Introduction to Machine Learning

Foundations and Applications

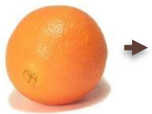
Paul J. Atzberger
University of California Santa Barbara

Statistical Learning Theory
PAC-Learning
Generalization Bounds

Modeling Learning in Supervised Case

Apple Images   Orange Images

Item



Features

Feature	Value
Roundness	0.8
Sweetness	0.9
Redness	0.1
Greenness	0.3

$\boldsymbol{x} = (0.8, 0.9, 0.1, 0.3)$

Example Task

Classify a collection of images as Apples or Oranges.

Domain $\mathcal{X}$: Learning involves identifying a collection of *features* of an object which we encode in a space $\mathcal{X}$.

Labels $\mathcal{Y}$: Each object has a class label, here $\mathcal{Y} = \{1, -1\}$ with 1: Apple and -1 Orange.

Training Data $\mathcal{S}$: We are given finite number of examples $\mathcal{S} = \{(x_1, y_1), \ldots (x_m, y_m)\}$ from which to try to learn a model $h = A(\mathcal{S})$ to classify previously unseen objects x as $y = h(x)$.

Learning Output $h = A(\mathcal{S})$: The learning algorithm A produces a *prediction rule* $h = A(\mathcal{S})$ with $h : \mathcal{X} \to \mathcal{Y}$. The h also referred to as a *predictor*, *hypothesis*, *classifier*.

Data Generation Process $\mathcal{D}$: The sample $\mathcal{S}$ we see in practice comes from some generating process (i.e. users posting photos online). We model this as a probability distribution $\mathcal{D}$ over $\mathcal{X}$. We also assume for the labels y there is some "correct rule" $f : \mathcal{X} \to \mathcal{Y}$, which we use for labels $y = f(x)$, $\; x \sim \mathcal{D}$.

Measuring Level of Success $L_{\mathcal{D},f}$: The *loss function* $L_{\mathcal{D},f}(h)$ measures the accuracy of h in assigning the correct labels. Here, $L_{\mathcal{D},f}(h) = \Pr_{x\sim\mathcal{D}}\{h(x) \neq f(x)\}$. Also, referred to as *generalization error*, *true risk*.

Statistical Learning Theory

"There is nothing more practical than a good theory."
-- James C. Maxwell.

Framework for characterizing learning problems and algorithms.

Goal: Assess how well a model predicts future input-output relations.

Leslie Valiant

Mathematical Definitions: Consider c: $X \to Y$, X-input, Y-output.
Let c = concept, $\mathcal{C}$ = {concept class}, $\mathcal{H}$ = {hypothesis function space},
$D_{x,y} \sim X \times Y$ be unknown probability distribution on $X \times Y$, and
$V(h(x_j), y_j) = L_{D,c}(h)$ = loss function.

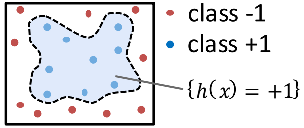
Vladimir Vapnik

Alexey Chervonenkis

Learning Problem: Find the best $h \in \mathcal{H}$ so that $E_D[V(\mathrm{h}(x), y)]$ is minimized when $\mathrm{c} \in C$, $\mathrm{y} = c(x)$.

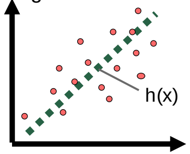
Loss Functions: common examples:
Classification: $V(\mathrm{h}(x), y) = I_{\mathrm{h}(x) \neq y}$, (zero-one loss).
Regression: $V(\mathrm{h}(x), y) = (\mathrm{h}(x) - y)^2$, (least-squares L^2 -loss).

Classification



Important to learning, the choice of hypothesis class $\mathcal{H}$ and loss used!

Regression



Practical Challenges: Distribution D usually unknown, optimization is often non-convex and in high-dimensional spaces and approximate.

Statistical Learning Theory

Notation and definitions:

$\mathcal{X}$ input space
$\mathcal{Y}$ output space
c(x): $\mathcal{X} \rightarrow \mathcal{Y}$ concept
$\mathcal{C}$ concept class
$\mathcal{H}$ hypothesis class

We receive samples $\mathcal{S} = (x_1, x_2, \ldots, x_m)$ and labels $\mathcal{T} = (y_1, y_2, \ldots y_m)$, where $y_i = c(x_i)$.

Task: Determine from $\mathcal{S}$ and $\mathcal{T}$ a hypothesis function $h_S \in \mathcal{H}$

Goal: We want $h_S(x)$ that

(i) fits to explain the training data $\mathcal{S}, \mathcal{T}$ well.
(ii) generalizes to give correct results for new unseen data points drawn from $D_{\mathcal{X}}$.

Definition: The **generalization error (risk)** for 0-1 classification $\mathcal{Y}=\{0,1\}$ is

$$R(h) = \Pr\{h_S(x) \neq c(x)\} = E_{x\sim D}\left[1_{h_S(x)\neq c(x)}\right]$$

However, in practice this is NOT directly computable since we do not know c(x) and D.

Statistical Learning Theory

Notation and definitions:

$\mathcal{X}$ input space
$\mathcal{Y}$ output space
c(x): $\mathcal{X} \rightarrow \mathcal{Y}$ concept
$\mathcal{C}$ concept class
$\mathcal{H}$ hypothesis class

We receive samples S = $(x_1,x_2,\ldots,x_m)$ and labels T = $(y_1=c(x_1),\ y_2=c(x_2),\ \ldots\ y_m=c(x_m))$.

Definition: The **empirical generalization error (empirical risk)** for 0-1 classification $\mathcal{Y}=\{0,1\}$ is

$$\hat{R}(h) = \frac{1}{m}\sum_i 1_{h_S(x_i)\neq c(x_i)}$$

This gives an unbiased estimator of the generalization error (true risk).

Lemma: $E_{\mathbf{x}\sim D^m}\left[\hat{R}(h)\right] = R(h)$

Proof:

$$E_{\mathbf{x}\sim D^m}\left[\hat{R}(h)\right] = \frac{1}{m}\sum_{i=1}^{m} E_{\mathbf{x}\sim D^m}\left[1_{h(x_i)\neq c(x_i)}\right] = \frac{1}{m}\sum_{i=1}^{m}\Pr\{h(x)\neq c(x)\} = \frac{1}{m}\sum_{i=1}^{m} R(h) = R(h). \quad \blacksquare$$

PAC-Learning

Probability Approximately Correct (PAC) Learning Framework.
Introduced by *Leslie Valiant* in 1984 to assess computational complexity of learning tasks.

