# CS-E4710 Machine Learning: Supervised Methods

Lecture 3: Learning with infinite hypothesis classes

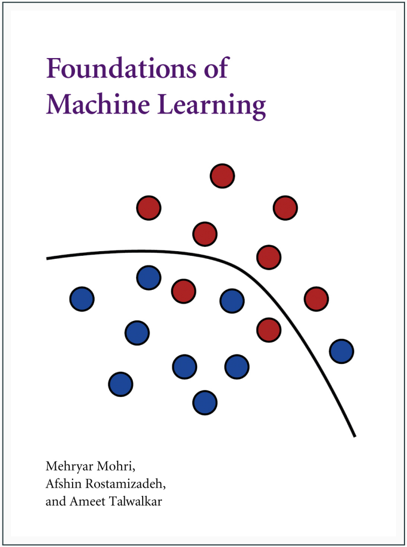
Juho Rousu

September 22, 2020

Department of Computer Science
Aalto University

# Additional reading

- Mohri et al: chapter 3



Foundations of
Machine Learning

Mehryar Mohri,
Afshin Rostamizadeh,
and Ameet Talwalkar

# Recall: PAC learnability

- A class $C$ is PAC-learnable, if there exist an algorithm $\mathcal{A}$ that given a training sample $S$ outputs a hypothesis $h_S$ that has generalization error satisfying
$$Pr(R(h_S) \leq \epsilon) \geq 1 - \delta$$
- for any distribution $D$, for arbitrary $\epsilon, \delta > 0$ and sample size $m = |S|$ that grows at polynomially in $1/\epsilon$,$1/\delta$

# Recall: PAC learning of a finite hypothesis class

- Sample complexity bound relying on the size of the hypothesis class (Mohri et al, 2012): $Pr(R(h_s) \leq \epsilon) \geq 1 - \delta$ if

$$m \geq \frac{1}{\epsilon}(\log(|\mathcal{H}|) + \log(\frac{1}{\delta}))$$

- An equivalent generalization error bound:

$$R(h) \leq \frac{1}{m}(\log(|\mathcal{H}|) + \log(\frac{1}{\delta}))$$

- Holds for any finite hypothesis class assuming there is a consistent hypothesis
- Extra term compared to the "family car" example is the term $\frac{1}{\epsilon}(\log(|\mathcal{H}|))$
- The more hypotheses there are in $|\mathcal{H}|$, the more training examples are needed

# Learning with infinite hypothesis classes

- The size of the hypothesis class is a useful measure of complexity for **finite** hypothesis classes (e.g boolean formulae)
- However, most classifers used in practise rely on infinite hypothesis classes, e.g.
    - $\mathcal{H} =$ axis-aligned rectangles in $\mathbb{R}^2$ (our "family car"!)
    - $\mathcal{H} =$ hyperplanes in $\mathbb{R}^d$ (e.g. Support vector machines)
    - $\mathcal{H} =$ neural networks with continuous input variables
- Need better tools to analyze these cases

# Vapnik-Chervonenkis dimension

# Intuition

- VC dimension can be understood as measuring the capacity of a hypothesis class to adapt to different concepts
- It can be understood through the following thought experiment:
    - Pick a fixed hypothesis class $\mathcal{H}$, e.g. axis-aligned rectangles in $R^2$
    - Let as enumerate all possible labelings of a training set of size $m$: $\mathcal{Y}^m = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{2^m}\}$, where $\mathbf{y}_j = (y_{j1}, \dots, y_{jm})$, and $y_{ij} \in \{0, 1\}$ is the label of $i$'th example in the $j$'th labeling
    - We are allowed to freely choose a distribution $D$ generating the inputs and to generate the input data $x_1, \dots, x_m$
    - $VCdim(\mathcal{H})$ = size of the **largest training set** that we can find a consistent classifier for **all labelings** in $\mathcal{Y}^m$
- Intuitively:
    - low $VCdim \implies$ easy to learn, low sample complexity
    - high $VCdim \implies$ hard to learn, high sample complexity
    - infinite $VCdim \implies$ cannot learn in PAC framework

# Shattering

- The underlying concept in VC dimension is **shattering**
- Given a set of points $S = \{x_1, \dots, x_m\}$ and a fixed class of functions $\mathcal{H}$
- $\mathcal{H}$ is said to **shatter** $S$ if for any possible partition of $S$ into positive $S_+$ and negative subset $S_-$ we can find a hypothesis for which $h(x) = 1$ if and only if $x \in S_+$

