}^*, \tau^*$

be the parameters that maximise accuracy on testing data. If we seek to minimise the number of false class 0 classifications (the number of samples in the test data which are in class 1 but are misclassified as class 0), then which of the following are NOT favourable? (Select all that apply)

i. $\hat{\mathbf{w}}^T\mathbf{w}^* > 0$
ii. $\hat{\mathbf{w}}^T\mathbf{w}^* < 0$
iii. $\tau^* > \hat{\tau}$
iv. $\tau^* < \hat{\tau}$

1.h /2

Answer:- ii,iii

2. 2.a In this problem, we consider 2 class classification using Support Vector Machines.

All datasets are not always linearly separable. Hence, we must transform them such that they become linearly separable. For instance, consider the dataset in Figure 1.

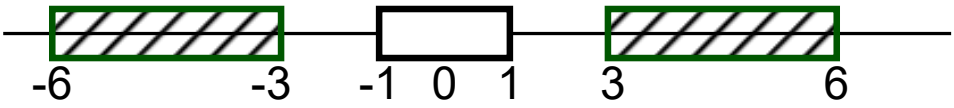

Figure 1: SVM Dataset

This data is not linearly separable. However, we can transform the data as $f(x) = x^2$ or to project it on a higher dimension, $f(x) = (x, x)$. Both these transforms make the data linearly separable.

Consider the dataset $\{(\mathbf{x}_i, y_i)\}_{i\in\mathcal{D}}$ having 2^d samples where each $\mathbf{x}_i \in \{0,1\}^d$. Each data point is a vector in d dimensions, with each coordinate being either 0 or 1 only, thus there being a total of 2^d distinct data points. Now, $y_i = 1$ if $\mathbf{x}_i$ has an odd number of coordinates equalling 1, else $y_i = -1$. For instance, $\mathbf{x}_i = \begin{bmatrix}1,0,1,1,0\end{bmatrix}^T$ has $y_i = 1$

Suggest any suitable transform (such a transform should be implementable; hence, use transforms utilising ReLU and matrix multiplication and addition only) to a minimal larger dimensional space such that this dataset becomes linearly separable.

2.a /3

Inductive approach, Consider 2 dimensional. We can take a transform

$$ReLU(\begin{bmatrix}1 & -1\\ -1 & 1\end{bmatrix} X), X = \begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

This is linearly separable and all class 1 points are at (1,1) and class 0 points are at (0,0). Hence, we can consider the 2nd coordinate of each of these points and add another dimension. Keep doing this inductively till n dimensions.

2.b State True or False. Any dataset containing a finite number of samples, each being a finite dimensional vector and having an assigned labelled class(either +1 or -1, for instance), can always be projected into a higher dimensional space wherein it will be linearly separable. (No justification required)

2.b /1

True. (Create a dimension of N*d, where N is number of samples in dataset and d is dimension of each data point)

3. You are given an input array $\boldsymbol{X} \in \mathbb{R}^D$. We apply a transformation on $\boldsymbol{X}$ using the following function:

```
import numpy as np

def cumulative_sum(X):
    #creates an array Op of the same shape as X. Op is filled with 0s
    Op = np.zeros(X.shape)
    total_till_now =0
    for i in range(len(Op)):
        total_till_now =total_till_now +X[i]
        Op[i] +=total_till_now
    return Op
```

Here are some example inputs and outputs obtained by running the above code:

```
>>> X = np.array([1,1,1,1,1])
>>> cumulative_sum(X)
array([1., 2., 3., 4., 5.])


>>> X = np.array([2,6,1,4,3])
>>> cumulative_sum(X)
array([ 2., 8., 9., 13., 16.])


>>> X = np.array([2,-6,1,4,-3])
>>> cumulative_sum(X)
array([ 2., -4., -3., 1., -2.])
```

You are now asked to do the following:

- Design linear algebraic transformations which generates the same output as `cumulative_sum` for any given $\boldsymbol{X}$
- Your goal is to propose a vectorized alternative to `cumulative_sum`, which can be implemented without any for loops.
- Do not write any code. Instead, describe your logic clearly in English.

b	/3	

Multiply with an upper triangular matrix of 0/1 s

4. Questions on SVM

SVM Formulation for general non-separable case

The optimization problem is specified as:

$$(\mathbf{w}^*, b^*, \boldsymbol{\xi}^*) = \underset{\mathbf{w},b,\xi}{\operatorname{argmin}} \lambda\|\mathbf{w}\|^2 + \sum_{i\in\mathcal{D}} \xi_i$$
$$\text{where } y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \tag{3}$$

The Lagrangian dual function $g(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is given by

$$g(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\mathbf{w},b,\boldsymbol{\xi}} \lambda\|\mathbf{w}\|^2 + \sum_{i\in\mathcal{D}} \left(\xi_i - \alpha_i(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1 + \xi_i) - \beta_i\xi_i\right) \tag{4}$$

The dual problem then is

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} g(\boldsymbol{\alpha},\boldsymbol{\beta}) \tag{5}$$

$$s.t. \alpha_i \geq 0, \beta_i \geq 0 \ \ \forall i \in \mathcal{D} \tag{6}$$

First order optimality conditions on $g(\boldsymbol{\alpha},\boldsymbol{\beta})$ give us the following:

$$\frac{\partial g}{\partial \mathbf{w}} = 0 \implies \mathbf{w}^* = \sum_{i\in\mathcal{D}} \frac{\alpha_i y_i \mathbf{x}_i}{2\lambda} \tag{7}$$

$$\frac{\partial g}{\partial b} = 0 \implies \sum_i \alpha_i y_i = 0 \tag{8}$$

$$\frac{\partial g}{\partial \xi_i} = 0 \implies \alpha_i + \beta_i = 1 \tag{9}$$

Substituting the optimal values into g we rewrite the dual problem as

$$\max_{\boldsymbol{\alpha}} \sum_i \alpha_i - \frac{1}{4\lambda} \sum_{i\in\mathcal{D}} \sum_{j\in\mathcal{D}} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j \tag{10}$$

$$s.t.\ 0 \leq \alpha_i \leq 1 \ \text{ and } \sum_i \alpha_i y_i = 0 \ \ \forall i \in \mathcal{D} \tag{11}$$

At the optimum, we have:

$$\xi_i^* \beta_i^* = 0 \tag{12}$$

$$\alpha_i^* (y_i (\mathbf{w}^{*\top} \mathbf{x}_i + b^*) - 1 + \xi_i^*) = 0 \tag{13}$$

During inference, for a given $\mathbf{x}$, the predicted label is given by $\text{sign}(w^{*\top}\mathbf{x} + b^*)$.

4.a First, we consider the hard margin separabale Support Vector Machine. For such a setup, we have

$$y_i(\mathbf{w}^T \mathbf{x}_i + b) \geq 1 \tag{14}$$

This is the canonical representation of the decision hyperplane. In the case for which the equality holds, the constraints are said to be active. Let the distance of such points from the hyperplane be ρ. First, show that

$$\frac{1}{\rho^2} = \|\mathbf{w}\|^2 \tag{15}$$

4.a	/1	

$$\rho = \frac{y_i(\mathbf{w}^T \mathbf{x}_i + b_i)}{|\mathbf{w}|}$$

$$\frac{1}{\rho^2} = \|\mathbf{w}\|^2$$

4.b The optimization problem for the hard margin SVM can be expressed as

$$\underset{\mathbf{w},b}{\text{argmin}} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{subject to the constraints in Eq. (14).} \tag{16}$$

