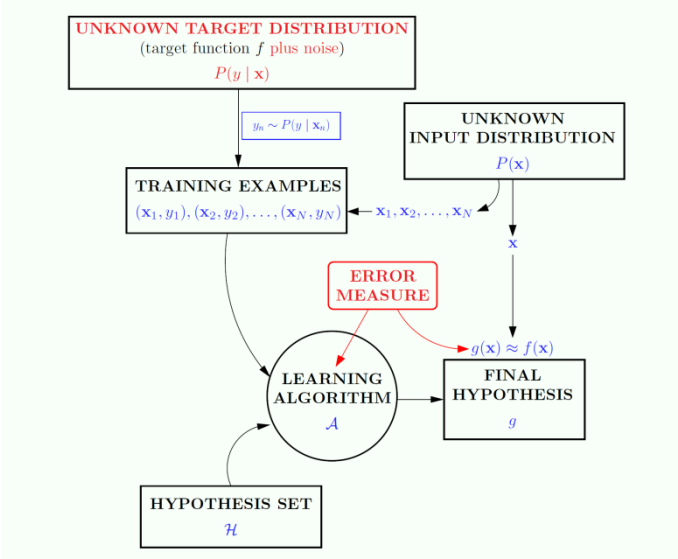
Full Summary

Theory



$$\mathbb{P}(\mathcal{D}\text{: } |E_{\text{out}}(h) - E_{\text{in}}(h)| > \epsilon) \leq 2e^{-2\epsilon^2 N}$$

Uniform Convergence

for any $\epsilon > 0$ *and* $\forall g \in \mathcal{H}$

$$\mathbb{P}(\mathcal{D}\text{: } |\mathrm{E}_{in}(g) - \mathrm{E}_{out}(g)| > \epsilon) < 2|\mathcal{H}|e^{-2\epsilon^2 N}$$

With probability at least $1 - \delta$,

$$E_{out}(g) \leq E_{in}(g) + \sqrt{\frac{1}{2N}\ln\frac{2|\mathcal{H}|}{\delta}}$$

Models

Classification: Perceptron

$$h(\boldsymbol{x}) = sign(\boldsymbol{w}^T\boldsymbol{x})$$

Error: 0/1

Algorithm: PLA / Pocket

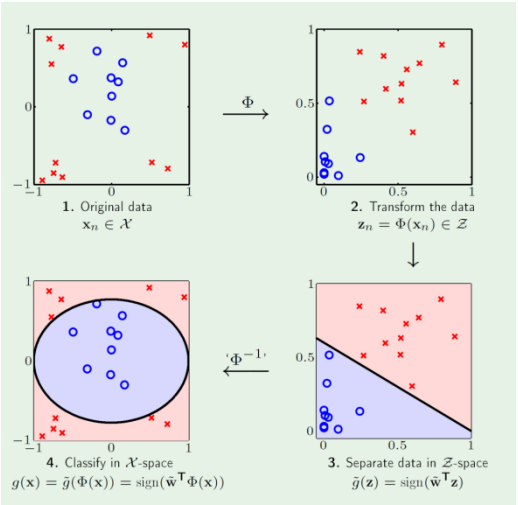
Regression: Lineal

$$h(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x}$$

Error: quadratic

Algorithm: Linear system using SVD

Adding more features



ERM theory of Generalization:

The Vapnik-Chervonenkis Dimension

Learning is feasible for finite $\mathcal{H}$

- Let's assume a probability distribution P(x) on $\mathcal{X}$
- Let´s $\mathcal{D}$ represents as a i.i.d sample from P(x)

- Then the ERM rule give us:

Result: With probability at least $1-\delta$, $E_{out}(g) \leq E_{in}(g) + \sqrt{\frac{1}{2N}\ln\frac{2|\mathcal{H}|}{\delta}}$

This bound does not depend on $\mathcal{X}$, $\mathbb{P}$(x), f or how g is found.

How many examples are required to guarantee uniform convergence?

Sample complexity: $N(\epsilon, \delta, \mathcal{H}) \geq \left\lceil \frac{1}{\epsilon^2}\ln\frac{2|\mathcal{H}|}{\delta} \right\rceil = \mathcal{O}(\frac{\ln|\mathcal{H}|}{\epsilon^2})$

$\mathcal{H}$- infinite: the discretization trick

- The discretization trick allows us to have an estimation for the sample complexity inequality on infinite classes

- Example:
  - A modern computer use a 64 bit representation for each scalar.
  - Whether we have to fit functions with only one free parameter, we only have 2^{64} possible values
  - The size of $\mathcal{H}$ now is 2^{64}
  - In the case of d free parameters the size will be 2^{64d}
  - Applying the inequality for finite classes, we obtain a bound the for the sample complexity given by

$$m_{\mathcal{H}}(\varepsilon,\delta) \leq \left\lceil \frac{1}{\epsilon^2} \log \frac{2|\mathcal{H}|}{\delta} \right\rceil = \frac{64d + 2\log({}^{2}/_{\delta})}{\epsilon^2}$$

  - This bound allow us to get a very rough estimate of the required sample complexity in practical situations

- Is there anything better…?

What is the uniform inequality pitfall ?

- Let's remember the simple bound we use:

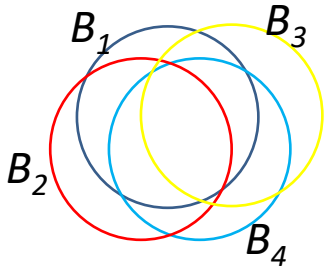
$$P\left(\bigcup_{i=1:|\mathcal{H}|} B_i\right) \leq \sum_{i=1}^{|\mathcal{H}|} P(B_i)$$

- and its consequence

$$P(D: |\mathrm{E}_{in}(g) - \mathrm{E}_{out}(g)| > \epsilon) < 2|\mathcal{H}|e^{-2\epsilon^2 N} \text{ for any } \epsilon > 0$$

- But in most of the cases, $B_i \cap B_j \neq \emptyset$ for almost all (i,j), hence

$$\bigcup_{i=1:|\mathcal{H}|} B_i = \bigcup_{j=1:|\mathcal{V}|} B_j \quad |\mathcal{V}| \leq |\mathcal{H}|$$



- This means, that counting only a few hypotheis could be sufficient!
- A better bound for the efective number of hypothesis in $\mathcal{H}$ is needed
- The Vapnik-Chevonenkis dimension is the answer !!

