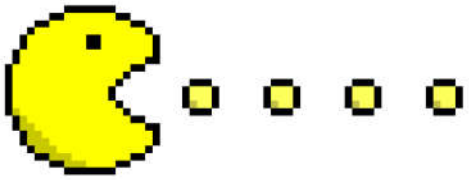
CS 236756 - Technion - Intro to Machine Learning

Tal Daniel

Tutorial 13 - PAC Learning & VC Dimension



- Based on slides by Shai Shalev-Schwarz (https://www.cs.huji.ac.il/~shais/Lectures2014/lecture2.pdf)
- Great (!) Reading Resource - CS229 - Stanford - Machine Learning - Learning Theory (http://cs229.stanford.edu/notes-spring2019/cs229-notes4.pdf)
  - It covers everything and goes into much more details

Agenda

- The PAC (**P**robably **A**pproximately **C**orrect) Learning Framework
  - Empirical Risk Minimization (ERM)
  - The Fundamental Theorem of Statistical Learning
- The VC Dimension
  - Theory
  - Examples

The PAC Learning Framework

PAC stands for "probably approximately correct", which is a framework and set of assumptions under which numerous results on learning theory were proven.

Classification Learning Problem

- The learner's *input*:
  - **Domain Set -** χ: the set of objects we wish to label.
  - **Label Set -** $\mathcal{Y}$: possible outcomes of an experiment.
  - **Training Data -** $S = \{(x^{(i)}, y^{(i)}); i = 1, \ldots, m\}$ a finite sequence of pairs in $\chi \times \mathcal{Y}$
    - Drawn iid from some probability distribution $\mathcal{D}$
- The learner's *output*:
  - **Prediction Rule - hypothesis** - $h : \chi \to \mathcal{Y}$: a function that must predict a label for new domain points.
    - The function is also called: predictor, hypothesis or classifier.
- Sample generating model
  - We assume the instances are generated by an **unknown** probability distribution over χ denoted $\mathcal{D}$.
  - **i.i.d.**: each $x^{(i)}$ is sampled independently from $\mathcal{D}$.
  - **Realizability**: we also assume: $\exists f, f : \chi \to \mathcal{Y}$ such that $y^{(i)} = f(x^{(i)}), \forall i$.
- Measures of success
  - **Training Error** (also called the **empirical risk** or **empirical error**):

$$\hat{\epsilon}(h) = \hat{L}(h) = \frac{1}{m} \sum_{i=1}^{m} 1\{h(x^{(i)}) \neq y^{(i)}\}$$

  - **Classifier Error** (also called the **generalization error**, the **risk** or the **true error**): the error of h is the probability to draw a random sample $(x, y) \sim \mathcal{D}$ such that $h(x) \neq y$:

$$\epsilon(h) = L(h) = P_{(x,y)\sim\mathcal{D}}(h(x) \neq y)$$

    - This is the probability that, if we now draw a new example (x, y) from $\mathcal{D}$, h will misclassify it.
    - We assume that the training data was drawn from the *same* distribution $\mathcal{D}$ with which we are going to evaluate our hypothesis (the assumption of training and testing on the same distribution is part of the **PAC assumptions**).

Classifier Error Example

- Assume binary features of *papayas* (the fruit...)

Softness	**Color**	$Pr(x)$	$h(x)$	$f(x)$
Soft	Green	0.1	Tasty	Not-Tasty
Hard	Green	0.1	Not-Tasty	Not-Tasty
Soft	Orange	0.7	Tasty	Tasty
Hard	Orange	0.1	Tasty	Not-Tasty

