CS5339 Lecture Notes #9: Statistical Learning Theory

Jonathan Scarlett

March 31, 2021

Useful references:

- Blog post by Jeremy Kun[1]
- Bousquet *et al.*'s "Introduction to Statistical Learning Theory" notes[2]
- Supplementary notes lec9.pdf and lec14a.pdf
- Simons Berkeley video lectures on Generalization[3]
- Part I of"Understanding Machine Learning" book, especially Chapters 3–6

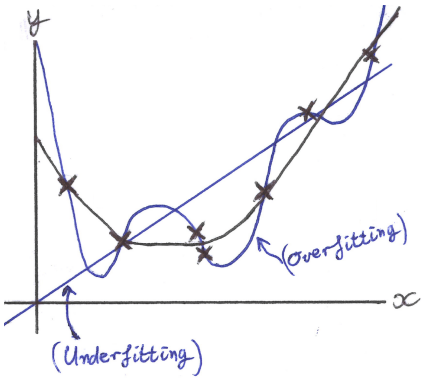
1 Introduction

Motivation:

- We have discussed a variety of classification and regression methods, typically based on attaining small error on *training data* (this is the data set that we plug into the algorithm we are using)
- What we are more interested in is the performance on *unseen data* (e.g., it is no good to classify all previous spam emails correctly if we get most future ones wrong)
- Hence, to get a better idea of performance in a practical scenario we should check the performance on *test data* (data that is not used by the algorithm)
- We have discussed how the complexity of the classifier can impact the performance:
  - If too simple, we may underfit (be unable to capture the patterns in the data)
  - If too complex, we may overfit (capture spurious patterns that are just due to fluctuations and noise), and do poorly on test data despite having low training error

  We also discussed how regularization can help with the latter of these.

[1] http://jeremykun.com/2014/09/19/occams-razor-and-pac-learning/
[2] http://www.kyb.mpg.de/fileadmin/user_upload/files/publications/pdfs/pdf2819.pdf
[3] https://simons.berkeley.edu/workshops/schedule/10624



- In this lecture, we take a theoretical view of the trade-off between complexity and generalization (generalization meaning the gap between train vs. test performance). By doing so, we can get a better understanding of when we might be underfitting or overfitting, and how to avoid doing so.

Abstract setup of statistical learning:

- The learner has access to a data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, drawn from $(\mathbf{x}_i, y_i) \sim P_{\mathbf{X}Y}$ independently.
  - Importantly, the distribution $P_{\mathbf{X}Y}$ is considered to be *unknown*
- The learner uses the data to decide on a classification function $f(\mathbf{x})$ mapping inputs to labels (i.e., to $\{-1, 1\}$ for binary classification, or $\mathbb{R}$ for regression). The set of all possible functions under consideration is called the *function class* $\mathcal{F}$:[4]
  - e.g., set of all linear classifiers
  - e.g., set of polynomial functions up to order p in regression
  - e.g., set of all classifiers that can be expressed by a weighted vote of $M = 1000$ decision stumps
- The performance is measured according to a *loss function* $\ell(y, f(\mathbf{x}))$ that we hope to be small
  - Particularly common for classification: 0-1 loss $\ell(y, f(\mathbf{x})) = \mathbb{1}\{y \neq f(\mathbf{x})\}$
  - Particularly common for regression: Squared loss $\ell(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$
  - Others we have seen: Hinge loss, logistic loss, exponential loss, etc.
- For a given choice of function $f \in \mathcal{F}$, the expected loss is given by

$$R(f) = \mathbb{E}[\ell(y, f(\mathbf{x}))]$$

  with $(\mathbf{x}, y) \sim P_{\mathbf{X}Y}$. This is called the *(true) risk*.
- If the distribution $P_{\mathbf{X}Y}$ were known, then one could find the *Bayes-optimal* classifier, given by

$$f^* = \arg\min_{f \in \mathcal{F}} \mathbb{E}[\ell(y, f(\mathbf{x}))]. \tag{1}$$

[4] In the literature, you will often see references to a *hypothesis class* $\mathcal{H}$, with general loss functions of the form $\ell_h(\mathbf{x}, y)$ for $h \in \mathcal{H}$. This is a more general formulation capturing other problems like clustering and density estimation. Since we are mainly interested in classification and regression, we focus on the "function class" formulation.

Since the distribution is unknown, it is natural to replace the expectation by the empirical average over the training set:

$$f_{\text{erm}} = \arg\min_{f \in \mathcal{F}} R_n(f) \qquad R_n(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, f(\mathbf{x}_i)). \tag{2}$$

This is known as the *empirical risk minimization* (ERM) rule.