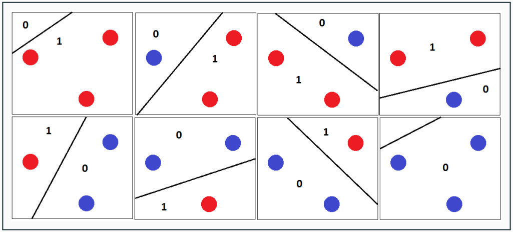


Figure source:

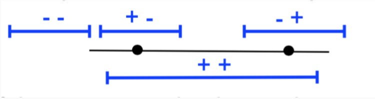
https://datascience.stackexchange.com

# How to show that $VCdim(\mathcal{H}) = d$

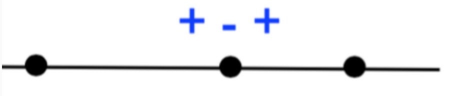
- How to show that $VCdim(\mathcal{H}) = d$ for a hypothesis class
- We need to show that
  - There exists a set of inputs of size $d$ that can be shattered by hypothesis in $\mathcal{H}$ (i.e. we can pick the set of inputs): $VCdim(\mathcal{H}) \geq d$
  - There does not exist any set of size $d + 1$ that can be shattered (i.e. need to show a general property): $VCdim(\mathcal{H}) < d + 1$

# Example: intervals on a real line

- Let the hypothesis class be intervals in $R$
- Each hypothesis is defined by two parameters $b_h, e_h \in \mathbb{R}$: the beginning and end of the interval, $h(x) = \mathbf{1}_{b_h \leq x \leq e_h}$
- We can shatter any set of two points by changing the end points of the interval:



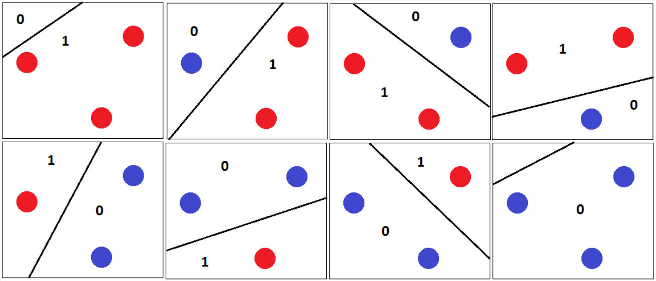
- We cannot shatter a three point set, as the middle point cannot be excluded while the left-hand and right-hand side points are included



We conclude that VC dimension for real intervals $= 2$

# Lines in $\mathbb{R}^2$

- A hypothesis class of lines $h(x) = ax + b$ shatters a set of three points $\mathbb{R}^2$.

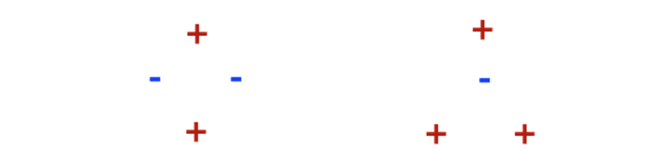


- We conclude that VC dimension is $\geq 3$

# Lines in $\mathbb{R}^2$

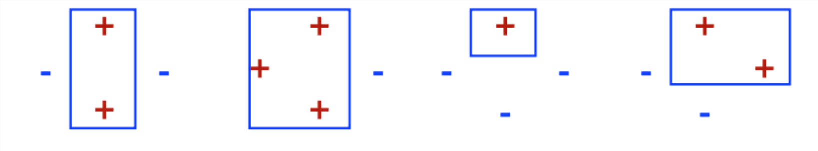
Four points cannot be shattered by lines in $\mathbb{R}^2$:

- Three are only two possible configurations of four points in $\mathbb{R}^2$:
  1. All four points reside on the convex hull
  2. Three points form the convex hull and one is in interior
- In the first case (left), we cannot draw a line separating the top and bottom points from the left-and and right-hand side points
- In the second case, we cannot separate the interior point from the points on the convex hull with a line
- The two examples are sufficient to show that $VCdim = 3$