- $\hat{L}(h) = \hat{\epsilon}(h) = 0.5$
- $L(h) = \epsilon(h) = 0.2$

- What is $L_D(h)$?
  - We can only approximate it with some probability.
- Why can it only be **approximately** correct?
  - **Claim**: we can't hope to find $h \in \mathcal{H}, \text{s.t. } L_{D,f}(h) = 0$
  - **Proof**:
    - For every $\epsilon \in (0, 1)$ take $X = \{x_1, x_2\}, P(x_1) = 1 - \epsilon, P(x_2) = \epsilon$
    - The probability not to see x_2 at all among m i.i.d. examples is $(1 - \epsilon)^m \approx e^{-\epsilon m}$
    - So, if $\epsilon << \frac{1}{m}$ we are likely not to see x_2 at all, but then we can't know its label!
  - **Relaxation**: we would be happy with $L_{D,f}(h) \leq \epsilon$
- Why can it only be **probably** correct?
  - Recall that the input to the learner is *randomly generated*.
  - There is always a (very small) chance to see the same example again and again.
  - **Claim**: no algorithm can guarantee $L_{D,f}(h) \leq \epsilon$ for sure, that is, with absolute certainty ($P = 1$)
  - **Relaxation**: we would allow the algorithm to fail with probability δ where $\delta \in (0, 1)$ is *user-specified*.

Probably Approximately Correct (PAC) Learning

- The learner doesn't know $\mathcal{D}$ and f.
- The learner receives 2 parameters:
  1. ϵ - *accuracy* parameter.
  2. δ - *confidence* parameter.
- The learner can ask for training data, S containing $m(\epsilon, \delta)$ examples.
- The learner should output a hypothesis h such that with probability of **at least** $1 - \delta$ it holds that $L_{D,f} \leq \epsilon$.

- That is, the learner should be **P**robably (with probability at least $1-\delta$) **A**pproximately (up to accuracy ϵ) **C**orrect.

Empirical Risk Minimization (ERM)

- Consider the setting of *linear classification* and let $h_\theta(x) = 1\{\theta^T x \geq 0\}$.
- Algorithm goal:
  - Find a hypothesis h_s that minimizes the error (risk) with respect to $\mathcal{D}$ and f.
  - But $\mathcal{D}$ and f are **unknown**!
- An alternative goal and a reasonable way to fit the parameters θ would be to try and minimize the training error:

$$\hat{L}(h) = L_s(h) = \frac{|\{i \in [m] : h(x^{(i)}) \neq y^{(i)}\}|}{m}, [m] = \{1, \ldots, m\}$$

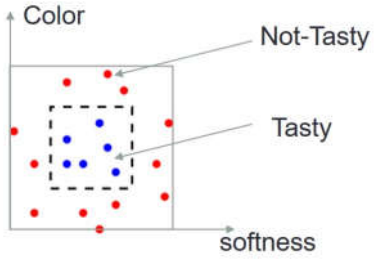
  and pick

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \hat{\epsilon}(h_\theta) = \underset{\theta}{\operatorname{argmin}} \hat{L}(h_\theta)$$

  - This process is called **empirical risk minimization** (ERM).
  - The resulting hypothesis output by the algorithm is $\hat{h} = h_{\hat{\theta}}$.
  - ERM can be thought of as the most basic learning algorithm.
    - Algorithms like Logistic Regression can also be viewed as approximations to ERM.
- We will leave out the specific parameterization of the hypothesis θ and will define the **hypothesis class** $\mathcal{H}$ used by the learning algorithm to be the set of all classifiers considered by it.
- ERM can now be thought of as a **minimization over the class of functions** $\mathcal{H}$, in which the learning algorithm picks the hypothesis:

$$\hat{h} = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \hat{\epsilon}(h)$$

- **Overfitting**:
  - ERM may result in overfitting for the obvious reasons.
  - Assuming the following distribution:



  - We may build a trivial estimator with 0 (empirical) error:

$$h_s(x) = \begin{cases} y^{(i)}, \text{if } \exists i \in [m] \text{ s.t. } x^{(i)} = x \\ 0, \text{ otherwise} \end{cases}$$

