COS 511: Foundations of Machine Learning

Rob Schapire
Scribe: Joel Predd

Lecture #13
March 25, 2003

1 Learning with Expert Advice (continued)

New goal: Compare the performance of the learner to the best *combination/committee* of experts.

Formulation:

N experts
for $t = 1, 2, ..., T$
  get $\vec{x_t} \in \{-1, +1\}^N$
  learner predicts $\hat{y_t} \in \{-1, +1\}$
  observe outcome $y_t \in \{-1, +1\}$
  assume $\exists \vec{u} \in \mathbb{R}^N$ s.t. $y_t = sgn(\vec{u} \cdot \vec{x_t})$

Essentially, we assume that there is a weighted majority vote amongst all N experts that yields perfect prediction performance. Our task is to find this optimal weighted majority vote.

General Algorithm:

Initialize $\vec{w_1}$
On round t:
  predict $\hat{y_t} = sgn(\vec{w_t} \cdot \vec{x_t})$
  update $\vec{w}_{t+1}$ using $\vec{w_t}, \vec{x_t}, y_t$

As always, the key questions are:

1. How should we initialize $\vec{w_1}$?
2. How should we update $\vec{w_t}$?

2 Perceptron Algorithm

The perceptron algorithm is an algorithm used to find a separating hyperplane for linearly separable data. We formulate it in a general case where our observations $\vec{x_t}$ take on values in $\mathbb{R}^N$ and prove a theorem that bounds the number of errors made by the algorithm. Then, we will apply the result to our special-case of interest when $\vec{x_t} \in \{-1, +1\}^N$ and $\vec{x_t}$ represents the vector of responses of our experts.

As an aside, note that the perceptron algorithm is a *conservative* algorithm. This is to say that it ignores samples that it classifies correctly. Note that any mistake bounded algorithm can be converted into an algorithm that is conservative.

Perceptron Algorithm:

- Initialize: $\overrightarrow{w_1} = 0$
- Update:

  if $\widehat{y_t} = y_t$
  $$\overrightarrow{w_{t+1}} = \overrightarrow{w_t}$$
  else ($\widehat{y_t} \neq y_t$)
  $$\overrightarrow{w_{t+1}} = \overrightarrow{w_t} + y_t \overrightarrow{x_t}$$

Note (intuition for update rule):

$$\overrightarrow{w}_{t+1} \cdot \overrightarrow{x_t} = (\overrightarrow{w_t} + y_t \cdot \overrightarrow{x_t}) \cdot \overrightarrow{x_t} = \overrightarrow{w_t} \cdot \overrightarrow{x_t} + y_t \parallel \overrightarrow{x_t} \parallel_2^2 \tag{1}$$

When, for instance, $y_t = 1$ and $\widehat{y_t} \neq 1$, then:

$$\overrightarrow{w}_{t+1} \cdot \overrightarrow{x_t} = \overrightarrow{w_t} \cdot \overrightarrow{x_t} + \parallel \overrightarrow{x_t} \parallel_2^2 \geq \overrightarrow{w_t} \cdot \overrightarrow{x_t} \tag{2}$$

Thus, we see that our adjustment makes $\overrightarrow{w}_{t+1} \cdot \overrightarrow{x_t}$ "more positive" than $\overrightarrow{w}_t \cdot \overrightarrow{x_t}$. In effect, $sgn(\overrightarrow{w}_{t+1} \cdot \overrightarrow{x_t})$ is "closer" to labelling $\overrightarrow{x_t}$ correctly. Similar intuition holds when $y_t = -1$ and $\widehat{y_t} \neq -1$.

2.1 Analysis

Assumptions:

- $\parallel \overrightarrow{x_t} \parallel_2 \leq 1$ (as in SVM)
- $\exists \overrightarrow{u} \in \mathbb{R}^N$, $\exists \delta > 0$ s.t. $y_t(\overrightarrow{u} \cdot \overrightarrow{x_t}) \geq \delta > 0$ for $\forall t = 1, ..., T$
- $\parallel \overrightarrow{u} \parallel_2 = 1$

Theorem 1: # of mistakes made by the perceptron algorithm $\leq \frac{1}{\delta^2}$.

