1 Recap from last week

Last time, we discussed what learning means. More formally, we considered a toy problem where we tried to estimate the sweetness or sourness of an apple based on two parameters, weight and height. There are many other interesting sources of real life data where we want to divide the data into distinct classes. For instance, we can measure the electrical activity of the human brain when a person looks at a face from different angles. The resulting data from such an experiment is displayed in Figure 1. Strikingly, we notice 6 distinct clusters, one from each experiment, which are very linearly separable. The fact that a simple linear classifier can determine which angle a person was looking at a face suggests that even simple models can be useful in real world applications.

Figure 1: MRI Data

1.1 Statistical Learning Theory: A Review

Recall that last week we defined a model for learning, called a statistical learning model.

Definition 1.1. A learning problem instance in the Statistical Learning Model is a 4-tuple \( \{X, Y, S, D\} \) where \( X \) is the domain (the parametrizations of the objects we are trying to classify), \( Y \) is the label set (the possible outputs), \( S \) is the data set \( \{(x_i, y_i) \in X \times Y\} \) with \( |S| = n \), and \( D \) is an unknown and arbitrary distribution on \( X \).
Example 1.2. To clarify the meaning of the previous definition, we ground it in a real-world example. Consider the apple problem from last week: here, \( X \) is the set of apples, parametrized by weight and diameter: \( x_i = (w_i, d_i) \). \( Y \) is the label set and here is simply \( \{0, 1\} \), where 1 may denote sweet and 0 may denote sour. We might be provided with \( n = 1000 \) samples \( S = \{(x_1, y_1), \cdots, (x_{1000}, y_{1000})\} \). Finally, we do not know the distribution \( D \) that these apples are drawn from, but we assume it exists for the whole population.

Definition 1.3. A learning algorithm outputs a hypothesis \( h : X \rightarrow Y \), mapping (unseen) examples/objects to labels.

Definition 1.4. We call \( H \) the hypothesis class: It is the set of hypothesis functions that we consider for a given learning problem.

Finally, we need a form of measurement to evaluate how well a specific hypothesis does.

Definition 1.5. The generalization error \( \operatorname{err}_D(h) = \Pr_{(x,y) \sim D}\{h(x) \neq y\} \) is the error the classifier \( h \) makes on all possible inputs according to the unknown distribution \( D \). Typically, \( \operatorname{err}_D(h) \) should be chosen to have a low error on the sample set.

To that effect, we define \( h_{\text{ERM}} \): i.e.,

Definition 1.6. The empirical risk minimization algorithm \( h_{\text{ERM}} : X \rightarrow Y \) is the classifier that achieves the minimum error on the training set \( S \). Formally,

\[
h_{\text{ERM}} = \arg \min_{h \in H} \operatorname{err}_S(h); \quad \text{where} \quad \operatorname{err}_S(h) = \frac{|\{(x_i, y_i) | h(x_i) \neq y_i\}|}{|S|} \tag{1}
\]

We also sometimes assume a condition on the hypothesis set \( H \).

Definition 1.7. The realizability assumption tells us that for a given hypothesis class \( H \), there exists a hypothesis \( h^* \in H \) with zero error \( \operatorname{err}_D(h^*) = 0 \).

The main result from the first lecture was a theorem that gave us some nice bounds on the error rate of \( h_{\text{ERM}} \) in the case of a finite hypothesis class \( H \) and assuming realizability.

Theorem 1.8. Under the realizability assumption, we have that \( \operatorname{err}_D(h_{\text{ERM}}) < \epsilon \) with probability \( 1 - \delta \), if \( |S| \geq \frac{1}{2} \ln \left( \frac{|H|}{\delta} \right) \).

Remark 1.9. A very important point: Theorem 1.8 holds for any distribution \( D \), even if it is degenerate.