Leslie Valiant

PAC-learning

We say a concept class $\mathcal{C}$ is **PAC-learnable** if there exists an algorithm $\mathcal{A}$ and polynomial bound so that given ε > 0 and δ > 0, the following holds for any distribution $D \in \mathfrak{D}$ on $\mathcal{X}$, target concept c in $\mathcal{C}$, and sample size m ≥ $m_{\mathcal{H}}(\epsilon, \delta)$, with $m_{\mathcal{H}} = O$(poly(1/ε,1/δ,n,size(c))).

$$\Pr\{R(h_S) \leq \epsilon\} \geq 1 - \delta$$

$m_{\mathcal{H}}(\epsilon, \delta)$ is the **sampling complexity.**

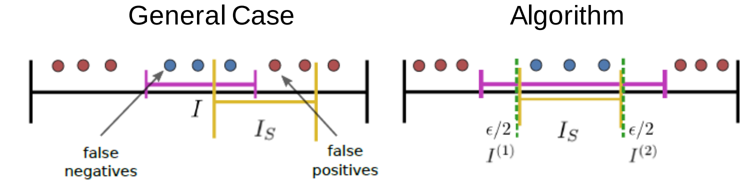
Efficient PAC-learnable

We say a problem is **efficiently PAC-learnable** if the algorithm $\mathcal{A}$ runs in at most a time τ = poly(1/ε,1/δ,n,size(x)).

We call $\mathcal{A}$ the PAC-learning algorithm for $\mathcal{C}$.

PAC-Learning

Example: Learning intervals on $\mathbb{R}$−line.



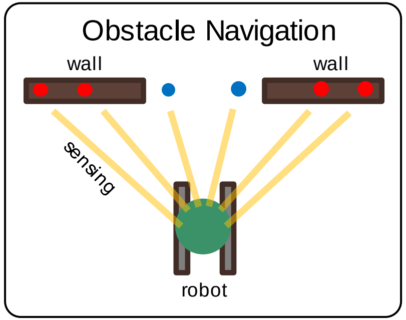
Algorithm:

$\mathcal{A}$: I_S = {smallest interval of in-class}.

$I_S = [a, b]$

$a = \min\{x_i | y_i = 1\}$,

$b = \max\{x_i | y_i = 1\}$.



We need to show: Given $\epsilon > 0$, $\delta > 0$ there exists a polynomial bound in samples m with $(\mathcal{S}, \mathcal{T}) = \{(x_i, y_i)\}_{i=1}^m, x_i \in \mathbb{R}, y_i \in \{0, 1\}$

$$\Pr_{\mathbf{x} \sim D^m} \{R(I_S) \leq \epsilon\} \geq 1 - \delta.$$

Since $I_S \subset I$, we only need to worry about false negatives. This has

$$R(I_S) = \Pr_{\mathbf{x} \sim D^m} \{x \notin I_S \cap x \in I\} = E_{x \sim D}\left[1_{h_S(x) \neq c(x)}\right].$$

We use that if $\mathcal{A} \Rightarrow \mathcal{B}$ then $\Pr\{\mathcal{A}\} \leq \Pr\{\mathcal{B}\}$ and we use $1 - x \leq \exp[-x]$.

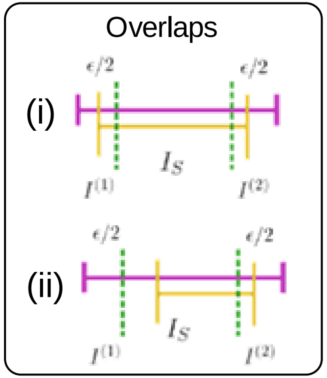
If $I_S \cap I^{(i)} \neq \emptyset, \forall i = 1, 2$ then $R(I_S) \leq \epsilon$. By contrapositive $R(I_S) > \epsilon \Rightarrow \exists i$ s.t $I_S \cap I^{(i)} = \emptyset.$

This gives the bound

$$\Pr\{R(I_S) > \epsilon\} \leq \Pr\{\cup_{i=1}^2 I_S \cap I^{(i)} = \emptyset\} \leq \sum_{i=1}^2 \Pr\{I_S \cap I^{(i)} = \emptyset\} \leq 2\,(1 - \epsilon/2)^m \leq 2 \exp\left[-\frac{\epsilon m}{2}\right] < \delta$$

$$\Rightarrow m > \frac{2}{\epsilon} \ln\left(\frac{2}{\delta}\right). \quad \blacksquare$$

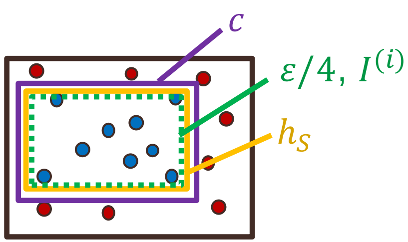
Shows is efficient PAC-learnable.



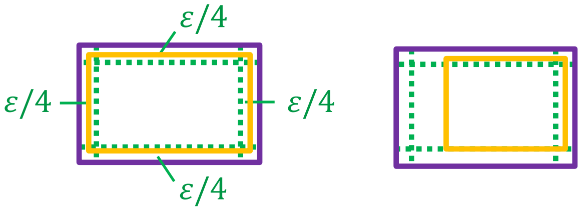
PAC-Learning

Example: Learning axis-aligned rectangles.

