Machine Learning

CSE427

Mahbub Majumdar
Typeset by:
Syeda Ramisa Fariha

BRAC University
66 Mohakhali
Dhaka, Bangladesh

May 28, 2018

Inspiring Excellence

Superhumungous Thanks

These slides were typeset by Syeda Ramisa Fariha.

Without her tremendous dedication, these slides would not exist.

Table of Contents

Topics Covered In Previous Lectures

- Finite *hypothesis classes*

The Probability Approximately Correct Learning Model

Topics Covered In Previous Lectures

- Finite *hypothesis classes*
- Use *ERM* procedure on large enough S

Topics Covered In Previous Lectures

- Finite *hypothesis classes*
- Use *ERM* procedure on large enough S
- Then output will be *Probability Approximately Correct*

PAC Learnability

Definition: PAC Learnability: $\mathcal{H}$ is PAC Learnable if

- $\exists$ a function $m_{\mathcal{H}}(\epsilon, \delta) : (0,1) \times (0,1) \implies \mathbb{N}$, and a learning algorithm such that

The Probability Approximately Correct Learning Model

PAC Learnability

Definition: PAC Learnability: $\mathcal{H}$ is PAC Learnable if

- $\exists$ a function $m_{\mathcal{H}}(\epsilon, \delta) : (0,1) \times (0,1) \implies \mathbb{N}$, and a learning algorithm such that
- for all $\epsilon, \delta \in (0,1)$ and $\forall$ D over X and $\forall$ $f : X \to \{0,1\}$, if the *Realizabilty Assumption* holds, wrt $\mathcal{H}, D, f$,

The Probability Approximately Correct Learning Model

PAC Learnability

Definition: PAC Learnability: $\mathcal{H}$ is PAC Learnable if

- $\exists$ a function $m_{\mathcal{H}}(\epsilon, \delta) : (0,1) \times (0,1) \implies \mathbb{N}$, and a learning algorithm such that
- for all $\epsilon, \delta \in (0,1)$ and $\forall$ D over X and $\forall$ $f : X \to \{0,1\}$, if the *Realizabilty Assumption* holds, wrt $\mathcal{H}, D, f$,
- then, when running the Learning Algorithm on $m > m_{\mathcal{H}}(\epsilon, \delta)$ iid examples generated by D, and labeled by f,

The Probability Approximately Correct Learning Model

PAC Learnability

Definition: PAC Learnability: $\mathcal{H}$ is PAC Learnable if

- $\exists$ a function $m_{\mathcal{H}}(\epsilon, \delta) : (0,1) \times (0,1) \implies \mathbb{N}$, and a learning algorithm such that
- for all $\epsilon, \delta \in (0,1)$ and $\forall$ D over X and $\forall$ $f : X \rightarrow \{0,1\}$, if the *Realizabilty Assumption* holds, wrt $\mathcal{H}, D, f$,
- then, when running the Learning Algorithm on $m > m_{\mathcal{H}}(\epsilon, \delta)$ iid examples generated by D, and labeled by f,

$\Rightarrow$ the algorithm returns a hypothesis h, such that, with probability of at least $1 - \delta$ (over the choice of examples):

PAC Learnability

Definition: PAC Learnability: $\mathcal{H}$ is PAC Learnable if

- $\exists$ a function $m_{\mathcal{H}}(\epsilon, \delta) : (0,1) \times (0,1) \implies \mathbb{N}$, and a learning algorithm such that
- for all $\epsilon, \delta \in (0,1)$ and $\forall$ D over X and $\forall$ $f : X \to \{0,1\}$, if the *Realizabilty Assumption* holds, wrt $\mathcal{H}, D, f$,
- then, when running the Learning Algorithm on $m > m_{\mathcal{H}}(\epsilon, \delta)$ iid examples generated by D, and labeled by f,

$\Rightarrow$ the algorithm returns a hypothesis h, such that, with probability of at least $1 - \delta$ (over the choice of examples):

$\Rightarrow$ then

$$L_{D,f}(h) \leq \epsilon.$$

The Probability Approximately Correct Learning Model

PAC Learnability

Note:

ϵ : difference between output classifier and optimal classifier

δ : how likely h is inaccurate

- Might accidentally sample the same point over and over, $S = \{\text{single point}\}$

