COMS 6998-4 Fall 2017 Presenter: Ji Xu
October 2, 2017 Scribe: Jiefu Ying

Active Learning Analysis in Realizable Case

1 Introduction

We have discussed several algorithms when training data and their labels are given at the same time. While chances are that the cost of obtaining unlabeled data and corresponding labels differs greatly. Say that we want to do spam email classification, it is easy to collect a lot of unlabeled emails, but asking people to label them is costly. As a result, it is important to think of certain methods which can avoid conducting too many label queries.

If all the labels are given in advance, we call it passive learning, and all previously introduced method is applicable. Instead, now we are concerning the case that the learning algorithm actively queries for labels when it is not sure about its own judgment. So a trivial upper bound of label complexity, standing for the number of queries, is the size of training dataset n. Ignore the trivial case that every potential hypothesis performs the same on training data set, suppose there are countable but enormous potential hypothesis, the best-case scenario is when we can apply a halving algorithm according to training examples, and the corresponding label complexity of figuring out hypotheses fit all examples can be reduced to $O(\log_2(n))$ instead of $O(n)$.

Similar to other learning algorithms, there are realizable and agnostic cases according to whether we have any specific assumption on the error rate of optimal hypothesis in initial hypothesis set. In this scribe, we focus on the realizable case.

2 Basic settings

Let $\mathcal{D}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$ where $\mathcal{X}$ is the input space and $\mathcal{Y} = \{\pm 1\}$ are the possible labels. $(X, Y) \in \mathcal{X} \times \mathcal{Y}$ is a pair of random variable with joint distribution $\mathcal{D}$. $\mathcal{H}$ represents a set of hypothesis mapping from $\mathcal{X}$ *to* $\mathcal{Y}$. The error of a hypothesis h marks as:

$$\text{err}(h) := \Pr(h(X) \neq Y)$$

Let $h^* = \text{argmin}\{\text{err}(h) : h \in \mathcal{H}\}$ be a hypothesis of minimum error in $\mathcal{H}$. Our goal here remains the same with passive learning, which is that with probability at least $1 - \delta$ over the choice of the random example, give a algorithm produces a hypothesis $h \in \mathcal{H}$ with error rate

$$\text{err}(h) \leq \text{err}(h^*) + \epsilon$$

Since we are focusing on the realizable case, the optimal hypothesis $h^* \in \mathcal{H}$ makes no error, so here we have:

$$\text{err}(h) \leq \text{err}(h^*) + \epsilon = \epsilon$$

.

3 PAC learning algorithm

The algorithm is first introduced by Cohn, Atlas, and Ladner [1], noted as CAL below. An intuitive understanding of the algorithm is that since h^* is in the initial hypothesis set, the learning algorithm only needs to eliminate hypothesis which makes mistakes during iterations, and after at most size of $\mathcal{H}$ rounds, the loop terminates. There are two questions to answer before we write down a PAC algorithm:

1. How to pick the next training data point to query to label?

2. How to maintain the hypothesis set according to the feedback from the oracle?

It's clear that we intend to use the feedback to eliminate a certain amount of hypothesis, so the training data (X_t, Y_t) provides no new information if $h_i(X_t)$ are the same for all $h_i \in \mathcal{H}_t$ at t^{th} round iteration. Instead, the algorithm should query points in regions where disagreement between hypothesis exists, in other words, with uncertainty, and use the feedback to reduce current potential hypothesis set.

We introduce formal notations to make this process well-defined. For a set of labeled examples $Z \subset \mathcal{X} \times \mathcal{Y}$, the version space $\mathcal{V}(Z)$ with respect to a hypothesis class $\mathcal{H}$ is

$$\mathcal{V}(Z) := \{h \in \mathcal{H} \ : \ h(x) = y, \ \forall (x, y) \in Z\}.$$

The subsets of hypothesis in $\mathcal{H}$ are consistent with examples in Z. Actually, the algorithm should choose to query an example if and only if it locates in a disagreement region $\mathcal{R}(\mathcal{V})$. For a set of hypothesis $\mathcal{V}$, the region of disagreement $\mathcal{R}(\mathcal{V})$ is

$$\mathcal{R}(\mathcal{V}) := \{x \in \mathcal{X} : \ \exists h, \ h^{'} \in \mathcal{V} \text{ such that } h(x) \neq h(x^{'})\}$$

With all the definitions above, CAL algorithm is represented below:

Algorithm 1 CAL active learning algorithm

Initialize $Z_0 := \emptyset,\ \mathcal{V}_0 := \mathcal{H}$.

