Machine Learning

CSE427

Mahbub Majumdar
Typeset by:
Syeda Ramisa Fariha

BRAC University
66 Mohakhali
Dhaka, Bangladesh

July 30, 2018

Superhumungous Thanks

These slides were typeset by Syeda Ramisa Fariha.

Without her tremendous dedication, these slides would not exist.

NUL

- So far we let *m* depend on ϵ, δ
- But m was independent of *D* and *h*
- Learnable classes limited to finite VC dimension classes
- Now consider a more general (but weaker) framework
- Let *m* depend on the hypothesis (Nonuniform Learnability)
- Then later let *m* depend on the distribution *D* (Consistency)

NUL

- NUL is a strict relaxation of agnostic PAC
- Sufficient condition for NUL is
$$X = \text{countable union of hypothesis classes } X_n$$
Each X_n is uniformly convergent
- Algorithm implementing NUL
$\Rightarrow$ Structural Risk Minimization (SRM)
- This is just like when ERM implements PAC learning

NUL

- *m* depends on *h* : $m_{\mathcal{H}}(\epsilon, \delta, h)$
- **Why does this make sense?**
- Some hypothesis might need more sample data to validate
- EG :
  $\rightarrow$ 2 pts determine a line
  $\rightarrow$ 3 pts determine a quadratic
  $\rightarrow$ *nH* points determine a *n*-degree polynomial

- Thus in choosing a K-degree polynomial predictor you would expect to need more points as K goes up
- We say that a hypothesis h is (ϵ, δ) competitive with respect to h', y with probability more than $1 - \delta$ that

$$L_D(h) \leq L_D(h') + \epsilon$$

- In PAC, APAC, competitiveness not very useful. This is because looking for hypothesis with absolute low/minimum risk

NUL

Definition : NUL

$\mathcal{H}$ is uniformly learnable if there exists a learning algorithm A and $m_{\mathcal{H}}^{NUL}(\epsilon, \delta, h)$ such that for every ϵ, δ, h if

$$m \geq m_{\mathcal{H}}^{NUL}(\epsilon, \delta, h)$$

then for every distribution D with probability at least $1 - \delta$ over the choice of $S \sim D^m$ that

$$L_D(A(S)) \leq L_D(h) + \epsilon$$

Note :

If $\mathcal{H}$ is APAC then it is also NUL

Characterizing NUL

- For APAC $\rightarrow$ finite VC dimension implied APAC
- For NUL, have the following

Theorem :

A hypothesis class of binary classifiers is NUL if and only if it is a countable union of APAC hypothesis classes

- Countable union means we can label the individual hypothesis classes using a "*Counting*" index n

Characterizing NUL

- This means

$$\mathcal{H} = \bigcup_n \mathcal{H}_n$$

- Proof of the theorem above relies on:

Theorem : **

Let $\mathcal{H}$ be a hypothesis class that can be written as a countable union of hypothesis classes $\mathcal{H} = \bigcup_n \mathcal{H}_n$, where each $\mathcal{H}_n$ is uniformly convergent. Then $\mathcal{H}$ is NUL.

Characterizing NUL

- This theorem generalizes the previous result of $UC \rightarrow APAC$ to nul
- Now prove the first theorem

Proof:

- Assume $\mathcal{H} = \bigcup\limits_{n} \mathcal{H}_n$
- Each $\mathcal{H}_n$ is UC by *Theorem *** , $\mathcal{H}$ is NUL
  Now prove the other way
- Assume $\mathcal{H}$ is NUL using some algorithm A

Characterizing NUL

Proof:

- Let,

$$\mathcal{H}_n = \left\{ h \in \mathcal{H} \mid m_{\mathcal{H}}^{NUL}\left(\frac{1}{8}, \frac{1}{7}, h\right) \le n \right\}$$

- Clearly $\mathcal{H} = \bigcup_n \mathcal{H}_n$
- Using the definition of $m_{\mathcal{H}}^{NUL}$ we know that for any distribution D, with the probability of at least $1 - \delta = \frac{6}{7}$ over $S \sim D^m \Rightarrow L_D(A(S)) \le \frac{1}{8} + L_D(h)$
- Since this is true for each class $\mathcal{H}_n$, each $\mathcal{H}_n$ is APAC

