0.1 Proof of the "Overfit theorem"

(Lesson 3 of 09/10/19) Compiled: 10 ottobre 2019

Note that in the "overfit theorem" the amount of samples does not depend on the form of f, nor on the specific underlying distribution D.

Proof.
Denote with P_{good} the probability to find a "good" h_S, such that $L_{D,f}(h_S) \leq \varepsilon$. We want to prove that $P_{\text{good}} \geq 1 - \delta$, which is equivalent to $P_{\text{bad}} \leq \delta$ (with $P_{\text{bas}} = 1 - P_{\text{bad}}$).
The idea is to consider the set of all possible training samples, that is m-tuples. There are some samples that are "misleading", meaning that they result in a $L_{D,f}(h_S) \geq \varepsilon$, while the other lead to $L_{D,f}(h_S) \leq \varepsilon$ that we want. The essence of the proof relies in finding a bound on these "misleading samples" size.

We start with denoting $S|_x = (x_1 \dots x_m)$ a m-tuple which will be used as a training set. Then, let h_S be the ERM solution, that satisfies $L_S(h_S) = 0$ (minimizes the training error).
The probability to get a "bad solution" is then:

$$P_{\text{bad}} = D^m(S|_x \colon L_{D,f}(h_S) > \varepsilon)$$

That is the probability to sample from D a m-tuple which leads to a generalization error higher than ε.
Then the set of "bad hypotheses" is:

$$\mathcal{H}_B = \{h \in \mathcal{H} \colon L_{D,f}(h) > \varepsilon\}$$

The set of "misleading samples" contains all the m-tuples which lead to a "bad hypothesis" after applying the ERM algorithm:

$$M = \{S|_x \colon \exists h \in \mathcal{H}_B, L_S(h) = 0\} = \bigcup_{h \in \mathcal{H}_B} \{S|_x \colon L_S(h) = 0\}$$

Note that:

$$D^m(\{S|_x \colon L_{D,f}(h_S) > \varepsilon\}) \leq D^m(M) = D^m\left(\bigcup_{h \in \mathcal{H}_B} \{S|_x \colon L_S(h) = 0\}\right) \quad (1)$$

because of course the ERM algorithm can produce a subset of the "bad hypotheses". Then we make use of the *union bound* :

$$D(A \cup B) \leq D(A) + D(B)$$

In fact, if A and B where disjoint, then $D(A \cup B) = D(A) + D(B)$. However, if $A \cap B \neq \emptyset$, then $D(A \cup B) < D(A) + D(B)$. This can be proved more formally, but we will not do that here.

Using the union bound (U.B.) we arrive at:

$$(??) \underset{\text{U.B.}}{\leq} \sum_{h \in \mathcal{H}_B} D^m(\{S|_x \colon L_S(h) = 0\})$$

All the ERM solutions are "perfect" when evaluated on the training set, meaning that they correctly classify all the samples:

$$D^m(\{S|_x \colon L_S(h) = 0\}) = D^m(S|_x \colon \forall i \, h(x_i) = x_i) \tag{2}$$

Recall that x_i are i.i.d, and so:

$$(??) = \prod_{i=1}^{m} D(x_i \colon h(x_i) = y(x_i)) \tag{3}$$

as the joint probability of independent events is merely the product of individual probabilities.

We can then estimate this probability for samples in $\mathcal{H}_B$. Recall that the generalization error is the probability of misclassification, and we can simply take its complementary:

$$D(\{x_i \colon h(x_i) = y(x_i)\}) = 1 - L_{D,f}(h) \leq 1 - \varepsilon$$

As in $\mathcal{H}_B$ we have, by definition, $L_{D,f}(h) > \varepsilon$.

Substituting this result in (**??**) we arrive at:

$$D^m(\{S|_x \colon L_S(h) = 0\}) \leq \prod_{i=1}^{m} (1 - \varepsilon) = (1 - \varepsilon)^m \leq e^{-\varepsilon m}$$

So, by applying (**??**), we know that:

$$P_{\text{bad}} = D^m(\{S|_x \colon L_{D,f}(h_S) > \varepsilon\}) \leq \sum_{h \in \mathcal{H}_B} e^{-\varepsilon m} = |\mathcal{H}_B| e^{-\varepsilon m} \underset{\mathcal{H}_B \subset \mathcal{H}}{\leq} |\mathcal{H}| e^{-\varepsilon m}$$

Finally, we have arrived at:

$$P_{\text{bad}} \leq |\mathcal{H}| e^{-\varepsilon m} \overset{!}{\leq} \delta$$

We then find a bound on m, by taking the log of both sides:

$$e^{-\varepsilon m} \leq \frac{\delta}{|\mathcal{H}|} \Rightarrow -\varepsilon m \leq \log\left(\frac{\delta}{|\mathcal{H}|}\right) \Rightarrow m \geq -\frac{1}{\varepsilon} \log\left(\frac{\delta}{|\mathcal{H}|}\right) \Rightarrow m \geq \frac{1}{\varepsilon} \log\left(\frac{|\mathcal{H}|}{\delta}\right)$$

Note that this theorem proves only a sufficient condition: so it is possible (and indeed happens) to have a good learner even with smaller training datasets (m lower than the bound).