1: **for** t = 1, 2, ..., n: **do**
2: Obtain unlabeled data point X_t
3: **if** $X_t \in \mathcal{R}(\mathcal{V}_{t-1})$ **then**
4: Query Y_t, and set $Z_t := Z_{t-1} \cup \{(X_t, Y_t)\}$
5: **else**
6: Set $\bar{Y}_t := h(X_t)$ for any $h \in \mathcal{V}_{t-1}$, and set $Z_t := Z_{t-1} \cup \{(X_t, \bar{Y}_t)\}$
7: **end if**
8: Set $\mathcal{V}_t := \{h \in \mathcal{H} : h(X_i) = Y_i,\ \forall (X_i, Y_i) \in Z_t\}$
9: **end for**
10: **return** $\forall h \in \mathcal{V}_n$.

Note that Line 6 is just equivalent to $Z_t := Z_{t-1}$. This is because when Line 6 is reached in round t, then every $h \in \mathcal{V}_{n-1}$ has $h(X_i) = Y_i$, in this case,

$$\mathcal{V}_t = \mathcal{V}_{t-1} \cap \{h \in \mathcal{V}_{t-1} : h(X_t) = \bar{Y}_t\} = \mathcal{V}_{t-1}.$$

The whole algorithm is quite straightforward except the condition of the if-statement. The mathematical representation of $\mathcal{R}(\mathcal{V}_{t-1})$ is not well defined, so how to judge the belonging relationship is ambiguous. Here is a Reduction-based representation of CAL algorithm shown below. These two algorithms are equivalent in fact, while the Reduction-based algorithm gives a particular understanding of the condition $X_t \in \mathcal{R}(\mathcal{V}_{t-1})$, and it is pretty practical if the hypothesis set is finite. If not, probably it's better to go back to original CAL and find another way to understand $\mathcal{R}(\mathcal{V}_{t-1})$.

Algorithm 2 CAL active learning algorithm (Reduction-based)

Initialize $Z_0 := \emptyset,\ \mathcal{V}_0 := \mathcal{H}$.

1: **for** t = 1, 2, ..., n: **do**
2: Obtain unlabeled data point X_t
3: **if** there exists both:
$h^+ \in \mathcal{H}$ consistent with $Z_{t-1} \cup \{(X_t, +1)\}$
$h^- \in \mathcal{H}$ consistent with $Z_{t-1} \cup \{(X_t, -1)\}$
then
4: Query Y_t, and set $Z_t := Z_{t-1} \cup \{(X_t, Y_t)\}$
5: **else**
6: Only h^y exists for some $y \in \{\pm 1\}$: Set $\bar{Y}_t := h(X_t)$ for any $h \in \mathcal{V}_{t-1}$, and set $Z_t := Z_{t-1} \cup \{(X_t, \bar{Y}_t)\}$
7: **end if**
8: Set $\mathcal{V}_t := \{h \in \mathcal{H} : h(X_i) = Y_i,\ \forall (X_i, Y_i) \in Z_t\}$
9: **end for**
10: **return** $\forall h \in \mathcal{V}_n$.

CAL algorithm follows the idea that deducing labels assigned by h^* whenever it does not query true label Y_t. The correctness is quite straightforward and can be formalized by induction on iteration times.

4 Label complexity analysis

The goal of CAL algorithm is to find a set $\mathcal{H}^{'}$ of good hypothesis h which is consistent with h^* on n training examples. Intuitively, cases below will increase the cost to find $\mathcal{H}^{'}$:

1. Many hypotheses have high error rates and need to be eliminated. In other words, $\sup_{h\in\mathcal{H}'} \mathrm{err}(h)$ is too large.

2. Hard to get an example locates in uncertainty region so that many rounds are needed to update potential hypothesis set once. In other words, $\Pr(\mathcal{R}(\mathcal{V}))$ is too small.