- In general, the ERM rule may be computationally infeasible to implement (e.g., 0-1 loss), but it is still of interest to understand its theoretical properties. Even for loss functions where ERM is infeasible, we can use a *surrogate loss* that is computationally feasible, and still infer results about the original loss function (see Section 12.3 of "Understanding Machine Learning").
- We have already seen a few examples of ERM in the special case that $\mathcal{F}$ is the class of linear classifiers (or linear predictors for regression):
  - If $\ell(\cdot, \cdot)$ is the logistic loss, we recover the maximum likelihood rule for logistic regression
  - If $\ell(\cdot, \cdot)$ is the squared loss in regression, then we recover the least squares rule (which, unlike most, has a closed-form expression)
  - More generally, when we set $\ell(\cdot, \cdot)$ to be the negative log-likelihood for any model, ERM becomes the maximum-likelihood rule
- Regularization (e.g., to reduce overfitting) is of significant interest in this broad framework, but we will only study the non-regularized ERM rule.

- **A decomposition of the risk.** As above, when we define the true risk and empirical risk as

$$R(f) := \mathbb{E}[\ell(y, f(\mathbf{x}))], \qquad R_n(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, f(\mathbf{x}_i)),$$

we can make use of the following useful decomposition:

$$\underbrace{R(f)}_{\text{test error}} = \underbrace{R_n(f)}_{\text{training error}} + \underbrace{\big(R(f) - R_n(f)\big)}_{\text{generalization error}}.$$

- $R(f)$ is called the test error because it is the average error we get on an *unseen* sample (drawn from the same distribution as the training data $\mathcal{D}$).
- Evidently, small training error does not imply small test error, because the generalization error might be large (in particular, we may be *overfitting*)
- Hence, a fundamental question is *when can we ensure small generalization error*?

2 Probably Approximately Correct (PAC) Learning

- While the following definition is rather technical, the general idea is rather simple:
  - Seek to attain a (true) risk within a small value ϵ of that achieved by the best f in the function class being considered;

- Since the data is random, we can't expect this to happen with probability one, so instead only insist on probability $1-\delta$;
- The number of samples required depends on both ϵ and δ, and also critically on the function class $\mathcal{F}$ (the richer $\mathcal{F}$ is, the harder it is to get close to the best $f \in \mathcal{F}$).

- **Definition.** Given a loss function $\ell(\cdot,\cdot)$, a function class $\mathcal{F}$ is said to be *PAC-learnable* if there exists an algorithm $\mathcal{A}(\mathcal{D}_n)$ (with $\mathcal{D}_n$ containing n independent samples) and a function $\overline{n}(\epsilon,\delta)$ such that the following holds: For any distribution $P_{\mathbf{X}Y}$ used to generate $\mathcal{D}$, and any $\epsilon,\delta \in (0,1)$, if $n \geq \overline{n}(\epsilon,\delta)$, then the following holds with probability at least $1-\delta$:

$$R(\hat{f}) \leq \min_{f\in\mathcal{F}} R(f) + \epsilon.$$

  - The probability $1-\delta$ corresponds to *probably* correct
  - The ϵ gap in the risk corresponds to *approximately* correct
  - The function $\overline{n}(\epsilon,\delta)$ is called the *sample complexity*

- Often a distinction is made between two settings:
  - (Realizable setting) It holds that $y = f(\mathbf{x})$ for some $f \in \mathcal{F}$, i.e., there exists a "perfect" classifier that classifies everything correctly. See the tutorial for a more detailed treatment of this case.
  - (Agnostic setting) There is no function in $\mathcal{F}$ that classifies perfectly, so we only try to compete with the best in $\mathcal{F}$. The results we give below are suited to this case.

  Two examples: (i) It is easy to imagine a scenario where even the best linear classifier still has, say, a 10% error rate. (ii) If Y is not deterministic given $\mathbf{X}$, then the inherent randomness means that we *must* incur some probability of being wrong.

- Sometimes people define PAC learnability with the extra condition that the computation time of $\mathcal{A}$ is polynomial in n, $\frac{1}{\delta}$ and $\frac{1}{\epsilon}$. We consider the above definition where this is not required.

3 PAC Learnability for Finite Hypothesis Classes

- Suppose that the loss function is bounded, and specifically satisfies

$$\ell(y, f(\mathbf{x})) \in [0,1]$$

  for all $f \in \mathcal{F}$, with probability one.[5] In addition, suppose that the function class is finite, i.e., $\mathcal{F}$ is a finite set.

  - **Theorem.** For any bounded loss function in $[0,1]$, any finite function class $\mathcal{F}$ is PAC-learnable with sample complexity $\overline{n}(\epsilon,\delta) = \frac{2}{\epsilon^2}\log\frac{2|\mathcal{F}|}{\delta}$.
  - Note 1: Stronger bounds (with $\frac{1}{\epsilon}$ dependence instead of $\frac{1}{\epsilon^2}$) are known in the realizable case.

[5] Any loss function taking values in $[a,b]$ can be shifted and scaled to an "equivalent" loss function with values in $[0,1]$.

– Note 2: The sample complexity is in the worst case sense with respect to all possible $P_{\mathbf{X}Y}$. Because of this, the bound can often be overly pessimistic for "easier" $P_{\mathbf{X}Y}$.

– Note 3: We will prove this result using the ERM rule as our choice of $\mathcal{A}(\mathcal{D}_n)$.