Algorithm



(i) Overlaps All (ii) Not Full Overlap



$R(h_S) \leq \epsilon$ $\exists i,\ h_s \cap \mathcal{I}^{(i)} = \emptyset$

We need to show

$$\Pr\{R(h_S) \leq \epsilon\} \geq 1 - \delta$$

This implies

$$R(h_S) > \epsilon \ \Rightarrow \exists i \text{ s.t. } h_S \cap I^{(i)} = \emptyset$$

$$\Pr_{x\sim D^m}\{R(h_S) > \epsilon\} \leq \Pr_{x\sim D^m}\left\{\bigcup_{i=1}^{4}\left\{h_S \cap I^{(i)} = \emptyset\right\}\right\} \leq \sum_{i=1}^{4} \Pr_{x\sim D^m}\left\{h_S \cap I^{(i)} = \emptyset\right\}$$

$$\leq 4\,(1 - \epsilon/4)^m \leq 4\exp[-\epsilon m/4] < \delta.$$

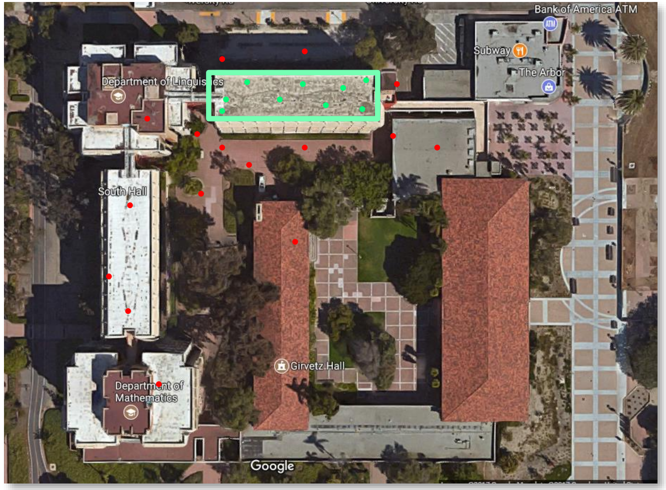
Bound on samples m

$$-\epsilon m/4 < \ln\left(\frac{\delta}{4}\right) \Rightarrow \boxed{m > \frac{4}{\epsilon}\ln\left(\frac{4}{\delta}\right)}$$

Bound on risk R

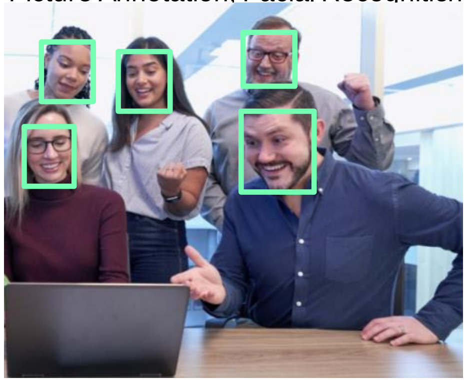
$$\epsilon = \frac{4}{m}\ln\left(\frac{4}{\delta}\right) \Rightarrow \Pr = 1 - \delta, \quad \boxed{R(h_S) \leq \frac{4}{m}\ln\left(\frac{4}{\delta}\right)}$$

Building Identification



Google Maps: UCSB South Hall

Picture Annotation, Facial Recognition



usplash

Data Sampling Complexity

Guarantees on Sampling Complexity $m_{\mathcal{H}}(\epsilon, \delta)$

How many samples do we need to **guarantee** a given **level of precision** ϵ, δ in PAC-learning?

What is bound M = $m_{\mathcal{H}}(\epsilon, \delta)$ so for $m \geq M$ we have $\Pr\{R(h_S) \leq \epsilon\} \geq 1 - \delta$?

This will depend on the hypothesis space $\mathcal{H}$ and concept class $\mathcal{C}$.

Two important cases:

(i) **consistent case:** $\mathcal{C} \subset \mathcal{H}$, hypotheses include all concepts.
(ii) **inconsistent case**: $\mathcal{C} \not\subset \mathcal{H}$, hypotheses can not capture all concepts.

Distinguish also case of finite vs infinite hypothesis spaces $\mathcal{H}$ and concept spaces $\mathcal{C}$.

Theorem: Consistent-Finite Hypothesis Spaces $\mathcal{H}$. Let $\mathcal{A}$ be any learning algorithm that has zero Empirical Generalization Error $\hat{R}(h_S) = 0$ then PAC-learning bound $\Pr\{R(h_S) \leq \epsilon\} \geq 1 - \delta$ is guaranteed to hold for m samples satisfying

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

Empirical Generalization Error

$$\hat{R}(h) = \frac{1}{m}\sum_{i=1}^{m} 1_{h_S(x_i) \neq c(x_i)}$$

Data Sampling Complexity

Finite Consistent-Case: Guarantees on Sampling Complexity

Theorem: Consistent-Finite Hypothesis Spaces $\mathcal{H}$. Let $\mathcal{A}$ be any learning algorithm that has zero empirical generalization error $\hat{R}(h_S) = 0$ then PAC-learning bound $\Pr\{R(h_S) \leq \epsilon\} \geq 1 - \delta$ is guaranteed to hold for m samples satisfying

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

Proof: Let $\mathcal{A}$ be any algorithm that returns for m samples $\mathcal{S}$ a hypothesis h_S s.t. $\hat{R}(h_S) = 0$.

$$\begin{aligned}
\Pr_{\mathcal{S}\sim D^m}\{h \in \mathcal{H} \wedge \hat{R}(h) = 0 \wedge R(h) > \epsilon\} &= \Pr_{\mathcal{S}\sim D^m}\{h_1 \in \mathcal{H} \wedge \hat{R}(h_1) = 0 \wedge R(h_1) > \epsilon \vee \cdots \vee h_{|\mathcal{H}|} \in \mathcal{H} \wedge \hat{R}(h_{|\mathcal{H}|}) = 0 \wedge R(h_{|\mathcal{H}|}) > \epsilon\} \\
&\leq \sum_{i=1}^{|H|} \Pr\{h_i \in \mathcal{H} \wedge \hat{R}(h_i) = 0 \wedge R(h_i) > \epsilon\} \\
&\leq \sum_{i=1}^{|H|} \Pr\{h_i \in \mathcal{H} \wedge \hat{R}(h_i) = 0 | R(h_i) > \epsilon\} \\
&\leq |H|(1-\epsilon)^m \leq |H|\exp(-\epsilon m) \leq \delta \\
&\Rightarrow \log(|H|) - \epsilon m \leq \log(\delta) \\
&\Rightarrow m \geq \frac{1}{\epsilon}\left(\log(|H|) + \log\left(\frac{1}{\delta}\right)\right) \quad \blacksquare
\end{aligned}$$

