

HOMEWORK 6: LEARNING THEORY AND ENSEMBLE METHODS

10-701 Introduction to Machine Learning
(PhD) (Spring 2024)

Carnegie Mellon University

pi piazza.com/cmu/spring2024/10701/home

OUT: Thursday, Apr 11th, 2024*

DUE: Saturday, Apr 20th, 2024 11:59 PM

START HERE: Instructions

- **Collaboration policy:** Collaboration on solving the homework is allowed, after you have thought about the problems on your own. It is also OK to get clarification (but not solutions) from books or online resources, again after you have thought about the problems on your own. There are two requirements: first, cite your collaborators fully and completely (e.g., “Jane explained to me what is asked in Question 2.1”). Second, write your solution *independently*: close the book and all of your notes, and send collaborators out of the room, so that the solution comes from you only. See the Academic Integrity Section in our course syllabus for more information: <https://www.cs.cmu.edu/~hchai2/courses/10701/#Syllabus>
- **Late Submission Policy:** See the late homework policy here: <https://www.cs.cmu.edu/~hchai2/courses/10701/#Syllabus>
- **Submitting your work to Gradescope:** There will be a single submission slot for this homework on Gradescope. Please use the provided template. The best way to format your homework is by using the Latex template released in the handout and writing your solutions in Latex. However submissions can be handwritten onto the template, but should be labeled and clearly legible. If your writing is not legible, you will not be awarded marks. Each derivation/proof should be completed in the boxes provided below the question, **you should not move or change the sizes of these boxes** as Gradescope is expecting your solved homework PDF to match the template on Gradescope. If you find you need more space than the box provides you should consider cutting your solution down to its relevant parts, if you see no way to do this, please add an additional page at the end of the homework and guide us there with a ‘See page xx for the rest of the solution’.

Regrade requests can be made after the homework grades are released, however this gives the TA the opportunity to regrade your entire paper, meaning if additional mistakes are found then points will be deducted.

For multiple choice or select all that apply questions, shade in the box or circle in the template document corresponding to the correct answer(s) for each of the questions. For \LaTeX users, use ■ and ● for shaded boxes and circles, and don’t change anything else. If an answer box is included for showing work, **you must show your work!**

*Compiled on Thursday 11th April, 2024 at 13:21

1 Learning Theory [24 points]

1. Recall that a set of points is *shattered* by a class of functions \mathcal{H} if all possible $\{-1, +1\}$ labellings of the points can be produced by some function in \mathcal{H} . The Vapnik-Chervonenkis (VC) dimension $d_{VC}(\mathcal{H})$ is the size of the largest set of points that can be shattered by the hypothesis set \mathcal{H} . Also recall that the growth function $g_{\mathcal{H}}(m)$ of \mathcal{H} is defined as the maximum number of distinct labellings \mathcal{H} can induce on any set of m data points.

In this problem, we will explore the hypothesis set where each hypothesis is a combination of two simpler hypotheses. More precisely, given two hypotheses h_1 and h_2 , we define $h = h_1 \sqcup h_2$ as a new hypothesis that labels an example $+1$ only if either h_1 or h_2 give the label $+1$, otherwise, it is labeled -1 . We can extend this to sets of hypotheses: given two sets of hypotheses \mathcal{H}_1 and \mathcal{H}_2 , define $\mathcal{H}_1 \sqcup \mathcal{H}_2 = \{h_1 \sqcup h_2 \mid h_1 \in \mathcal{H}_1, h_2 \in \mathcal{H}_2\}$ as the set of all unions of hypothesis pairs from the two classes \mathcal{H}_1 and \mathcal{H}_2 .

- (a) **[8 points]** Show that for any m and any pair of hypothesis sets \mathcal{H}_1 and \mathcal{H}_2 , if $\mathcal{H}^* = \mathcal{H}_1 \sqcup \mathcal{H}_2$ then $g_{\mathcal{H}^*}(m) \leq g_{\mathcal{H}_1}(m)g_{\mathcal{H}_2}(m)$.

