Generalization and Simplification in Machine Learning

Shay Moran
School of Mathematics, IAS Princeton

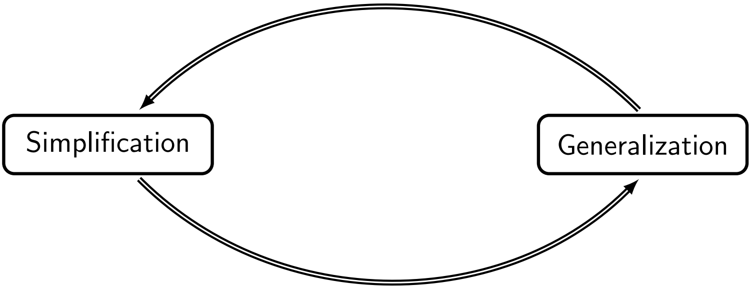
Two dual aspects of "learning"

Two aspects:

1. Generalization:
Infer new knowledge from existing knowledge.

2. Simplification:
Provide simple(r) explanations for existing knowledge.

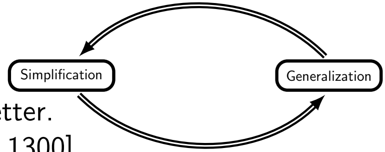
Interrelations



e.g. math:

theorem $\xrightarrow{\text{simplification}}$ simpler proof $\xrightarrow{\text{generalization}}$ more general theorem

Philosophical heuristics



Simpler (consistent) explanations are better.
[Occam's razor – William of Ockham $\approx$ 1300].

simplification $\implies$ generalization

If I can't reduce it to a freshman level
then I don't really understand it. [Richard Feynman 1980's].

when James Gleick (a science reporter) asked him to explain why spin-1/2 particles obey Fermi-Dirac statistics

When presented with a complicated proof, Erdös used to reply:
"Now, let's find the book's proof..." [Paul Erdös]

generalization $\implies$ simplification

Can these relations be manifested as theorems in learning theory?

This talk

"Simplification $\equiv$ Generalization"

in

Learning Theory

Plan

Generalization

Simplification/compression

The "generalization – compression" equivalence

- Binary classification
- Multiclass categorization
- Vapnik's general setting of learning

Discussion

Generalization:

General Setting of Learning
[Vapnik '95]

Intuition

Imagine a scientist that performs m experiments with outcomes

$$z_1, \ldots, z_m$$

and wishes to predict the outcome of future experiments.

Classification example: intervals

$\mathcal{D}$ – **unknown** distribution over $\mathbb{R}$
c – **unknown** interval:

Given: Training set

$$S = \big(x_1, c(x_1)\big), \ldots, \big(x_m, c(x_m)\big) \;\sim\; \mathcal{D}^m$$

Goal: Find $h = h(S) \subseteq \mathbb{R}$ that minimizes the disagreement with c

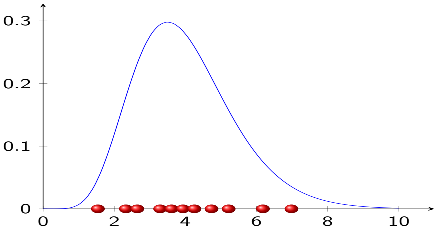
$$\mathbb{E}_{x \sim \mathcal{D}} \left[1_{c(x) \neq h(x)} \right]$$

in the Probably (w.p. $1 - \delta$) Approximately Correct (up to ϵ) sense

Regression example: mean estimation

$\mathcal{D}$ – **unknown** distribution over $[0, 1]$

Given: Training set $S = z_1, \ldots, z_m \sim \mathcal{D}^m$



Goal: Find $h = h(S) \in [0, 1]$ that minimizes

$$\mathbb{E}_{x \sim D}\left[(x - h)^2\right]$$

in the Probably (w.p. $1 - \delta$) Approximately Correct (up to ϵ) sense

The General Setting of Learning: definition

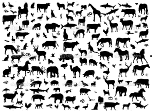
$\mathcal{H}$ hypothesis class
$\mathcal{D}$ distribution over examples
ℓ loss function

