Theory of Statistical Learning
Part I

Damien Garreau

Université Côte d'Azur

2022-2023

1. General presentation

Who am I?

- maître de conférence (= assistant professor) in LJAD (Laboratoire Jean Dieudonné)
- before that: postdoctoral researcher (Max Planck Institute, Tübingen, Germany)
- even before: PhD in Inria Paris
- teaching ($\approx$ 200 hours per year)
- research (rest of the time):
  - interpretability of machine learning algorithms
  - using statistical tools to understand methods in computer vision

Goal of the course

- **Goal i):** learn about *statistical learning*
- understand the *maths* behind the algorithms that you learn
- this can save a lot of time!
- one often works with limited resources
- **Goal ii):** learn about theoretical guarantees on existing algorithms
- a way to be reassured: under some assumptions, my method works
- see more clearly the *limitations* of the methods: if some assumption is not satisfied, we can prove that it will fail

Organization of the course

- (provisional) calendar:
  1. February 9 (today), 1:30pm-4:30pm
  2. February 10, 1:30pm-4:30pm
  3. February 15, 9am-12am
  4. February 16, 1:30pm-4:30pm
  5. March 2, 1:30pm-4:30pm (midterm)
  6. March 9, 1:30pm-4:30pm
  7. March 10, 9am-12am
  8. March 16, 2pm-5pm
  9. March 23, 1:30pm-4:30pm
  10. March 30, 1:30pm-4:30pm (exam)
- final grade $=$ (midterm $+$ final)$/2$

Requirements

- **Elementary real analysis:** functions of a real variable, usual functions, continuity, Lipschitz continuity
- **Calculus:** derivative, partial derivatives, gradient, Taylor series
- **Basic probability theory:** measurable space, probability measure, random variable, expectation, conditional expectation, probability density function, cumulative density functions
- **Limit theorems:** law of large numbers, central limit theorem
- **Linear algebra:** vector space, matrix, norms, diagonalization of a matrix, singular value decomposition

If you feel like you are not up to date on one of these points, write me and I will point you towards some resources.

Useful resources

- **Main reference:** Shalev-Schwartz, Ben-David, *Understanding Machine Learning: from Theory to Algorithms*, Cambridge University Press, 2014
- **Also a good read:** Hastie, Tibshirani, Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer Series in Statistics, 2001 (second edition: 2009)
- **Wikipedia:** as good as ever.
- **Wolfram alpha:** if you have computations to make and you do not know want to use a proper language: `https://www.wolframalpha.com/`
- **Google scholar:** use it!

2. Introduction to Statistical Learning

2.1. General introduction

The goal of statistical learning

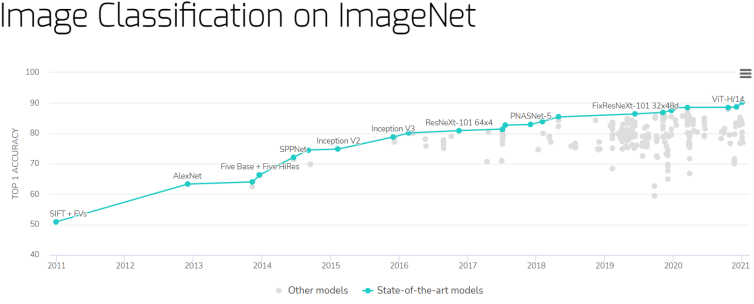
- **Fundamental example:** image classification
- **Goal:** given any image x, we want to predict which object / animal y is in the image

$\longmapsto$ "lion"

- **Main idea:** instead of defining the function f ourselves, we are going to *learn it* from data
- **Why?** no clear definition of a "lion"
- **Motivation:** industry (advertisement), healthcare (automated patient triage), military (automated defense systems)

How is this even possible?

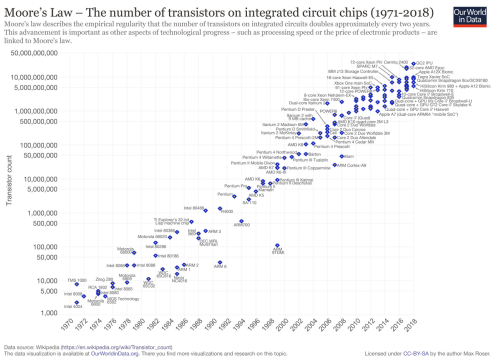
- four ingredients made statistical learning a viable paradigm:
- **Ingredient (i):** data to feed to the models
- previous example from ImageNet[1]: roughly 1 million images for training (150GB of data)



[1]Deng et al., *ImageNet: A large hierarchical image database*, CVPR, 2009

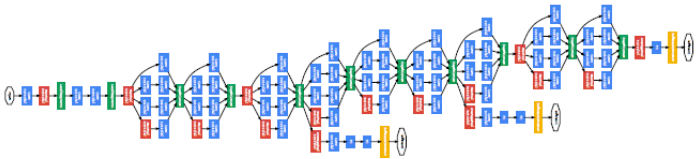
How is this even possible?

- **Ingredient (ii):** computing power
- we have the processing power to deal with these data



How is this even possible?

- **Ingredient (iii):** models that are complex enough
- sate-of-the-art today: (deep) neural networks (originating from much earlier research[2])
- Inception:[3] 24M parameters, GTP-3:[4] 175B



[2]Rosenblatt, *The perceptron, a perceiving and recognizing automaton*, tech report, 1957
[3]Szegedy et al., *Going deeper with convolutions*, CVPR, 2015
[4]Brown et al., *Language Models are Few-Shot Learners*, tech report, 2020

How is this even possible?

SWITCH TRANSFORMERS: SCALING TO TRILLION PARAMETER MODELS WITH SIMPLE AND EFFICIENT SPARSITY

William Fedus*
Google Brain
liamfedus@google.com

Barret Zoph*
Google Brain
barretzoph@google.com

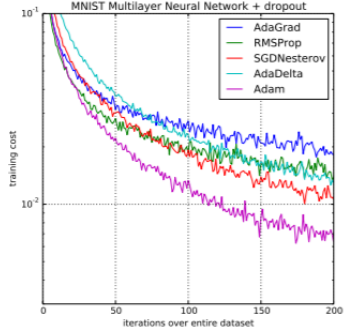
Noam Shazeer
Google Brain
noam@google.com

ABSTRACT

In deep learning, models typically reuse the same parameters for all inputs. Mixture of Experts (MoE) models defy this and instead select *different* parameters for each incoming example. The result is a sparsely-activated model – with an outrageous number of parameters – but a constant computational cost. However, despite several notable successes of MoE, widespread adoption has been hindered by complexity, communication costs, and training instability. We address these with the Switch Transformer. We simplify the MoE routing algorithm and design intuitive improved models with reduced communication and computational costs. Our proposed training techniques mitigate the instabilities, and we show large sparse models may be trained, for the first time, with lower precision (bfloat16) formats. We design models based off T5-Base and T5-Large (Raffel et al., 2019)

How is this even possible?

