COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire
Scribe: Yannan Wang

Lecture #16
April 3, 2014

1 Introduction

The goal of our online learning scenario from last class is C comparing with best expert and do as well as the best expert.

An alternative scenario is there is no single good expert. But we could form a committee of experts and they might be much better.

We'll formalize this as follows:

- We have N experts.
- For $t = 1, ..., T$ we get $\mathbf{x}_t \in \{1, -1\}^N$
  Note: $\mathbf{x}_t$: a set of predictions
  N: dimension
  i_{th} component: prediction of expert i
- In each round, learner predicts $\hat{y_t} \in \{1, -1\}$
- In each round, we observe the outcome $y_t \in \{1, -1\}$
- The above is the same; what we changed is the assumption of data. We assume that there is a perfect committee, i.e. a weighted sum of experts that are always right. Formally, this means that $\mathbf{u} \in \mathbb{R}^N$,

$$\forall t : y_t = sign(\sum_{i=1}^{N} u_i x_{t,i}) = sign(\mathbf{x}_t \cdot \mathbf{u})$$

$$\iff y_t(\mathbf{u} \cdot \mathbf{x}_t) > 0$$

  Geometrically, the perfect committee means that there is a linear threshold that separates the 1 points and -1 points, generated by the appropriate weighted sum of the experts.

2 How to do updates

We are focusing on $\mathbf{w}_t$, the prediction of $\mathbf{u}$. It is sort of a "guess" of the correct weighting of the experts. We will update the weighting on each round. Today we are looking at two algorithms. For each algorithm, we only need to focus on

$$(1) initialize$$

$$(2) update$$

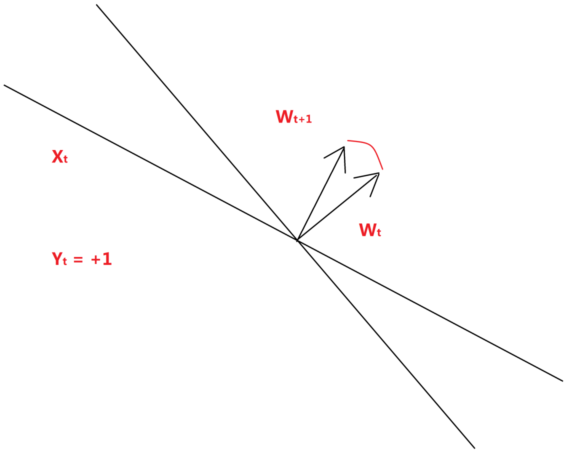

Figure 1: Perceptron geometric intuition: tiping the hyperplane

2.1 Perceptron

The first way to update weights will give us an algorithm called Perceptron. The update rules are as follows:

- *Initialize*: $\mathbf{w}_1 = \mathbf{0}$
- *Update*: If mistake($\iff \hat{y}_t \neq y_t \iff y_t(\mathbf{w}_t \cdot \mathbf{x}_t) \leq 0$),

  $$\mathbf{w}_{t+1} = \mathbf{w}_t + y_t \mathbf{x}_t$$

  else,

  $$\mathbf{w}_{t+1} = \mathbf{w}_t$$

  Not adjusting the weights when there are no mistakes makes the algorithm *conservative*; the algorithm ignores the correctly classifying samples.

The intuition is that in case of a wrong answer we "shift" the weights on all the experts in the direction of the correct answer. Figure 1 gives a geometrical intuition of the Perceptron algorithm. Here $y_t = +1$, when $(\mathbf{x}_t, y_t)$ is classified incorrectly, then we add $\mathbf{x}_t y_t$ to $\mathbf{w}_t$ to such a direction that is more likely to correctly classify $(\mathbf{x}_t, y_t)$ next time; we are shifting the hyperplane defined by $\mathbf{w}_t$ in such a direction that we are more likely to correctly classify $\mathbf{x}_t$.

Now let's state a theorem to formally analyze the performance of the Perceptron algorithm. However, first we will make a a few assumptions:

- Mistakes happens in every round. This is because no algorithmic change happens during other rounds. So: $T = \#$ of rounds $= \#$ of mistakes.
- We normalize the vector of predictions $\mathbf{x}_t$, so that $\|\mathbf{x}_t\|_2 \leq 1$.

