Lecture 11: Online Learning

Instructor: Aditya Bhaskara *Scribe: Maks Cegielski-Johnson*

CS 5966/6966: Theory of Machine Learning

January 1st, 2017

Abstract

In this lecture, we considered online learning in the case of not having a consistent hypothesis in the hypothesis class. We also compared the regret bound between deterministic and randomized algorithms.

1 Introduction

As we have previously seen, in the online learning model: points arrive and the learner predicts the label, with the true label being revealed later. The goal of online learning is to compete with the best single hypothesis "in hindsight" after T rounds of learning, where we view each hypothesis as an *expert*. We define *regret* as the difference between "the number of mistakes the learner made" and "the number of mistakes the best expert made".

2 Finite Set of Hypotheses

2.1 theorem. *If there exists hypothesis $h \in \mathcal{H}$ that makes no mistakes, then we can ensure that the learning algorithm A makes at most $\log |\mathcal{H}|$ mistakes overall.*

From this theorem, the idea for an algorithm is to select what the majority of the "non-eliminated" experts say (and there always is an expert that is never eliminated). If the algorithm makes a mistake, then that means we can eliminate at least half of the hypotheses, which also made mistakes.

3 Infinite Set of Hypotheses

Suppose we know there is a *consistent* hypothesis, can we ensure that the learning algorithm only makes $O(1)$ mistakes? In the last lecture we saw the Littlestone dimension, and we saw that

$$\text{L-Dim} \leq \text{VC-Dim}$$

4 Non-consistent Finite Set of Hypotheses

What if there is no perfect expert? Can we ensure that

*Turns out this is **not possible***

$$\#\text{-mistakes}(Alg) < \#\text{-mistakes}(h_{\text{best}}) + \text{small}, \qquad \text{where "small" is } \log n$$

This is known as *regret*.

We have previously seen the idea of eliminating hypotheses that made a mistake. What if we keep track of how many mistakes each hypothesis has made? At each step, each hypothesis h_i is assigned some weight w_i, where w_i is small if h_i made many mistakes. In other words, reweight hypotheses rather than eliminating.

5 Multiplicative Weight Algorithm

Suppose we have $\mathcal{H} = \{h_1, h_2, \ldots, h_N\}$ and initial weights $\{w_1^{(1)}, w_1^{(2)}, \ldots, w_1^{(N)}\}$ with each $w_1^{(i)} = 1$ for each h_i. And, suppose that if a hypothesis h_i makes a mistake, then $w_2^{(i)} = w_1^{(i)} \cdot (1 - \eta)$ for some $\eta \in (0, 1)$. This means that after t steps, h_i has weight $w_t^{(i)} = (1 - \eta)^{\#\text{-mistakes}(h_i)}$.

This gives us the following algorithm:

- At each step t, keep track of $w_t^{(i)}$ for each hypothesis h_i
- Prediction at time t is

$$\begin{cases} 0 & \text{if } \sum_{i \text{ who predict 0 at } t} w_t^{(i)} \geq \sum_{j \text{ who predict 0 at } t} w_t^{(j)} \\ 1 & \text{otherwise.} \end{cases}$$

6 Analysis of Multiplicative Weight Algorithm

What happens every time we make a mistake? Let's define the "sum of weights" (also known as *potential*) as

This plays the role of the number of hypotheses which are not eliminated

$$(1) \quad \Phi_t = \sum_i w_t^{(i)} \qquad \text{at step } t$$

And let's call $\Phi_t^{(0)}$ the weights of hypothesis that predict 0 at time t, and similarly $\Phi_t^{(0)}$ the weights of the hypothesis that predict 1 at step t.