The General Setting of Learning: definition

$\mathcal{H}$ hypothesis class
$\mathcal{D}$ distribution over examples
ℓ loss function

Output: h_{out}

known to learner: $\mathcal{H}$
unknown to learner: $\mathcal{D}$



Nature

$$\underbrace{z_1, \ldots, z_m}_{\text{i.i.d examples from } \mathcal{D}}$$

Learning algorithm

The General Setting of Learning: definition

$\mathcal{H}$ hypothesis class

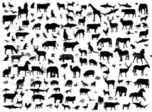
$\mathcal{D}$ distribution over examples

ℓ loss function

Output: h_{out}

known to learner: $\mathcal{H}$

unknown to learner: $\mathcal{D}$



Nature

$\xrightarrow{\underbrace{z_1, \ldots, z_m}_{\text{i.i.d examples from } \mathcal{D}}}$

Learning algorithm

$$\text{Goal: } \textbf{loss} \text{ of } h_{out} \underbrace{\leq}_{\text{in the PAC sense}} \textbf{loss} \text{ of best } h \in \mathcal{H}$$

classification problems, regression problems, some clustering problems,...

Examples

Binary classification:

- $\mathcal{Z} = X \times \{0, 1\}$
- $\mathcal{H}$ – class of $X \to \{0, 1\}$ functions
- $\ell\big(h, (x, y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Examples

Binary classification:

- $\mathcal{Z} = X \times \{0, 1\}$
- $\mathcal{H}$ – class of $X \to \{0, 1\}$ functions
- $\ell\big(h, (x, y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Multiclass categorization:

- $\mathcal{Z} = X \times Y$
- $\mathcal{H}$ – class of $X \to Y$ functions
- $\ell\big(h(x, y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Examples

Binary classification:

- $\mathcal{Z} = X \times \{0, 1\}$
- $\mathcal{H}$ – class of $X \to \{0, 1\}$ functions
- $\ell\big(h, (x, y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Multiclass categorization:

- $\mathcal{Z} = X \times Y$
- $\mathcal{H}$ – class of $X \to Y$ functions
- $\ell\big(h(x, y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Mean estimation:

- $\mathcal{Z} = [0, 1]$
- $\mathcal{H} = [0, 1]$
- $\ell(h, z) = (h - z)^2$

Examples

Binary classification:

- $\mathcal{Z} = X \times \{0,1\}$
- $\mathcal{H}$ – class of $X \to \{0,1\}$ functions
- $\ell\big(h,(x,y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Multiclass categorization:

- $\mathcal{Z} = X \times Y$
- $\mathcal{H}$ – class of $X \to Y$ functions
- $\ell\big(h(x,y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Mean estimation:

- $\mathcal{Z} = [0,1]$
- $\mathcal{H} = [0,1]$
- $\ell(h,z) = (h-z)^2$

Linear regression:

- $\mathcal{Z} = \mathbb{R}^d \times \mathbb{R}$
- $\mathcal{H}$ – class of affine $\mathbb{R}^d \to \mathbb{R}$ functions
- $\ell\big(h,(x,y)\big) = \big(h(x)-y\big)^2$

Agnostic and realizable-case Learnability

$\mathcal{H}$ – hypothesis class

$\mathcal{H}$ is agnostic learnable:
$\exists$ algorithm $\mathcal{A}$, s.t. for every $\mathcal{D}$, if $m > n^{agn}(\epsilon, \delta)$

$$\Pr_{S \sim \mathcal{D}^m}[L_{\mathcal{D}}(\mathcal{A}(S)) \geq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon] \leq \delta$$

Agnostic and realizable-case Learnability

$\mathcal{H}$ – hypothesis class

$\mathcal{H}$ is agnostic learnable:
$\exists$ algorithm $\mathcal{A}$, s.t. for every $\mathcal{D}$, if $m > n^{agn}(\epsilon, \delta)$

$$\Pr_{S \sim \mathcal{D}^m}[L_{\mathcal{D}}(\mathcal{A}(S)) \geq \min_{h \in \mathcal{H}} L_{\mathcal{D}}(h) + \epsilon] \leq \delta$$