  - In order to avoid overfitting, we induce bias.
- **ERM with Inductive Bias**:
  - A common solution to overfitting is to restrict the hypothesis search space.
  - The learner chooses in advance a set of predictors (the hypothesis class $\mathcal{H}$).
  - The choice of $\mathcal{H}$ imposes an *inductive* bias (prior knowledge).
  - In the following we will assume **realizability**:

$$\exists h^* \in \mathcal{H}, \text{ s.t. } L_{D,f}(h^*) = \epsilon(h^*) = 0$$

The Fundamental Theorem of Statistical Learning

- Let $\mathcal{H}$ denote a hypothesis class of binary classifiers.
- Then, there are absolute **constants** C_1, C_2 such that the *sample complexity* (how many samples to draw, roughly) of PAC learning $\mathcal{H}$ is:

$$C_1 \frac{d(\mathcal{H}) + \log(\frac{1}{\delta})}{\epsilon} \leq m_{\mathcal{H}}(\epsilon, \delta) \leq C_2 \frac{d(\mathcal{H}) \log(\frac{1}{\epsilon}) + \log(\frac{1}{\delta})}{\epsilon}$$

  - $d(\mathcal{H})$ - the *VC Dimension* (which will be introduced shortly) of hypotheses class $\mathcal{H}$.
- Furthermore, this sample complexity is achieved by the ERM learning rule

What Is Learnable and How to Learn?

- From the fundamental theorem of statistical learning:
  - The sample complexity is characterized by the **VC Dimension**.
  - The ERM learning rule is generic (near) optimal learner.

VC Dimension

- Great video by Alexander Ihler - https://www.youtube.com/watch?v=puDzy2XmR5c (https://www.youtube.com/watch?v=puDzy2XmR5c)

Motivation

- **Complexity of a learner -** representational power, the ability to generalize.
  - The usual **trade-off**:
    - More power - represent more complex systems $\rightarrow$ may lead to **overfitting.**
    - Less power - won't overfit, but may not find the "best" learner.
  - How to quantify the representational power? Not easily...
    - One solution is the **VC Dimension**
- **No Free Lunch**
  - Suppose that $|\mathcal{X}| = \infty$
  - For any finite subset $\mathcal{C} \subset \mathcal{X}$ take $\mathcal{D}$ to be *uniform* distribution over $\mathcal{C}$
  - If number of training examples is $m \leq \frac{\mathcal{C}}{2}$, then the learner has no knowledge on at least half the elements in $\mathcal{C}$
  - Formally: **No Free Lunch Theorem**
    - Fix $\delta \in (0,1), \epsilon < \frac{1}{2}$. For every learner $\mathcal{A}$ and training set size m, there exists $\mathcal{D}$, f such that with probability of at least δ over the generation of training data S of m examples, it holds that
    $$L_{\mathcal{D},f}(A(S)) \geq \epsilon$$
      - For a *random guess*, $L_{\mathcal{D},f} = \frac{1}{2}$, so the theorem states that you can't be better than a random guess.
- Suppose we got a **training** set $S = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\}$ and we choose classifiers or hypotheses from a hypotheses class $\mathcal{H}$.
  - We try to explain the labels using a hypothesis from $\mathcal{H}$
  - It turned out that the labels we received were *incorrect* and now we get the same instances with different labels: $S' = \{(x^{(1)}, y'^{(1)}), \ldots, (x^{(m)}, y'^{(m)})\}$
  - We try again to explain the labels using a hypothesis from $\mathcal{H}$
  - If we succeed in doing so (that is, find a hypothesis that explains these labels), then something is fishy...
  - Conclusion: if the classifier is able to explain everything, then it is useless...
  - Formally, if $\mathcal{H}$ allows all functions over some set $\mathcal{C}$ of size m, then based on the **No Free Lunch** theorem, we can't learn from a subset of size $\frac{m}{2}$, for example.