PAC Learnability

Note:

$$
\begin{aligned}
\epsilon &: \text{difference between output classifier and optimal classifier} \\
\delta &: \text{how likely } h \text{ is inaccurate}
\end{aligned}
$$

- Might accidentally sample the same point over and over, $S = \{\text{single point}\}$
- *Nonzero* ϵ enables forgiveness. The learner is allowed to make minor errors

Sample Complexity

- if $\mathcal{H}$ is *PAC learnable*
  $\Rightarrow$ many $m_{\mathcal{H}}$ satisfy requirements of *PAC learnability*
  $\Rightarrow$ define *sample complexity* as the minimal $m_{\mathcal{H}}(\epsilon, \delta)$

The Probability Approximately Correct Learning Model

Sample Complexity

- if $\mathcal{H}$ is *PAC learnable*
  $\Rightarrow$ many $m_{\mathcal{H}}$ satisfy requirements of *PAC learnability*
  $\Rightarrow$ define *sample complexity* as the minimal $m_{\mathcal{H}}(\epsilon, \delta)$

- Recall from last lecture: Every finite hypothesis class is PAC learnable with sample complexity $m_{\mathcal{H}} \leq \frac{1}{\epsilon} \log(\frac{|\mathcal{H}|}{\delta})$

Sample Complexity

- if $\mathcal{H}$ is *PAC learnable*
  $\Rightarrow$ many $m_{\mathcal{H}}$ satisfy requirements of *PAC learnability*
  $\Rightarrow$ define *sample complexity* as the minimal $m_{\mathcal{H}}(\epsilon, \delta)$
- Recall from last lecture: Every finite hypothesis class is PAC learnable with sample complexity $m_{\mathcal{H}} \leq \frac{1}{\epsilon} \log(\frac{|\mathcal{H}|}{\delta})$
- Because we have chosen the the sample complexity to be the *minimal* $m_{\mathcal{H}}$, it will not be bigger than $\frac{1}{\epsilon} \log(\frac{|\mathcal{H}|}{\delta})$. That's why we have $m_{\mathcal{H}} \leq \frac{1}{\epsilon} \log(\frac{|\mathcal{H}|}{\delta})$

The Probability Approximately Correct Learning Model

Sample Complexity

- if $\mathcal{H}$ is *PAC learnable*
  $\Rightarrow$ many $m_{\mathcal{H}}$ satisfy requirements of *PAC learnability*
  $\Rightarrow$ define *sample complexity* as the minimal $m_{\mathcal{H}}(\epsilon, \delta)$

- Recall from last lecture: Every finite hypothesis class is PAC learnable with sample complexity $m_{\mathcal{H}} \leq \frac{1}{\epsilon} \log(\frac{|\mathcal{H}|}{\delta})$

- Because we have chosen the the sample complexity to be the *minimal* $m_{\mathcal{H}}$, it will not be bigger than $\frac{1}{\epsilon} \log(\frac{|\mathcal{H}|}{\delta})$. That's why we have $m_{\mathcal{H}} \leq \frac{1}{\epsilon} \log(\frac{|\mathcal{H}|}{\delta})$

- There are infinite classes that are learnable also, *VC Dimension* determines learnability.

Removing Some Assumptions

- If we remove *Realizability*, this becomes an *Agnostic PAC Model*

Removing Some Assumptions

- If we remove *Realizability*, this becomes an *Agnostic PAC Model*
- Realizability requires $\exists\ h^*$ such that,

$$\mathbb{P}_{x\sim D}\Big(h^*(x) = f(x)\Big) = 1$$

The Probability Approximately Correct Learning Model

Removing Some Assumptions

- If we remove *Realizability*, this becomes an *Agnostic PAC Model*
- Realizability requires $\exists\ h^*$ such that,

$$\mathop{\mathbb{P}}_{x \sim D}\Big(h^*(x) = f(x)\Big) = 1$$

- In real life the output labels will not be fully determined by the *features* we measure

The Probability Approximately Correct Learning Model

Removing Some Assumptions

- If we remove *Realizability*, this becomes an *Agnostic PAC Model*
- Realizability requires $\exists\ h^*$ such that,

$$\mathop{\mathbb{P}}_{x\sim D}\Big(h^*(x) = f(x)\Big) = 1$$

- In real life the output labels will not be fully determined by the *features* we measure
- For example, player quality not determined by only *2* features such as, $x_i = \begin{pmatrix} speed_i \\ stamina_i \end{pmatrix}$

Removing Some Assumptions

- In the absence of realizability, often label the distribution $D(x)$ as $D(x, y)$. (Notation used by UML.)