Characterizing NUL

Example : NUL is a strict relaxation of APAC

- $\mathcal{X} = \mathbb{R}$
- $\mathcal{H}_n =$ class of nth degree polynomials
- Binary Classifiers
$$h(\chi) = sign(p(\chi))$$
- $\mathcal{H} = \bigcup_n \mathcal{H}_n$
- VC dim $(\mathcal{H}_n) = n + 1 = d_n$ for every set C, with VC dim $= d$, there is a set with higher VC dim
- VC dim $(\mathcal{H}) = \infty$
- Thus $\mathcal{H}$ is not PAC/APAC
- But $\mathcal{H}$ is NUL

Characterizing NUL

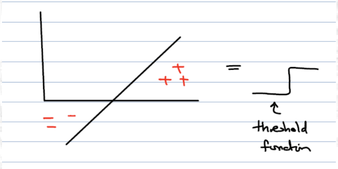


Figure: Linear Classifier

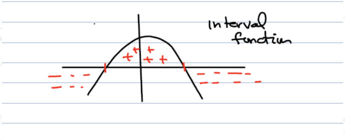


Figure: Quadratic Classifier

Standard Risk Minimization (SRM)

- So far encoded prior knowledge with choice of $\mathcal{H}$
- A more powerful way is to specify preferences over $\mathcal{H}$
- In SRM, assume
  - $\mathcal{H} = \bigcup_n \mathcal{H}_n$
  - Specify a weight factor $w : \mathbb{N} \to [0, 1]$ and $\sum_{n=1}^{\infty} w(n) \leq 1$
  - higher weight $\Rightarrow$ stronger preference

Standard Risk Minimization (SRM)

- one example of a weighting scheme is **Minimum Descriptive length (MDL)**

Define ϵ_n :

$$\epsilon_n(m, \delta) = min\left(\{\epsilon \mid m^{UC}_{\mathcal{H}_n}(\epsilon, \delta) \leq m\}\right)$$

Here,

$$m = \text{Sample size}$$
$$\delta = \text{Confidence}$$
$$UC = \text{Uniform Convergence}$$

  - ϵ_n is the most accuracy you can get by sampling up to m data points

Standard Risk Minimization (SRM)

- From the definition of UC, for every ϵ, δ and $prob \geq 1 - \delta$, we find that
$$\forall h \in \mathcal{H}_n, \mid L_D(h) - L_S(h) \mid \leq \epsilon_n(m, \delta)$$
- If all $\mathcal{H}$ have equal height
$$w(n) = \frac{1}{N} \text{ (total number of classes } = N)$$
- If the highest weight for low n (for example, low degree polynomials) choice is
$$w(n) = \frac{6}{\pi^2 n^2}$$

Note :

$$\sum_{n=1}^{\infty} w(h) = \frac{6}{\pi^2} \sum \frac{1}{n^2} = 1$$

Standard Risk Minimization (SRM)

- The SRM is a bound minimization approach
- It minimizes a certain upper bound on *The true risk*
- The bound in tries to minimize is the following:

Theorem :

Let $w(n)$ be a weight function with $w(n) \leq 1$. Suppose $\mathcal{H} = \underset{n}{\cup} \mathcal{H}_n$, and each $\mathcal{H}$ is UC, with sample complexity $m^{UC}_{\mathcal{H}_n}$. Let ϵ_n be defined as

$$\epsilon_n(m, \delta) = min\left(\{\epsilon \mid m^{UC}_{\mathcal{H}_n}(\epsilon, \delta) \leq m\}\right)$$

Standard Risk Minimization (SRM)

Then for every ϵ, D with probability $\geq 1-\delta$ over the choice of $S \sim D^m$, the following bound holds simultaneously for every n and $\mathcal{H}_n$

$$|L_D(h) - L_S(h)| \leq \epsilon_n(m, w(n) \cdot \delta)$$

Therefore for every δ and D with probability $\geq 1-\delta$, it holds that

$$\forall h \in \mathcal{H}, L_D(h) \leq L_S(h) + \min_{n: h \in \mathcal{H}_n} \epsilon_n(m, w(n) \cdot \delta)$$

[pick the smallest ϵ_n]

Standard Risk Minimization (SRM)

Proof :