VC Generalization Bound

- The VC generalization bound is

$$E_{out}(h) \leq E_{in}(h) + \sqrt{\frac{8}{N}\log\frac{4((2N)^{d_{VC}}+1)}{\delta}}$$

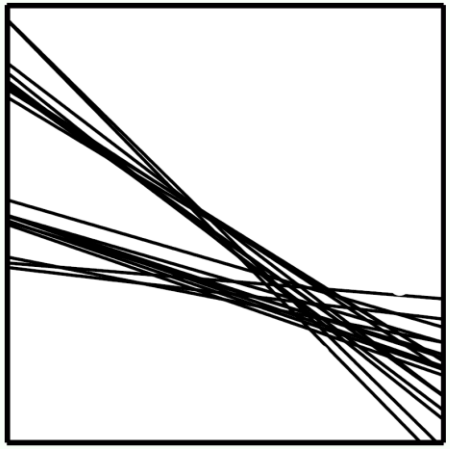
or equivalently $$E_{out}(h) \leq E_{in}(h) + \mathcal{O}\left(\sqrt{d_{VC}\frac{\log N}{N}}\right)$$

- This shows that for d_{VC} finite and N >> 0, generalization is guaranteed
- As conclusion any model can be considered either Good model or Unknown model
  - Good models: $we\ can\ obtain\ a\ good\ generalization$
  - Unknown models: d_{VC} is infinite (no answer in VC theory)

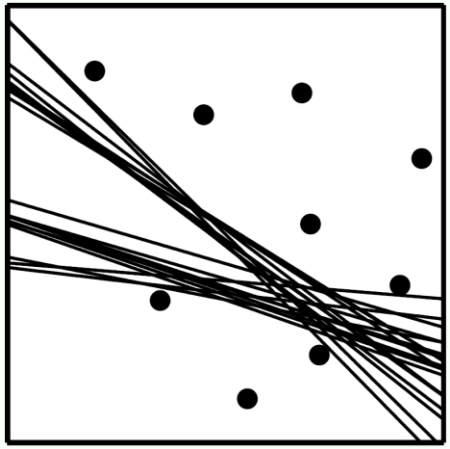
Measuring the diversity of $\mathcal{H}$

- We need a way to measure the diversity of $\mathcal{H}$
- Here we focus on binary {-1,+1} target functions and finite sample of points

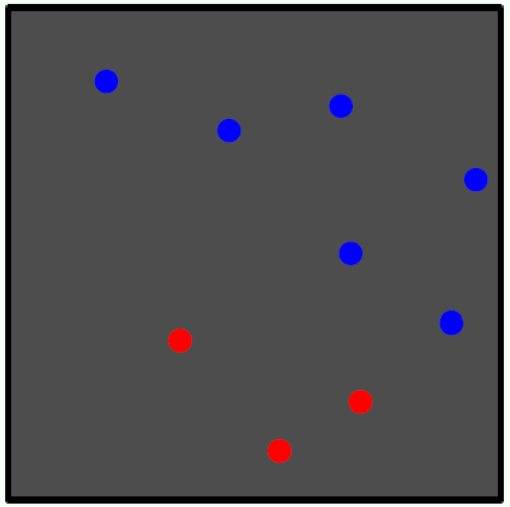
- The approach is combinatorial :
  - Consider a sample of fixed size N
  - Explore if $\mathcal{H}$ can implement ALL possible functions (labeling) on THESE N points
  - Evaluate for all N values



$\mathcal{H}$



$\mathcal{H}$ through the eyes of the $\mathcal{D}$



dichotomy

The Growth Function

That is the effective number of function in the class

- The Growth Function $m_{\mathcal{H}}$: Given a sample size N and a class $\mathcal{H}$, $m_{\mathcal{H}}$ return the **maximum number of binary patterns** generated by $\mathcal{H}$ on N points.

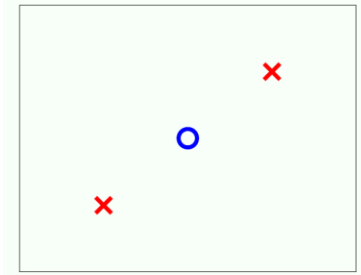
$$m_{\mathcal{H}}(N) = \max_{x_1,\cdots,x_N} |\mathcal{H}(x_1, x_2, \cdots, x_N)|$$

where | | represents number of elements in the set

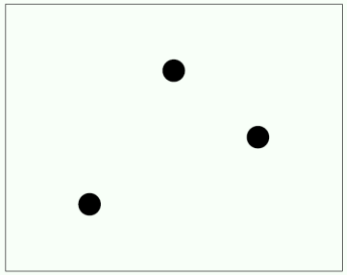
- The **maximum** is computed on all possible samples of size N
- In general $m_{\mathcal{H}}(N) \leq 2^N$
- When $m_{\mathcal{H}}(N) = 2^N$ we say that $\mathcal{H}$ shatter the set $\{x_1, x_2, \cdots, x_N\}$
- It is independent of $\mathcal{P}$, and therefore a worst-case analysis