$\mathcal{H}$ is realizable-case learnable:
$\exists$ algorithm $\mathcal{A}$ s.t. for every **realizable** $\mathcal{D}$, if $m > n^{real}(\epsilon, \delta)$

$$\Pr_{S \sim \mathcal{D}^m}[L_{\mathcal{D}}(\mathcal{A}(S)) \geq \epsilon] \leq \delta$$

- $\mathcal{D}$ is realizable if there is $h \in \mathcal{H}$ with $L_{\mathcal{D}}(h) = 0$

Compression:

Sample compression schemes

[Littlestone,Warmuth '86]

Intuition

Imagine a scientist that performs m experiments with outcomes

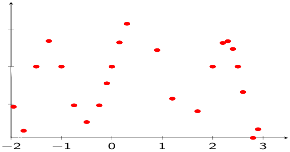
$$z_1, \ldots z_m$$

and wishes to choose $d \ll m$ of them in a way that allows to explain all other experiments (choose d axioms)

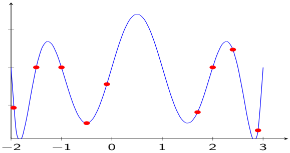
Example: polynomials

P – **unknown** polynomial of degree $\leq d$:

Input: training set of m evaluations of P $\quad (d \ll m)$



Compression: Keep $d + 1$ points



Reconstruction: Lagrange Interpolation

Evaluates to the correct value on the whole training set

Compression algorithm: definition

[Littlestone,Warmuth '86]

$\mathcal{H}$ hypothesis class

ℓ loss function

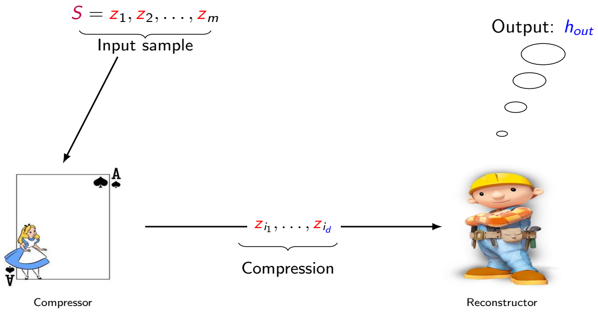
Compression algorithm: definition

[Littlestone, Warmuth '86]

$\mathcal{H}$ hypothesis class

ℓ loss function

Compression scheme of size d:



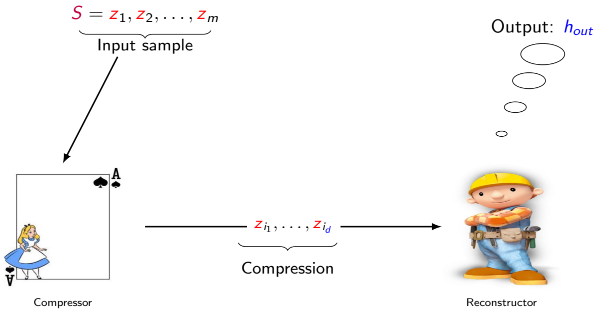
Compression algorithm: definition

[Littlestone,Warmuth '86]

$\mathcal{H}$ hypothesis class

ℓ loss function

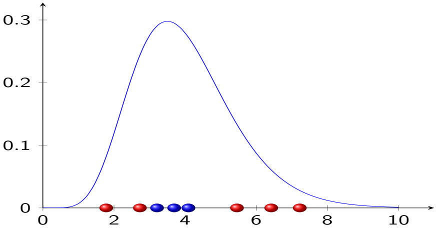
Compression scheme of size d:



Compression algorithms examples

Compression algorithm for interval approximation of size 2:
"output the smallest interval containing the *positive* *examples"*

Compression algorithm for mean estimation of size 3:
"output the average of 3 sample points with minimal empirical error"



Data fitting – A fundamental property of compression algorithms

S – a sample drawn from $\mathcal{D}^m$
$\mathcal{A}$ – sample compression algorithm of size d
$h = \mathcal{A}(S)$

