Learning Theory

Aarti Singh

Machine Learning 10-701
Mar 29, 2023

Slides courtesy: Carlos Guestrin

Learning Theory

- We have explored **many** ways of learning from data
- But…
  - Can we certify how good is our classifier, really?
  - How much data do I need to make it "good enough"?

PAC Learnability

Probably Approximately Correct (ε, δ)

- True function space, F
- Model space, H

F is **PAC Learnable** by a learner using H if

there exists a learning algorithm s.t. for all functions in F, for all distributions over inputs, for all $0 < \varepsilon, \delta < 1$, with probability $> 1-\delta$, the algorithm outputs a model $h \in H$ s.t. $\text{error}_{\text{true}}(h) \leq \varepsilon$ in time and samples that are polynomial in $1/\varepsilon$, $1/\delta$.

probably

approximately correct

\# data m

$m \sim \frac{1}{\varepsilon}$ $\quad m \sim \frac{1}{\varepsilon^2}$ $\quad \sim \frac{1}{\sqrt{\varepsilon}}$ $\quad \cancel{m \sim \frac{1}{2^{\varepsilon}}}$

A simple setting

- Classification
  - m i.i.d. data points
  - **Finite** number of possible classifiers in model class (e.g., dec. trees of depth d)
- Lets consider that a learner finds a classifier *h* that gets zero error in training
  - $\text{error}_{\text{train}}(h) = 0$
- What is the probability that *h* has more than ε true (= test) error?
  - $\text{error}_{\text{true}}(h) \geq \varepsilon$

Even if *h* makes zero errors in training data, may make errors in test

How likely is a bad classifier to get m data points right?

$P(h(X) \neq Y)$
$\shortparallel$

- Consider a bad classifier *h* i.e. $\underline{\underline{\text{error}_{\text{true}}(h) \geq \varepsilon}}$

- Probability that *h* gets one data point right

$$\leq 1 - \varepsilon$$

- Probability that *h* gets *m* data points right

$$\leq \underline{\underline{(1 - \varepsilon)^m}}$$

How likely is a learner to pick a bad classifier?

- Usually there are many (say k) bad classifiers in model class

  $h_1, h_2, \ldots, h_k$ s.t. $\text{error}_{\text{true}}(h_i) \geq \varepsilon \quad i = 1, \ldots, k$

- Probability that learner picks a bad classifier = Probability that some bad classifier gets 0 training error

  Prob(h_1 gets 0 training error OR

  h_2 gets 0 training error OR … OR

  h_k gets 0 training error)

  $P(A \cup B) \leq P(A) + P(B)$

  $\leq$ Prob(h_1 gets 0 training error) +

  Prob(h_2 gets 0 training error) + … +

  Prob(h_k gets 0 training error)

  $\leq k(1-\varepsilon)^m$

Union bound

Loose but works

How likely is a learner to pick a bad classifier?

- Usually there are many many (say k) bad classifiers in the class

$$h_1, h_2, \ldots, h_k \qquad \text{s.t. } \mathrm{error}_{true}(h_i) \geq \varepsilon \quad i = 1, \ldots, k$$

- Probability that learner picks a bad classifier

$$\leq \; k\,(1-\varepsilon)^m \;\; \leq \;\; |H|\,(1-\varepsilon)^m \leq \;\; |H|\, e^{-\varepsilon m}$$

Size of model class

PAC (Probably Approximately Correct) bound

- ***Theorem [Haussler'88]***: Model class *H* finite, dataset *D* with *m* i.i.d. samples, $0 < \varepsilon < 1$: for any learned classifier *h* that gets 0 training error:

$$P(\text{error}_{true}(h) \geq \epsilon) \leq |H|e^{-m\epsilon} \leq \delta$$

- Equivalently, with probability $\geq 1 - \delta$

$$\text{error}_{true}(h) \leq \epsilon$$

Important: PAC bound holds for all *h* with 0 training error, but doesn't guarantee that algorithm finds best *h*!!!