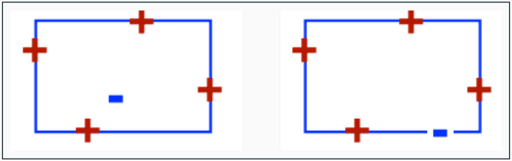
# VC-dimension of axis-aligned rectangles

- What about our "family car" example?
- With axis aligned rectangles we can shatter a set of four points (picture shows 4 of the 16 configurations)
- This implies $VCdim(\mathcal{H}) \geq 4$



# VC-dimension of axis-aligned rectangles

- For five distinct points, consider the minimum bounding box of the points
- There are two possible configurations:
  1. There are one or more points in the interior of the box: then one cannot include the points on the boundary and exclude the points in the interior
  2. At least one of the edges contains two points: in this case we can pick either of the two points and verify that this point cannot be excluded while all the other points are included
- Thus by the two examples we have established that $VCdim(\mathcal{H}) = 4$



# Vapnik-Chervonenkis dimension formally

- Formally $VCdim(\mathcal{H})$ is defined through the growth function

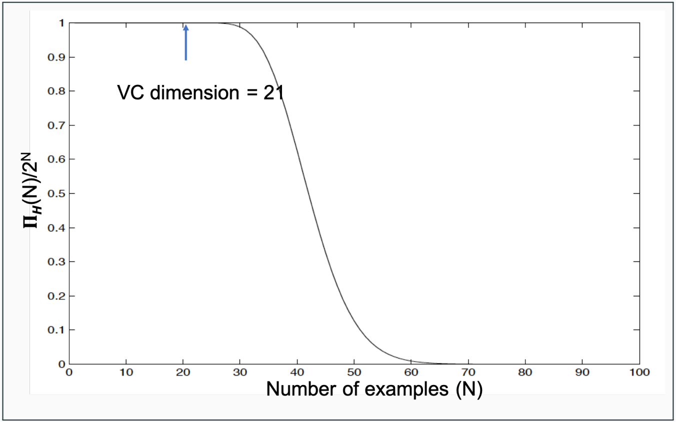
$$\Pi_{\mathcal{H}}(m) = \max_{\{x_1,\ldots,x_m\}\subset X} |\{(h(x_1),\ldots,h(x_m)) : h \in \mathcal{H}\}|$$

- The growth function gives the maximum number of unique labelings the hypothesis class $\mathcal{H}$ can provide for an arbitrary set of input points
- The maximum of the growth function is $2^m$ for a set of $m$ examples
- Vapnik-Chervonenkis dimension is then

$$VCdim(\mathcal{H}) = \max_{m}\{m|\Pi_{\mathcal{H}}(m) = 2^m\}$$

# Visualization

- The ratio of the growth function $\Pi_{\mathcal{H}}(m)$ to the maximum number of labelings of a set of size $m$ is shown
- Hypothesis class is 20-dimensional hyperplanes (VC dimension $= 21$)



# VC dimension of finite hypothesis classes

- A finite hypothesis class have VC dimension $VCdim(\mathcal{H}) \leq \log_2 |\mathcal{H}|$
- To see this:
  - Consider a set of $m$ examples $S = \{x_1, \ldots, x_m\}$
  - This set can be labeled $2^m$ different ways, by choosing the labels $y_i \in \{0, 1\}$ independently
  - Each hypothesis in $h \in \mathcal{H}$ fixes one labeling $(h(x_1), \ldots, h(x_m))$
  - All hypotheses in $\mathcal{H}$ can provide at most $|\mathcal{H}|$ different labelings in total
  - If $|\mathcal{H}| < 2^m$ we cannot shatter $S \implies$ we cannot shatter a set of size $m > \log_2 |\mathcal{H}|$

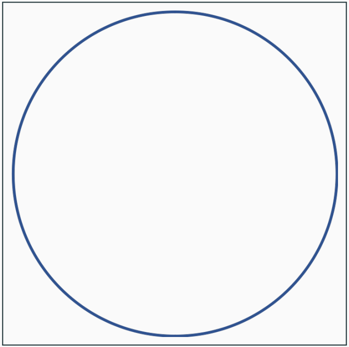
# VC dimension: Further examples

Examples of classes with a finite VC dimension:

- convex $d$-polygons in $\mathbb{R}^2$: $VCdim = 2d + 1$ (e.g. for general, not restricted to axis-aligned, rectangles $VCdim = 5$)
- hyperplanes in $\mathbb{R}^d$: $VCdim = d + 1$ - (e.g. single neural unit, linear SVM)
- neural networks: $VCdim = |E| \log |E||$ where $E$ is the set of edges in the networks (for *sign* activation function)
- boolean monomials of $d$ variables: $VCdim = d$ (Aldo and sports example)
- arbitrary boolean formulae of $d$ variables: $VCdim = 2^d$

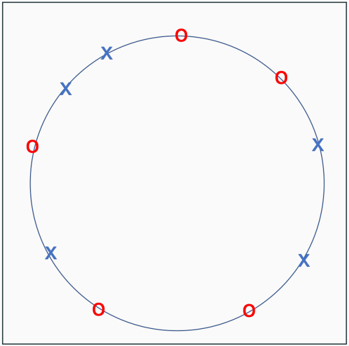
# Convex polygons have VC dimension $= \infty$

- Let our hypothesis class be convex polygons in $\mathbb{R}^2$ without restriction of number of vertices $d$
- Let us draw an arbitrary circle on $\mathbb{R}^2$ - the distribution $D$ will be concentrated on the circumference of the circle
  - This is a difficult distribution for learning polygons - we choose it on purpose



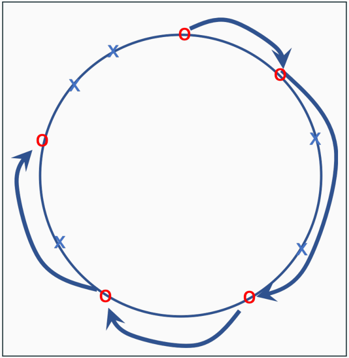
# Convex polygons have VC dimension $= \infty$

- Let us consider a set of *m* points with arbitrary binary labels
- For any *m*, let us position *m* points on the circumference of the circle
  - simulating drawing the inputs from the distribution *D*



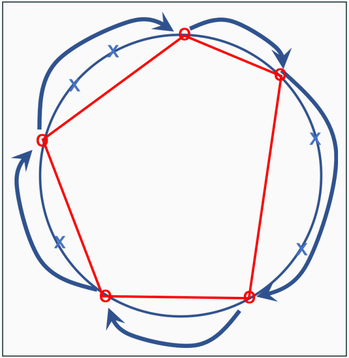
# Convex polygons have VC dimension $= \infty$

- Start from an arbitrary positive point (red circles)
- Traverse the circumference clockwise skipping all negative points and stopping and positive points



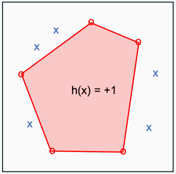
# Convex polygons have VC dimension $= \infty$

- Connect adjacent positive points with an edge
- This forms a *p*-polygon inside the circle, where is the number of positive data points



# Convex polygons have VC dimension $= \infty$

- Define $h(x) = +1$ for points inside the polygon and $h(x) = 0$ outside
- Each of the $2^m$ labelings of $m$ examples gives us a $p$-polygon that includes the $p$ positive points in that labeling and excludes the negative points $\implies$ we can shatter a set of size $m$: $VCdim(\mathcal{H}) \geq m$
- Since $m$ was arbitrary, we can grow it without limit $VCdim(\mathcal{H}) = \infty$



# Generalization bound based on the VC-dimension

- (Mohri, 2012) Let $\mathcal{H}$ be a family of functions taking values in $\{-1, +1\}$ with VC-dimension $d$. Then for any $\delta > 0$, with probability at least $1 - \delta$ the following holds for all $h \in \mathcal{H}$:

$$R(h) \leq \hat{R}(h) + \sqrt{\frac{2\log(em/d)}{m/d}} + \sqrt{\frac{\log(1/\delta)}{2m}}$$

- $e \approx 2.71828$ is the base of the natural logarithm
- The bound reveals that the critical quantity is $m/d$, i.e. the number of examples divided by the VC-dimension
- Manifestation of the Occam's razor principle: to justify an increase in the complexity, we need reciprocally more data

