Lecture 10. (Sketch.)

Today we’ll cover “online concept learning”.

▶ This is a classical online setting where the vector we receive \( x_i \), but then work only with the evaluations of some hypothesis/concept set \( H \):

\[
(h_1(x_i), h_2(x_i), \ldots, h_d(x_i)) \in \{0,1\}^d.
\]

(Sometimes it’s \([-1,+1]^d\).)

▶ Sometimes these predictors are called “experts”, and sometimes we assume there is a perfect expert, meaning \( \exists h \in H \) with \( h(x_i) = y_i \) for all \( i \).

[ I discussed a few things at the start of lecture which I’ll omit here, for instance a guarantee for Perceptron in the nonseparable case. ]

1. Two baseline methods.

Consider the setting that \( \exists \bar{h} \in H, \bar{h}(x_i) = y_i \) for all \( i \). This means that each iteration can permanently ignore any \( h \in H \) which makes any mistakes.

**CONSISTENT.**

1. Initialize \( H_0 = H \).

2. For \( i \in \{1,2,\ldots\} \):
   2.1 Receive \( x_i \).
   2.2 Choose any \( h_i \in H_{i-1} \), output \( \hat{y}_i := h_i(x_i) \).
   2.3 Receive \( y_i \), construct \( H_i \):

\[
H_i := \begin{cases} 
H_{i-1} & \hat{y}_i = y_i, \\
H_{i-1} \setminus \{h_i\} & \hat{y}_i \neq y_i.
\end{cases}
\]

Rather than removing only one hypothesis, we can remove all hypotheses that make a mistake on \( x_i \).

In general, the subset of \( H \) which is consistent with all examples seen up through time \( i \), meaning \( ((x_j,y_j))_{j \leq i} \), is called the **version space**: specifically,

\[
\{ h \in H : \forall j \leq i \cdot h(x_j) = y_j \}.
\]

We can update **CONSISTENT** to remove more hypotheses, but without another change we can still guarantee only \( |H_i| \leq |H_{i-1}| - I[\hat{y}_i \neq y_i] \).

**Theorem.** If \( \exists \bar{h}, \bar{h}(x_i) = y_i \), and \((\hat{y}_i)_{i \geq 1}\) are output by **CONSISTENT**, \n
\[
\sum_{i \geq 1} I[\hat{y}_i \neq y_i] \leq |H| - 1.
\]

**Proof.** By the update rule for \( H_i \),

\[
|H_i| = |H_{i-1}| - I[\hat{y}_i \neq y_i]
\]

which rearranges to give \( I[\hat{y}_i \neq y_i] \leq |H_{i-1}| - |H_i| \). Applying \( \sum_{i \leq t} \) to both sides,

\[
\sum_{i \leq t} I[\hat{y}_i \neq y_i] = \sum_{i \leq t} (|H_{i-1}| - |H_i|) = |H_0| - |H_t|
\leq |H| - |\bar{h}| = |H| - 1.
\]
We can remove more hypotheses by choosing $h_i$ more carefully.

**Halving.**

1. Initialize version space $H_0 = \mathcal{H}$.
2. For $i \in \{1, 2, \ldots\}$:
   1. Receive $x_i$.
   2. Choose majority label:
      $$\hat{y}_i := \arg \max_y \{ h \in \mathcal{H} : h(x_i) = y \}.$$
   3. Receive $y_i$, update version space $H_i$:
      $$H_i := \{ h \in H_{i-1} : h(x_i) = y_i \}.$$

**Remark** (comparison to Perceptron). Suppose linear separability: $\exists \bar{u}, \gamma$ such that $\|\bar{u}\| = 1$ and $\inf_i \langle \bar{u}, x_iy_i \rangle \geq \gamma > 0$, and $\sup_i \|x_iy_i\| \leq 1$. Recall that perceptron makes at most $1/\gamma^2$ mistakes, and uses $O(d)$ computation per round.

For **Consistent** and **Halving**, it suffices to choose $\mathcal{H}$ to be (the linear predictors corresponding to) a cover $\mathcal{W}$ of $\{ w \in \mathbb{R}^d : \|w\| = 1 \}$ at scale $\gamma/2$, since

$$\inf_{w \in \mathcal{W}} \inf_i \langle w, x_iy_i \rangle = \inf_{w \in \mathcal{W}} \inf_i \langle \bar{u}, x_iy_i \rangle + \langle w - \bar{u}, x_iy_i \rangle \geq \inf_{w \in \mathcal{W}} \inf_i \gamma - \|w - \bar{u}\|.$$ 

Thus $|\mathcal{H}| = O(1/\gamma)^d$, thus **Consistent** and **Halving** respectively make $O(1/\gamma)^d$ and $O(d \lg(1/\gamma))$ mistakes, but both have $O(1/\gamma)^d$ computation. Also, they must guess $\gamma$, for instance with a double (halving) trick.

**Theorem.** If $\exists \bar{h}, \tilde{h}(x_i) = y_i$, and $(\hat{y}_i)_{i \geq 1}$ are output by **Halving**, then

$$\sum_{i \geq 1} \mathbb{1}[\hat{y}_i \neq y_i] \leq \lg |\mathcal{H}|.$$

**Proof.** Since we chose the majority label, on mistake we know we remove at least half the hypotheses:

$$|\mathcal{H}_i| \leq |\mathcal{H}_{i-1}| 2^{-1}[\hat{y}_i \neq y_i],$$

which by induction gives

$$1 = |\{ \tilde{h} \}| \leq |\mathcal{H}_t| \leq |\mathcal{H}_0| \prod_{i=1}^t 2^{-1}[\hat{y}_i \neq y_i] = |\mathcal{H}| 2^{-\sum_{i=1}^t 1[\hat{y}_i \neq y_i]},$$

which rearranges to give the desired bound.

2. **Winnow.**

Now let’s suppose the perfect predictor is a logical or of $k$ elements of $\mathcal{H}$: $\tilde{h}(x) = h_1 \lor \cdots \lor h_k$.

- We can using **Halving** and make only $k \ln(d)$ mistakes, but we still spend $O(d^k)$ computation.
- Let’s make an algorithm which maintains a candidate set of disjunction terms (initially everything). On iteration $i$:
  - If $y_i = -1$, remove any $h$ which (mistakenly) output $+1$.
  - If $y_i = +1$, we can’t be wrong (we started with everything, and never incorrectly remove), and we shouldn’t remove anything.

Even if $k = 1$, this can unfortunately take $d - 1$ not $O(\ln(d))$ mistakes: suppose the target disjunction is just $h_n(x)$, but the sequence of inputs is $(e_1, e_2, \ldots)$, all with label $-1$. 

- **Halving.**
Winnow will get roughly the same mistake bound as Halving, while simultaneously being computationally efficient. One of its tricks is to maintain a linear predictor rather than a disjunction.

**Remark.** Methods that learn a hypothesis outside the target class are called improper.

**Winnow.**

1. Initialize $w_j = 1$ for $j \in [d]$.
2. For $i \in \{1, 2, \ldots \}$ :
   1. Receive $x_i$, predict $\hat{y}_i := \text{sgn}(\sum_j w_j h_j(x_i) - d)$.
   2. Receive $y_i$; if $y_i \neq \hat{y}_i$, update $w$:

      $$w_j := \begin{cases} 
      2w_j & h_j(x_i) = +1, \\
      w_j & h_j(x_i) = 0. 
      \end{cases}$$

      (Note, $h_j(x) \in \{0, 1\}$.)

**Theorem.** If $(\hat{y}_i)_{i \geq 1}$ are computed by Winnow,

$$\sum_{i \geq 1} \mathbb{1}[y_i \neq \hat{y}_i] \leq 1 + 2k \lceil \lg d \rceil.$$ 

**Proof.**

- Each mistake when $y = +1$ doubles $w_j$ for some true disjunction term $h_j$, and they are never decreased when $y = -1$; together, mistake on $y = +1$ can happen at most \( r \lceil \lg(d) \rceil \) times.

- Each mistake when $y = -1$ has $\sum_j h_j(x) = 1 = \sum_j w_j h_j(x) > d$, thus $\|w\|_1$ decreases by at least $d$. Initially, $\|w\|_1 = d$, and each (of at most $P \leq r \lceil \lg(d) \rceil$) mistakes on positive add at most $d$, so the number of mistakes on negative, $N$, can not decrease $\|w\|_1$ below $0$, and so $N \leq 1 + P$.

**Remark.** This simple proof for disjunctions is by Avrim Blum. In class I discussed learning linear predictors (not just disjunctions), but the math is messy.