We use that

$$\Pr\{A \wedge B \wedge C\} = \Pr\{A \wedge B | C\}\Pr\{C\} \leq \Pr\{A \wedge B\}$$

$$1 - x \leq e^{-x}$$

Generalization Bounds

Finite-Consistent Case: Guarantees on Sampling Complexity

Corollary: Consistent-Finite Hypothesis Spaces $\mathcal{H}$. Let $\mathcal{A}$ be any learning algorithm that has zero empirical generalization error $\hat{R}(h_S) = 0$ then the generalization error is bounded by

$$R(h_S) \leq \frac{1}{m}\left(\log|H| + \log\frac{1}{\delta}\right)$$

Proof: Follows setting $\epsilon = \frac{1}{m}\left(\log\left(|H|\right) + \log\left(\frac{1}{\delta}\right)\right) \longrightarrow m \geq \frac{1}{\epsilon}\left(\log\left(|H|\right) + \log\left(\frac{1}{\delta}\right)\right) \longrightarrow R\left(h_S\right) \leq \epsilon$ ■

Consistent-Finite Hypothesis Case

- **1/m – error decay rate** is in fact very good relative to other cases we shall investigate.
- **Sample complexity bounds** are logarithmic in the hypothesis space size $|\mathcal{H}|$.
- $\log(|\mathcal{H}|)$ ~ number of bits needed to distinguish a hypothesis function.
- This indicates **smaller hypothesis** space → **easier to learn concepts**.
- However, **consistency** $\mathcal{C} \subset \mathcal{H}$ requires **"big enough" hypothesis space** $\mathcal{H}$ to capture target concepts.

Data Sampling Complexity

Example: Boolean Conjunctions.
Let z_i be Boolean variable, a conjunction is: $c = \overline{z}_1 \wedge z_2 \wedge z_5 \wedge z_6$.

Learning algorithm $\mathcal{A}$: Use only the positive examples.
- if $z_i = 1$ then include z_i.
- if $z_i = 0$ then include $\overline{z}_i$.

The concept class $|\mathcal{C}_n| = 3^n$, since in n-conjunction either z_i, $\overline{z}_i$, or ϕ.

Note could learn directly with as few as 2n examples if special ones chosen.

Let $\mathcal{H} = \mathcal{C}_n$ then we have consistent-finite hypothesis space and $\hat{R}(h_S) = 0$.
Sample complexity:

$$m \geq \frac{1}{\epsilon}\left(n \log(3) + \log\left(\frac{1}{\delta}\right)\right)$$

This shows $\mathcal{C}_n$ is **PAC-learnable**.

Note statistical learning might not be as efficient as direct methods when available.

0	1	1	0	1	1	+
0	1	1	1	1	1	+
0	0	1	1	0	1	-
0	1	1	1	1	1	+
1	0	0	1	1	0	-
0	1	0	0	1	1	+
0	1	?	?	1	1	

Mohri 2012

Example 2:

$$C = z_1 \wedge \bar{z}_2 \wedge z_3$$

$z_1 \sim$ "it is raining"
$z_2 \sim$ "have umbrella"
$z_3 \sim$ "getting wet"

Confidence desired:
$\epsilon = 0.01 \rightarrow 99\%$
$\delta = 0.05 \rightarrow 95\%$

Bound on number samples:
$m \geq 630$
(larger than direct testing 2n)

Data Sampling Complexity

Example: Universality Class $\mathcal{U}_n = \{c: \{0,1\}^n \rightarrow \{0,1\}\}$. All functions $c(z_1,z_2,\ldots,z_n) \rightarrow \{0,1\}$.

A consistent-finite hypothesis class $\mathcal{H}$ must contain $\mathcal{U}_n$ giving $|\mathcal{H}| \geq |\mathcal{U}_n| = 2^{2^n}$.

This suggests a sample complexity (if bounds tight) of

$$m \geq \frac{1}{\epsilon}\left(2^n \log(2) + \log\left(\frac{1}{\delta}\right)\right)$$

This suggests learning problem requires exponential number of samples in the input size n.

Not hard to show this concept class is in fact **not PAC-Learnable.**

Efficient learnability requires our concept class not be too broad.

Task specific mathematical structure needed to develop efficient algorithms for representing concepts and distinguishing hypotheses.

Completely generic functions can not be learned efficiently (too many possibilities / complexity).

Agnostic PAC-Learning

Inconsistent case when $\mathcal{C} \notin \mathcal{H}$.
For all h we may have $R(h) \neq 0$. Our aim is to achieve as small a generalization error as possible.

Agnostic PAC-Learning:
We say a concept class $\mathcal{C}$ is **Agnostic PAC-Learnable** if there exists an algorithm $\mathcal{A}$ and polynomial bound so that given ε > 0 and δ > 0, the following holds for any distribution D on $\mathcal{X}$ x $\mathcal{Y}$, target concept c in $\mathcal{C}$, and sample size m ≥ poly(1/ε,1/δ,n,size(x))

$$\Pr\{R(h_S) - \min_{h \in \mathcal{H}} R(h) \leq \epsilon\} \geq 1 - \delta$$

Note, **generalization error** is now $R(h) = \Pr_{(x,y)\sim D}[h(x) \neq y] = \mathop{\mathrm{E}}_{(x,y)\sim D}[1_{h(x)\neq y}]$.

If computational complexity of algorithm is poly(1/ε,1/δ,n,size(x)) we say the concept class is **Efficiently Agnostic PAC-Learnable**.

Stochastic vs Deterministic Learning: Above applies also when label **y** for feature vector **x** is not unique, as in many real-world data sets. Uncertainty captured by D ~ $\mathcal{X}$ x $\mathcal{Y}$, allowing for a type of stochastic learning.

$$h(x) = \left\{ \begin{array}{ll} 1, & \text{if } \Pr\{h(x) = y\} \geq 1/2 \\ 0, & \text{otherwise} \end{array} \right\}$$

Goal: Find best assignment y = h(x) minimizing generalization error (i.e. 0-1, Bayes classifier).

Generalization Bounds

Finite-Inconsistent Case: Guarantees on Sampling Complexity

Theorem: Inconsistent-Finite Hypothesis Spaces $\mathcal{H}$. Let $\mathcal{A}$ be any learning algorithm that has empirical generalization error $\hat{R}(h_S)$ then for any h ∈ $\mathcal{H}$ we have

$$R(h) \leq \widehat{R}(h) + \sqrt{\frac{\log|H| + \log\frac{2}{\delta}}{2m}} \quad \blacksquare$$

This shows training error is indicative of the generalization error with enough samples

$$\left|\hat{R}(h) - R(h)\right| \leq \sqrt{\frac{\log(|H|) + \log(\frac{2}{\delta})}{2m}}$$

This means if we have small training set error $\hat{R}(h_S)$ then "with enough" samples we can obtain small gap in generalization errors.