We give a brief example of the way Theorem 1.8 works:

Example 1.10. Learning the class of conjunctions. Consider the problem of email spam-classification. Let \( H \) be the set of conjunctions over \( n \) boolean variables; i.e. \( X = \{x_1, \cdots, x_n\} \in \{0, 1\}^n \) and \( Y = \{0, 1\} \) where 0 denotes spam and 1 denotes not spam. To ground our example in the real world, each boolean variable might indicate the presence of an associated word in an email. For instance, in some simple model, \( x_1 = 1_{\text{Viagra}} \) where \( 1_p = 1 \) if the predicate \( p \) is true, and 0 if \( p \) is false. Here, \( p \) is true if the word Viagra is in the email. In this case, it seems likely that the email should be classified as spam if \( x_1 = 1 \). An example hypothesis is given by \( h = f(x^1, \cdots, x^n) = x^1 \land x^3 \land \bar{x}^5 \land x^7 \). Here \( \bar{x} \) denotes the negation of the variable \( x \).

Are conjunctions learnable in the statistical learning model, and if so, how many examples do we need to learn them? Suppose we have \( S = \{(x_i, y_i) | i \in [k]\} \). Let us use the ERM algorithm to find a conjunction...
which fits the data in the best way. Let us also assume realizability: That there in fact does exist a perfect conjunction which splits the data perfectly. First notice that $|\mathcal{H}| = 3^n$, a boolean variable can either appear in a conjunction, appear negated, or not appear at all. Then we note that $\mathcal{H}$ is finite, and we can apply Theorem 1.8. For $|S| = \frac{1}{\epsilon} \ln \left( \frac{|\mathcal{H}|}{\delta} \right) = \frac{1}{\epsilon} \left( \ln \left( \frac{1}{\delta} \right) + n \ln(3) \right) \leq 2 \frac{\ln(3)n}{\epsilon} \ln \left( \frac{1}{\delta} \right) \leq \frac{2}{\epsilon} \ln \left( \frac{1}{\delta} \right)$, we have $\text{err}_D(h_{\text{ERM}}) \leq \epsilon$ with probability $1 - \delta$. Thus the required size of $S$ grows linearly with the number of variables. This application of the theorem was easy – but say we want to learn linear half-spaces instead, an non-finite $\mathcal{H}$. It is clear this theorem still has weaknesses.

We finished the last lecture by listing a few weaknesses of this theorem; namely, the realizability condition and our assumption that $\mathcal{H}$ is finite. In this lecture, we will start removing these weaknesses.

2 Agnostic Learning

We would like to remove the realizability assumption.

2.1 Noisy Data

Recall that last week, we considered the hypothesis class of axis-aligned rectangles with finite precision, and sought to find a hypothesis that perfectly classified apples as sweet or sour (see Figure 2).

![Figure 2: Apples: Sweet or Sour?](image)

What if there is some noise, and it is not possible to perfectly classify the training set with a rectangle? We must be wary of overclassifying the data (Figure 3). The problem with finding a hypothesis that unconstrainedly correctly classifies all of the sample data is that such hypotheses tend to generalize poorly.

2.2 New Definitions

**Definition 2.1. Probably-Approximately-Correct (PAC) Learning.**

A hypothesis class is said to be statistically learnable if for sample complexity $m_{\mathcal{H}}(\epsilon, \delta) \forall \epsilon, \delta > 0$, we have an algorithm that produces $h$ such that $\text{err}_D(h) < \epsilon$ with probability $1 - \delta$ and $m_{\mathcal{H}}(\epsilon, \delta) = \text{poly} \left( \frac{1}{\epsilon}, \ln \left( \frac{1}{\delta} \right), \ln \left( |\mathcal{H}| \right) \right)$. Furthermore, the hypothesis class is PAC-learnable if the runtime of the algorithm is polynomial in $S$ (the runtime depends on the representation of $S$).

The results from last week showed that every finite $\mathcal{H}$ is statistically-learnable. Now, instead of looking at the distance from a hypothesis with absolute error of 0 on the training set, we will look at the error compared to the best hypothesis in the class, which is a relative error. We introduce a new definition of learning:
Figure 3: The dangers of overclassification

Definition 2.2. A hypothesis class $\mathcal{H}$ is agnostically learnable if $\forall \epsilon, \delta > 0$, there exists $m_{\mathcal{H}}(\epsilon, \delta) = \text{poly} \left( \frac{1}{\epsilon}, \ln \left( \frac{1}{\delta} \right), \ln \left( |\mathcal{H}| \right) \right)$ and an algorithm that given $|S| = m_{\mathcal{H}}(\epsilon, \delta)$ returns a hypothesis $h$ such that $\text{err}_D(h) \leq \min_{h^* \in \mathcal{H}} (\text{err}_D(h^*)) + \epsilon$.

