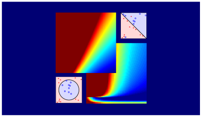
Machine Learning Foundations
(機器學習基石)



Lecture 16: Three Learning Principles

Hsuan-Tien Lin (林軒田)
htlin@csie.ntu.edu.tw

Department of Computer Science
& Information Engineering

National Taiwan University
(國立台灣大學資訊工程系)

Roadmap

1. When Can Machines Learn?
2. Why Can Machines Learn?
3. How Can Machines Learn?
4. How Can Machines Learn **Better**?

Lecture 15: Validation

(**crossly**) reserve **validation data** to simulate testing procedure for **model selection**

Lecture 16: Three Learning Principles

- Occam's Razor
- Sampling Bias
- Data Snooping
- Power of Three

Occam's Razor

An explanation of the data should be made as simple as possible, but no simpler.—Albert Einstein**?** (1879-1955)

entia non sunt multiplicanda praeter necessitatem
(entities must not be multiplied **beyond necessity**)
—William of Occam (1287-1347)

'**Occam's razor**' for trimming down unnecessary explanation

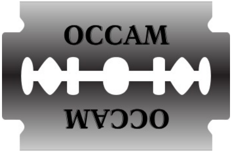
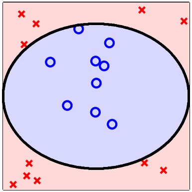


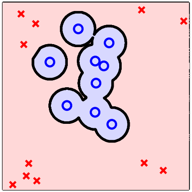
figure by Fred the Oyster (Own work) [CC-BY-SA-3.0], via Wikimedia Commons

Occam's Razor for Learning

The simplest model that fits the data is also the most plausible.



which one do you prefer? :-)



two questions:

1. What does it mean for a model to be simple?
2. How do we know that simpler is better?

Simple Model

simple hypothesis h

- small $\Omega(h)$ = 'looks' simple
- specified by **few parameters**

simple model $\mathcal{H}$

- small $\Omega(\mathcal{H})$ = not many
- contains **small number of hypotheses**

connection

h specified by ℓ bits $\Leftarrow |\mathcal{H}|$ of size 2^ℓ

small $\Omega(h) \Leftarrow$ small $\Omega(\mathcal{H})$

simple: **small hypothesis/model complexity**

Simple is Better

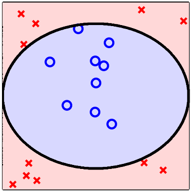
in addition to **math proof** that you have seen, philosophically:

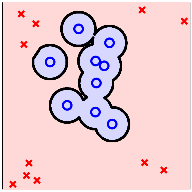
simple $\mathcal{H}$

$\Longrightarrow$ smaller $m_{\mathcal{H}}(N)$

$\Longrightarrow$ less 'likely' to fit data perfectly $\frac{m_{\mathcal{H}}(N)}{2^N}$

$\Longrightarrow$ more significant when fit happens





direct action: **linear first**;
always ask whether **data over-modeled**

Fun Time

Consider the decision stumps in $\mathbb{R}^1$ as the hypothesis set $\mathcal{H}$. Recall that $m_{\mathcal{H}}(N) = 2N$. Consider 10 different inputs $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{10}$ coupled with labels y_n generated iid from a fair coin. What is the probability that the data $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^{10}$ is separable by $\mathcal{H}$?

1. $\frac{1}{1024}$
2. $\frac{10}{1024}$
3. $\frac{20}{1024}$
4. $\frac{100}{1024}$

Fun Time

Consider the decision stumps in $\mathbb{R}^1$ as the hypothesis set $\mathcal{H}$. Recall that $m_{\mathcal{H}}(N) = 2N$. Consider 10 different inputs $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{10}$ coupled with labels y_n generated iid from a fair coin. What is the probability that the data $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^{10}$ is separable by $\mathcal{H}$?

