Measuring “richness” in $\mathcal{H}$

Given a hypothesis set $\mathcal{H}$ with $|\mathcal{H}| = m$, we have previously shown that if

$$h^* = \underset{h \in \mathcal{H}}{\arg\min}\ \widehat{R}_n(h)$$

then for any $\epsilon > 0$

$$\mathbb{P}\left[\left|\widehat{R}_n(h^*) - R(h^*)\right| \geq \epsilon\right] \leq 2m \exp(-2n\epsilon^2).$$

The factor m in this bound is the result of the *union bound*, which we used to show that for $\epsilon > 0$

$$\begin{aligned} \mathbb{P}\left[\left|\widehat{R}_n(h^*) - R(h^*)\right| \geq \epsilon\right] &\leq \mathbb{P}\left[\max_{h \in \mathcal{H}} \left|\widehat{R}_n(h) - R(h)\right| \geq \epsilon\right] \\ &\leq \sum_{j=1}^{m} \mathbb{P}\left[\left|\widehat{R}_n(h_j) - R(h_j)\right| \geq \epsilon\right]. \end{aligned}$$

The second inequality can sometimes hold with equality, but this only occurs when the events $\mathcal{E}_j = \{|\widehat{R}_n(h_j) - R(h_j)| \geq \epsilon\}$ are *disjoint.* In most practical choices for $\mathcal{H}$, this is rarely the case. This is illustrated in Fig. 1 below, where the two classifiers shown are distinct but have exactly the same empirical risk on the training set. Since the two classifiers are very similar, if one has a large deviation between its empirical and true risk, we would also expect the other to as well.

This observations suggests that our bound might be extremely loose and that $|\mathcal{H}|$ may not necessarily be the right measure of the *richness* of the hypothesis set $\mathcal{H}$. Most of our work in the next two lectures will be devoted to finding a suitable replacement for $|\mathcal{H}|$, which will enable use to prove a generalization bound even in settings for which $|\mathcal{H}| = \infty$, as is the case for linear classifiers.

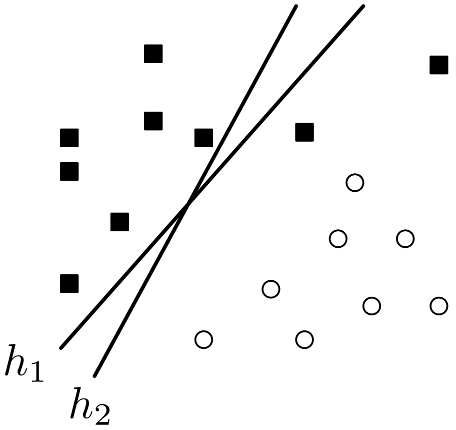


Figure 1: Two distinct classsifiers with the same empirical risk

Dichotomies and the growth function

Motivated by the situation in Fig. 1, where multiple (in fact, infinitely many) different $h \in \mathcal{H}$ have the *same* empirical risk, we will attempt to assess the number of hypotheses that lead to *distinct* labelings for a given dataset. Intuitively, we are hoping that the number of distinct labelings is a quantity that better captures the richness of the hypothesis class $\mathcal{H}$. For simplicity, we will restrict our attention for now to the case of binary classification where $\mathcal{Y} = \{+1, -1\}$.

Formally, for a dataset $\mathcal{D} = \{\boldsymbol{x}_i\}_{i=1}^n$, a **dichotomy** is a particular labelling of the dataset $\mathcal{D}$, i.e., a particular sequence of n labels of ± 1. Given a set of hypotheses $\mathcal{H}$, the set of all possible dichotomies generated by $\mathcal{H}$ on $\mathcal{D}$ is the set of labelings that can be generated by classfiers in $\mathcal{H}$ on the dataset, which we denote by

$$\mathcal{H}(\{\boldsymbol{x}_i\}_{i=1}^n) = \{\{h(\boldsymbol{x}_i)\}_{i=1}^n : h \in \mathcal{H}\}.$$

