# COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire
Scribe: Evan Cofer

Lecture #3
February 12, 2018

## 1 Probably Approximately Correct (PAC) Learning

We say that a concept class $\mathcal{C}$ is PAC-learnable by a hypothesis space $\mathcal{H}$ if there exists an algorithm $\mathcal{A}$ such that for every target concept $c \in \mathcal{C}$, every $\epsilon > 0$, $\delta > 0$, and any target distribution $D$, $\mathcal{A}$ requires a sequence $S = \langle (x_1, c(x_1)), (x_2, c(x_2)) ..., (x_m, c(x_m)) \rangle$ of $m = \text{poly}\left(\frac{1}{\epsilon}, \frac{1}{\delta}, ...\right)$ examples where each $x_i$ is chosen independently from $D$ to produce a hypothesis $h \in \mathcal{H}$ for which $\Pr\left[\text{err}_D(h) \leq \epsilon\right] \geq 1 - \delta$ holds. Importantly, $D$ can be any distribution, so the PAC-learning model is considered to be a "distribution free" model. The variable $\delta > 0$ is used to define the confidence $1 - \delta$, and $\epsilon > 0$, defines the error. The sample size $m$ is allowed to get larger as $\epsilon$ and $\delta$ are made smaller since, for an algorithm to have greater accuracy (smaller $\epsilon$) or greater confidence (smaller $\delta$), it will typically require a greater amount of data. Finally, we say that $\mathcal{C}$ is efficiently PAC-learnable if $\mathcal{A}$ runs in $\text{poly}\left(\frac{1}{\epsilon}, \frac{1}{\delta}, ...\right)$ time.

## 2 Learning positive half-lines

We turn now to an example of a PAC-learnable concept class. Specifically, we consider the concept class of positive half-lines. A positive half-line is a ray extending rightwards (i.e. towards $+\infty$) from some real-valued point. All values to the right of this point are labeled positive. Values to the left are labeled negative. For terseness, we denote both the concept and threshold with $c$. For this example, our domain $\mathcal{X} = \mathbb{R}$, and our hypothesis space and concept class are the set of positive half lines, i.e. $\mathcal{H} = \mathcal{C} = \{\text{positive half-lines}\}$.

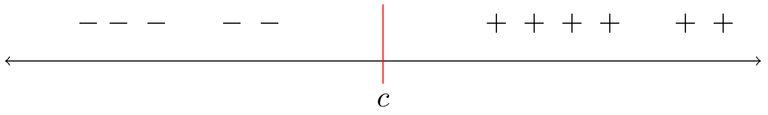

Figure 1: The target concept $c$ is shown on a number line.

One approach for choosing a hypothesis would be to take the arithmetic mean of the largest negative example seen and smallest positive example seen. Any value between the two aforementioned extremes would be a consistent hypothesis. We show an example of such a hypothesis in Figure 2 below.

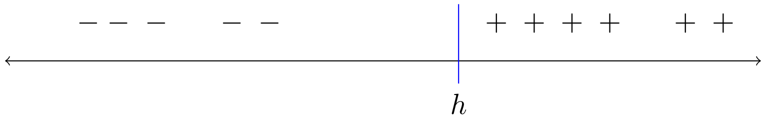

Figure 2: The hypothesis $h$ is shown on a number line.

Only points outside of $[c, h]$, the "error region", will be properly labeled by $h$. Since $\forall x \in [c, h], c(x) \neq h(x)$ holds, $\mathrm{err}_D(h)$ will be the probability mass in $[c, h]$. We provide a visualization of this in Figure 3.

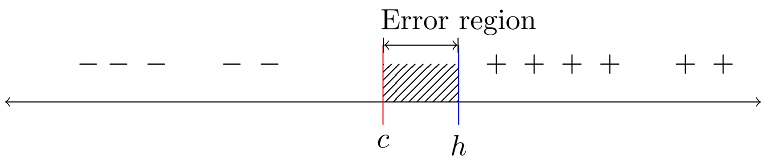

Figure 3: The target concept $c$ and a hypothesis $h$ are shown on a number line. The probability mass in the region $[c, h]$ is the error.

We herein refer to $h$ with error greater than $\epsilon$ as being "$\epsilon$-bad". PAC-learning can also be phrased as requiring $h$ to be $\epsilon$-good (i.e. not $\epsilon$-bad) with probability at least $1 - \delta$. If $h$ is too far to the left or right of $c$, then our error will be too high, and greater than $\epsilon$. We illustrate these bad cases in Figure 4 below, where it is immediately apparent that they are symmetric scenarios. We must show that, in most cases, this does not happen. That is, we must show $\Pr[\mathrm{err}_D(h) > \epsilon] \leq \delta$.

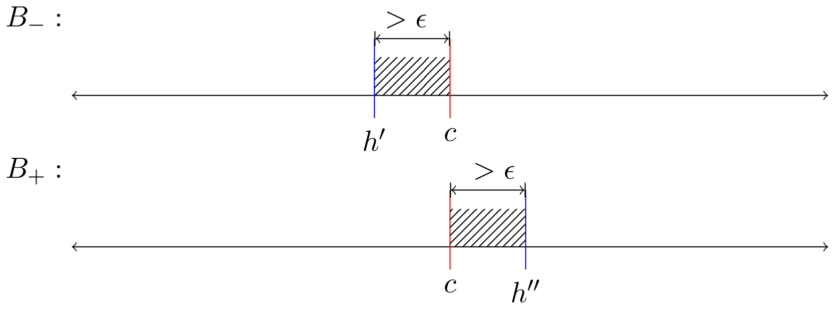

Figure 4: The target concept $c$ and two examples of bad events $B_-$ and $B_+$, with hypotheses $h'$ and $h''$ respectively.

Now, we consider the probability of $B_+$. Imagine the point found by sweeping out from $c$ until the probability mass is exactly $\epsilon$. We call this point $r_+$, and the region $[c, r_+]$ as $R_+$. A visual is presented in Figure 5.

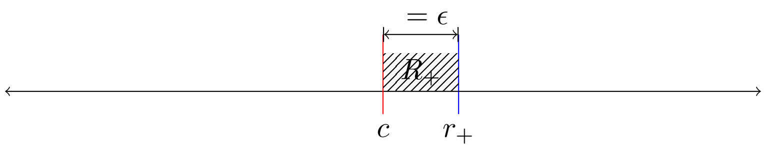

Figure 5: The region $R_+$ for which the probability mass is $\epsilon$.

Clearly, if one training example were to fall into $R_+$, then $h < r_+$, the error will be less than $\epsilon$, and $B_+$ does not occur. Thus, the probability of $B_+$ is at most the probability that no point lands in $[c, r_+]$. A single point $x_1$ falls into $R_+$ with probability $\epsilon$, so $\Pr[x_1 \notin R_+] = 1 - \epsilon$ as well. As all the points are independent and identically distributed (i.i.d.), it follows that

$$\begin{aligned}
& \Pr\left[x_1 \notin R_+ \wedge x_2 \notin R_+ \wedge \ldots \wedge x_m \notin R_+\right] \\
= \; & \Pr\left[x_1 \notin R_+\right] \cdot \Pr\left[x_2 \notin R_+\right] \cdot \ldots \cdot \Pr\left[x_m \notin R_+\right] \\
= \; & (1-\epsilon)^m
\end{aligned}$$

As such, the probability of $B_+$ is at most the probability of all of these, or $\Pr\left[B_+\right] \leq (1-\epsilon)^m$. Recall that there are two symmetric events, $B_+$ and $B_-$. By the union bound we know that $\Pr\left[B_+ \vee B_-\right] \leq \Pr\left[B_+\right] + \Pr\left[B_-\right]$, and so $\Pr\left[B_+ \vee B_-\right] \leq 2\left(1-\epsilon\right)^m$. As $\forall x, 1+x \leq e^x$, we get $\Pr\left[B_+ \vee B_-\right] \leq 2e^{-\epsilon m}$, which we would like to be $\delta$ at most. To satisfy this requirement, it must be that $m \geq \frac{1}{\epsilon} \ln \frac{2}{\delta}$. Recall that $\Pr\left[\mathrm{err}_D(h) > \epsilon\right] = \Pr\left[B_+ \vee B_-\right]$, and so $\Pr\left[\mathrm{err}_D(h) > \epsilon\right] \leq \delta$ if $m \geq \frac{1}{\epsilon} \ln \frac{2}{\delta}$. We have shown that $\mathcal{C}$ is PAC-learnable by $\mathcal{H}$.

■

## 3 Learning intervals

We now consider the concept class $\mathcal{C} = \{\text{intervals on } \mathbb{R}\}$, and again let $\mathcal{C} = \mathcal{H}$. As shown below in Figure 6, a target concept $c \in \mathcal{C}$ is an interval $[c_L, c_R]$ where $c_L, c_R \in \mathbb{R}$. All values that fall within the interval are labeled positive. All other values are labeled negative. Our algorithm $\mathcal{A}$ must find a hypothesis $h \in \mathcal{H}$, and this can be done in a similar manner as in the previous example. The proof that $\mathcal{A}$ is PAC-learnable by $\mathcal{H}$ is also similar.

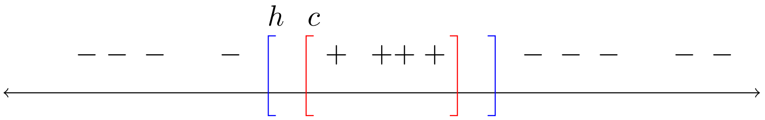

Figure 6: The target concept $c$ and the hypothesis $h$ are shown on a number line.

We sweep out an interval with probability mass $\frac{\epsilon}{2}$ on either side of $c_L$, and again for $c_R$. The two cases are symmetric. Clearly, the case of $c_L$ is the same as the case of the positive half-line shown above, and the same can be said for $c_R$. Although the PAC-learnability of $\mathcal{A}$ by $\mathcal{H}$ is within reach, we leave its proof as an exercise for the reader.

## 4 Learning axis-aligned rectangles

Finally, we consider the concept class $\mathcal{C} = \{\text{axis-aligned rectangles}\}$ where $\mathcal{H} = \mathcal{C}$ again. We now want an algorithm $\mathcal{A}$ that finds the smallest consistent rectangle $h \in \mathcal{H}$. A visual is presented below in Figure 7.

We now create four bands within $c$ along the top, right, bottom, and left edges. We specify that each band has probability mass of exactly $\frac{\epsilon}{4}$, as seen in Figure 8. If at least one point falls into each band, then $h$ will be $\epsilon$-good. There are four bad events, each corresponding to a point not falling into each of the four bands. The probability that $h$ is $\epsilon$-good is the probability that at least one point falls into each of the bands. It is apparent that this is a similar argument to the previous examples.

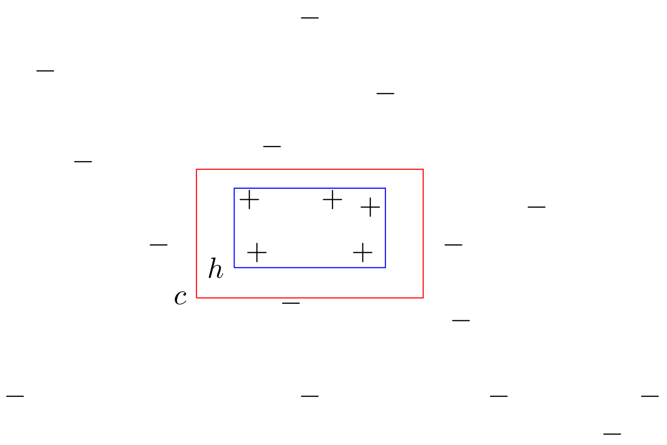

Figure 7: The target concept $c$ and the hypothesis $h$, both in $\mathbb{R}^2$, are shown.

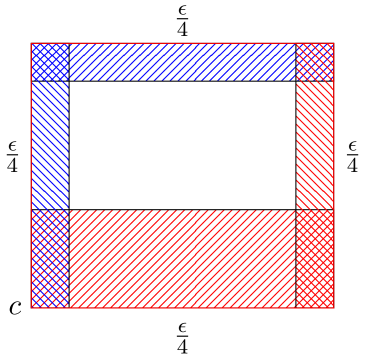

Figure 8: The target concept $c$ shown with four error regions, each having probability mass of $\frac{\epsilon}{4}$. It does not matter that these regions overlap.

## 5 General proof of PAC learnability for finite $|\mathcal{H}|$

Thus far, we have only considered proofs of PAC-learnability for very specific cases. This is a very ad hoc and inefficient manner of proving things. We are also interested in knowing when consistency is sufficient for learning in a general model (e.g. PAC-learning). We turn now to a theorem of PAC-learnability for hypothesis spaces of finite cardinality.

**Theorem 1** Suppose algorithm $\mathcal{A}$ finds hypothesis $h_A \in \mathcal{H}$ consistent with $m$ examples where $m \geq \frac{1}{\epsilon}\left(\ln|\mathcal{H}| + \ln\frac{1}{\delta}\right)$. Then $\Pr\left[\mathrm{err}_D(h_A) > \epsilon\right] \leq \delta$.

Note that this theorem does not directly involve a concept class $\mathcal{C}$ at all, but does apply whenever we manage to find a consistent hypothesis. We can rephrase it as follows: with probability of at least $1-\delta$, if $h_A \in \mathcal{H}$ is consistent, then $\mathrm{err}_D(h_A) \leq \frac{\ln|\mathcal{H}| + \ln\frac{1}{\delta}}{m}$.

This new term, $\ln|\mathcal{H}|$ is interesting, and it is related to a notion of complexity or how "not simple" our hypothesis is. Specifically, it is a measure of the complexity of $\mathcal{H}$. When the base for this logarithm is 2, the term is the number of bits needed to write names for each hypothesis $h \in \mathcal{H}$. We can also think of $\ln|\mathcal{H}|$ as a measure of the description length of the hypotheses in $\mathcal{H}$. Importantly, this speaks only to the complexity of $\mathcal{H}$, not the individual hypotheses.

To explore this more, Dr. Schapire asked students to consider the problem of determining binary labels (i.e. 0 or 1) for the set of integers $\{1, 2, ..., 30\}$. After everyone wrote down binary labels for $\{1, 2, ..., 30\}$, Dr. Schapire divided the examples into two sets: $\{1, 2, ..., 10\}$ for the training data, and $\{11, 12, ..., 20\}$ for the test data. The student hypothesis with the lowest training error had a training error of 20%, but a testing error of 40%. However, unbeknownst to the class, the proper labels had been generated by coin flips. As such, even hypotheses with low training error are expected to mislabel half of the test examples. When the hypothesis space grows, so too does the probability of finding a hypothesis that performs well on the training data by chance. However, such hypotheses will still have an expected test error of 50%. This means that larger hypothesis spaces will require more training data to avoid these bad hypotheses that only have low training error by chance.

### 5.1 Monotone conjunctions

We consider $\mathcal{C} = \{\text{monotone conjunctions of length } n\}$. Again, we let $\mathcal{C} = \mathcal{H}$. As each $h$ either includes or excludes a given variable, there are $2^n$ hypothesis and $|\mathcal{H}| = 2^n$. From previous lectures, we know of an algorithm $\mathcal{A}$ that finds a consistent hypothesis $h \in \mathcal{H}$. By Theorem 1, $m \geq \frac{1}{\epsilon}\left(n \ln 2 + \ln \frac{1}{\delta}\right)$ implies $\Pr\left[\text{err}_D(h_A) > \epsilon\right] \leq \delta$. This is polynomial w.r.t. $n$, $\frac{1}{\epsilon}$, $\frac{1}{\delta}$. We have shown monotone conjunctions to be PAC-learnable. ■