Choose our potential function as: $\Phi_t = \frac{\overrightarrow{w_t} \cdot \overrightarrow{u}}{\|\overrightarrow{w_t}\|_2} = \cos(\text{angle between } \overrightarrow{u} \text{ and } \overrightarrow{w_t}) \leq 1$.

Proof: Assume there is a mistake on every round. We can make this assumption due to the fact that the algorithm is conservative and the weights are not adjusted when there isn't a mistake.

Let $T = \#$ of mistakes.

Step 1: $\overrightarrow{w}_{T+1} \cdot \overrightarrow{u} \geq T\delta$.
Proof:

$$\begin{aligned}
\overrightarrow{w}_{T+1} \cdot \overrightarrow{u} &= (\overrightarrow{w}_T + y_T \overrightarrow{x}_T) \cdot \overrightarrow{u} \\
&= \overrightarrow{w}_T \cdot \overrightarrow{u} + y_T(\overrightarrow{x}_T \cdot \overrightarrow{u}) \\
&\geq \overrightarrow{w}_T \cdot \overrightarrow{u} + \delta \\
&= (\overrightarrow{w}_{T-1} + y_{T-1} \overrightarrow{x}_{T-1}) \cdot \overrightarrow{u} + \delta \\
&= \overrightarrow{w}_{T-1} \cdot \overrightarrow{u} + y_{T-1}(\overrightarrow{x}_{T-1} \cdot \overrightarrow{u}) + \delta
\end{aligned}$$

$$
\begin{aligned}
&\geq \overrightarrow{w}_{T-1} \cdot \overrightarrow{u} + \delta + \delta \\
&. \\
&. \quad (recursion) \\
&. \\
&\geq \overrightarrow{w_1} \cdot \overrightarrow{u} + T\delta \\
&= T\delta
\end{aligned}
$$

Step 2: $\| \overrightarrow{w}_{T+1} \|_2^2 \leq T$.
Proof:

$$
\begin{aligned}
\| \overrightarrow{w}_{T+1} \|_2^2 &= (\overrightarrow{w}_T + y_T \overrightarrow{x}_T) \cdot (\overrightarrow{w}_T + y_T \overrightarrow{x}_T) \\
&= \| \overrightarrow{w}_T \|_2^2 + 2 y_T \overrightarrow{w}_T \cdot \overrightarrow{x_T} + y_T^2 \| \overrightarrow{x}_T \|_2^2 \\
&\leq \| \overrightarrow{w}_T \|_2^2 + 0 + 1 \\
&. \\
&. \quad (recursion) \\
&. \\
&\leq \| \overrightarrow{w_1} \|_2^2 + T \\
&= T
\end{aligned}
$$

So, combining steps 1 and 2, we have:

$$
\delta\sqrt{T} = \frac{T\delta}{\sqrt{T}} \leq \frac{\overrightarrow{w}_{T+1} \cdot \overrightarrow{u}}{\| \overrightarrow{w}_{T+1} \|_2} = \Phi_{T+1} \leq 1 \tag{3}
$$

Thus,

$$
T \leq \frac{1}{\delta^2} \tag{4}
$$

and the proof is complete.

2.2 Committees of Experts

Let us relate the Perceptron Algorithm to the original problem.

In the originally stated problem, $\overrightarrow{x_t}$ will have the form $\frac{1}{\sqrt{N}}(+1, -1, -1, +1, ..., 1)$ (constant for normalization) and $\overrightarrow{u}$ will have the form $\frac{1}{\sqrt{K}}(0, 1, 0, 1, ..., 1)$ (constant for normalization). Here, K is the number of experts in the "best" subcommittee of N experts. We do *not* assume the learner has prior knowledge of K.