0.2 Generalization

Definition 1. A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}}\colon (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that for every $\delta, \varepsilon \in (0,1)$, for every distribution D over $\mathcal{X}$, and for every labelling function $f\colon \mathcal{X} \to \{0,1\}$, if the realizability assumption holds with respect to $\mathcal{H}$, D, f, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d. examples generated by D and labeled by f, the algorithm returns a hypothesis h such that, with probability $\geq 1 - \delta$ (over the choice of examples):

$$L_{D,f}(h) \leq \varepsilon$$

$m_{\mathcal{H}}\colon (0,1)^2 \to \mathbb{N}$ is called the **sample complexity** of learning $\mathcal{H}$, and $m_{\mathcal{H}}$ is the minimal integer that satisfies the requirements.

Corollary. Every finite hypothesis class is PAC learnable with sample complexity:

$$m_{\mathcal{H}}(\varepsilon, \delta) \leq \frac{1}{\varepsilon} \log\left(\frac{|\mathcal{H}|}{\varepsilon}\right)$$

Let examine more closely the assumptions we made:

- **Realizability assumption**: $\exists h^* \in \mathcal{H}$ such that $L_{D,f}(h) = 0$.
  This is a condition too strong for many real world applications.
- **Function f**: in many applications it isn't possible to fully determine labels from the measured features, because of some intrinsic ambiguity in the data. So there is no function such that $y_i = f(x_i)$.
  So, we need to use a stochastic approach, considering a probability distribution over the set of couples feature-label $D(\mathcal{X}, \mathcal{Y})$. So, given a x_i, we can compute a certain probability that the label will be y_i.
  Note that this precludes the existence of a perfect classifier - and so the realizability assumption must be dropped.

Defining D as a probability distribution over $\mathcal{X} \times \mathcal{Y}$, the generalization error needs to be redefined:

$$L_D(h) \stackrel{\text{def}}{=} \mathbb{P}_{(x,y)\sim D}[h(x) \neq y] \stackrel{\text{def}}{=} D(\{(x,y)\colon h(x) \neq y\})$$

Note that, differently than before, we don't have a labelling function f anymore, and we also sample y from D.
As before, however, D is not known to the learner, who only knows the training data S.

Also the empirical risk can be adapted:

$$L_S(h) \stackrel{\text{def}}{=} \frac{1}{m}|\{i, 0 \leq i \leq m\colon h(x_i) \neq y_i\}|$$

Note that $L_S(h)$ is the probability that for a pair (x_i, y_i) taken uniformly at random from S, the event $h(x_i) \neq y_i$ holds.

0.2.1 Bayes Optimal Predictor

We now want an algorithm for finding $h\colon \mathcal{X} \to \mathcal{Y}$ that minimizes $L_D(h)$.

Given a probability distribution D over $\mathcal{X} \times \{0,1\}$, the best predictor is the **Bayes Optimal Predictor**:

$$f_D(x) = \begin{cases} 1 & \text{if } \mathbb{P}[y=1|x] \geq 1/2 \\ 0 & \text{otherwise} \end{cases}$$

Basically, if we know that the probability of x being classified as y is better than chance ($> 1/2$), then we classify x as y.
Proposition. For any classifier $g\colon \mathcal{X} \to \{0,1\}$, it holds:

$$L_D(f_D) \leq L_D(g)$$

However, we do not know how to compute $\mathbb{P}[y=1|x]$, as this would require knowing D.

0.2.2 Agnostic PAC Learnability

As finding the Bayes Optimal Predictor is not feasible, we do not require it for our algorithm.
However, we desire to have a good estimate, that is not too far away from the BOP. We then introduce the following definition:

Definition 2. A hypothesis class $\mathcal{H}$ is **agnostic** PAC learnable if there exist a function $m_{\mathcal{H}}\colon (0,1)^2 \to \mathbb{N}$ and a learning algorithm such that for every $\delta, \varepsilon \in (0,1)$ and for every distribution D over $\mathcal{X} \times \mathcal{Y}$, when running the algorithm on $m \geq m_{\mathcal{H}}(\varepsilon, \delta)$ i.i.d. examples generated by D the algorithm returns a hypothesis h such that, with probability $\geq 1-\delta$ (over the choice of the m training examples):

$$L_D(h) \leq \min_{h' \in \mathcal{H}} L_D(h') + \varepsilon$$

Note that:

- We dropped the requirement to get the best possible solution
- We dropped the realizability theorem, which would mean that:

  $$\min_{h' \in \mathcal{H}} L_D(h') = 0$$

  returning to the previous definition.