Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

Instructions: *Total:* **100 marks**

1. Total duration: **3 hours**. Please write your name, roll number, department on **all pages**.
2. This booklet has 10 pages (8 pages + 2 pages for rough work). No part of your answers should be on pages designated for rough work. Additional rough sheets may be provided if needed.
3. Write/mark your answers clearly in the provided space. Please keep your answers precise and concise.
4. Avoid showing very detailed derivations. You may do those on rought sheet and only show key steps.
5. **Answer each question using information provided in the question (no clarifications during the exam). If you want to make any assumptions, please state them in your answer.**

Section 1 (9 Descriptive Answer Questions: Total 100 marks). .

1. Consider a classification problem with K classes. Assume that for each class $c = 1, 2, \ldots, K$, we are given a class-attribute vector $\boldsymbol{a}_c \in \mathbb{R}^M$. We are giving training data $\{\boldsymbol{x}_n, y_n\}_{n=1}^N$ with inputs $\boldsymbol{x}_n \in \mathbb{R}^D$, but the training examples are only from the first S classes and the remaining $U = K - S$ classes do not have any training examples. The test input $\boldsymbol{x}_*$ can be from any of the K classes. Explain (with all necessary equations) how you would learn a learning with prototypes (LwP) classifier for this problem. Your solution must not use anything other than simple vector operations like addition or dot products. (**12 marks**).

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

2. Consider the ridge regression problem:

$$\hat{\boldsymbol{w}} = \arg\min_{\boldsymbol{w}} \frac{1}{2}\sum_{n=1}^{N}(y_n - \boldsymbol{w}^\top \boldsymbol{x}_n)^2 + \frac{\lambda}{2}\boldsymbol{w}^\top \boldsymbol{w} = \arg\min_{\boldsymbol{w}} \frac{1}{2}(\boldsymbol{y} - \mathbf{X}\boldsymbol{w})^\top(\boldsymbol{y} - \mathbf{X}\boldsymbol{w}) + \frac{\lambda}{2}\boldsymbol{w}^\top \boldsymbol{w}$$

where $\mathbf{X}$ is the $N \times D$ feature matrix and $\boldsymbol{y}$ is the $N \times 1$ vector of labels of the N training examples. Note that the factor of $\frac{1}{2}$ has been used in the above expression just for convenience of derivations required for this problem and does not change the solution to the problem.

Derive the Newton's method's update equations for each iteration. For this model, how many iterations would the Newton's method will take to converge? (**14 marks**)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

3. Consider K-means clustering where we are trying to learn K means $\mu_1, \ldots, \mu_K$, given N observations $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, with each $\boldsymbol{x}_n \in \mathbb{R}^D$. Suppose we have some *a priori* information that the K means are "close" to known vectors $\mu_1^*, \ldots, \mu_K^*$, respectively. Propose a suitable prior distribution for each mean μ_k that makes use of this information. Now derive the K-means algorithm updates for cluster assignments $\boldsymbol{z}_n$ and cluster means μ_k when we use this prior distribution as a regularizer. (**12 marks**)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

4. Suppose we have collected N observations $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ using a sensor. Let us assume each $\boldsymbol{x} \in \mathbb{R}^D$ as generated from a Gaussian distribution $\mathcal{N}(\mu, \Sigma)$. We would like to estimate the mean and covariance of this Gaussian. However, suppose the sensor was faulty and each $\boldsymbol{x}_n$ could only have part it as observed (think of a blacked out image). Denote $\boldsymbol{x}_n = [\boldsymbol{x}_n^{obs}, \boldsymbol{x}_n^{miss}]$ where $\boldsymbol{x}_n^{obs}$ and $\boldsymbol{x}_n^{miss}$ denote the observed and missing parts, respectively, of $\boldsymbol{x}_n$. We only get to see $\boldsymbol{x}_n^{obs}$. Note that different observations could have different parts as missing (e.g., different images may have different sets of pixels as missing), so the indices of the observed/missing entries of the vector $\boldsymbol{x}_n$ may be different for different n.

