Introduction to Machine Learning

Lecture 13: PAC Learning

Nov 18, 2019

Jie Wang

Department of Electronic Engineering and Information Science (EEIS)

http://staff.ustc.edu.cn/~jwangx/

jiewangx@ustc.edu.cn

Machine Intelligence Research and Applications Lab

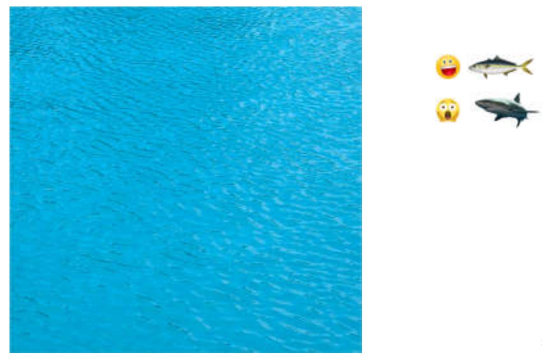
Contents

- Introduction
- Probably Approximately Correct (PAC)
- Quick Review of Probability
- Sample Complexity
- Learning Positive Half-lines
- PAC Results in General - Finite Hypotheses Space
- VC-dimension

Introduction

An Example Problem

Where to drop my hook?



An Example Problem

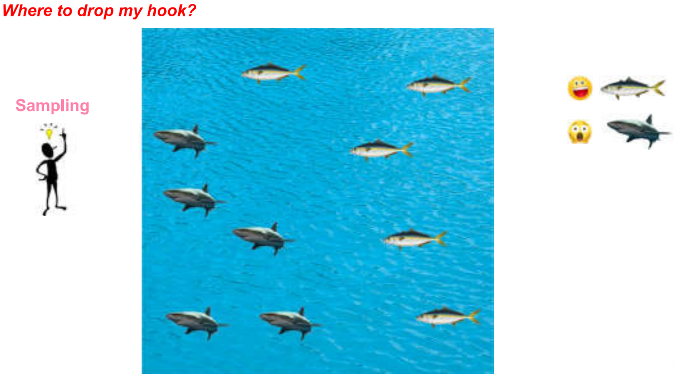
Where to drop my hook?

From God's Perspective (Unknown)

How to map the shark area accurately?

An Example Problem

Where to drop my hook?



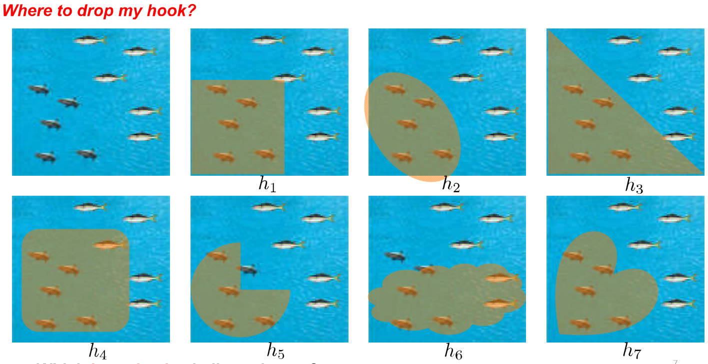
An Example Problem

Where to drop my hook?

Which sampling strategy is better?

An Example Problem

Where to drop my hook?



Which hypothesis shall we choose?

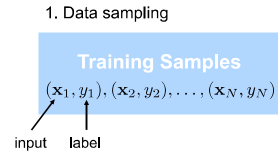
The Scientific Problems

- ***Which hypothesis shall we choose?***
  - Intuitively, we call the strategy to select a hypothesis from the hypothesis set “the learner”

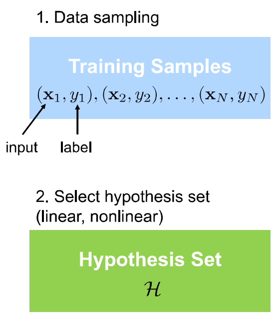
$$\mathcal{H} = \{h_1, h_2, \ldots, h_7\}$$

- ***How to measure the performance of the learner? Does it always work?***
  - The samples can be misleading

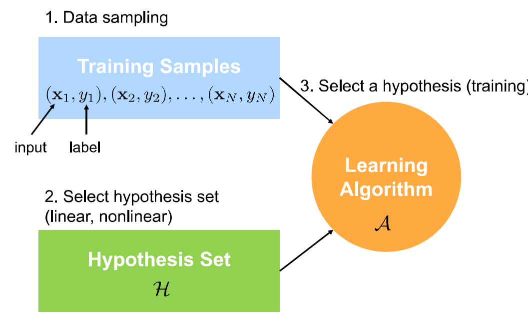
Pipeline of a Typical ML Solution



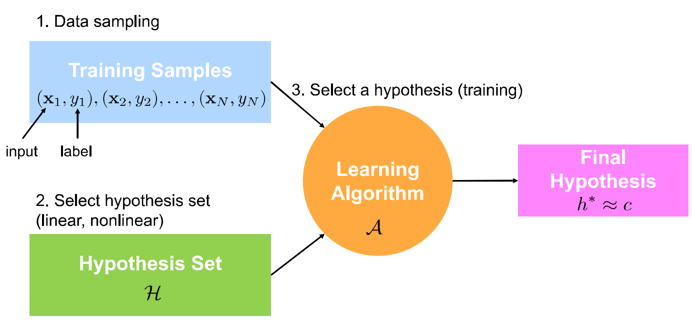
Pipeline of a Typical ML Solution



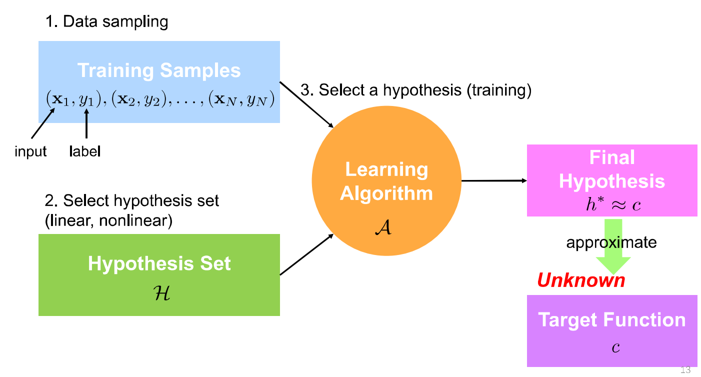
Pipeline of a Typical ML Solution



Pipeline of a Typical ML Solution



Pipeline of a Typical ML Solution



The Best Solution?

