Evaluation of the Learning Performances

We have a set of examples and a learning algorithm of which we can tune certain parameters. This algorithm returns a hypothesis. How can we evaluate the performances of this hypothesis

A solution consists in applying the theoretical results that provide probability bounds on the real risk according to the empirical risk. These bounds have the general form:

$$R_{real}(h) = R_{emp}(h) + \Phi\left(d_{VC}(\mathcal{H}), m\right)$$

where Φ is a function of the Vapnik-Chervonenkis dimension of the hypothesis space of $\mathcal{H}$ and m is the sample size of training $\mathcal{S}$. If one can obtain in theory asymptotically tightened bounds, the assumptions to be made to compute in practice, $\Phi\left(d_{VC}(\mathcal{H}), m\right)$ imply often such margins that the obtained bounds are too loose and do not allow to estimate the real performance precisely. It is why, except favorable particular cases, the estimate of the learning performance is estimated generally, by empirical measurements.

L. Devroye, L. Gyorfi, G. Lugosi – *A Probabilistic Theory of Pattern Recognition* (Cap.8), Springer, 1996

ERROR COUNTING

Estimating the error probability

$$L_n = \mathbf{P}\{g_n(X) \neq Y | D_n\}$$

of a classification function g_n is of essential importance.

Suppose that we want to estimate the error probability of a classifier g_n designed from the training sequence

$$T_m = ((X_{n+1}, Y_{n+1}), \ldots, (X_{n+m}, Y_{n+m}))$$

is available, which is a sequence of i.i.d. pairs that are independent of (X,Y) and D_n and that are distributed as (X,Y). An obvious way to estimate L_n is to count the number of errors that g_n commits on T_n.

The *error-counting estimator* $\widehat{L}_{n,m}$ is defined by the relative frequency

$$\widehat{L}_{n,m} = \frac{1}{m} \sum_{j=1}^{m} I_{\{g_n(X_{n+j}) \neq Y_{n+j}\}}$$

The estimator is clearly *unbiased* in the sense that

$$\mathbf{E}\left\{\widehat{L}_{n,m} | D_n\right\} = L_n$$

and the conditional distribution of $m\widehat{L}_{n,m}$ given the training data D_n is binomial with parameters m and L_n.

A distribution-free performance bound for the counting error estimate is given by:

COROLLARY 8.1 *For every* $\varepsilon > 0$

$$\mathbf{P}\left\{ \left| \widehat{L}_{n,m} - L_n \right| > \epsilon \,\middle|\, D_n \right\} \leq 2e^{-2m\epsilon^2}$$

The variance of the estimate can easily be computed using the fact that, conditioned on the data D_n, $m\widehat{L}_{n,m}$ is binomially distributed:

$$\mathbf{E}\left\{ \left(\widehat{L}_{n,m} - L_n \right)^2 \,\middle|\, D_n \right\} = \frac{L_n(1 - L_n)}{m} \leq \frac{1}{4m}.$$

SELECTING CLASSIFIERS

Probably the most important application of error estimation is the selection of a classification function from a class C of function. If a class of classifiers is given, then it is tempting to pick the one that minimizes an estimate of the error probability over the class.

Let C be a class of classifiers, that is, a

class of mappings of the form $\phi : \mathcal{R}^d \to \{0, 1\}$. Assume that the error count

$$\widehat{L}_n(\phi) = \frac{1}{n} \sum_{j=1}^{n} I_{\{\phi(X_j) \neq Y_j\}}$$

is used to estimate the error probability $L(\phi) = \mathbf{P}\{\phi(X) \neq Y\}$ of each classifier $\phi \in \mathcal{C}$. Denote by ϕ_n^* the classifier that minimizes the estimated error probability over the class:

$$\widehat{L}_n(\phi_n^*) \leq \widehat{L}_n(\phi) \quad \text{for all } \phi \in \mathcal{C}.$$

i.e.

$$L(\phi_n^*) = \mathbf{P}\left\{\phi_n^*(X) \neq Y \,\middle|\, D_n\right\}$$

REMARK. WITHOUT TESTING DATA. Very often, a class of rules C of the form $\phi_n(x) = \phi_n(x, D_n)$ is given, and the same data D_n are used to select a rule by minimizing some estimates $\widehat{L}_n(\phi_n)$ of the error probabilities $L(\phi_n) = \mathbf{P}\{\phi_n(X) \neq Y | D_n\}$. A similar analysis can be carried out in this case. In particular, if ϕ_n^* denotes the selected rule, then we have similar to Lemma 8.2:

Theorem 8.4.

$$L(\phi_n^*) - \inf_{\phi_n \in C} L(\phi_n) \leq 2 \sup_{\phi_n \in C} |\widehat{L}_n(\phi_n) - L(\phi_n)|.$$

and

$$|\widehat{L}_n(\phi_n^*) - L(\phi_n^*)| \leq \sup_{\phi_n \in C} |\widehat{L}_n(\phi_n) - L(\phi_n)|.$$

ESTIMATING THE BAYES ERROR

It is important to have a good estimate of the optimal error probability L^*. First of all, if L^* is large, we would know beforehand that any rule is going to perform poorly.

Clearly, if the estimate $\widehat{L}_n$ we use is consistent in the sense that $\widehat{L}_n - L_n \to 0$ with probability one as $n \to \infty$, and the rule is strongly consistent, then

$$\widehat{L}_n \to L^*$$

Unfortunately, there is no method that guarantees a certain finite sample performance for all distributions. This disappointing fact is reflected in the following negative result:

Theorem 8.5. *For every n, for any estimate $\widehat{L}_n$ of the Bayes error probability L^*, and for every $\epsilon > 0$, there exists a distribution of (X, Y), such that*

$$\mathbf{E}\left\{|\widehat{L}_n - L^*|\right\} \geq \frac{1}{4} - \epsilon.$$

ERROR ESTIMATION WITHOUT TESTING DATA

A serious problem concerning the practical applicability of the estimate introduced above is that it requires a large, independent testing sequence. In practice, however, an additional sample is rarely available. One usually wants to incorporate all available (X_i,Y_i) pairs in the decision function. In such cases, to estimate L_n, we have to rely on the training data only.

There are well-known methods:

- cross-validation with the leave-one-out variant,
- bootstrapping,
- jackknife,
- resubstitution,
- smooting.

Analysis of these methods, in general, is clearly a much harder problem, as g_n can depend on D_n in a rather complicated way.