Define : $\delta_n = w(n) \cdot \delta$

- using *UC* for each n with *probability* $\geq 1 - \delta$ and a fixed n,

$$D^m(S \mid \forall h \in \mathcal{H}_n, |L_D(h) - L_S(h)| \leq \epsilon_n) \geq 1 - \delta_n$$

$$\Rightarrow D^m(S \mid \exists h \in \mathcal{H}_n, |L_D(h) - L_S(h)| > \epsilon_n) < \delta_n$$

$$\cup \; D^m(S \mid \exists h \in \mathcal{H}_n, |L_D(h) - L_S(h)| > \epsilon_n)$$

$$\leq \delta_1 + \delta_2 ..... = \sum \delta_n = \delta \sum w(n) \leq \delta$$

Standard Risk Minimization (SRM)

- Now apply the Union bound over all n = 1,2..... so that this holds for every n

$$\forall h \in \mathcal{H}_n, \mid L_D(h) - L_S(h) \mid \leq \epsilon_n(m, \delta_n)$$

- Since

$$1 - \sum_n \delta_n = 1 - \delta \sum w(n)$$

$$\geq 1 - \delta$$

have that with probability $\geq 1 - \delta$

$$|L_D(h) - L_S(h)| \leq \epsilon_n(m, w(n) \cdot \delta)$$

Standard Risk Minimization (SRM)

- Now define

$$n(h) = \min(\{n \mid h \in \mathcal{H}_n\})$$

- $n(h)$ is the smallest n for which h is in a Subclass $\mathcal{H}_n$
  Thus $\min_{n(h)} \epsilon_n(m, \delta_n)$ is the smallest n that gives error ϵ_n
- Then we can rewrite

$$\forall h \in \mathcal{H}, L_D(h) \leq L_S(h) + \min_{n:h \in \mathcal{H}_n} \epsilon_n(m, w(n) \cdot \delta)$$

  as

$$L_D(h) \leq L_S(h) + \epsilon_n(h)(m, w(n(h)) \cdot \delta)$$

- This gives rise to the *SRM* paradigm

Standard Risk Minimization (SRM)

Structural Risk Minimization (SRM)

prior knowledge:
$\mathcal{H} = \bigcup_n \mathcal{H}_n$ where $\mathcal{H}_n$ has uniform convergence with $m^{\mathrm{UC}}_{\mathcal{H}_n}$
$w : \mathbb{N} \rightarrow [0,1]$ where $\sum_n w(n) \leq 1$
define: ϵ_n as in Equation (7.1); $n(h)$ as in Equation (7.4)
input: training set $S \sim \mathcal{D}^m$, confidence δ
output: $h \in \mathrm{argmin}_{h \in \mathcal{H}} \left[L_S(h) + \epsilon_{n(h)}(m, w(n(h)) \cdot \delta) \right]$

Figure: SRM

- In SRM we care about
  1. Sample loss $L_S(h)$
  2. Size of $\epsilon_n(h)$

Standard Risk Minimization (SRM)

- Since ϵ_n increases with n
  $\Rightarrow$ tradeoff between estimation error $\downarrow$ and $\epsilon_n(h)(m, w(n(h)) \cdot \delta)$
- Now we can show that SRM can be used for NUL problems ....

Theorem :

Let $\mathcal{H}$ be a hypothesis class such that $\mathcal{H} = \cup\, \mathcal{H}_n$. Here $\mathcal{H}_n$ is *UC* with saple complexity $m^{UC}_{\mathcal{H}_n}$.
Let $0 \leq w(n) \leq 1$ be such that,

$$w(n) = \frac{6}{n^2 \pi^2}$$

Standard Risk Minimization (SRM)

Then $\mathcal{H}$ is non-uniformly learnable using the SRM role with

$$m_{\mathcal{H}}^{NUL}(\epsilon, \delta, h) \leq m_{\mathcal{H}_{n(h)}}^{UC}\left(\frac{\epsilon}{2}, \frac{6}{(\pi n(h))^2} \cdot \delta\right)$$

Proof :