Removing Some Assumptions

- In the absence of realizability, often label the distribution $D(x)$ as $D(x, y)$. (Notation used by UML.)

- $D(x, y) =$ Joint Probability Distribution over X and Y.

Removing Some Assumptions

- In the absence of realizability, often label the distribution $D(x)$ as $D(x, y)$. (Notation used by UML.)
- $D(x, y) =$ Joint Probability Distribution over X and Y.
- $D_x \equiv D(x) = \sum_{y_i} D(x, y)$ is the *marginal distribution* over unlabeled points.

The Probability Approximately Correct Learning Model

Removing Some Assumptions

- In the absence of realizability, often label the distribution $D(x)$ as $D(x, y)$. (Notation used by UML.)
- $D(x, y) =$ Joint Probability Distribution over X and Y.
- $D_x \equiv D(x) = \sum_{y_i} D(x, y)$ is the *marginal distribution* over unlabeled points.
- For example, $D_x \equiv$ probability that:

$$\begin{aligned} a \leq \quad & \mathit{feature}_1 \quad \leq b \\ c \leq \quad & \mathit{feature}_2 \quad \leq d \\ & \vdots \end{aligned}$$

The Probability Approximately Correct Learning Model

Removing Some Assumptions

- $D(y \mid x)$ = Conditional Probability of getting label y given x

The Probability Approximately Correct Learning Model

Removing Some Assumptions

- $D(y \mid x) =$ Conditional Probability of getting label y given x
- Notation used in *"Understanding Machine Learning"*

$$D(y \mid x) \equiv D\Big((x, y) \Bigm| x\Big)$$

The Probability Approximately Correct Learning Model

Removing Some Assumptions

- $D(y \mid x) =$ Conditional Probability of getting label y given x
- Notation used in *"Understanding Machine Learning"*

$$D(y \mid x) \equiv D\Big((x, y) \,\Big|\, x\Big)$$

- By using a probability distribution for y,

Removing Some Assumptions

- $D(y \mid x) =$ Conditional Probability of getting label y given x
- Notation used in *"Understanding Machine Learning"*

$$D(y \mid x) \equiv D\Big((x, y) \,\Big|\, x\Big)$$

- By using a probability distribution for y,
- Allows two different mangoes with identical x's to have different y's because the feature set doesn't fully parameterize the set of mangoes.

Empirical And True Error Revised

- We want to measure how likely h will mislabel points.

Empirical And True Error Revised

- We want to measure how likely h will mislabel points.
- True error of Prediction Rule h

$$L_D(h) \equiv \underset{x \sim D}{\mathbb{P}}\Big(h(x) \neq y\Big) \equiv D(x, y \mid h(x) \neq y)$$

Empirical And True Error Revised

- We want to measure how likely h will mislabel points.
- True error of Prediction Rule h

$$L_D(h) \equiv \underset{x \sim D}{\mathbb{P}}\Big(h(x) \neq y\Big) \equiv D(x, y \mid h(x) \neq y)$$

- Note, because we are not assuming realizability, the true error is labeled $L_D(h)$ not $L_{D,f}(h)$.

Empirical And True Error Revised

- We want to measure how likely h will mislabel points.
- True error of Prediction Rule h

$$L_D(h) \equiv \underset{x \sim D}{\mathbb{P}}\Big(h(x) \neq y\Big) \equiv D(x, y \mid h(x) \neq y)$$

- Note, because we are not assuming realizability, the true error is labeled $L_D(h)$ not $L_{D,f}(h)$.
- Empirical Risk

$$L_S(h) \equiv \frac{|\{i \in [m] \mid h(x_i \neq y_i)\}|}{m}$$

Empirical And True Error Revised

- We want to measure how likely h will mislabel points.
- True error of Prediction Rule h

$$L_D(h) \equiv \underset{x\sim D}{\mathbb{P}}\Big(h(x) \neq y\Big) \equiv D(x, y \mid h(x) \neq y)$$

