Machine Learning — Statistical Methods for Machine Learning

Risk analysis for tree predictors

Instructor: *Nicolò Cesa-Bianchi* version of August 2, 2019

The risk analysis for ERM over a finite class $\mathcal{H}$ of predictors states that, with probability at least $1-\delta$ with respect the random draw of training set of size m, we have

$$\ell_{\mathcal{D}}(\widehat{h}) \leq \min_{h\in\mathcal{H}} \ell_{\mathcal{D}}(h) + \sqrt{\frac{2}{m}\ln\frac{2|\mathcal{H}|}{\delta}}\ . \tag{1}$$

We can see what happens when applying this result to the class $\mathcal{H}$ of predictors computed by tree classifiers over $\mathcal{X} = \{0,1\}^d$ (i.e., d binary attributes).

Fact 1. *The set of all classifiers computed by tree predictors on $\{0,1\}^d$ contains all functions of the form $h : \{0,1\}^d \to \{-1,1\}$.*

Proof. To see that, consider a complete binary tree with 2^d leaves. The root node implements a binary test on x_1, the 2 nodes at depth 1 implement binary tests on x_2, and so on until the 2^{d-1} nodes at depth $d-1$ which test x_d. Now note that any path from root to a leaf corresponds to a binary sequence in $\{0,1\}^d$. Given any $h : \{0,1\}^d \to \{-1,1\}$, we can assign a label $y_\ell \in \{-1,1\}$ to each leaf ℓ so that if the path to the leaf corresponds to $\boldsymbol{x} \in \{0,1\}^d$, then the label is set to $h(\boldsymbol{x})$. The classifier computed by the tree then corresponds to h. □

Since there are 2^{2^d} binary functions over $\{0,1\}^d$, $|\mathcal{H}| = 2^{2^d}$. The upper bound (1) then becomes

$$\ell_{\mathcal{D}}(\widehat{h}) \leq \min_{h\in\mathcal{H}} \ell_{\mathcal{D}}(h) + \sqrt{\frac{2}{m}\left(2^d \ln 2 + \ln\frac{2}{\delta}\right)}\ .$$

Therefore, in order to make the risk of ERM small, the training set must contain a number m of training examples of the order of 2^d, which is the cardinality of $\mathcal{X} = \{0,1\}^d$. This is a truly extreme case of overfitting.

In order to reduce overfitting, we can minimize training error within a smaller class of trees. Consider the set $\mathcal{H}_N$ of all classifiers computed by tree predictors with N nodes on $\{0,1\}^d$, where $N \ll 2^d$.

Fact 2. $|\mathcal{H}_N| \leq (2de)^N$.

Proof. Note that $|\mathcal{H}_N|$ can be expressed as the product of: the number of binary trees with N nodes, the number of ways of assigning binary tests to attributes at the internal nodes, the number of ways of assigning binary labels to the leaves. If we conventionally assign the left child of a node to the negative result of a test, and the left child to a positive result, a test is uniquely identified just by the index of the tested attribute. Therefore, if the tree has M internal nodes, there are d^M ways of assigning tests to internal nodes. Moreover, since there are $N-M$ leaves, there are 2^{N-M} ways of assigning binary labels to leaves. Therefore, each tree of N nodes can implement

up to $d^M 2^{N-M} \leq d^N$ (since $d \geq 2$) classifiers. Finally, the number of binary trees with N nodes is given by the $(N-1)$-th Catalan number $C_{N-1} = \frac{1}{N}\binom{2N-2}{N-1}$. Thus, using the standard upper bound $\binom{n}{k} \leq \left(\frac{en}{k}\right)^k$ derived from Stirling approximation to binomial coefficients, we get

$$|\mathcal{H}_N| \leq \frac{1}{N}\left(\frac{2e(N-1)}{N-1}\right)^{N-1} d^N \leq (2ed)^N$$

concluding the proof. □

Hence, if $\widehat{h} = \operatorname{argmin}_{\mathcal{H}_N} \widehat{\ell}(h)$ for a given N, the upper bound (1) becomes

$$\ell_{\mathcal{D}}(\widehat{h}) \leq \min_{h \in \mathcal{H}_N} \ell_{\mathcal{D}}(h) + \sqrt{\frac{2}{m}\left(N\big(1+\ln(2d)\big) + \ln\frac{2}{\delta}\right)} \ .$$

From that, we deduce that in this case a training set of size of order $N \ln d$ is enough to control the risk of $\widehat{h} \in \mathcal{H}_N$.