- We normalize the vector of weights for the perfect committee, so that$\|\mathbf{u}\|_2 = 1$. (This is fine because the value of the sign function will not be affected by this normalization.)
- We make the assumption that the points are linearly separable with margin at least δ: $\exists \delta, \mathbf{u} \in \mathrm{I\!R}^{\mathrm{N}}, \forall \mathrm{t} : \mathrm{y_t}(\mathbf{u}\cdot\mathbf{x}_\mathrm{t}) \geq \delta > 0$. Note that this assumption is with loss of generality.

Theorem 2.1 *Under the assumptions above, $T = \#$ mistakes, we have*

$$T \leq \frac{1}{\delta^2}$$

Proof : In order to prove this, we will find some quantity that depends on the state of the algorithm at time t, upper bound and lower bound it, and derive a bound from there. The quantity here is Φ_t, which is cosine of the angle Θ between $\mathbf{w}_t$ and $\mathbf{u}$. More formally,

$$\Phi_t = \frac{\mathbf{w}_t \cdot \mathbf{u}}{\|\mathbf{w}_t\|_2} = \cos\Theta \leq 1$$

Now for the lower bound, we will prove that

$$\Phi_{T+1} \geq \sqrt{T}\delta$$

We will do this in two parts — by lower bounding the numerator of Φ_t and by upper bounding the denominator.

- **step 1: $\mathbf{w}_{T+1} \cdot \mathbf{u} \geq T\delta$:**

  $$\mathbf{w}_{t+1} \cdot \mathbf{u} = (\mathbf{w}_t + y_t\mathbf{x}_t) \cdot \mathbf{u} = \mathbf{w}_t \cdot \mathbf{u} + y_t(\mathbf{u} \cdot \mathbf{x}_t) \geq \mathbf{w}_t \cdot \mathbf{u} + \delta$$

  The inequality is by the $4th$ assumption above. Initially we have set $\mathbf{w_1} \cdot \mathbf{u} = 0$, thus the above bound implies that $\mathbf{w}_{T+1} \cdot \mathbf{u} \geq T\delta$.

- **step 2: $\|\mathbf{w}_{T+1}\|^2 \leq T$:**

  $$\|\mathbf{w}_{t+1}\|^2 = \mathbf{w}_{t+1} \cdot \mathbf{w}_{t+1} = (\mathbf{w}_t + y_t\mathbf{x}_t) \cdot (\mathbf{w}_t + y_t\mathbf{x}_t) = \|\mathbf{w}_t\|_2^2 + 2y_t(\mathbf{x}_t \cdot \mathbf{w}_t) + \|\mathbf{x}_t\|^2$$

Since we have made the assumption that we get a mistake at each round, $y_t(\mathbf{x}_t \cdot \mathbf{w}_t) \leq 0$, and from the normalization assumption, $\|\mathbf{x}_t\|_2^2 \leq 1$, so that we get $\|\mathbf{w}_{t+1}\|^2 \leq \|\mathbf{w}_t\|_2^2 + 1$. Initially we have set $\|\mathbf{w}_1\|_2^2 = 0$, so we get $\|\mathbf{w}_{T+1}\|_2^2 \leq T$

Now we put step 1 and step 2 together, $1 \geq \Phi_{T+1} \geq \frac{T\delta}{\sqrt{T}}$, i.e. $T \leq \frac{1}{\delta^2}$. □

Let $\mathcal{H}$ be the hypothesis space and $M_{perceptron}(\mathcal{H})$ be the number of mistakes made by the Perceptron algorithm. As a simple consequence of the above, since the VC dimension of the hypothesis space is upper bounded by the number of mistakes the algorithm makes, we get the VC dimension of threshold functions with margin at least δ is at most $\frac{1}{\delta^2}$:

$$VC\text{-}dim(\mathcal{H}) \leq opt(\mathcal{H}) \leq M_{perceptron}(\mathcal{H}) \leq \frac{1}{\delta^2}$$

Now consider a scenario where the target $\mathbf{u}$ consists of 0s and 1s, and the number of 1s in the vector is k.