How to find the best approximation?

1. In terms of what (metrics)
2. Fair comparison
3. ……

The most influential factors

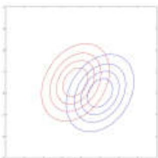
1. The sampled data instances
2. The hypothesis set
3. The learning algorithms
4. ……

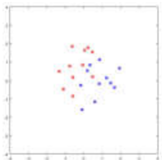
Data Sampling

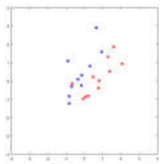
1. Data sampling

Training Samples
$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)$

"The quality of the sampled data determines the upper bound of the learning algorithms' performances"


The underlying (unknown) distribution


A good sample

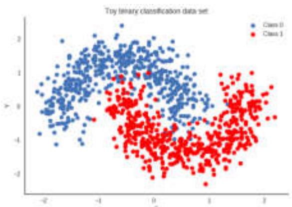

A bad (misleading) sample

Hypothesis Set

2. Select hypothesis set
(linear, nonlinear)

Hypothesis Set
$\mathcal{H}$

"The hypotheses should have sufficient expressive power"


https://docs.pymc.io/notebooks/bayesian_neural_network_with_sgfs.html

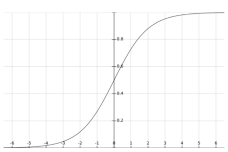
1. Linear classifier is not a good idea
2. It is not easy to visualize the high dimensional data

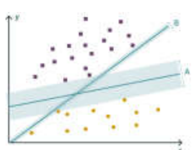
Learning Algorithm

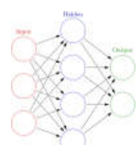
3. Select a hypothesis (training)

Learning Algorithm
$\mathcal{A}$

"Different learning algorithms are based on different assumptions"


Logistic Regression: probabilistic discriminative models


SVM: maximum margin
https://dsc-spidal.github.io/harp/docs/examples/svm/


Neural Networks: bio. sys.
wikipedia

Target Function

Unknown

Target Function
c

"Although the target function is unknown, a good news is that, the more data instances we sample, the more accurate we can picture it (with high probability)"

Example: The weak law of large numbers

Let $X_1, X_2, \ldots, X_n$ be independent identically distributed random variables with mean μ.
For every $\epsilon > 0$, we have

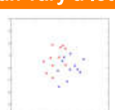
$$\mathbf{P}\left(\left|\frac{X_1 + \ldots + X_n}{n} - \mu\right| \geq \epsilon\right) \to 0, \text{ as } n \to \infty.$$

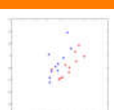
We can compute the expectation of an arbitrary random variable very accurately ***without*** knowing its distribution.

Random variable is indeed a ***function*** defined over the ***sample space.***

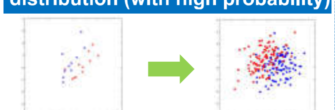
Summary

The quality of the data samples can vary a lot


good sample


bad sample

More samples can help to better approximates the underlying distribution (with high probability)



How many training samples are necessary or sufficient for successful learning, i.e., the hypothesis computed by learning algorithms could well approximate the (unknown) target function, (of course, with high probability)?

Quick Review of Probability

Probabilistic Models

- **Sample space** Ω
  - Ω is the set of all possible outcomes (elementary events) in a trial of an experiment.
- **Events set** $\mathcal{F}$
  - $\mathcal{F}$ is a collection of subsets of Ω which forms a $\sigma - algebra$, that is, $\mathcal{F}$ is closed under completion and countable union (therefore also countable intersection). The certain set Ω and the impossible set $\emptyset$ belong to $\mathcal{F}$.
- **Probability distribution** $\mathbf{P}$
  - $\mathbf{P}$ is a mapping from the events set $\mathcal{F}$ to $[0, 1]$, such that
    - (Nonnegativity) $\mathbf{P}(A) \geq 0, \forall A \in \mathcal{F}$.
    - (Additivity) $\mathbf{P}(A \cup B) = \mathbf{P}(A) + \mathbf{P}(B), \forall A, B \in \mathcal{F}, A \cap B = \emptyset$.
      More generally, if $A_1, A_2, \ldots,$ is a sequence of disjoint events, then the probability of their union satisfies
      $$\mathbf{P}(A_1 \cup A_2 \cup \ldots) = \mathbf{P}(A_1) + \mathbf{P}(A_2) + \ldots.$$
    - (Normalization) $\mathbf{P}(\Omega) = 1$.