- (b) **[8 points]** Let \mathcal{H} be a hypothesis space with VC dimension d . Define $\mathcal{H}^* = \mathcal{H} \sqcup \mathcal{H}$ as the hypothesis space produced by all unions of pairs of hypotheses from \mathcal{H} (assuming that $\mathcal{H}_1 = \mathcal{H}_2 = \mathcal{H}$ in our above definitions). Show that the VC dimension d_* of \mathcal{H}^* is bounded by $O(d \log d)$. You may use the fact that if $2^x \leq x^y$, then $x \leq k \cdot y \log y$ for some constant k ; you may also use the result from the question **Hint:** Since d_* is the VC dimension of \mathcal{H}^* , then by definition, there exists a set S of d_* points that is shattered by \mathcal{H}^* . By the Sauer-Shelah lemma, we know that $g_{\mathcal{H}}(|S|)$ is bounded by $O(d_*^d)$.

2. **[4 points]** Consider the set of all positive circles in \mathbb{R}^2 i.e. circles where the interior is classified as positive and everything else is classified as negative. What is the VC-dimension of this hypothesis set? **For full credit, you must justify your answer.**

3. **[4 points]** Consider the hypothesis set implicitly defined by the k -NN algorithm for $k = 1$ using the Euclidean distance metric. What is the VC-dimension of this hypothesis set? **For full credit, you must justify your answer.**

2 Random Forests [14 points]

1. [2 points] In a random forest, is it better to have high correlation between the individual trees? Why or why not?

2. [2 points] **Select all that apply:** Which of the following is true about OOB error?

- ☐ OOB error is calculated on a held-out dataset separate from the dataset used to generate bootstrap samples
- ☐ OOB error is the aggregated value of the errors of subsets of the ensemble on samples those subsets were not trained on
- ☐ The B -fold cross-validation error is the same as OOB error where B is the number of decision trees in the random forest
- ☐ OOB error can be used to tune model hyperparameters
- ☐ None of the above

3. [2 points] **Select all that apply:** Which of the following are hyperparameters that can be tuned in a random forest?

- ☐ Number of trees trained
- ☐ Number of points used to train each decision tree
- ☐ Size of feature subsets used to train each decision tree
- ☐ Which features are used for splits in each decision tree
- ☐ None of the above

4. In this question, we will now derive an error bound for random forests in the case of **binary classification**. Given a random forest of B trees $\{h_i(x)\}_{i=1}^B$ and a sample (x, y) drawn from some data distribution \mathcal{D} , define the classification margin as:

$$m(x, y) = \frac{1}{B} \left(\sum_{i=1}^B \mathbb{I}[h_i(x) = y] - \sum_{i=1}^B \mathbb{I}[h_i(x) \neq y] \right)$$

In words, the margin $m(x, y)$ is the difference between the average vote for the correct label and the average vote for the incorrect label.

- (a) [2 points] **Fill in the blank:** For any example (x, y) , the example is classified incorrectly if and only if $m(x, y) \leq \underline{\hspace{1cm}}$. Assume majority vote ties are classified incorrectly.

- (b) **[4 points]** Observe that $P_{(x,y) \sim \mathcal{D}}(m(x,y) \leq c)$, where c is your answer to part (a), corresponds to the generalization error of the ensemble. Additionally, for an ensemble $\{h_i(x)\}_{i=1}^B$, define the *strength* of the ensemble as $s = \mathbb{E}_{(x,y) \sim \mathcal{D}}[m(x,y)]$.

Assume $s > 0$. Derive a bound for the generalization error in the form $P(m(x,y) < c) \leq d$, where d is an expression in terms of s and $\text{Var}(m(x,y))$. Show your work.