For **Agnostic PAC-Learnable** concepts we have $\Pr\{R(h_S) - \min_{h\in\mathcal{H}} R(h) \leq \epsilon\} \geq 1-\delta$

These results show even in the inconsistent case for enough samples m a small training set error is still indicative for obtaining an hypothesis h with best generalization error.

Note, only $\mathrm{m}^{-1/2}$ scaling in the bound (compare to the finite-consistent case ~ m^{-1}).

Concentration Inequalities

Probability Theory and Inequalities

Concentration Inequalities

Lemma: Markov Inequality $\Pr[X \geq \epsilon] = \Pr[e^{tX} \geq e^{t\epsilon}] \leq e^{-t\epsilon}\,\mathrm{E}[e^{tX}]$ for $t \geq 0$.

Proof: $\Pr\{e^{tx} \geq e^{t\epsilon}\} \leq \int_{\Omega} \mathbf{1}_{e^{tx} \geq e^{t\epsilon}}(x)\, d\mathcal{D}_x \leq \int e^{-t\epsilon} e^{tx}\; d\mathcal{D}_x = e^{-t\epsilon}\mathbb{E}\left[e^{tX}\right]$ ■

Lemma: (Hoeffding's Lemma) Let X be a random variable with $E[X] = 0$, $a \leq X \leq b$, and $b > a$, then we have the bound

$$\mathrm{E}[e^{tX}] \leq e^{\frac{t^2(b-a)^2}{8}}$$

Proof: We have that $e^{tx} \leq \frac{b-x}{b-a}e^{ta} + \frac{x-a}{b-a}e^{tb}$ using $a \leq x \leq b$, $x \to e^{tx}$ is a convex function.

From $\mathbb{E}[X] = 0$, we have $\mathbb{E}\left[e^{tX}\right] \leq \frac{b}{b-a}e^{ta} + \frac{-a}{b-a}e^{tb} = e^{\phi(t)} \longleftarrow$ $\phi(t) = \log\left(\frac{b}{b-a}e^{ta} - \frac{a}{b-a}e^{tb}\right)$

$$= \log\left(e^{ta}\left(\frac{b}{b-a} - \frac{a}{b-a}e^{t(b-a)}\right)\right)$$

$$= ta + \log\left(\frac{b}{b-a} - \frac{a}{b-a}e^{t(b-a)}\right)$$

For any t > 0, we have for $\phi'(t), \phi''(t)$

$$\phi'(t) = a - \frac{a}{b/(b-a)e^{-t(b-a)} - a/(b-a)},\quad \phi''(t) = u(1-u)(b-a)^2$$

$$u = \alpha/\left((1-\alpha)e^{-t(b-a)} + \alpha\right),\quad \alpha = -a/(b-a),\quad u \in [0,1].$$

This gives $\phi(0) = \phi'(0) = 0$, $\phi''(t) \leq \frac{(b-a)^2}{4}$, since $u\cdot(1-u) \leq 1/4$.

By the Taylor Remainder Theorem $\exists \xi \in [a,b]$ s.t. $\phi(t) = \phi(0) + t\phi'(0) + \frac{t^2}{2}\phi''(\xi) \leq \frac{t^2(b-a)^2}{8}$

$$\Rightarrow \mathbb{E}[e^{tX}] \leq e^{\phi(t)} \leq e^{\frac{t^2(b-a)^2}{8}}$$ ■

Probability Theory and Inequalities

Concentration Inequalities

Lemma: (Hoeffding's Inequality) Let $X_1, X_2, \ldots, X_m$ be random variables with $a_i \leq X_i \leq b_i$, $b_i > a_i$ and $S_m = \sum_{i=1}^{m} X_i$ then we have the bounds

$$\Pr[S_m - \mathrm{E}[S_m] \geq \epsilon] \leq e^{-2\epsilon^2 / \sum_{i=1}^{m}(b_i - a_i)^2}$$

$$\Pr[S_m - \mathrm{E}[S_m] \leq -\epsilon] \leq e^{-2\epsilon^2 / \sum_{i=1}^{m}(b_i - a_i)^2}$$

Proof:

Let $Z_m = S_m - E[S_m]$ and $Q = \sum_{i=1}^{m}(b_i - a_i)^2$.

$$\Pr\{S_m - E[S_m] \geq \epsilon\} = \Pr\{Z_m \geq \epsilon\} \underset{\uparrow}{\leq} e^{-t\epsilon} E\left[e^{tZ_m}\right] = e^{-t\epsilon}\Pi_{i=1}^{m} E\left[e^{t(X_i - E[X_i])}\right] \underset{\uparrow}{\leq} e^{-t\epsilon} \exp\left(\frac{t^2 \sum_{i=1}^{m}(b_i - a_i)^2}{8}\right) = \exp\left(\psi(t)\right)$$

Markov Inequality

Hoeffding Lemma

We minimize $\psi(t)$ in t to obtain optimal upper bound.

$$\psi(t) = \frac{-8t\epsilon + t^2 Q}{8} \;\longrightarrow\; \psi'(t_*) = \frac{-8\epsilon + 2t_* Q}{8} = 0 \Rightarrow -8\epsilon + 2t_* Q = 0 \Rightarrow t_* = \frac{4\epsilon}{Q}.$$

$$\psi(t_*) = \frac{-32\epsilon^2}{8Q} + \frac{16\epsilon^2}{8Q} = \frac{-2\epsilon^2}{Q} \;\longrightarrow\; \exp(\psi(t_*)) = \exp\left(-2\epsilon^2 / \sum_{i=1}^{m}(b_i - a_i)^2\right).$$

Similarly, we obtain the other case using $\tilde{Z}_m = -Z_m$. ■

Generalization Bounds

Finite-Inconsistent Case: Guarantees on Sampling Complexity

Lemma: Let samples S={(x_1,y_1),(x_2,y_2),…,(x_m,y_m)} be chosen i.i.d. on {0,1} from D ~ $\mathcal{X}$ x $\mathcal{Y}$ then

$$\Pr_{S\sim D^m}\left[|\widehat{R}(h) - R(h)| \geq \epsilon\right] \leq 2\exp(-2m\epsilon^2)$$