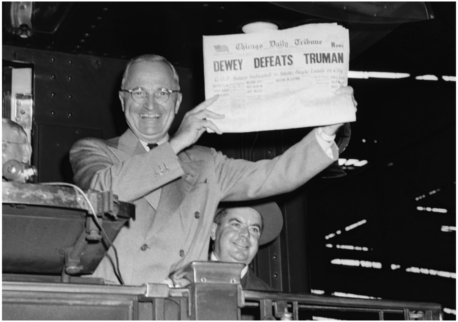
1. $\frac{1}{1024}$
2. $\frac{10}{1024}$
3. $\frac{20}{1024}$
4. $\frac{100}{1024}$

Reference Answer: ③

Of all 1024 possible $\mathcal{D}$, only $2N = 20$ of them is separable by $\mathcal{H}$.

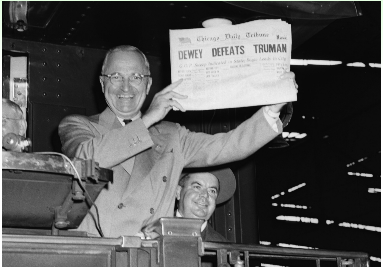
Presidential Story

- 1948 US President election: Truman versus Dewey
- a newspaper phone-poll of how people **voted**,
  and set the title '**Dewey Defeats Truman**' based on polling



who is this? :-)

The Big Smile Came from . . .



Truman, and yes he won

suspect of the mistake:

- editorial bug?—**no**
- bad luck of polling (δ)?—**no**

hint: phones were **expensive :-)**

Sampling Bias

If the data is sampled in a biased way, learning will produce a similarly biased outcome.

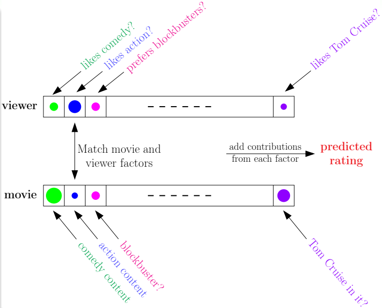
- technical explanation:
  data from $P_1(\mathbf{x}, y)$ but test under $P_2 \neq P_1$: **VC fails**
- philosophical explanation:
  study Math hard but test English: **no strong test guarantee**

'minor' VC assumption:
data and testing **both iid from** P

Sampling Bias in Learning

A True Personal Story

- Netflix competition for movie recommender system: **10% improvement = 1M US dollars**
- formed $\mathcal{D}_{\text{val}}$, in my **first shot**, $E_{\text{val}}(g)$ showed 13% improvement
- **why am I still teaching here? :-)**



validation: random examples within $\mathcal{D}$;
test: 'last' user records 'after' $\mathcal{D}$

Dealing with Sampling Bias

If the data is sampled in a biased way, learning will produce a similarly biased outcome.

- practical rule of thumb:
  match test scenario as much as possible
- e.g. if test: 'last' user records 'after' $\mathcal{D}$
  - training: emphasize later examples (KDDCup 2011)
  - validation: use 'late' user records

last puzzle:

danger when learning 'credit card approval' with existing bank records?

Fun Time

If the data $\mathcal{D}$ is an unbiased sample from the underlying distribution P for binary classification, which of the following subset of $\mathcal{D}$ is also an unbiased sample from P?

1. all the positive ($y_n > 0$) examples
2. half of the examples that are randomly and uniformly picked from $\mathcal{D}$ without replacement
3. half of the examples with the smallest $\|\mathbf{x}_n\|$ values
4. the largest subset that is linearly separable

Fun Time

If the data $\mathcal{D}$ is an unbiased sample from the underlying distribution P for binary classification, which of the following subset of $\mathcal{D}$ is also an unbiased sample from P?

