Date: Tuesday, September 21, 2020. 9:00 am - 10:15 am
There are 4 questions. All Questions Carry 6 points. Max Marks: 24

You must start answer to each question on a new page.
You need to justify all your answers. Answers with insufficient justification may not get any points.

1. **Regression for Countable Data** In class, we looked at the problem of linear regression, where the target variable $y \in \mathcal{R}$. In this question, we will examine the problem, learning with countable data. Let the feature vector be given as: $x \in \mathcal{R}^n$, and let $y \in \{0, 1, 2, \cdots \infty\}$. Let the distribution of $y|x; \theta$ be modeled as a Poisson distribution. Consider learning a model with training data given as $\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$. Recall that if $y \sim Poisson(\lambda)$, then $P(y = k) = \frac{\lambda^k e^{-\lambda}}{k!}$

   (a) Show that Poisson distribution belongs to the exponential family. Recall that y is distribution according to an exponential family if $P(y; \eta) = b(y)exp(\eta y - a(\eta))$. Using the fact that for linear models, $\eta = \theta^T x$, find the relationship between θ and λ (mean of the Poisson distribution).

   (b) Let the relationship between θ parameters and λ be given as derived in part above. We will refer the model learned under these conditions as *regression for countable data.* Derive the expression for the gradient of the log-likelihood function under regression for countable data.

   (c) Show that the Log-likelihood function is concave under the above modeling assumptions.

2. **Conceptual Understanding**

   (a) Suppose you are teaching a class on Machine Learning, and in an assignment for the course, you ask the students learn an Machine Learning model on a given training data. Assume that the features are real valued and the target value is binary, i.e., they need to learn a classification model. Suppose, there are n students in the class, and each one submits a model h_l $l \in \{1, \cdots, n\}$. To keep things simple, let $n = 3$. Corresponding to each model h_l, you can compute the training accuracy given as t_l. You also have some held out validation set, on which you can compute the accuracy v_l for the submitted model h_l. Your task is now to evaluate the quality of the models submitted by students, and also give suggestions to students on how to improve their models. Consider the following scenarios:

      i. For a student model h_1, the training accuracy t_1 is 100%. But the validation accuracy v_1 is poor.

      ii. For a student model h_2, the training accuracy t_2 is above 90%. And the validation accuracy v_2 is also close to training accuracy.

      iii. For a student model h_3, the training accuracy t_3 is close to 50%, and and the validation accuracy v_3 is also close to training accuracy.

For each of the cases above, describe how would you judge the model. Describe whether the model is under-fitting or over-fitting, or possibly none of these. What advice would you give to the student to improve their model and why: imprecise/vague answers will not fetch any points.

(b) In the Naive Bayes model (with discrete attributes), recall that the smoothed estimate for the distribution of a attribute (given the class) is given as:

$$\theta_{jl|y=k} = \frac{\sum_i \mathbb{1}\{y^{(i)} = k\}\mathbb{1}\{x_j^{(i)} = 1\} + c.1}{\sum_i \mathbb{1}\{y^{(i)} = k\} + c.L} \tag{1}$$

where the symbols are as used in the class (j varies over each attribute, l over the values it can take). Describe how does the parameter c, which controls the strength of smoothing is also responsible for keeping over-fitting under check. Argue. What happens if you use a very large, or a very small value of c?

3. **Gaussian Naive Bayes** Consider learning a Gaussian Naive Bayes model on the training data of the from $\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$. Recall that Gaussian Naive Bayes model is a GDA model which has an additional Naive Bayes assumption for the distribution of its attributes given the class variable. Let each $y \in \{1, 2, \cdots, r\}$. Without loss of generality, let r be even. Let the parameters of the model be given as $\Theta = (\phi, \mu_1, \Sigma_1, \mu_2, \Sigma_2, \cdots, \mu_r, \Sigma_r)$ where ϕ is the parameter associated with class prior, and μ_k, Σ_k denote the mean and co-variance matrix, for the distribution of the attributes (x) conditioned on $y = k$. Assume $\Sigma_1 = \Sigma_3 = \cdots = \Sigma_{r-1}$, and $\Sigma_2 = \Sigma_4 = \cdots = \Sigma_r$.

(a) What form does Σ_1 (or Σ_2) take? Argue. Write down the joint log-likelihood of the training data expressed as a function of the model parameters.

(b) Derive the ML (maximum-likelihood) estimate for the parameters $\mu_1, \cdots, \mu_r$, and Σ_1 and Σ_2 from first principles. Show that the expressions that you derive are consistent with those derived in the class for the case of a GDA model.

4. **Optimization Methods**

(a) Consider applying Newton's method to optimize (minimize) the quadratic function $f(\theta) = \theta^T A\theta + a^T\theta + b$. Assume $A \in \mathcal{R}^{n\times n}$, $\theta \in \mathcal{R}^n$, $a \in \mathcal{R}^n$, $b \in \mathcal{R}$. Show that Newton's method would converge in a single iteration in this setting. What is the value of the θ parameters at the local optima? You can assume you are given Newton's parameter update rule in the multi-variate setting as discussed in the class. For proving this result, you may require to compute first/second order-gradient of $\theta^T A\theta$ term: you should derive your expression from first principles, i.e., rules of calculus over single variables. You can also assume the fact that for any vector $c \in \mathcal{R}^n$, $\nabla_\theta c^T\theta = c$, but not assume any additional results about vector or matrix calculus. You can assume that the matrix A or any related forms (as required by your answer) are invertible. Hint: You should consider two cases (i) A is symmetric (ii) A is not symmetric.

(b) Consider a learning problem with training data $\{x^{(i)}, y^{(i)}\}_{i=1}^{m}$. Let θ denote the set of parameters under some probabilistic modeling assumptions, and let $LL(\theta)$ denote the log-likelihood of the data parameterized by θ. Recall that in ML estimation, the goal is to find those θ parameters which maximise $LL(\theta)$. Consider computing the likelihood over a randomly sampled mini-batch of examples of size r, given as $LL_b(\theta)$. Assume that we are using the strategy for choosing a mini-batch where each example is sampled uniformly at random from the training set with replacement. Consider using Gradient Descent (GD) and Stochastic Gradient Descent (SGD) for finding the optimal values of the theta parameters. Recall that the parameter update equations in GD, and SGD, are respectively given as:

$$\theta^{(t+1)} \leftarrow \theta^{(t)} + \eta . \frac{1}{m} \nabla_\theta LL(\theta) \text{ (Gradient Descent)}$$

$$\theta^{(t+1)} \leftarrow \theta^{(t)} + \eta . \frac{1}{r} \nabla_\theta LL_b(\theta) \text{ (Stochastic Gradient Descent)}$$

i. Show that expected value $E[\theta^{t+1} - \theta^{(t)}]$ under the SGD update, is identical to $\theta^{(t+1)}$ - $\theta^{(t)}$ in the GD update, where the expectation is taken over randomly sampled examples (with replacement) from the training set in each mini-batch.

ii. Let $\mathrm{Var}(\nabla_\theta LL_1(\theta))$ denote the variance of the log-likelihood computed over a mini-batch of size 1, where the sampling distribution corresponds to uniformly at random choosing one example from the training set. Then, compute the expression $\mathrm{Var}(\theta^{(t+1} - \theta^{(t)})$ in SGD (i.e., variance in estimate of change in value of $\theta^{(t)}$ parameters), in terms of $\mathrm{Var}(\nabla_\theta LL_1(\theta))$, and the batch size r. You should clearly show the steps used to obtain your expression.