# Rademacher complexity

# Experiment: how well does your hypothesis class fit noise?

- Consider a set of training examples $S_0 = \{(x_i, y_i)\}_{i=1}^m$
- Generate $M$ new datasets $S_1, \dots, S_M$ from $S_0$ by randomly drawing a new label $\sigma \in \mathcal{Y}$ for each training example in $S_0$

$$S_k = \{(x_i, \sigma_{ik})\}_{i=1}^m$$

- Train a classifier $h_k$ minimizing the empirical risk on training set $S_k$, record its empirical risk

$$\hat{R}(h_k) = \frac{1}{m} \sum_{i=1}^m \mathbf{1}_{h_k(x_i) \neq \sigma_{ik}}$$

- Compute the average empirical risk over all datasets: $\bar{\epsilon} = \frac{1}{M} \sum_{k=1}^M \hat{R}(h_k)$

# Experiment: how well does your hypothesis class fit noise?

- Observe the quantity

$$\hat{\mathcal{R}} = \frac{1}{2} - \bar{\epsilon}$$

- We have $\hat{\mathcal{R}} = 0$ when $\bar{\epsilon} = 0.5$, that is when the predictions correspond to random coin flips (0.5 probability to predict either class)
- We have $\hat{\mathcal{R}} = 0.5$ when $\bar{\epsilon} = 0$, that is when all hypotheses $h_i, i = 1, \ldots, M$ have zero empirical error (perfect fit to noise, not good!)
- Intuitively we would like our hypothesis
  - to be able to separate noise from signal - to have low $\hat{\mathcal{R}}$
  - have low empirical error on real data - otherwise impossible to obtain low generalization error

# Rademacher complexity

- Rademacher complexity defines complexity as the capacity of hypothesis class to fit random noise
- For binary classification with labels $\mathcal{Y} = \{-1, +1\}$ empirical Rademacher complexity can be defined as

$$\hat{\mathcal{R}}_S(\mathcal{H}) = \frac{1}{2} E_{\boldsymbol{\sigma}} \big( \sup_{h \in \mathcal{H}} \frac{1}{m} \sum_{t=1}^{m} \sigma^i h(\mathbf{x}_i) \big)$$

- $\sigma_i \in \{-1, +1\}$ are Rademacher random variables, drawn independently from uniform distribution (i.e. $Pr\{\sigma = 1\} = 0.5$)
- Expression inside the expectation takes the highest correlation over all hypothesis in $h \in \mathcal{H}$ between the random true labels $\sigma_i$ and predicted label $h(\mathbf{x}_i)$

# Rademacher complexity

$$\hat{\mathcal{R}}_S(\mathcal{H}) = \frac{1}{2} E_{\boldsymbol{\sigma}} \big( \sup_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^{m} \sigma^i h(\mathbf{x}_i) \big)$$

- Let us rewrite $\hat{\mathcal{R}}_S(\mathcal{H})$ in terms of empirical error
- Note that with labels $\mathcal{Y} = \{+1, -1\}$,

$$\sigma_i h(\mathbf{x}_i) = \begin{cases} 1 & \text{if } \sigma_t = h(\mathbf{x}_i) \\ -1 & \text{if } \sigma_i \neq h(\mathbf{x}_i) \end{cases}$$

- Thus

$$\begin{aligned} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(\mathbf{x}_i) &= \frac{1}{m} (\sum_i \mathbf{1}_{\{\mathbf{h}(\mathbf{x}_i) = \sigma_i\}} - \sum_i \mathbf{1}_{\{\mathbf{h}(\mathbf{x}_i) \neq \sigma_i\}}) \\ &= \frac{1}{m} (m - 2 \sum_i \mathbf{1}_{\{\mathbf{h}(\mathbf{x}_i) \neq \sigma_i\}}) = 1 - 2\epsilon(\hat{h}) \end{aligned}$$

# Rademacher complexity

- Plug in

$$
\begin{aligned}
\hat{\mathcal{R}}_S(\mathcal{H}) &= \frac{1}{2} E_{\boldsymbol{\sigma}}\big(\sup_{h\in\mathcal{H}}(1 - 2\hat{\epsilon}(h))\big) \\
&= \frac{1}{2}(1 - 2E_{\boldsymbol{\sigma}} \inf_{h\in\mathcal{H}} \hat{\epsilon}(h)) = \frac{1}{2} - E_{\boldsymbol{\sigma}} \inf_{h\in\mathcal{H}} \hat{\epsilon}(h))
\end{aligned}
$$