Growth function: example-1

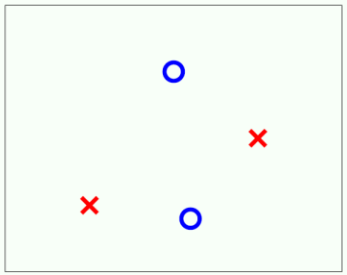
- Let be $\mathcal{H}$ the class of 2D-perceptron



Cannot implement



Can implement all 8



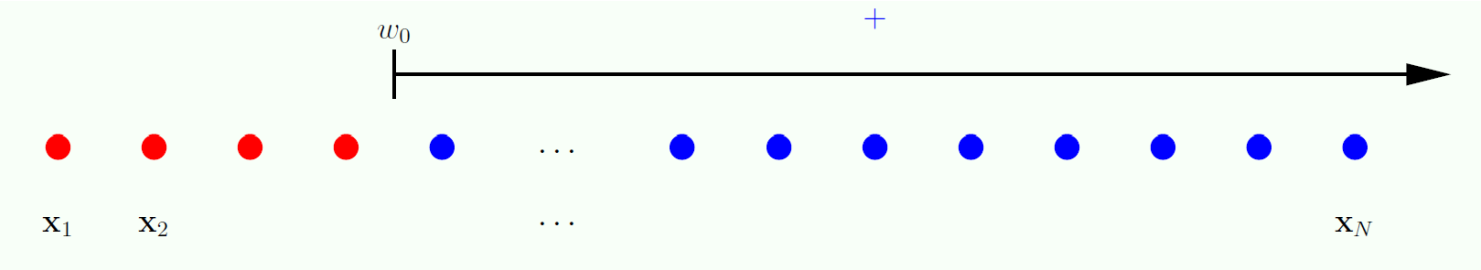
Can implement at most 14

- What is the value of $m_{\mathcal{H}}(N)$?

$$m_{\mathcal{H}}(2) = 4 = 2^2$$
$$m_{\mathcal{H}}(3) = 8 = 2^3$$
$$m_{\mathcal{H}}(4) = 14 < 2^4$$

- Let be $\mathcal{H}$ the perceptron class (binary linear predictors) and $\mathcal{X} = \mathbb{R}^3$
  - Can be shattered a sample of 2 points? , 3 points ?, 4 points ?, etc
- Can you guess any rule for points in $\mathbb{R}^k$?

Growth function: example-2

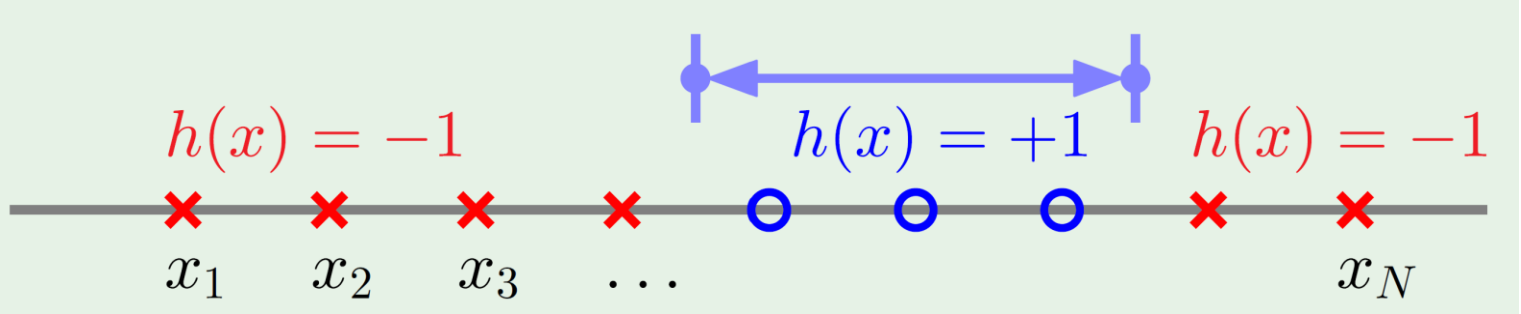


- Let be $\mathcal{H}$ the class of h: $\mathbb{R} \rightarrow \{-1,+1\}$ (Positives Rays)
  $h(x) = \text{sign}(x - w_0)$

- Let consider a sample of N points from $\mathbb{R}$.
  - Question: What is the value of $m_{\mathcal{H}}(N)$?
  - Answer: {N+1,N,N-1}, which of them ?

- What size of points can be shattered ?