Theorem

$$\Pr_{S \sim \mathcal{D}^m}\left[\left|L_{\mathcal{D}}(h) - L_S(h)\right| \geq \epsilon\right] \leq \delta,$$

where

$$\epsilon \approx \sqrt{\frac{d + \log(1/\delta)}{m}}.$$

In order to **generalize** it suffices to find a **short compression** with **low empirical error**

Sample compression schemes for hypothesis classes

$\mathcal{H}$ – a hypothesis class

An agnostic-case sample compression scheme for $\mathcal{H}$:
A compression algorithm $\mathcal{A}$ s.t. for every S

$$L_S(\mathcal{A}(S)) \leq \min_{h \in \mathcal{H}} L_S(h)$$

A realizable-case sample compression scheme for $\mathcal{H}$:
A compression algorithm $\mathcal{A}$ s.t. for every **realizable** S

$$L_S(\mathcal{A}(S)) = 0$$

- S is realizable if there is $h \in \mathcal{H}$ with $L_S(h) = 0$

Plan

Generalization

Simplification/compression

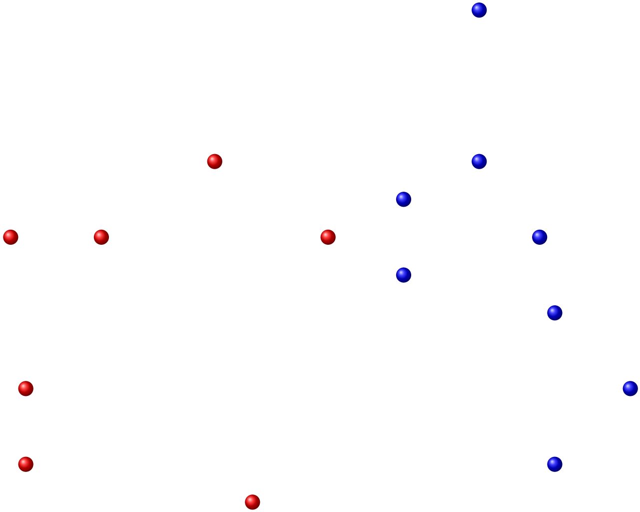
The "generalization – compression" equivalence

- Binary classification
- Multiclass categorization
- Vapnik's general setting of learning