2.3 Is every hypothesis class agnostically learnable?

For statistically learnable, we say that the ERM algorithm was good enough. It turns out that the ERM algorithm will also be good enough in the agnostically learnable case, but we will get a slightly different condition on the size of the training set required. A brief reminder: We are trying to get the smallest possible error that exists rather than assuming realizability.

Before stating and proving our theorem, we introduce a lemma needed to prove the theorem. We first state a useful inequality:

Lemma 2.3. Hoeffding Inequality.
Let $x_1, \cdots, x_n$ be i.i.d. random variables with $|x_i| \leq M$ and $\mu = E[x_i]$. Then,

$$
\Pr\left\{ \frac{1}{n} \sum_{i=1}^{n} x_i - \mu > t \right\} \leq e^{-\frac{2nt^2}{M^2}} 
$$

$$
\Pr\left\{ \left| \frac{1}{n} \sum_{i=1}^{n} x_i - \mu \right| > t \right\} \leq 2e^{-\frac{2nt^2}{M^2}}
$$

Lemma 2.4. For all $h \in \mathcal{H}$, we have

$$
\Pr\{ |\text{err}_S(h) - \text{err}_D(h)| > \epsilon \} \leq 2e^{-2\epsilon^2|S|}
$$

Recall $\text{err}_S(h)$ is given in Definition 1.6.

Proof. The proof follows directly from the Hoeffding Inequality. We let $x_i = \mathbf{1}_{h(x_i) \neq y_i}$ for $(x_i, y_i) \sim D$, and therefore $\text{err}_S(h) = \frac{1}{|S|} \sum_{(x_i, y_i) \in S} x_i$. Also, $E[x_i] = \text{err}_D(h)$. Then since $x_i$ is an indicator function, $M = 1$ and the result follows from Lemma 2.3.

4
Theorem 2.5. Every finite $\mathcal{H}$ is agnostically learnable with $m_{\mathcal{H}}(\epsilon, \delta) \in \Theta\left(\frac{1}{\epsilon^2} \ln\left(\frac{|\mathcal{H}|}{\delta}\right)\right)$.

Proof. Let $h^*$ be the optimal hypothesis from $\mathcal{H}$ over $\mathcal{D}$. We would like to prove that $\text{err}_D(h_{\text{ERM}}) \leq \text{err}_D(h^*) + \epsilon$ with probability $1 - \delta$ for sample size $m_{\mathcal{H}}(\epsilon, \delta)$ as specified in theorem. Let us use the ERM algorithm (see Definition 1.6). By definition,

$$\text{err}_S(h_{\text{ERM}}) \leq \text{err}_S(h^*)$$

Suppose that in addition, our claim does not hold, i.e.

$$\text{err}_D(h_{\text{ERM}}) > \text{err}_D(h^*) + \epsilon$$

The latter two inequalities imply that

$$\text{err}_D(h_{\text{ERM}}) - \text{err}_S(h_{\text{ERM}}) \geq \text{err}_D(h^*) - \text{err}_S(h^*) + \epsilon$$

Let $\Delta_{\text{err}}(h) = \text{err}_D(h) - \text{err}_S(h)$ be the difference between generalization and sample error for a particular hypothesis. Rewriting the above, we have

$$|\Delta_{\text{err}}(h_{\text{ERM}})| + |\Delta_{\text{err}}(h_{\text{ERM}})| \geq \Delta_{\text{err}}(h_{\text{ERM}}) - \Delta_{\text{err}}(h^*) \geq \epsilon$$