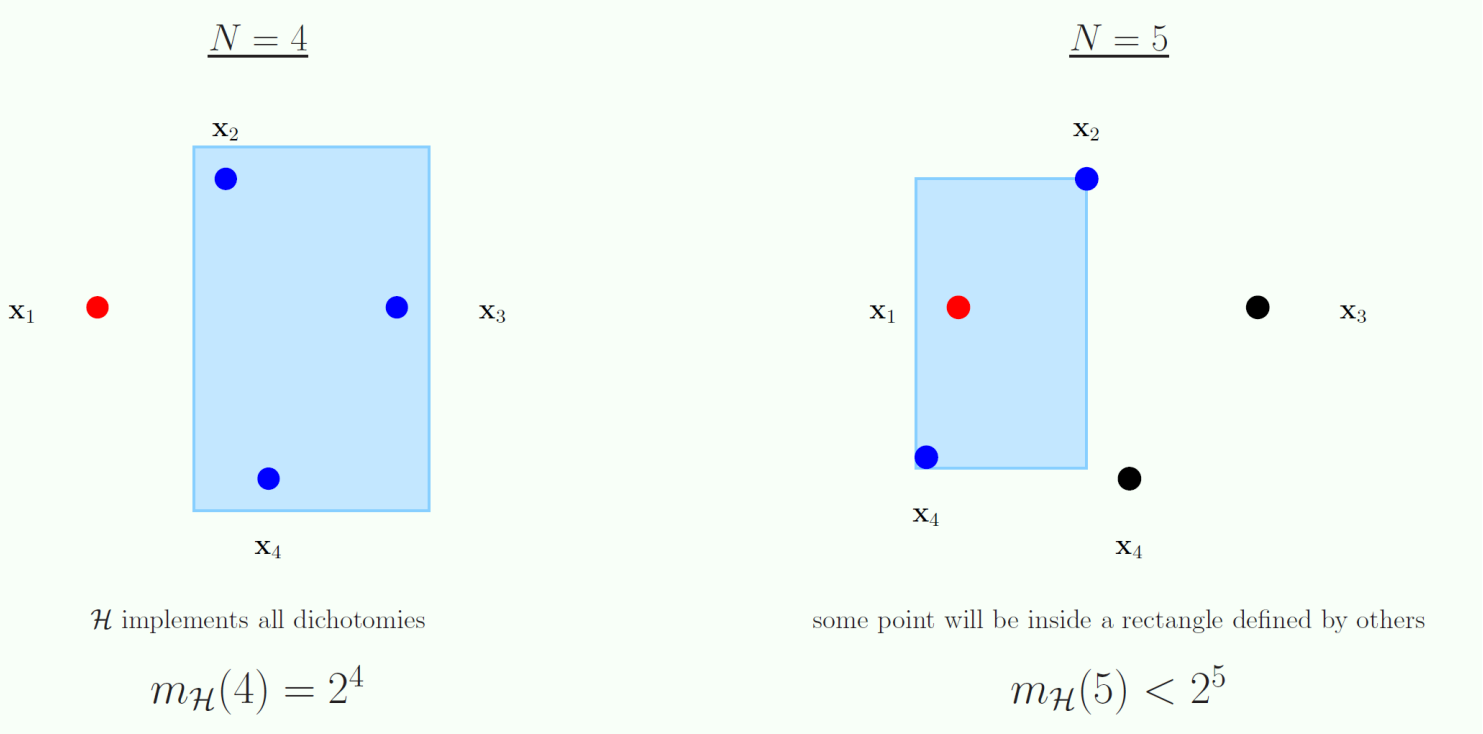
Growth function: example-3



- Let be $\mathcal{H}$ the class of functions h: $\mathbb{R} \rightarrow$ {-1,+1} (Intervals)
- $h_{a,b}(\mathrm{x}) = \begin{cases} +1 & \text{if } \mathrm{x} \in [a,b] \\ -1 & \text{if } \mathrm{x} \notin [a,b] \end{cases}$
- Let consider a sample of N points from $\mathbb{R}$.
- Now: $m_{\mathcal{H}}(N) = \binom{N+1}{2} = \frac{1}{2}N^2 + \frac{1}{2}N + 1$ Why?
- How many points can be shattered ?

Growth function: example-4

- Let $\mathcal{H}$ be the class of positive rectangles



To compute $m_{\mathcal{H}}(5)$ is NOT easy!!

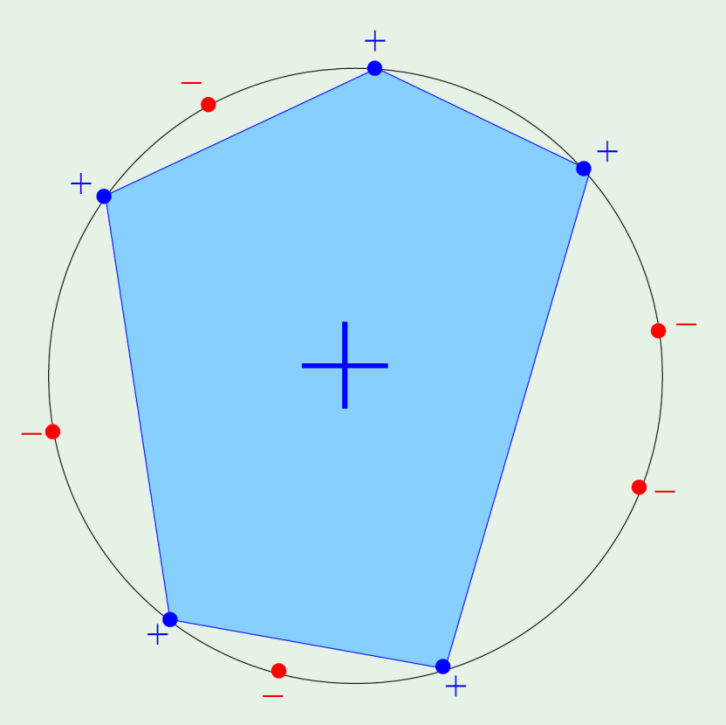
Growth function: example-5

- Let $\mathcal{H}$ be the class of convex set
- Consider the case where all point lies on a circle

$\mathcal{H}$ is set of $h: \mathbb{R}^2 \rightarrow \{-1, +1\}$

$h(\mathbf{x}) = +1$ is convex

$m_{\mathcal{H}}(N) = 2^N$



Growth Function and Generalization

- Let's have a look to the new bound we get:

$$P(\mathcal{D}: |E_{in}(h) - E_{out}(h)| > \epsilon) \leq 4m_{\mathcal{H}}(2N)e^{-\frac{N\epsilon^2}{8}}$$

$$E_{out}(h) \leq E_{in}(h) + \sqrt{\frac{8}{N}\log\frac{4m_{\mathcal{H}}(2N)}{\delta}}$$

- This expressión provides a better bound but required to compute the growth function.

- A constant upper bound on $m_{\mathcal{H}}(N)$ will solve the problem
- The new bound **is not** a direct replacement of $|\mathcal{H}|$ by $m_{\mathcal{H}}(N)$!!

Break Point

- It is not practical to try to compute $m_{\mathcal{H}}(N)$ for every hypothesis set we use, an upper bound will be sufficient.