1. all the positive ($y_n > 0$) examples
2. half of the examples that are randomly and uniformly picked from $\mathcal{D}$ without replacement
3. half of the examples with the smallest $\|\mathbf{x}_n\|$ values
4. the largest subset that is linearly separable

Reference Answer: 2

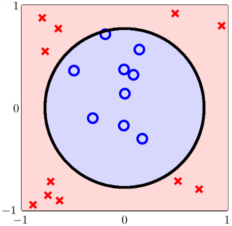
That's how we form the validation set, remember? :-)

Visual Data Snooping

Visualize $\mathcal{X} = \mathbb{R}^2$

- full $\mathbf{\Phi}_2$: $\mathbf{z} = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2)$, $d_{\text{VC}} = 6$
- or $\mathbf{z} = (1, x_1^2, x_2^2)$, $d_{\text{VC}} = 3$, **after visualizing**?
- or better $\mathbf{z} = (1, x_1^2 + x_2^2)$, $d_{\text{VC}} = 2$?
- or even better $\mathbf{z} = \left(\text{sign}(0.6 - x_1^2 - x_2^2)\right)$?

—careful about **your brain's 'model complexity'**

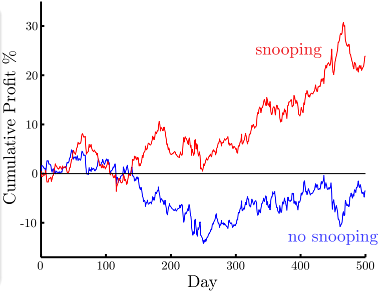


for VC-safety, $\mathbf{\Phi}$ shall be
decided **without 'snooping'** data

Data Snooping by Mere Shifting-Scaling

If a data set has affected any step in the learning process, its ability to assess the outcome has been compromised.

- 8 years of currency trading data
- first 6 years for training, last two 2 years for testing
- **x** = previous 20 days, *y* = 21th day
- snooping versus no snooping: superior profit possible



- snooping: shift-scale all values by training + testing
- no snooping: shift-scale all values by training only

Data Snooping by Data Reusing

Research Scenario

benchmark data $\mathcal{D}$

- paper 1: propose $\mathcal{H}_1$ that works well on $\mathcal{D}$
- paper 2: find room for improvement, propose $\mathcal{H}_2$ —and **publish only if better** than $\mathcal{H}_1$ on $\mathcal{D}$
- paper 3: find room for improvement, propose $\mathcal{H}_3$ —and **publish only if better** than $\mathcal{H}_2$ on $\mathcal{D}$
- ...

- if all papers from the same author in **one big paper**: bad generalization due to $d_{\text{VC}}(\cup_m \mathcal{H}_m)$
- step-wise: later author **snooped** data by reading earlier papers, bad generalization worsen by **publish only if better**

if you torture the data long enough, it will confess :-)

Dealing with Data Snooping

- truth—**very hard to avoid**, unless being extremely honest
- extremely honest: **lock your test data in safe**
- less honest: **reserve validation and use cautiously**

- be blind: avoid **making modeling decision by data**
- be suspicious: interpret research results (including your own) by proper **feeling of contamination**

one secret to winning KDDCups:

careful balance between
data-driven modeling (snooping) and
validation (no-snooping)

Fun Time

Which of the following can result in unsatisfactory test performance in machine learning?

1. data snooping
2. overfitting
3. sampling bias
4. all of the above

Fun Time

Which of the following can result in unsatisfactory test performance in machine learning?

1. data snooping
2. overfitting
3. sampling bias
4. all of the above

Reference Answer: ④

A professional like you should be aware of those! :-)

Three Related Fields

Power of Three

Data Mining

- use **(huge)** data to **find property** that is interesting
- difficult to distinguish ML and DM in reality

Artificial Intelligence

- compute something that shows **intelligent behavior**
- ML is one possible route to realize AI

Statistics

- use data to **make inference** about an unknown process
- statistics contains many useful tools for ML

