Introduction to Machine Learning (67577)

Exercise 4
PAC and Boosting

Semester B, 2020

For your convenience we add that from all of PAC questions only question 4 and 5 are on material of the second PAC lecture and you will see more about question 8 on the second PAC recitation. The practical adaboost question based on material of boosting lecture that you will see on 20/5.

PAC Learnability

1. Let A be a learning algorithm, $\mathcal{D}$ be any distribution, and our loss function is in the range $[0,1]$ (e.g., the 0-1 loss). Prove that the following two statements are equivalent:

   (a) For every $\epsilon, \delta > 0$, there exists $m(\epsilon, \delta)$ such that $\forall m \geqslant m(\epsilon, \delta)$:

   $$\underset{S \sim \mathcal{D}^m}{\mathbb{P}}[L_{\mathcal{D}}(A(S)) \leqslant \epsilon] \geqslant 1 - \delta$$

   (b)

   $$\lim_{m \to \infty} \mathbb{E}_{S \sim \mathcal{D}^m}[L_{\mathcal{D}}(A(S))] = 0$$

   Hint: Use Markov's inequality.

2. **Sample Complexity of Concentric Circles in the Plane** Let $\mathcal{X} = \mathbb{R}^2$, $\mathcal{Y} = \{0,1\}$ and let $\mathcal{H}$ be the class of concentric circles in the plane, i.e., $\mathcal{H} = \{h_r : r \in \mathbb{R}_+\}$, where $h_r(x) = \mathbf{1}[\|x\|_2 \leqslant r]$. Prove that $\mathcal{H}$ is PAC learnable and its sample complexity is bounded by

   $$m_{\mathcal{H}}(\epsilon, \delta) \leqslant \frac{\log(1/\delta)}{\epsilon} .$$

   Note: Please do not use VC dimension arguments but instead prove the claim directly by showing a specific algorithm and analyzing its sample complexity.

   Hint: Remember that for every ϵ,

   $$1 - \epsilon \leqslant e^{-\epsilon}$$

VC dimension

3. **Boolean Conjunctions** Let $\mathcal{X} = \{0,1\}^d$ and $\mathcal{Y} = \{0,1\}$, and assume $d \geqslant 2$. Each sample $(\mathbf{x}, y) \in \mathcal{X} \times \mathcal{Y}$ consists of an assignment to d boolean variables ($\mathbf{x}$) and a label (y). For each boolean variable x_k, $k \in [d]$, there are two literals: x_k and $\overline{x}_k = 1 - x_k$. The class $\mathcal{H}_{\text{con}}$ is defined by boolean conjunctions over any subset of these $2d$ literals. For example: let $d = 5$ and consider the hypothesis that labels $\mathbf{x}$ according to the following conjunction

$$x_1 \wedge x_2 \wedge \overline{x}_3$$

For $\mathbf{x} = (0,1,1,1,1)$ the label would be 0, and for $\mathbf{x} = (1,1,0,0,0)$ the label would be 1. Compute the VC dimension of $\mathcal{H}_{\text{con}}$ and prove your answer.

Agnostic PAC

We highly recommend you to answer both questions but you are only required to submit one of the fallowing questions 4 and 5.

4. Prove that if $\mathcal{H}$ has the uniform convergence property with function $m_{\mathcal{H}}^{UC} : (0,1)^2 \to \mathbb{N}$ then $\mathcal{H}$ is Agnostic-PAC learnable with sample complexity $m_{\mathcal{H}}(\epsilon, \delta) \leqslant m^{UC}(\epsilon/2, \delta)$.

5. Let $\mathcal{H}$ be a hypothesis class over a domain $Z = \mathcal{X} \times \{\pm 1\}$, and consider the 0-1 loss function. Assume that there exists a function $m_{\mathcal{H}}(\epsilon, \delta)$, for which it holds that for every distribution $\mathcal{D}$ over Z there is an algorithm A with the following property: when running A on $m \geqslant m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples drawn from $\mathcal{D}$, it is guaranteed to return, with probability at least $1 - \delta$, a hypothesis $h_S : \mathcal{X} \to \{\pm 1\}$ with $L_{\mathcal{D}}(h_S) \leqslant \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon$. Is $\mathcal{H}$ agnostic PAC learnable? Prove or show a counter example.

Monotonicity

We highly recommend you to answer both questions but you are only required to submit one of the fallowing questions 6 and 7.

6. **Of Sample Complexity** Let $\mathcal{H}$ be a hypothesis class for a binary classification task. Suppose that $\mathcal{H}$ is PAC learnable and its sample complexity is given by $m_{\mathcal{H}}(\cdot, \cdot)$. Show that $m_{\mathcal{H}}$ is monotonically non-increasing in each of its parameters. That is, show that given $\delta \in (0,1)$, and given $0 < \epsilon_1 \leqslant \epsilon_2 < 1$, we have that $m_{\mathcal{H}}(\epsilon_1, \delta) \geqslant m_{\mathcal{H}}(\epsilon_2, \delta)$. Similarly, show that given $\epsilon \in (0,1)$, and given $0 < \delta_1 \leqslant \delta_2 < 1$, we have that $m_{\mathcal{H}}(\epsilon, \delta_1) \geqslant m_{\mathcal{H}}(\epsilon, \delta_2)$.

7. **of VC-Dimension** Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two classes for binary classification, such that $\mathcal{H}_1 \subseteq \mathcal{H}_2$. Show that $\mathrm{VC}\,\mathcal{H}_1 \leqslant \mathrm{VC}\,\mathcal{H}_2$.

