Alin Tomescu

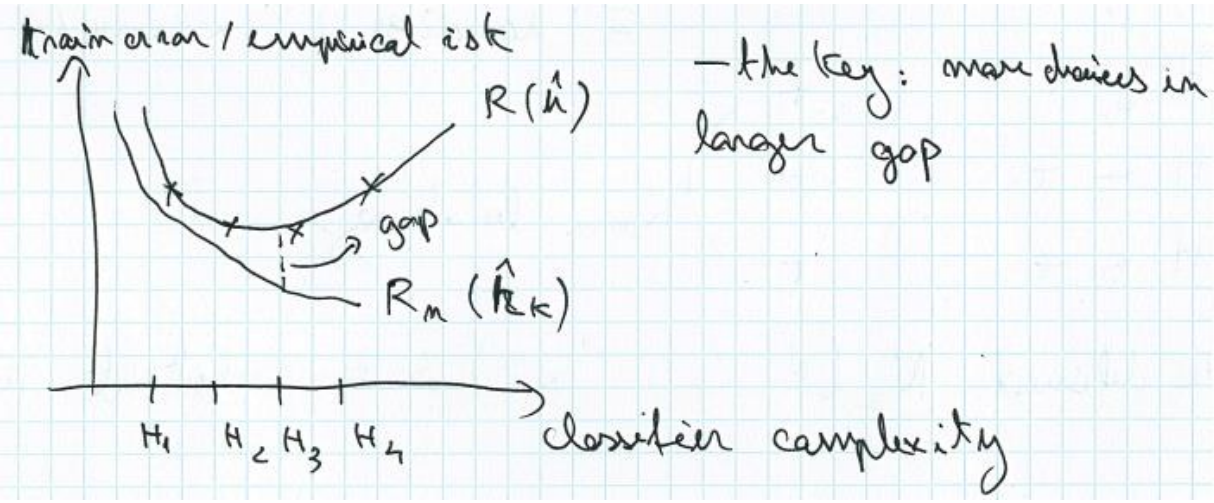
Lecture 13

Previously, we were looking at having empirical risk as just a fraction of misclassified examples:

$$R_n(h) = \frac{1}{n}\sum_{i=1}^{n}[[y_i \neq h(x_i)]]$$

$$R(h) = E_{(x,y)\sim p}\{[[y \neq h(x)]]\}$$

If I have a set of classifiers $\mathcal{H}$ and I pick $\hat{h} \in \mathcal{H}$, how well will it generalize?



Key idea: The more choices you have for your classifier, the bigger the gap between training and generalization error.

We looked (will look) at:

- Finite set of classifiers $\mathcal{H}$, $|\mathcal{H}| < \infty$
- Infinite set of classifiers (eg., set of linear classifiers, uncountable)
- Distributions over classifiers

Case 2: $|\mathcal{H}| < \infty, \forall h^* \in \mathcal{H}, R(h^*) \neq 0$

Assume $|\mathcal{H}| < \infty$ (finite), then we can say with probability at least $1 - \delta$ (and we can fix δ: you tell me the confidence that you want), for all classifiers in my set, that **the generalization error will not be too far from the training error** and the gap is related to the size of the set of classifiers.

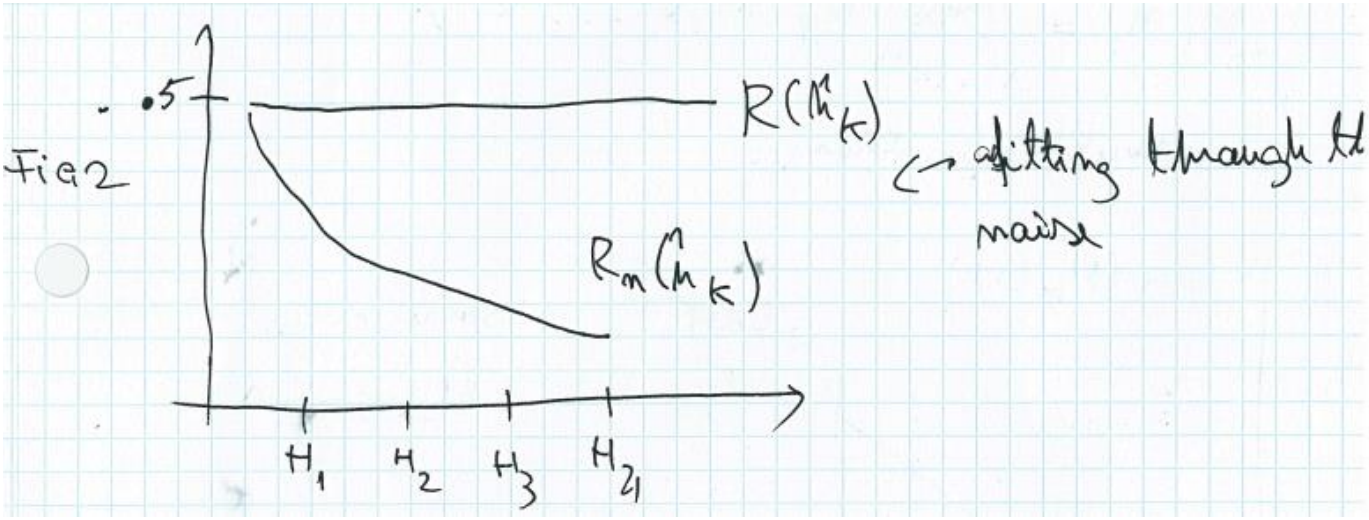
$$R(h) \leq R_n(H) + \sqrt{\frac{\log|\mathcal{H}| + \log 1/\delta}{2n}}$$