Probabilistic Models

- **Sample space** Ω
  - Ω is the set of all possible outcomes (elementary events) in a trial of an experiment.
- **Events set** $\mathcal{F}$
  - $\mathcal{F}$ is a collection of subsets of Ω which forms a $\sigma - algebra$, that is, $\mathcal{F}$ is closed under completion and countable union (therefore also countable intersection). The certain set Ω and the impossible set ϕ belong to $\mathcal{F}$.
- **Probability distribution** $\mathbf{P}$
  - $\mathbf{P}$ is a mapping from the events set $\mathcal{F}$ to $[0, 1]$, such that

    Probability Axioms
    - (Nonnegativity) $\mathbf{P}(A) \geq 0, \forall A \in \mathcal{F}$.
    - (Additivity) $\mathbf{P}(A \cup B) = \mathbf{P}(A) + \mathbf{P}(B), \forall A, B \in \mathcal{F}, A \cap B = \emptyset$.
      More generally, if $A_1, A_2, \ldots,$ is a sequence of disjoint events, then the probability of their union satisfies
      $$\mathbf{P}(A_1 \cup A_2 \cup \ldots) = \mathbf{P}(A_1) + \mathbf{P}(A_2) + \ldots.$$
    - (Normalization) $\mathbf{P}(\Omega) = 1$.

Random Variable

- **Random variable.** Given a probability triple $(\Omega, \mathcal{F}, \mathbf{P})$, a random variable is a function X from Ω to the real numbers $\mathbb{R}$, such that
$$\{\omega \in \Omega : X(\omega) \leq x\} \in \mathcal{F},\ x \in \mathbb{R}.$$
- For a random variable X and a real number x, the probability
$$\mathbf{P}(X \leq x)$$
is indeed an abbreviation of
$$\mathbf{P}(\{\omega \in \Omega : X(\omega) \leq x\}).$$

Probably Approximately Correct (PAC)

Goal

How many training samples are necessary or sufficient for successful learning, i.e., *the hypothesis computed by learning algorithms could well approximate the (unknown) target function with high probability*?

Goal

How many training samples are necessary or sufficient for successful learning, i.e., *the hypothesis computed by learning algorithms could well approximate the (unknown) target function with high probability*?

successful learning

C correct

Goal

How many training samples are necessary or sufficient for successful learning, i.e., *the hypothesis computed by learning algorithms could well approximate the (unknown) target function with high probability*?

high probability

P probably

successful learning

C correct

Goal

How many training samples are necessary or sufficient for successful learning, i.e., *the hypothesis computed by learning algorithms could well approximate the (unknown) target function with high probability*?

high probability

P probably

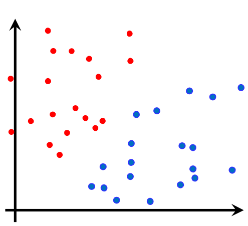
successful learning

C correct

approximation

A approximately

Problem Setting

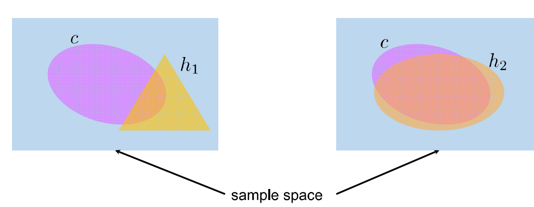


We study a binary classification problem

- X: the instance space
- $c : X \to \{0, 1\}$: the target function
- D : the training examples are generated at random from X according to some (unknown) probability distribution $\mathcal{D}$
- H : the hypothesis set considered by the learner

Approximation: Error of a Hypothesis I

- Which hypothesis better approximates the target concept?



- We need to find a metric to measure the distance between two functions: from the hypothesis set and h from the target concept set c

Approximation: Error of a Hypothesis II

- Distance between two functions (random variables): h from the hypothesis set and c from the target function set

> ***Definition:* The true error** (denoted by $error_{\mathcal{D}}(h)$) of hypothesis h with respect to concept c and distribution $\mathcal{D}$ is the probability that h will misclassify an instance drawn at random according to $\mathcal{D}$.
>
> $$error_{\mathcal{D}}(h) := \mathbf{P}_{x\sim\mathcal{D}}(c(x) \neq h(x)).$$

- The true error depends on the unknown distribution $\mathcal{D}$, and it is unrelated to samples
- For real problems, the learner can only observe the performance of h over the training examples
- **Training error**: the fraction of training examples misclassified by h

Logic of the Learner

- Given only the performances of a set of hypotheses over the training examples, how shall the learner pick one of them to approximate the target concept?

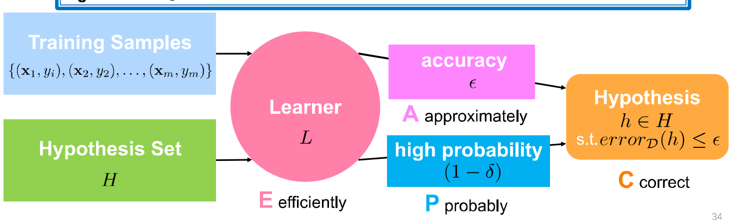
small training error → **small true error**

Could be WRONG!

- How probable is it that the observed training error for h gives a misleading estimate of the true error $error_{\mathcal{D}}(h)$?

PAC Learnability

> ***Definition:*** Consider a concept class C defined over a set of instances X of dimension n and a learner L using hypothesis space H. C is ***PAC-learnable*** by L using H if for all $c \in C$, distributions $\mathcal{D}$ over X, $\epsilon, \delta \in (0,1)$, learner L will with probability at least $(1-\delta)$ output a hypothesis $h \in H$ such that $error_{\mathcal{D}}(h) \leq \epsilon$, if the sample size $m \geq poly(1/\epsilon, 1/\delta, n, size(c))$. If further, L runs in $poly(1/\epsilon, 1/\delta, n, size(c))$, then C is said to be ***efficiently PAC-learnable***, and L is called a ***PAC-learning algorithm*** for C.