It looks like label complexity can be measured by the ratio between $\sup_{h\in\mathcal{H}'} \mathrm{err}(h)$ and $\Pr(\mathcal{R}(\mathcal{V}))$. Before further exploration, Let's formalize a couple of basic concepts and measurements.
For a random variable $X \in \mathcal{X}$, the disagreement (pseudo) metric ρ on $\mathcal{H}$ is defined by

$$\rho(h, h^{'}) := \Pr(h(X) \neq h^{'}(X)).$$

It is called pseudo metric because $\rho(h, h^{'}) = 0$ does not necessarily mean $h = h^{'}$. While attributes like triangle inequality still hold. Let $B(h, r) := \{h^{'} \in \mathcal{H} : \rho(h, h^{'}) \leq r\}$ denote the ball centered at $h \in \mathcal{H}$ of radius $r \geq 0$. And then we can define an important concept, disagreement coefficient $\theta(\mathcal{H}, \mathcal{D})$, which is:

$$\theta(\mathcal{H}, \mathcal{D}) := \sup\Big\{ \frac{\Pr(X \in \mathcal{R}(B(h^*, r)))}{r} \ : \ r > 0 \Big\},$$

where h^* is a particular hypothesis of minimum error under $\mathcal{D}$.

Let's see some intuitions behind the definition. Since $\rho(h, h^{'}) \leq r$, r is a certain description of uncertainty region. As the denominator, r represents the $\Pr(\mathcal{R}(\mathcal{V}))$ in fact. At the same time, being aware of $\mathcal{R}(B(h^*, r))$ is the only region disagreements take place, the numerator means a bound of error rate. So the disagreement coefficient exactly describes the intuitive idea mentioned at the beginning of this paragraph.

Disagreement coefficient tells information about the joint distribution of $\mathcal{D}$ and hypothesis set $\mathcal{H}$. Let's see a simple example of how to calculate this value. Define single-variable threshold function f_θ as:

$$f_\theta := \begin{cases} +1 & \text{if } x \geq \theta \\ -1 & \text{if } x < \theta \end{cases}$$

Assume that $\mathcal{H}$ is a set of single-variable threshold functions, and argument X has a uniform distribution on [0, 1]. For all $r > 0$, any $h_\theta \in B(h_{\theta^*}, r)$ has $\theta \in [\theta^* - r, \theta^* + r]$, with the possibility $2r$. According to the definition, $\theta(\mathcal{H}, \mathcal{D}) = 2r/r = 2$.
Besides knowledge of $\mathcal{D}$ and $\mathcal{H}$, ϵ and δ, error rate upper bound and confidence level also influence label complexity. It's reasonable to infer that the upper bound of label complexity can be represented in terms of $\theta(\mathcal{H}, \mathcal{D})$, ϵ and δ.

Theorem 1. *The expected number of labels queried by Reduction-based CAL after n iterations is at most*

$$O(\theta(h^*, \mathcal{H}, \mathcal{D}) d (\log_2 n)^2),$$

where d is the VC-dimension of class $\mathcal{H}$. For any $\epsilon > 0$ and $\delta > 0$, if we have

$$n = O\left(\frac{d \cdot \log \frac{1}{\epsilon} + \log \frac{1}{\delta}}{\epsilon}\right),$$

then with probability $1 - \delta$, the return of Reduction-based CAL $\hat{h}$ satisfies that $\mathrm{err}(\hat{h}) \leq \epsilon$.

Before proof starts, let's recall a theorem learned in the previous lecture on statistical learning.

Theorem 2. *Assume that $\mathcal{F}$ is the loss class of $\mathcal{H}$,*

$$\mathcal{F} = \{f : (x, y) \mapsto \mathbb{1}_{h(x) \neq y} : h \in \mathcal{H}\}.$$

$\mathcal{F}$ is a class of $\{0, 1\}$-valued function with VC-dimension equals d. For any $\delta \in (0, 1)$, with probability at least $1 - \delta$:

$$Pf - P_n f \leq 2\sqrt{Pf \ \frac{d \cdot \log(2n + 1) + \log \frac{4}{\delta}}{n}}, \ \forall f \in \mathcal{F}$$

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

$$P_n f - Pf \leq 2\sqrt{P_n f \ \frac{d \cdot \log(2n + 1) + \log \frac{4}{\delta}}{n}}, \ \forall f \in \mathcal{F}$$

Corollary: If $P_n f_n = O\left(\frac{d \cdot \log(n) + \log \frac{1}{\delta}}{n}\right)$,

$$Pf_n = O\left(\frac{d \cdot \log(n) + \log \frac{1}{\delta}}{n}\right).$$

We will prove Theorem 1 using Theorem 2.