- **Ingredient (iv):** efficient algorithms to train the models
- gradient descent on steroids[5]
- efficient gradient computations[6]



[5] Kingma, Ba, *ADAM: A method for stochastic optimization*, ICLR, 2015
[6] Rumelhart et al., *Learning representations by back-propagating errors*, Nature, 1986

2.2. First concepts

Input space

- **Input space:** measurable space $\mathcal{X}$ containing all the objects that we want to label
- also called *domain*, or *domain set*
- elements $x \in \mathcal{X}$ are usually described as vectors ($\mathcal{X} = \mathbb{R}^d$)
- coordinates of the vector $=$ *features*
- **Example:** ImageNet images: RGB images $\rightarrow$ 3 8-bits channels

$$\in [\![0, 255]\!]^{299 \times 299 \times 3}$$

- $\mathcal{X}$ can be very high-dimensional in modern applications (here $299 \times 299 \times 3 = 268,203$)

Labels and training data

- **Output:** labels / responses belong to a set $\mathcal{Y}$
- **Example:** $\mathcal{Y}$ is the set of names of object and animals of the dataset

```
{0: 'tench, Tinca tinca',
 1: 'goldfish, Carassius auratus',
 2: 'great white shark, white shark, man-eater, man-eating shark, Carcharodon carcharias',
 3: 'tiger shark, Galeocerdo cuvieri',
 4: 'hammerhead, hammerhead shark',
 5: 'electric ray, crampfish, numbfish, torpedo',
 6: 'stingray',
 7: 'cock',
 8: 'hen',
 9: 'ostrich, Struthio camelus',
```

- we restrict ourselves to $\mathcal{Y} = \{0, 1\}$ for the time being (sometimes identified with $\{-1, +1\}$ for convenience)
- can be much larger in modern applications ($1,000$ for ImageNet)
- **Training data:** $S = ((x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)}))$ *finite* sequence of points of $\mathcal{X} \times \mathcal{Y}$
- also called *training set*

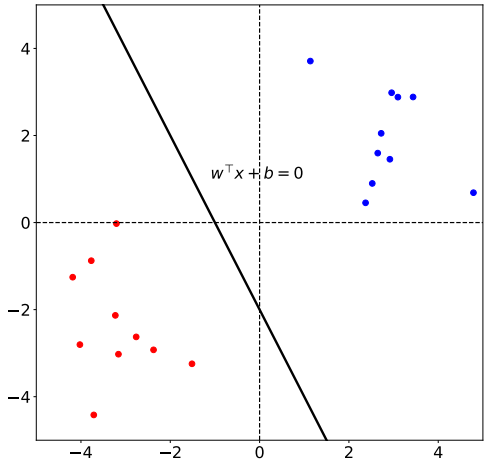
Hypothesis class

- **Hypothesis:** $h : \mathcal{X} \to \mathcal{Y}$ a prediction rule. also called *predictor*, *classifier* (in the context of classification)
- we are looking for a good h
- **Hypothesis class:** $\mathcal{H}$ some space of functions. if no restrictions, set of all measurable functions
- **Example:** linear classifiers:

$$\mathcal{H} = \{h : x \mapsto \text{sign}(\langle w, x \rangle + b), w \in \mathbb{R}^d, b \in \mathbb{R}\},$$

  where $\langle w, x \rangle$ denotes the scalar product between w and x
- given an algorithm A and a dataset S, we will write $h = A(S)$ the output of our algorithm on S

Example: linear classifier



Data generation

- **Data generation:** for now, we assume that there is a true distribution $\mathcal{D}$ of the data on $\mathcal{X}$
- the training examples are i.i.d. samples from $\mathcal{D}$
- i.i.d.: *independent identically distributed*
- **Example:** sample images uniformly at random from a larger set (all the images on the internet)
- hard to satisfy: there is always a bias in the way your dataset is constructed

Assumption (noiseless setting): there exists a function $f : \mathcal{X} \to \mathcal{Y}$ such that $y = f(x)$ for any $x \in \mathcal{X}$.

- **Important:** we know neither $\mathcal{D}$ nor f! we only have access to S

Measure of success

- **Risk of a classifier:** probability that h does not return the correct label on a (new) random sample:
$$\mathcal{R}_{\mathcal{D},f}(h) = \mathbb{P}_{x\sim\mathcal{D}}\left(h(x) \neq f(x)\right) .$$
- **Intuition:** we want to be good, *on average*, for new samples of the same distribution
- subscript often omitted when clear from context
- also called *generalization error*, *true error* (notation L or $\mathcal{E}$)
- **Important:** we want to find h with small generalization error. Ideally,
$$\mathcal{R}_{\mathcal{D},f}(h) = 0 .$$
- **Question:** how to do this?

2.3. Empirical risk minimization

Empirical risk minimization

- as we have seen, what we would *like* to do is find

$$h \in \arg\min_{h \in \mathcal{H}} \mathcal{R}_{\mathcal{D},f}(h) = \arg\min_{h \in \mathcal{H}} \mathbb{P}_{x \sim \mathcal{D}} \left(h(x) \neq f(x) \right) .$$

- **Problem:** we know neither $\mathcal{D}$ nor f...
- ...and even if we did it would still be a very difficult problem (there are *a lot* of measurable functions!)
- **Idea:** replace $\mathcal{R}_{\mathcal{D},f}$ by an *empirical* version
- empirical risk (or training error):

$$\hat{\mathcal{R}}_S(h) = \frac{1}{n} \left| \{ i \in \{1, \dots, n\} \text{ s.t. } h(x^{(i)}) \neq y^{(i)} \} \right| ,$$

  where $|E|$ denotes the cardinality of (finite) set E
- minimizing the empirical risk = empirical risk minimization[7] (ERM)

[7] Vapnik, *Principles of risk minimization for learning theory*, NIPS, 1992

Exercise

Exercise: set $h \in \mathcal{H}$. Let n be a fixed integer.

1. Show that
$$\mathbb{E}_S\left[\hat{\mathcal{R}}_S(h)\right] = \mathcal{R}_{\mathcal{D},f}(h)\,,$$
where the expectation is taken with respect to all i.i.d. draws of S.
2. Show that $\hat{\mathcal{R}}_S(h) \xrightarrow{\mathbb{P}} \mathcal{R}_{\mathcal{D},f}(h)$ when $n \to +\infty$.