PAC Learnability

How many training samples are necessary or sufficient for successful learning, i.e., *the hypothesis computed by learning algorithms could well approximate the (unknown) target function with high probability*?

in the language of PAC Learning

Given parameters ϵ and δ, how many training samples did the leaner L need to output a hypothesis $h \in H$ such that $error_{\mathcal{D}}(h) \leq \epsilon$ with probability $(1-\delta)$ at a time cost of $\mathrm{poly}(1/\epsilon, 1/\delta, n, size(c))$?

Sample Complexity

Sample Complexity

> ***Definition: sample complexity*** is the least number of training samples required to guarantee the PAC solution.
>
> $$m \geq \mathrm{poly}(1/\epsilon, 1/\delta, n, size(c))$$

In most practical settings, the limited availability of training data is the factor that most limits the success of the learner.

Remarks of PAC Learnability

> ***Definition:*** Consider a concept class C defined over a set of instances X of dimension n and a learner L using hypothesis space H. C is ***PAC-learnable*** by L using H if for all $c \in C$, distributions $\mathcal{D}$ over X, $\epsilon, \delta \in (0,1)$, learner L will with probability at least $(1-\delta)$ output a hypothesis $h \in H$ such that $error_{\mathcal{D}}(h) \leq \epsilon$, if the sample size $m \geq poly(1/\epsilon, 1/\delta, n, size(c))$. If further, L runs in $poly(1/\epsilon, 1/\delta, n, size(c))$, then C is said to be ***efficiently PAC-learnable***, and L is called a ***PAC-learning algorithm*** for C.

- The above definition of PAC learning has an implicit assumption that *any* target concept $c \in C$ can be approximated by a hypothesis $h \in H$ with any predefined accuracy.
- The above claim has a more concise form: the hypothesis set H contains the closure of the target concept class C.

$$|H| < \infty \quad + \quad C \subset \bar{H} \quad \Rightarrow \quad C \subset H$$

A special case where the hypothesis set contains a finite number of hypotheses.

Consistent Learner

> ***Definition:*** A learner is ***consistent*** if it outputs hypotheses that perfectly fit the training data, whenever possible.

- Examples:

  Logistic Regression, SVM, Least Squares, Decision Tree…

  without regularization

- We shall derive a bound on the number of training examples required by consistent learner.

Logic of Consistent Learner I

- Why consistent learner?

$|H| < \infty$ + $C \subset \bar{H}$ $\Rightarrow$ $C \subset H$

Logic of Consistent Learner I

- Why consistent learner?

$|H| < \infty$ + $C \subset \bar{H}$ $\Rightarrow$ $C \subset H$

- The consistent learner will reject the hypotheses that do not match the training examples, as they can not be the target concepts of interests.

Why?

Definition: Version space is the set of all hypotheses $h \in H$ that correctly classify the training examples D.

$$VS_{H,D} \equiv \{h \in H : h(x) = c(x), \forall (x, c(x)) \in D\}.$$

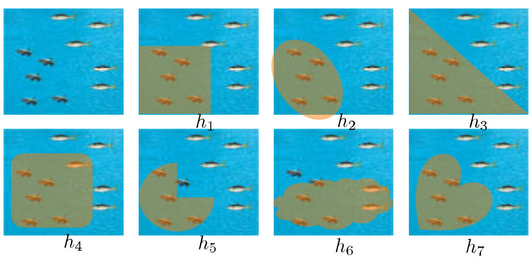
The version space is said to be ϵ - exhausted if

$$error_{\mathcal{D}}(h) < \epsilon, \forall h \in VS_{H,D}.$$

Logic of Consistent Learner I

Definition: Version space is the set of all hypotheses $h \in H$ that correctly classify the training examples D.

$$VS_{H,D} \equiv \{h \in H : h(x) = c(x), \forall (x, c(x)) \in D\}.$$



What is the version space?

Logic of Consistent Learner II

- The target concept is in the version space
- The learner can reject a hypothesis once it found a mismatched sample in the training data
- The more samples in the training data, the higher probability the learner can screen bad hypotheses with

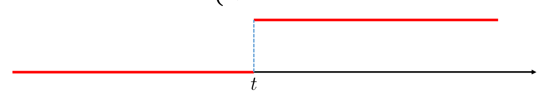
Learning Positive Half-Lines

Problem Settings

- We would like to learn an unknown target concept

$$c(x) = \begin{cases} 1, & \text{if } x \geq t \\ 0, & \text{otherwise} \end{cases}$$

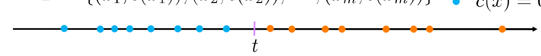
The concept class:
$C = \{\text{positive half-lines}\}$



- We randomly sample a set of i.i.d. data points

$$D = \{(x_1, c(x_1)), (x_2, c(x_2)), \ldots, (x_m, c(x_m))\}$$

- $c(x) = 1$
- $c(x) = 0$



- According to the samples, how can we find a hypothesis $h(x)$ from the hypotheses space H to approximate the target concept $c(x)$?

The hypotheses space:
$H = C$

A Consistent Learner

The Algorithm

data samples D

↓

Learner L
Find:
$x^{(0)} = \max\{x_i \in D : c(x_i) = 0\}$
$x^{(1)} = \min\{x_i \in D : c(x_i) = 1\}$

↓

Output $\hat{t} \in (x^{(0)}, x^{(1)}]$

$$h(x) = \begin{cases} 1, & \text{if } x \geq \hat{t} \\ 0, & \text{otherwise} \end{cases}$$

A Consistent Learner

The Algorithm

data samples D

↓

Learner L
Find:
$x^{(0)} = \max\{x_i \in D : c(x_i) = 0\}$
$x^{(1)} = \min\{x_i \in D : c(x_i) = 1\}$

↓

Output $\hat{t} \in (x^{(0)}, x^{(1)}]$

$$h(x) = \begin{cases} 1, & \text{if } x \geq \hat{t} \\ 0, & \text{otherwise} \end{cases}$$