Hint: You should use Chebyshev's inequality, which states that for any random variable X with finite expectation and variance and any constant $a > 0$, we have

$$P(|X - \mathbb{E}[X]| \geq a) \leq \frac{\text{Var}(X)}{a^2}$$

5. **[2 points] Select one:** Through some additional manipulation, it is possible to show that $\text{Var}(m(x,y)) \leq \bar{\rho}(1 - s^2)$, where $\bar{\rho}$ is the mean correlation between trees in the ensemble. Substitute this into your bound from part (b). Which of the following describes how the error bound is affected by s and $\bar{\rho}$?
- ☐ The error bound gets smaller as $\bar{\rho}$ increases and s increases.
 - ☐ The error bound gets smaller as $\bar{\rho}$ increases and s decreases.
 - ☐ The error bound gets smaller as $\bar{\rho}$ decreases and s increases.
 - ☐ The error bound gets smaller as $\bar{\rho}$ decreases and s decreases.

3 AdaBoost [18 points]

1. [2 points] Consider some training point $(x^{(i)}, y^{(i)})$ to the AdaBoost algorithm. If for all t , the weak learner h_t learned during training at time t correctly classifies $h_t(x^{(i)}) = y^{(i)}$, will there eventually be a finite time t such that the weight assigned to $x^{(i)}$ in the training distribution ω_t reaches *exactly* 0? Briefly justify why or why not.

2. [2 points] Karthik believes that if the ensemble learned by AdaBoost reaches perfect training accuracy, the training can be stopped since all weak learners created in subsequent iterations will be identical (i.e., they will produce the same output on any input). Michael disagrees with Karthik in this regard. Who is correct in this argument? Assume we are using deterministically selected weak learners.

3. Assume we use a deterministic training procedure for weak learners. Suppose for some iteration t' of AdaBoost we find that the weak classifier $h_{t'}$ learned by the algorithm at time t' has error $\epsilon_{t'} = 0.5$ on the training distribution weighted by $\omega_{t'}$.

- (a) [2 points] What importance $\alpha_{t'}$ will AdaBoost assign to the classifier $h_{t'}$ from above?

- (b) [2 points] **Select all that apply:** In which of the following cases is $\omega_{t'+1}^{(i)} > \omega_{t'}^{(i)}$ (in other words, in which of the following cases will the weight on $(x^{(i)}, y^{(i)})$ *strictly* increase from time step t' to $t' + 1$)?

- ☐ $h_{t'}(x^{(i)}) = y^{(i)}$ ($h_{t'}$ classifies $x^{(i)}$ correctly)
☐ $h_{t'}(x^{(i)}) \neq y^{(i)}$ ($h_{t'}$ classifies $x^{(i)}$ incorrectly)
☐ None of the above

- (c) [2 points] **Select all that apply:** Which of the following are true about the next iteration of the AdaBoost algorithm?

- ☐ The errors $\epsilon_{t'+1}$ and $\epsilon_{t'}$ are equivalent
☐ The weak learners $h_{t'+1}$ and $h_{t'}$ will be equivalent (i.e., they will have the same output for every input)
☐ None of the above

4. In the following question, we will examine the generalization error of AdaBoost using a concept known as the *classification margin*.

Throughout the question, use the following definitions:

- T : The number of iterations used to train AdaBoost.
- N : The number of training samples.
- $S = \{(x^{(1)}, y^{(1)}), \dots, (x^{(N)}, y^{(N)})\}$: The training samples with binary labels ($y^{(i)} \in \{-1, +1\}$).
- $\omega_t^{(i)}$: The weight assigned to training example i at time t . Note that $\sum_i \omega_t^{(i)} = 1$.
- h_t : The weak learner constructed at time t (a function $X \rightarrow \{-1, +1\}$).
- ϵ_t : The weighted (by ω_t) error of h_t .
- $Z_t = 2\sqrt{\epsilon_t(1 - \epsilon_t)}$: The normalization factor for the distribution update at time t .
- $\alpha_t = \frac{1}{2} \ln((1 - \epsilon_t)/\epsilon_t)$: The weight assigned to the learner h_t in the composite hypothesis.
- $H_t(x) = (\sum_{t'=1}^t \alpha_{t'} h_{t'}(x)) / (\sum_{t'=1}^t \alpha_{t'})$: The majority vote of the weak learners, rescaled based on the total weights.
- $g_t(x) = \text{sign}(H_t(x))$: The voting classifier decision function.