Note that many sets $\{\{h(\boldsymbol{x}_i)\}_{i=1}^n$ for distinct h are actually identical because the labelings induced on the dataset are identical. By definition, for our binary labeling problem, $|\mathcal{H}(\{\boldsymbol{x}_i\}_{i=1}^n)| \leq 2^n$ and in general $|\mathcal{H}(\{\boldsymbol{x}_i\}_{i=1}^n)| \ll |\mathcal{H}|$. Unfortunately, $|\mathcal{H}(\{\boldsymbol{x}_i\}_{i=1}^n)|$ is not a particularly useful quantity because it is not only potentially difficult to compute but also dependent on a specific dataset. This motivates the definition of the *growth function* as follows.

For a set of hypotheses $\mathcal{H}$, the **growth function** of $\mathcal{H}$ is

$$m_{\mathcal{H}}(n) = \max_{\{\boldsymbol{x}_i\}_{i=1}^n} |\mathcal{H}(\{\boldsymbol{x}_i\}_{i=1}^n)|.$$

Note that the growth function depends on the number of datapoints n but not on the exact datapoints $\{\boldsymbol{x}_i\}_{i=1}^n$. The growth function measures the maximum number of dichotomies that $\mathcal{H}$ can generate over *all* possible datasets. By definition, we still have that $m_{\mathcal{H}}(n) \leq 2^n$, but we will see that in many cases $m_{\mathcal{H}}(n) \ll 2^n$, that is, no matter how you choose the $\boldsymbol{x}_i$, you may not be able to achieve all possible labelings of the data using classifiers in $\mathcal{H}$.

Example: Positive rays

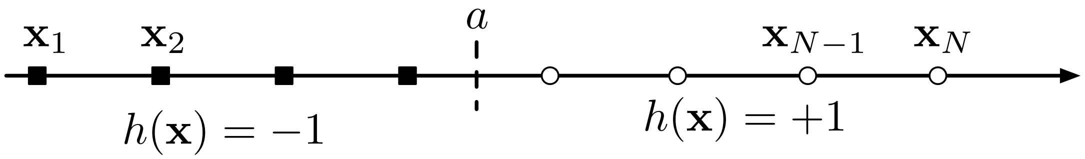
Consider a binary classification problem in $\mathbb{R}$ with the set of positive rays

$$\mathcal{H} = \{h_a : h_a(x) = \operatorname{sign}(x-a), a \in \mathbb{R}\}.$$

Recall that $\operatorname{sign}(x)$ is the function

$$\operatorname{sign}(x) = \begin{cases} 1 & \text{if } x > 0, \\ 0 & \text{if } x = 0, \\ -1 & \text{if } x < 0. \end{cases}$$

As illustrated below, the threshold a defines a classifier such that all points to the left are assigned label -1 while all points to the right are assigned label $+1$.



Although $|\mathcal{H}| = \infty$, the number of dichotomies is still finite, and one can actually compute the growth function exactly. In general, this is challenging because we need to identify the *worst case* dataset that generates the highest number of dichotomies; here, this is only tractable because the situation is simple.

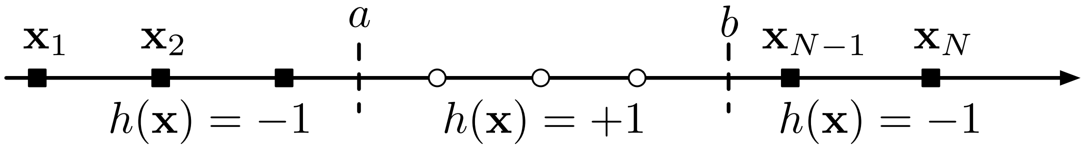
Without losing generality, we can assume that all n points $\{x_i\}_{i=1}^n$ are distinct. Let us introduce $x_0 = -\infty$ and $x_{n+1} = \infty$. For any $i \geq 0$, all classifiers h_a with $x_i \leq a < x_{i+1}$ induce the same labeling. Consequently, the number of distinct labelings is at most $n + 1$ and $m_{\mathcal{H}}(n) = n + 1$. Interestingly, the growth function is growing polynomially in n, which is much slower than the exponential growth 2^n allowed by the upper bound.