- The consistent learner may gives us a bad hypothesis only if the version space contains a bad hypothesis

A Consistent Learner

The Algorithm

data samples D

Learner L

Find:
$x^{(0)} = \max\{x_i \in D : c(x_i) = 0\}$
$x^{(1)} = \min\{x_i \in D : c(x_i) = 1\}$

Output $\hat{t} \in (x^{(0)}, x^{(1)}]$

$$h(x) = \begin{cases} 1, & \text{if } x \geq \hat{t} \\ 0, & \text{otherwise} \end{cases}$$

- The consistent learner may gives us a bad hypothesis only if the version space contains a bad hypothesis

  $\mathbf{P}[h \in VS_{H,D} \text{ and } error_{\mathcal{D}}(h) > \epsilon]$

A Consistent Learner

The Algorithm

data samples D

Learner L

Find:
$x^{(0)} = \max\{x_i \in D : c(x_i) = 0\}$
$x^{(1)} = \min\{x_i \in D : c(x_i) = 1\}$

Output $\hat{t} \in (x^{(0)}, x^{(1)}]$

$$h(x) = \begin{cases} 1, & \text{if } x \geq \hat{t} \\ 0, & \text{otherwise} \end{cases}$$

- The consistent learner may gives us a bad hypothesis only if the version space contains a bad hypothesis

  $\mathbf{P}[h \in VS_{H,D} \text{ and } error_{\mathcal{D}}(h) > \epsilon]$

- An even more scary form

  $\mathbf{P}[h \in VS_{H,D} \text{ and } \mathbf{P}[h(x) \neq c(x)] > \epsilon]$

A Consistent Learner

The Algorithm

data samples D

Learner L

Find:
$x^{(0)} = \max\{x_i \in D : c(x_i) = 0\}$
$x^{(1)} = \min\{x_i \in D : c(x_i) = 1\}$

Output $\hat{t} \in (x^{(0)}, x^{(1)}]$

$$h(x) = \begin{cases} 1, & \text{if } x \geq \hat{t} \\ 0, & \text{otherwise} \end{cases}$$

- The consistent learner may gives us a bad hypothesis only if the version space contains a bad hypothesis

  $\mathbf{P}[h \in VS_{H,D} \text{ and } error_{\mathcal{D}}(h) > \epsilon]$

- An even more scary form

  $\mathbf{P}[h \in VS_{H,D} \text{ and } \mathbf{P}[h(x) \neq c(x)] > \epsilon]$

- What are the sample spaces regarding the two different probabilities?

A Consistent Learner

The Algorithm

data samples D

Learner L

Find:
$x^{(0)} = \max\{x_i \in D : c(x_i) = 0\}$
$x^{(1)} = \min\{x_i \in D : c(x_i) = 1\}$

Output $\hat{t} \in (x^{(0)}, x^{(1)}]$

$$h(x) = \begin{cases} 1, & \text{if } x \geq \hat{t} \\ 0, & \text{otherwise} \end{cases}$$

- The consistent learner may gives us a bad hypothesis only if the version space contains a bad hypothesis

  $\mathbf{P}[h \in VS_{H,D} \text{ and } error_{\mathcal{D}}(h) > \epsilon]$

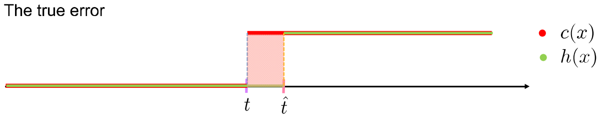
- An even more scary form

  $\mathbf{P}[h \in VS_{H,D} \text{ and } \mathbf{P}[h(x) \neq c(x)] > \epsilon]$

- What are the sample spaces regarding the two different probabilities?
  - For the inside $\mathbf{P}$: $\mathbb{R}$
  - For the outside $\mathbf{P}$: $VS_{H,D}$

Error Region

- The true error



$$error_{\mathcal{D}}(h) = \mathbf{P}(c(x) \neq h(x)) = \int_t^{\hat{t}} p(x)dx$$

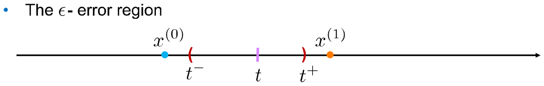
- The ϵ- error region

$$t^+ = \sup\left\{\tau \geq t : \int_t^{\tau} p(x)dx = \epsilon\right\}$$

$$t^- = \inf\left\{\tau \leq t : \int_{\tau}^{t} p(x)dx = \epsilon\right\}$$

Probability of Picking up a Bad Hypothesis I

- The ϵ- error region



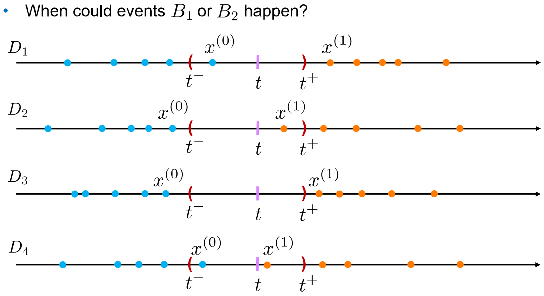
$$h \in VS_{H,D} \text{ and } error_{\mathcal{D}}(h) > \epsilon \Leftrightarrow \hat{t} \in (x^{(0)}, x^{(1)}] \setminus [t^-, t^+] \Leftrightarrow \hat{t} \in (x^{(0)}, t^-) \cup (t^+, x^{(1)}]$$

Event: B $\quad B_1 \quad B_2$

$$\mathbf{P}[B] = \mathbf{P}[B_1 \cup B_2] = \mathbf{P}[B_1] + \mathbf{P}[B_2]$$

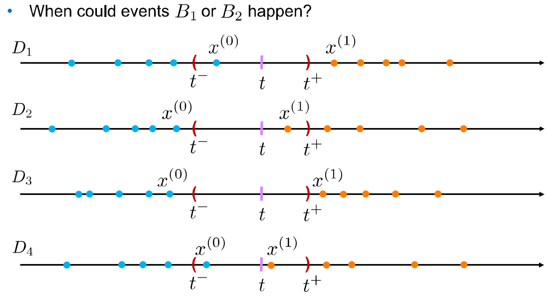
Probability of Picking up a Bad Hypothesis II

- When could events B_1 or B_2 happen?