   We can use EM to get maximum likelihood estimates of μ and Σ given this partially observed data. To do so, you will treat each $\boldsymbol{x}_n^{miss}$ as a latent variable and estimate its conditional posterior $p(\boldsymbol{x}_n^{miss}|\boldsymbol{x}_n^{obs}, \mu, \Sigma)$, given the current estimates μ and Σ of the parameters. In the M step, you will re-estimate μ and Σ. Clearly write down the following: (1) The expression for $p(\boldsymbol{x}_n^{miss}|\boldsymbol{x}_n^{obs}, \mu, \Sigma)$; (2) The expected CLL for this model; (3) The M step update equations for μ and Σ. (**14 marks**)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

5. Suppose we have data from two classes whose inputs are assumed to be generated from two Gaussian distributions $\mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{bmatrix}2 & 0\\0 & 2\end{bmatrix}\right)$ and $\mathcal{N}\left(\begin{pmatrix}2\\2\end{pmatrix}, \begin{bmatrix}5 & 0\\0 & 5\end{bmatrix}\right)$, respectively. Suppose we have learned a logistic regression model on this data but the model isn't performing on the test set.

   - If we have infinite amount of training data, will logistic regression model achieve zero training error? Briefly justify your answer. (**3 marks**)

   - If we also add a regularizer to the above logistic regression model (infinite training data), will it achieve zero training error? Briefly justify your answer. (**3 marks**)

   - If we switch to a kernel SVM (but keep the training set size as finite), is it possible to achieve zero training error? Briefly justify your answer. (**3 marks**)

   - If we switch to a deep neural network (but keep the training set size as finite), is it possible to achieve zero training error? Briefly justify your answer. (**3 marks**)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

6. Given query, key, and value matrices $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$, respectively (and all being of size $N \times d$), one way to define the $N \times d$ output of the self-attention mechanism is $\text{ATT}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\mathbf{Q}\mathbf{K}^\top/\sqrt{d}\right)\mathbf{V}$, where the softmax function is assumed to be applied row-wise on the matrix $\mathbf{Q}\mathbf{K}^\top/\sqrt{d}$.

   Note that we can also write the same as $\text{ATT}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{D}^{-1}\mathbf{A}\mathbf{V}$ where $\mathbf{A} = \exp(\mathbf{Q}\mathbf{K}^\top/\sqrt{d})$, and $\mathbf{D} = \text{diag}(\mathbf{A}\mathbf{1}_N)$ and $\mathbf{1}_N$ denotes a column vector of all 1s. Denoting $\mathbf{A}$ as $\text{SM}(\mathbf{Q}, \mathbf{K})$, we can define a "softmax" kernel function $\text{SM}(\boldsymbol{q}_i, \mathbf{k}_j) = \exp(\boldsymbol{q}_i^\top \mathbf{k}_j/\sqrt{d})$, such that the $(i,j)^{th}$ entry of $\mathbf{A}$ equals $\text{SM}(\boldsymbol{q}_i, \mathbf{k}_j)$.

   - Show that the above softmax" kernel function $\text{SM}(\boldsymbol{q}_i, \mathbf{k}_j)$ can be written as a scalar multiplied with well-known kernel function. You must show this with precise mathematical expressions. (**6 marks**)
   - Let's define $\text{NEW\_ATT}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}\mathbf{V}$ where $\tilde{\mathbf{A}} = \text{tril}(\mathbf{A})$, $\tilde{\mathbf{D}} = \text{diag}(\tilde{\mathbf{A}}\mathbf{1}_N)$, and tril(.) extracts the lower-triangular part of the matrix including its diagonal. Briefly explain the difference between the outputs of the two attention functions $\text{ATT}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ and $\text{NEW\_ATT}_*(\mathbf{Q}, \mathbf{K}, \mathbf{V})$. (**6 marks**)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

7. Consider modeling some data $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^N$, $\boldsymbol{x}_n \in \mathbb{R}^D$, $y_n \in \{0, 1\}$, using a mixture of logistic regression models, where we model each binary label y_n by first picking one of the K logistic regression models, based on the value of a latent variable $z_n \sim \text{multinoulli}(\pi_1, \ldots, \pi_K)$, and then generating y_n *conditioned* on z_n as $y_n \sim \text{Bernoulli}[\sigma(\boldsymbol{w}_{z_n}^\top \boldsymbol{x}_n)]$. Now consider the *marginal* probability of the label $y_n = 1$, given $\boldsymbol{x}_n$, i.e., $p(y_n = 1|\boldsymbol{x}_n)$, and show that this can be quantity can also be thought of as the output of a neural network. Clearly specify what is the input layer, hidden layer(s), activations, the output layer, and the connection weights of this neural network. (**5 marks**)