The result we just showed holds for a specific predictor, the one minimizing the training error in $\mathcal{H}_N$ for a given and fixed N. In practice it is not clear what N should be used. In order to circumvent this problem, instead of bounding the variance error of the empirical error minimizer, as we did so far, we take a different approach. Namely, we simultaneously upper bound the risk of all tree predictors, where the risk bound for each tree depends on both its training error and on its number of nodes. To this purpose, we introduce a function $w : \mathcal{H} \to [0,1]$ and call $w(h)$ the weight of predictor h. We assume

$$\sum_{h \in \mathcal{H}} w(h) \leq 1 \ . \tag{2}$$

We can then write the following chain of inequalities, where $\varepsilon_h > 0$ will be chosen later on,

$$\mathbb{P}\left(\exists h \in \mathcal{H} : \ \big|\widehat{\ell}(h) - \ell_{\mathcal{D}}(h)\big| > \varepsilon_h\right) \leq \sum_{h \in \mathcal{H}} \mathbb{P}\left(\big|\widehat{\ell}(h) - \ell_{\mathcal{D}}(h)\big| > \varepsilon_h\right) \leq \sum_{h \in \mathcal{H}} 2e^{-2m\varepsilon_h^2} \ .$$

Note that we used Chernoff-Hoeffding bound in the last step. Now, choosing

$$\varepsilon_h = \sqrt{\frac{1}{2m}\left(\ln\frac{1}{w(h)} + \ln\frac{2}{\delta}\right)}$$

we get that

$$\mathbb{P}\left(\exists h \in \mathcal{H} : \ \big|\widehat{\ell}(h) - \ell_{\mathcal{D}}(h)\big| > \varepsilon_h\right) \leq \sum_{h \in \mathcal{H}} \delta w(h) \leq \delta$$

where we used the property (2) of the function w.

A consequence of this analysis is that, with probabilty at least $1-\delta$ with respect to the training set random draw, we have

$$\ell_{\mathcal{D}}(h) \leq \widehat{\ell}(h) + \sqrt{\frac{1}{2m}\left(\ln\frac{1}{w(h)} + \ln\frac{2}{\delta}\right)} \tag{3}$$

simultaneously for every $h \in \mathcal{H}$. This suggests an alternative algorithm to training error minimization: while ERM uses

$$\widehat{h} = \operatorname*{argmin}_{h \in \mathcal{H}_N} \widehat{\ell}(h)$$

for a given N, the new approach leads to the choice

$$\widehat{h} = \operatorname*{argmin}_{h \in \mathcal{H}} \left(\widehat{\ell}(h) + \sqrt{\frac{1}{2m} \left(\ln \frac{1}{w(h)} + \ln \frac{2}{\delta} \right)} \right) \ . \tag{4}$$

The function w can be naturally viewed as a complexity measure for the predictor h. Note that this analysis offers a different viewpoint on overfitting: $\widehat{\ell}(h)$ becomes a good estimate of $\ell_{\mathcal{D}}(h)$ when it is "penalized" by the term

$$\sqrt{\frac{1}{2m} \left(\ln \frac{1}{w(h)} + \ln \frac{2}{\delta} \right)}$$

this accounts for the fact that we used the m training examples to choose a predictor h of complexity $w(h)$.

We now view a concrete example for tree predictors on $\mathcal{X} = \{0,1\}^d$. Let $\mathcal{H}$ be the set of 2^{2^d} tree predictors encoding all classifiers $h : \{0,1\}^d \to \{-1,1\}$. Using coding theoretic techniques, we can encode each tree predictor h with N_h nodes using a binary string $\sigma(h)$ of length $|\sigma(h)| = (N_h + 1)\lceil \log_2(d+3) \rceil + 2\lfloor \log_2 N_h \rfloor + 1 = \mathcal{O}(N_h \log d)$, so that there are no two predictors h and h' such that $\sigma(h)$ is a prefix of $\sigma(h')$. Codes of this kind are called *instantaneous* and always satisfy the Kraft inequality

$$\sum_{h \in \mathcal{H}} 2^{-|\sigma(h)|} \leq 1 \ .$$

Thanks to Kraft inequality —which implies property (2)— we can assign weight $w(h) = 2^{-|\sigma(h)|}$ to a classifier h computed by a tree predictor with N_h nodes. Applying bound (3) we get that, with probability at least $1 - \delta$ with respect to the training set random draw,

$$\ell_{\mathcal{D}}(h) \leq \widehat{\ell}(h) + \sqrt{\frac{1}{2m} \left(|\sigma(h)| + \ln \frac{2}{\delta} \right)} \qquad \text{con } |\sigma(h)| = \mathcal{O}(N_h \log d)$$

simultaneously for each $h \in \mathcal{H}$. Hence, a learning algorithm for tree predictors can control overfitting by generating predictors $\widehat{h}$ defined by

$$\widehat{h} = \operatorname*{argmin}_{h \in \mathcal{H}} \left(\widehat{\ell}(h) + \sqrt{\frac{1}{2m} \left(|\sigma(h)| + \ln \frac{2}{\delta} \right)} \right) \ .$$

This kind of analysis justifies the empirical observation that, in a set of classifiers with the same value of training error the least complex classifier is generally performing best. In contrast to that, note the choice of the weight function w is not determined by the analysis. In particular, we may choose any other w satisfying (2). We should then interpret w as a bias term, giving preference to certain trees as opposed to others. A bias towards smaller trees is an instance of the principle known as *Occam Razor*: if two explanations agree with a set of observations, then the shortest explanation is the one with the biggest predictive power.