Solution

1. First, we see that

$$\left|\{i \in \{1,\dots,n\} \text{ s.t. } h(x^{(i)}) \neq y^{(i)}\}\right| = \sum_{i=1}^{n} \mathbb{1}_{h(x^{(i)})\neq y^{(i)}}\,.$$

Then we write

$$\begin{aligned}
\mathbb{E}\left[\hat{\mathcal{R}}_S(h)\right] &= \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{h(x^{(i)})\neq y^{(i)}}\right] \\
&= \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[\mathbb{1}_{h(x^{(i)})\neq y^{(i)}}\right] && \text{(linearity)} \\
&= \frac{1}{n}\sum_{i=1}^{n} \mathbb{P}\left(h(x^{(i)}) \neq y^{(i)}\right) && (\mathbb{E}\left[\mathbb{1}_A\right] = \mathbb{P}\left(A\right)) \\
&= \frac{1}{n}\sum_{i=1}^{n} \mathbb{P}\left(h(x^{(i)}) \neq f(x^{(i)})\right) && \text{(noiseless assumption)}
\end{aligned}$$

Solution, ctd.

Further, since the $x^{(i)}$ are i.i.d., for any $1 \leq i \leq n$,

$$\mathbb{P}\left(h(x^{(i)}) \neq f(x^{(i)})\right) = \mathbb{P}\left(h(x) \neq f(x)\right) .$$

We recognize the definition of the true risk. Therefore,

$$\begin{aligned} \mathbb{E}\left[\hat{\mathcal{R}}_S(h)\right] &= \frac{1}{n}\sum_{i=1}^{n} \mathcal{R}_{\mathcal{D},f}(h) \\ &= \mathcal{R}_{\mathcal{D},f}(h) \qquad \text{(does not depend on } i\text{)} \end{aligned}$$

2. Since the $x^{(i)}$ are i.i.d. random variables, so are the Z_i defined by

$$Z_i = \mathbb{1}_{h(x^{(i)}) \neq f(x^{(i)})} .$$

Solution, ctd.

Moreover, the Z_is are bounded almost surely (by 1). In particular, they are integrable. Therefore, we can use the law of large numbers and write

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{h(x^{(i)})\neq y^{(i)}} \xrightarrow{\mathbb{P}} \mathbb{E}\left[\mathbb{1}_{h(x^{(1)})\neq f(x^{(1)})}\right] .$$

From question 1., we deduce that

$$\hat{\mathcal{R}}_S(h) \xrightarrow{\mathbb{P}} \mathcal{R}_{\mathcal{D},f}(h) .$$

Reminder: the law of large numbers

Theorem (Strong Law of Large Numbers = SLLN): Let $Z_1, Z_2, \ldots$ be a sequence of i.i.d. random variables. Assume that $\mathbb{E}[|Z_1|] < +\infty$ and set $\mu := \mathbb{E}[Z_1]$. Then

$$\frac{Z_1 + \cdots + Z_n}{n} \xrightarrow{\text{a.s.}} \mu\,.$$

- **Intuition:** average of measurements converges towards the true value
- multivariate extension: coordinate-wise

2.4. Overfitting

Overfitting

- **Problem:** is the hypotheses class $\mathcal{H}$ is too large, then we can bring the empirical risk to *zero*
- easy when $\mathcal{H}$ is the set of all measurable functions:

$$h(x) = \begin{cases} y^{(i)} & \text{if } \exists i \in \{1, \dots, n\} \text{ s.t. } x = x^{(i)} \\ 0 & \text{otherwise.} \end{cases}$$

- in particular,

$$\forall 1 \leq i \leq n, \quad h(x^{(i)}) = y^{(i)} .$$

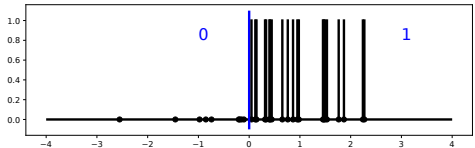
- in that case,

$$\hat{\mathcal{R}}_S(h) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{h(x^{(i)}) \neq y^{(i)}} = 0 .$$

- our predictor has *memorized* the examples, but cannot *generalize*

Overfitting: a simple example

- consider binary classification in $\mathcal{X} = \mathbb{R}$
- very simple problem: $f(x) = \mathbb{1}_{x>0}$ (examples have label 1 if they are positive, 0 otherwise)
- suppose that $\mathcal{D}$ is symmetric and has a density over $\mathbb{R}$ (for instance $\mathcal{D} = \mathcal{N}(0, 1)$)
- if $\mathcal{H}$ is the class of **all** functions $\mathcal{X} \to \{0, 1\}$, at training we will learn the following:



- since X has a density, $\mathbb{P}\left(X = x^{(i)}\right) = 0$: always predict 0
- thus the generalization error is equal to $1/2 \to$ not very good

One solution to overfitting

- one possible solution: **reduce the hypothesis class**
- in advance, choose a restricted $\mathcal{H}$ and solve

$$h \in \arg\min_{h \in \mathcal{H}} \hat{\mathcal{R}}_S(h)\,. \qquad (\star)$$

- by doing so, we *bias* the predictor
- **Example:** take $\mathcal{H}$ the class of linear predictor $\rightarrow$ much less functions
- but surely some problems are too complicated for linear classifiers!
- **One of the fundamental questions of statistical learning theory:** how to choose $\mathcal{H}$ for a given class of problems?
- **Notation:** we will write h_S the solution of $(\star)$

Finite hypothesis class

- as a starting point, let us investigate *finite* $\mathcal{H}$
- **Remark:** for a given class of algorithms, we are limited by our computer $\Rightarrow$ always finite in a sense
- let us analyze ERM for finite hypothesis classes

Assumption (realizability): there exists $h^\star \in \mathcal{H}$ such that

$$\mathcal{R}_{\mathcal{D},f}(h^\star) = 0\,.$$

- **Consequence:** in the noiseless setting, $\hat{\mathcal{R}}_S(h^\star) = 0$ with proba. 1 over the sampling of S and therefore $\hat{\mathcal{R}}_S(h_S) = 0$ (see next slide)
- but remember: we are interested in the *true* risk $\mathcal{R}_{\mathcal{D},f}(h_S)$