Probability of Picking up a Bad Hypothesis II

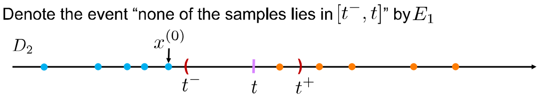
- When could events B_1 or B_2 happen?



Answer: B_2; B_1; B_1 and B_2; none

Probability of Picking up a Bad Hypothesis III

- Denote the event "none of the samples lies in $[t^-, t]$" by E_1



- Event $B_1 : \hat{t} \in (x^{(0)}, t^-)$ happens only if E_1 happens

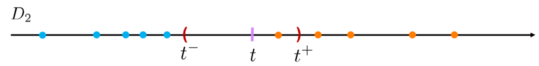
$$B_1 \subset E_1 \Leftrightarrow \mathbf{P}[B_1] \leq \mathbf{P}[E_1]$$

- How to find $\mathbf{P}[E_1]$?

$$\mathbf{P}[E_1] = \mathbf{P}\left[x_1 \notin [t^-, t] \cap x_2 \notin [t^-, t] \cap \cdots x_m \notin [t^-, t]\right]$$

Probability of Picking up a Bad Hypothesis III

- Denote the event "none of the samples lies in $[t^-, t]$" by E_1



- Event $B_1 : \hat{t} \in (x^{(0)}, t^-)$ happens only if E_1 happens

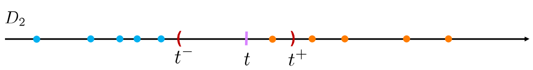
$$B_1 \subset E_1 \Leftrightarrow \mathbf{P}[B_1] \leq \mathbf{P}[E_1]$$

- How to find $\mathbf{P}[E_1]$?

$$\begin{aligned}\mathbf{P}[E_1] =& \mathbf{P}\left[x_1 \notin [t^-, t] \cap x_2 \notin [t^-, t] \cap \cdots x_m \notin [t^-, t]\right] \\ =& \prod_{i=1}^{m} \mathbf{P}\left[x_i \notin [t^-, t]\right]\end{aligned}$$

Probability of Picking up a Bad Hypothesis III

- Denote the event "none of the samples lies in $[t^-, t]$" by E_1



- Event $B_1 : \hat{t} \in (x^{(0)}, t^-)$ happens only if E_1 happens

$$B_1 \subset E_1 \Leftrightarrow \mathbf{P}[B_1] \leq \mathbf{P}[E_1]$$

- How to find $\mathbf{P}[E_1]$?

$$\begin{aligned}\mathbf{P}[E_1] =& \mathbf{P}\left[x_1 \notin [t^-, t] \cap x_2 \notin [t^-, t] \cap \cdots x_m \notin [t^-, t]\right] \\ =& \prod_{i=1}^{m} \mathbf{P}\left[x_i \notin [t^-, t]\right] \\ =& (1-\epsilon)^m\end{aligned}$$

Probability of Picking up a Bad Hypothesis IV

- We can similarly define E_2 and find its probability

- Putting all together

$$\mathbf{P}[h \in VS_{H,D} \text{ with } error_{\mathcal{D}}(h) > \epsilon] = \mathbf{P}[B_1] + \mathbf{P}[B_2] \leq \mathbf{P}[E_1] + \mathbf{P}[E_2] \leq 2(1-\epsilon)^m \underset{\uparrow \text{why}}{\leq} 2e^{(-m\epsilon)}$$

- If we would like this probability (for picking up a bad hypothesis) be bounded by a predefined value δ, then we need the number of samples

$$m \geq \frac{1}{\epsilon} \ln \frac{2}{\delta}$$

PAC Bound in General – Finite Hypotheses Space

PAC Bound - Consistent Learner I

Theorem: If the hypothesis space H is finite and D is a sequence of $m \geq 1$ independently randomly drawn samples of some target concept c, then for any $\epsilon \in (0,1)$, the probability that the version space contains at least one hypothesis h with $error_{\mathcal{D}}(h) > \epsilon$ is less than or equal to

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

PAC Bound - Consistent Learner I

Theorem: If the hypothesis space H is finite and D is a sequence of $m \geq 1$ independently randomly drawn samples of some target concept c, then for any $\epsilon \in (0,1)$, the probability that the version space contains at least one hypothesis h with $error_{\mathcal{D}}(h) > \epsilon$ is less than or equal to

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

Proof: Assume that $\{h_1, \ldots, h_k\} = H_0 \subset H$ with $error_{\mathcal{D}}(h_i) > \epsilon, \forall\, i = 1, \ldots, k.$

$$\begin{aligned}&\mathbf{P}[VS_{H,D} \text{ contains at least one } h \in H_0] \\ =&\mathbf{P}[h_1 \in VS_{H,D} \cup \ldots \cup h_k \in VS_{H,D}] \\ \leq& \sum_{I=1}^{k} \mathbf{P}[h_i \in VS_{H,D}]\end{aligned}$$