Proof:

$$\hat{R}(h) = \frac{1}{m}\sum_{i=1}^{m} 1_{h(x_i)\neq c(x_i)} = \sum_{i=1}^{m} X_i = S_m, \quad X_i = \frac{1}{m}1_{h(x_i)\neq c(x_i)} \in \left[0, \frac{1}{m}\right]$$

By Hoeffding's Inequality

$$\Pr\{|\hat{R}(h) - R(h)| \geq \epsilon\} \leq 2e^{-2\epsilon^2/\sum_{i=1}^{m}(b_i-a_i)^2} = 2e^{\frac{-2\epsilon^2 m^2}{m}} = 2\exp\left(-2\epsilon^2 m\right) \blacksquare$$

Generalization Bounds

Finite-Inconsistent Case: Guarantees on Sampling Complexity

Theorem: Inconsistent-Finite Hypothesis Spaces $\mathcal{H}$. Let $\mathcal{A}$ be any learning algorithm that has empirical generalization error $\hat{R}(h_S)$ then for any $h \in \mathcal{H}$ we have with probability at least $1 - \delta$

$$R(h) \leq \widehat{R}(h) + \sqrt{\frac{\log|H| + \log\frac{2}{\delta}}{2m}}$$

Proof:

$$\Pr\{h \in \mathcal{H}, |\hat{R}(h) - R(h)| > \epsilon\} = \Pr\{h_1 \in \mathcal{H} \wedge |\hat{R}(h_1) - R(h_1)| > \epsilon \vee \cdots \vee h_{|\mathcal{H}|} \in \mathcal{H} \wedge |\hat{R}(h_{|\mathcal{H}|}) - R(h_{|\mathcal{H}|})| > \epsilon\}$$

$$\leq \sum_{i=1}^{|\mathcal{H}|} \Pr\{h_i \in \mathcal{H} \wedge |\hat{R}(h_i) - R(h_i)| > \epsilon\} \leq |\mathcal{H}| 2 \exp(-2m\epsilon^2) \leq \delta$$

$$\Rightarrow \log(|\mathcal{H}|) - 2m\epsilon^2 \leq \log\left(\frac{\delta}{2}\right) \Rightarrow 2m\epsilon^2 \geq \log\left(|\mathcal{H}|\right) + \log\left(\frac{2}{\delta}\right)$$

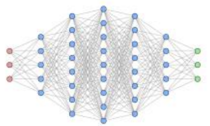
$$\Rightarrow \quad m \geq \frac{1}{2\epsilon^2}\left(\log(|\mathcal{H}|) + \log\left(\frac{2}{\delta}\right)\right)$$

$$\Rightarrow \epsilon \geq \sqrt{\frac{\log\left(|\mathcal{H}|\right) + \log\left(\frac{2}{\delta}\right)}{2m}} \quad \blacksquare$$

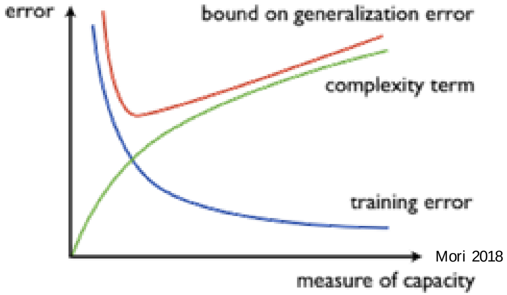
Generalization Behaviors

Generalization Error and Model Capacity

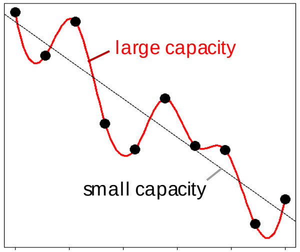
Deep Neural Network



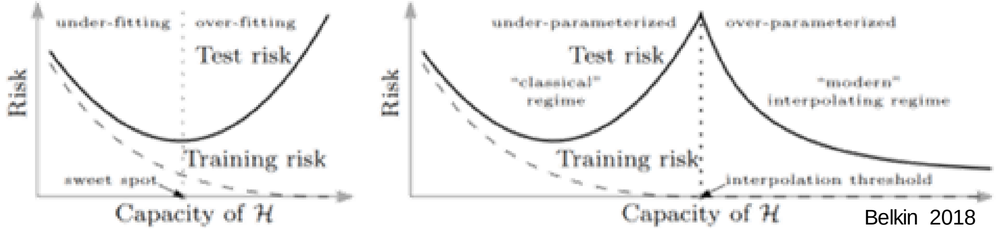
Classic Bias-Variance Trade-off



Fitting Data

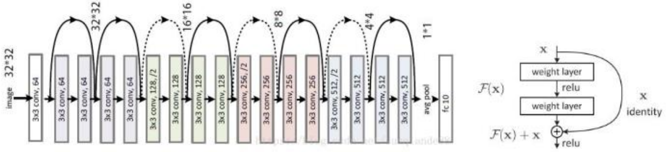


Double Descent and Deep Learning



Belkin 2018

ResNet-18



He 2016

Larger model capacity often allows for **smaller training error** (model capacity ~ $|\mathcal{H}|$).

Complexity of $\mathcal{H}$ tends to hinder generalization to new inputs.

Smallest generalization error arises intermediate trading-off in model complexity and training error (bias-variance trade-off).

Recent results show situation can be more subtle. Deep learning methods (neural networks) exhibit "double-descent."

Central challenge in machine learning been to find appropriate hypothesis classes for given learning tasks.

Central challenge in deep learning is to design appropriate neural network architectures, regularizations, initialization, training protocols.