Example: Positive intervals

Consider a binary classification in $\mathbb{R}$ with the set of positive intervals

$$\mathcal{H} = \{h_{a,b} : h_{a,b} = \text{sign}(x - a) - \text{sign}(x - b), a < b \in \mathbb{R}\}.$$

As illustrated below, the thresholds $a < b$ define a classifier such that all points with $[a; b]$ are assigned label $+1$ while all points outside are assigned label -1.



Again, this is a situation for which we can compute the growth function exactly. Without loss of generality, we assume that all n datapoints are distinct and we introduce $x_0 = -\infty$ and $x_{n+1} = \infty$. We need to be a bit more careful when counting dichotomies:

- If $x_0 < a < b \leq x_1$, all classifiers $h_{a,b}$ induce a labelling of all -1's;
- for any $0 \leq i < j \leq n$, all classifiers $h_{a,b}$ such that $x_i \leq a \leq x_{i+1} < x_j \leq b \leq x_{j+1}$ induce the same labelings;
- for any $0 \leq i \leq n$, all classifiers $h_{a,b}$ such that $x_i \leq a < b < x_{i+1}$ induce again a labelling of all -1's.

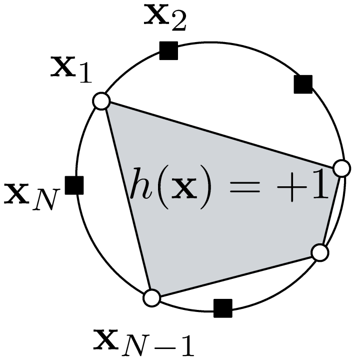
Consequently, the number of classifiers is $1 + \binom{n+1}{2}$ and $m_{\mathcal{H}}(n) = \frac{n^2}{2} + \frac{n}{2} + 1$, which grows again polynomially in n.

Example: Convex sets

Consider a binary classification in $\mathbb{R}^2$ with the set

$$\mathcal{H} = \{h : \{\boldsymbol{x} \in \mathbb{R}^2 : h(\boldsymbol{x}) = +1\} \text{ is a convex set}\}. \tag{1}$$

Consider a set of n distinct points distributed on the unit circle, as illustrated below.



Notice that irrespective of the desired labeling of the datapoints, the datapoints for which $h(\boldsymbol{x}_i) = +1$ define the vertices of a polytope, which is convex. Said differently, irrespective of the labeling, there exists $h \in \mathcal{H}$ that generates the labeling. Therefore, by definition, $m_{\mathcal{H}}(n) = 2^n$.

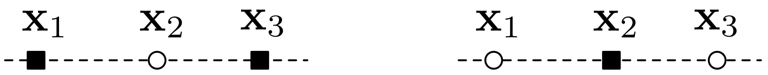
The three previous examples are not at all representative of a general situation because it is nearly impossible to compute the growth function exactly in most practical cases. As shown next, even for linear classifiers this can become a formidable task.

Example: Linear classifiers
Consider a binary classification in $\mathbb{R}^2$ with the set of linear classifiers

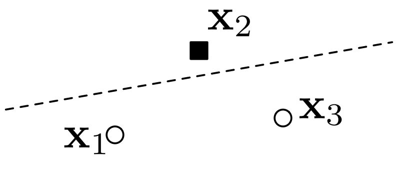
$$\mathcal{H} = \{h : h(\boldsymbol{x}) = \operatorname{sign}(\boldsymbol{w}^\intercal \boldsymbol{x} + b), \boldsymbol{w} \in \mathbb{R}^2, b \in \mathbb{R}\} \tag{2}$$

The challenge again is to identify the worst case dataset that generates the most dichotomies. We first note that $\{\boldsymbol{x} : \boldsymbol{w}^\intercal \boldsymbol{x} + b = 0\} = \{\boldsymbol{x} : -\boldsymbol{w}^\intercal \boldsymbol{x} + b = 0\}$, so that a single line actually defines two classifiers (that differ only in terms of how they label each side).