Consequence of realizability

- by assumption, there exists $h^\star \in \mathcal{H}$ such that $\mathcal{R}_{\mathcal{D},f}(h^\star) = 0$
- by definition of the risk,
$$\mathbb{P}\left(h^\star(X) \neq f(X)\right) = 0\,.$$
- that is, $h^\star(x) = f(x)$ almost surely when x is sampled according to $\mathcal{D}$
- in particular, since $X^{(1)}, \ldots, X^{(n)}$ is an i.i.d. sample from $\mathcal{D}$,
$$\forall 1 \leq i \leq n, \quad h^\star(x^{(i)}) = f(x^{(i)})\,.$$
- we deduce that $\hat{\mathcal{R}}_S(h^\star) = 0$
- but remember: h_S **minimizes the empirical risk** over $\mathcal{H}$
- thus $\hat{\mathcal{R}}_S(h_S) \leq \hat{\mathcal{R}}_S(h^\star) = 0$

Randomness of the sample

- recall that S is an i.i.d. random sample from $\mathcal{D}$
- we could be unlucky!
- for instance, sample only images with a lion $\rightarrow$ this predictor will surely *fail* when presented with images of other animals
- $\Rightarrow$ in our analysis, we allow for a margin of error δ
- $1 - \delta =$ confidence parameter (you can imagine $\delta = 0.01$ if you want)
- **Typical statement:** Let $\delta \in (0, 1)$. With probability $1 - \delta$, it holds that h_S satisfies this property
- this means with probability $1 - \delta$ *on the sampling of* S
- sometimes abridged to "with high probability"

Probably Approximately Correct learning

- we can show our first result:

Proposition: Assume that $|\mathcal{H}|$ is finite. Let $\delta \in (0,1)$ and $\varepsilon \in (0,1)$, let n be an integer such that

$$n \geq \frac{\log\left(|\mathcal{H}|/\delta\right)}{\varepsilon}\,.$$

Then, in the **noiseless setting** for any labeling function f and any distribution $\mathcal{D}$ such that the **realizability** assumption holds, with probability at least $1-\delta$, it holds that

$$\mathcal{R}_{\mathcal{D},f}(h_S) \leq \varepsilon\,.$$

- *probably:* with probability $\geq 1-\delta$ over the sampling
- *approximately correct:* with tolerance ε on the test error

Proof of the proposition

- let us introduce the set of *bad hypotheses*

$$\mathcal{H}_B = \{h \in \mathcal{H} \text{ s.t. } \mathcal{R}_{\mathcal{D},f}(h) > \varepsilon\},$$

- and the set of *misleading examples*

$$M = \{S \text{ s.t. } \exists h \in \mathcal{H}_B, \hat{\mathcal{R}}_S(h) = 0\}.$$

- let S such that $\mathcal{R}_{\mathcal{D},f}(h_S) > \varepsilon$
- by definition, $h_S \in \mathcal{H}_B$
- by the realizability assumption, $\hat{\mathcal{R}}_S(h_S) = 0$
- we deduce that $S \in M$: we have showed that

$$\boxed{\{S \text{ s.t. } \mathcal{R}_{\mathcal{D},f}(h_S) > \varepsilon\} \subseteq M.}$$

- thus

$$\mathbb{P}\left(\mathcal{R}_{\mathcal{D},f}(h_S) > \varepsilon\right) \leq \mathbb{P}\left(\exists h \in \mathcal{H}_B, \hat{\mathcal{R}}_S(h) = 0\right).$$

Proof of the proposition, ctd.

- now we upper bound the right-hand side by the *union bound*:

$$\mathbb{P}\left(\exists h \in \mathcal{H}_B, \hat{\mathcal{R}}_S(h) = 0\right) \leq \sum_{h \in \mathcal{H}_B} \mathbb{P}\left(\hat{\mathcal{R}}_S(h) = 0\right) .$$

- **Reminder:** let A and B be two events, then

$$\mathbb{P}(A \cup B) \leq \mathbb{P}(A) + \mathbb{P}(B) .$$

- with our definition of the empirical risk,

$$\hat{\mathcal{R}}_S(h) = 0 \quad \Leftrightarrow \quad \forall 1 \leq i \leq n, \quad h(x^{(i)}) = y^{(i)} .$$

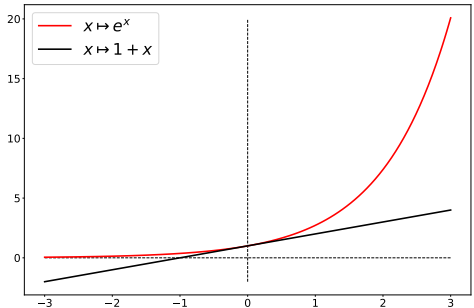
- since the sample is i.i.d. and $h \in \mathcal{H}_B$,

$$\begin{aligned} \mathbb{P}\left(h(x^{(1)}) = y^{(1)}, \ldots, h(x^{(n)}) = y^{(n)}\right) &= \mathbb{P}(h(x) = y)^n \\ &= (1 - \mathcal{R}_{\mathcal{D},f}(h))^n \\ &\leq (1 - \varepsilon)^n . \end{aligned}$$

Proof of the proposition, ctd.

- we notice that $\forall x \in \mathbb{R}$,

$$e^x \geq 1 + x\,.$$



- thus $1 - \varepsilon \leq e^{-\varepsilon}$
- we deduce that

$$(1 - \varepsilon)^n \leq e^{-n\varepsilon}\,.$$

Proof of the proposition, ctd.

- let us put everything together:

$$
\begin{aligned}
\mathbb{P}\left(\mathcal{R}_{\mathcal{D},f}(h_S) > \varepsilon\right) &\leq \mathbb{P}\left(\exists h \in \mathcal{H}_B, \hat{\mathcal{R}}_S(h) = 0\right) && \text{(first part of the proof)} \\
&\leq |\mathcal{H}_B|\,(1-\varepsilon)^n && \text{(union bound)} \\
&\leq |\mathcal{H}_B|\,\mathrm{e}^{-n\varepsilon} && \text{(exponential bound)} \\
&\leq |\mathcal{H}|\,\mathrm{e}^{-n\varepsilon} && (\mathcal{H}_B \subseteq \mathcal{H}) \\
&\leq |\mathcal{H}| \exp\left(-\varepsilon \cdot \frac{\log\left(|\mathcal{H}|/\delta\right)}{\varepsilon}\right) && \text{(hyp. on } n\text{)} \\
\mathbb{P}\left(\mathcal{R}_{\mathcal{D},f}(h_S) > \varepsilon\right) &\leq \delta\,. \quad \square
\end{aligned}
$$

2.5. PAC learning

PAC learning

- first definition:

Definition (PAC learnability): A hypothesis class $\mathcal{H}$ is *PAC learnable* if there exists a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm A such that for every $\varepsilon, \delta \in (0,1)^2$, for every labeling function f and any distribution $\mathcal{D}$ such that the **realizability assumption** holds, then, if $h = A(S)$ with S a dataset containing more than $m_{\mathcal{H}}(\varepsilon, \delta)$ samples,

$$\mathbb{P}\left(\mathcal{R}_{\mathcal{D},f}(h) > \varepsilon\right) \leq \delta\,.$$

- $m_{\mathcal{H}}$ is called the *sample complexity* of learning $\mathcal{H}$
- many $m_{\mathcal{H}}$, we take the minimal one

A corollary

- we have already showed the following:

Corollary: Every finite hypothesis class is PAC learnable with sample complexity

$$m_{\mathcal{H}}(\varepsilon, \delta) \leq \left\lceil \frac{\log\left(|\mathcal{H}| / \delta\right)}{\varepsilon} \right\rceil .$$

- **Spoiler alert:** infinite class are also PAC learnable, but we need to define VC dimension
- first let us **generalize** our definition of PAC learning

Removing assumptions

- **Realizibility:** too restrictive!
- maybe there are no functions in our class mapping exactly examples to the label
- **Example:** consider $\mathcal{H}$ the set of functions defined by $h(x) = \mathbb{1}_{x \in A}$ where A are rectangles
- maybe examples labeled 1 are not contained in a rectangle
- $\Rightarrow$ *agnostic* PAC learning
- further, we now consider $\mathcal{D}$ a distribution on $\mathcal{X} \times \mathcal{Y}$, not only $\mathcal{X}$
- for a fixed x, y is now a random variable with distribution $\mathcal{D}_x$, there is no more f
- the definition of the risk is slightly modified:

$$\mathcal{R}_{\mathcal{D}}(h) = \mathbb{P}_{(X,Y)\sim\mathcal{D}}\left(h(X) \neq Y\right) .$$

Agnostic PAC learning

- our goal remains the same: find $h \in \mathcal{H}$ that minimizes $\mathcal{R}_{\mathcal{D}}(h)$
- we generalize slightly the definition:

Definition (agnostic PAC learnable): A hypothesis class $\mathcal{H}$ is *agnostic PAC learnable* if there exists a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm A with the following properties: for every $\varepsilon, \delta \in (0,1)$, and for every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, when running the algorithm on $n \geq m_{\mathcal{H}}(\varepsilon, \delta)$ examples i.i.d. generated from $\mathcal{D}$, $h = A(S)$ satisfies

$$\mathcal{R}_{\mathcal{D}}(h) \leq \min_{h' \in \mathcal{H}} \mathcal{R}_{\mathcal{D}}(h') + \varepsilon\,,$$

with probability $\geq 1 - \delta$.

Exercise

Exercise: Set $g(x) = \mathbb{P}(Y = 1 \mid X = x)$. We define the *Bayes optimal predictor* as

$$f_{\mathcal{D}}(x) = \begin{cases} 1 & \text{if } g(x) \geq 1/2 \\ 0 & \text{otherwise.} \end{cases}$$

1. let $h : \mathcal{X} \to \{0, 1\}$ be a classifier. Show that
$$\mathbb{P}(h(X) \neq Y \mid X = x) = g(x) \cdot \mathbb{P}(h(X) = 0 \mid X = x) + (1 - g(x)) \cdot \mathbb{P}(h(X) = 1 \mid X = x) \, .$$
2. deduce that
$$\mathbb{P}(f_{\mathcal{D}}(X) \neq Y \mid X = x) = \min(g(x), 1 - g(x)) \, .$$
3. show that
$$\mathbb{P}(h(X) \neq Y \mid X = x) \geq \mathbb{P}(f_{\mathcal{D}}(X) \neq Y \mid X = x) \, .$$
4. deduce that $f_{\mathcal{D}}$ is risk optimal, that is, for any predictor h,
$$\mathcal{R}_{\mathcal{D}}(f_{\mathcal{D}}) \leq \mathcal{R}_{\mathcal{D}}(h) \, .$$

Correction of the exercise

1. there are two mutually exclusive possibility for misclassifiction: either $h(X) = 0$ and $Y = 1$, or $h(X) = 1$ and $Y = 0$. Thus

$$\begin{aligned}
\mathbb{P}(h(X) \neq Y \mid X = x) &= \mathbb{P}(h(X) = 0 \text{ and } Y = 1 \mid X = x) \\
&\quad + \mathbb{P}(h(X) = 1 \text{ and } Y = 0 \mid X = x) \\
&= \mathbb{P}(h(X) = 0 \mid X = x) \cdot \mathbb{P}(Y = 1 \mid X = x) \\
&\quad + \mathbb{P}(h(X) = 1 \mid X = x) \cdot \mathbb{P}(Y = 0 \mid X = x) \\
&= \mathbb{P}(h(X) = 0 \mid X = x) \cdot g(x) \\
&\quad + \mathbb{P}(h(X) = 1 \mid X = x) \cdot (1 - g(x))
\end{aligned}$$

by definition of g. **Remark:** we keep the proba formulation since h maybe non-deterministic.

Correction of the exercise, ctd.

2. we specialize the result of the previous question to $h = f_{\mathcal{D}}$. We obtain:

$$\begin{aligned}\mathbb{P}\left(f_{\mathcal{D}}(X) \neq Y \,|\, X = x\right) = g(x) \cdot \mathbb{P}\left(f_{\mathcal{D}}(X) = 0 \,|\, X = x\right) \\ + (1 - g(x)) \cdot \mathbb{P}\left(f_{\mathcal{D}}(X) = 1 \,|\, X = x\right) .\end{aligned}$$

By definition of $f_{\mathcal{D}}$, $\mathbb{P}\left(f_{\mathcal{D}}(X) = 0 \,|\, X = x\right)$ is a *deterministic* quantity depending only on $g(x)$. We find that

$$\begin{aligned}\mathbb{P}\left(f_{\mathcal{D}}(X) \neq Y \,|\, X = x\right) &= g(x) \cdot \mathbb{1}_{g(x)<1/2} + (1 - g(x)) \cdot \mathbb{1}_{g(x)\geq 1/2} \\ &= \min(g(x), 1 - g(x)) ,\end{aligned}$$

where the last step is obtained after careful inspection of the two possible cases.

Correction of the exercise, ctd.