- Note, because we are not assuming realizability, the true error is labeled $L_D(h)$ not $L_{D,f}(h)$.
- Empirical Risk

$$L_S(h) \equiv \frac{|\{i \in [m] \mid h(x_i \neq y_i)\}|}{m}$$

- Goal: Want an h minimizing $L_D(h)$ that is *PAC*

Bayes' Optimal Predictor

- For a given D over $X \times \{0,1\}$, the best labeling function $f : X \to \{0,1\}$ is

$$f_D(x) = \begin{cases} 1 & \text{if } \mathbb{P}(y = 1 \mid x) \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

Bayes' Optimal Predictor

- For a given D over $X \times \{0,1\}$, the best labeling function $f : X \to \{0,1\}$ is

$$f_D(x) = \begin{cases} 1 & \text{if } \mathbb{P}(y = 1 \mid x) \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

- Note, $f_D(x)$ is not the *true* labeling function f since we don't know what the true f is

Bayes' Optimal Predictor

- For a given D over $X \times \{0, 1\}$, the best labeling function $f : X \to \{0, 1\}$ is

$$f_D(x) = \begin{cases} 1 & \text{if } \mathbb{P}(y = 1 \mid x) \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

- Note, $f_D(x)$ is not the *true* labeling function f since we don't know what the true f is

- We can show that $\forall\ D$, the *Bayes' Optimal Predictor* f_D, is optimal.

The Probability Approximately Correct Learning Model

Bayes' Optimal Predictor

- For a given D over $X \times \{0,1\}$, the best labeling function $f : X \to \{0,1\}$ is

$$f_D(x) = \begin{cases} 1 & \text{if } \mathbb{P}(y = 1 \mid x) \geq \frac{1}{2} \\ 0 & \text{otherwise} \end{cases}$$

- Note, $f_D(x)$ is not the *true* labeling function f since we don't know what the true f is
- We can show that $\forall\ D$, the *Bayes' Optimal Predictor* f_D, is optimal.
- No other classifier $g : X \to \{0,1\}$ has lower error $L_D(f_D) \leq L_D(g)$

Bayes' Optimal Predictor

- Don't know D, thus can't utilize f_D

The Probability Approximately Correct Learning Model

Bayes' Optimal Predictor

- Don't know D, thus can't utilize f_D
- No algorithm can be guaranteed to find a predictor as good as f_D

Bayes' Optimal Predictor

- Don't know D, thus can't utilize f_D
- No algorithm can be guaranteed to find a predictor as good as f_D
- Seek a *predictor* as close to f_D as possible

Bayes' Optimal Predictor

- Don't know D, thus can't utilize f_D
- No algorithm can be guaranteed to find a predictor as good as f_D
- Seek a *predictor* as close to f_D as possible
- Ability to do this depends on the hypothesis class of h

What is Agnostic PAC Learnability?

Definition: Agnostic PAC Learnability

A hypothesis class h is *agnostic PAC learnable* if

- $\exists\ m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and $\exists$ a learning algorithm such that

Agnostic PAC Learnability

What is Agnostic PAC Learnability?

Definition: Agnostic PAC Learnability

A hypothesis class h is *agnostic PAC learnable* if

- $\exists\ m_{\mathcal{H}} : (0,1)^2 \rightarrow \mathbb{N}$ and $\exists$ a learning algorithm such that
- $\forall\ \epsilon, \delta \in (0,1)$. and $\forall\ D$ over $X \times Y$, when running the algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ iid samples generated by D

Agnostic PAC Learnability

What is Agnostic PAC Learnability?

Definition: Agnostic PAC Learnability

A hypothesis class h is *agnostic PAC learnable* if

- $\exists\ m_{\mathcal{H}} : (0,1)^2 \rightarrow \mathbb{N}$ and $\exists$ a learning algorithm such that
- $\forall\ \epsilon, \delta \in (0,1)$. and $\forall\ D$ over $X \times Y$, when running the algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ iid samples generated by D
- the algorithm return a hypothesis h,

Agnostic PAC Learnability

What is Agnostic PAC Learnability?