VC Dimension - Formal Definition

- Let $\mathcal{C} = \{x_1, \ldots, x_{|C|}\} \subset \mathcal{X}$
- Let $\mathcal{H}_C$ be the restriction of $\mathcal{H}$ to $\mathcal{C}$, namely, $\mathcal{H}_C = \{h_C : h \in \mathcal{H}\}$ where $h_C : \mathcal{C} \rightarrow \{0,1\}$ or $\{-1,+1\}$ is s.t. $h_C(x_i) = h(x_i)$ for every $x_i \in C$
- Observation: we can represent each h_c as the vector:
$$\begin{bmatrix} h(x_1) \\ \vdots \\ h(x_{|C|}) \end{bmatrix} \in \{\pm 1\}^{|C|}$$
- Therfore: $\mathcal{H}_C \leq 2^{|C|}$
- We say that $\mathcal{H}$ **shatters** $\mathcal{C}$ if $|\mathcal{H}_C| = 2^{|C|}$
  - That is, $\mathcal{H}$ can realize any labeling on $\mathcal{C}$, i.e., if for *any* set of labels $\{y^{(1)}, \ldots, y^{(m)}\}$ there exists some $h \in \mathcal{H}$ so that $h(x^{(i)}) = y^{(i)}$ for **all** $i = 1, \ldots, m$
- $VCdim(\mathcal{H}) = sup\{|C| : \mathcal{H} \text{ shatters } \mathcal{C}\}$
  - The VC dimension is the maximal size of a set $\mathcal{C}$ such that $\mathcal{H}$ gives no prior knowledge w.r.t. $\mathcal{C}$, or, the size of the largest set that is shattered by $\mathcal{H}$.
  - In other words, the VC dimension is the maximum number of points that can be arranged such that $h \in \mathcal{H}$ can shatter them.
  - **Dichotomy**: a possible seperation of the sample space into sub-samples.
    - For example: $\{(x_1,1),(x_2,0),(x_3,1)\}$ is a dichotomy, and also $\{(x_1,0),(x_2,0),(x_3,1)\}$ (a total of 2^3 for this example).
  - **Theorem**: Let $\mathcal{H}$ be given, and let $d = VCdim(\mathcal{H})$. Then with probability at least $1-\delta$, we have that for all $h \in \mathcal{H}$:
  $$|\epsilon(h) - \hat{\epsilon}(h)| \leq O(\sqrt{\frac{d}{m}\log\frac{m}{d} + \frac{1}{m}\log\frac{1}{\delta}})$$
  Thus, with probability at least $1-\delta$ we also have that:
  $$\epsilon(\hat{h}) \leq \epsilon(h^*) + O(\sqrt{\frac{d}{m}\log\frac{m}{d} + \frac{1}{m}\log\frac{1}{\delta}})$$
    - $\epsilon(h)$ is the real (test) error and $\hat{\epsilon}(h)$ is the training error (empirical risk).
    - In other words, if a hypothesis class has finite VC dimension, then uniform convergence occurs as m becomes large.
    - This is a very strong result because we can make a statement on data we have not seen!

Finding VC Dimension

- To show that $VCdim(\mathcal{H}) = d$ we need to show that:

1. There **exists** a set $\mathcal{C}$ of size d which is shattered by $\mathcal{H}$
    - That is, show that for some ordering of the points, **any** kind of labeling can be attained by hypothesis from $\mathcal{H}$
2. **Every** set $\mathcal{C}$ of size $d+1$ is not shattered by $\mathcal{H}$

- Can be thought of as a **2-player game**:
    - Fix the definition of $h_\theta = f(x;\theta)$ (the hypotheses class, e.g. linear classifiers)
    - **Player 1**: choose locations $x^{(1)}, \ldots, x^{(d)}$
    - *Player 2*: choose target labels $y^{(1)}, \ldots, y^{(d)}$
    - **Player 1**: choose a hypothesis $h \in \mathcal{H}$, e.g., choose θ in the linear classifier
    - If $f(x;\theta)$ can reproduce the target labeles, **Player 1** wins.
        - $\exists \{x^{(1)}, \ldots, x^{(d)}\}$ s.t. $\forall \{y^{(1)}, \ldots, y^{(d)}\} \exists \theta$ s.t. $\forall i, f(x^{(i)}) = y^{(i)}$
    - The VC dimension would be the value d if *Player 2* covered all the possibles labels and **Player 1** won every game.