3. Starting from question 1., we know that

$$\begin{aligned}\mathbb{P}\left(h(X) \neq Y \,|\, X = x\right) &= \mathbb{P}\left(h(X) = 0 \,|\, X = x\right) \cdot g(x) \\ &\quad + \mathbb{P}\left(h(X) = 1 \,|\, X = x\right) \cdot (1 - g(x)) \\ &\geq \min(g(x), 1 - g(x)) \cdot \mathbb{P}\left(h(X) = 0 \,|\, X = x\right) \\ &\quad \min(g(x), 1 - g(x)) \cdot \mathbb{P}\left(h(X) = 1 \,|\, X = x\right) \\ &= \min(g(x), 1 - g(x)) \,,\end{aligned}$$

since $h(X) = 0$ and $h(X) = 1$ are mutually exclusive. According to question 2.,

$$\min(g(x), 1 - g(x)) = \mathbb{P}\left(f_{\mathcal{D}}(X) \neq Y \,|\, X = x\right) \,,$$

and we deduce that

$$\mathbb{P}\left(h(X) \neq Y \,|\, X = x\right) \geq \mathbb{P}\left(f_{\mathcal{D}}(X) \neq Y \,|\, X = x\right) \,.$$

Correction of the exercise, ctd.

4. Let $h \in \mathcal{H}$. We write

$$
\begin{aligned}
\mathcal{R}_{\mathcal{D}}(f_{\mathcal{D}}) &= \mathbb{P}\left(f_{\mathcal{D}}(X) \neq Y\right) && \text{(definition)} \\
&= \mathbb{E}_{X,Y}\left[\mathbb{1}_{f_{\mathcal{D}}(X)\neq Y}\right] && (\mathbb{E}\left[\mathbb{1}_A\right] = \mathbb{P}(A)) \\
&= \mathbb{E}_{x\sim X}\left[\mathbb{E}\left[\mathbb{1}_{f_{\mathcal{D}}(X)\neq Y} \mid X = x\right]\right] && \text{(law of total expectation)} \\
&= \mathbb{E}_{x\sim X}[\mathbb{P}\left(f_{\mathcal{D}}(X) \neq Y \mid X = x\right)] \\
&\leq \mathbb{E}_{x\sim X}[\mathbb{P}\left(h(X) \neq Y \mid X = x\right)] && \text{(question 3.)} \\
&= \mathbb{P}\left(h(X) \neq Y\right) \\
&= \mathcal{R}_{\mathcal{D}}(h). \quad \square
\end{aligned}
$$

Beyond binary classification

- **Multiclass classification:** $\mathcal{Y}$ is some (potentially large) finite set
- we have already encountered such examples ImageNet (1000 classes)
- **Important:** classes are not numbers: being in class 2 instead of 1 is equally bad as 53 instead of 1
- **Regression:** $\mathcal{Y} \subseteq \mathbb{R}^k$, we say *target set* instead of labels
- other notion of success:

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

  for instance
- **Other examples:** structured prediction, functional regression, etc.

Generalized loss functions

- to accommodate for these various settings, we consider arbitrary **loss functions** $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$
- **Intuition:** $\ell(y, y') \approx 0$ means that y and y' are close
- we generalize the notions of risk and empirical risk:

$$\mathcal{R}_{\mathcal{D}}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(y, h(x))] \quad \text{and} \quad \hat{\mathcal{R}}_S(h) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h(x_i))\,.$$

- **Example (i):** $0-1$ loss:

$$\ell(y, y') = \begin{cases} 0 & \text{if } y = y' \\ 1 & \text{otherwise.} \end{cases}$$

- this is the loss we used for binary classification

Generalized loss functions, ctd.

- **Example (ii):** hinge loss:
$$\ell(y, y') = \max(0, 1 - yy') \,.$$
- **Example (iii):** square loss:
$$\ell(y, y') = (y - y')^2 \,.$$
used for regression.
- **Example (iv):** ℓ_1 loss:
$$\ell(y, y') = |y - y'| \,.$$
also used for regression.
- we will see more of them, generally symmetric and increasing at infinity

PAC learnability

- we can now give the general definition of PAC learnability:

Definition (PAC learnability): A hypothesis class $\mathcal{H}$ is *agnostic PAC learnable* for the loss ℓ if there exists a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm A with the following properties: for every $\varepsilon, \delta \in (0,1)$, and for every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, when running the algorithm on $n \geq m_{\mathcal{H}}(\varepsilon, \delta)$ examples i.i.d. generated from $\mathcal{D}$, $h = A(S)$ satisfies

$$\mathcal{R}_{\mathcal{D}}(h) \leq \min_{h' \in \mathcal{H}} \mathcal{R}_{\mathcal{D}}(h') + \varepsilon\,,$$

with probability $\geq 1 - \delta$.

- **Intuition:** running the algorithm on a sufficient number of examples gives a good predictor most of the time, *for any data distribution*

Example: least-square regression

- consider data $(x_i, y_i) \in \mathbb{R}^d \times \mathbb{R}$
- *regression* problem
- let us restrict ourselves to *linear* hypotheses:

$$\mathcal{H} = \{h : x \mapsto w^\top x + b, w \in \mathbb{R}^d, b \in \mathbb{R}\}\,.$$

- **Remark:** $w^\top x$ is the scalar product between w and x:

$$w^\top x = \sum_{j=1}^{d} w_j x_j\,.$$

- **Appropriate loss?** $\ell(y, y') = (y - y')^2$
- **ERM** $\Rightarrow$

$$(\hat{w}, \widehat{b}) \in \operatorname*{arg\,min}_{w, b \in \mathbb{R}^d, \mathbb{R}} \sum_{i=1}^{n} (y_i - w^\top x_i - b)^2\,,$$

ordinary least squares

Example: ridge regression

- we may want to **keep only solutions that have a small ℓ_2 norm**
- we can, for instance, restrict $\mathcal{H}$ even further
- take $\lambda > 0$ and set

$$\mathcal{H}_\lambda = \{h \text{ linear s.t. } \|h\| \leq \lambda\}.$$

- ridge regression solves

$$(\hat{w}, \widehat{b}) \in \underset{\substack{w,b \in \mathbb{R}^d, \mathbb{R} \\ \|w\| \leq \lambda}}{\arg\min} \sum_{i=1}^{n} (y_i - w^\top x_i - b)^2 .$$

- equivalent to *regularization*:

$$(\hat{w}, \widehat{b}) \in \underset{w,b \in \mathbb{R}^d, \mathbb{R}}{\arg\min} \left\{ \sum_{i=1}^{n} (y_i - w^\top x_i - b)^2 + \lambda \|w\|^2 \right\}.$$

Example: LASSO

- maybe we want **a lot of coordinates to be zero**
- then ℓ_1 norm is appropriate
- take $\lambda > 0$ and set

$$\mathcal{H}_\lambda = \{h \text{ linear s.t. } \|h\|_1 \leq \lambda\}.$$

- the LASSO solves

$$(\hat{w}, \widehat{b}) \in \underset{\substack{w,b \in \mathbb{R}^d, \mathbb{R} \\ \|w\|_1 \leq \lambda}}{\arg\min} \sum_{i=1}^{n} (y_i - w^\top x_i - b)^2 .$$