8. Consider the following activation function: $h(x) = x\sigma(\beta x)$ where σ denotes the sigmoid function $\sigma(z) = \frac{1}{1+\exp(-z)}$. Show that, for appropriately chosen values of β, this activation function can approximate (1) the linear activation function, and (2) the ReLU activation function. (**5 marks**)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

9. You are given an $N \times M$ matrix $\mathbf{R}$ with binary entries. Your goal is to approximate $\mathbf{R}$ using a product of two matrices $\mathbf{U} \in \mathbb{R}^{N \times K}$ and $\mathbf{V} \in \mathbb{R}^{M \times K}$ where K is usually smaller than N and M. In particular, assume $p(\mathbf{R}|\mathbf{U}, \mathbf{V}) = \text{Bernoulli}(\mathbf{R}|\sigma(\mathbf{U}\mathbf{V}^\top))$ where the sigmoid operation and Bernoulli are applied elementwise on their respective input matrices. Note that this is also equivalent to $p(R_{nm}|\boldsymbol{u}_n, \mathbf{v}_m) = \text{Bernoulli}(R_{nm}|\sigma(\boldsymbol{u}_n^\top \mathbf{v}_m))$, $\boldsymbol{u}_n^\top$ and $\mathbf{v}_m$ denote the n^{th} row of $\mathbf{U}$ and m^{th} column of $\mathbf{V}$, respectively.

   Give an ALT-OPT algorithm to learn $\mathbf{U}$ and $\mathbf{V}$ and show that it reduces to solving $N + M$ logistic regression problems in each iteration. For each logistic regression problem, what is the corresponding input matrix, labels, and the weight vector? (**14 marks**)

Name:

Roll No.: **Dept.**:

IIT Kanpur
CS771A (IML)
End-sem Exam
Date: November 22, 2023

Some distributions and their properties:

- For $x \in \mathbb{R}$, the PDF of univariate Gaussian: $\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{(x-\mu)^2}{2\sigma^2}\}$. If using precision $\beta = 1/\sigma^2$, the PDF is $\mathcal{N}(x|\mu, \beta^{-1}) = \sqrt{\frac{\beta}{2\pi}} \exp\{-\frac{\beta}{2}(x-\mu)^2\}$.
- For $x \in \mathbb{R}^D$, D-dimensional Gaussian: $\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^D|\boldsymbol{\Sigma}|}} \exp\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\}$.
- PDF of a Bernoulli random variable x is $p(x) = \text{Bernoulli}(x|\mu) = \mu^x(1-\mu)^{1-x}$ where $\mu \in (0,1)$ is the success probability.
- Covariance of a vector random variable X is $\text{cov}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2$ where $\mathbb{E}$ denotes expectation.

Some other useful results:

- Given a joint distribution of two groups of random variables $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ which is Gaussian with mean vector $\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_a \\ \boldsymbol{\mu}_b \end{pmatrix}$ and covariance matrix $\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{aa} & \boldsymbol{\Sigma}_{ab} \\ \boldsymbol{\Sigma}_{ba} & \boldsymbol{\Sigma}_{bb} \end{bmatrix}$, the marginal and conditional distributions for Gaussians are:

  $p(\boldsymbol{x}_a) = \mathcal{N}(\boldsymbol{x}_a|\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_{aa})$,
  $p(\boldsymbol{x}_a|\boldsymbol{x}_b) = \mathcal{N}(\boldsymbol{x}_a|\boldsymbol{\mu}_{a|b}, \boldsymbol{\Sigma}_{a|b})$
  where $\boldsymbol{\Sigma}_{a|b} = \boldsymbol{\Sigma}_{aa} - \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}\boldsymbol{\Sigma}_{ba}$, and $\boldsymbol{\mu}_{a|b} = \boldsymbol{\mu}_a + \boldsymbol{\Sigma}_{ab}\boldsymbol{\Sigma}_{bb}^{-1}(\boldsymbol{x}_b - \boldsymbol{\mu}_b)$
- $\frac{\partial}{\partial \boldsymbol{\mu}}[\boldsymbol{\mu}^\top \mathbf{A} \boldsymbol{\mu}] = [\mathbf{A} + \mathbf{A}^\top]\boldsymbol{\mu}$, $\frac{\partial}{\partial \mathbf{A}} \log|\mathbf{A}| = \mathbf{A}^{-\top}$, $\frac{\partial}{\partial \mathbf{A}}\text{trace}[\mathbf{A}\mathbf{B}] = \mathbf{B}^\top$

Name:

Roll No.: **Dept.**: