An Introduction to the Theory of Active Machine Learning

Robert D. Nowak
University of Wisconsin-Madison
rdnowak@wisc.edu

Abstract

This is a short introduction to some of the theory of active machine learning.

1 Motivation: Binary Search

Consider a function $f(x)$ defined on $x \in [0,1]$ that takes the value -1 for $x < t$ and $+1$ for $x \geq t$. Suppose we wish to locate the threshold t to within $\epsilon \in (0,1)$. A simple approach is to evaluate f at $k = \lceil \epsilon^{-1} \rceil$ points spaced uniformly between 0 and 1. This is essentially a *linear search* technique. It requires $O(\epsilon^{-1})$ function evaluations.

Alternatively, we can perform a binary search over the k points. First evaluate the point closest to $1/2$. If the evaluation is positive, then the threshold must be in the left half interval, otherwise it's in the right half interval. Repeat by evaluating at the midpoint of the indicated half interval. Continuing this process, it is easy to see that we can locate t to within ϵ using only $\log \epsilon^{-1}$ function evaluations.

The adaptive selection of function evaluations provides an exponential improvement relative to linear search. In the parlance of machine learning, adaptively selecting "examples" (points x) to "label" (i.e., $y = f(x)$) drastically reduces the number of labels needed to learn the target function to within a desired accuracy. This is called active learning, and we observe similar gains in more general settings. This note describes some of the core ideas in the theory of active learning by considering this simple 1-d classification in more detail.

2 Confidence Intervals

The key technical ingredient in the analysis of learning algorithms is the notion of confidence intervals. Consider a Gaussian random varaible $y \sim \mathcal{N}(\mu, \sigma^2)$. Straightforward consideration of the Gaussian density function provides probabilistic bounds on the deviations of x from the mean μ:

$$\begin{aligned} \mathbb{P}(y - \mu \geq t) &\leq \frac{1}{2}\exp(-\frac{t^2}{2\sigma^2}) \\ \mathbb{P}(\mu - y \geq t) &\leq \frac{1}{2}\exp(-\frac{t^2}{2\sigma^2}) \end{aligned}$$

Now suppose that we observe $y_1, \ldots, y_m \overset{\text{iid}}{\sim} \mathcal{N}(\mu, 1)$ and consider the sample mean $\widehat{\mu}_m = \frac{1}{m}\sum_{i=1}^m y_i$. Recall that in this case $\widehat{\mu}_m \sim \mathcal{N}(\mu, 1/m)$ and therefore the bounds above tell us that

$$\begin{aligned} \mathbb{P}(\widehat{\mu}_m - \mu \geq t) &\leq \frac{1}{2}\exp(-\frac{mt^2}{2}) \\ \mathbb{P}(\mu - \widehat{\mu}_m \geq t) &\leq \frac{1}{2}\exp(-\frac{mt^2}{2}) \end{aligned}$$

In words, the distribution of the sample mean is "concentrating" in the vicinity of the true mean exponentially fast with m.

The same sort of bounds hold for averages of independent random variables with distributions with tails decaying at least as fast as the Gaussian tail (i.e., tails bounded by a function of the form $c_1 e^{-c_0 t^2}$, where c_0 and c_1 are positive constants). Such distributions are called *subGaussian*, and in particular all bounded random variables are subGaussian. Suppose $y_1, \ldots, y_m$ are iid random variables bounded in $[-1, 1]$. Then Chernoff's bound states

$$\begin{aligned} \mathbb{P}(\widehat{\mu}_m - \mu \geq t) &\leq \exp(-\frac{mt^2}{2}) \\ \mathbb{P}(\mu - \widehat{\mu}_m \geq t) &\leq \exp(-\frac{mt^2}{2}) \, , \end{aligned}$$

the same bounds (up to the constant factor) that we had for Gaussian averages. This shouldn't be surprising since we know that averages of independent random variables tend to a Gaussian distribution.

We can combined the bounds above as follows.

$$\begin{aligned} \mathbb{P}(|\widehat{\mu}_m - \mu| \geq t) &= \mathbb{P}(\mu - \widehat{\mu}_m \geq t \text{ or } \widehat{\mu}_m - \mu \geq t) \\ &\leq \mathbb{P}(\mu - \widehat{\mu}_m \geq t) + \mathbb{P}(\widehat{\mu}_m - \mu \geq t) \\ &\leq 2\exp(-\frac{mt^2}{2}) \, . \end{aligned}$$

This bound yields a confidence interval of the following form. Let $\delta = 2\exp(-\frac{mt^2}{2})$. Solving for t we have

$$t = \sqrt{\frac{2\log(2/\delta)}{m}} \, ,$$

and thus with probability at least $1 - \delta$

$$\widehat{\mu}_m - \sqrt{\frac{2\log(2/\delta)}{m}} \leq \mu \leq \widehat{\mu}_m + \sqrt{\frac{2\log(2/\delta)}{m}} \, .$$

In words, the true mean is probably within an interval of length $2\sqrt{\frac{2\log(2/\delta)}{m}}$ centered at $\widehat{\mu}_m$. Next we will apply such intervals to analyze a noise-tolerant version of binary search.

3 Noisy Binary Search

Recall the binary search problem from above, and suppose that instead of perfect function evaluations we instead observe $f(x)$ with probability $(1 + \Delta)/2$ and $-f(x)$ with probability $(1 - \Delta)/2$. In other words, label y we receive is probably correct, but not perfectly correct. To compensate for this, suppose that we can query the point x multiple times and each time receive an independent response that is probably correct. We can then average the responses to try to decide whether $f(x)$ is truly $+1$ or -1. So the basic algoritm we will analyze is one that follows the usual binary search procedure, but at each step collects m probably correct labels and averages them to decide whether $f(x)$ is positive or negative. The key question is how large does m need to be so that the algorithm terminates with a correct solution with large probability.

Let $\widehat{\mu}_m(x)$ be the average of responses at point x. The obvious test to decide on the correct label is the sign of this average. So we need to determine how large m needs to be so that the sign is correct with probability at least $1 - \delta$. Without loss of generality, assume $f(x) = +1$. Then the true mean $\mu(x) = (1 - \Delta)/2 - (1 - \Delta)/2 = \Delta$. Using the confidence intervals above, we know that with probability at least $1 - \delta$

$$\widehat{\mu}_m(x) + \sqrt{\frac{2\log(2/\delta)}{m}} \geq \mu(x) = \Delta$$

Therefore, the sign of $\widehat{\mu}_m(x)$ is correct with probability at least $1 - \delta$ if

$$\Delta - \sqrt{\frac{2\log(2/\delta)}{m}} \geq 0$$

or equivalently

$$m \geq \frac{2\log(2/\delta)}{\Delta^2}$$

This is the guarantee for the test at just one point. If we test a k points during the binary search process, then the probability that one or more of the tests fails is bounded by $k\delta$. Thus, if we replace δ with δ/k the bound on m above, then the probability that all k tests are correct is at least $1-\delta$. So we will perform k steps of the binary search and collect

$$m \;=\; O\left(\frac{\log(2k/\delta)}{\Delta^2}\right)$$

samples in each step. Taking $k = \log(1/\epsilon)$ yields an ϵ-accurate solution with probability at least $1-\delta$ and the total number of (noisy) function evaluations is

$$O\left(\log(1/\epsilon)\;\underbrace{\frac{\log(2\log(1/\epsilon)/\delta)}{\Delta^2}}_{\text{price paid due to noise}}\right)\;.$$

4 Binary Classification

The binary search problems above assume one can request the label at any chosen point x, but this is not the case in most machine learning (ML) settings. Rather, in ML applications have access to a pool of examples of x distributed according to some distribution and a subset of these examples can be labeled (usually at a cost). The goal is to learn a good classifier using as few labeled examples as possible.

Consider the finite set of hypotheses f_i, $i = 1, \ldots, k$. Each hypothesis has a probability of error

$$R(f_i) \;:=\; \mathbb{P}(f_i(x) \neq y)$$

where the probability is computed over the distribution on x and the binary distribution of y given x. Empirical risk minimization (ERM) is a standard ML approach which proceeds as follows. Select the hypothesis that makes the fewest prediction errors on a set of iid labeled examples $\{(x_i, y_i)\}_{i=1}^m$. The basic ingredient in ERM is an estimate of the error rate of each hypothesis

$$\widehat{R}(f_i) \;=\; \frac{1}{m}\sum_{i=1}^{M} \mathbb{1}(f(x_i) \neq y_i)\;,$$

which is usually referred to as the *empirical risk*. Note that the terms in this average are iid binary random variables, and using the Chernoff bound confidence intervals we have

$$|R(f_i) - \widehat{R}(f_i)| \;\leq\; \sqrt{\frac{\log(2/\delta)}{2m}}$$

with probability at least $1-\delta$. In order to have the confidence intervals hold simultaneously for all hypotheses, we apply the union bound (replace δ with δ/k) to have

$$|R(f_i) - \widehat{R}(f_i)| \;\leq\; \sqrt{\frac{\log(2k/\delta)}{2m}} \text{ for } i = 1, \ldots, k$$

Now suppose we want to select a hypothesis with true error rate within $\gamma > 0$ of $\min_i R(f_i)$. Define true and empirical risk minimizers

$$\begin{aligned} f^\star &= \arg\min_{f\in\{f_i\}} R(f) \\ \widehat{f} &= \arg\min_{f\in\{f_i\}} \widehat{R}(f) \end{aligned}$$

To determine how many samples will be sufficient to guarantee that $R(\widehat{f}) \leq R(f^\star) + \gamma$, consider the confidence intervals for these two hypotheses. The lower confidence bound for $f^\star$ is

$$R(f^\star) \geq \widehat{R}(f^\star) - \sqrt{\frac{\log(2k/\delta)}{2m}}$$

and the upper bound for $\widehat{f}$ is

$$R(\widehat{f}) \leq \widehat{R}(\widehat{f}) + \sqrt{\frac{\log(2k/\delta)}{2m}}$$

Since we want to guarantee that $R(\widehat{f}) - R(f^\star) \leq \gamma$, it suffices to guarantee that the difference between the upper and lower bound is less than or equal to γ. By definition $\widehat{R}(\widehat{f}) \leq \widehat{R}(f^\star)$, so it follows that the upper bound for $\widehat{f}$ is bounded above by the upper confidence bound for $f^\star$. Therefore, a sufficient condition is

$$2\sqrt{\frac{\log(2k/\delta)}{2m}} \leq \gamma \ \text{ or } \ m \geq \frac{2\log(2k/\delta)}{\gamma^2} \tag{1}$$

Note that this argument holds for any data distribution (i.e., no where did we use any special distributional assumptions).

To illustrate ERM, let us consider learning a 1-dimensional linear classifier in the unit interval. Assume the points x distributed uniformly at random on $[0, 1]$, and for any x assume that its label $y = f(x)$ with probability $(1 + \Delta)/2$ and $y = -f(x)$ with probability $(1 - \Delta)/2$, where $f(x)$ is the binary function considered above. Let the finite set of hypotheses f_i, $i = 1, \ldots, k$ be defined as follows:

$$f_i(x) \ = \ \begin{cases} -1 & \text{if } x < i/k \\ \\ +1 & \text{if } x \geq i/k \end{cases}$$

It is easy to bound the error. Let I_i denote subinterval where $f_i \neq f$.

$$\begin{aligned} R(f_i) &= \mathbb{P}(f_i(x) \neq y) \\ &= \mathbb{P}(x \in I_i)\,\mathbb{P}(f_i(x) \neq y | x \in I_i) \ + \ (1 - \mathbb{P}(x \in I_i))\mathbb{P}(f_i(x) \neq y | x \notin I_i) \\ &= |t - i/k|(1 + \Delta)/2 \ + \ (1 - |t - i/k|)(1 - \Delta)/2 \\ &= |t - i/k|\Delta + (1 - \Delta)/2 \end{aligned}$$

Note that $R(f_i) - R(f) = |t - i/k|\Delta$ and $\min_i R(f_i) - R(f) \leq \Delta/k$. So if we wish to achieve this best possible performance, then we should set $\gamma = \Delta/k$ in equation (1) yielding the following bound on the sufficient number of samples

$$m \geq \frac{2k^2 \log(2k/\delta)}{\Delta^2}$$

Recall that in the noisy binary search problem we sought a solution within ϵ of the correct threshold. For this we require $k = 1/\epsilon$ leading to a sufficient condition

$$m = O\left(\frac{1}{\epsilon^2}\frac{\log(2/\epsilon\delta)}{\Delta^2}\right)$$

Observe that this has exponentially worse dependence on ϵ compared to the noisy binary search complexity. We will show that active learning can close the gap.

5 Active Learning of Binary Classifiers

The basic active learning procedure we will pursue is a form of *successive elimination*. The procedure operates in stages, collecting labeled training examples and then removing hypotheses that are inconsistent with the training data (i.e., have large error rates). The key idea is that once some hypotheses are removed, we will only request labels for examples where the remaining hypotheses disagree (labeling examples where they all agree would clearly be unnecessary and wasteful). This sort of scheme has its roots in the seminal papers [2, 1].

Consider a finite set of hypotheses $f_1, \ldots, f_k$ and two stages of this process. In the first stage we will collect a random set of m training data and aim to remove hypotheses with error rates γ greater than the minimum error rate $\min_i R(f_i)$. To do this we will eliminate all hypotheses with lower error bounds greater than or equal to the smallest upper error bound. Specifically, remove all hypotheses for which

$$\widehat{R}(f_j) - \sqrt{\frac{\log(2k/\delta)}{2m}} \geq \widehat{R}(\widehat{f}) + \sqrt{\frac{\log(2k/\delta)}{2m}}$$

since with probability at least $1-\delta$ these do not achieve the minimum error. To determine how large m should be so that no hypotheses with $R(f_j) \geq \min_i R(f_i) + \gamma$ remains, observe that such hypotheses have lower confidence bounds of at least $\min_i R(f_i) + \gamma - 2\sqrt{\frac{\log(2k/\delta)}{2m}}$ and that $\min_i R(f_i) \leq \widehat{R}(\widehat{f}) + \sqrt{\frac{\log(2k/\delta)}{2m}}$, the cutoff used to remove hypotheses. In other words, such an f_j will be removed if $\min R(f_i) \leq \min R(f_i) + \gamma - 2\sqrt{\frac{\log(2k/\delta)}{2m}}$ or equivalently if

$$m \geq \frac{2\log(2k/\delta)}{\gamma^2}$$

In the second stage we will only label examples where the remaining hypotheses disagree in their predictions. This can be easily accomplished by picking unlabeled examples uniformly at random from the pool, checking if they are in the disagreement region, and if so labeling them. This effectively is sampling from the restriction of the distribution to the disagreement region. We can now aim to further remove hypotheses with error rates larger significantly larger than the minimum error rate (on the disagreement region) as done in the first stage. This can significantly reduce the total number of labeled examples needed to learn a good classifier.

5.1 Active Learning of 1-d Classifier

To illustrate the gains of this sort of active learning procedure, consider the problem of learning a 1-dimensional linear classifier in the unit interval. As above, assume the points x distributed uniformly at random on $[0,1]$, and for any x assume that its label $y = f(x)$ with probability $(1+\Delta)/2$ and $y = -f(x)$ with probability $(1-\Delta)/2$, where $f(x)$ is a binary function threshold function. We will consider the same set of candidate classifiers $f_1, \ldots, f_k$, as above. The procedure will operate in $\log_2 k$ stages. The first stage will eliminate classifiers with decision boundaries more than $1/4$ away from the optimal threshold, the second stage will remove all those more than $1/8$ away, and so forth.

Let's look closely at the first stage. We want to remove f_j for which $|t - j/k| \geq 1/4$. Note that for such classifiers

$$R(f_j) - \min R(f_i) \geq \Delta/4$$

So we can set $\gamma = \Delta/4$ and use the results above to see that a sufficient number of samples is

$$m \geq \frac{2\log(4k/\delta)}{(\Delta/4)^2}$$

Note that after this stage, at most $k/2$ hypotheses remain. Iterating this process will reduce the size of the set remaining after each stage by a factor of 2. We require confidence intervals only for the set of hypotheses remaining after each stage: k in stage 1, $k/2$ in stage 2, and so on for a total of $k + k/2 + \cdots \leq 2k$. Note the extra factor of 2 in the log that accounts for this.

In the second stage we will repeat this process, but restricted to the interval containing the remaining hypotheses. Note that this interval is at most $1/2$ in length. Because of this, stage 2 is essentially a carbon copy of the first stage. We want to remove f_j for which $|t - j/k| \geq 1/8$. Note that for such classifers we again have

$$R(f_j) - \min R(f_i) \geq \Delta/4$$

Note that the lower bound is again $\Delta/4$ not $\Delta/8$ as it would have been in first stage. This is due to the fact that our sampling is now restricted to a half interval, so there is twice as great a probability that x will belong to the subinterval where f_j and $f^\star$ disagree. Thus, again a sufficient number of samples is

$$m \geq \frac{2\log(4k/\delta)}{(\Delta/4)^2}$$

and at most $k/4$ hypotheses will remain.

A similar situation is encountered in each subsequent stage and after $\log_2 k$ stages we will correctly identify $f^\star$ with probability at least $1-\delta$ and the total number of labeled examples is

$$m \;=\; O\left(\frac{\log k \log(4k/\delta)}{\Delta^2}\right)$$

If we wish to have a classifer with a decision boundary within ϵ of the correct threshold, then as before we take $k = 1/\epsilon$ leading to a sample bound of

$$m \;=\; O\left(\frac{\log(\frac{1}{\epsilon})\log(\frac{4}{\epsilon\delta})}{\Delta^2}\right)$$

Observe that this has exponentially better dependence on ϵ compared to the ERM and essentially the same dependence we observed for the noisy binary search problem.

5.2 Dealing with Unknown Noise Level

The active learning result above based the sample size at each stage based on the noise level Δ. Of course, this may not be known in practice. There is a simple remedy for this known as *doubling*. Recall, the goal is to remove $1/2$ of the hypotheses in a given stage. We will start with a small sample size and check if we can remove at least half of the hypotheses based on the confidence interval criterion. If not, then collect a new sample of double the size and check again. The doubling continues until we are able to eliminate at least $1/2$ of the hypotheses.

Consider the first stage of the active learning algorithm considered above. We will start with $m = 1$ and double this until we can remove at least $k/2$ hypotheses. We need to ensure that the confidence intervals are correct (with probability at least $1 - \delta$) at every doubling step, so for doubling step $\ell = 1, 2, \dots$ we collect $m_\ell = 2^\ell$ samples and set $\delta_\ell = \delta/2^\ell$. Note that no matter how many steps are needed $\sum_{\ell\geq 1}\delta_\ell \leq \delta$. After collecting training data in step ℓ the confidence intervals have width

$$2\sqrt{2\log(2^{\ell+1}k/\delta)/2^\ell}$$

These are essentially the same as the intervals derived above, but we have an extra factor of 2^ℓ in the logarithm due to the union bound over doubling steps. This is easy to see if we substitute the number of samples $m_\ell = 2^\ell$ into the expression.

$$2\sqrt{2\log(2m_\ell k/\delta)/m_\ell}$$

In short, the intervals have width $O(\sqrt{\log m_\ell/m_\ell})$ instead of $O(\sqrt{1/m_\ell})$.

Following the reasoning above, we will remove $1/2$ of the hypotheses if the interval width is less than $\Delta/4$, which translates into the requirement:

$$2\sqrt{\log(2m_\ell k/\delta)/m_\ell} \leq \Delta/4$$

which is met with probability at least $1 - \delta$ as soon the total number of samples collected

$$m \;=\; O\left(\frac{\log(2k/\delta)}{\Delta^2}\log(4\log(2k/\delta)/(\Delta/4)^2)\right)$$

Note the resulting extra logarithmic factor incurred by the doubling steps.

So, up to logarithmic factors, the sample complexity of the noise-adaptive algorithm is of the same order as that of the previous algorithm.

References

[1] Maria-Florina Balcan, Alina Beygelzimer, and John Langford. Agnostic active learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 65–72. ACM, 2006.

[2] David Cohn, Les Atlas, and Richard Ladner. Improving generalization with active learning. *Machine learning*, 15(2):201–221, 1994.