What does this mean? The gap will increase logarithmically with the size/complexity of the set of classifiers and will decrease with a larger number of training examples

$$\text{complexity} \uparrow \Rightarrow \text{gap} \uparrow$$

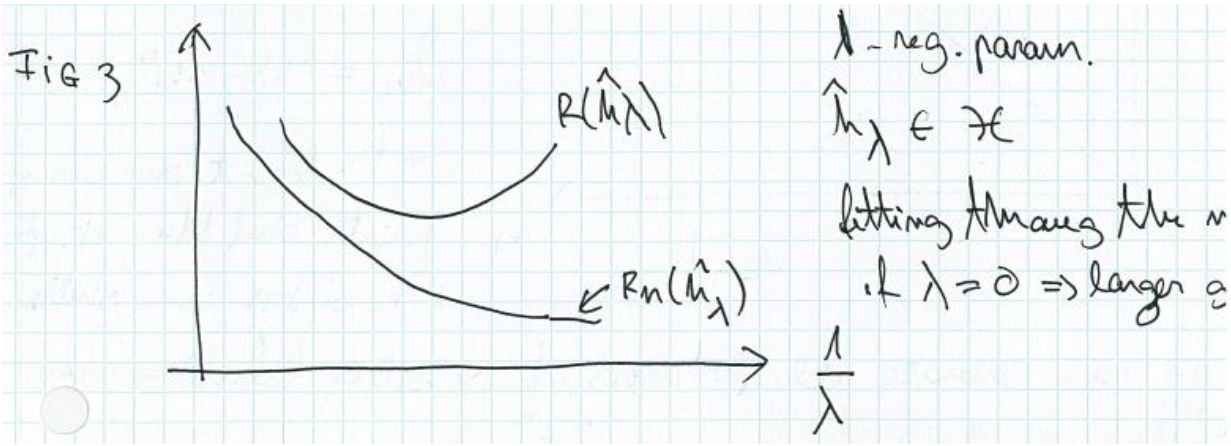
$$\#\text{ of training examples} \uparrow \Rightarrow \text{gap} \downarrow$$

Trivial example: Labels are completely random. So this is a classification task you don't want to solve since there's no relation between x and y.



Here, since the y labels are random and independent of the x values, increasing the complexity will only fit the noise in the data and will only serve to increase the training/generalization error gap, as the training error is decreased.

In general, **regularization can help avoid fitting through the noise** and result in a smaller gap, as can be seen below:



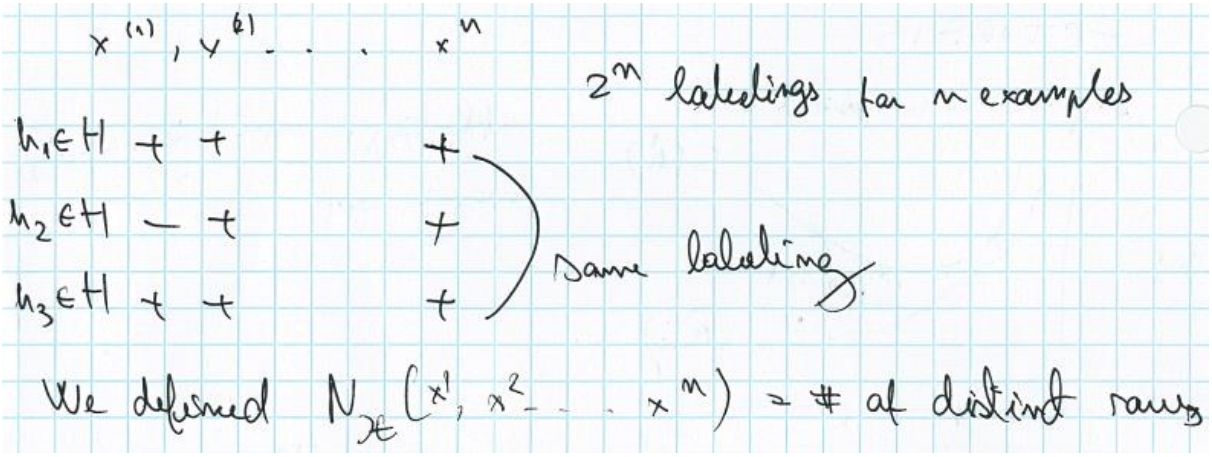
Case 3: $|\mathcal{H}| = \infty$

If we have a fixed $\mathcal{H}$ and an $S_n = \{x_1, x_2, \ldots, x_n\}$ and we pick a classifier h_i that gives a particular labeling of the training examples.

We defined $N_{\mathcal{H}}(x_1, x_2, \ldots, x_n)$ to be **the number of distinct labelings** that we can generate with classifiers from $\mathcal{H}$ on the $x_1, x_2, \ldots, x_n$ training set, aka a **growth function**.

Alin Tomescu

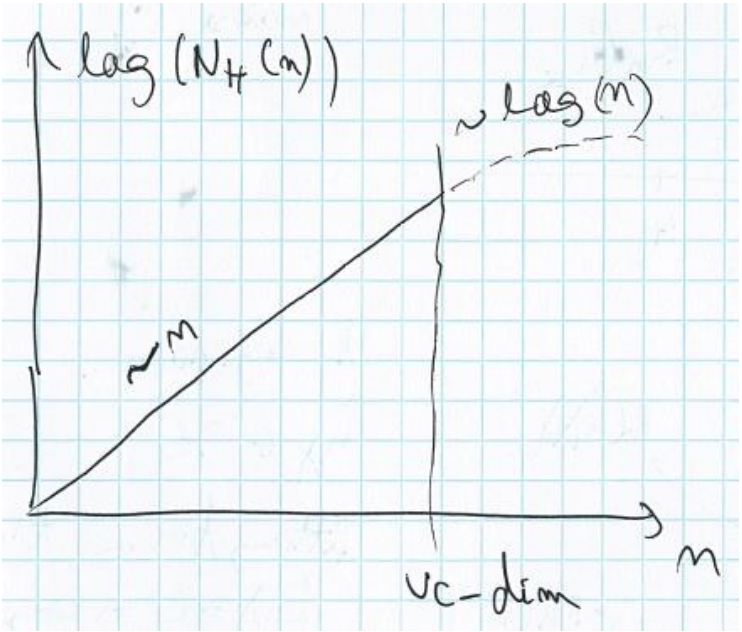
Note: There are 2^n distinct labelings for n items in a training set:

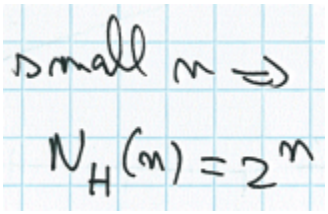


We can defined $N_{\mathcal{H}}(n) = \max_{x_1, x_2, \ldots, x_n} N_{\mathcal{H}}(x_1, x_2, \ldots, x_n)$ as a better measure over all the training examples.

Definition: $N_{\mathcal{H}}(n)$ the **maximum** number of distinct labelings that we can generate with classifiers from $\mathcal{H}$ over a *particular* training set of n examples. ***(Important:** Not over all training sets however)*

"For this training set of size n we can label it in $N_{\mathcal{H}}(n)$ ways using well-picked classifiers from $\mathcal{H}$"





$$d_{VC} = \max\{n : N_{\mathcal{H}}(n) = 2^n\}$$

Definition: *VC-dimension* is the **largest** number of points (in some configuration) that the set of classifiers can shatter (i.e. that the classifier can generate all instances of possible labelings over training examples, or that the classifier can classify independent of the labeling)

The VC dimension of the class $\{ f(x, \alpha) \}$ is defined to be the largest number of points (in some configuration) that can be shattered by members of $\{ f(x, \alpha) \}$.

Alin Tomescu

"Is there a training set of size n that we can shatter? Can we still shatter sets of size $n+1$?"