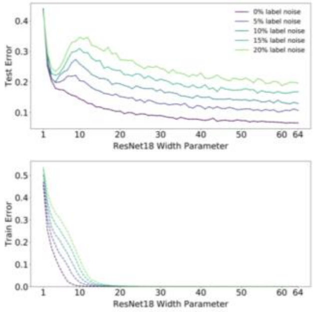
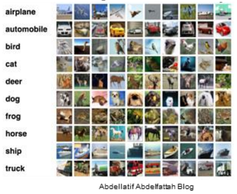


Image Classification



Minimax Rates and PAC-Learning

Minimax Rate

$$\mathcal{V}_m(\mathcal{C}) = \inf_{h_S=\mathcal{A}(\cdot)} \sup_{D_X, c\in\mathcal{C}} E_{S:|S|=m}\left[R(h_S)\right]$$

$\mathcal{X}$ input space, $\mathcal{Y}$ output space, c(x): $\mathcal{X} \rightarrow \mathcal{Y}$ concept
$\mathcal{C}$ concept class, $\mathcal{H}$ hypothesis class.

PAC-learning Classification:

$$\mathcal{V}_m^{PAC}(\mathcal{C}) = \inf_{h_S=\mathcal{A}(\cdot)} \sup_{D_X, c\in\mathcal{C}} E_{S:|S|=m}\left[\Pr_{x\sim D}\left\{h_S(x) \neq c(x)\right\}\right]$$

A concept class $\mathcal{C}$ is PAC-learnable if $\mathcal{V}_m^{PAC}(\mathcal{C}) \rightarrow 0$.
More precisely, given $\epsilon > 0$, $\exists M = \mathrm{poly}(1/\epsilon)$ such that $m \geq M$, we have

$$\mathcal{V}_m^{PAC}(\mathcal{C}) \leq \epsilon.$$

Theorem (PAC Learning ←→ Minimax): For a concept class $\mathcal{C}$ the minimax rate converges to zero with polynomial sampling complexity if and only if the concept class $\mathcal{C}$ is PAC-learnable.

Minimax Rates and PAC-Learning

Minimax Rate and PAC-Learning Classification $\mathcal{V}^{PAC}(\mathcal{C}) = \inf_{\hat{\mathcal{A}}} \sup_{\mathcal{D}_X, c \in \mathcal{C}} E_{S:|S|=m}\left[R(h_S = \mathcal{A}(S)\right]$

Theorem (PAC Learning ←→ Minimax):
Given $\epsilon > 0$, $\mathcal{V}_m^{PAC}(\mathcal{C}) \leq \epsilon$ with $m \geq \text{poly}(1/\epsilon)$ holds if and only if there is an algorithm $\mathcal{A}$ so that given $\epsilon > 0$, $\delta > 0$, $\Pr_{S \sim \mathcal{D}^m}\{R(h_S) \leq \epsilon\} \geq 1 - \delta$ for $m \geq \text{poly}(1/\epsilon, 1/\delta)$ holds.

Proof: $(i) \Rightarrow (ii)$ follows readily.

We show $(ii) \Rightarrow (i)$

$$R(h_S) = \Pr_{X \sim \mathcal{D}}\{h_S(X) \neq c(X)\}$$

Given (ii) we have $\exists \tilde{\tilde{A}}$ s.t. given $\epsilon > 0$, $\delta = \epsilon/2$, $\exists M = \text{poly}\left(\frac{1}{\epsilon}, \frac{1}{\epsilon}\right)$ s.t. for $D_X \in \mathfrak{D}$, $c \in \mathcal{C}$,

$$\Pr_{S \sim \mathcal{D}^m}\{R(\tilde{\tilde{A}}(S)) \leq \epsilon\} \geq 1 - \delta \Rightarrow \Pr_{S \sim \mathcal{D}^m}\{R(\tilde{\tilde{A}}(S)) > \epsilon\} < \delta,\ m \geq M.$$

We obtain the bound

$$E_{S:|S|=m}\left[R(\tilde{\tilde{A}}(S))\right] \leq \Pr_{S \sim \mathcal{D}^m}\{R(\tilde{\tilde{A}}(S) \leq \epsilon\} \cdot \epsilon + \Pr_{X \sim \mathcal{D}}\{R(\tilde{\tilde{A}}(S) > \epsilon\} \cdot 1 \leq \epsilon + \delta \leq \epsilon + \frac{1}{2}\epsilon = \frac{3}{2}\epsilon = \tilde{\epsilon}.$$

$$\Rightarrow \begin{cases} \mathcal{V}^{PAC}(\mathcal{C}) \leq \tilde{\epsilon} \\ m \geq \text{poly}(1/\tilde{\epsilon}) \end{cases}$$

$\Rightarrow \mathcal{V}_m^{PAC} \to 0$, as $m \to \infty$. ■

Minimax Rates and Learning Tasks

PAC-Learning Classification

$$\mathcal{V}_m^{PAC}(\mathcal{C}) = \inf_{h_S = \mathcal{A}(\cdot)} \sup_{D_X, c \in \mathcal{C}} E_{S:|S|=m} \left[\Pr_{x \sim D} \{h_S(x) \neq c(x)\} \right]$$

Non-parameteric Regression

$$\mathcal{V}_m^{NR}(\mathcal{C}) = \inf_{h_S = \mathcal{A}(\cdot)} \sup_{D_X, c \in \mathcal{C}} E_{S:|S|=m} \left[\left(h_S(x) - c(x)\right)^2 \right]$$

Agnostic PAC-Learning

$$\mathcal{V}_m^{A-PAC}(\mathcal{C}) = \inf_{h_S = \mathcal{A}(\cdot)} \sup_{D_X, c \in \mathcal{C}} E_{S:|S|=m} \left[R(h_S) - \inf_{h' \in \mathcal{H}} R(h') \right]$$

Comparison of learning problems:

Case: $\mathcal{C} \subset \{\pm 1\}^{\mathcal{X}}$

$$4\mathcal{V}_m^{PAC}(\mathcal{C}) \leq \mathcal{V}_m^{NR}(\mathcal{C}) \leq \mathcal{V}_m^{A-PAC}(\mathcal{C})$$

Case: $\mathcal{C} \subset R^{\mathcal{X}}$

$$\mathcal{V}_m^{NR}(\mathcal{C}) \leq \mathcal{V}_m^{A-PAC}(\mathcal{C})$$

Statistical Learning Theory

Machine Learning Algorithms and Tasks

- **Guaranteed performance** for unknown distributions $D_{\mathcal{X}}$ requires we have some restriction on the hypothesis class $\mathcal{H}$ and concept class $\mathcal{C}$.
- There is **no general learning algorithm** that works for all possible tasks.
- These assertions correspond to so-called "**No Free Lunch Theorems.**"
- **To achieve good performance** learning algorithms must make some use of knowledge / mathematical structure of the specific task.

