Vashchylko Alexei
group 5130203/20102

In binary classification, we deal with an input space (space of instances) X and and output space (label space) Y. We identify the label space with the set {−1, +1}. Mathematically, the goal is to find a function, called a classifier, that maps inputs from a given input space X to labels in a label space Y = {−1, 1}. In order to do this, we get access to some training points $(X1, Y1), \ldots, (Xn, Yn) \in \mathcal{X} \times \mathcal{Y}$. And in the standard setting of SLT makes the following assumptions:

1. No assumptions on P. The probability distribution P can be any distribution on X × Y.

2. Non-deterministic labels due to label noise or overlapping classes.

The important quantity here is the probability that the label Y is 1, under the condition that the data point under consideration is the point x:

η(x) := P(Y = 1 | X = x).

3. Independent sampling (data points are sampled independently).

4. The distribution P is fixed (there is no "time" parameter)

5. The distribution P is unknown at the time of learning.

Thus the goal of supervised learning is to learn a function f : X → Y. The main measure of the quality of the classifier f is the loss function ℓ, which tells us the “cost” of classifying instance $X \in \mathcal{X}$ as $Y \in \mathcal{Y}$. For example,

$$\ell(X, Y, f(X)) = \begin{cases} 1 & \text{if } f(X) \neq Y \\ 0 & \text{otherwise.} \end{cases}$$

or in regression, where the output variables $Y$ take values that are real numbers rather than class labels, a well-known loss function is the squared error loss function

$\ell(X, Y, f(X)) = (Y - f(X))^2$

That is, the risk of a classifier f is the expected loss of the function f at all points X ∈ X (R(f)).

The optimal classifier under the assumption that the probability distribution is known is the Bayes classifier, which minimizes the risk (expected loss):

$$f_{Bayes}(x) := \begin{cases} 1 & \text{if } P(Y = 1 \mid X = x) \geq 0.5 \\ -1 & \text{otherwise.} \end{cases}$$

We can formulate the standard problem of binary classification as follows:

Given some training points (X1, Y1), ...,(Xn, Yn) which have been drawn independently from some unknown probability distribution P, and given some loss function l , how can we construct a function f : X → Y which has risk R(f) as close as possible to the risk of the Bayes classifier?

In this situation, SLT helps, providing a framework to analyze this situation, to come up with solutions, and to provide guarantees on the goodness of these solution.