Name: _____

Submit your answers via the submission site. Your submission must be easily legible.

This exam is *designed* to be completed within 75 minutes, though you actually have from Thursday 3/30 through Monday 4/3 to complete it. You may use any notes that you have created or any photocopies (printouts) of publicly available CMSC 678 course material. You may not share or otherwise distribute the exam. The must be completed on your own, without any assistance from or discussion with other people, and you must adhere to the course policies on academic integrity.

Good luck!

Question	Points	Score
1	18	
2	19	
3	18	
4	20	
Total:	75	

1. This question considers a multi-class perceptron classifier. Your answer must consider at least three classes, but the exact number is up to you. (You may want to read through this entire question first.)

   (a) (2 points) Provide a multidimensional dataset that your multi-class perceptron could *not* learn. Explain why this dataset cannot be learned. Ensure your dataset has at least two instances per class.

(b) (8 points) For the dataset you provided in part (a), step through the first three training steps. Here, a single training step means predicting and potentially updating from a single item from the training set. Be sure to specify the observed input order and the initial values of any weights (or hyperparameters).

(c) (8 points) How can we arrive at the classic perceptron algorithm using empirical risk minimization (ERM)? Provide the necessary specifics for ERM.

2. In class we discussed a number of different ways of creating multi-class classifiers. This question asks you to compare some of those approaches.

   In particular, assume we compute L-dimensional, unnormalized log-probability class scores u (also called *logits*) via a linear layer based on K features $f(X)$, i.e., $u = \theta f(X)$, where θ is a LxK matrix of weights. The options for this question are:

   1. In option 1, we compute $p(y = l|X) = \text{softmax}(u)_l$.
   2. In option 2, we compute $p(y = l|X) = \sigma(u_l)$, where σ is the sigmoid function.
   3. In option 3, we form one-vs-all classifiers.

   (a) (5 points) Assuming our goal is to minimize empirical posterior 0-1 loss, do the decoding rules for options 1 vs. 2 differ? If not, why not? If so, how?

(b) (5 points) Aside from any differences (if any) you already listed, what interpretation differences are there between options 1 and 2?

(c) (5 points) When are options 2 and 3 the same, and when are they different?

(d) (4 points) Let's say your training data is balanced: what challenges might that pose for option 3?

3. (a) (9 points) For classification problems probabilistically modeled as $p(y|x)$, discuss the connection between minimizing cross-entropy loss and optimizing expected posterior 0-1 loss. Make as few assumptions as possible about the form of $p(y|x)$ in your answer.

(b) (9 points) Let $p(y|x) = \text{softmax}(\theta f(x))$ be a classifier learned by maximizing the posterior label likelihood. Discuss how the learned model can be viewed as respecting constraints on the available training data. Your answer should refer to specific components of the model, learning objective, or components derived during learning.

4. You just started a job at Sweet Data Science Inc., one of the hottest startups around. The first project you've been assigned is to design a machine learning system to accurately predict the number of times Y that a tweet or other social media post X will be reshared. Two of your colleagues, Alex and Taylor, have been working on this problem and have come up with two different approaches.

   (a) (1 point) Provide the set of values Y could take, either explicitly or as a range.

   (b) (3 points) Alex *really* likes discrete classifiers, and so wants to build a conditional **classifier**, $p_\theta^{\text{cls}}(Y \mid X)$, parametrized by the weight vector θ and learned via cross-entropy loss. Taylor is trying to convince Alex that this isn't a good option. Why is this discrete classifier approach not a good option? Think about what could go wrong, what is mis-specified, or what computational trouble might there be.

Taylor studied statistics and has heard about this technique called Poisson regression. In Poisson regression we model a non-negative integer response Y based off of a linear model involving our K-dimensional vector input $\phi(X)$. What makes Poisson regression different is that we model the log of our predicted value $\hat{y}$ as a linear model:

$$\log \hat{y} = \omega^\intercal \phi(X). \tag{1}$$

This means we compute our predicted value as $\hat{y} = \exp\left(\omega^\intercal \phi(X)\right)$.

(c) (2 points) If $\phi(X)$ is K-dimensional, how many different weights need to be learned in this model?

For a predicted value $\hat{y}$ and correct value $y^\star$, the loss function is

$$\ell(y^\star, \hat{y}) = \hat{y} - y^\star \log \hat{y} - \log\left((y^\star)!\right). \tag{2}$$

The $(y^\star)!$ represents the factorial operation: $(y^\star)! = 1 \times 2 \times 3 \times \cdots y = \prod_{i=1}^{y^\star} i$, where by definition, $0! = 1$.

Alex is having a hard time understanding this loss function.

(d) (2 points) Indicate the term(s) on the right-hand side of the equation that depend on the weights we need to learn.

$$\ell(y^\star, \hat{y}) = \hat{y} - y^\star \log \hat{y} - \log\left((y^\star)!\right)$$

(e) (2 points) Rewrite and simplify the loss function (complete the right hand side of the below equation). Your answer should be in terms of ω, $\phi(X)$, and the terms that you did not circle in part d (the parameter list to ℓ has intentionally been left off):

$\ell =$

(f) (4 points) Given a labeled dataset of size N (e.g., $\{(X_1, y_1^\star), \ldots, (X_N, y_N^\star)\}$), complete the right hand side of the following equation to formulate the empirical risk minimization objective. Call this objective $\mathcal{L}(\omega)$, and simplify it (you can use more than one line if helpful):

$$\mathcal{L}(\omega) = \frac{1}{N} \sum$$

(g) (6 points) Given a collection of $N = 250,000$ labeled tweets, how should you, Alex, and Taylor use that data to rigorously and correctly develop effective models?