- **Break Point Concept:**

  if for some value k, $m_{\mathcal{H}}(k) < 2^k$, then k is a **break point** for $\mathcal{H}$

  - **That is, $\mathcal{H}$ CANNOT shatter a sample of size k**

- **Examples:**
  - **Which is the break point for the 2D Perceptron? k=4**
  - **Which is the break point for the Positive Rays? k=2**
  - **Which is the break point for the Interval? k=3**
  - **Which is the break point for the Positive Rectangle? k=5**
  - **Which is the break point for the Convex Set? k=∞**

Bounding the Growth Function

- **Main Result (Vapnik&Chervonenkis, 1971)**: if $k\ is\ a\ break\ point\ for\ \mathcal{H}$, then for all N

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{k-1} \binom{N}{i}$$

The RHS **is polynomial in N** of degree k-1: $\mathcal{O}(N^{k-1})$

This result says: if $\mathcal{H}$ has a break point : $m_{\mathcal{H}}(N)$ is **polinomia**l in N

if $\mathcal{H}$ has NOT break point : $m_{\mathcal{H}}(N) = 2^N$

- Regarding the generalization bound we replace $\log m_{\mathcal{H}}(N)$ by $\mathcal{O}(k \log N)$.
  - For N >> 0 we can guarantee a good generalization since log(N)/N -> 0

- What happens at the generalization bound when $\mathcal{H}$ has NOT break point ?

Vapnik&Chervonenkis: VC-dimension:

- **Definition:** *The VC dimension of a hypothesis set* $\mathcal{H}$*, denote by* $d_{VC}(\mathcal{H})$ *or simply* d_{VC}*, is the largest value of N for which* $m_{\mathcal{H}}(N) = 2^N$*. If* $m_{\mathcal{H}}(N) = 2^N$ *for all N, then* $d_{VC} = \infty$.

- It can be proved that the main bound can be written as

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{d_{VC}} \binom{N}{i} \leq \begin{cases} N^{d_{VC}} + 1 \\ \left(\frac{eN}{d_{VC}}\right)^{d_{VC}} \end{cases}$$

Two bounds for the growth function

- The d_{VC} value measure the "**effective** " number of parameters associated with $h \in \mathcal{H}$ (Perceptron 2D, d_{VC}=3; in **linear models** $\mathbf{d_{VC}=d+1}$)

VC Generalization Bound

- Combining bounds

$$E_{out}(h) \leq E_{in}(h) + \sqrt{\frac{8}{N}\log\frac{4((2N)^{d_{VC}}+1)}{\delta}}$$

or equivalently $$E_{out}(h) \leq E_{in}(h) + \mathcal{O}\left(\sqrt{d_{VC}\frac{\log N}{N}}\right)$$

- This shows that for d_{VC} finite and N >> 0, generalization is guaranteed
- A conclusion of this results is that there are a division of models in two classes: "Good" models and "Useless" models (in terms of ERM learning)
  - "Good" models: d_{VC} is finite $we\ can\ obtain\ a\ good\ generalization$
  - "Useless" models: d_{VC} is infinite (we can not learn using the ERM rule!!)

Sample Complexity

- Remember: The sample complexity is the minimum number of training examples (N) needed to achieve a certain generalization performance
  - ε, δ *have to be fixed*
  - How fast grows N(ε, δ) indicates how much data is needed to get good generalization.

- Fix $\delta > 0$ and suppose the generalization error to be at most ε

$$\sqrt{\frac{8}{N}\ln\frac{4m_{\mathcal{H}}(2N)}{\delta}} \leq \varepsilon \Rightarrow N \geq \frac{8}{\varepsilon^2}\ln\left(\frac{4m_{\mathcal{H}}(2N)}{\delta}\right) \Rightarrow N \geq \frac{8}{\varepsilon^2}\ln\left(\frac{4\left((2N)^{d\text{VC}}+1\right)}{\delta}\right)$$

  - This is an implicit equation in N, we solve it iteratively

Sample Complexity: An Example

- Example:
  - Suppose d_{VC}=3
  - Assume $\varepsilon = 0.1$, $\delta = 0.1$. How big a data set do we need?

$$N \geq \frac{8}{0.1^2} \ln\left(\frac{4(2N)^3+4}{0.1}\right) \xrightarrow{N=1000} N \geq 21.193 \Rightarrow N \geq 30.000$$

fixed point of the equation

  - For d_{VC}=4, we get $N \geq 40.000$
  - For d_{VC}=5, we get $N \geq 50.000$
  - This suggest the bound should be proportional to d_{vc}

  - A good rule of thumb : $N > 10 \times d_{vc}$

VC means Penalty by Model Complexity

- In most practical situations the sample data set is given, so N is fixed !

- The relevant question now is what performance can we expected given this particular N

$$E_{out}(g) \leq E_{in}(g) + \sqrt{\frac{8}{N}\ln\frac{4m_{\mathcal{H}}(2N)}{\delta}} \leq E_{in}(g) + \sqrt{\frac{8}{N}\ln\frac{4\left((2N)^{d_{\mathrm{VC}}}+1\right)}{\delta}}$$

$$\Omega(N,\mathcal{H},\delta) = \sqrt{\frac{8}{N}\ln\frac{4\left((2N)^{d_{\mathrm{VC}}}+1\right)}{\delta}} = \mathcal{O}\left(\sqrt{\frac{d_{VC}\ln N - \ln\delta}{N}}\right)$$

- This term can be seen as a penalty due to the $\mathcal{H}$ complexity.