Important note: By definition, if $\text{VCdim}(H) = d$, there exists a set of size d that can be fully shattered. But, this does not imply that all sets of size d or less are fully shattered, in fact, this is typically not the case.

Consequence: If you are told a set of classifiers has VC-dimension d_{VC}, then to confirm or prove it:

- Find a set of size d_{VC} that can be shatter (pick/place your points wisely)
- Prove that any set of larger size cannot be shattered (this means that for any placement/positioning of points, there exists some labeling that cannot be achieved/created/reproduced by a classifier in the set)

The *VC-dimension* will represent the complexity of our sets of classifiers.

$$N_{\mathcal{H}}(n) = \begin{cases} 2^n, n \leq d_{VC} \\ \leq \left(\frac{en}{d_{VC}}\right)^{d_{VC}}, n > d_{VC} \end{cases}$$

Examples: RBF and linear classifiers

For radial basis kernel, the VC-dimension is infinite because it can shatter an infinite number of points.

What makes a classifier with infinite VC-dimension still good for generalization? The notion of margin. (RBF kernel margin is not good?)

Linear classifiers in $2D \Rightarrow d_{VC} = 3$

Linear classifiers in $\mathbb{R}^d \Rightarrow d_{VC} = d + 1$ (the # of parameters in $\vec{\theta} \in \mathbb{R}^d$ and $\theta_0 \in \mathbb{R}$)

$$S_n = \{(x_1, y_1), (x_2, y_2) \dots, (x_n, y_n)\}$$

$$\vec{x}_i = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} = 1 \text{ in } i^{th} \text{ position}$$

$$\overrightarrow{x_{d+1}} = \vec{0}$$

Solution: I can find a θ and θ_0 that classify these correctly (I can set θ and θ_0 such that $sign\,(\vec{\theta}\vec{x}_i + \theta_0) = sign\,(\theta_i + \theta_0)$ will classify example i only).

We can take care of the last example $\overrightarrow{x_{d+1}} = \vec{0}$ using θ_0:

$$\theta_0 = y_{d+1}$$

The rest of the examples $\vec{x}_i$ can be classified by "overpowering" θ_0:

$$\theta_i = 2y_i$$

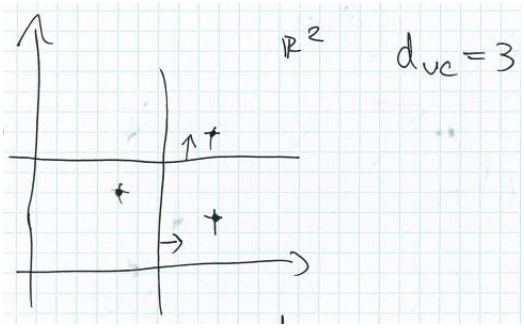
Now we have to show we can't shatter a larger set. This is because any extra $\vec{x}_{new}$ examples will be a linear combination of the previous $d + 1$ ones (look at HW6 exercise 1, part c). This means that the corresponding $\vec{y}_{new}$ label will be

determined by how the $d+1$ $\vec{x}_i$ points were classified, which means that, by changing the label, $\overrightarrow{y_{new}}$ could not be classified correctly.

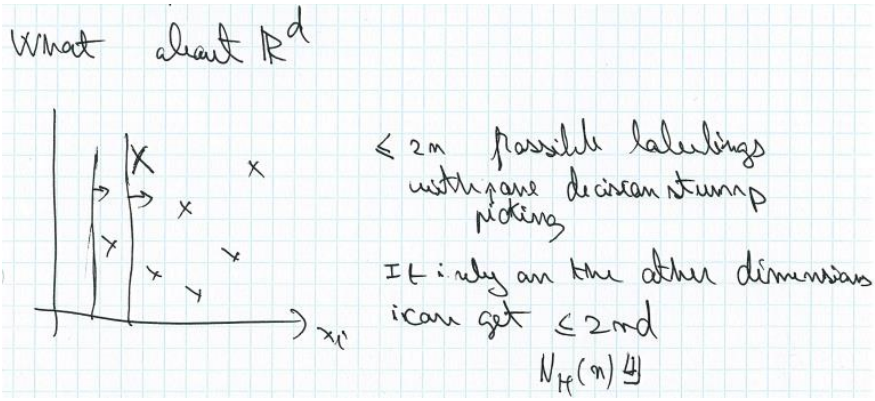
Examples: Decision stumps and ensembles

Decision stumps: how powerful are they?

