Stat 928: Statistical Learning Theory **Lecture: 18**

Mistake Bound Model, Halving Algorithm, Linear Classifiers, & Perceptron

Instructor: Sham Kakade

1 Introduction

This course will be divided into 2 parts. In each part we will make different assumptions about the data generating process:

Online Learning No assumptions about data generating process. Worst case analysis. Fundamental connections to Game Theory.

Statistical Learning Assume data consists of independently and identically distributed examples drawn according to some fixed but *unknown* distribution.

Our examples will come from some space $\mathcal{X} \times \mathcal{Y}$. Given a *data set*

$$\{(x_t, y_t)\}_{t=1}^T \in (\mathcal{X} \times \mathcal{Y})^T ,$$

our goal is to predict y_{T+1} for a new point x_{T+1}. A *hypothesis* is simply a function $h : \mathcal{X} \to \mathcal{Y}$. Sometimes, a hypothesis will map to a set $\mathcal{D}$ (for decision space) larger than $\mathcal{Y}$. Depending on the nature of the set $\mathcal{Y}$, we get special cases of the general prediction problem.

Binary classification $\mathcal{Y} = \{-1, +1\}$

Multiclass classification $\mathcal{Y} = \{1, 2, \ldots, K\} =: [K]$ for $K \geq 3$

Regression $\mathcal{Y} = [-B, B]$ or $\mathcal{Y} = \mathbb{R}$

A set of hypotheses is often called a *hypotheses class*. If the range of a hypothesis is $\{-1, +1\}$ (or $\{0, 1\}$) then it also called a *concept*. A concept can be identified with the subset of $\mathcal{X}$ on which it is 1.

2 Mistake Bound Model

In this model, learning proceeds in rounds, as we see examples one by one. Suppose $\mathcal{Y} = \{-1, +1\}$. At the beginning of round t, the learning algorithm $\mathcal{A}$ has the hypothesis h_t. In round t, we see x_t and predict $h_t(x_t)$. At the end of the round, y_t is revealed and $\mathcal{A}$ makes a mistake if $h_t(x_t) \neq y_t$. The algorithm then updates its hypothesis to h_{t+1} and this continues till time T.

Suppose the labels were actually produced by some function f in a given concept class $\mathcal{C}$. Then it is natural to bound the total number of mistakes the learner commits, no matter how long the sequence. To this end, define

$$\text{mistake}(\mathcal{A}, \mathcal{C}) := \max_{f \in \mathcal{C}, T, x_{1:T}} \sum_{t=1}^{T} \mathbf{1}\left[h_t(x_t) \neq f(x_t)\right] \ .$$

We can now define what it means for an algorithm to learn a class in the *mistake bound model*.

Definition 2.1. *An algorithm $\mathcal{A}$* learns a class $\mathcal{C}$ with mistake bound M *iff*

$$\text{mistake}(\mathcal{A}, \mathcal{C}) \leq M \ .$$

Note that we are ignoring efficiency issues here. We have not said anything about the amount of computation $\mathcal{A}$ has to do in each round in order to update its hypothesis from h_t to h_{t+1}. Setting this issue aside for a moment, we have a remarkably simple algorithm HALVING $(\mathcal{C})$ that has a mistake bound of $\lg(|\mathcal{C}|)$ for any finite concept class $\mathcal{C}$.

For a finite set $\mathcal{H}$ of hypotheses, define the hypothesis $\text{majority}\left(\mathcal{H}\right)$ as follows,

$$\text{majority}\left(\mathcal{H}\right)(x) := \begin{cases} +1 & |\{h \in \mathcal{H} \mid h(x) = +1\}| \geq |\mathcal{H}|/2 \ , \\ -1 & \text{otherwise} \ . \end{cases}$$

Algorithm 1 HALVING $(\mathcal{C})$

```
$\mathcal{C}_1 \leftarrow \mathcal{C}$
$h_1 \leftarrow \text{majority}\left(\mathcal{C}_1\right)$
for $t = 1$ to $T$ do
  Receive $x_t$
  Predict $h_t(x_t)$
  Receive $y_t$
  $\mathcal{C}_{t+1} \leftarrow \{f \in \mathcal{C}_t \mid f(x_t) = y_t\}$
  $h_{t+1} \leftarrow \text{majority}\left(\mathcal{C}_{t+1}\right)$
end for
```

Theorem 2.2. *For any finite concept class $\mathcal{C}$, we have*

$$\text{mistake}(\text{HALVING}\left(\mathcal{C}\right), \mathcal{C})) \leq \lg |\mathcal{C}| \ .$$