Image Classification

Abdelfattah 2018

Robotics and Control

MIT and Boston Dynamics

Forecasting

washingtonpost.com

Statistical Learning Theory

No Free Lunch Theorem

Theorem: Let concept class be all binary functions, $\mathcal{C} = \mathcal{U}$ = {all functions $f(z): \mathcal{X} \rightarrow \{0,1\}$},
where $\mathcal{X}$ is discrete space of finite binary sequences $\{\{0,1\}^N, N \in \mathbb{N}\} = \{(z_1,z_2,\ldots,z_N),\ z_i \in \{0,1\}\}$.
For the universal concept class $\mathcal{U}$ we have $\mathcal{V}_m^{PAC}(\mathcal{C}) \not\to 0$.
Therefore, $\mathcal{U}$ is **not** **PAC-Learnable.**

Proof:
For a given sample size, let $\mathcal{X} \subset \Omega$ of binary sequences s.t. $|\mathcal{X}| = 2n$.
Let $\mathcal{D}_f$ ~ uniform distribution over all functions $f : \mathcal{X} \rightarrow \{0,1\}$. Note $|\mathcal{Y}^{\mathcal{X}}| = 2^{2n}$ when $\mathcal{X} \in \{0,1\}^{2n}$.
Consider $Q = E_{\mathcal{D}_f}\left[E_{\mathcal{S}:|\mathcal{S}|=m}\left[R\left(\mathcal{A}(\mathcal{S})\right)\right]\right]$, $R\left(\mathcal{A}(\mathcal{S})\right) = R(h_{\mathcal{S}}) = E\left[1_{h_{\mathcal{S}}(x)\neq f(x)}\right] = \Pr\{h_{\mathcal{S}}(X) \neq f(X)\}$.

We will show that $Q \geq 1/4$ for $\mathcal{C} = \mathcal{U}$, which will prevent $\mathcal{V}_m(\mathcal{C}) \not\to 0$.

By Fubini's Theorem

$$Q = E_{\mathcal{S}:|\mathcal{S}|=m}\left[E_{\mathcal{D}_f}\left[R\left(\mathcal{A}(\mathcal{S})\right)\right]\right] = E_{\mathcal{S}:|\mathcal{S}|=m}\left[E_{\mathcal{D}_f}\left[E_{X\sim\mathcal{D}}\left[1_{h_{\mathcal{S}}(X)\neq f(X)}\right]\right]\right] = E_{\mathcal{S}:|\mathcal{S}|=m}\left[E_{X\sim\mathcal{D}}\left[E_{\mathcal{D}_f}\left[1_{h_{\mathcal{S}}(X)\neq f(X)}\right]\right]\right]$$

$$= E_{\mathcal{S},X\sim\mathcal{D}}\left[E_{\mathcal{D}_f}\left[1_{h_{\mathcal{S}}(X)\neq f(X)}|X \in \mathcal{S}\right]\right] \cdot \Pr\{X \in \mathcal{S}\} + E_{\mathcal{S},X\sim\mathcal{D}}\left[E_{\mathcal{D}_f}\left[1_{h_{\mathcal{S}}(X)\neq f(X)}|X \notin \mathcal{S}\right]\right] \cdot \Pr\{X \notin \mathcal{S}\}$$

$$\geq E_{\mathcal{S},X\sim\mathcal{D}}\left[E_{\mathcal{D}_f}\left[1_{h_{\mathcal{S}}(X)\neq f(X)}|X \notin \mathcal{S}\right]\right] \cdot \Pr\{X \notin \mathcal{S}\}$$

$$\geq \frac{1}{2}\Pr\{X \notin \mathcal{S}\} \geq \frac{1}{2}\cdot\frac{1}{2} = \frac{1}{4}. \qquad \blacksquare$$

$\mathcal{D} \sim$ uniform on $\mathcal{X}$, $|\mathcal{X}| = 2n$
$|\mathcal{S}| = n \Rightarrow \Pr\{X \notin \mathcal{S}\} \geq \frac{1}{2}$

Challenges in Machine Learning

Guarantees for performance. Typically, unknown distributions $D_{\mathcal{X}}$, may shift in time, good choices needed for hypothesis class $\mathcal{H}$, types and amount of data.

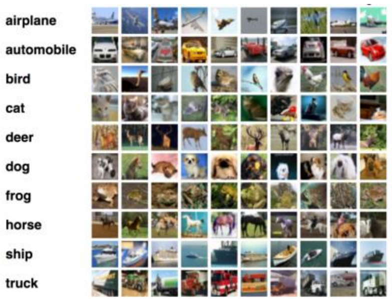
No Free Lunch Theorems: If the hypothesis class $\mathcal{H}$, target concept class $\mathcal{C}$ are too general and the distribution $D_{\mathcal{X}}$ is unknown then there is no guarantees on algorithmic performance on the tasks.

This means **no generic all purpose learning algorithms** exist.

Must utilize prior knowledge or structure of the tasks to be solved.

Central goal of this course is to consider wide variety of specific tasks and develop associated theory and well-suited learning algorithms.

Image Classification

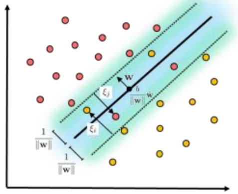


Abdelfattah 2018

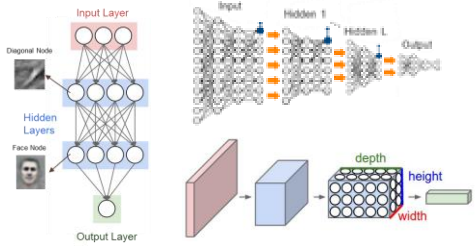
Robotics and Control

MIT and Boston Dynamics

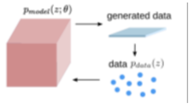
Support Vector Machines



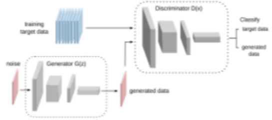
Neural Networks and Deep Learning



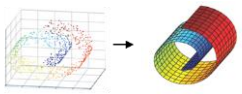
Generative Methods



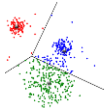
GANs



Manifold Learning



Clustering Methods



Forecasting

washingtonpost.com