- equivalent to the regularized version:

$$(\hat{w}, \widehat{b}) \in \underset{w,b \in \mathbb{R}^d, \mathbb{R}}{\arg\min} \left\{ \sum_{i=1}^{n} (y_i - w^\top x_i - b)^2 + \lambda \|w\|_1 \right\}.$$

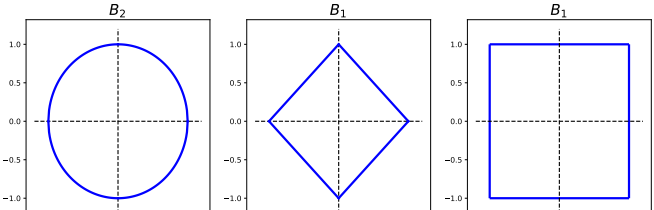
Reminder on norms

Definition: for any $p > 0$, we define the p-norm on $\mathbb{R}^d$ by

$$\forall u \in \mathbb{R}^d, \qquad \|u\|_p := \sqrt[p]{\sum_{j=1}^{d} |u_j|^p}.$$

When $p = +\infty$, we set $\|u\|_\infty := \max_k(|u_k|)$.

- most commonly used: 2-norm = Euclidean norm (Pythagoras)



3. A closer look at test error

Recap

- take $\mathcal{H}$ hypotheses class
- until now, we have seen two central quantities:
- **Empirical risk:**

$$\hat{\mathcal{R}}_S(h) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, h(x_i))\,.$$

- **Generalization error:**

$$\mathcal{R}_{\mathcal{D}}(h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(y, h(x))]\ .$$

- we want to minimize $\mathcal{R}_{\mathcal{D}}(h)$ for $h \in \mathcal{H}$
- unfortunately, we only have access to a proxy, $\hat{\mathcal{R}}_S(h)$
- Empirical risk minimization: find $h_S \in \mathcal{H}$ such that $\hat{\mathcal{R}}_S(h)$ is minimal

Error decomposition

- we decompose the error of h_S in two parts:

$$\mathcal{R}_{\mathcal{D}}(h_S) = \mathcal{E}_{\text{approx}} + \mathcal{E}_{est}\,, \text{ with}$$

- **Approximation error:**

$$\mathcal{E}_{\text{approx}} = \min_{h \in \mathcal{H}} \mathcal{R}_{\mathcal{D}}(h)\,.$$

- minimum risk achievable by a predictor in the class. Measures how much we loose because we restrict ourselves (0 if we consider all functions)
- **Estimation error:**

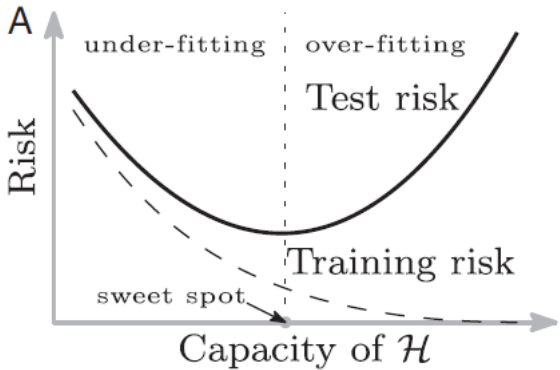
$$\mathcal{E}_{est} = \mathcal{R}_{\mathcal{D}}(h_S) - \min_{h \in \mathcal{H}} \mathcal{R}_{\mathcal{D}}(h)\,.$$

- error because the empirical risk is only an estimate of the true risk

Bias and complexity

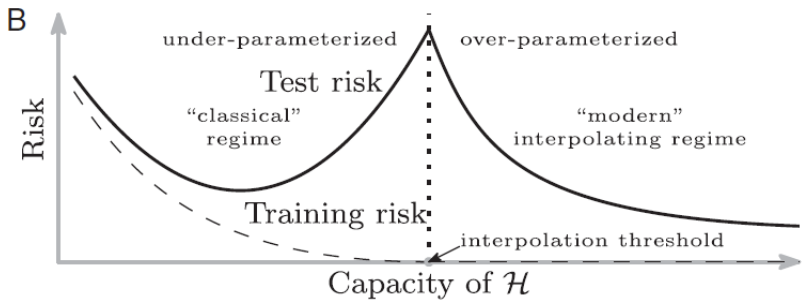
- recall that we want to **minimize** $\mathcal{R}_{\mathcal{D}}(h_S)$
- approximation error: large if $\mathcal{H}$ is small, can go to zero if $\mathcal{H}$ large enough
- estimation error: small if $\mathcal{H}$ is small, can be large if $\mathcal{H}$ large
- in many cases, **trade-off** between these two possibilities
- test error has a U shape (sum of the two)
- but that is not the whole picture
- it can happen that $\mathcal{E}_{est}$ small even for rich classes of functions

The classical picture



Belkin, Hsu, Mandal, *Reconciling modern machine-learning practice and the classical bias-variance trade-off*, Proceedings of the National Academy of Science, 2019

A more accurate picture



Belkin, Hsu, Mandal, *Reconciling modern machine-learning practice and the classical bias-variance trade-off*, Proceedings of the National Academy of Science, 2019

4. VC dimension

4.1. Infinite classes can be PAC learnable

Introduction

- **What we have seen so far:** (i) finite $\mathcal{H}$ are PAC learnable
- (ii) sample complexity grows as $\log |\mathcal{H}|$
- what happens for infinite classes?
- clearly this is not the right notion ("large"?)
- in this section we introduce *Vapnik-Chervonenkis* dimension[8] (VC is shorter)
- VC dimension is one possible notion of complexity of the class $\mathcal{H}$
- we introduce it in the setting of binary classification

[8] Vapnik, Chervonenkis, *On the uniform convergence of relative frequencies of events to their probabilities*, Theory of Probability and its Applications, 1971

Shattering

Definition (restriction of $\mathcal{H}$ to C): let $\mathcal{H}$ be a set of functions $\mathcal{X} \to \{0,1\}$ and $C = \{c_1, \ldots, c_n\} \subseteq \mathcal{X}$. The restriction of $\mathcal{H}$ to C is the set of functions from C to $\{0,1\}$ that can be derived from $\mathcal{H}$. That is,

$$\mathcal{H}_C := \{(h(c_1) \ldots, h(c_n)) \text{ s.t. } h \in \mathcal{H}\},$$

- **Remark:** we can represent functions $C \to \{0,1\}$ as vectors of $\{0,1\}^n$

Definition (shattering): A hypothesis class $\mathcal{H}$ shatters a finite set $C \subseteq \mathcal{X}$ if the restriction of $\mathcal{H}$ to C is the set of all functions from C to $\{0,1\}$.