Suppose $\Phi_t^{(0)} > \Phi_t^{(1)}$. Then, if the learning algorithm predicted 0, and it was wrong, we have that

$$\Phi_{t+1} = \Phi_t^{(1)} + (1 - \eta)\Phi_t^{(0)}$$

If we made a mistake, in the worst case, then

$$\Phi_{t+1} \leq \frac{1}{2}\Phi_t + \frac{1}{2}\Big(1 - \eta\Big)\Phi_t = \Big(1 - \frac{\eta}{2}\Big)\Phi_t$$

We have the $|\mathcal{H}|$ term because at $t = 0$, each $w_i = 1$ for each hypothesis in $\mathcal{H}$

So after T steps,

$$\Phi_T \leq |\mathcal{H}| \cdot \Big(1 - \frac{\eta}{2}\Big)^{\#\text{-mistakes}(Alg)}$$

Say $\exists h$ that made only a "few" mistakes

Is there a lower bound on Φ_t?
If there exists a best hypothesis h_i that made $\leq k$ mistakes, then

$$w_T^{(i)} = (1 - \eta)^k \implies \Phi_T \geq (1 - \eta)^k \implies (1 - \eta)^k \leq |\mathcal{H}| \leq \Big(1 - \frac{\eta}{2}\Big)^m$$

where $m = \#\text{-mistakes}(Alg)$. Then,

$$k \log(1-\eta) \leq m \log(1-\eta/2) + \log|\mathcal{H}|$$

and so this implies

$$m \leq k \frac{\log(1-\eta)}{\log(1-\eta/2)} - \frac{\log|\mathcal{H}|}{\log(1-\eta/2)}$$

and then we have

$\log(1-x) \approx -x$ *for small* x

$$\frac{\log(1-\eta)}{\log(1-\eta/2)} \leq (2+\eta) \qquad \text{and} \qquad \frac{\log|\mathcal{H}|}{\log(1-\eta/2)} \approx 2\frac{\log|\mathcal{H}|}{\eta}$$

6.1 THEOREM. *For any η, the weighted majorty algorithm achieves the guarantee:*

$$\#\text{-mistakes}(Alg) \leq (2+\eta) \cdot \#\text{-mistakes}(h_{opt}) + \frac{1}{\eta}\log|\mathcal{H}|$$

Notice that the optimum-mistake-count term has a factor of 2. We are interested in whether we can reduce this value.

7 Randomized Algorithm

7.1 THEOREM. *No deterministic algorithm can achieve a number of mistakes $< 2 \cdot$ $\#\text{-mistakes}(opt)$.*

Proof (Cover Example)
Consider $\mathcal{H} = \{h_1, h_2\}$ where h_1 always predicts 1 and h_2 always predicts 0. Then, suppose we have any deterministic algorithm A and we get learn and predict on T examples. Now, suppose that the true label is always the opposite of what A predicts. Then, it must be the case that

$$\#\text{-mistakes}(A) = T \qquad \text{and} \qquad \#\text{-mistakes}(\text{opt}) \leq \frac{T}{2}$$

And this is a 2-factor loss $\square$.

The model we want to consider is one where the adversary (who knows the true label function f) knows what the learning algorithm is, but does not see the coin-tosses. Suppose the algorithm is

$$\begin{cases} 0 & \text{with probability } p \\ 1 & \text{with probability } 1-p. \end{cases}$$

What happens in the cover example? The claim is that the algorithm will do as well as the optimal hypothesis in this model.

Question. In this setting, can we beat the factor of 2? That is, can we achieve

$$\mathbb{E}[\#\text{-mistakes}(A)] \leq (1+\epsilon)\#\text{-mistakes}(opt) + \frac{\log|\mathcal{H}|}{\epsilon}$$

8 Weighted Majority - Version 2

A natural idea is to treat the weights as probabilities. Consider the following algorithm:

- Start with $w_0^{(i)} = 1, \ \forall i$
- At time t:
  - Algorithm picks h_i with probability $\frac{w_t^{(i)}}{\Phi_t}$ and outputs h_i output
  - For each $j \in [|\mathcal{H}|]$,

$$w_{t+1}^{(j)} = \begin{cases} (1-\eta)w_t^{(j)} & \text{if } j \text{ was wrong} \\ w_t^{(j)} & \text{otherwise.} \end{cases}$$

8.1 theorem. *The expected regret of the algorithm satisfies the "right" bound:*

$$\mathbb{E}[\#\text{-}mistakes(A)] \leq (1+2\eta)\#\text{-}mistakes(opt) + \frac{\log|\mathcal{H}|}{\eta}$$