+

$$\begin{aligned}&\mathbf{P}[h_i \in VS_{H,D}] \\ =&\mathbf{P}[h_i(x_1) = c(x_1) \cap \ldots \cap h_i(x_m) = c(x_m)] \\ =&\prod_{j=1}^{m} \mathbf{P}[h_i(x_j) = c(x_j)] \\ =&\prod_{j=1}^{m} (1 - \mathbf{P}[h_i(x_j) \neq c(x_j)]) \\ \leq& (1-\epsilon)^m \leq e^{-\epsilon m}\end{aligned}$$

PAC Bound - Consistent Learner II

Theorem: If the hypothesis space H is finite and D is a sequence of $m \geq 1$ independently randomly drawn samples of some target concept c, then for any $\epsilon \in (0,1)$, the probability that the version space contains at least one hypothesis h with $error_{\mathcal{D}}(h) > \epsilon$ is less than or equal to

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

- We can use this theorem to determine the number of training samples required to reduce this probability of failure below some desired level δ.

$$|H|e^{-\epsilon m} \leq \delta \Rightarrow m \geq \frac{1}{\epsilon}(\ln |H| + \ln(1/\delta))$$

Agnostic Learning

- In most practical settings, the target concept is not in the hypothesis set and the consistent learner may not work.
- Still, the learner can output the hypothesis that has the smallest training error.

Definition: A learner is ***agnostic*** if it makes no assumption that the target concept is representable by H, and outputs the hypothesis with minimum training error.

PAC Bound - Agnostic Learner I

Theorem: If the hypothesis space H is finite and D is a sequence of $m \geq 1$ independently randomly drawn samples of some target concept c, then for any $\epsilon \in (0,1)$, the probability that H contains at least one hypothesis h with $error_{\mathcal{D}}(h) \geq error_D(h) + \epsilon$ is less than or equal to

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

training error

PAC Bound - Agnostic Learner I

Hoeffding's Inequality: Let $X_1, \ldots, X_m$ be a sequence of independent random variables such that $\mathbf{P}[a \leq X_i \leq b] = 1$ for all $i = 1, 2, \ldots, m$. Then

$$\mathbf{P}\left[\frac{1}{m}\sum_i^m (X_i - \mathbf{E}(X_i)) \geq \epsilon\right] \leq \exp\left(-\frac{2m\epsilon^2}{(b-a)^2}\right)$$

and

$$\mathbf{P}\left[\frac{1}{m}\sum_i^m (X_i - \mathbf{E}(X_i)) \leq -\epsilon\right] \leq \exp\left(-\frac{2m\epsilon^2}{(b-a)^2}\right)$$

PAC Bound - Agnostic Learner I

Theorem: If the hypothesis space H is finite and D is a sequence of $m \geq 1$ independently randomly drawn samples of some target concept c, then for any $\epsilon \in (0,1)$, the probability that H contains at least one hypothesis h with $error_{\mathcal{D}}(h) \geq error_D(h) + \epsilon$ is less than or equal to

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

training error

Proof:

$$\mathbf{P}[error_{\mathcal{D}}(h) \geq error_D(h) + \epsilon] \leq e^{-2m\epsilon^2}$$

By Hoeffding inequality

$$\mathbf{P}[\exists h \in H : error_{\mathcal{D}}(h) \geq error_D(h) + \epsilon] \leq |H|e^{-2m\epsilon^2}$$

PAC Bound - Agnostic Learner II

Theorem: If the hypothesis space H is finite and D is a sequence of $m \geq 1$ independently randomly drawn samples of some target concept c, then for any $\epsilon \in (0,1)$, the probability that H contains at least one hypothesis h with $error_{\mathcal{D}}(h) \geq error_D(h) + \epsilon$ is less than or equal to

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

- We can use this theorem to determine the number of training samples required to reduce this probability of failure below some desired level δ.

$$|H|e^{-2m\epsilon^2} \leq \delta \Rightarrow m \geq \frac{1}{2\epsilon^2}(\ln|H| + \ln(1/\delta))$$

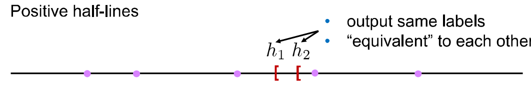
VC-dimension

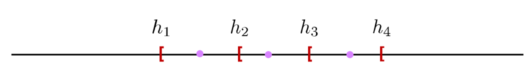
Infinite Hypotheses Space

- Due to the term $|H|$, the sample complexity becomes useless when the cardinality of the hypotheses space is huge or even infinite.
- However, we can still derive the sample complexity for the task of learning positive half-lines, where the cardinality of the hypotheses space is infinite.
- We need another measure for the complexity (expressive power) of the hypotheses space; that is, the Vapnik-Chervonenkis dimension.
- In many cases, the sample complexity based on VC-dimension will lead to much tighter bounds.

Growth Function - Examples

- Positive half-lines



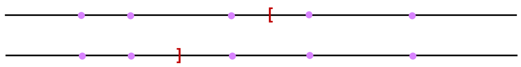


h_1	$+$	$+$	$+$
h_2	$-$	$+$	$+$
h_3	$-$	$-$	$+$
h_4	$-$	$-$	$-$

The number of equivalent classes is $m+1$; that is $\mathcal{O}(m)$.

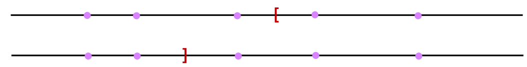
Growth Function - Examples

- Positive/negative half-lines



Growth Function - Examples

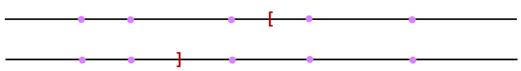
- Positive/negative half-lines



The number of equivalent classes is $2(m+1)-2=2m$; that is $\mathcal{O}(m)$.