Definition: Agnostic PAC Learnability

A hypothesis class h is *agnostic PAC learnable* if

- $\exists\ m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and $\exists$ a learning algorithm such that
- $\forall\ \epsilon, \delta \in (0,1)$. and $\forall\ D$ over $X \times Y$, when running the algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ iid samples generated by D
- the algorithm return a hypothesis h,
- such that with probability of at least $1 - \delta$ over the training samples

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

Agnostic PAC Learnability

- If realizability holds, agnostic PAC learnability provides the same guarantee as PAC learning

Agnostic PAC Learnability

- If realizability holds, agnostic PAC learnability provides the same guarantee as PAC learning

$\Rightarrow$ Agnostic PAC Learning generalizes PAC learning

Agnostic PAC Learnability

- If realizability holds, agnostic PAC learnability provides the same guarantee as PAC learning

$\Rightarrow$ Agnostic PAC Learning generalizes PAC learning

- When *realizability* doesn't hold,

Agnostic PAC Learnability

- If realizability holds, agnostic PAC learnability provides the same guarantee as PAC learning

$\Rightarrow$ Agnostic PAC Learning generalizes PAC learning

- When *realizability* doesn't hold,

$\Rightarrow$ Can't guarantee arbitrarily small error

Agnostic PAC Learnability

- If realizability holds, agnostic PAC learnability provides the same guarantee as PAC learning

$\Rightarrow$ Agnostic PAC Learning generalizes PAC learning

- When *realizability* doesn't hold,

$\Rightarrow$ Can't guarantee arbitrarily small error

- But using *agnostic PAC learning*

Agnostic PAC Learnability

- If realizability holds, agnostic PAC learnability provides the same guarantee as PAC learning

$\Rightarrow$ Agnostic PAC Learning generalizes PAC learning

- When *realizability* doesn't hold,

$\Rightarrow$ Can't guarantee arbitrarily small error

- But using *agnostic PAC learning*

$\Rightarrow$ Can get smallest error predictor in the class $\mathcal{H}$

Agnostic PAC Learnability

Scope of Learning Problems Modeled

- *Multiclass Classification:*

$$h: \ \mathit{Documents} \begin{array}{l} \nearrow \mathit{sports} \\ \rightarrow \mathit{news} \\ \searrow \mathit{entertainment} \end{array}$$

error: Probability of misclassifying a document.

Scope of Learning Problems Modeled

- *Multiclass Classification:*

$$h: \ \textit{Documents} \begin{array}{l} \nearrow \textit{sports} \\ \rightarrow \textit{news} \\ \searrow \textit{entertainment} \end{array}$$

  error: Probability of misclassifying a document.

- **Regression:** Looking for simple patterns

$$h: \ \text{ultrasound measurements} \ \rightarrow \ \text{baby's weight}$$

  error = *Expected difference between true labels and predicted labels*

$$L_D(h) \equiv \mathop{\mathbb{E}}_{(x,y)\sim D} [(h(x) - y)^2]$$

Scope of Learning Problems Modeled

Generalized Measure of Success

- Given $\mathcal{H}$, domain $Z = X \times Y$,

Scope of Learning Problems Modeled

Generalized Measure of Success

- Given $\mathcal{H}$, domain $Z = X \times Y$,
- Let $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$, where

$$\ell = \text{loss function}$$
$$\mathbb{R}_+ = \text{set of non negative real numbers}$$

Scope of Learning Problems Modeled

Generalized Measure of Success

- Given $\mathcal{H}$, domain $Z = X \times Y$,
- Let $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$, where

$$\ell = \text{loss function}$$
$$\mathbb{R}_+ = \text{set of non negative real numbers}$$

- In unsupervised problems, Z is not $X \times Y$, since there is *"no"* Y

Scope of Learning Problems Modeled

Generalized Measure of Success

- Given $\mathcal{H}$, domain $Z = X \times Y$,
- Let $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$, where

$$\ell = \text{loss function}$$
$$\mathbb{R}_+ = \text{set of non negative real numbers}$$

- In unsupervised problems, Z is not $X \times Y$, since there is *"no"* Y
- **Definition:** *Risk function* of $h \in \mathcal{H}$ wrt D over Z

$$L_D(h) \equiv \mathop{\mathbb{E}}_{Z \sim D} [\ell(h, z)]$$

This is the expected loss of h sampled from Z using D.