This means that $|\Delta_{\text{err}}(h_{\text{ERM}})| \geq \frac{\epsilon}{2}$ or $|\Delta_{\text{err}}(h^*)| \geq \frac{\epsilon}{2}$, and in particular that there exists some hypothesis $h$ for which $|\Delta_{\text{err}}(h)| \geq \frac{\epsilon}{2}$. Define

$$\mathcal{H}_{\text{misleading}} = \{h | \Delta_{\text{err}}(h) > \frac{\epsilon}{2}\}$$

Then,

$$\Pr[\text{err}_D(h_{\text{ERM}}) > \text{err}_D(h^*) + \epsilon] \leq \Pr[\mathcal{H}_{\text{misleading}} \neq \emptyset]$$

$$= \Pr[\bigcup_{h \in \mathcal{H}} h \in \mathcal{H}_{\text{misleading}}]$$

$$\leq \sum_{h \in \mathcal{H}} \Pr[h \in \mathcal{H}_{\text{misleading}}]$$

$$\leq |\mathcal{H}| \cdot 2e^{-\epsilon^2|S|/2} \quad \text{Lemma 2.4}$$

$$\leq \delta \quad \text{choice of } |S|$$

Remark 2.6. Note that if we made the realizability assumption in our theorem, we could use the statistical learning theorem, which gives a better bound on the size of $m_{\mathcal{H}}(\epsilon, \delta)$.

Remark 2.7. What is reasonable to assume about the data? We only assume it comes from an arbitrary distribution. You can sample from this unknown distribution as long as you have the source of data.

Remark 2.8. In the homework, we will see that generalizing to agnostic learning implies some resilience for noise in the data.

3 VC Theory

Now we will remove the finite restriction on $\mathcal{H}$ and consider infinite-cardinality hypothesis classes.
3.1 Examples of infinite-cardinality hypothesis classes

Can we in fact learn any infinite $|\mathcal{H}|$? The answer is yes, so let us begin by considering a few examples of when learning is possible.

**Example 3.1.** A simple infinite-cardinality hypothesis class.

**Definition 3.2.** Let us define

$$ h_r(x) = \begin{cases} 
1 & : x \geq r \\
0 & : x < r 
\end{cases} $$

for $r \in \mathbb{R}$. Then let $\mathcal{H}_{\mathbb{R}+} = \{ h_r | r \in \mathbb{R} \}$ be the hypothesis class consisting of all positive half-lines. Note that $\mathcal{X} = \mathbb{R}$ and $\mathcal{Y} = \{0, 1\}$.

We will now prove that $\mathcal{H}_{\mathbb{R}+}$ is statistically PAC-learnable. $\mathcal{H}_{\mathbb{R}+}$ also satisfies the conditions for agnostic learning, but we defer that as an exercise to the reader. We choose to assume realizability for simplicity. Note that we cannot directly apply Theorem 1.8 since $|\mathcal{H}_{\mathbb{R}+}|$ is infinite.

**Proof.** Since we have assumed realizability, we know that a half-line can perfectly classify the data. Let $h_*$ be the perfect half-line classifier over $\mathcal{D}$. We have $h_*(x) = +$ if $x$ is to the right of $h_*$, and $h_*(x) = -$ if $x$ is to the left of $h_*$. Then, our algorithm to find $h_{ERM}$ is to pick maximal $r$ such that no $x < r$ is assigned to + in $S$ (see Figure 4). Thus, $r > h_*$.  

![Figure 4: Choosing $h_{alg}$ and $h_*$ for $\mathcal{H}_{\mathbb{R}+}$](image)

Let $h_{alg}$ be the hypothesis chosen by our algorithm. Note that $[h_*, h_{alg}]$ is the region where the perfect classifier and our algorithm-produced classifier disagree. The probability of landing in this region is therefore the generalization error. Then, let $\Pr_{x \sim \mathcal{D}}\{ x \in [h_*, h_{alg}] \} = \epsilon$. If $err_{D}(h_{alg}) > \epsilon$, then there is no $(x_i, y_i) \in [h_*, h_{alg}]$ (otherwise, we would have chosen $h_{alg}$ further to the left). Therefore,

$$ \Pr\{err_{D}(h_{alg}) > \epsilon\} \leq \Pr\{\forall(x_i, y_i) \in S, x_i \notin [h_*, h_{alg}]\} $$

$$ = \prod_{i=1}^{|S|} \Pr\{x_i \notin [h_*, h_{alg}]\} $$

$$ \leq (1 - \epsilon)^{|S|} \leq e^{-\epsilon |S|} \leq \delta $$

where we have used the fact that $1 + x \leq e^x$ from Taylor expansion. Thus, we choose $|S| \geq \frac{1}{\epsilon} \ln \left( \frac{1}{\delta} \right)$.  