For $n = 3$, we need to distinguish two cases. If all three points are aligned, all dichotomies except those illustrated below are possible, we therefore obtain six dichotomies.

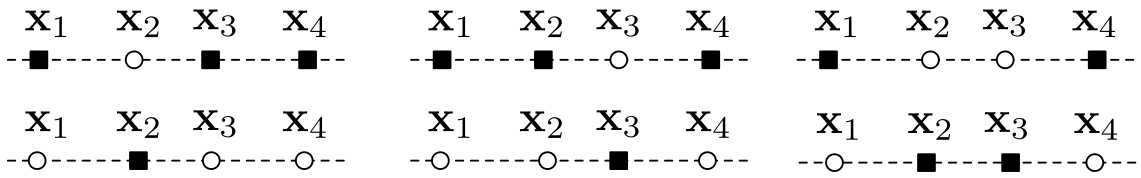


However, we are interested in the *maximum* possible number of dichotomies. If the three points are not aligned, they form the vertices of a polytope and any hyperplane cutting the polytope will isolate one point. In addition, any hyperplane no cutting the polytope will assign the same label to all three points. Consequently, the number of dichotomies generated is $8 = 2^3$.

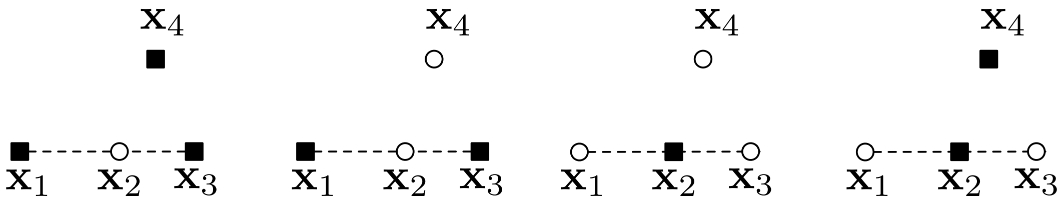


Consequently, $m_{\mathcal{H}}(3) = 8$.

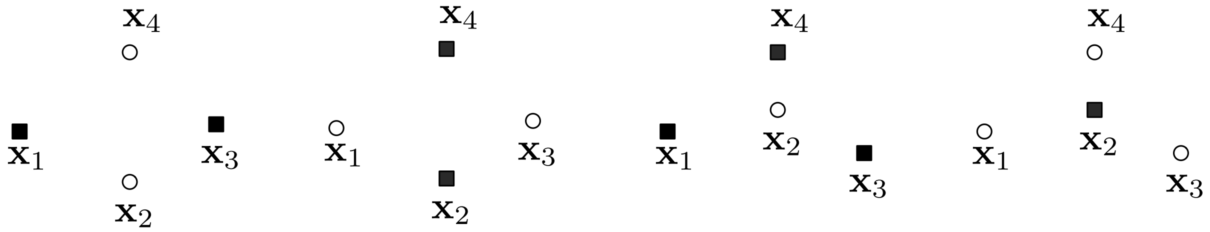
For $n = 4$, we need to distinguish even more cases. If all four points are aligned, all dichotomies except those illustrated below are possible we therefore obtain 10 dichotomies.



If three out of four points are aligned, the four points form a 3-vertex polytope, and one point, say $\boldsymbol{x}_2$, is on the edge, say defined by $\boldsymbol{x}_1$ and $\boldsymbol{x}_3$. Any hyperplane cutting through the polytope cannot assign a label to $\boldsymbol{x}_2$ that is distinct of both $\boldsymbol{x}_1$ and $\boldsymbol{x}_3$. Consequently, the dichotomies illustrated below cannot be generated and we obtain 12 dichotomies.



