Homework #2

RELEASE DATE: 09/23/2024

RED CORRECTION: 10/02/2024 14:15

DUE DATE: 10/07/2024, BEFORE 13:00 on GRADESCOPE

QUESTIONS ARE WELCOMED ON DISCORD (INFORMALLY) OR VIA EMAILS (FORMALLY).

You will use Gradescope to upload your scanned/printed solutions. For problems marked with (*), please follow the guidelines on the course website and upload your source code to Gradescope as well. Any programming language/platform is allowed.

Any form of cheating, lying, or plagiarism will not be tolerated. Students can get zero scores and/or fail the class and/or be kicked out of school and/or receive other punishments for those kinds of misconducts.

Discussions on course materials and homework solutions are encouraged. But you should write the final solutions alone and understand them fully. Books, notes, and Internet resources can be consulted, but not copied from.

Since everyone needs to write the final solutions alone, there is absolutely no need to lend your homework solutions and/or source codes to your classmates at any time. In order to maximize the level of fairness in this class, lending and borrowing homework solutions are both regarded as dishonest behaviors and will be punished according to the honesty policy.

You should write your solutions in English with the common math notations introduced in class or in the problems. We do not accept solutions written in any other languages.

This homework set comes with 200 points and 20 bonus points. In general, every homework set would come with a full credit of 200 points, with some possible bonus points.

1. (10 points, auto-graded) What is the growth function of diagonally aligned perceptrons in 2D for $N \geq 4$? Those perceptrons are all perceptrons with $w_1 = w_2$ or $w_1 = -w_2$. That is, they are lines with slope 1 or -1 on the 2D plane. Choose the correct answer.

[a] $4N + 4$

[b] $4N + 2$

[c] $4N$

[d] $4N - 2$

[e] $4N - 4$

2. (10 points, auto-graded) Consider having 16 bags, each containing some super large number of cards. Half of the cards in each bag is white, and the other half is black. Now, someone is going to draw five cards randomly from each bag. Each drawing is going to independently form a hand. A big prize of four million dollars is wired to zir account if one of the 16 hands is purely white. What is the probability that such an event will happen?

[a] $\frac{1}{32^{16}}$

[b] $\frac{31^{16}}{32^{16}}$

[c] $\frac{32^{16}-31^{16}}{32^{16}}$

[d] $\frac{16}{32}$

[e] $\frac{14}{32}$

3. (10 points, auto-graded) What is the growth function of $(11, 26)$-passing perceptrons on $\mathcal{X} = \mathbb{R}^2$? Those perceptrons are

$$\mathcal{H} \quad = \quad \{h\colon h(\mathbf{x}) = \text{sign}(w_1(x_1 - 11) + w_2(x_2 - 26)) \quad \textit{i.e.}\text{ perceptrons that pass } (11, 26)\}$$

[a] N

[b] N^2

[c] $2N$

[d] $2N^2$

[e] $N^2 - N + 2$

4. (10 points, auto-graded) Consider a hypothesis set that contains 6211 perceptrons

$$h_m(\mathbf{x}) = \text{sign}(\mathbf{w}_m^T\mathbf{x}), \text{ for } m = 1, 2, \cdots, 6211$$

with $\mathbf{x} \in \mathbb{R}^{1+1126}$ (including x_0). What is the tightest upper bound on the possible VC dimension of this hypothesis set? Choose the correct answer.

[a] $\log_2(1126)$

[b] $\log_2(6211)$

[c] $\sqrt{6211}$

[d] $\log_2(1126 + 6211)$

[e] 1126

5. (20 points, human-graded) In class, we discussed whether it is possible to predict the next term of an integer sequence based on the first few terms. We asked chatGPT this question "*If we know that the first $N - 1$ terms of an integer sequence is generated from some polynomial of degree N, is it possible to predict the next integer?*"

Here is zir answer

`https://chatgpt.com/share/66e88dfc-0b44-8002-a738-91a5d1613354`

Argue with 10-20 English sentences on whether you agree with the chatGPT agent or not, as if you are the "boss" of the agent. The TAs will grade based on the persuasiveness of your arguments—please note that our TAs are more used to being persuaded by humans than machines. So if your arguments do not look very human-written, the TAs may not be persuaded.