– Note 4: Solving $n = \frac{2}{\epsilon^2}\log\frac{2|\mathcal{F}|}{\delta}$ for ϵ gives $\epsilon = \sqrt{\frac{2}{n}\log\frac{2|\mathcal{F}|}{\delta}}$, so the theorem statement is equivalent to saying that

$$R(f_{\text{erm}}) - \min_{f\in\mathcal{F}} R(f) \leq \sqrt{\frac{2}{n}\log\frac{2|\mathcal{F}|}{\delta}} \tag{3}$$

with probability at least $1-\delta$.

- We proceed with the proof. For any fixed function $f \in \mathcal{F}$, the generalization error satisfies

$$R(f) - R_n(f) = \mathbb{E}[\ell(y, f(\mathbf{x}))] - \frac{1}{n}\sum_{i=1}^{n}\ell(y_i, f(\mathbf{x}_i))$$

by definition, and hence

$$\mathbb{P}\big[\big|R(f) - R_n(f)\big| \geq \epsilon_0\big] = \mathbb{P}\bigg[\bigg|\frac{1}{n}\sum_{i=1}^{n}\ell(y_i, f(\mathbf{x}_i)) - \mathbb{E}[\ell(y, f(\mathbf{x}))]\bigg| \geq \epsilon_0\bigg].$$

- Recall that $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ are i.i.d. with some joint distribution $P_{\mathbf{X}Y}$. Hence, the random variables $\ell(y_i, f(\mathbf{x}_i))$ for $i = 1, \ldots, n$ are i.i.d., bounded in $[0,1]$, and have mean $\mathbb{E}[\ell(y, f(\mathbf{x}))]$. Therefore, we can bound the above probability using Hoeffding's inequality from the previous lecture:

$$\mathbb{P}\big[\big|R(f) - R_n(f)\big| \geq \epsilon_0\big] \leq 2e^{-2n\epsilon_0^2}. \tag{4}$$

– Problem. We would like to apply this concentration bound to the selected f_{erm}, but simply substituting $f = f_{\text{erm}}$ in (4) would not be valid. This is because (4) holds for any *fixed* choice of f, but f_{erm} is not fixed – it is a random variable (because it depends on the data set $\mathcal{D}$).

– Solution. Bound the probability that there exists *any* $f \in \mathcal{F}$ satisfying $\big|R(f) - R_n(f)\big| \geq \epsilon_0$. If this probability can be made small for any choice of ϵ_0 by a suitably large choice of n, we say that *uniform convergence* holds.

- Since we are considering finite $\mathcal{F}$, we can simply apply the union bound $\mathbb{P}[A_1 \cup \ldots \cup A_m] \leq \sum_{i=1}^m \mathbb{P}[A_i]$ to (4) to obtain

$$\mathbb{P}\bigg[\bigcup_{f\in\mathcal{F}}\Big\{\big|R(f) - R_n(f)\big| > \epsilon_0\Big\}\bigg] \leq 2|\mathcal{F}|e^{-2n\epsilon_0^2}. \tag{5}$$

By setting the right-hand side to a target value δ and re-arranging, we find that a sufficient number of samples is $n = \frac{1}{2\epsilon_0^2}\log\frac{2|\mathcal{F}|}{\delta}$.

– Solving $n = \frac{1}{2\epsilon_0^2}\log\frac{2|\mathcal{F}|}{\delta}$ for ϵ_0 gives $\epsilon_0 = \sqrt{\frac{1}{2n}\log\frac{2|\mathcal{F}|}{\delta}}$, so similarly to (3), an equivalent statement is that $\big|R(f) - R_n(f)\big| \leq \sqrt{\frac{1}{2n}\log\frac{2|\mathcal{F}|}{\delta}}$ for all $f \in \mathcal{F}$, with probability at least $1-\delta$.

– Notice that the size of the function class serves as a measure of its complexity – to get good generalization, it suffices to have $n \gg \log|\mathcal{F}|$ samples.

- In the following, we consider the ERM rule, but in fact we have shown that the generalization error is small (for large enough n) for *any* algorithm that always returns a function in $\mathcal{F}$.

- Now assume that the probability $1-\delta$ event occurs, namely, $\left|R(f) - R_n(f)\right| \leq \epsilon_0$ for all $f \in \mathcal{F}$. Letting f^* be the function that minimizes $R(f)$, we have

$$\begin{aligned} R(f_{\text{erm}}) - R(f^*) &= \underbrace{R(f_{\text{erm}}) - R_n(f_{\text{erm}})}_{\leq \epsilon_0} + \underbrace{R_n(f_{\text{erm}}) - R_n(f^*)}_{\leq 0} + \underbrace{R_n(f^*) - R(f^*)}_{\leq \epsilon_0} \\ &\leq 2\epsilon_0, \end{aligned}$$

  where the middle ≤ 0 claim is by the definition of f_{erm}.