If no three out of four points are aligned, the four points could form a 4-vertex polytope, in which case a hyperplane cutting through the polytope cannot assign distinct labels to a vertex and all its neighbors. The four points could also form a 3-vertex polytope with a point in the interior, in which case a hyperplane cutting through the polytope cannot assign a label to the interior point distinct from all the vertices. Consequently, the dichotomies illustrated below cannot be generated and we obtain 14 dichotomies.



Thus, $m_{\mathcal{H}}(4) = 14$.

This last example illustrates the essentially combinatorial nature of the calculation of the growth function. As we will soon seen, we will conveniently only care about the *scaling* of the growth function with n – in particular, whether it is polynomial or exponential.

Shattering and break point

Above we introduced the notion of growth function, $m_{\mathcal{H}}(\mathrm{n})$, which characterizes the maximum number of labelings that can be obtained with a given hypothesis set $\mathcal{H}$ over all datasets $\boldsymbol{x}\rangle_{i=1}^{n}$ of size n. The behavior of the growth function as a function of n can be different depending on the structure of the hypotheses in $\mathcal{H}$, and we saw examples in which $m_{\mathcal{H}}(n)$ grows polynomially or exponentially in n.

The problem of computing $m_{\mathcal{H}}(n)$ is often intractable, quickly becoming an intricate computational problem that depends not only on all possible configurations of points in the dataset but also on the constraints induced by the structure of hypotheses in $\mathcal{H}$. We will focus instead on determining the behavior of $m_{\mathcal{H}}(n)$ as a function of n, which will conveniently tell us a lot about generalization in a next lecture.

We start by introducing the notion of shattering and break points. If a hypothesis set $\mathcal{H}$ can generate all dichotomies on $\{\boldsymbol{x}_i\}_{i=1}^{n}$, we say that $\mathcal{H}$ **shatters** $\{\boldsymbol{x}_i\}_{i=1}^{n}$. If no data set of size k can be shattered by $\mathcal{H}$, then k is a **break point** for $\mathcal{H}$. Note that if k is a break point, any $\ell > k$ is also a break point.

Example

For a binary linear classifier in $\mathbb{R}^2$, we saw that $m_{\mathcal{H}}(4) = 14 < 16$. In other words, no dataset of size 4 can be shattered by linear classifiers and $k = 4$ is a break point.

Although we gave up computing $m_{\mathcal{H}}(n)$ for linear classifiers in $\mathbb{R}^2$ for $N > 4$, it turns out that the existence of break point k is already enough to for us to bound $m_{\mathcal{H}}(n)$ for every n. We will formalize this shortly, but we first illustrate this point with an example.