$$E_{out} \leq E_{in} + \Omega(d_{\mathrm{VC}})$$

Approximation-GeneralizationTradeoff: A new insight to choose g

VC Bound Quantifies Approximation vs Generalization

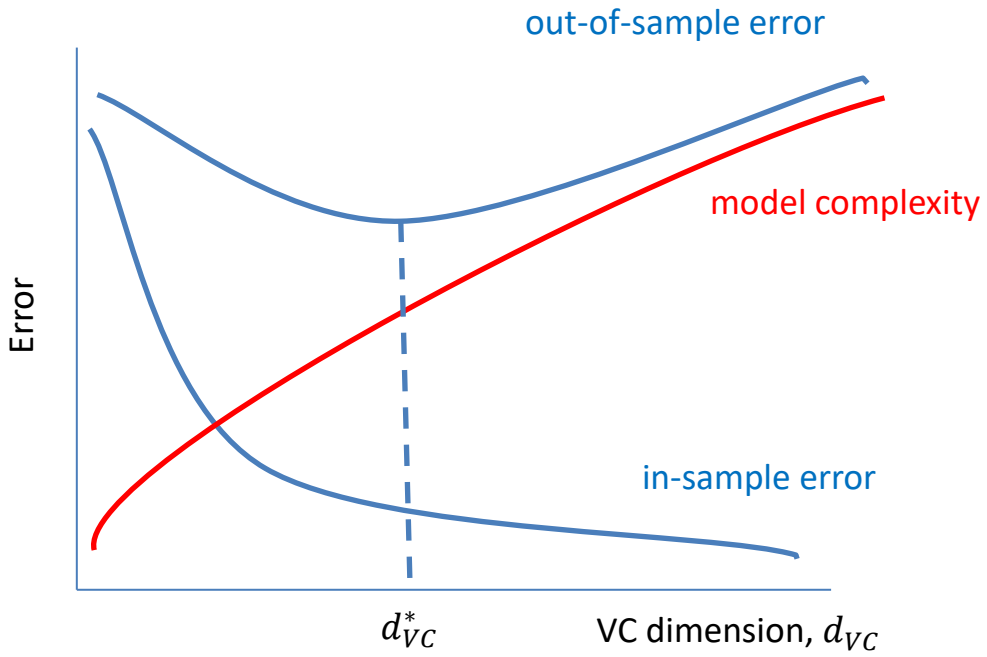
- In fact, we have a tradeoff: More complex models help E_{in} and hurt $\Omega(N, \mathcal{H}, \delta)$

- $d_{\text{vc}} \uparrow \Longrightarrow$ better chance of approximating f ($E_{in} \approx 0$).
- $d_{\text{vc}} \downarrow \Longrightarrow$ better chance of generalizing to out of sample ($E_{in} \approx E_{out}$).

$$E_{out} \leq E_{in} + \Omega(d_{\text{vc}})$$

- **VC analysis only depends on $\mathcal{H}$.**
  - Independent of f, $\mathbb{P}(\mathcal{X})$, $\mathcal{A}$ (learning algorithm)
  - Mainly applicable to classification and regression problems
  - Nevertheless, for square loss a better insight is given by the B-V tradeoff.
  - Quite loose bound

Penalty by Model Complexity



- The figure shows the fitting error vs the VC dimension
- A tradeoff, using the out-of-sample error, attains a minimum at some intermediate value d_{VC}^*
- This is a generalization to the finite case tradeoff !!

Model Complexity (d_{vc}) ↑ → E_{in} ↓ → better chance of approximating f

Model Complexity (d_{vc}) ↓ → $E_{\text{out}} - E_{\text{in}}$ ↓ → better chance of good generalization

Summary of the VC Bound

- If $d_{VC}(\mathcal{H})$ is finite $\Longleftrightarrow$ The class $\mathcal{H}$ is "PAC" learnable

- The VC bound is independent of : f, $\mathbb{P}(\mathcal{X})$, $\mathcal{A}$
  - **(Binary target function, Input distribution, Learning Algorithm)**

- The VC dimension give us a measure of the complexity of the class $\mathcal{H}$
  - The higher the complexity the bigger the training set for a fixed error

- The VC dimension of a class $\mathcal{H}$ is related to the "effective" number of free parameters of its elements.

- The VC analysis was developed for to 0-1 loss function (classification)
  - But it can be extended to real-valued loss functions (regression)

How assess our fitting ?

$$E_{\text{out}}(g) \leq E_{\text{in}}(g) + \Omega(N, \mathcal{H}, \delta)$$

is good to guide the training process BUT is useless if we want to get an accuracy forecast of E_{out}.

- In real problems a precise estimate of E_{out} is what the customer is expecting to have.

- The best formula is to challenge our trained hypothesis with absolutely new examples NEVER SEEN BEFORE. This is called a TEST SET
  - The samples of the test set MUST be i.i.d samples from the same probability distribution used in training
- Let us call the error on the test set E_{test}.
- We use E_{test} as an estimator of E_{out}

NLT- Discussion

- How does the feature transform affect to the PLA VC-bound?
- If we honestly fix the transfom before seeing the data, then $d_{VC}(\mathcal{H}_\Phi) = d_{VC}(\mathcal{H})$ at least with probability 1-δ
- What if we first try separating with lines , fail, and then use the circles?
  - This is equivalent to use a transformation where the original features are keeped and we add the square of all of them.
  - We have increased the dimension of the feature space !!
- What if we explore the data but we do not try any model?
  - Even worst !! Our mind has explored a huge hypothesis space that we must add to the real transformations dimension.
  - Inadvertently, you have decided your data is the problem and not a sample of it !!
- In classification problems if we insist in getting full separability between classes we can be compelled to use high degree transformations
  - Nevertheless, this increase dramatically the feature space dimension and the VC-dimension
- Let analyze in more detail these implications ……

Computation and Generalization

- Let denote by Φ_Q the *Q-th order polynomial transform*
  - $\Phi_4(\boldsymbol{x}) = (1, x_1, x_2, x_1^2, x_2^2, x_1x_2, x_1^3, x_2^3, x_1x_2^2, x_2x_1^2, x_1^4, x_2^4, x_1^2x_2^2, x_2^1x_1^3, x_1^1x_2^3)$

- A larger *Q* provides a larger flexibility in terms of the shape of the decision boundary but there is a price to pay.