VC Dimension - Examples

1️⃣ Example 1 - Toy Example

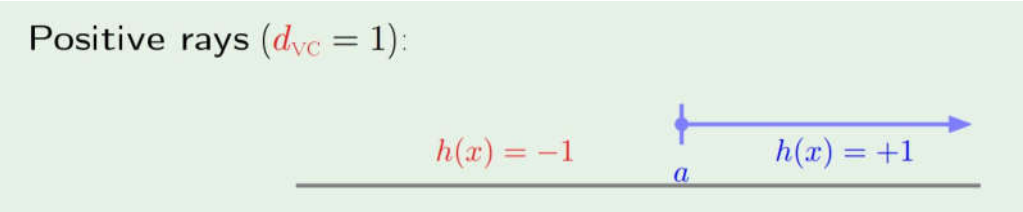
Consider 9 samples, and 8 hypotheses as follows:

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
h_1	0	0	1	0	0	0	1	0	0
h_2	0	1	0	0	0	1	0	0	0
h_3	1	0	0	0	1	1	0	0	0
h_4	0	0	0	1	1	0	0	0	1
h_5	0	0	1	0	0	0	0	1	0
h_6	0	1	0	0	0	0	1	0	0
h_7	1	0	0	0	0	1	0	0	0
h_8	0	0	0	0	0	0	0	0	0

- The first thing to notice is that the whole sample set (1-9) cannot be shattered as we don't have enough hypotheses. In order to shatter the whole set we would need at least 2^9 hypotheses.
- **Excercise**: Are the following sets shattered?
    - $\{x_1\}$
    - $\{x_5, x_6\}$
    - $\{x_1, x_2\}$
    - $\{x_5, x_6, x_7\}$
- **Solution**:
    - $\{x_1\}$ - **yes**, by $\{h_2, h_3\}$
    - $\{x_5, x_6\}$ - **yes**, by $\{h_1, h_2, h_3, h_4\}$
    - $\{x_1, x_2\}$ - **no**, can't get the classification: $x_1 = 1$ and $x_2 = 1$
    - $\{x_5, x_6, x_7\}$ - **no**, can't get the classification: $x_5 = x_6 = x_7 = 1$
- **Excercise**: What is the VC dimension of $\mathcal{X}$?
- **Solution**:
    - The only 3 points with the dichotomy $\{1, 1, 1\}$ are $\{x_1, x_5, x_6\}$
    - But the dichotomy $\{1, 0, 0\}$ isn't achievable.
        - $\rightarrow$ No 3 points can be shattered
    - $\rightarrow VCdim(\mathcal{H}) = 2$

2 Example 2 -Threshold Functions

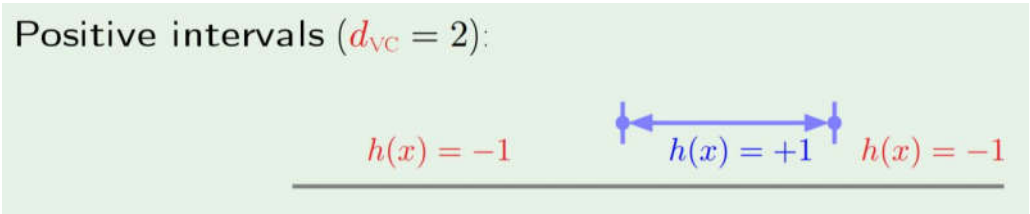
- Threshold functions - $f \in \mathcal{H}$ is a single-parametric threshold classifier on real numbers, i.e., for a certain threshold θ, the classifier f_θ returns 1 if the input number is larger than θ and 0 otherwise. Formally:
$$\mathcal{X} = \mathbb{R}, \mathcal{H} = \{x \to sign(x - \theta) : \theta \in \mathbb{R}\}$$
- Let's "prove" that $VCdim(\mathcal{H}) = 1$:
  1. One ($n = 1$) point can be shattered because for every point x, a classifier $f_\theta(x)$ labels it as 0 if $\theta > x$ and 1 if $\theta < x$. For example, for $(x = 0, label = 0), \theta = 1$ and for $(x = 0, label = 1), \theta = -1$.
  2. No two ($n + 1 = 2$) points can be shattered - because for every set of 2 points, if the smaller is labeled 1, then the larger must also be labeled 1, so not all labelings are possible.



- Image from CalTech's free machine Learning online course by Yaser Abu-Mostafa Learning from Data (http://work.caltech.edu/lectures.html)

3 Example 3 - Intervals Functions

- Intervals functions - $f \in \mathcal{H}$ is a single-parametric interval classifier on real numbers, i.e, for a certain parameter θ, the classifier f_θ returns 1 if the input number is in the interval $[\theta, \theta + 4]$ and 0 otherwise.
- Let's "prove" that $VCdim(\mathcal{H}) = 2$:
  1. Two ($n = 2$) points can be shattered because for every set $\{x, x + 2\}$, a classifier $f_\theta(x)$ labels it as:
     - $(0, 0)$ - if $\theta < x - 4$ or if $\theta > x + 2$.
     - $(1, 0)$ - if $\theta \in [x - 4, x - 2)$.
     - $(1, 1)$ - if $\theta \in [x - 2, x]$.
     - $(0, 1)$ - if $\theta \in (x, x + 2]$.
  2. No three ($n + 1 = 3$) points can be shattered - because for every set of three numbers, if the smallest and the largest are labeled 1, then the middle one must also be labeled 1, so not all labelings are possible.
- This result can be generalized for a two-parametric interval classifier $h_{a,b}$:
$$\mathcal{X} = \mathbb{R}, \mathcal{H} = \{h_{a,b} : a < b \in \mathbb{R}\}$$
where
$$h_{a,b}(x) = 1 \iff x \in [a, b]$$



- Image from CalTech's free machine Learning online course by Yaser Abu-Mostafa Learning from Data (http://work.caltech.edu/lectures.html)

4 Example 4 - Axis Aligned Rectangles

- Axis aligned rectangles:

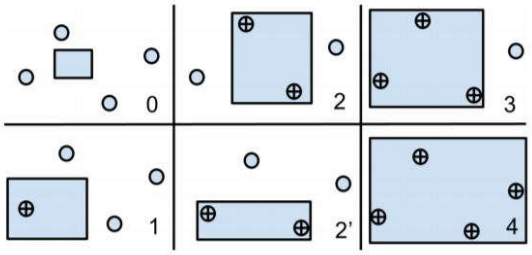
$$\mathcal{X} = \mathbb{R}^2, \mathcal{H} = \{h_{a_1,a_2,b_1,b_2} : a_1 < a_2 \text{ and } b_1 < b_2\}$$

  , where

$$h_{a_1,a_2,b_1,b_2}(x_1, x_2) = 1 \iff x_1 \in [a_1, a_2] \text{ and } x_2 \in [b_1, b_2]$$

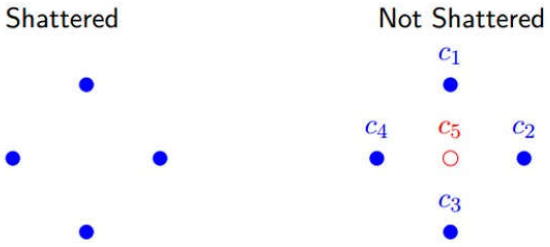
- Let's "prove" that $VCdim(\mathcal{H}) = 4$:

1. Four ($n = 4$) points can be shattered as seen in the following arrangement:



   - Image from Princeton's COS 511: Theoretical Machine Learning, Lecture on VC-Dimension (https://www.cs.princeton.edu/courses/archive/spring14/cos511/scribe_notes/0220.pdf)

2. No five ($n + 1 = 5$) can be shattered - for any 5-point set, we can construct a data assignment in this way: pick the topmost, bottommost, leftmost and rightmost points and give them the label "+". Because there are 5 points, there must be at least one point left to which we assign "–". Any rectangle that contains all the "+" points must contain the "–" point, which is a case where shattering is not possible.