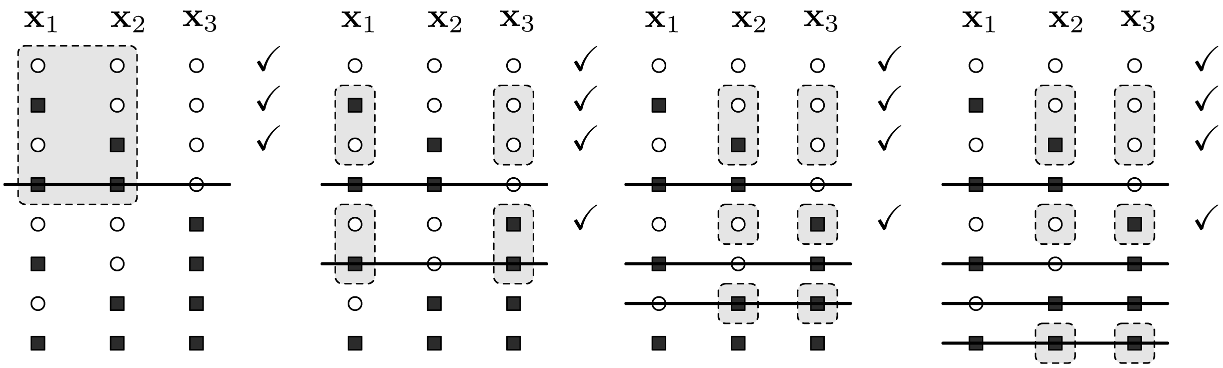
Example

Consider a binary classification problem and assume that $k = 2$ is a break points for $\mathcal{H}$. How many dichotomies can we generate of set of size $N = 3$? Our assumption says that $\mathcal{H}$ cannot shatter a set of size 2, so that no $h \in \mathcal{H}$ can assign all four possible distinct labelings to any set of two points.

Consider the table below, which illustrates all possible binary (○, ■) labelings on a set size 3.

$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$
○	○	○
■	○	○
○	■	○
■	■	○
○	○	■
■	○	■
○	■	■
■	■	■

As illustrated below, we proceed to eliminate labelings forbidden by our assumption that $k = 2$ is breakpoint starting from the top. You can check for yourself that any other order of labeling would result in us eliminating the same number of dichotomies.



The first three rows correspond to labelings that do not violate our assumption. The fourth row has to be excluded because it would otherwise allow us to shatter a set of size 2, as illustrated by the gray region. The procedure continues and one can see that only 4 labelings are allowed out of the 8 possible.

The previous example shows that knowing a break point allows us to reason about the growth function without really knowing much about $\mathcal{H}$.

Bounding the growth function and Sauer's lemma

We now formalize the intuition developed above. Assume $\mathcal{H}$ has break point k. Define $B(n, k)$ as the maximum number of dichotomies of n points such that *no subset* of size k can be shattered by the dichotomies.

Note that $B(n, k)$ is a purely combinatorial quantity, which depends on the fact that k is a break point for $\mathcal{H}$ but otherwise not on the specific nature of $\mathcal{H}$. By definition, if k is a break point for $\mathcal{H}$, then $m_{\mathcal{H}}(n) \leq B(n,k)$.

What makes the definition of $B(n,k)$ useful is that we can bound it much more easily than $m_{\mathcal{H}}(n)$.

Lemma 1 (Sauer's lemma)

$$B(n,k) \leq \sum_{i=0}^{k-1} \binom{n}{i}$$

Proof See Section 2.1.2 of *Learning from Data.* ■

If a sum of binomial coefficients isn't already nice enough for you, note that

$$\sum_{i=0}^{d} \binom{n}{i} \leq n^d + 1,$$

and hence $B(n,k) \leq n^{k-1} + 1$. There are many ways to prove this, but there is an elegant one that uses induction. Specifically, we begin with the case base of $d = 0$. In this case we have

$$\sum_{i=0}^{0} \binom{n}{i} = \binom{n}{0} = 1 \leq n^0 + 1.$$

Thus, we now assume that the inequality holds up to $d-1$, in which case we have

$$\sum_{i=0}^{d} \binom{n}{i} = \left(\sum_{i=0}^{d-1} \binom{n}{i} \right) + \binom{n}{d} \leq n^{d-1} + 1 + \binom{n}{d}.$$

Note that

$$\begin{aligned}\binom{n}{d} &= \frac{n!}{d!(n-d)!} \\ &= \frac{n(n-1)(n-2)\cdots(n-d+1)}{d!} \\ &\le n(n-1)(n-2)\cdots(n-d+1) \\ &\le n^{d-1}(n-1) \\ &= n^d - n^{d-1}.\end{aligned}$$

Plugging this in to the bound above, we obtain

$$\sum_{i=0}^{d}\binom{n}{i} \le n^{d-1} + 1 + \binom{n}{d} \le n^{d-1} + 1 + n^d - n^{d-1} = n^d + 1,$$

as desired.