\[\square\]
If we compare this result to Theorem 1.8, we note that \(|\mathcal{H}|\) kind of behaves the way 1 does in this derivation. We also note that there is only one continuous parameter involved, \(r\).

Let us do another example to see if this intuition holds up.

**Example 3.3.** Infinitely many rectangles.
Let us revisit the axis-aligned rectangle example from last week in the infinite-cardinality case. We will assume realizability again for simplicity.

**Definition 3.4.** Let \(\mathcal{H} = \{(a_1, a_2, b_1, b_2) \mid a_1, a_2, b_1, b_2 \in \mathbb{R}\}\) be the hypothesis class of infinite-precision axis-aligned rectangles. We have that \(X = \mathbb{R}^2, Y = \{0, 1\}\).

The idea here is that since there are 4 parameters, \(\mathcal{H}\) should behave as though \(|\mathcal{H}| = 4\), even though \(\mathcal{H}\) has infinite cardinality. We will show that \(\mathcal{H}\) is statistically learnable for \(m_{\mathcal{H}}(\epsilon, \delta) = \frac{4}{\epsilon} \ln \left(\frac{4}{\delta}\right)\).

*Proof.* We can apply the same rules from the previous example to the rectangle case: Let \(a_1^* \in \mathbb{R}\) satisfy \(x_i > a_1^*\), \(a_2^* \in \mathbb{R}\) satisfy \(x_i < a_2^*\), \(b_1^* \in \mathbb{R}\) satisfy \(y_i > b_1^*\), and finally \(b_2^* \in \mathbb{R}\) satisfy \(y_i < b_2^*\) for all \((x_i, y_i) \in \mathcal{S}\). Since we have realizability, we know that there is a perfect classifying rectangle \(h_*\) over \(\mathcal{D}\). Let the optimal classifying rectangle for the given data \(\mathcal{S}\) be \(h_{\text{alg}} = (a_1^*, a_2^*, b_1^*, b_2^*)\). We are interested in bounding the difference between them. We know that \(h_*\) must contain \(h_{\text{alg}}\) since otherwise, \(h_*\) would misclassify \((x_i, y_i) \in \mathcal{S}\). Then, let the probability mass of an \(x\) appearing between \(h_*\) and \(h_{\text{alg}}\) be \(\epsilon\). We can divide up the inner rectangular space into four rectangles \(R_1, R_2, R_3, R_4\), each with area > \(\frac{\epsilon}{4}\), which can be seen in Figure 5.

![Figure 5: \(h_{\text{alg}}\) and \(h_*\) for rectangles](image-url)

Then

\[
\Pr\{\text{err}_\mathcal{D}(h_{\text{alg}}) > \epsilon\} \leq \Pr\{\forall (x_i, y_i) \in \mathcal{S}, x_i \not\in R_1, R_2, R_3, R_4\} \\
\leq 4 \Pr\{\forall (x_i, y_i) \in \mathcal{S}, x_i \not\in R_1\} \\
= 4 \prod_{i=1}^{\lvert \mathcal{S} \rvert} \left(1 - \frac{\epsilon}{4}\right) = 4 \left(1 - \frac{\epsilon}{4}\right)^{\lvert \mathcal{S} \rvert} \\
\leq e^{-\frac{\epsilon \lvert \mathcal{S} \rvert}{4}} \leq \delta
\]

(6)
and thus $|S| \geq \frac{4}{\epsilon} \ln \left( \frac{4}{\delta} \right)$.

