Math Camp for Machine Learning

Synferlo

Mar. 22, 2021

1 Statistical Learning

1.1 Elements in ML

1. Instance/example:

x, $x \in X$

2. Instance space/domain:

X (where the instance comes from).

3. Label:

Each instance has a label, or class. Label can be 0/1 or +/-.

4. Concept:

There's a function c, called concept, that tells the **true** relationship between instance and label.

$$\text{concept } c : X \to \{0, 1\}$$

Each instance x is labeled by $c(x)$. Our goal is to find this $c(\cdot)$.

5. Hypothesis:

Note, this is NOT the same one as we say in econometrics. Hypothesis, $h(\cdot)$, is a function that the machine use to do the prediction given an instance x.

$$h : X \to \{0, 1\}$$

6. Concept VS Hypothesis:

Concept is the TRUE relationship between x and label.

Hypothesis is the GUESS of our machine given the training data.

7. Concept class:

C is where concept c comes from, $c \in C$.

8. Distribution:

All instances are generated from a particular distribution D. We call it target distribution or distribution for short.

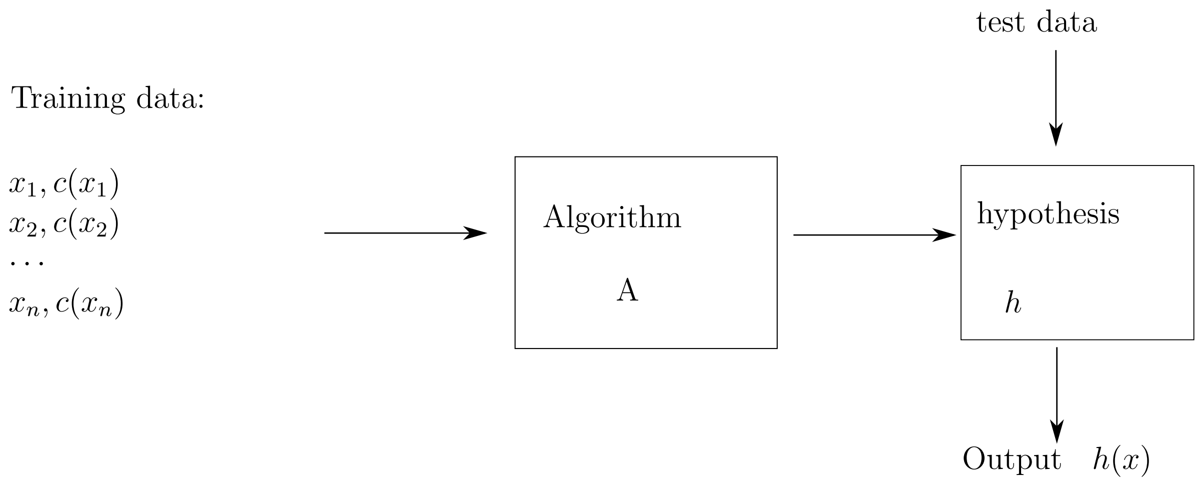
$$x_i \sim D, i.i.d.$$

Hypothesis class:

It tells where the hypothesis comes from. We allow h and c come from different classes.

$$h \in \mathcal{H}$$

1.2 ML Process



$x_i \in X, \quad x_i \sim D, \quad c \in C, \quad h \in \mathcal{H}$

Figure 1: ML process

1.3 PAC Learning

We want to see $h(x) = c(x)$

We DO NOT want to see $h(x) \neq c(x)$

1.3.1 How we measure error:

$$err_D(h) = Pr_{x \sim D}\left[h(x) \neq c(x)\right]$$

We want this,

$$err_D(h) \leq \varepsilon,$$

where ε is a small positive number.

To guarantee the machine work well, we require the following condition,

$$Pr\left(err_D(h) \leq \varepsilon\right) \geq 1 - \delta$$

where δ is a small positive number.

Hence, $err_D \leq \varepsilon$ requires algorithm to be more accurate. $Pr(err \leq \varepsilon) \geq 1 - \delta$ requires the probability of this correction to be high.

This method is called Probability approximately correct, or PAC for short.

==

We say concept space C is PAC-learnable by $\mathcal{H}$,

if there exist an algorithm (alg.) A, $\forall c \in C$, $\forall$ distribution D, $\forall \varepsilon > 0, \delta > 0$,

A takes $m = poly(\frac{1}{\varepsilon}, \frac{1}{\delta}, \cdots)$ random examples $x_i \sim D$,

that it makes output hypothesis $h \in \mathcal{H} \quad s.t.\ Pr(err_D(h) \leq \varepsilon) \geq 1 - \delta$.

NB: m is sample size. The more data we have, the higher accuracy that $h(\cdot)$ will be. Hence, m is negative correlated with ε and δ.

==