$$\mathbf{u} = \frac{1}{\sqrt{k}}(0 \quad 1 \quad 0 \quad 0 \quad 1 \quad ...)$$

Note that here $\frac{1}{\sqrt{k}}$ is for normalization. Think of k as being small compared to N, the number of experts, i.e. it could be a very sparse vector. This is also one example of the problems we earlier examined — the k experts are the "perfect" committee. We have,

$$\mathbf{x}_t = \frac{1}{\sqrt{N}}(+1, \quad -1, \quad -1, \quad +1, \quad ...)$$

$$y_t = sign(\mathbf{u} \cdot \mathbf{x}_t)$$

$$y_t(\mathbf{u} \cdot \mathbf{x}_t) \geq \frac{1}{\sqrt{kN}}$$

Note that here $\frac{1}{\sqrt{N}}$ is for normalization. So using $\frac{1}{\sqrt{kN}}$ as δ, by Theorem 2.1, the Perceptron algorithm would make at most kN mistakes. However this is not good — consider interpreting the experts as features, and we have millions of irrelevant features, and the committee is the important (maybe a dozen) features. We get a linear dependencies on N, which is usually large.

Motivated by this example, we present another update algorithm, called the Winnow algorithm, which will get a better bound.

2.2 Winnow Algorithm

- **Initialize**:

  $$\forall i, w_{1,i} = \frac{1}{N}$$

  we start with a uniform distribution over all experts.

- **Update:** If we make a mistake,

  $$\forall i : w_{t+1,i} = \frac{w_{t,i} \cdot e^{\eta y_t x_t}}{Z_t}$$

  Here η is a parameter we will define later, and Z_t is a normalization factor. Else,

  $$\mathbf{w}_{t+1} = \mathbf{w}_t$$

This update rule is like exponential punishment for the experts that are wrong. If we ignore the normalization factors, the above update is equivalent to $w_{t+1,i} = w_{t,i}e^{\eta}$, if i predicts correctly, and $w_{t+1,i} = w_{t,i}e^{-\eta}$ otherwise. Ignoring the normalization factor, we could see it as $w_{t+1,i} = w_{t,i}$, if i predicts correctly, and $w_{t+1,i} = w_{t,i}e^{-2\eta}$ otherwise. This is the same as the weighted majority vote.

Before stating the formal theorem for the Winnow algorithm, we make a few assumptions without loss of generality:

- We make mistake at every round.
- $\forall t : \|\mathbf{x}_t\|_\infty \leq 1$.
- $\exists \delta, \mathbf{u} : \forall t : y_t(\mathbf{u} \cdot \mathbf{x}_t) \geq \delta > 0$.
- $\| \mathbf{u} \|_1 = 1$ and $\forall i$: $u_i \geq 0$.

Notice here we used L_1 and L_∞ norm here instead of the L_2 norm that we used in Perceptron algorithm.

Theorem 2.2 *Under the assumptions above, We have the following upper bound on the number of mistakes:*

$$T \leq \frac{\ln N}{\eta\delta + \ln(\frac{2}{e^{\eta}+e^{-\eta}})}$$

If we choose an optimal η to minimize the bound , we get when $\eta = \frac{1}{2}\ln(\frac{1+\delta}{1-\delta})$,

$$T \leq \frac{2\ln N}{\delta^2}$$

Proof The approach is similar to the previous one. We use a quantity Φ_t, which we both upper and lower-bound. The quantity we use here is $\Phi_t = RE(\mathbf{u} \parallel \mathbf{w}_t)$. Immediately we have, $\Phi_t \geq 0$ for all t.

$$\begin{aligned}
\Phi_{t+1} - \Phi_t &= \sum_i u_i \ln(\frac{u_i}{w_{t+1,i}}) - \sum_i u_i \ln(\frac{u_i}{w_{t,i}}) \\
&= \sum_i u_i \ln(\frac{w_{t,i}}{w_{t+1,i}}) \\
&= \sum_i u_i \ln(\frac{Z_t}{e^{\eta y_t x_{t,i}}}) \\
&= \sum_i u_i \ln Z_t - \sum_i u_i \ln e^{\eta y_t x_{t,i}} \\
&= \ln Z_t - \eta y_t(\mathbf{u} \cdot \mathbf{x}_t) \\
&\leq \ln Z_t - \eta\delta
\end{aligned} \tag{1}$$