- $A(s)$ is SRM algorithm with respect to weight function w
- For every $h \in \mathcal{H}$ and $\epsilon, \delta, m_{\mathcal{H}_{n(h)}}^{UC}(\epsilon, w(n(h)) \cdot \delta) \leq m$
- Now use Theorem

$$\forall h \in \mathcal{H}, L_D(h) \leq L_S(h) + \min_{n:h \in \mathcal{H}_n} \epsilon_n(m, w(n) \cdot \delta)$$

with the fact that $\sum_n w(n) = 1$

Standard Risk Minimization (SRM)

- Then with probability of at least $1 - \delta$

$$L_D(h') \leq L_S(h') + \epsilon_{n(h')}(m, w(n(h')) \cdot \delta)$$

- This also holds for the $A(S)$ returned by the SRM rule
- By definition of SRM

$$L_D(A(S)) \leq \min_{h' \in \mathcal{H}} L_S(h') + \epsilon_{n(h')}(m, w(n(h')) \cdot \delta)$$

$$\leq L_S(h) + \epsilon_{n(h)}(m, w(n(h)) \cdot \delta)$$

- Here h is the minimum of $L_S(h') + \epsilon_{n(h')}(m, w(n(h')) \cdot \delta)$

Standard Risk Minimization (SRM)

- Consider

$$m^{UC}_{\mathcal{H}_{n(h)}}\left(\frac{\epsilon}{2}, \frac{6}{(\pi n(h))^2}\cdot\delta\right)$$

if

$$m \geq m^{UC}_{\mathcal{H}_{n(h)}}\left(\frac{\epsilon}{2}, \frac{6}{(\pi n(h))^2}\cdot\delta\right)$$

then plugging this higher m into $\epsilon_{n(h)}$

$$\epsilon_{n(h)}(m, w(n(h))\cdot\delta) \leq \frac{\epsilon}{2}$$

Standard Risk Minimization (SRM)

- Since each $\mathcal{H}$ is UC

$$L_D(h) \leq L_S(h) + \frac{\epsilon}{2}$$

- The

$$L_D A(S) \leq L_S(A(S)) + \frac{\epsilon}{2} \text{ (by UC)}$$

$$\leq L_S(h') + \frac{\epsilon}{2} \text{ (by SRM)}$$

$$\leq L_D(h') + \frac{\epsilon}{2} + \frac{\epsilon}{2} \text{ (by UC)}$$

Standard Risk Minimization (SRM)

- $\Rightarrow L_D(A(S)) \leq L_D(h) + \epsilon$
  Thus $\mathcal{H}$ is NUL

Remarks :

- One can show that for any infinite domain set the hypothesis class of a binary predictor is not equal to a countable union

$$\mathcal{H} \neq \bigcup_n \text{(classes of finite VC dimension)}$$

- Thus NFL holds for NUL also
  $\Rightarrow \infty$ domain $\Rightarrow$ no universal learner $+ \exists$ perfect adversary

Standard Risk Minimization (SRM)

- More samples are needed in NUL than APAC
  $\Rightarrow$ NUL must search over all $\mathcal{H}_n$
  $\Rightarrow$ APAC, search over one $\mathcal{H}_n$
  (consequence of having more prior knowledge)
- Assume for all n

$$n = VCdim(\mathcal{H}_n)$$

$$m_{\mathcal{H}_n}(\epsilon, \delta) = C\ \frac{n + \log \frac{1}{\delta}}{\epsilon^2}$$

Standard Risk Minimization (SRM)

- one can then show

$$m_{\mathcal{H}}^{NUL}(\epsilon, \delta, h) - m_{\mathcal{H}_n}^{UC}\left(\frac{\epsilon}{2}, \delta\right) \leq 4C \cdot \frac{2\log(2n)}{\epsilon^2}$$

$\Rightarrow$ Thus NUL needs many more samples
$\Rightarrow$ grows logarithmically with V dim of $\mathcal{H}_n$
$\Rightarrow$ grows as $\frac{1}{\epsilon^2}$

Minimum Description Length and Occam's Razor

- NUL is actually a practical and widely used learning paradigm
- But what is $w(n)$?
  How to express preferences with respect to different hypotheses?
- Consider the case of singleton classes

$$\mathcal{H}_1 = \left\{ h_z \mid z \in \mathcal{X}, h_z(\chi) = \left\{ \begin{array}{l} 1 \text{ if } \chi = z \\ 0 \text{ otherwise} \end{array} \right\} \right\}$$

$$\mathcal{H}_2 = \left\{ h_{z_1, z_2} \mid z_1, z_2 \in \mathcal{X}, h_z = \begin{array}{l} \text{zero unless} \\ \chi = z_1 \text{ or } \chi = z_2 \end{array} \right\}$$