Theoretical Claim

8. Let X be a sample space and $\mathcal{Y} = \{\pm 1\}$. Let $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ be a hypothesis class. For $C \subset \mathcal{X}$, recall the notation $\mathcal{H}_C$ for the restriction of $\mathcal{H}$ to the subset C. Define the function $\tau_m(\mathcal{H}) : \mathbb{N} \to \mathbb{N}$ corresponding to $\mathcal{H}$ to be

$$\tau_{\mathcal{H}}(m) := \max\left\{ |\mathcal{H}_C| \,\middle|\, C \subseteq \mathcal{X}\,,\ |C| = m \right\}.$$

(a) Explain, in your own words, the meaning of $\tau_{\mathcal{H}}$.

(b) Suppose that $VCdim(\mathcal{H}) = \infty$. Find an expression for the value of $\tau_{\mathcal{H}}(m)$ for $m \in \mathbb{N}$.

(c) Now suppose that $VCdim(\mathcal{H}) = d$. Find an expression for the value of $\tau_{\mathcal{H}}(m)$ for $m \leqslant d$.

(d) You will now prove the following important result: suppose that $VCdim(\mathcal{H}) = d$ and let $m > d$. Then

$$\tau_{\mathcal{H}}(m) \leqslant \left(\frac{em}{d}\right)^d ,$$

where e is the natural logarithm base. You'll do this in three steps:

i. Using induction, show that for any finite $C \subset \mathcal{X}$,

$$|\mathcal{H}_C| \leqslant \left| \left\{ B \subseteq C \,\middle|\, \mathcal{H} \text{ shutters } B \right\} \right| .$$

Hint: in the induction step divide $\mathcal{H}_C$ to two groups. one of them can be $\mathcal{H}_{C'}$ when $C' = \{c_2, ..., c_m\}$.

ii. Explain in your own words the meaning of this inequality.

iii. Show that, for any finite $C \subseteq \mathcal{X}$, we have

$$\left| \left\{ B \subseteq C \,\middle|\, \mathcal{H} \text{ shutters } B \right\} \right| \leqslant \sum_{k=0}^{d} \binom{m}{k}$$

iv. Use the following inequality (which you are not required to prove)

$$\sum_{k=0}^{d} \binom{m}{k} \leqslant \left(\frac{em}{d}\right)^d$$

to finish the proof that $\tau_{\mathcal{H}}(m) \leqslant \left(\frac{em}{d}\right)^d$.

(e) If $m = d$, does the inequality $\tau_{\mathcal{H}}(m) \leqslant \left(\frac{em}{d}\right)^d$ hold? If it does hold, is it tight?

(f) Characterize in words the behavior of $\tau_{\mathcal{H}}(m)$ for $m \leqslant VCdim(\mathcal{H})$ and for $m > VCdim(\mathcal{H})$. Can you use your characterization to offer an alternative definition of the VC-dimension $VCdim(\mathcal{H})$?

Separate the Inseparable - Adaboost

In this part we will boost a Decision Stump classifier by Adaboost and analyze its performance.

- As we saw in class, one of the main motivations for boosting is computational complexity. Try to solve the following questions in a reasonable runtime. Two tips to help reducing runtime:

  - Try to **Avoid for loops**. If you are not familiar with the concept of vectorization in numpy, this is the time for you to read about it. You can find many tutorials on the web, for example **this**. (loops are allowed as long as your code runs in reasonable time).

- You can use **line_profiler**. It is a tool that helps in finding the slower parts of your code. After you located the bottleneck try to think if you can accelerate it.

9. The file adaboost.py contains a template for implementation of the adaboost classifier. Fill in the template and implement the Adaboost algorithm as learned in class, where you assume WL is a weak-learner class that can be used as follows:

   h = WL(D,X,y) - constructs a weak classifier trained on X, y weighted by D.
   h.predict(X) - returns the classifier's prediction on a set X.

   In ex4_tools you are provided with such an implementation for a Decision Stump classifier, called DecisionStump.

   You may add methods as you see fit, but please implement the functions that are declared in the template.

10. In ex4_tools you are provided with the function generate_data. Use it to generate 5000 samples without noise (i.e. noise_ratio=0). Train an Adaboost classifier over this data. Use the DecisionStump weak learner mentioned above, and $T = 500$. Generate another 200 samples without noise ("test set") and plot the training error and test error, as a function of T. Plot the two curves on the same figure.

11. Plot the decisions of the learned classifiers with $T \in \{5, 10, 50, 100, 200, 500\}$ together with the test data. You can use the function decision_boundaries together with plt.subplot for this purpose.

12. Out of the different values you used for T, find $\hat{T}$, the one that minimizes the test error. What is $\hat{T}$ and what is its test error? Plot the decision boundaries of this classifier together with the training data.

13. Look into the AdaBoost: Take the weights of the samples in the last iteration of the training (D^T). Plot the training set with size proportional to its weight in D^T, and color that indicates its label (again, you can use decision_boundaries). Oh! we cannot see any point! the weights are to small... so we will normalize them: D = D / np.max(D) * 10. What do we see now? can you explain it?

14. Repeat 10,11,12,13 with noised data. Try noise_ratio=0.01 and noise_ratio=0.4.

    - Add all the graphs to the pdf.
    - Describe the changes.
    - Explain 10 in terms of the bias complexity tradeoff.
    - Explain the differences in 12.