- Setting $\epsilon_0 = \frac{\epsilon}{2}$ gives the desired bound $R(f_{\text{erm}}) - R(f^*)$ in the definition of PAC learnability, and yields a number of samples given by $n = \frac{2}{\epsilon^2} \log \frac{2|\mathcal{F}|}{\delta}$. This proves the theorem.

4 Infinite Hypothesis Classes and the VC Dimension

Overview.

- In most cases, we have infinitely many classifiers to potentially choose from (e.g., the set of all linear classifiers of the form $f_{\boldsymbol{\theta}}(\mathbf{x}) = \text{sign}(\boldsymbol{\theta}^T \mathbf{x})$). So the analysis of the previous section does not apply.

  - **Note.** Despite this, finite classes can certainly arise naturally – imagine a scenario where 100 researchers independently develop 100 classifiers using their own data, and our goal is to select the best one among them using our own independent data, hence $|\mathcal{F}| = 100$.

- VC theory [Vapnik and Chervonenkis, 1971] provides analogous bounds for infinite function classes in the *binary setting* (i.e., $y \in \{+1, -1\}$), with the 0-1 *loss* ($\ell(y, f(\mathbf{x})) = \mathbb{1}\{f(\mathbf{x}) \neq y\}$).

- **Intuition.** Even with infinitely many hypotheses, there may be only finitely many *effective hypotheses*.

- **Motivating example 1.**

  - Consider 1D data with binary labels (i.e., $x \in \mathbb{R}$ and $y \in \{-1, 1\}$), and suppose that $\mathcal{F}$ is the class of all threshold classifiers: There exists a threshold $a \in \mathbb{R}$ such that $f(x) = +1$ for $x \leq a$, and $f(x) = -1$ for $x > a$.

  - If we re-order the points as $x^{(1)} \leq x^{(2)} \leq \ldots \leq x^{(n)}$, then we see that any choice of $a \in (x^{(i)}, x^{(i+1)})$ gives an equivalent classifier (for this data set). Observe that there are at most $n+1$ such classifiers – a finite number!

  - Hence, intuitively (but not quite formally according to this argument), we should expect to get a similar PAC sample complexity to the finite-$\mathcal{F}$ case, with $n+1$ in place of $|\mathcal{F}|$.

- **Motivating example 2.**

  - To make things a little less straightforward, assume now that $\mathcal{F}$ contains all interval classifiers: There are two numbers a and b such that $f(x) = +1$ for $x \in [a, b]$, and $f(x) = -1$ otherwise.

- Again, any $a \in (x^{(i)}, x^{(i+1)})$ gives an equivalent classifier (when b is fixed), and similarly for b (when a is fixed). Hence, for a given data set, there are at most $(n+1)^2$ possible classifiers.
- So by the same reasoning as the first example, we should expect $(n+1)^2$ in place of $\mathcal{F}$.
- Generalizing this intuition, we can expect a result with $(n+1)^{d_{\text{VC}}}$ in place of $|\mathcal{F}|$, where d_{VC} is a quantity known as the *VC dimension*. Taking logs, this means we can expect $d_{\text{VC}} \log(n+1)$ in place of $\log|\mathcal{F}|$; in fact, with more sophisticated reasoning (not covered here), the $\log(n+1)$ term can be avoided.

Formal definition and statement.

- **Definitions.**
  1. For an integer n, the *growth function* $S_n(\mathcal{F})$ (also known as *shattering number*) is $S_n(\mathcal{F}) = \sup_{\mathbf{x}_1,\ldots,\mathbf{x}_n} \left|\left\{(f(\mathbf{x}_1),\ldots,f(\mathbf{x}_n)) : f \in \mathcal{F}\right\}\right|$.
     - To be clear, $|\{\cdot\}|$ means the size of the set defined in the $\{\cdot\}$.
     - This is the number of different assignments that hypotheses from $\mathcal{F}$ can make to $\mathbf{x}_1,\ldots,\mathbf{x}_n$. It is an integer between 1 and 2^n.
  2. The *VC dimension* $d_{\text{VC}} = d_{\text{VC}}(\mathcal{F})$ of a function class $\mathcal{F}$ is the largest k such that $S_k(\mathcal{F}) = 2^k$. If $S_k(\mathcal{F}) = 2^k$ for all k, then we define $d_{\text{VC}} = \infty$.
     - A set of points $\mathbf{x}_1,\ldots,\mathbf{x}_k$ satisfying $\left|\left\{(f(\mathbf{x}_1),\ldots,f(\mathbf{x}_k)) : f \in \mathcal{F}\right\}\right| = 2^k$ is said to be *shattered by* $\mathcal{F}$ (i.e., hypotheses from f can produce all 2^k possible assignments)
- A result known as Sauer's lemma states that $S_n(\mathcal{F}) \le \sum_{i=0}^{d_{\text{VC}}} \binom{n}{i}$.
  - Using some standard combinatorial bounds, this can be used to show that

$$S_n(\mathcal{F}) \begin{cases} = 2^n & n \le d_{\text{VC}} \\ \le \left(\frac{d_{\text{VC}}\, e}{n}\right)^{d_{\text{VC}}} & n > d_{\text{VC}}. \end{cases}$$