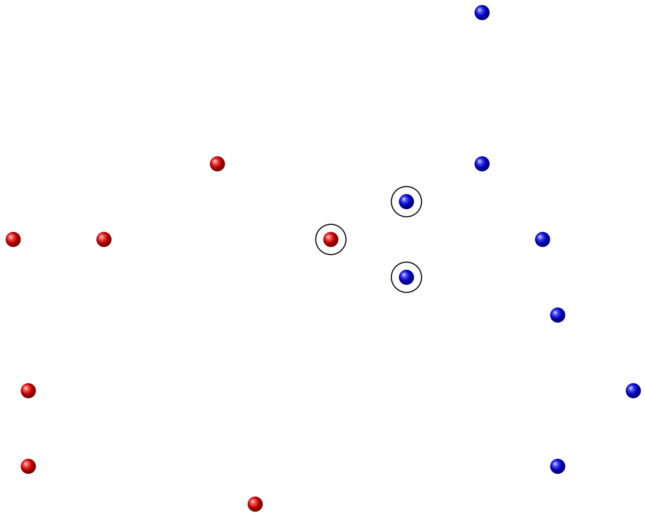
Discussion

The "generalization – compression" equivalence

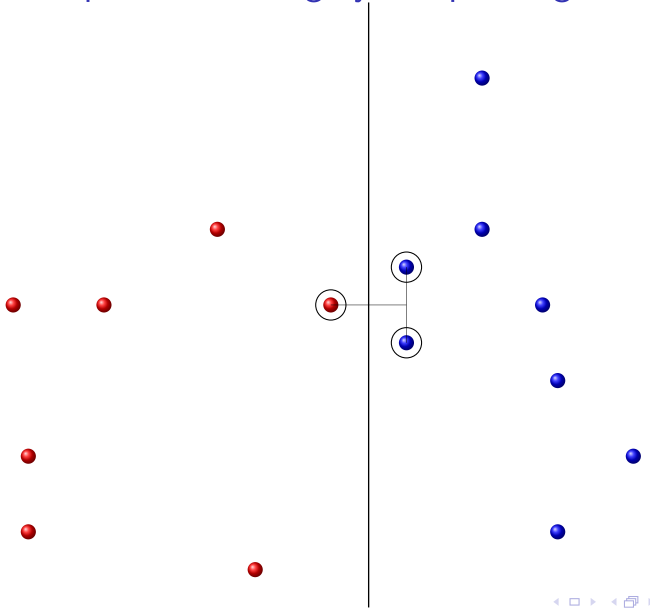
Support Vector Machines:
an example of “learning by compressing”



Support Vector Machines: an example of "learning by compressing"



Support Vector Machines:
an example of "learning by compressing"



Binary classification:

Probably Approximately Correct (PAC) learning

[Vapnik-Chervonenkis '71], [Valiant '84]

Binary classification

Hypothesis class:
$\mathcal{H}$ – class of $X \to \{0, 1\}$ functions

Loss function:
$\ell\big(h, (x, y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Distribution:
$\mathcal{D}$ on $X \times \{0, 1\}$

The VC dimension captures the sample complexity in binary classification problems

[Sample complexity]:
minimum sample-size sufficient
for learning $\mathcal{H}$.
$\big($with confidence $2/3$ and error $1/3\big)$

The VC dimension captures the sample complexity in binary classification problems

[Sample complexity]:
minimum sample-size sufficient for learning $\mathcal{H}$.
(with confidence 2/3 and error 1/3)

[VC dimension]:
$dim(\mathcal{H}) = \max\{|Y| : Y \text{ is shattered}\}$,
where $Y \subseteq X$ is shattered if $\mathcal{H}|_Y = \{0,1\}^Y$.

Theorem
[Vapnik,Chervonenkis], [Blumer,Ehrenfeucht,Haussler,Warmuth], [Ehrenfeucht,Haussler,Kearns,Valiant]:

The sample complexity of $\mathcal{H} \approx dim(\mathcal{H})$

The VC dimension captures the sample complexity in binary classification problems

[Sample complexity]:
minimum sample-size sufficient for learning $\mathcal{H}$.
(with confidence $2/3$ and error $1/3$)

[VC dimension]:
$dim(\mathcal{H}) = \max\{|Y| : Y \text{ is shattered}\}$,
where $Y \subseteq X$ is shattered if $\mathcal{H}|_Y = \{0,1\}^Y$.

$$\begin{array}{ccccc} & \overbrace{\quad\quad}^{Y} & & & \\ v_1 & 0 & 0 & 1 & 1 \\ v_2 & 0 & 1 & 1 & 1 \\ v_3 & 1 & 0 & 1 & 1 \\ v_4 & 1 & 1 & 0 & 1 \\ v_5 & 0 & 0 & 0 & 0 \end{array}$$

Theorem
[Vapnik,Chervonenkis], [Blumer,Ehrenfeucht,Haussler,Warmuth], [Ehrenfeucht,Haussler,Kearns,Valiant]:

The sample complexity of $\mathcal{H} \approx dim(\mathcal{H})$

Compression vs simplification

[Littlestone, Warmuth '86]

Theorem (simplification $\implies$ generalization):
If $\mathcal{H}$ has a compression scheme of size k then $dim(\mathcal{H}) = O(k)$.

A manifestation of Occam's razor.

Question (generalization $\implies$ simplification?):

Is there a compression scheme of size depending only on $dim(\mathcal{H})$?

A manifestation of Feynman's statement.

Previous works

Boosting: $dim(\mathcal{H}) \log m$ compression scheme
[Freund,Schapire '95]

Compression schemes for special well-studied concept classes
[Floyd,Warmuth '95],[Floyd '89],[Helmbold,Sloan,Warmuth '92], [Ben-David,Litman '98],[Chernikov,Simon '13],[Kuzmin,Warmuth '07], [Rubinstein,Bartlett,Rubinstein '09],[Rubinstein,Rubinstein '13], [Livni,Simon '13], [M,Warmuth '15] ...

