STA250 — Theoretical Foundations of Modern AI
Homework 0

Spencer Frei
UC Davis
sfrei@ucdavis.edu

January 30, 2024

Problem 1

A random variable X is with finite mean μ is *sub-Gaussian with parameter* σ if

$$\mathbb{E}\left[\exp(\lambda(X-\mu))\right] \leq \exp(\sigma^2\lambda^2/2), \quad \forall \lambda \in \mathbb{R}.$$

We say X is σ-sub-Gaussian, and we call σ^2 its *variance proxy*.

(a) If X is σ-sub-Gaussian with mean μ, show that it has the following tail bound:

$$\mathbb{P}(|X-\mu| \geq t) \leq 2\exp(-t^2/(2\sigma^2)), \quad \forall t \in \mathbb{R}.$$

(b) Let X_i, $i = 1, \ldots, n$, be random variables where X_i has mean μ_i and is σ_i-sub-Gaussian. Show that $Z = \sum_{i=1}^n X_i$ is sub-Gaussian with mean $\sum_{i=1}^n \mu_i$. Give an upper bound for the variance proxy of Z. If in addition we assume the random variables are independent, what does this imply?

(c) Derive a tail bound for $|Z - \mathbb{E}Z|$. If $\delta \in (0,1)$, how large can we expect $|Z - \mathbb{E}Z|$ to be with probability at least $1-\delta$? Answer each of these two questions in the following two situations: (i) we assume X_i are independent, and (ii) we make no assumptions on the relationships between the X_i's.

Problem 2

If A and B are symmetric positive definite matrices, is their product AB positive definite? If yes, provide a proof. If not, provide a counterexample and state conditions under which AB is positive definite.

Problem 3

Suppose that $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$ for $i = 1, \ldots, n$. Let

$$\widehat{L}_2(w) := \frac{1}{n}\sum_{i=1}^n (y_i - x_i^\top w)^2, \quad \widehat{L}_1(w) := \frac{1}{n}\sum_{i=1}^n |y_i - x_i^\top w|.$$

1. If $w_2^* \in \mathrm{argmin}\{\widehat{L}_2(w)\}$, what must be true about w_2^* (i.e., what are necessary conditions for minimizing the L_2 empirical loss)?

2. If $w_1^* \in \mathrm{argmin}\{\widehat{L}_1(w)\}$, what must be true about w_1^* (i.e., what are necessary conditions for minimizing the L_1 empirical loss)?

3. Are there any conditions under which you can write closed-form solutions for w_2^* and/or w_1^*?

Problem 4

Consider training data $S = \{(x_i, y_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} \mathcal{D}$ which is generated from the following distribution. Let $\mu \in \mathbb{R}^d$ be a fixed vector. A pair $(x, y) \sim \mathcal{D}$ is generated as follows

- $y \sim \mathsf{Unif}(\{\pm 1\})$.
- $z \sim \mathsf{N}(0, I_d)$.
- $x = y\mu + z$.

Let $\ell(q) = \log(1 + \exp(-q))$ be the binary cross-entropy loss. Let $m \in \mathbb{N}$, and let $\phi(q) = \max(0, q)$ be the ReLU. Let $a \in \mathbb{R}^m$ and $W \in \mathbb{R}^{m \times d}$ be a matrix with rows $w_j^\top \in \mathbb{R}^d$. Concatenate all of the parameters into $\theta = (a, W)$. Consider a two-layer network with ReLU activations without biases,

$$f(x; \theta) = \sum_{j=1}^m a_j \phi(\langle w_j, x \rangle).$$

Define the empirical risk over the training data using the *margin* of the training examples $y_i f(x_i; W)$,

$$\widehat{L}(\theta) = \frac{1}{n} \sum_{i=1}^n \ell(y_i f(x_i; \theta)).$$

For initialization $W^{(0)}$, consider gradient descent with fixed learning rate η over the logistic loss for two-layer ReLU networks,

$$\theta_{t+1} = \theta_t - \eta \nabla \widehat{L}(\theta_t).$$

This is "vanilla" (full-batch) gradient descent.

1. Provide Python code which creates a dataset S as a function of n, d, and μ.

2. Provide PyTorch code which instantiates a two-layer neural network class, where the user can specify the number of neurons in the network. (PyTorch hint: you should be using Linear layers, and nn.Sequential or nn.ModuleList are easy ways to build multi-layer neural nets)

3. Provide PyTorch code which trains the two-layer neural network (using the default PyTorch initialization scheme) using full-batch gradient descent with logs for both the training loss and validation loss. Ensure that the code allows for a user-specified number of training samples n_{train}, validation samples n_{valid}, dimension d, number of neurons m, and learning rate η.

4. Plot the results of training in the following two settings. In both settings, use $\eta = 0.001$ and $\mu = d^{0.26} s$ where s is uniform on the sphere.

(a) $d = 1000, n = 100$

(b) $d = 100, n = 1000$

In each of the two settings above, there should be two plots with two lines on each of them. One plot should have the training loss and the validation loss. The other plot should have the training accuracy and the validation accuracy.

If you are unfamiliar with deep learning software packages, I recommend that you start with the PyTorch tutorials listed on the course website. You can also think about using ChatGPT, but you will be responsible for any errors in your code and/or plots.