Proof. The key idea is that if the algorithm makes a mistake then at least half of the hypothesis in $\mathcal{C}_t$ are eliminated. Formally,

$$h_t(x_t) \neq y_t \quad \Rightarrow \quad |\mathcal{C}_{t+1}| \leq |\mathcal{C}_t|/2 \ .$$

Therefore, denoting the number of mistakes up to time t by M_t,

$$M_t := \sum_{t=1}^{T} \mathbf{1}\left[h_t(x_t) \neq y_t\right] \ ,$$

we have

$$|\mathcal{C}_{t+1}| \leq \frac{|\mathcal{C}_1|}{2^{M_t}} = \frac{|\mathcal{C}|}{2^{M_t}} \tag{1}$$

Since there is an $f \in \mathcal{C}$ which perfectly classifies all x_t, we also have

$$1 \leq |\mathcal{C}_{t+1}| \ . \tag{2}$$

Combining (1) and (2), we have

$$1 \leq \frac{|\mathcal{C}|}{2^{M_t}} \ ,$$

which gives $M_t \leq \lg(|\mathcal{C}|)$. □

3 Linear Classifiers and Margin

Let us now look at a concrete example of a concept class. Suppose $\mathcal{X} = \mathbb{R}^d$ and we have a vector $w \in \mathbb{R}^d$. We define the hypothesis,

$$h_w(x) = \operatorname{sgn}(w \cdot x) ,$$

where $\operatorname{sgn}(z) = 2 \cdot \mathbf{1}[z \geq 0] - 1$ gives the sign of z. With some abuse of terminology, we will often speak of "the hypothesis w" when we actually mean "the hypothesis h_w". The class of *linear classifiers* in the (uncountable) concept class

$$\mathcal{C}_{\text{lin}} := \left\{ h_w \,\middle|\, w \in \mathbb{R}^d \right\} .$$

Note that w and cw yield the same linear classifier for any $c > 0$.

Suppose we have a data set that is *linearly separable*. That is, there is a w^* such that,

$$\forall t \in [T],\ y_t = \operatorname{sgn}(w^* \cdot x_t) . \tag{3}$$

Separability means that $y_t(w^* \cdot x_t) > 0$ for all t. The minimum value of this quantity over the data set is referred to as the *margin*. Let us make the assumption that the margin is at least γ for some $\gamma > 0$.

Assumption M. *There exists a $w^* \in \mathbb{R}^d$ for which* (3) *holds. Further assume that*

$$\min_{t \in [T]} y_t(w^* \cdot x_t) \geq \gamma , \tag{4}$$

for some $\gamma > 0$.

Define

$$\|x_{1:t}\| := \max_{t \in [T]} \|x_t\| .$$

We now show that under Assumption M, our simple halving algorithm can be used with a suitable finite subset of $\mathcal{C}_{\text{lin}}$ to derive a mistake bound. Let W_γ be those w such that w_i is of the form $m\gamma/2\|x_{1:T}\|d$ for some

$$m \in \{-\lceil 2\|x_{1:t}\|\|w^*\|d/\gamma\rceil, \ldots, -1, 0, +1, \ldots, \lceil 2\|x_{1:t}\|\|w^*\|d/\gamma\rceil\} .$$

In other words, since each coordinate of w^* is in the range $[-\|w^*\|, \|w^*\|]$, we have discretized that interval at a scale of $\gamma/2\|x_{1:T}\|d$. We want to run the halving algorithm on the (finite) concept class,

$$\mathcal{C}_{\text{lin}}^{\gamma} := \{h_w \,|\, w \in W_\gamma\} .$$

The size of this class is $\left(\lceil \frac{4\|x_{1:t}\|\|w^*\|d}{\gamma} \rceil + 1\right)^d$. Note that there exists a $\tilde{w} \in W_\gamma$ such that,

$$\forall i \in [d],\ |w^*_i - \tilde{w}_i| \leq \gamma/2\|x_{1:T}\|d .$$

Thus, we have, for any $t \in [T]$,

$$\begin{aligned}
|y_t(\tilde{w} \cdot x_t) - y_t(w^* \cdot x_t)| &= |\tilde{w} \cdot x_t - w^* \cdot x_t| \\
&\leq \sum_{i=1}^{d} |\tilde{w}_i - w_i| \cdot |x_{t,i}| \\
&\leq \sum_{i=1}^{d} \frac{\gamma}{2\|x_{1:T}\|d} \|x_t\| \\
&\leq \gamma/2 .
\end{aligned}$$

This, together with Assumption R, implies that $y_t(\tilde{w} \cdot x_t) \geq \gamma/2 > 0$. Thus, there exists a hypothesis in $\mathcal{C}_{\text{lin}}^{\gamma}$ that classifies the data set perfectly. Theorem 2.2 immediately gives the following corollary.