Using a PAC bound

80% accuracy

eg. 20% error

eg. prob ≥ 0.9

$\delta = 0.1$

$$|H|e^{-m\epsilon} \leq \delta$$

- Given ε and δ, yields **sample complexity**

#training data, $$m \geq \frac{\ln|H| + \ln\frac{1}{\delta}}{\epsilon}$$

- Given m and δ, yields error bound

error, $$\epsilon \geq \frac{\ln|H| + \ln\frac{1}{\delta}}{m}$$

Poll

$$m \geq \frac{\ln|H| + \ln 1/\delta}{\varepsilon}$$

Assume m is the minimum number of training examples sufficient to guarantee that with probability 1 − δ a consistent learner using model class H will output a classifier with true error at worst ε.

Then a second learner that uses model space H′ will require 2m training examples (to make the same guarantee) if |H′ | = 2|H|.

A. True B. False

If we double the number of training examples to 2m, the error bound ε will be halved.

C. True D. False

Limitations of Haussler's bound

- Only consider classifiers with 0 training error

  h such that zero error in training, $\text{error}_{\text{train}}(h) = 0$

- Dependence on size of model class |H|

$$m \geq \frac{\ln |H| + \ln \frac{1}{\delta}}{\epsilon}$$

  what if |H| too big or H is continuous (e.g. linear classifiers)?

What if our classifier does not have zero error on the training data?

- A learner with zero training errors may make mistakes in test set
- What about a learner with $error_{train}(h) \neq 0$ in training set?
- The error of a classifier is like estimating the parameter of a coin!

$$error_{true}(h) := P(h(X) \neq Y) \qquad \equiv \qquad P(H=1) =: \theta$$

$$error_{train}(h) := \frac{1}{m} \sum_i \mathbf{1}_{h(X_i) \neq Y_i} \qquad \equiv \qquad \frac{1}{m} \sum_i Z_i =: \widehat{\theta} \qquad E[\widehat{\theta}] = \theta$$

Hoeffding's bound for a single classifier

- Consider *m* i.i.d. flips $x_1,\ldots,x_m$, where $x_i \in \{0,1\}$ of a coin with parameter θ. For $0<\varepsilon<1$:

Haussler

$e^{-m\varepsilon}$

$$P\left(\left|\theta - \frac{1}{m}\sum_i x_i\right| \geq \epsilon\right) \leq 2e^{-2m\epsilon^2}$$

- Central limit theorem: $z_1 \ldots z_m$ iid $E[z_i]=\mu$, $\mathrm{var}(z_i)=\sigma^2$

$\sqrt{m}\left(\frac{1}{m}\sum_i z_i - \mu\right) \to N(0,\sigma^2)$

$\mu = \theta$

$\sigma^2 = \theta(1-\theta) \leq \frac{1}{4}$

$e^{-\frac{\varepsilon^2}{2\sigma^2}}$

$\frac{1}{m}\sum_i x_i - \theta \to N\left(0,\frac{\sigma^2}{m}\right)$

$\to N\left(0, \leq \frac{1}{4m}\right)$

$\sim e^{-2m\varepsilon^2}$

Hoeffding's bound for a single classifier

- Consider m i.i.d. flips $x_1,\ldots,x_m$, where $x_i \in \{0,1\}$ of a coin with parameter θ. For $0<\varepsilon<1$:

$$P\left(\left|\theta - \frac{1}{m}\sum_i x_i\right| \geq \epsilon\right) \leq 2e^{-2m\epsilon^2}$$

- For a single classifier h

$$P\left(|\text{error}_{true}(h) - \text{error}_{train}(h)| \geq \epsilon\right) \leq 2e^{-2m\epsilon^2}$$

Hoeffding's bound for |H| classifiers

- For each classifier h_i:

$$P\left(|\text{error}_{true}(h_i) - \text{error}_{train}(h_i)| \geq \epsilon\right) \leq 2e^{-2m\epsilon^2}$$

- What if we are comparing |H| classifiers?

  Union bound

- ***Theorem***: Model class *H* finite, dataset *D* with *m* i.i.d. samples, $0 < \varepsilon < 1$: for any learned classifier $h \in H$:

$$P\left(|\text{error}_{true}(h) - \text{error}_{train}(h)| \geq \epsilon\right) \leq 2|H|e^{-2m\epsilon^2} \leq \delta$$

Important: PAC bound holds for all *h*, but doesn't guarantee that algorithm finds best *h*!!!