Connection with model theory
[Chernikov,Simon '13],[Livni,Simon '13],[Johnson '09],...

Connection with algebraic topology
[Rubinstein,Bartlett,Rubinstein '09],[Rubinstein,Rubinstein '12]

Enough to compress finite classes (A compactness theorem)
[Ben-David,Litman '98]

$\log |\mathcal{H}|$ compression scheme
[Floyd,Warmuth '95]

$\exp(dim(\mathcal{H})) \log \log |\mathcal{H}|$ compression scheme
[M,Shpilka,Wigderson,Yehudayoff '15]

Generalization $\Longrightarrow$ Compression

Theorem[M-Yehudayoff]
There exists a sample compression scheme of size $\exp(dim(\mathcal{H}))$

Proof uses: Minimax theorem, duality ,ϵ-net theorem (ϵ-approximation)

Generalization $\implies$ Compression

Theorem[M-Yehudayoff]
There exists a sample compression scheme of size $\exp(dim(\mathcal{H}))$

Proof uses: Minimax theorem, duality ,ϵ-net theorem (ϵ-approximation)

Further research 1: (Manfred Warmuth offers 600$!)
Replace $\exp(dim(\mathcal{H}))$ by $O(dim(\mathcal{H}))$

Further research 2:
Extend to other learning models

Multiclass categorization

Multiclass categorization

Hypothesis class:
$\mathcal{H}$ – class of $X \to Y$ functions

Loss function:
$\ell\big(h(x,y)\big) = \mathbf{1}\big[h(x) \neq y\big]$

Distribution:
$\mathcal{D}$ on $X \times Y$

Compressibility $\equiv$ Learnability

Theorem[David-M-Yehudayoff]

$\mathcal{H}$ is learnable $\iff \mathcal{H}$ has "$m \to \tilde{O}(\log m)$" compression

big oh hides efficient dependency on the weak sample complexity of $\mathcal{H}$ ($\epsilon = \delta = 1/3$)

Compressibility $\equiv$ Learnability

Theorem[David-M-Yehudayoff]

$$\mathcal{H} \text{ is learnable } \iff \mathcal{H} \text{ has ``} m \to \tilde{O}(\log m) \text{'' compression}$$

big oh hides efficient dependency on the weak sample complexity of $\mathcal{H}$ $(\epsilon = \delta = 1/3)$

Open question:

$$\mathcal{H} \text{ is learnable } \overset{?}{\iff} \mathcal{H} \text{ has ``} m \to O(1) \text{'' compression}$$

yes, when number of categories is $O(1)$ (e.g. binary classification)

Vapnik's general setting of learning

General setting

Hypothesis class:
$\mathcal{H}$ – a set

Loss function (bounded):
$\ell(h, z)$

Distribution:
$\mathcal{D}$ on Z

e.g. mean estimation

Compressibility $\equiv$ learnability? not so fast...

Agnostic compression scheme for "mean estimation" means:

Find a compression κ and a reconstruction ρ s.t.

Given: $S = z_1, \ldots, z_m \in [0, 1]$
Goal:

- $\mathcal{S}' = \kappa(S)$ is a small subsample of S, and
- $\rho(\mathcal{S}')$ is the mean of S:

$$\rho(\mathcal{S}') = \frac{z_1 + \ldots + z_m}{m}$$

Compressibility $\equiv$ learnability? not so fast...

Agnostic compression scheme for "mean estimation" means:

Find a compression κ and a reconstruction ρ s.t.