  - A slightly weaker bound is simply $S_n(\mathcal{F}) \le (n+1)^{d_{\text{VC}}}$, which generalizes the $(n+1)$ and $(n+1)^2$ terms in the motivating examples.
  - Hence, the VC dimension and the general shattering numbers are closely related.
- **Theorem.** If the VC dimension $d_{\text{VC}} = d_{\text{VC}}(\mathcal{F})$ satisfies $d_{\text{VC}} < \infty$, then the function class $\mathcal{F}$ is PAC-learnable under the 0-1 loss with sample complexity

$$\overline{n}(\epsilon, \delta) = C \cdot \frac{d_{\text{VC}} + \log\frac{1}{\delta}}{\epsilon^2} \tag{6}$$

  for some constant C. Conversely, if $d_{\text{VC}} = \infty$ then $\mathcal{F}$ is not PAC-learnable.
  - Hence, d_{VC} serves as a fundamental measure of *richness* of the function class – to get good generalization, it suffices to have $n \gg d_{\text{VC}}$ samples.
  - Even if d_{VC} is infinite, efficient learning might be possible for a *given* data distribution $P_{\mathbf{X}Y}$. The VC theory only establishes the difficulty of *worst-case* distributions.

- For worst-case distributions, the number of samples increasing linearly with d_{VC} (as in (6)) is unavoidable. Intuitively, if d_{VC} points are shattered by $\mathcal{F}$ but only $d_{\text{VC}}/2$ of them are observed in the data set, then there is no way to know the labels of the other half.
- The proof is outlined below. Alternatively, see Section 6.5.2 of 'Understanding Machine Learning" for a simpler analysis, albeit one that leads to a weaker result.

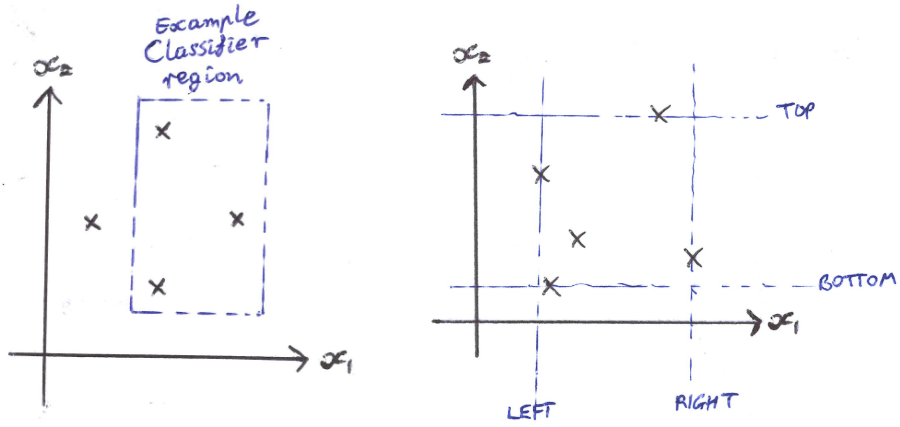
5 VC Dimension Examples

Example 1 (Rectangular classifiers):

- Consider the case of $d = 2$ features, and let $\mathcal{F}$ be the set of all *axis-aligned rectangular classifiers*:

$$h_{a_1,a_2,b_1,b_2} = \begin{cases} +1 & a_1 \le x_1 \le a_2 \text{ and } b_1 \le x_2 \le b_2 \\ -1 & \text{otherwise.} \end{cases}$$

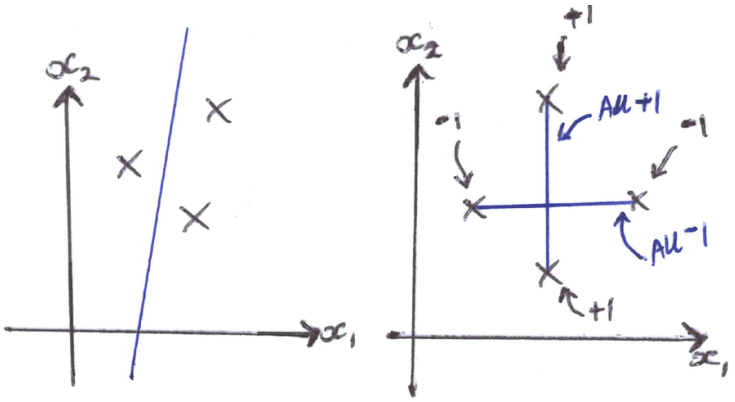
- We will show that $d_{\text{VC}}(\mathcal{F}) = 4$. An illustration:



- The left figure shows a set of 4 points that is clearly shattered by $\mathcal{F}$ (a classification of one assignment is drawn; see if you can convince yourself of the other 15)
- The right figure illustrates why 5 points can never be shattered: If we try to label the left-most, right-most, top-most, and bottom-most to $+1$, then there is no way the final one can be -1.