We solve the above using the dual of it with Lagrange multipliers $\{\alpha_i\}$ by representing the dual function $g(\boldsymbol{\alpha},\boldsymbol{\beta})$ similar (not exactly– check the ALERT below) to what is described in the box above titled "SVM Formulation for general non-separable case". Say α_i^* is the value of α_i at the optimal point of the Lagrangian function g. Now, show that

$$||\mathbf{w}^*||^2 = \sum_{i\in\mathcal{D}} \alpha_i^* \tag{17}$$

where $\mathbf{w}^*$ is the solution of the hard margin SVM in Eq. (16). (ALERT: Do not blindly apply the dual formulation in Eq. (3), since it is for non separable plus soft margin setup. Hence, you must modify it for the hard margin separable case)

4.b /4

From Slater's condition, we have

$$\alpha_i^*(y_i(\mathbf{w}^\top \mathbf{x}_i + b) - 1) = 0$$

This gives,

$$\sum_{i\in\mathcal{D}} \alpha_i^* = \sum_i \mathbf{w}^\top(\alpha_i^* y_i \mathbf{x}_i) + \sum_i (\alpha_i^* b y_i)$$

Since $\sum(\alpha_i^* y_i) = 0$ We have

$$\begin{aligned}\sum_{i\in\mathcal{D}} \alpha_i^* &= \sum_i \mathbf{w}^\top(\alpha_i^* y_i \mathbf{x}_i) \\ &= \mathbf{w}^\top \sum_i (\alpha_i^* y_i \mathbf{x}_i) \\ &= \|\mathbf{w}\|^2 (\text{for } \lambda = \frac{1}{2}, \xi_i = 0)\end{aligned}$$

4.c Now, continuing with the dual representation, let L represent the maximal value we obtain by maximising g, that is

$$L(\boldsymbol{\alpha}^*) = \sum_i \alpha_i^* - \frac{1}{2}\sum_{i\in\mathcal{D}}\sum_{j\in\mathcal{D}} \alpha_i^* \alpha_j^* y_i y_j \mathbf{x}_i^\top \mathbf{x}_j. \tag{18}$$

Verify that the optimal value $||\mathbf{w}^*|| = \sqrt{2L(\boldsymbol{\alpha})}$

4.c /2

Setting derivative of the Lagrangian dual function wrt $\mathbf{w}$ to 0

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

Hence, using this and the result from part (a) and (b),

$$\begin{aligned}2L(\boldsymbol{\alpha}) &= 2\sum_i \alpha_i - \sum_{i\in\mathcal{D}}\sum_{j\in\mathcal{D}} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j \\ &= 2\|\mathbf{w}\|^2 - \|\mathbf{w}\|^2 = \|\mathbf{w}\|^2\end{aligned}$$

4.d Now consider the problem of classification using soft margin SVM and a hinge loss function with regularisation. Hence, the optimisation problem becomes

$$\min_{\mathbf{w}} L(\mathbf{w}) = \min_{\mathbf{w}} \sum_{i\in\mathcal{D}} \lambda\|\mathbf{w}\|^2 + \text{ReLU}(1 - y_i \mathbf{w}^T \mathbf{x}_i) \tag{19}$$

Where $\mathcal{D}$ is the dataset. Now, suppose your dataset had only one point, say $(\mathbf{x}, y)$, and that the optimal parameter is $\mathbf{w}^*$

$$\mathbf{w}^* = \underset{\mathbf{w}}{\text{argmin}}\, \lambda\|\mathbf{w}\|^2 + \text{ReLU}(1 - y\mathbf{w}^T \mathbf{x}) \tag{20}$$

Prove that

$$L(\mathbf{w}^*) \geq \min\left\{\frac{\lambda}{\mathbf{x}^T\mathbf{x}}, \frac{1}{2}\right\} \tag{21}$$

4.d /4

$$L(\mathbf{w}^*) \geq \max_{\alpha\in[0,1]} \alpha - \frac{\alpha^2 \mathbf{x}^T\mathbf{x}}{4\lambda} \geq \min\{\frac{\lambda}{\mathbf{x}^T\mathbf{x}}, \frac{1}{2}\}$$

Note: Can also be solved taking cases over RELU

Total: 30