For the general case of $d$-dimensional rectangles, we have $m_{\mathcal{H}_V}(\epsilon, \delta) = \frac{2^d}{\epsilon} \ln \left( \frac{2^d}{\delta} \right)$, and we can show this result with the same method.

After this example, it is clear that the notion of a parameter describing a hypothesis class is something important. It was fairly clear how to parametrize in the two previous examples, but what about in other types of situations? Perhaps this approach will not always work. To that end, we give one more example.

**Example 3.5.** Disjunction over $n$ boolean variables.

We have $n$ boolean variables $x_1, \ldots, x_n \in \{0, 1\}^n$. For simplicity, let each variable be positive. Then, the domain $\mathcal{X}$ are bit strings of length $n$, where a $v$ in position $i$ means $x_i = v$, $v \in \{0, 1\}$. We let $\mathcal{Y} = \{0, 1\}$.

Interpreting this setup, we see that we are trying to learn a boolean function.

**Definition 3.6.** Let $\mathcal{H}_V = \{\text{Disjunctions over } n \text{ boolean variables}\}$. Each hypothesis is a boolean formula using the $x_i$ in disjunctive normal form (DNF). For instance, $(x_1 \land x_6) \lor (x_3 \land x_5)$ is a simple formula in DNF. There are $k = 2^n$ conjunctive formulas (either a variable is contained in the formula or it is not), and therefore a total of $2^{2n}$ DNFs (since we can choose to include or not include each of these $k$ conjunctive formulas).

First let us show that $h_{ERM}$ exists, and figure out how to describe it in this learning problem. For every $(x_i, y_i) \in S$, we can transform $x_i$ into a conjunctive formula $C_i$. Then let $C_i(x)$ denote applying $x$ to formula $C_i$. $C_i(x) = 1$ if the variables activated in $x$ match perfectly the variables activated in $C_i$, Therefore, only one of the $C_i$ can match a given $x$. Let us define

$$h_{ERM}(x) = \begin{cases} y_i &: C_i(x) = 1 \\ 0 &: C_i(x) = 0 \forall i \end{cases}$$

Note that this is identical to writing $h_{ERM} = C_1 \lor C_2 \lor \cdots \lor C_{|S|}$, for we have

$$h_{ERM}(x) = (C_1 \lor C_2 \lor \cdots \lor C_{|S|})(x) = C_1(x) \lor C_2(x) \lor \cdots \lor C_{|S|}(x) = 0 \lor 0 \lor \cdots \lor y \lor 0 \lor \cdots \lor 0 = y$$

since if $y = 0$, the total disjunction is 0, and if $y = 1$, the total disjunction is 1.

We therefore have the property that $h_{ERM}(x_i) = y_i$ for all $x_i$. So it seems reasonable that we can learn this problem.

To prove we can learn it, we note that $\mathcal{H}_V$ is realizable since any boolean formula can be converted to disjunctive normal form (that is, there exists a DNF $h_* \in \mathcal{H}_V$ that perfectly classifies all $(x, y) \sim \mathcal{D}$, since $(x, y) \sim \mathcal{D}$ is equivalent to a boolean function). Then we can apply Theorem 1.8 to get $m_{\mathcal{H}_V}(\epsilon, \delta) = \frac{1}{\epsilon} \ln \left( \frac{2^d}{\delta} \right) = \frac{1}{\epsilon} \left( 2^n \ln(2) + \ln \left( \frac{1}{\delta} \right) \right)$.

Thus we require at least $\frac{2^d}{\epsilon} > 2^n$ inputs to train it properly, but there are only $2^n$ possible inputs to the problem, the bit strings of length $n$. So this result is empty.