Note that $y_t(\overrightarrow{u} \cdot \overrightarrow{x_t}) \geq \frac{1}{\sqrt{NK}}$. Thus, using $\frac{1}{\sqrt{NK}}$ as δ, we see that all of the assumptions stated for Theorem 1 have been met. Then, by Theorem 1, if we use the Perceptron Algorithm to learn the best weighted majority vote amongst the panel of N experts, we can be assured that:

$$
\#\text{ of mistakes} \leq NK
$$

3 "Winnow" Algorithm

The "Winnow" Algorithm is another conservative algorithm for accomplishing the same goal. As before, we will formulate the model in a general case and prove a theorem bounding the number of mistakes made by the algorithm. Finally, we will apply the result to our case of interest: committees of experts.

Winnow Algorithm:

- Parameter $\eta > 0$
- Initialize: $w_{1,i} = \frac{1}{N}$, for $i = 1, ..., N$
- Predict $\widehat{y_t} = sgn(w_t \cdot x_t)$
- Update on mistake:

  $w_{t+1,i} = \frac{w_{t,i} \exp(\eta y_t x_{t,i})}{Z_t}$

  $Z_t = \sum_i w_{t,i} \exp(\eta y_t x_{t,i})$

In the original Winnow algorithm, $\widehat{y_t} = sgn(w_t \cdot x_t + \theta)$ for a threshold parameter θ; the algorithm manipulates both $\overrightarrow{w_t}$ and θ. We consider a special case where this threshold is assumed equal to zero.

Note (intuition for update rule):

$$\overrightarrow{x_t} \in \{-1, +1\}^N$$

$$w_{t+1,i} \propto \begin{cases} e^{\eta}, & y_t = x_{t,i} \\ e^{-\eta}, & y_t \neq x_{t,i} \end{cases} \tag{5}$$

$$w_{t+1,i} \propto \begin{cases} 1, & y_t = x_{t,i} \\ \beta = e^{-2\eta}, & y_t \neq x_{t,i} \end{cases} \tag{6}$$

So, we observe that it is simply the weighted majority algorithm discussed earlier.

3.1 Analysis

New Assumptions:

- $\| \overrightarrow{x_t} \|_\infty \leq 1$
- $\exists \overrightarrow{u} \in \mathbb{R}^{\mathrm{N}}$, $\exists \delta > 0$ s.t. $y_t(\overrightarrow{u} \cdot \overrightarrow{x_t}) \geq \delta > 0$ for $\forall t = 1, ..., T$
- $\| \overrightarrow{u} \|_1 = 1$
- $u_i \geq 0$ (can be removed)

Using previously derived results for weighted majority algorithms, we arrive at this theorem.

Theorem 2: # of mistakes made by the Winnow Algorithm $\leq \frac{\ln N}{\eta\delta + \ln(\frac{2}{e^{\eta} + e^{-\eta}})}$.

Since η is arbitrary, we can choose it so as to minimize this upper bound. Doing so yields:

# of mistakes made by the Winnow Algorithm $\leq \frac{2 \ln N}{\delta^2}$
achieved when $\eta = \frac{1}{2} \ln \frac{1+\delta}{1-\delta}$.

3.2 Committees of Experts

Let us now apply the Winnow algorithm to our original problem. Note that the meaning of δ has now changed due to the change in norms introduced within the assumptions.

From above, $\overrightarrow{x_t}$ will have the form $(+1, -1, -1, +1, ..., 1)$ and $\overrightarrow{u}$ will have the form $\frac{1}{K}(0, 1, 0, 1, ..., 1)$. Thus, $\parallel x_t \parallel_\infty \leq 1$ and $\parallel u \parallel_1 \leq \frac{1}{K}$. Thus, $y_t(\overrightarrow{u} \cdot \overrightarrow{x_t}) \geq \frac{1}{K}$. By noting that each of the assumptions have been met and using $\frac{1}{K}$ as our δ, we can apply the above theorem to the committee of experts problem to find that:

$$\# \text{ of mistakes made by the Winnow Algorithm} \leq 2\ln(N)K^2.$$

We note the bound is logarithmic in N for the Winnow algorithm as compared to linear in N for the perceptron algorithm.