Summary of PAC bounds for finite model classes

With probability $\geq 1-\delta$,

1) For all $h \in H$ s.t. $\text{error}_{\text{train}}(h) = 0$,

$$\text{error}_{\text{true}}(h) \leq \varepsilon = \frac{\ln|H| + \ln\frac{1}{\delta}}{m}$$

$\boxed{\text{Haussler's bound}}$

2) For all $h \in H$

$$|\text{error}_{\text{true}}(h) - \text{error}_{\text{train}}(h)| \leq \varepsilon = \sqrt{\frac{\ln|H| + \ln\frac{2}{\delta}}{2m}}$$

$\boxed{\text{Hoeffding's bound}}$

PAC bound and Bias-Variance tradeoff

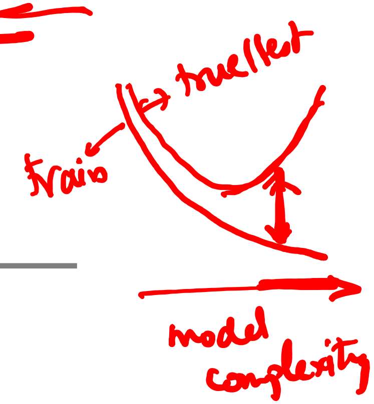
$$P\left(\left|\text{error}_{true}(h) - \text{error}_{train}(h)\right| \geq \epsilon\right) \leq 2|H|e^{-2m\epsilon^2} \leq \delta$$

- Equivalently, with probability $\geq 1 - \delta$

$$\text{error}_{true}(h) \leq \text{error}_{train}(h) + \sqrt{\frac{\ln|H| + \ln\frac{2}{\delta}}{2m}}$$

- Fixed m

Model class		
complex	small	large
simple	large	small



What about the size of the model class?

$$2|H|e^{-2m\epsilon^2} \leq \delta$$

- Sample complexity

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{2}{\delta}\right)$$

- How to measure the complexity of a model class?

  – E.g. decision trees:

    trees with depth k

    trees with k leaves

Number of decision trees of depth k

Recursive solution:

features

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{2}{\delta}\right)$$

Given *n* **binary** attributes

H_k = Number of **binary** decision trees of depth k

$H_0 = 2$

H_k = (#choices of root attribute)

*(# possible left subtrees)

*(# possible right subtrees) $= n * H_{k-1} * H_{k-1}$

$\log_2 \rightarrow \log_2 n + 2\underbrace{\log_2 H_{k-1}}_{L_{k-1}}$

Write $L_k = \log_2 H_k$

$L_0 = 1$

$L_k = \log_2 n + 2L_{k-1} = \log_2 n + 2(\log_2 n + 2L_{k-2})$

$= \log_2 n + 2\log_2 n + 2^2\log_2 n + \ldots + 2^{k-1}(\log_2 n + 2L_0)$

So $L_k = (2^k-1)(1+\log_2 n) + 1$ $\Rightarrow H_k \sim 2^{2^k}$, $m \sim 2^k$

PAC bound for decision trees of depth k

$$m \geq \frac{\ln 2}{2\epsilon^2}\left((2^k - 1)(1 + \log_2 n) + 1 + \log_2 \frac{2}{\delta}\right)$$

- Bad!!!
  - Number of points is exponential in depth k!

- But, for *m* data points, decision tree can't get too big...

Number of leaves never more than number data points, so we are over-counting a lot!

Number of decision trees with k leaves

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{2}{\delta}\right)$$

H_k = Number of binary decision trees with k leaves

H_1 =2

H_k = (#choices of root attribute) *

[(# left subtrees wth 1 leaf)*(# right subtrees wth k-1 leaves)

+ (# left subtrees wth 2 leaves)*(# right subtrees wth k-2 leaves)

+ …

+ (# left subtrees wth k-1 leaves)*(# right subtrees wth 1 leaf)]

$$\underbrace{H_k = n\sum_{i=1}^{k-1} H_i H_{k-i}} = n^{k-1} C_{k-1}$$

(C_{k-1} : Catalan Number)