Corollary 3.1. *Under Assumption M,* HALVING $(\mathcal{C}_{\text{lin}}^{\gamma})$ *makes at most*

$$d \lg\left(\left\lceil \frac{4d\|x_{1:t}\| \cdot \|w^*\|}{\gamma} \right\rceil + 1\right)$$

mistakes.

This bound is nice because even though we had an uncountable concept class to begin with, the margin assumption allowed us to work with a finite subset of the concept class and we were able to derive a mistake bound. However, the result is unsatisfactory because running the halving algorithm on $\mathcal{C}_{\text{lin}}^{\gamma}$ is extremely inefficient. One might wonder if one can use the special structure of the space of linear classifiers to implement the halving algorithm more efficiently. Indeed, it possible to implement a variant of the halving algorithm efficiently using the ellipsoid method developed for the linear programming feasibility problem.

Note that the mistake bound depends explicitly on the dimension d of the problem. We would also like to be able to give a dimension independent mistake bound. Indeed, a classic algorithm called PERCEPTRON has such a mistake bound.

4 The Perceptron Algorithm

Algorithm 2 PERCEPTRON

```
$w_1 \leftarrow \mathbf{0}$
for $t = 1$ to $T$ do
    Receive $x_t \in \mathbb{R}^d$
    Predict $\text{sgn}(w_t \cdot x_t)$
    Receive $y_t \in \{-1, +1\}$
    if $\text{sgn}(w_t \cdot x_t) \neq y_t$ then
        $w_{t+1} \leftarrow w_t + y_t x_t$
    else
        $w_{t+1} \leftarrow w_t$
    end if
end for
```

The following theorem gives a dimension independent bound on the number of mistakes the PERCEPTRON algorithm makes.

Theorem 4.1. *Suppose Assumption M holds. Let*

$$M_T := \sum_{t=1}^{T} \mathbf{1}\left[\text{sgn}(w_t \cdot x_t) \neq y_t\right]$$

denote the number of mistakes the PERCEPTRON *algorithm makes. Then we have,*

$$M_T \leq \frac{\|x_{1:T}\|^2 \cdot \|w^*\|^2}{\gamma^2} \ .$$

Proof. The key idea of the proof is to look at how the quantity $w^* \cdot w_t$ evolves over time. We first provide an lower bound for it. Define $m_t = \mathbf{1}\left[\text{sgn}(w_t \cdot x_t) \neq y_t\right]$. Note that $w_{t+1} = w_t + y_t x_t m_t$ and $M_T = \sum_t m_t$. We have,

$$\begin{aligned} w^* \cdot w_{t+1} &= w^* \cdot w_t + y_t x_t m_t \\ &= w^* \cdot w_t + y_t(w^* \cdot x_t) m_t \\ &\geq w^* \cdot w_t + \gamma m_t \ . && \text{(Assumption M)} \end{aligned}$$

Unwinding the recursion, we get

$$w^* \cdot w_{T+1} \geq w^* \cdot w_1 + \gamma M_T = \gamma M_T \ . \tag{5}$$

Now, we use Cauchy-Schwarz inequality to get the upper bound,

$$w^* \cdot w_{T+1} \leq \|w^*\| \cdot \|w_{T+1}\| \ . \tag{6}$$

Moreover,

$$\begin{aligned} \|w_{t+1}\|^2 &= \|w_t + y_t x_t m_t\|^2 \\ &= \|w_t\|^2 + 2y_t(w_t \cdot x_t)m_t + \|x_t\|^2 m_t \\ &\leq \|w_t\|^2 + 0 + \|x_{1:T}\|^2 m_t \ , \end{aligned}$$

where the last step follows because $y_t(w_t \cdot x_t) < 0$ when a mistake is made and $\|x_t\| \leq \|x_{1:T}\|$. Unwinding the recursion once again, we get,

$$\|w_{T+1}\|^2 \leq \|w_1\|^2 + \|x_{1:T}\|^2 M_T = \|x_{1:T}\|^2 M_T \ . \tag{7}$$

Combining (5), (6) and (7) gives,

$$\gamma M_T \leq w^* \cdot w_{T+1} \leq \|w^*\| \cdot \|w_{T+1}\| \leq \|w^*\| \cdot \|x_{1:T}\| \sqrt{M_T} \ .$$

This implies that $M_T \leq \|w^*\|^2 \cdot \|x_{1:T}\|^2 / \gamma^2$. □