PAC Intro from Understanding ML

Cody Melcher

February 27, 2023
Quick conceptual ML

1. Necessary for complex tasks, useful for inherent flexibility.
Overview

▶ Quick conceptual ML
  1. Necessary for complex tasks, useful for inherent flexibility.

▶ Learning Environment
  1. Framework
  2. ERM (without and with Inductive Bias)
Overview

- Quick conceptual ML
  1. Necessary for complex tasks, useful for inherent flexibility.

- Learning Environment
  1. Framework
  2. ERM (without and with Inductive Bias)

- Formal Definition of the PAC learning model
  1. General bounds
ML Conceptually

- ML: Automating converting "experience" or data into knowledge.
  1. Necessary for complex tasks, useful for inherent flexibility.
Statistical Learning Framework

Basic setting:

1. **Domain space** $\mathcal{X}$ ie covariate space.
2. **Label set**: space of possible responses/explanatory variables. Focus here on $\mathcal{Y} = \{1, 2\}$
3. **Training data** as finite **sequence** of ordered pairs in $\mathcal{X} \times \mathcal{Y}$
4. **Prediction rule** $h : \mathcal{X} \rightarrow \mathcal{Y}$; also called hypothesis or classifier.
Notes on basic framework

2. Assume perfect/correct but unknown classifier $f_i$ exists and $y_i$ are mapped to by $f_i$ from $\mathcal{X}$.
3. **Error of** $h$ is
   \[ L_D(h) = P_{x \sim D}(h(x) \neq f(x)) = D(\{x : h(x) \neq f(x)\}). \]
   Minimize!
4. Note $D : A \rightarrow [0, 1]$ where $A \subset \mathcal{X}$
5. $L$ called risk or generalization error
Empirical Risk Minimization

1. Don’t know $\mathcal{D}$ or $f$. Minimize training error (empirical risk/error) instead.

2. $L_S(h) = \frac{|\{i \in [m]: h(x_i) \neq y_i\}|}{m} \in [0, 1]

3. Problem: can easily lead to overfitting ie define perfect classifier for training data that is terrible classifier on other data.

4. Solution: restriction on set of possible classifiers $\mathcal{H}$. Big topic: conditions on $\mathcal{H}$ to guarantee no overfitting.

5. First restriction: $|\mathcal{H}| < \infty$ which leads to...
ERM

- Realizability Assumption (2.1) etc.

1. There exists $h^* \in \mathcal{H}$ s.t. $L_{D,f}(h^*) = 0$ i.e. there exists for every empirical risk minimization problem there exists a perfect classifier.

2. Possible issue: data drawn from $\mathcal{D}$ could be bad representation of $\mathcal{D}$. Need to think of error as random.

3. Let $\delta$ be probability sample from $\mathcal{D}$ is non-representative sample so $(1 - \delta)$ is our confidence parameter (our sample is representative).

4. Let $\epsilon$ be the accuracy parameter

5. Want to use these to put upper bound on what samples can be realized that lead to classification failure.
ERM cont.

- Want to figure out bounding $D^m(S_x : L_{(D,f)}(h_S) > \epsilon)$ ie bound the realized samples of size m mapped by the classifier incorrectly classifies by $\epsilon$.

- Define the set of bad classifiers as
  $$\mathcal{H}_B = \{ h \in \mathcal{H} : L_{D,f}(h) > \epsilon \}$$
  and let
  $$M = \{ S_x : \exists h \in \mathcal{H}_B, L_S(h) = 0 \}$$
  be the set of misleading samples.

- Follows that $\{ S_x : L_{D,f}(h_S) > \epsilon \} \subset M$ ie set of samples that give a bad classifier are a subset of the misleading samples set.
ERM cont.

- Can relatedly think of all the possible bad classifiers that manage to give perfect prediction due to a misleading sample; this is equivalent to $M$.

- It follows that

$$D^m(\{S|_x : L(D,f)(h_s) > \epsilon\}) \leq D^m(\cup_{h \in H_b}\{S|_x : L_S(h) = 0\})$$

- By $D(A \cup B) \leq D(A) + D(B)$, it follows that RHS

$$\leq \sum_{h \in H_b} D^m(\{S|_x : L_S(h) = 0\})$$

- Next due to iid assumption and how we defined $\epsilon$, it follows that $D^m(\{x_i : h(x_i) = y_i\}) = (1 - L_{D,f})^m \leq (1 - \epsilon)^m \leq e^{-\epsilon m}$
Combining the previous inequality with the inequality from two slides back we get, $D^m(\{S_x : L_{(D,f)}(h_s) > \epsilon\}) \leq |\mathcal{H}| e^{-\epsilon m}$ which provides the upper bound on the realized samples of size $m$ mapped by the classifier incorrectly, which we can summarize as,

**Corollary 2.3**: If $\mathcal{H}$ is finite, $\delta \in (0, 1)$ and $\epsilon > 0$ and let $m$ be any integer st. $m \geq \log(\frac{|\mathcal{H}| \delta}{\epsilon})^{\frac{1}{\epsilon}}$. Then for any true classifier $f$ and distribution $D$ assuming realizability wit probability at least $1 - \delta$ over iid sample of size $m$, we have that for every ERM classifier $h_s$ it holds that $L_{D,f}(h_s) \leq \epsilon$. 