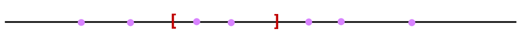
Growth Function - Examples

- Positive/negative half-lines



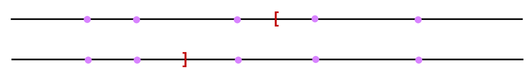
The number of equivalent classes is $2(m+1)-2=2m$; that is $\mathcal{O}(m)$.

- Intervals



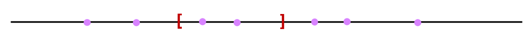
Growth Function - Examples

- Positive/negative half-lines



The number of equivalent classes is $2(m+1)-2=2m$; that is $\mathcal{O}(m)$.

- Intervals



The number of equivalent classes is $\binom{m+1}{2}+1$; that is $\mathcal{O}(m^2)$.

Growth Function - Definition

- Given a set of unlabeled samples $D=(x_1,x_2,\ldots,x_m)$, we define the set of all possible labels given by the hypotheses space H by

$$\prod_H(D)=\{h(x_1),h(x_2),\ldots,h(x_m):h\in H\}$$

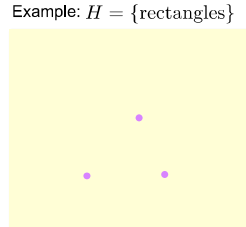
- Growth function

$$\prod_H(m)=\max_{|D|=m}|\prod_H(D)|$$

Shattering

Definition: A set of instances D is ***shattered*** by hypothesis space H if and only if for every possible labeling (dichotomy) of D, there exists some hypothesis in H consistent with this dichotomy.

Example: $H=\{\text{rectangles}\}$

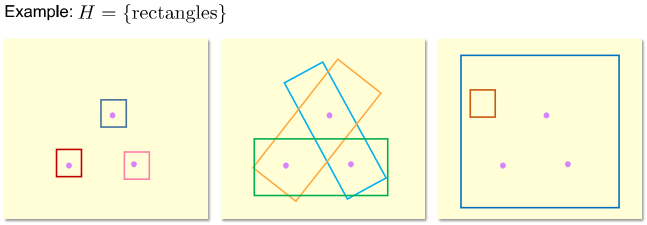


Can these three points be shattered by H?

Shattering

Definition: A set of instances S is ***shattered*** by hypothesis space H if and only if for every possible labeling (dichotomy) of S, there exists some hypothesis in H consistent with this dichotomy.

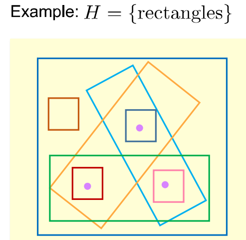
Example: $H=\{\text{rectangles}\}$



Shattering

Definition: A set of instances S is ***shattered*** by hypothesis space H if and only if for every possible labeling (dichotomy) of S, there exists some hypothesis in H consistent with this dichotomy.

Example: $H=\{\text{rectangles}\}$



Yes, they can be shattered

What about this one?

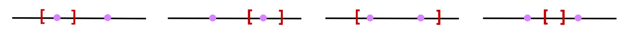
VC-dimension - Definition

> ***Definition:*** The ***Vapnik-Chervonenkis dimension***, $VC(H)$, of hypothesis space H defined over instance space X is the size of the **largest** finite subset of X shattered by H, that is $VC(H) = \max\left\{m : \prod_H(m) = 2^m\right\}$.
> If there exists arbitrarily large finite sets of X can be shattered by H, then $VC(H) = \infty$.

- For any finite H, we have $VC(H) \leq \log_2 |H|$
- VC dimension may depend on the dimension of the instance space.

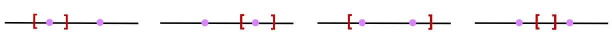
VC-dimension - Examples

Example: $H = \{\text{intervals}\}$



VC-dimension - Examples

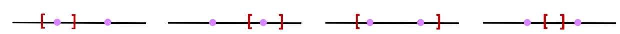
Example: $H = \{\text{intervals}\}$



$$VC(H) \geq 2$$

VC-dimension - Examples

Example: $H = \{\text{intervals}\}$

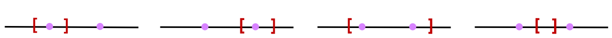


$$VC(H) \geq 2$$

What about three points?

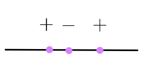
VC-dimension - Examples

Example: $H = \{\text{intervals}\}$



$$VC(H) \geq 2$$

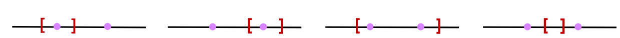
What about three points?



Any h?

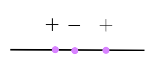
VC-dimension - Examples

Example: $H = \{\text{intervals}\}$



$$VC(H) \geq 2$$

What about three points?

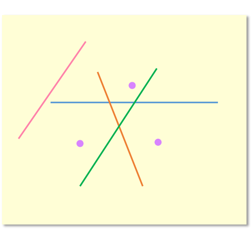


$$VC(H) = 2$$

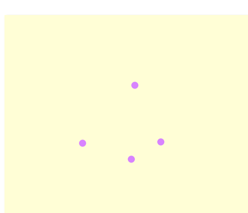
Any h?

VC-dimension - Examples

Example: $H = \{\text{linear classifier}\}$



$$VC(H) \geq 3$$



Is there any set of four points that can be shattered?

VC-dimension – sample complexity

> ***Theorem:*** Let H be the hypothesis space, D is a sequence of $m \geq 1$ independently randomly drawn samples of some target concept c, and $\epsilon, \delta \in (0, 1)$. Then, to ϵ-exhaust the version space with probability $1 - \delta$, we need
> $$m \geq \frac{1}{\epsilon}\left(4\log_2(2/\delta) + 8VC(H)\log_2(13/\epsilon)\right).$$

Questions