Example 2 (Linear classifiers):

- For linear classifiers in dimension d, $d_{\text{VC}} = d$ with no offset, or $d_{\text{VC}} = d + 1$ with an offset.
  - Hence, this is a simple case where VC dimension = dimension. Of course, such a property need not hold for classes of non-linear classifiers, as the previous example shows.
- Let's show this for $d = 2$ when there is an offset: (See the tutorial for the general case)
- The left figure shows 3 points that can clearly be shattered (try all 8 combinations)



- The right figure (partially) shows why 4 points can never be shattered: If the left-most and right-most points are -1 and the top-most and bottom-most points are $+1$, then we get a contradiction:
  - If a linear classifier classifies two points the same, it also classifies all points on the line connecting those points that way (this is easily shown by the definition of linearity)
  - But the lines connecting $(+1, +1)$ and $(-1, -1)$ labeled points intersect, so the point of intersection is labeled in two different ways – a contradiction!
- This argument applies to any set of 4 points such that no point lies inside the triangle formed by the other 3. The case that some point *does* lie inside such a triangle is even easier – just label that point '+', and label the 3 points forming the triangle as '$-$'.

Example 3 (Motivating examples):

- Using similar (but simpler) reasoning to the first two examples, it can be shown that the "Motivating examples" (where we argued there are effective at most $(n+1)$ and $(n+1)^2$ classifiers) have function classes with $d_{\text{VC}} = 1$ and $d_{\text{VC}} = 2$ respectively.
- Related examples are explored in the tutorial.

Example 4 (Finite function classes):

- Let's return to the case that $\mathcal{F}$ is finite.
- For a set of k points to be shattered, we need 2^k different assignments, so we certainly need $|\mathcal{F}| \geq 2^k$.
- This means that $d_{\text{VC}}(\mathcal{F}) \leq \log_2 |\mathcal{F}|$, and therefore, the finite function class theorem is a special case of the VC-based theorem (at least up to the constant factor of C in (6))

Example 5 (Infinite VC dimension):

- As a trivial example, suppose that $\mathcal{F}$ is the class of *all* functions from $\mathbb{R}^d$ to $\{-1, 1\}$
- Clearly, we can shatter any set of k distinct points. Hence, the VC dimension is infinite.

- This should be unsurprising, as this function class is far too rich to expect to always achieve the zero error achieved by f^*.
- Many other (non-trivial) examples when $d_{\text{VC}} = \infty$ are also known.

Note: Although we will not cover it, the book 'Understanding Machine Learning" uses the VC dimension and related ideas to provide theoretical performance bounds for some of the algorithms we have studied, including SVM (Section 15.2) and boosting (Section 10.3).

Other approaches to bounding generalization error: There are many other interesting notions in the statistical learning theory literature that can be shown to imply good generalization. We will not cover them, but some examples include:

- A relatively small number of data points being "relevant" for producing the final classifier (think of the support vectors in SVM)
- The classifier attaining a large margin (think SVM, boosting)
- Algorithmic stability (changing one training data point doesn't change the final classifier too much)
- Robustness (if a test sample is "similar" to a training sample, it incurs a similar loss)
- Other more technical notions: Rademacher complexity, PAC-Bayes, mutual information bounds

6 Rademacher Complexity and a Proof of the VC Dimension Based PAC Learnability Result

The following is only a brief outline of a more advanced topic, and is based on the lecture notes `http://web.eecs.umich.edu/~cscott/past_courses/eecs598w14/notes/10_rademacher.pdf`. The textbook "Understanding Machine Learning" in the reference list above has considerably more detail.

Definitions.

- As usual, let $\mathcal{F}$ be a function class consisting of classifiers with $f(\mathbf{x}) \in \{-1, 1\}$,[6] and consider a data set $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ drawn i.i.d. from an unknown distribution $P_{\mathbf{X}Y}$.
- The *empirical Rademacher complexity* of $\mathcal{F}$ with respect to $\mathcal{D}$ is defined as

$$\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F}) = \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i f(\mathbf{x}_i)\right],$$

  where $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$ is an i.i.d. sequence of Rademacher random variables (i.e., $\sigma_i \in \{-1, 1\}$ with probability $\frac{1}{2}$ each). The *(true) Rademacher complexity* of $\mathcal{F}$ is defined as

$$\mathcal{R}_n(\mathcal{F}) = \mathbb{E}_{\mathcal{D}}\big[\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F})\big].$$

[6] These definitions can also be extended to more general real-valued predictors and loss functions.

– Intuitively, Rademacher complexity measures how well the function $\mathcal{F}$ can fit *uniformly random labels* $\{\sigma_i\}$ on average. The richer the function class, the better it can fit random labels, and the higher the Rademacher complexity.

– **Note:** The notation $\mathcal{R}_n(\mathcal{F})$ should not be confused with the empirical risk $R_n(f)$.