- Now we have expressed the empirical Rademacher complexity in terms of expected empirical error of classifying randomly labeled data
- But how does the Rademacher complexity help in model selection?
    - We need to relate it to generalization error

# Generalization bound with Rademacher complexity

(Mohri et al. 2012): For any $\delta > 0$, with probability at least $1 - \delta$ over a sample drawn from an unknown distribution $D$, for any $h \in \mathcal{H}$ we have:

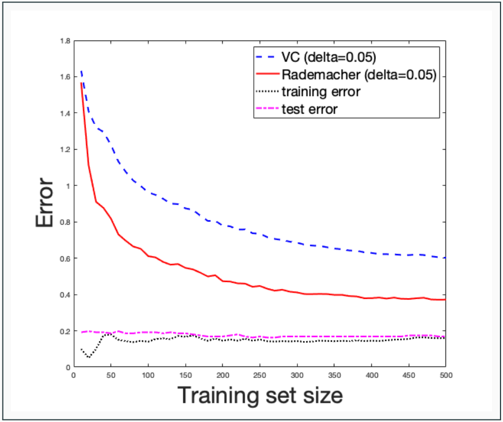
$$R(h) \leq \hat{R}_S(h) + \hat{\mathcal{R}}_S(\mathcal{H}) + 3\sqrt{\frac{\log \frac{2}{\delta}}{2m}}$$

The bound is composed of the sum of :

- The empirical risk of $h$ on the training data $S$ (with the original labels): $\hat{R}_S(h)$
- The empirical Rademacher complexity: $\hat{\mathcal{R}}_S(\mathcal{H})$
- A term that tends to zero as a function of size of the training data as $O(1/\sqrt{m})$ assuming constant $\delta$.

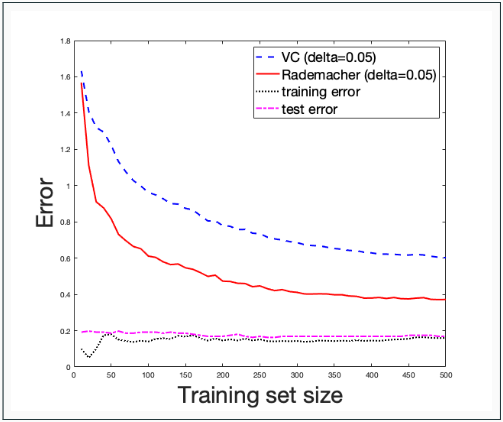
# Example: Rademacher and VC bounds on a real dataset

- Prediction of protein subcellular localization
- 10-500 training examples, 172 test examples
- Comparing Rademacher and VC bounds using $\delta = 0.05$
- Training and test error also shown



# Example: Rademacher and VC bounds on a real dataset

- Rademacher bound is sharper than the VC bound
- VC bound is not yet informative with 500 examples ($> 0.5$) using ($\delta = 0.05$)
- The gap between the mean of the error distribution ($\approx$ test error) and the 0.05 probability tail (VC and Rademacher bounds) is evident (and expected)



# Rademacher vs. VC

Note the differences between Rademacher complexity and VC dimension

- VC dimension is independent of any training sample or distribution generating the data: it measures the worst-case where the data is generated in a bad way for the learner
- Rademacher complexity depends on the training sample thus is dependent on the data generating distribution
- VC dimension focuses the extreme case of realizing all labelings of the data
- Rademacher complexity measures smoothly the ability to realize random labelings

# Rademacher vs. VC

- Generalization bounds based on Rademacher Complexity are applicable to any binary classifiers (SVM, neural network, decision tree)
- It motivates state of the art learning algoritms such as support vector machines
- But computing it might be hard, if we need to train a large number of classifiers
- Vapnik-Chervonenkis dimension (VCdim) is an alternative that is usually easier to derive analytically

# Summary: Statistical learning theory

- Statistical learning theory focuses in analyzing the generalization ability of learning algorithms
- Probably Approximately Correct framework is the most studied theoretical framework, asking for bounding the generaliation error ($\epsilon$) with high probability ($1 - \delta$), with arbitrary level of error $\epsilon > 0$and confidence $\delta > 0$
- Vapnik-Chervonenkis dimension lets us study learnability infinite hypothesis classes through the concept of shattering
- Rademacher complexity is a practical alternative to VC dimension, giving typically sharper bounds (but requires a lot of simulations to be run)