In 2D, a decision stump can shatter (a particular set of) 3 points, but cannot shatter any 4 points. $d_{VC} = 3$



In $\mathbb{R}^d$:



Explanation (informal):

- Consider decision stumps in $\mathbb{R}^2$. Limit yourself to only picking one decision stump on the x-axis. Worst case is if all points are on a line parallel to the x-axis (or on some sort of curve above the x-axis, as long as no two points have the same y coordinate). Then we can place our decision stumps anywhere in between the points or on the left of the left-most point. We get 2 labelings for each placement. There are n placements, so we get $2n$ labelings. (The reason we don't consider a decision stump on the right of the right-most point is because it gives us the same labelings as the decision stump on the left of the left-most point).
- If we remove the limitation of only picking our decision stump on the x-axis then we add at most another $2n$ labelings.
- Thus, in $\mathbb{R}^2$, we can get at most $2n + 2n$ labelings: $N_{\mathcal{H}}(n) = 4n$
- In general in $\mathbb{R}^d$, we can get at most $2nd$ labelings: $N_{\mathcal{H}}(n) = 2nd$
- Since $d_{vc} = \max\{\text{n}: N_{\mathcal{H}}(n) = 2^n\}$ and $N_{\mathcal{H}}(n) \leq 2nd$, then the maximum n for which $N_{\mathcal{H}}(n) = 2^n$ would need to have $N_{\mathcal{H}}(n) = 2^n \leq 2nd$.

Thus, for a single decision stump, we have $d_{vc} = \max_{\text{n}}\{n: 2nd \geq 2^n\} \cong \log d$

The VC-dimension of an ensemble with m stumps is $\geq \frac{m}{2}$ (Why?)

Theorem: If we know d_{VC} for $\mathcal{H}$, then ∃ a set of points $x_1, x_2, \ldots, x_n$, with $n < d_{VC}$ such that the following holds:

> I can find $h \in \mathcal{H}$ that correctly classifies all the training examples. I can also find another classifier that also reproduces those labels since $d_{VC} > n$, but for these two classifiers I can find a point x_{n+1} that they will disagree on.

Finally, we can modify the gap result in the infinite case. With probability at least $1 - \delta$:

$$R(h) \leq R_n(h) + \sqrt{\frac{\log N_{\mathcal{H}}(2n) + \log\frac{4}{\delta}}{n}}$$

$$\log N_{\mathcal{H}}(n) \leq \log\left[\left(\frac{en}{d_{VC}}\right)^{d_{VC}}\right]$$

$$\Rightarrow \sqrt{\frac{\log N_{\mathcal{H}}(2n) + \log\frac{4}{\delta}}{n}} \leq \sqrt{\frac{\log\left(\left(\frac{2en}{d_{VC}}\right)^{d_{VC}}\right) + \log\frac{4}{\delta}}{n}} = \sqrt{\frac{d_{VC}\log\left(\frac{2en}{d_{VC}}\right) + \log\frac{4}{\delta}}{n}} = \sqrt{\frac{d_{VC}\left(1 + \log\left(\frac{2n}{d_{VC}}\right)\right) + \log\frac{4}{\delta}}{n}}$$

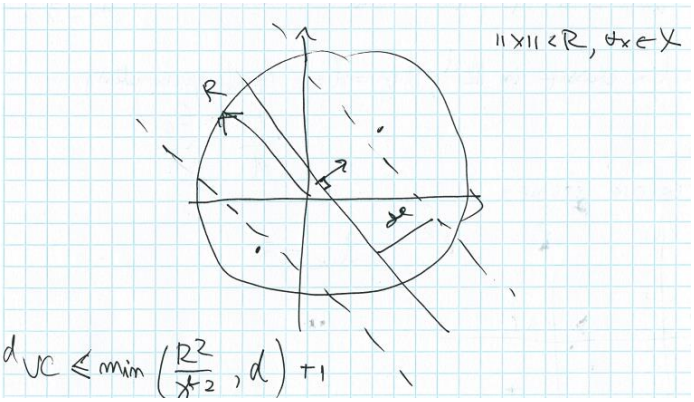
Thus,

$$R(h) \leq R_n(h) + \sqrt{\frac{d_{VC}\left(1 + \log\left(\frac{2n}{d_{VC}}\right)\right) + \log\frac{4}{\delta}}{n}}$$

Infinite VC-dimension

How do we handle cases where VC-dimension is infinite? We need some notion of margin.

Consider the case for linear classifiers in $\mathbb{R}^d$ where all examples are bounded by a circle: $\|x\| < R, \forall x \in \mathcal{X}$



Then, $d_{VC} \leq \min\left(\frac{R^2}{\gamma^2}, d\right) + 1$