6. (20 points, human graded) Next, we illustrate what happens with multiple bins. Consider a special lottery game as follows. The game operates by having four kinds of lottery tickets placed in a big black bag, each kind with the same (super large) quantity. Exactly sixteen numbers $1, 2, \ldots, 16$ are written on each ticket. The four kinds are

- A: all even numbers are colored orange, all odd numbers are colored green
- B: all even numbers are colored green, all odd numbers are colored orange
- C: all small numbers (1-8) are colored orange, all big numbers (9-16) are colored green
- D: all small numbers (1-8) are colored green, all big numbers (9-16) are colored orange

Every person is expected to draw five tickets from the bag. A small prize of 1450 is given if the five tickets contain "some number" that is purely green. What is the probability that such an event will happen?

7. (20 points) Continuing from Problem 6, a bigger price of five million dollars will be delivered by a bagman to you if the five tickets contain five green 5's. What is the probability that such an event will happen?

Hint: Each number can be viewed as a "hypothesis" and the drawn tickets can be viewed as the data. The E_{OUT} of each hypothesis is simply $\frac{1}{2}$ (You are welcome. ;-)). Problem 7 asks you to calculate the BAD probability for hypothesis 5; Problem 6 asks you to calculate the BAD probability for all hypotheses, taking the sampling dependence into consideration. Actually, Problem 2 can be viewed as the same game that cuts the tickets to pieces of different numbers (with some recoloring), and placing different numbers in different bags so they'd be sampled independently.

8. (20 points, human-graded) Assume that we have M slot machines, or one-armed bandits, in front of us. Each machine has an unknown probability of μ_m for returning one coin, and a probability of $1 - \mu_m$ for returning no coin. For each of the time step $t = 1, 2, \ldots$, assume that we pull the machine $m = ((t-1) \bmod M) + 1$. After some $t > M$ time steps, we'd have pulled machine m for N_m times, and collected c_m coins from machine m. Note that $N_m \geq 1$ because $t > M$. Using the following one-sided Hoeffding's inequality (which is slightly different from what we taught in class)

$$P(\mu > \nu + \epsilon) \leq \exp(-2\epsilon^2 N),$$

where ν, μ, ϵ, N have been defined in our class, we can easily prove that when given a fixed machine m and a fixed δ with $0 < \delta < 1$,

$$P\left(\mu_m > \frac{c_m}{N_m} + \sqrt{\frac{\ln t - \frac{1}{2}\ln \delta}{N_m}}\right) \leq \delta t^{-2}.$$

Use the fact above to prove that for $M \geq 2$, for all slot machines $m = 1, 2, \ldots, M$ and for all $t = M+1, M+2, \ldots$, with probability at least $1 - \delta$,

$$\mu_m \leq \frac{c_m}{N_m} + \sqrt{\frac{\ln t + \ln M - \frac{1}{2}\ln \delta}{N_m}}.$$

You can use the magical fact that

$$\sum_{t=1}^{\infty} t^{-2} = \frac{\pi^2}{6}.$$

Hint: The fact that we can upper-bound all μ_m confidently and simultaneously by $\frac{c_m}{N_m}$ plus a deviation term is the core technique for deriving the so-called upper-confidence bound algorithm for multi-armed bandits, which is an important algorithm for the task of online and reinforcement learning. The actual algorithm differs from what we do here by pulling the machine with the largest upper confidence bound in each iteration, instead of periodically going through each machine. Those who are interested can certainly search for more about this.

If the verbal description above is confusing to you, please prove the mathematical statement below. For any finite $M \geq 2$ and any given $0 < \delta < 1$,

$$P\left(\forall m \in \{1, 2, \ldots, M\} \text{ and } t \in \{M+1, M+2, \ldots\} \quad \mu_m \leq \frac{c_m}{N_m} + \sqrt{\frac{\ln t + \ln M - \frac{1}{2}\ln \delta}{N_m}}\right) \geq 1 - \delta.$$

In other words, there is an iteration-dependent deviation term that needs to be jointly satisfied for all iterations and all slot machines with a high probability.

9. (20 points, human-graded) A boolean function $h\colon \{-1, +1\}^k \to \{-1, +1\}$ is called *symmetric* if its value does not depend on the permutation of its inputs, i.e., its value only depend on the number of ones in the input. What is the VC dimension of the set of all symmetric boolean functions?