What went wrong? Effectively, the problem here is that the length of our hypothesis increased with the number of samples. We can better explain why we failed with the following theorem, which we will prove next week. It roughly tells us that if we do not restrict our hypothesis class, we cannot learn an optimal hypothesis.
Theorem 3.7. (No Free Lunch Theorem)
For all algorithms $A$ over $X$, $Y = \{0, 1\}$: Let $A_S$ denote the hypothesis outputted by $A$ with training data $S$, and let $m = |S|$. If $|X| > 2m$, there exists a distribution $D$ such that

1. $\exists f : X \to \{0, 1\}$ such that $\text{err}_D(f) = 0$
2. $\Pr\{\text{err}_D(A_S) > \frac{1}{8}\} > \frac{1}{8}$

We can apply this theorem to explain what happened in the previous example. We did not restrict our hypothesis class, and ended up being unable to learn.

3.2 VC Dimension

Is there a parameter that captures learnability? The answer is yes, and even better, it is usable for both finite and infinite hypothesis classes. This parameter is known as the VC dimension, named after Vapnik and Chervonenkis, who discovered the result.

First consider a learning problem $(X, Y, H, f, D)$. We denote the restriction of $H$ to $C \subseteq X$ by $H_C$. In other words, $H_C = \{(h(c_1), \ldots, h(c_{|S|})) | h \in H\}$, where we will assume for now that $|S|$ is finite. $H_C$ is the set of all possible labelings that a function in the hypothesis class could have produced. We may consider this a way of representing the predictive power of $H$. We have $|H_C| \leq 2^{|C|}$ since we are assuming $Y = \{0, 1\}$ (recall $h : X \to Y$).

Definition 3.8. Shattering.
We say $C \subseteq X$ is shattered by $H$ if $|H_C| = 2^{|C|}$.

As we said, this is a method we can use to capture the power of $H$. The larger $|H_C|$, the more powerful $H$ is.

Example 3.9. Positive half-line.
What is the largest subset $C$ of $H_{R^+}$ that can be shattered? We have that subsets $|C| = 1$ can be shattered trivially: a positive half-line can classify a single point $x$ as $+$ if we place $h_r$ to the left of $x$, and as $-$ if we place $h_r$ to the right of $x$.

However, a positive half-line cannot produce all four possible classifications of two points $x_1, x_2$. It can produce the classifications $(x_1, x_2) = (+, +); (-, -); (-, +)$, but fails to produce $(x_1, x_2) = (+, -)$, since the positive half-line defines the right side of the line as positive and the left side of the line as negative. Thus the maximally sized subset $C \subseteq X$ that is shattered by $H_{R^+}$ has $|C| = 1$.

Having gained some intuition, we are now ready to define the VC-dimension.

Definition 3.10. VC-dimension.
The VC-dimension of a hypothesis class $H$ is given by

$$\text{VC-dim}(H) = \max_{C \subseteq X} |C| \text{ s.t. } C \text{ is shattered by } H$$

(8)

Therefore, by the previous example, $\text{VC-dim}(H_{R^+}) = 1$. Now we try another example. Let us find the VC-dimension of $H_{\square}$.

Example 3.11. VC-dimension of $H_{\square}$.
We want to find the maximally-sized $C$ such that $H_{\square}$ can produce every possible labeling of the elements of $C$. If $|C| = 4$, we can produce every labeling of the points $(x_1, x_2, x_3, x_4)$, because we can choose whether or not the interior of an axis-aligned rectangle is positive or negative classification (see Figure 6).

However, it is not possible to form every classification of five points, with a failing placement of $x_5$ presented in Figure 6 (consider the diagonals). Therefore, $\text{VC-dim}(H_{\square}) = 4$. 

9
Figure 6: What is the VC-dimension of $\mathcal{H}_{\square}$?

We end the lecture with the statement of an important result that we will prove next time.

**Theorem 3.12. (Fundamental Theorem of Statistical Learning)**

$\mathcal{H}$ is learnable if and only if $\text{VC-dim} (\mathcal{H}) < \infty$. If this condition holds, then

$$m_{\mathcal{H}} (\epsilon, \delta) = \Theta \left( \frac{\text{VC-dim} (\mathcal{H})}{\epsilon} \ln \left( \frac{1}{\delta} \right) \right)$$

(9)

In the case of Example 3.5, we can understand our inability to learn with our selected hypothesis class $\mathcal{H}_\lor$ as the result of $\text{VC-dim} (\mathcal{H}_\lor)$ being too large.