ERM cont.
ERM cont.

- i.e. for sufficiently large $m$, classifiers from finite classifier class will be probably (with confidence $1 - \delta$) approximately (up to $\epsilon$ wrong) correct.

- Hence why, PAC learning environment $= \text{probably approximately correct}$. 
PAC Learnability Definition

(3.1) PAC Learnability: \( \mathcal{H} \) is **PAC learnable** if there exists a function \( m_\mathcal{H} : (0, 1)^2 \rightarrow \mathbb{N} \) and a learning algorithm with the following property: \( \forall \epsilon, \delta \in (0, 1), \forall \mathcal{D} \) over \( \mathcal{X} \) and for every true classifier \( f : \mathcal{X} \rightarrow \{0, 1\} \), and if the realizable assumptions holds, then when running the learning algorithm on \( m \geq m_\mathcal{H}(\epsilon, \delta) \) iid samples generated by \( \mathcal{D} \) and classified by \( f \), the algorithm returns a classifier \( h \) st with probability at least \( 1 - \delta \) that \( L_{\mathcal{D}, f}(h) \leq \epsilon \)
PAC Learnability notes

- $\epsilon$ = how far the classifier can be from the optimal classifier $f$
- $\delta$ = how likely the classifier is to be $\epsilon$ close to $f$
- notice that $m_h$ determines how many samples needed to be probably accurate; we will focus on finding the minimal $m_h$ ie minimal integer that guarantees probable accuracy.

(3.2) $\forall |\mathcal{H}| < \infty$, $m_h(\epsilon, \delta) \leq \frac{\log(|\mathcal{H}|)}{\epsilon}$
Agnostic PAC

- Realizability seems unrealistic....solution: agnostic PAC!

- (3.3) A hypothesis class $\mathcal{H}$ is **agnostic PAC learning** if 
  \[ \exists m_h : (0, 1)^2 \to \mathcal{N} \text{ and a learning algorithm with the following property: } \forall \epsilon, \delta \in (0, 1) \text{ and } \forall D \text{ over } \mathcal{X} \times \mathcal{Y} \text{ when running the learning algorithm on } m \geq m_h(\epsilon, \delta) \text{ iid examples generated by } D, \text{ a hypothesis } h \text{ is returned st with probability at least } 1 - \delta, \]
  \[ L_D(h) \leq \min_{h' \in H} (L_D(h') + \epsilon) \]

- I.e we can guarantee we are only $\epsilon$ worse than the best predictor $h'$ from a class of classifiers $\mathcal{H}$. 
Extensions to Agnostic PAC

- Generally we refer to the agnostic PAC learning environment as PAC learning.
- Can expand $\mathcal{Y}$ to be larger than $\{0, 1\}$, but still assuming it is finite.
- Can change risk to be expected squared difference (or something else) to deal with regression problems.
- Both these extensions change $L_D(h)$, and thus we need to rethink our above definition....
Agnostic PAC with General Loss

(3.4) A hypothesis class $\mathcal{H}$ is **agnostic PAC learning** with respect to set $Z$ and loss function $l : H \times Z \to \mathbb{R}_+$ if

$\exists m_h : (0, 1)^2 \to \mathcal{N}$ and a learning algorithm with the following property: $\forall \epsilon, \delta \in (0, 1)$ and $\forall D$ over $Z$ when running the learning algorithm on $m \geq m_h(\epsilon, \delta)$ iid examples generated by $D$, a hypothesis $h$ is returned st with probability at least $1 - \delta$,

$L_D(h) \leq \min_{h' \in H}(L_D(h') + \epsilon)$

where $L_D(h) = E_{(z \sim D)}(l(h, z)$ and $Z = \mathcal{X} \times \mathcal{Y}$ (for our problems though this can be generalized)
TLDR;

- PAC learning feels natural to stats/math world, learning from "experience" ie data ie we are still trying to approximate a function.

- New ish to us maybe because PAC learning is about thinking about bounds on what is possible and sorta avoiding asymptotics.

- Still very general, but allows us to move to VC dimension (what characteristics of $\mathcal{H}$ allow us to deduce it is PAC learnable?), be more rigorous with a lot of the techniques we have learned, etc.