For a binary classification task, assume that we use a probabilistic classifier that provides a probability distribution over the possible labels (i.e. $p(y|x)$ for $y \in \{+1, -1\}$). The classifier output is the label with highest probability. We define the *classification margin* for an input as the signed difference between the probability assigned to the correct label and the incorrect label $p_{\text{correct}} - p_{\text{incorrect}}$, which takes on values in the range $[-1, 1]$. Recall from lecture that $\text{margin}_t(x^{(i)}, y^{(i)}) = y^{(i)} H_t(x^{(i)})$.

- (a) **[2 points]** Recall the update AdaBoost performs on the distribution of weights:

$$\begin{aligned} \bullet \omega_1^{(i)} &= 1/N \\ \bullet \omega_{t+1}^{(i)} &= \omega_t^{(i)} \frac{\exp(-y^{(i)} \alpha_t h_t(x^{(i)}))}{Z_t} = \frac{1}{N} \left(\prod_{t'=1}^t \frac{1}{Z_{t'}} \right) \exp(-\sum_{t'=1}^t y^{(i)} \alpha_{t'} h_{t'}(x^{(i)})) \end{aligned}$$

We define $C_{t+1} = \frac{1}{N} \left(\prod_{t'=1}^t \frac{1}{Z_{t'}} \right)$ and $M_{t+1}(i) = -\sum_{t'=1}^t y^{(i)} \alpha_{t'} h_{t'}(x^{(i)})$. We then have

$$\omega_{t+1}^{(i)} = C_{t+1} \exp(M_{t+1}(i))$$

Let $\alpha = \sum_{t'=1}^t \alpha_{t'}$. Rewrite $M_{t+1}(i)$ in terms of $\text{margin}_t(x^{(i)}, y^{(i)})$ and α . (Hint: first rewrite $M_{t+1}(i)$ in terms of $y^{(i)}, \alpha, H_t, x^{(i)}$, then apply our given formula for the margin).

- (b) **[2 points] Select one:** Note that C_{t+1}, α are treated as positive constants with respect to the input points. Using the classification margin and the above formulation of the weights assigned by AdaBoost, fill in the blanks to describe which points AdaBoost assigns high weight to at time t .

At time t , AdaBoost assigns higher weight to points $x^{(i)}$ with _____ value of margin on the current ensemble classifier (i.e., $\text{margin}_t(x^{(i)}, y^{(i)})$).

- ☐ higher absolute
- ☐ higher signed
- ☐ lower absolute
- ☐ lower signed

- (c) **[2 points] Select one:** This weighting behavior causes the margins of the points you chose in part (b) to _____.

- ☐ increase
- ☐ decrease
- ☐ stay the same

- (d) **[2 points] Select all the apply:** How does this change in the margins explain the empirical result of test error continuing to decrease after training error has converged?

- ☐ AdaBoost can continue to increase the confidence of its predictions, particularly on lower confidence training examples, which makes it less likely at test time to misclassify points similar to those it has seen in the training set.
- ☐ Adaboost continues to increase confidence of its predictions on high confidence training examples only, leading to a highly accurate classifier which, at test time, will outperform the training error.
- ☐ Adaboost lowers the confidence in high confidence areas by continuing to train on low confidence training examples. This averages out the confidence between high and low confidence training examples, overall creating a more generalizable classifier.
- ☐ The test error does not continue to decrease after the training error has converged as the model stops updating once we reach convergence.
- ☐ None of the above.

4 Collaboration Questions

1. (a) Did you receive any help whatsoever from anyone in solving this assignment?
(b) If you answered 'yes', give full details (e.g. "Jane Doe explained to me what is asked in Question 3.4")

2. (a) Did you give any help whatsoever to anyone in solving this assignment?
(b) If you answered 'yes', give full details (e.g. "I pointed Joe Smith to section 2.3 since he didn't know how to proceed with Question 2")

3. (a) Did you find or come across code that implements any part of this assignment?
(b) If you answered 'yes', give full details (book & page, URL & location within the page, etc.).