The last inequality follows from the margin property we assumed. Now let's approximate Z_t. We know that Z is the normalization factor and can be computed as:

$$Z = \sum_i w_i e^{\eta y x_i} \tag{2}$$

Note that here we are dropping the subscript t for simplicity; Z and w_i are same as Z_t and $w_{t,i}$. We will bound the exponential term by a linear function, as illustrated in figure 2:

$$e^{\eta x} \leq (\frac{1+x}{2})e^{\eta} + (\frac{1-x}{2})e^{-\eta}, \quad for -1 \leq x \leq 1.$$

Using this bound, we have:

$$\begin{aligned}
Z &= \sum_i w_i e^{\eta y x_i} \\
&\leq \sum_i w_i (\frac{1+yx_i}{2})e^{\eta} + \sum_i w_i (\frac{1-yx_i}{2})e^{-\eta} \\
&= \frac{e^{\eta}+e^{-\eta}}{2}\sum_i w_i + \frac{e^{\eta}-e^{-\eta}}{2} y \sum_i w_i x_i \\
&= \frac{e^{\eta}+e^{-\eta}}{2} + \frac{e^{\eta}+e^{-\eta}}{2} y(\mathbf{w} \cdot \mathbf{x}) \\
&\leq \frac{e^{\eta}+e^{-\eta}}{2}
\end{aligned} \tag{3}$$

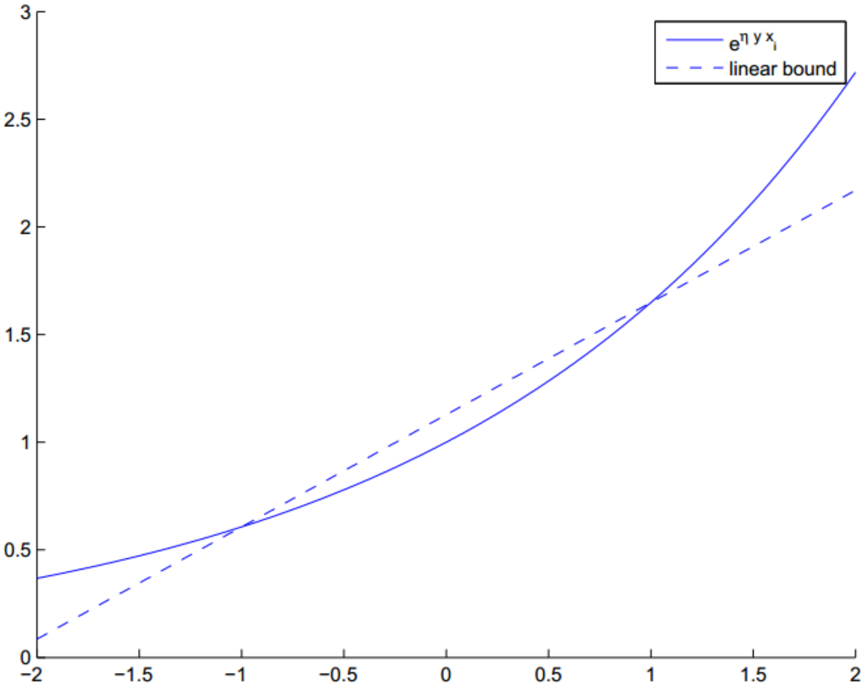

Figure 2: Using linear function to bound exponential function

The last inequality comes from the assumption that the expert makes a wrong prediction every time, so the second term is less than 0. So we have,

$$\begin{aligned}\Phi_{t+1} - \Phi_t &\leq \ln Z_t - \eta\delta \\ &\leq \ln(\frac{e^{\eta} + e^{-\eta}}{2}) - \eta\delta = -C\end{aligned} \tag{4}$$

Note that here $\ln(\frac{e^{\eta}+e^{-\eta}}{2}) - \eta\delta$ is a constant and let's make it equals $-C$. So for each round Φ_t is decreasing by at least $C = \ln(\frac{2}{e^{\eta}+e^{-\eta}}) + \eta\delta$.

In the next class, we will finish the proof of Theorem 2.2 and we will study a modified version of Winnow Algorithm called Balanced Winnow Algorithm that gets rid of the assumption that $\forall i : u_i \geq 0$.