Given: $S = z_1, \ldots, z_m \in [0, 1]$
Goal:

- $\mathcal{S}' = \kappa(S)$ is a small subsample of S, and
- $\rho(\mathcal{S}')$ is the mean of S:

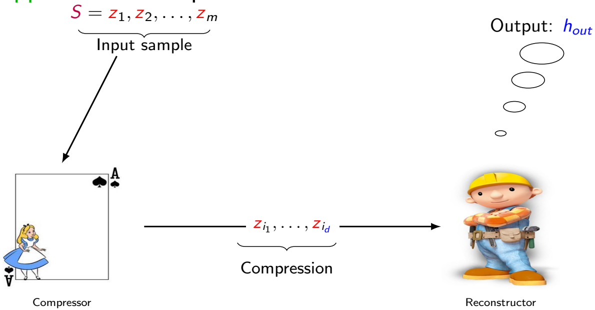
$$\rho(\mathcal{S}') = \frac{z_1 + \ldots + z_m}{m}$$

Theorem. [David-M-Yehudayoff]
There is no agnostic sample compression scheme for mean estimation of size $\leq m/2$.

Approximate sample compression schemes save the day

$\mathcal{H}$ hypothesis class
ℓ loss function
ϵ approximation parameter

Approximate compression scheme of size d:



Goal: [**empirical loss** of h_{out}] $\leq$ [**empirical loss** of best $h \in \mathcal{H}$] $+\ \epsilon$

Compressibility $\equiv$ learnability

general loss function

multiclass categorization,
regression models,
unsupervised models (e.g. k-means clustering)

Theorem[David-M-Yehudayoff]

$\mathcal{H}$ is learnable $\iff$ $\mathcal{H}$ is approximately compressible

ϵ-error learning sample size $\approx$ ϵ-error compressing sample size

Plan

Generalization

Simplification/compression

The “generalization – compression” equivalence

- Binary classification
- Multiclass categorization
- Vapnik’s general setting of learning

Discussion

Conclusions of the compression-generalization equivalence

1. **Practice**: universal guideline for designing learning algorithms:

   "Find a small and insightful subset of the input data"

2. **Theory**: link between statistics and combinatorics/geometry

3. **Didactic**: Compressibility is "simpler" than learnability.

Generalization bounds in the era of deep learning

A learning algorithm does **not overfit** if:

$$\text{empirical error} \approx \text{test error}$$

Generalization bounds in the era of deep learning

A learning algorithm does **not overfit** if:

$$\text{empirical error} \approx \text{test error}$$

Statistical learning provides a rich theory for **uniform-convergence bounds**.

- These bounds are tailored to Empirical Risk Minimizers

  (output hypothesis with minimum training error within a class of bounded capacity)
- Cannot explain why Deep-Learning algorithms does not overfit

Generalization bounds in the era of deep learning

A learning algorithm does **not overfit** if:

$$\text{empirical error} \approx \text{test error}$$

Statistical learning provides a rich theory for **uniform-convergence bounds**.

- These bounds are tailored to Empirical Risk Minimizers

  (output hypothesis with minimum training error within a class of bounded capacity)
- Cannot explain why Deep-Learning algorithms does not overfit

Need algorithm-dependent based generalization bounds

- E.g. margin, stability, PAC-Bayes, . . . , **compression**

Summary

Learning:

- Generalization
- Simplification/compression

$$\text{"}\textit{simplification} \equiv \textit{generalization}\text{"}$$

Further research

- Extend the equivalence to other models
  (e.g. interactive learning models)
- Find compression algorithms for important learning problems
  (e.g. regression, neural nets, etc.)

Спасибо
Gracias
شكر
Obrigado
Спасибо
Dank U
Grazie
Ευχαριστώ
Danke
Ευχαριστώ
Merci
Thank You
Ngiyabonga
Dank U
Diolch
Ngiyabonga
Obrigado
Thank You
Tack
Merci
Ευχαριστώ
Dziękuję
Danke
Grazie
תודה
Dank U
Terima Kasih
Diolch
Merci
Dziękuję
Grazie
Tack
Diolch
Gracias
Merci
תודה
Tack

Agnostic learnability vs. realizable-case learnability

Case I: Multiclass categorization

$\mathcal{H}$ – a class of $X \to Y$ functions
ℓ – loss function $\ell(h, (x, y)) = \mathbf{1}[h(x) \neq y]$
Clearly, if $\mathcal{H}$ is agnostic learnable then $\mathcal{H}$ is learnable in the realizable-case

Case I: Multiclass categorization

$\mathcal{H}$ – a class of $X \to Y$ functions
ℓ – loss function $\ell(h,(x,y)) = \mathbf{1}[h(x) \neq y]$
Clearly, if $\mathcal{H}$ is agnostic learnable then $\mathcal{H}$ is learnable in the realizable-case

How about the other direction?
Can a realizable-case learner be transformed to an agnostic learner?