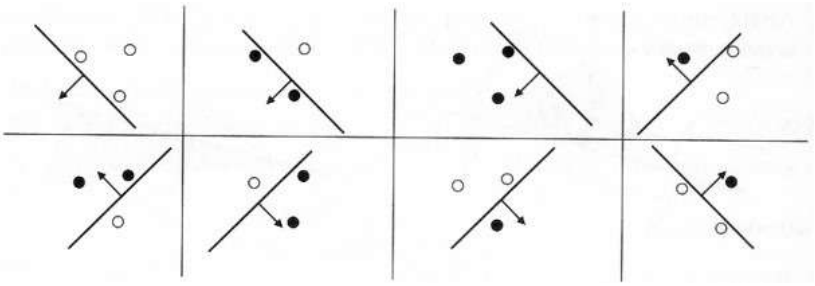
5 Example 5 - Halfspaces

- Halfspaces (linear classifiers):

$$\mathcal{X} = \mathbb{R}^2, \mathcal{H} = \{x \to sign(\langle w, x \rangle)\} : w \in \mathbb{R}^2$$

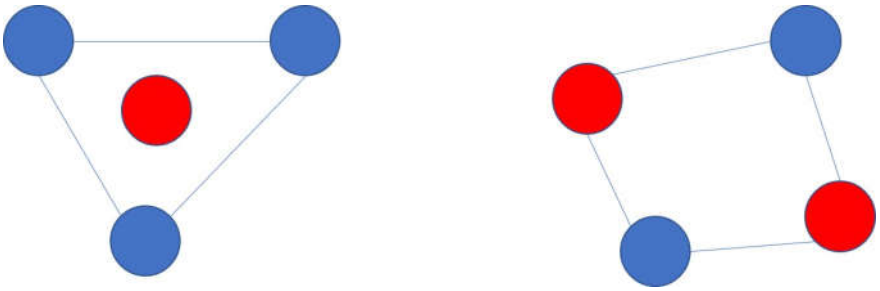
  - For example: $h(x) = 1\{\theta_1 x_1 + \theta_2 x_2 \geq 0\}$
- Let's "prove" that $VCdim(\mathcal{H}) = 3$:

1. Three ($n = 3$) points can be shattered as seen in the following arrangement:



2. No four ($n + 1 = 4$) can be shattered - We consider two cases:
   A. The four points form a convex region, i.e., lie on the convex hull defined by the 4 points.
   B. Three of the 4 points define the convex hull and the 4th point is internal.

   In the first case, the labeling which is positive for one diagonal pair and negative to the other pair cannot be realized by a separating line. In the second case, a labeling which is positive for the three hull points and negative for the interior point cannot be realized.



- The results is generalized for hyperplanes: VC dimension of hyperplanes in $\mathbb{R}^d$ is $d + 1$.

VC Dimension - Special Cases

1. $VCdim(\mathcal{H}) = 0$ - When is the VC dimension equals to zero? Assume $\mathcal{X} = \mathbb{R}^2$. Let $\mathcal{H}$ contain a **single** hypothesis h_1. Thus, the VC dimension of $\mathcal{H}$ is **always** 0! A single hypothesis can impose only one classification, can only assign one labeling to a set of points.
2. $VCdim(\mathcal{H}) = \infty$ - When does the VC dimension go to infinity? Assume $\mathcal{X} = \mathbb{R}^2$. Let $\mathcal{A}$ be the **set of all convex polygons** in $\mathcal{X}$. Define $\mathcal{H}$ as the class of all hypotheses $h_p(x), p \in \mathcal{A}$:

$$h_p(x) = \begin{cases} 1, \text{ if } x \text{ is contained within polygon } p \\ 0, \text{ otherwise} \end{cases}$$

Let's see why $VCdim(\mathcal{H}) = \infty$: for any positive integer n, take n points from $\mathcal{X}$. Place the n points **uniformly spaced** on the **unit circle.** For each 2^n subset of this data, there is a convex polygon with vertices at these n points. For each subset, the convex polygon contains the set and excludes its complement.

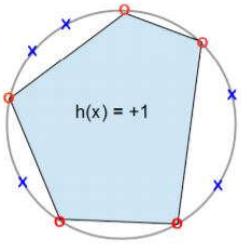


Image from Learnability and VC Dimension (http://www.mathematik.uni-muenchen.de/~deckert/teaching/SS17/ATML/media/VC_dimension.pdf) at LMU Munchen

Credits

- Icons from Icon8.com (https://icons8.com/) - https://icons8.com (https://icons8.com)