Loose bound (using Sterling's approximation):

$$H_k \leq n^{k-1} 2^{2k-1}$$

→ m ~ 2k

Number of decision trees

- With k leaves

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{2}{\delta}\right)$$

$$\log_2 H_k \leq (k-1)\log_2 n + 2k - 1$$

linear in k

number of points m is linear in #leaves

- With depth k

$\log_2 H_k = (2^k\text{-}1)(1+\log_2 n) + 1$

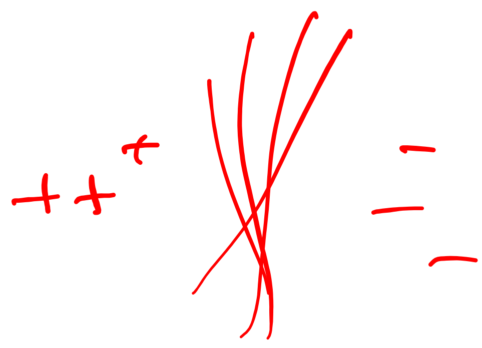
exponential in k

number of points m is exponential in depth

What did we learn from decision trees?

- Moral of the story:

  Complexity of learning not measured in terms of size of model space, but in maximum *number of points* that can be classified using a classifier from this model space



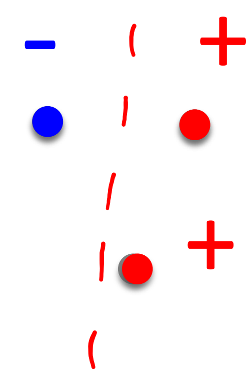
Rademacher Complexity

size of model space

- Instead of ~~all possible labelings~~, measure complexity by how accurately a model space can match a random labeling of the data.

For each data point i, draw random label

σ_i s.t. $P(\sigma_i = +1) = ½ = P(\sigma_i = -1)$



Then empirical Rademacher complexity of H is

$$\widehat{R}_m(H) = \mathbb{E}_\sigma \left[\sup_{h \in H} \left(\frac{1}{m} \sum_{i=1}^{m} \sigma_i h(X_i) \right) \right]$$

Max correlation possible with random labels

Rademacher Bounds

$\sqrt{\frac{\ln |H| + \ln 1/\delta}{m}}$

- With probability ≥ 1-δ,

$$\text{error}_{true}(h) \leq \text{error}_{train}(h) + \widehat{R}_m(H) + 3\sqrt{\frac{\log(2/\delta)}{m}}$$

where empirical Rademacher complexity of H

$$\widehat{R}_m(H) = \mathbb{E}_\sigma\left[\sup_{h\in H}\left(\frac{1}{m}\sum_{i=1}^{m}\sigma_i h(X_i)\right)\right]$$

is purely data-dependent.

Finite model class

- Rademacher complexity can be upper bounded in terms of model class size |H|:

$$\widehat{R}_m(H) \leq \sqrt{\frac{2\ln|H|}{m}}$$

- Often Rademacher bounds are significantly better, e.g. …

Linear models with bounded norm

- Consider $h(X_i) = < w, X_i >$ with fixed $\|w\|, \|X_i\| \leq R$

$$\widehat{R}_m(H) = \mathbb{E}_\sigma \left[\sup_{h \in H} \left(\frac{1}{m} \sum_{i=1}^{m} \sigma_i h(X_i) \right) \right]$$

⋮

$$\leq \frac{\|w\| R}{\sqrt{m}}$$

vs. $\sqrt{\frac{\ln |H|}{m}} = \infty$

Complexity increases with number of parameters d and norm of weights

Summary of PAC bounds

With probability ≥ 1-δ,

1) for all $h \in H$ s.t. $error_{train}(h) = 0$,

$$error_{true}(h) \le \varepsilon = \frac{\ln|H| + \ln\frac{1}{\delta}}{m}$$

2) for all $h \in H$,

$$|error_{true}(h) - error_{train}(h)| \le \varepsilon = \sqrt{\frac{\ln|H| + \ln\frac{2}{\delta}}{2m}}$$

Finite hypothesis space

3) For all $h \in H$,

Infinite hypothesis space

$$|error_{true}(h) - error_{train}(h)| \le \varepsilon = \widehat{R}_m(H) + 3\sqrt{\frac{\log(2/\delta)}{m}}$$