Minimum Description Length and Occam's Razor

- Suppose we don't have a true predictor
  $\Rightarrow$ APAC predictor needed
- Use Hoeffding's inequality or
  Find them of Statistical Learnining

$$m^{UC}(\epsilon, \delta) = \frac{\log\left(\frac{2}{\delta}\right)}{\epsilon^2}$$

- Now invert this to

$$m = \frac{\log \frac{2}{\delta}}{\epsilon_n^2}$$

$$\Rightarrow \epsilon_n(m, \delta) = \sqrt{\frac{\frac{2}{\delta}}{2m}}$$

$$\Rightarrow \delta \Rightarrow \delta \cdot w(n)$$

- Thus

  $\epsilon_{n(h)}(m, w(n(h)) \cdot \delta)$

  $= \sqrt{\frac{\log \frac{2}{w \cdot \delta}}{2m}}$

  $= \sqrt{\frac{-\log w(n) + \log \frac{2}{\delta}}{2m}}$

- The SRM rule then becomes

$$h \in \underset{h \in \mathcal{H}}{\text{argmin}} \left(L_S(h) + \sqrt{\frac{-\log w(h) + \log \frac{2}{\delta}}{2m}} \right)$$

- A convenient way to assign weights is to use a complexity measure for the hypotheses

Minimum Description Length and Occam's Razor

- Consider the bit-length required to describe a hypothesis
  $\Rightarrow$ the more complicated the *h* , the longer the description
- Thus fix a description language
  $\Rightarrow$ can be English/Bangla/Python/Math formulas/etc...
- A description consists of a finite string of symbols from a fixed alphabet
- Fix a finite set $\sum$, of symbols.
  Call them *"characters"*
- For example, $\sum = \{0, 1\}$
- $\sigma =$ string of symbols from $\sum$
  Suppose
  $\sigma = (0, 1, 1, 1, 1)$
  $|\sigma| = 5$ (length of string)

Minimum Description Length and Occam's Razor

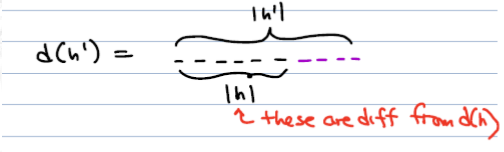
- The set of all *finite* length string is denoted $\sum^*$
- A descriptive language for $\mathcal{H}$ is
$$d : \mathcal{H} \to \sum^*$$
maps every $h \in \mathcal{H}$ to a string $d(h)$
- $d(h) \equiv$ description of h
$|h| =$ length of h
- To make sure that each $d(h)$ uniquely describes a string h
$\Rightarrow$ require the description language to be **Prefix-free**

Minimum Description Length and Occam's Razor

- Prefix-free means if

$$h' \neq h$$

then the first $|h|$ symbols of $d(h')$ cannot be $d(h)$ or vice -versa



- Prefix-free strings can be used to weight different hypotheses

Minimum Description Length and Occam's Razor

Kraft Inequality :

If $S \subseteq \{0,1\}^*$ is a prefix-free set of strings then

$$\sum_{\sigma \in S} \frac{1}{2^{|\sigma|}} \leq 1$$

Proof :

- Suppose S is a binary prefix-free set of string S
- given a string σ, what is the probability of randomly choosing σ?
  $\Rightarrow$ Suppose $P(0) = P(1) = \frac{1}{2}$
  $\Rightarrow$ because S is prefix-free once the first $|\sigma|$ flips match σ,
  we know that this sequence of coin flips uniquely corresponds to σ

- Thus,

$$P(\sigma) = \frac{1}{2^{|\sigma|}}$$

  for every $\sigma \in S$
- Since the $P(\sigma)$ add up to 1, proof is finished

- There we can weight h as

$$w(h) = \frac{1}{2^{|h|}}$$

- Inserting this weight into the

$$\sqrt{-\log w(h) + \log \frac{2}{\delta}}$$

Minimum Description Length and Occam's Razor

Theorem :

Let $\mathcal{H}$ be a hypothesis class, and let d be a descriptive language

$$d : \mathcal{H} \to \{0,1\}^{* \leftarrow \text{finite}}$$

that is prefix-free.