- For the analysis, it will be useful to consider the set $\mathcal{L} = \{\ell_f : f \in \mathcal{F}\}$, the set of all 0-1 loss functions $\ell_f(\mathbf{x}, y) = \mathbf{1}\{f(\mathbf{x}) \neq y\}$ induced by some function $f \in \mathcal{F}$. Then we can make the following analogous definitions:

$$\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}) = \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i \ell_f(\mathbf{x}_i, y_i)\right],$$
$$\mathcal{R}_n(\mathcal{L}) = \mathbb{E}_{\mathcal{D}}\big[\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L})\big].$$

Note that the $\sup_{f\in\mathcal{F}}$ can equivalently be replaced by $\sup_{\ell\in\mathcal{L}}$, since $\mathcal{L}$ is just the set of all $\{\ell_f\}_{f\in\mathcal{F}}$.

– **Note:** Here it is more convenient to work with the notation $\ell_f(\mathbf{x}, y)$ instead of $\ell(y, f(\mathbf{x}))$ that was used previously, but the two are equivalent.

- **Claim.** The above definitions satisfy $\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}) = \frac{1}{2}\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F})$ and hence $\mathcal{R}_n(\mathcal{L}) = \frac{1}{2}\mathcal{R}_n(\mathcal{F})$.

– Proof. Using the fact that $\ell_f(\mathbf{x}, y) = \mathbf{1}\{f(\mathbf{x}) \neq y\} = \frac{1-y_i f(\mathbf{x}_i)}{2}$ (just check the two cases of whether or not $y_i = f(\mathbf{x}_i)$), the above definition of $\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L})$ simplifies to

$$\begin{aligned}
\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{L}) &= \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i \frac{1-y_i f(\mathbf{x}_i)}{2}\right] \\
&= \frac{1}{2}\mathbb{E}_{\boldsymbol{\sigma}}\left[\frac{1}{n}\sum_{i=1}^{n}\sigma_i + \sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i(-y_i) f(\mathbf{x}_i)\right] \\
&\overset{(a)}{=} \frac{1}{2}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \sigma_i f(\mathbf{x}_i)\right] \\
&= \frac{1}{2}\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F}),
\end{aligned}$$

where (a) uses the fact that $\mathbb{E}[\sigma_i] = 0$, and also the fact that σ_i and $\sigma_i \cdot (-y_i)$ have the same distribution (both are ± 1 with equal probability).

Relating Rademacher complexity to uniform convergence.

- **Theorem.** For any function class $\mathcal{F}$, the empirical risk $R_n(f)$ and and true risk $R(f)$ satisfy the following uniform convergence bound with probability at least $1 - \delta$:

$$\sup_{f\in\mathcal{F}} \big|R(f) - R_n(f)\big| \leq \mathcal{R}_n(\mathcal{F}) + \sqrt{\frac{\log\frac{2}{\delta}}{2n}}.$$

In addition, we have the following *data-dependent* bound with probability at least $1 - \delta$:

$$\sup_{f\in\mathcal{F}} \big|R(f) - R_n(f)\big| \leq \hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F}) + 3\sqrt{\frac{\log\frac{4}{\delta}}{2n}}.$$

– Proof outline. The idea is to (i) bound the average of the left-hand side in terms of the Rademacher complexity, and (ii) bound the deviation from the mean via Hoeffding's inequality.

– Regarding step (i), here we briefly outline how to characterize $\mathbb{E}_{\mathcal{D}}\big[\sup_{f\in\mathcal{F}}(R(f)-R_n(f))\big]$; the idea for $\mathbb{E}_{\mathcal{D}}\big[\sup_{f\in\mathcal{F}}|R(f)-R_n(f)|\big]$ is similar. Recalling that $R_n(f)=\frac{1}{n}\sum_{i=1}^n \ell_f(\mathbf{x}_i,y_i)$ one can do a few basic tricks to show

$$\mathbb{E}_{\mathcal{D}}\bigg[\sup_{f\in\mathcal{F}}\big(R(f)-R_n(f)\big)\bigg]\le \mathbb{E}_{\mathcal{D},\mathcal{D}'}\bigg[\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{i=1}^n\big(\ell_f(\mathbf{x}_i,y_i)-\ell_f(\mathbf{x}_i',y_i')\big)\bigg]$$

where $\mathcal{D}'=\{(\mathbf{x}_i',y_i')\}_{i=1}^n$ is a second *independent* data set. But then $\ell_f(\mathbf{x}_i,y_i)-\ell_f(\mathbf{x}_i',y_i')$ is symmetric about zero, so it has the exactly the same distribution as $\sigma_i\big(\ell_f(\mathbf{x}_i,y_i)-\ell_f(\mathbf{x}_i',y_i')\big)$ for an independent Rademacher variable σ_i.[7] Hence,

$$\mathbb{E}_{\mathcal{D}}\bigg[\sup_{f\in\mathcal{F}}\big(R(f)-R_n(f)\big)\bigg]\le \mathbb{E}_{\mathcal{D},\mathcal{D}',\boldsymbol{\sigma}}\bigg[\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{i=1}^n\sigma_i\big(\ell_f(\mathbf{x}_i,y_i)-\ell_f(\mathbf{x}_i',y_i')\big)\bigg].$$

Applying $\sup_f\big(A(f)+B(f)\big)\le\sup_f A(f)+\sup_f B(f)$ and doing some manipulations, we can bound the right-hand side by $2R_n(\mathcal{L})$, which equals $\mathcal{R}_n(\mathcal{F})$ due to the "Claim" stated above.