Three Theoretical Bounds

Power of Three

Hoeffding

$$P[\text{BAD}] \leq 2\exp(-2\epsilon^2 N)$$

- **one** hypothesis
- useful for **verifying/testing**

Multi-Bin Hoeffding

$$P[\text{BAD}] \leq 2M\exp(-2\epsilon^2 N)$$

- M hypotheses
- useful for **validation**

VC

$$P[\text{BAD}] \leq 4m_{\mathcal{H}}(2N)\exp(\ldots)$$

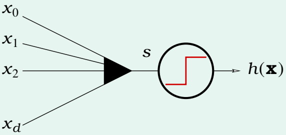
- all $\mathcal{H}$
- useful for **training**

Three Linear Models

Power of Three

PLA/pocket

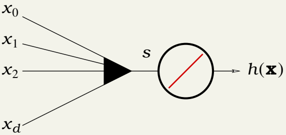
$h(\mathbf{x}) = \text{sign}(s)$



plausible err = 0/1
(small flipping noise)
minimize **specially**

linear regression

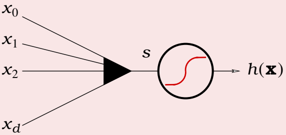
$h(\mathbf{x}) = s$



friendly err = squared
(easy to minimize)
minimize **analytically**

logistic regression

$h(\mathbf{x}) = \theta(s)$



plausible err = CE
(maximum likelihood)
minimize **iteratively**

Three Key Tools

Power of Three

Feature Transform

$$E_{in}(\mathbf{w}) \rightarrow E_{in}(\tilde{\mathbf{w}})$$
$$d_{VC}(\mathcal{H}) \rightarrow d_{VC}(\mathcal{H}_{\mathbf{\Phi}})$$

- by using **more complicated Φ**
- **lower E_{in}**
- higher d_{VC}

Regularization

$$E_{in}(\mathbf{w}) \rightarrow E_{in}(\mathbf{w}_{REG})$$
$$d_{VC}(\mathcal{H}) \rightarrow d_{EFF}(\mathcal{H}, \mathcal{A})$$

- by augmenting **regularizer Ω**
- **lower d_{EFF}**
- higher E_{in}

Validation

$$E_{in}(h) \rightarrow E_{val}(h)$$
$$\mathcal{H} \rightarrow \{g_1^-, \ldots, g_M^-\}$$

- by reserving K examples as $\mathcal{D}_{val}$
- **fewer choices**
- fewer examples

Three Learning Principles

Power of Three

Occam's Razer	Sampling Bias	Data Snooping
simple is good	class matches exam	honesty is best policy

Three Future Directions

Power of Three

More Transform | **More Regularization** | **Less Label**

bagging · decision tree · support vector machine · **neural network** · *kernel*

AdaBoost · aggregation · *sparsity* · autoencoder · **coordinate descent**

dual · uniform blending · deep learning · nearest neighbor · decision stump

kernel LogReg · large-margin · *prototype* · quadratic programming · **SVR**

GBDT · **PCA** · `random forest` · *matrix factorization* · **Gaussian kernel**

soft-margin · *k-means* · OOB error · **RBF network** · probabilistic SVM

ready for the **jungle**!

Fun Time

What are the magic numbers that repeatedly appear in this class?

1. 3
2. 1126
3. both 3 and 1126
4. neither 3 nor 1126

Fun Time

What are the magic numbers that repeatedly appear in this class?

1. 3
2. 1126
3. both 3 and 1126
4. neither 3 nor 1126

Reference Answer: ③

3 as illustrated, and **you may recall** 1126 **somewhere :-)**

Summary

1. When Can Machines Learn?
2. Why Can Machines Learn?
3. How Can Machines Learn?
4. How Can Machines Learn **Better**?

Lecture 15: Validation

Lecture 16: Three Learning Principles

- Occam's Razor
  simple, simple, simple!
- Sampling Bias
  match test scenario as much as possible
- Data Snooping
  any use of data is 'contamination'
- Power of Three
  relatives, bounds, models, tools, principles

- **next: ready for jungle!**