10. (20 points, human-graded) In class, we taught about the learning model of "positive and negative rays" (which is simply one-dimensional perceptron) for one-dimensional data. The model contains hypotheses of the form:

$$h_{s,\theta}(x) = s \cdot \text{sign}(x - \theta).$$

You can take $\text{sign}(0) = -1$ for simplicity but it should not matter much for the following problems. The model is frequently named the "decision stump" model and is one of the simplest learning models. As shown in class, for one-dimensional data, the growth function of the model is $m_{\mathcal{H}}(N) = 2N$ and its VC dimension is 2.

In the following problems, you are asked to first play with decision stumps on an artificial data set. First, start by generating a one-dimensional data by the procedure below:

a) Generate x by a uniform distribution in $[-1, 1]$.

b) Generate y by $y = \text{sign}(x) + \text{noise}$, where the noise flips the sign with probability $0 \le p < \frac{1}{2}$.

With the (x, y) generation process above, prove that for any $h_{s,\theta}$ with $s \in \{-1, +1\}$ and $\theta \in [-1, 1]$,

$$E_{\text{out}}(h_{s,\theta}) = u + v \cdot |\theta|, \text{ where}$$

~~Prove that~~

$$\begin{aligned} v &= s(\frac{1}{2} - p) \\ u &= \frac{1}{2} - v \end{aligned}$$

11. (20 points, code needed, human-graded) In fact, the decision stump model is one of the few models that we could minimize E_{in} efficiently by enumerating all possible thresholds. In particular, for N examples, there are at most $2N$ dichotomies (see the slides for positive rays), and thus at most $2N$ different E_{in} values. We can then easily choose the hypothesis that leads to the lowest E_{in} by the following decision stump learning algorithm.

(1) sort all N examples x_n to a sorted sequence $x'_1, x'_2, \ldots, x'_N$ such that $x'_1 \le x'_2 \le x'_3 \le \ldots \le x'_N$

(2) for each $\theta \in \{-1\} \cup \{\frac{x'_i + x'_{i+1}}{2} : 1 \le i \le N-1 \text{ and } x'_i \ne x'_{i+1}\}$ and $s \in \{-1, +1\}$, calculate $E_{\text{in}}(h_{s,\theta})$

(3) return the $h_{s,\theta}$ with the minimum E_{in} as g; if multiple hypotheses reach the minimum E_{in}, return the one with the smallest $s \cdot \theta$.

(*Hint: CS-majored students are encouraged to think about whether the second step can be carried out efficiently, i.e. $O(N)$, using dxxxxxc pxxxxxxxxxg instead of the naive implementation of $O(N^2)$.*)

Generate a data set of size 12 by the procedure above with $p = 15\%$, and run the one-dimensional decision stump algorithm on the data set to get g. Record $E_{\text{in}}(g)$ and compute $E_{\text{out}}(g)$ with the formula in Problem 10. Repeat the experiment 2000 times. Plot a scatter plot of $(E_{\text{in}}(g), E_{\text{out}}(g))$, and calculate the median of $E_{\text{out}}(g) - E_{\text{in}}(g)$. Then, provide the first page of the snapshot of your code as a proof that you have written the code.

12. (20 points, code needed, human-graded) Repeat Problem 11, but instead of picking the best-E_{in} hypothesis as g, randomly generate some $s \in \{-1, +1\}$ and $\theta \in [-1, 1]$ uniformly, and take the resulting hypothesis as your g_{RND}. Plot a scatter plot of $(E_{\text{in}}(g_{\text{RND}}), E_{\text{out}}(g_{\text{RND}}))$, and calculate the median of $E_{\text{out}}(g_{\text{RND}}) - E_{\text{in}}(g_{\text{RND}})$. Compare the scatter plot and the median value with those of Problem 11. Describe your findings. Then, provide the first page of the snapshot of your code as a proof that you have written the code.

13. (Bonus 20 points, human-graded) The decision stump model can be extended to multi-dimensional as follows. For data sets that contains $\mathbf{x} \in \mathbb{R}^d$, we can apply a decision stump on the i-th dimension:

$$h_{s,i,\theta}(\mathbf{x}) = s \cdot \text{sign}(x_i - \theta).$$

Consider a hypothesis set $\mathcal{H}$ that contains all such decision stumps in $\mathbb{R}^d$. Derive the tightest upper bound on the VC dimension of $\mathcal{H}$ that you can think of. The TAs are allowed to give partial credits based on the tightness of your upper bound.