Case I: Multiclass categorization

$\mathcal{H}$ – a class of $X \to Y$ functions
ℓ – loss function $\ell(h,(x,y)) = \mathbf{1}[h(x) \neq y]$
Clearly, if $\mathcal{H}$ is agnostic learnable then $\mathcal{H}$ is learnable in the realizable-case

How about the other direction?
Can a realizable-case learner be transformed to an agnostic learner?

$|Y|$ is small $\implies$ yes, via standard VC-theory
agnostic $\equiv$ realizable $\equiv$ uniform convergence

$|Y|$ is large $\implies$???
- poorly understood
- mysterious behaviour
- learning rate can be much faster than uniform convergence rate
(see e.g. Daniely-Sabato-(Ben-David)-(Shalev-Shwartz) '15)

In multiclass categorization, agnostic and realizable-case learnability are equivalent

Theorem[David-M-Yehudayoff]
$\mathcal{H}$ is realizable-case learnable $\implies$ $\mathcal{H}$ is agnostic learnable

Sketch of proof:
Compression $\equiv$ learnability gives:

$$\text{realiable-case learner} \implies \text{realizable-case compression}$$
$$\text{agnostic compression} \implies \text{agnostic learner}$$

Enough to show:

$$\text{realizable-case compression} \implies \text{agnostic compression}$$

In multiclass categorization, agnostic and realizable-case learnability are equivalent

Theorem[David-M-Yehudayoff]
$\mathcal{H}$ is realizable-case learnable $\implies$ $\mathcal{H}$ is agnostic learnable

Sketch of proof:
Compression $\equiv$ learnability gives:

$$\text{realiable-case learner} \implies \text{realizable-case compression}$$
$$\text{agnostic compression} \implies \text{agnostic learner}$$

Enough to show:

$$\text{realizable-case compression} \implies \text{agnostic compression}$$

Given a sample S, pick a largest realizable $S' \subseteq S$ and compress S' using the realizable-case compression...

Application: agnostic learnability $\neq$ realizable-case learnability

Under the zero/one loss function (mulitclass categorization) agnostic and realizable-case learning are equivalent

Application: agnostic learnability $\neq$ realizable-case learnability

Under the zero/one loss function (mulitclass categorization) agnostic and realizable-case learning are equivalent

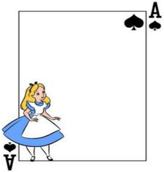
This equivalence breaks for general loss functions

Theorem[David-M-Yehudayoff] There exists a learning problem, with a loss function taking values in $\{0, \frac{1}{2}, 1\}$ that is learnable in the realizable-case but not agnostic learnable

generalization – compression equivalence reduces the separation to combinatorial problems such as:

Output: B

$x_1, x_2, x_3, \ldots, x_{2m}$



$\longrightarrow x_{i_1}, \ldots, x_{i_{m-1}} \longrightarrow$

- Alice's input: a list $x_1, x_2, \ldots, x_{2m}$ of real numbers

- Sends to Bob a sublist of size $m - 1$

- Bob outputs a finite $B \subseteq \mathbb{R}$ (as large as he wants)

- Success: if $\left|B \cap \{x_1, \ldots, x_{2m}\}\right| \geq m$.

- Is there a strategy that is successful for every input?

More applications

This work:

Dichotomy:
non-trivial compression implies logarithmic compression

Compactness theorem (multiclass categorization):
learnability of finite subclasses implies learnability

and more...

More applications

This work:

Dichotomy:
non-trivial compression implies logarithmic compression

Compactness theorem (multiclass categorization):
learnability of finite subclasses implies learnability

and more. . .

Other works:

Boosting
[Freund,Schapire '95]

Learnability with robust generalization guarantees
[Cummings, Ligett, Nissim, Roth, Wu '16]

and more. . .