Lectures

Ilia Kamyshev

January 6, 2021

Contents

1 The Learning Problem

1.1 Introduction

It is assumed that training data is randomly generated. It is also assumed that there is a probability distribution P defined on Z. P is fixed and unknown for given problem. A sequence of labelled examples of the form (x, y) is presented to the neural network during training. For some positive integer m there is training sample:

$$z = ((x_1, y_1), \ldots, (x_m, y_m)) = (z_1, \ldots, z_m) \in Z^m$$

Random training sample of length m is an element of Z^m distributed according to the product probability distribution P^m. Let's denote the set of all functions the network can approximate as H. Then, an *error* of $h \in H$ will be:

$$error_P(h) = P\{(x, y) \in Z : h(x) \neq y\}$$

The sample error (*observed error*) is defined as:

$$error_z(h) = \frac{1}{m}|\{i : 1 \leq i \leq m \text{ and } h(x_i) \neq y_i\}|$$

Given h after the training is *hypothesis* (the error of this function should have minimum value), so:

$$opt_P(H) = \inf_{g \in H} error_P(g)$$

We may say that h is $\epsilon -$ good if for $\epsilon \in (0,1)$:

$$error_P(h) < opt_P(H) + \epsilon$$

1.2 Formal definition of learning

Let's denote δ as a *confidence parameter* to ensure that the learning algrotihm will be $\epsilon -$ good with probability at least $1-\delta$. Suppose that H maps from a set X to $\{0,1\}$. A learning algorithm L for H is a *function*:

$$L : \bigcup_{m=1}^{\infty} Z^m \to H$$

So if z is a training sample drawn randomly from distribution P^m, then the hypothesis $L(z)$ is such that:

$$error_P(L(z)) < opt_P(H) + \epsilon$$

In other words:

$$P^m\{error_P(L(z)) < opt_P(H) + \epsilon\} \geq 1-\delta$$

H is learnable *if there is a learning algorithm for* H. Let's denote $m_0(\epsilon, \delta)$ as a minimum sample size sufficient to learn with ϵ and δ prescribed. Then, m_L is a sample complexity:

$$m_L(\epsilon,\delta) = min\{m : m \text{ is a sufficient sample size for } (\epsilon,\delta)\text{-learning } H \text{ by } L\}$$

Similarly, *estimation error* can be defined as the smallest possible estimation error bound $\epsilon_L(m,\delta)$ of L. Also similarly, *sample complexity* $m_H(\epsilon,\delta)$ for H (absolute lower bound on sample size to be sufficient to $(\epsilon,\delta) -$ learn H):

$$m_H(\epsilon,\delta) = \min_L m_L(\epsilon,\delta)$$

Theorem 1. *Suppose h is a function from X to $\{0,1\}$, then:*

$$P^m\{|error_z(h) - error_P(h)| \geq \epsilon\} \leq 2\exp(-2\epsilon^2 m)$$

Proof: The equation above has a form of Hoeffding's Inequality: $P\{|\bar{X} - \mathbb{E}[\bar{X}]| \geq t\} \leq \exp(-2mt^2)$, where $X = \{X_1,\ldots,X_m\}$ is independent random variables bounded by $[0,1]$ and $t>0$, and m is still sample size. In our case $X_i = 1$ on $(x_i,y_i) \in Z$ if and only if $h(x_i) \neq y$. It is easy to see that $error_z(h) = \frac{1}{m}(X_1 + X_2 + \ldots + X_m) = \bar{X}$. Since $error_P(h) = P\{h(x) \neq y\}$, then $\mathbb{E}[\bar{X}] = P\{h(x) \neq y\} = error_P(h)$.

This theorem is not sufficient to prove that L learns. This theorem implies that the true error $error_P(h)$ is approximately minimazing with minimization of the estimation error $error_z(h)$.