  1. Computation is an issue because the feature transform Φ_Q maps x (the initial vector) to $d = \frac{Q(Q+3)}{2}$ dimensions, incrementing memory and computational cost.
  2. The VC-dimension can increases till $\frac{Q(Q+3)}{2} + 1$ and the VC-bound can grow significantly
     - For Q=50 the VC-dim is $\frac{Q(Q+3)}{2}$+1 =1326 instead of 3 (initial)
  3. According to the rule: " .. number of samples needed is proportional to the VC-dim", the higher the Q-value the higher (quadratic order) the number of samples we will need to get the same level of generalization error.

- In general when choosing the appropiate dimension for the feature transform, we must use an approximation-generalization tradeoff:

higher d better chance of being linearly separable $(E_{\text{in}} \downarrow)$ and $E_{out} \uparrow$

lower d possibly non linearly separable(E_{in}) and $\text{E}_{out} \downarrow$

What happens when $d_{VC} = \infty$?

- UNIFORM LEARNING: From the VC dimension analysis we know that ERM rule is a general learning rule for finite d_{vc}

- NONUNIFORM LEARNING
- Now we consider $\mathcal{H} = \bigcup_n \mathcal{H}_n$, $d_{vc}(\mathcal{H}_n) < \infty, n = 1,2,3, \ldots \ldots$
  - This means a class with an infinite VC dimension but defined as the union of a numerable infinity of classes each with $d_{VC} < \infty$
- Example:
  - Class of all polynomials on R. $\mathcal{H} = \bigcup_n \mathcal{H}_n$ where $\mathcal{H}_n$ represents the class of the polynomials of degree n. It's not difficult to show that VCdim$(\mathcal{H}) = \infty$ and VCdim$(\mathcal{H}_n) = \mathrm{n} + 1$

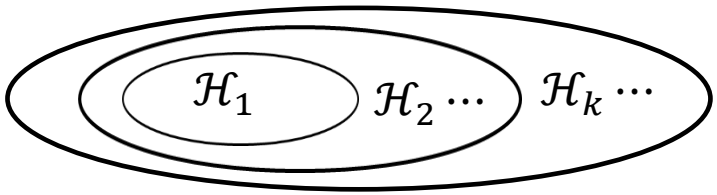
Nonuniform learning rule: SRM

$$\Omega(N, \mathcal{H}, \delta) = \mathcal{O}\left(\sqrt{\frac{d_{VC} \ln N - \ln \delta}{N}}\right)$$

- What happens when $\frac{N}{d_{vc}} < 20?$,
  - small number of samples with respect to the number of effective parameters.
- In this case the ERM rule is not a guarantee for learning

A new induction rule is introduced : **Structural RISK Minimization (SRM)**

$$g^* = arg \min_{i=1,2,\ldots} (E_{in}(g_i) + \Omega(\mathcal{H}_i))$$



$$d_{vc}(\mathcal{H}_1) \leq d_{vc}(\mathcal{H}_2) \leq \cdots \leq d_{vc}(\mathcal{H}_k) \leq \cdots$$

SRM

1. Select a nested sequence of hypothesis set
2. Estimate *g* from each set of the sequence

SRM Implementation Criteria

- Keeps the model complexity fixed and minimize empirical error
- Keeps the empirical error constant (small) and minimize VC dimension

Valid for approaches that minimize the true error rather than empirical

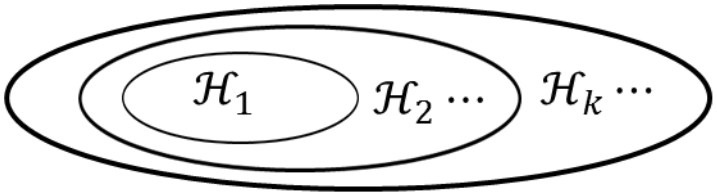
INDUCTION RULES: SUMMARY

Uniform Learning (Minimizing Empirical Error)

- If $d_{VC}(\mathcal{H})$ is finite $\Longleftrightarrow$ $\mathcal{H}$ is agnostic-PAC learnable
- The VC bound is independent of : f, $\mathbb{P}(\mathcal{X})$, $\mathcal{A}$

SRM Learning Criteria (Nonuniform Learning)

$\Omega(N, \mathcal{H}, \delta) = \mathcal{O}\left(\sqrt{\frac{d_{VC} \ln N - \ln \delta}{N}}\right)$ when $\frac{N}{d_{vc}} < 20$ it does not a good guarantee for $E_{out} \approx 0$



$$d_{vc}(\mathcal{H}_1) \leq d_{vc}(\mathcal{H}_2) \leq \cdots \leq d_{vc}(\mathcal{H}_k) \leq \cdots$$

$$g^* = arg \min_{i=1,2,\ldots} (E_{in}(g_i) + \Omega(\mathcal{H}_i))$$

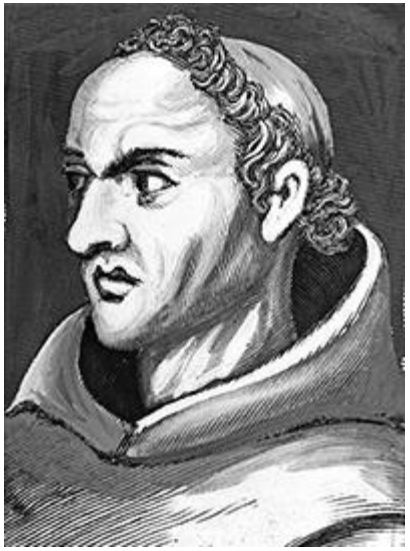
SRM Implementation Criteria

- Keeps model complexity fixed and minimize empirical error
- Keeps empirical error constant (small) y minimize VC dimension