Scope of Learning Problems Modeled

Definition: Empirical Risk

Given $S = \{z_1, z_2, \ldots, z_m\} \in Z^m$

$$L_S(h) \equiv \frac{1}{m} \sum_{i=1}^{m} \ell(h, z_i)$$

Loss Functions

- **0-1 Loss:** $Z \in X \times Y$

$$\ell_{01}\Big(h,(x,y)\Big) \equiv \begin{cases} 0 \text{ if } h(x) = y \\ 1 \text{ if } h(x) \neq y \end{cases}$$

Scope of Learning Problems Modeled

Loss Functions

- **0-1 Loss:** $Z \in X \times Y$

$$\ell_{01}\Big(h, (x, y)\Big) \equiv \begin{cases} 0 \text{ if } h(x) = y \\ 1 \text{ if } h(x) \neq y \end{cases}$$

$\Rightarrow$ Binary/multiclass classification problem

Scope of Learning Problems Modeled

Loss Functions

- **0-1 Loss:** $Z \in X \times Y$

$$\ell_{01}\Big(h, (x, y)\Big) \equiv \begin{cases} 0 \text{ if } h(x) = y \\ 1 \text{ if } h(x) \neq y \end{cases}$$

$\Rightarrow$ Binary/multiclass classification problem

$\Rightarrow$ For random variable $\alpha \in \{0, 1\}$

$$\mathop{\mathbb{E}}_{\alpha \sim D}[\alpha] = \mathop{\mathbb{P}}_{\alpha \sim D}(\alpha = 1)$$

Scope of Learning Problems Modeled

Loss Functions

$\Rightarrow$ Then the different definitions of $L_D(h)$ coincide

$$\text{a)} \quad L_D(h) \equiv \mathop{\mathbb{E}}_{Z\sim D}(\ell(h,z))$$

$$\text{b)} \quad L_D(h) \equiv \mathop{\mathbb{P}}_{z\sim D}\Big(h(x) \neq y\Big)$$

Loss Functions

$\Rightarrow$ Then the different definitions of $L_D(h)$ coincide

$$\text{a)}\quad L_D(h) \equiv \underset{Z\sim D}{\mathbb{E}}(\ell(h,z))$$

$$\text{b)}\quad L_D(h) \equiv \underset{z\sim D}{\mathbb{P}}\Big(h(x) \neq y\Big)$$

- **Square Loss:** $Z \in X \times Y$

$$\ell_{01}\big(h(x,y)\big) \equiv \big(h(x) - y\big)^2$$

Scope of Learning Problems Modeled

Agnostic PAC Learnability For General Loss Functions

Definition: *Same as before with $\mathcal{H}, Z, D, \ell$ where $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and*

$$L_D(h) = \mathop{\mathbb{E}}_{Z \sim D}[\ell(h, z)]$$

Sidenote: Proper vs Representational Independent Learning

- $\mathcal{H} \in \mathcal{H}'$

Scope of Learning Problems Modeled

Agnostic PAC Learnability For General Loss Functions

Definition: *Same as before with $\mathcal{H}, Z, D, \ell$ where $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and*

$$L_D(h) = \mathop{\mathbb{E}}_{Z \sim D}[\ell(h, z)]$$

Sidenote: Proper vs Representational Independent Learning

- $\mathcal{H} \in \mathcal{H}'$
- Extend *Loss Function* to $\mathcal{H}' \times Z \to \mathbb{R}_+$

Scope of Learning Problems Modeled

Agnostic PAC Learnability For General Loss Functions

Definition: *Same as before with $\mathcal{H}, Z, D, \ell$ where $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and*

$$L_D(h) = \mathop{\mathbb{E}}_{Z \sim D}[\ell(h, z)]$$

Sidenote: Proper vs Representational Independent Learning

- $\mathcal{H} \in \mathcal{H}'$
- Extend *Loss Function* to $\mathcal{H}' \times Z \to \mathbb{R}_+$
- Return algorithm from $\mathcal{H}'$ instead of $\mathcal{H}$ *(This is representational independent learning)* as long as it satisfies

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

Scope of Learning Problems Modeled

Agnostic PAC Learnability For General Loss Functions

Definition: *Same as before with $\mathcal{H}, Z, D, \ell$ where $\ell : \mathcal{H} \times Z \to \mathbb{R}_+$ and*

$$L_D(h) = \mathop{\mathbb{E}}_{Z \sim D}[\ell(h, z)]$$

Sidenote: Proper vs Representational Independent Learning

- $\mathcal{H} \in \mathcal{H}'$
- Extend *Loss Function* to $\mathcal{H}' \times Z \to \mathbb{R}_+$
- Return algorithm from $\mathcal{H}'$ instead of $\mathcal{H}$ *(This is representational independent learning)* as long as it satisfies

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

- Proper learning is when the algorithm outputs an h from $\mathcal{H}$ not from $\mathcal{H}'$