Example: threshold functions

- define the set of threshold functions

$$\mathcal{H} = \{h_a : t \mapsto \mathbb{1}_{t>a}, a \in \mathbb{R}\}.$$

- let $C = \{c_1\}$:
  - if $a = c_1 + 1$, $h_a(c_1) = 1$;
  - if $a = c_1 - 1$, $h_a(c_1) = 0$.
- therefore $\mathcal{H}_C$ is the set of all functions from $C = \{c_1\}$ to $\mathcal{H}$, we say that $\mathcal{H}$ **shatters** C
- now let us set $C = \{c_1, c_2\}$, with $c_1 < c_2$
- **We can not get the labeling** $(1, 0)$
- indeed, if $h_a(c_2) = 0$, it means that $a > c_2 > c_1$ and therefore $h_a(c_1) = 0$
- we say that $\mathcal{H}$ **does not shatter** C

VC dimension

Definition (VC dimension): the VC dimension of a hypothesis class $\mathcal{H}$, denoted $VC(\mathcal{H})$, is the maximal size of a set C that can be shattered by $\mathcal{H}$. If $\mathcal{H}$ can shatter sets of arbitrary large size, we say that $\mathcal{H}$ has infinite VC dimension.

- **Intuition:** complex hypothesis classes can assign arbitrary labels to any subset of $\mathcal{X}$

Theorem: Let $\mathcal{H}$ be a class of infinite VC dimension. Then $\mathcal{H}$ is not PAC learnable.

- we will see that the converse is true: finite VC dimension guarantees PAC learnability

4.2. Examples

Examples

- we now see more examples
- general proof recipe to show that $VC(\mathcal{H}) = d$:
  1. show that there exists a set C that is shattered by $\mathcal{H}$;
  2. show that every set of size $d + 1$ is not shattered by $\mathcal{H}$.
- general idea: find something symmetrical, then add one point
- quite difficult to explain in words, make a lot of drawings!

Threshold functions

- recall that we considered
$$\mathcal{H} = \{h_a : t \mapsto \mathbb{1}_{t>a}, a \in \mathbb{R}\}.$$
- we showed that (i) $\mathcal{H}$ shatters all sets of size 1
- but (ii) there exists sets of size 2 that are not shattered by $\mathcal{H}$
- in conclusion, $VC(\mathcal{H}) = 1$

Bump functions

- let $\mathcal{H}$ be the class defined by

$$\mathcal{H} := \{h_{a,b} : t \mapsto \mathbb{1}_{t\in[a,b]}, \text{ with } a < b\}.$$

- take $C = \{c_1, c_2\}$
- then $h_{c_1-1,(c_1+c_2)/2}$ gives the labeling $(1, 0)$
- the other cases are similar: $\mathcal{H}$ shatters sets of size 2
- now set $C = \{c_1, c_2, c_3\}$, it is not possible to obtain the labeling $(1, 0, 1)$
- we deduce that $VC(\mathcal{H}) = 2$

Axis aligned rectangles

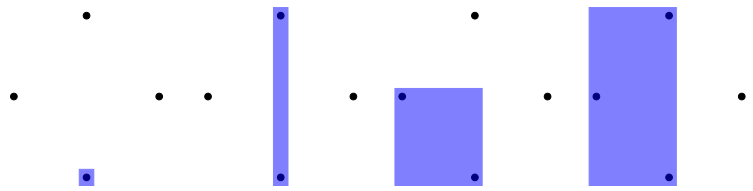
Exercise: Set $\mathcal{X} = \mathbb{R}^2$ and let $\mathcal{H}$ be the class of axis aligned rectangles

$$\mathcal{H} := \left\{ h_{a_1,a_2,b_1,b_2} : x \mapsto \mathbb{1}_{a_1 \leq x_1 \leq a_2 \text{ and } b_1 \leq x_2 \leq b_2}, \text{ with } a_1 < a_2 \text{ and } b_1 < b_2 \right\}.$$

Show that $VC(\mathcal{H}) = 4$.
Extend this proof to the d-dimensional case: what is the VC dimension of the class of axis-aligned rectangles in dimension d?

Correction of the exercise

1. It is easy to show that a diamond-shaped 4-point configuration can be shattered by axis-aligned rectangles in $\mathbb{R}^2$:



Moreover, adding one point inside is not an option: label 1 for the outside points and label 0 inside cannot be achieved. The other cases are similar and we deduce that $VC(\mathcal{H}) = 4$.

Correction of the exercise, ctd.

2. Let us make generalize the previous construction. Formally, the class of axis-aligned rectangles in dimension d is defined by

$$\mathcal{H} = \{h_{(a_1,b_1,\ldots,a_d,b_d)} \text{ s.t. } a_j \leq b_j \ \forall 1 \leq j \leq d\},$$

where $h_{(a_1,b_1,\ldots,a_d,b_d)}(x) = \prod_{j=1}^{d} \mathbb{1}_{x_j \in [a_j,b_j]}$. We are going to show that $VC(\mathcal{H}) = 2d$.
Finding a set of size $2d$ that can be shattered. Consider the set $\{x_1, \ldots, x_{2d}\}$, where $x_i = e_i$ if $1 \leq i \leq d$ and $x_i = -e_{i-d}$ if $i > d$. Let us show that it can be shattered and consider an arbitrary labeling $(y_1, \ldots, y_{2d}) \in \{0,1\}^{2d}$. Let us set $a_i = -2$ if $y_{i+d} = 1$ and $a_i = 0$ otherwise, and $b_i = 2$ if $y_i = 1$ and $b_i = 0$ otherwise. Then $a_i \leq b_i$ for any i and $h_{(a_1,b_1,\ldots,a_d,b_d)}(x_i) = y_i$ for any i.

Correction of the exercise, ctd.

Showing that no set of size $2d+1$ can be shattered. Let C be a set of size $2d+1$. On each dimension, there is a point realizing the minimum and another realizing the maximum. There are $2d$ such points and $2d+1$ in the set, thus there exists s $x \in C$ such that (i) for all j, there exists $x' \in C$ with $x'_j \leq x_j$, and (ii) for all j there exists $x'' \in C$ such that $x''_j > x_j$. The labeling 0 for x and 1 for all other elements of C cannot be obtained since x is inside the rectangle. □