Then for every m, S, D with probability $\geq 1-\delta$ over $S \sim D^m$, we have

$$\forall h \in \mathcal{H} \; L_D(h) \leq L_S(h) + \sqrt{\frac{|h| + \ln \frac{2}{\delta}}{2m}}$$

where $|h|$ is the length of $d(h)$

Minimum Description Length and Occam's Razor

Proof :

use $\forall h \in \mathcal{H}, L_D(h) \leq L_S(h) + \min_{n:h\in\mathcal{H}_n} \epsilon_n(m, w(n)\cdot\delta)$ and let $w(h) = 2^{-|h|}$, use

$$\epsilon_n(m,\delta) = \sqrt{\frac{\ln\frac{2}{\delta}}{2m}}$$

and

$$\ln 2^{|h|} = |h| \ln 2 < |h|$$

Minimum Description Length and Occam's Razor

- From this we can construct a new learning paradigm
  1. Find a training set S
  2. Search for a hypothesis h
     Minimizing the bound

$$L_S(h) + \sqrt{\frac{|h| + \ln \frac{2}{\delta}}{2m}}$$

- This trades low empirical risk $L_S(h)$ for low description length

Minimum Description Length (MDL)

prior knowledge:
$\mathcal{H}$ is a countable hypothesis class
$\mathcal{H}$ is described by a prefix-free language over $\{0,1\}$
For every $h \in \mathcal{H}$, $|h|$ is the length of the representation of h
input: A training set $S \sim \mathcal{D}^m$, confidence δ
output: $h \in \operatorname{argmin}_{h \in \mathcal{H}} \left[L_S(h) + \sqrt{\frac{|h| + \ln(2/\delta)}{2m}} \right]$

Figure: MDL

Minimum Description Length and Occam's Razor

- For example $|h|$ could be the length of a program (in binary).

  Assume compiler stops if program 1 *is a header for program* 2

Occam's Razor

- Prefer hypotheses with shorter lengths : less complex
- Two hypotheses with equal sample error
  $\Rightarrow$ prefer the one with shorter *MDL* length
- Choice of language is one way of implementing prior knowledge
  $\Rightarrow$ For example use the more complex language of relativity to describe physics instead of Galilean/Newtonian relativity

- Let m depend on underlying distribution D

Definition : (Consistency)

Let Z be a domain set.
Let P a set of distributions over Z.
Let $\mathcal{H}$ be a hypothesis class.
A learning role is <u>*consistent*</u> with respect to $\mathcal{H}$ and P, if $\exists$ a $m(\epsilon, \delta, h, D)$ such that for every $(\epsilon, \delta) \in (0,1)$, every $h \in \mathcal{H}$, every $D \in P$ that if $m \geq m_{\mathcal{H}}^{NUL}(\epsilon, \delta, h, D)$ then with probability $\geq 1 - \delta$ that

$$L_D(A(S)) \leq L_D(h) + \epsilon$$

If P is the set of all distributions, we say that A is universally consistent with respect to $\mathcal{H}$

Other Notions of Learnability - Consistency

- Consistency is a strict relaxation of NUL
- If $A(S)$ is NUL with respect to $\mathcal{H}$, it is universally consistent with respect to $\mathcal{H}$
- Consistency is desirable, but not powerful
- Suppose $A_1(S)$ and $A_2(S)$ are NUL
  But $A_1(S)$ is consistent, $A_2(S)$ is not.
- East to make $A_2(S)$ consistent
  $\Rightarrow$ Just memorize the sample data and output the most frequent y value of x, y
  x appears in the sample
  $\Rightarrow$ This memorization also can be shown to be consistent

Other Notions of Learnability - Consistency

- Same subtlely regarding the NFL theorem for consistent algorithms

  $\rightarrow$ in APAC/NUL fix training set size m, then find a distribution and labeling the function for this training set size

  $\rightarrow$ in Consistency guarantees, first fix the distribution and the labeling function and then find a training set size that works for learning this part distribution and labeling function