– For step (ii), note that Hoeffding's inequality gives the term $\sqrt{\frac{\log\frac{2}{\delta}}{2n}}$ just like in the case of finite function classes. This gives the first part of the theorem.

– To get the second part of the theorem, we replace δ by $\delta/2$ in the first part, and then use a more advanced concentration bound to show that $\mathcal{R}_n(\mathcal{F})$ is close to $\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F})$ with probability at least $1-\delta/2$ (then summing two $\delta/2$-probability events gives an overall probability of δ). This is done using *McDiarmid's inequality* (final section of the previous lecture) – viewed as a function of $\boldsymbol{\sigma}$, the quantity $\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F})$ changes by at most $\frac{1}{n}$ when one of the σ_i is changed.

Deducing VC Dimension Based PAC Learnability.

- A useful preliminary step is to bound the Rademacher complexity in the special case of a finite function class $\mathcal{F}$. For this, a result known as *Massart's lemma* states that if $f_1,\dots,f_n$ is a sequence taking values in $\{-1,1\}$, then

$$\mathbb{E}_{\boldsymbol{\sigma}}\bigg[\sup_{f\in\mathcal{F}}\frac{1}{n}\sum_{i=1}^n\sigma_i f_i\bigg]\le\sqrt{\frac{2\log|\mathcal{F}|}{n}}.$$

  The proof is omitted here, but the key steps are similar to our analysis of PAC learnability for finite $\mathcal{F}$. Instead of using Hoeffding's inequality directly (which is a high-probability result, whereas here we are looking at an average), we use some of the ingredients of its proof.

- Now the key idea, as we mentioned previously, is that even for infinite $\mathcal{F}$, the *effective number* of functions is finite. In particular, if we let $|\mathcal{F}|_{\mathcal{D}}$ be the number of different vectors $(f(\mathbf{x}_1),\dots,f(\mathbf{x}_n))$ that can be produced by some $f\in\mathcal{F}$ (for fixed $(\mathbf{x}_1,\dots,\mathbf{x}_n)$), then combining the previous display

[7] This technique is commonly referred to as *symmetrization*.

equation with the definition of empirical Rademacher complexity gives

$$\hat{\mathcal{R}}_{\mathcal{D}}(\mathcal{F}) \leq \sqrt{\frac{2\log|\mathcal{F}|_{\mathcal{D}}}{n}}.$$

- Now, recalling the definition of the growth function (shattering number) $S_n(\mathcal{F})$ from Section 4, we always have $|\mathcal{F}|_{\mathcal{D}} \leq S_n(\mathcal{F})$. Hence, and averaging both sides above over $\mathcal{D}$, we have

$$\mathcal{R}_n(\mathcal{F}) \leq \sqrt{\frac{2\log S_n(f)}{n}}.$$

- Substituting into the key theorem on Rademacher complexity, and applying the basic inequality $\sqrt{a}+\sqrt{b} \leq 2\sqrt{a+b}$, we can get to

$$\sup_{f\in\mathcal{F}} \left|R(f) - R_n(f)\right| \leq \sqrt{\frac{8(\log S_n(f) + \log\frac{2}{\delta})}{n}}$$

  with probability at least $1-\delta$.

- Finally using Sauer's lemma (Section 4) to get $\log S_n(f) \leq d_{\text{VC}} \log(n+1)$, we obtain a good enough bound to establish uniform convergence (and hence PAC learnability) whenever d_{VC} is finite.
  - Recall that we already established that $\sup_{f\in\mathcal{F}} \left|R(f)-R_n(f)\right| \leq \epsilon_0$ implies that empirical risk minimization yields true risk $R(f)$ within $2\epsilon_0$ of the best $f \in \mathcal{F}$.
  - The sample complexity from the previous equation (i.e., setting the right-hand side to $\frac{\epsilon}{2}$ and solving for n) leads to the replacement d_{VC} by a slightly higher term $d_{\text{VC}} \log n$ in (6); avoiding this log factor requires a refined analysis.

- **Note:** The theorem on Rademacher complexity is not only useful for obtaining the VC dimension based bound – it is also useful in its own right. In particular, for several function classes of interest in can be bounded directly, so we don't need to resort to the (looser) notion of VC dimension.
  - The notion of a *data dependent* bound (second part of the theorem above) is also appealing, because this can allow us to understand the generalization behavior based on the specific data we have observed. In contrast, the notion of VC dimension concerns the worst-case behavior over all possible data distributions $P_{\mathbf{X}Y}$.