Proof. According to Theorem 2, with probability $1-\delta_t$, $\forall h \in H$ consistent with Z_t has error $\text{err}(h)$ at most

$$O\left(\frac{d \cdot \log(n) + \log\frac{1}{\delta}}{n}\right).$$

Define the value as r_t. CAL algorithm focuses on realizable case, so when $P_n f_n = 0, Pf = 0$. In addition, we know the $\text{err}(h)$ is at most ϵ after n rounds ($\text{err}(h^*) = 0$), as a result,

$$n = O\left(\frac{d \cdot \log\frac{1}{\epsilon} + \log\frac{1}{\delta}}{\epsilon}\right).$$

Call G_t the event that describes whether the above happens, condition on G_t, and we will have:

$$\{h \in \mathcal{H} : h \text{ is consistent with } Z_t\} \subset B(h^*, r_t).$$

because anything outside of $B(h^*, r_t)$ cannot distinguish members of set $\mathcal{H}$.

We won't query Y_{t+1} unless there exists a h disagree with h^* on X_{t+1}. Write it in a formal representation, Y_{t+1} is queried if and only if:

$$\exists h \in \mathcal{H} \text{ consistent with } Z_t \cup \{(X_{t+1}, -h^*(X_{t+1}))\}.$$

So, condition on G_t, if we query Y_{t+1}, then obviously, $X_{t+1} \in \mathcal{R}(h^*, r_t)$. Therefore,

$$\Pr(Y_{t+1} \text{ is queried} | G_t) \leq \Pr(X_{t+1} \in \mathcal{R}(h^*, r_t) | G_t).$$

Define indicator $Q_t = \mathbb{1}_{Y_t \text{ is queried}}$. The expected total number of queries is

$$\begin{aligned}
\mathbb{E}\left(\sum_{t=1}^{n} Q_t\right) &\leq 1 + \sum_{t=1}^{n-1} \Pr(Q_{t+1} = 1) \\
&= 1 + \sum_{t=1}^{n-1} \Pr(Q_{t+1} = 1 | G_t) \Pr(G_t) + \sum_{t=1}^{n-1} \Pr(Q_{t+1} = 1 | \text{not } G_t) \Pr(\text{not } G_t) \\
&\leq 1 + \sum_{t=1}^{n-1} \Pr(Q_{t+1} = 1 | G_t) \Pr(G_t) + \delta_t \\
&\leq 1 + \sum_{t=1}^{n-1} \Pr(X_{t+1} \in \mathcal{R}(h^*, r_t) | G_t) \Pr(G_t) + \delta_t
\end{aligned}$$

Obviously,

$$\Pr(X_{t+1} \in \mathcal{R}(h^*, r_t) | G_t) \Pr(G_t) \leq \Pr(X_{t+1} \in \mathcal{R}(h^*, r_t)),$$

and according to the definition of disagreement,

$$\Pr(X_{t+1} \in \mathcal{R}(h^*, r_t)) = r_t \theta(h^*, \mathcal{H}, \mathcal{D}).$$

Hence, we have

$$\begin{aligned}\mathbb{E}\left(\sum_{t=1}^{n} Q_t\right) &\leq 1 + \sum_{t=1}^{n-1}(r_t\theta(h^*, \mathcal{H}, \mathcal{D}) + \delta_t) \\ &= \sum_{t=1}^{n-1} O\left(\frac{\theta(h^*, \mathcal{H}, \mathcal{D})}{t}(d \cdot \log(t) + \log\frac{1}{\delta_t}) + \delta_t\right)\end{aligned}$$

The maximum of right hand side is reached if let $\delta_t = \frac{1}{t}$, at that time:

$$\mathbb{E}\left(\sum_{t=1}^{n} Q_t\right) = O\left(\theta(h^*, \mathcal{H}, \mathcal{D})d(\log_2 n)^2\right).$$

□

Bibliographic notes

CAL algorithm is due to Cohn, Atlas, and Ladner [1], and the detailed label complexity analysis is introduced by Hsu [2]. Theorem 2 comes from statistic learning theory material Bousquet, Boucheron, and Lugosi [3].

References

[1] Cohn, Atlas, and Ladner. Improving generalization with active learning. *Machine Learning*, 15: 201, 1994.

[2] Daniel Joseph Hsu. Algorithms for Active Learning. *UC San Diego:Computer Science*, b6846726, 2010.

[3] Bousquet, Boucheron, and Lugosi. Introduction to Statistical Learning Theory. *Springer Berlin Heidelberg*, 169-207, 2004.