Sample size depends on function

Three Learning Principles

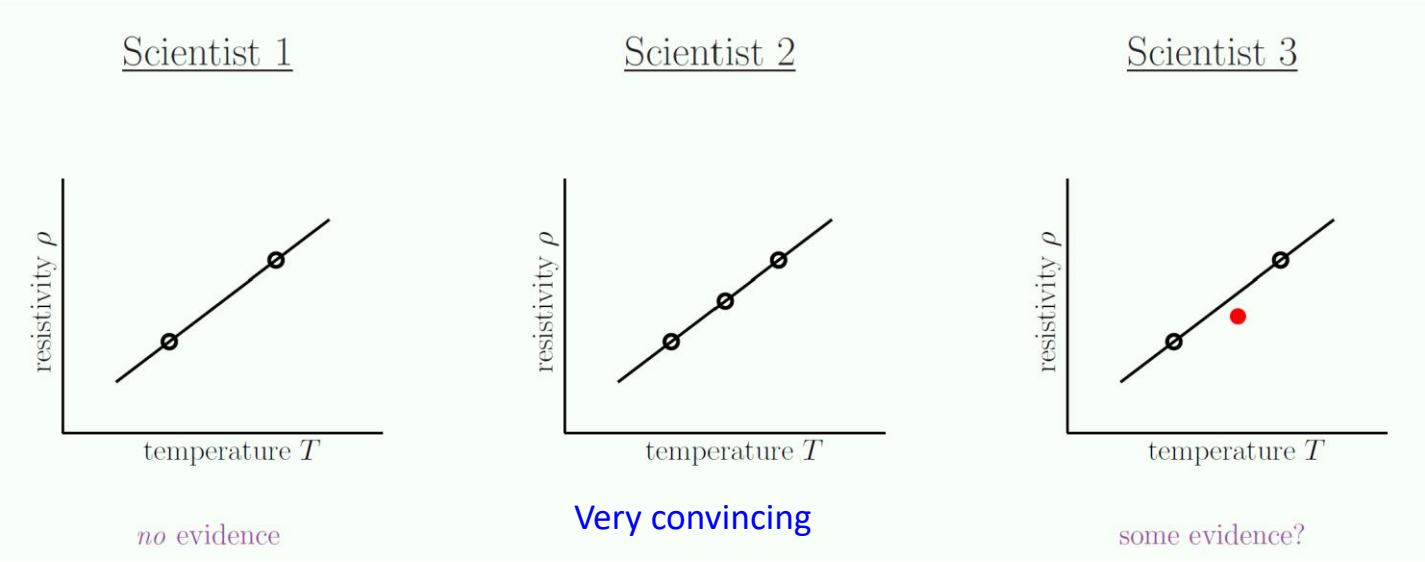
Occam's razor

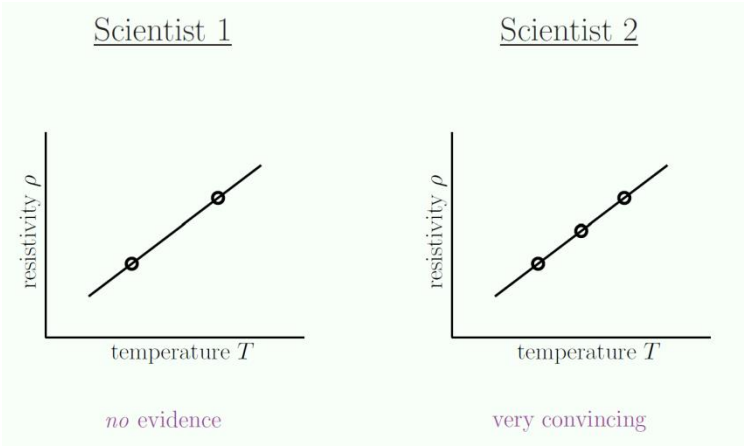


Entities should not be multiplied beyond necessity ‘‘Occam's razor’’ principle attributed to William of Occam c. 1280–1349

We should seek simpler models over complex ones and optimize the tradeoff between model complexity and the accuracy of model's description of the training data

Scientific Experiment





Axiom of Non-Falsifiability.
If an experiment has no chance of falsifying a hypothesis, then the result of that experiment provides no evidence one way or the other for the hypothesis.

Sampling Bias in Learning

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

. . . or, make sure the training and test distributions are the same.
You cannot draw a sample from one distribution and make claims about another distribution

Example.1: Consider the data set of customer loan from a Bank. ¿where is the bias when used to build a rule for new credit appointment ?

Example.2: Consider the data set of historical values of the current trading companies to select companies in which to invest ¿where is the bias?

Data Snooping

If a data set has affected any step in the learning process, it cannot be fully trusted in assessing the outcome.

. . . or, estimate performance with a completely uncontaminated test set
. . . and, **choose** $\mathcal{H}$ **before** looking at the data

Example : Consider a 8 years data set of the daily changes USD/EURO. We want predict UP/DOWN.

Before fitting a model:

1.- We normalize the fully data set.
2.- We separate the last two years from the data set as Test Set.
3.- We fit a model
4.- The fitted model works very well on the data set,
5.- But when used with new data the prediction error was very high,

is there any rational explanation ?

Data Snooping is a Subtle Happy Hell

- The data looks linear, so I will use a linear model, and it worked.
  - If the data were different and didn't look linear, would you do something different?
- Try linear, it fails; try circles it works.
  - If you torture the data enough, it will confess.
- Try linear, it works; so I don't need to try circles.
  - Would you have tried circles if the data were different?
- Read papers, see what others did on the data. Modify and improve on that.
  - If the data were different, would that modify what others did and hence what you did?
  - the data snooping can happen all at once or sequentially by different people
- Input normalization: normalize the data, now set aside the test set.
  - Since the test set was involved in the normalization, wouldn't your *g* change if the test set changed?