Machine Learning I

Supervised learning framework - Optimal predictions

Souhaib Ben Taieb

March 7, 2022

University of Mons

Table of contents

Optimal prediction function

$$\boxed{f = \underset{h:\mathcal{X}\to\mathcal{Y}}{\operatorname{argmin}}\ E_{\text{out}}(h)} \qquad \boxed{g = \underset{h\in\mathcal{H}}{\operatorname{argmin}}\ E_{\text{in}}(h)}$$

Recall that the **optimal prediction function** is given by

$$f = \underset{h:\mathcal{X}\to\mathcal{Y}}{\operatorname{argmin}}\ \underbrace{\mathbb{E}_x\left[E_{\text{out}}(h,x)\right]}_{E_{\text{out}}(h)}, \tag{1}$$

where

$$E_{\text{out}}(h,x) = \mathbb{E}_{y|x}[L(y,h(x))|x].$$

and $L(\cdot,\cdot)$ is the loss function.

It sufficed to minimize the error pointwise, i.e. compute

$$f(x) = \underset{h:\mathcal{X}\to\mathcal{Y}}{\operatorname{argmin}}\ E_{\text{out}}(h,x), \tag{2}$$

for all $x \in \mathcal{X}$.

Table of contents

Optimal predictions in regression (squared error loss)

With the squared error loss function $L(y, \hat{y}) = (y - \hat{y})^2$, the optimal prediction function is given by

$$f(x) = \underset{h:\mathcal{X}\to\mathcal{Y}}{\operatorname{argmin}} \mathbb{E}_{y|x}[(y - h(x))^2|x] \tag{3}$$

$$= \mathbb{E}_{y|x}[y|x], \tag{4}$$

i.e. the conditional expectation, also known as the **regression function**.

In other words, when *best is measured by expected squared error*, the best prediction for y at any point x is the conditional expectation at x.

Optimal predictions in regression (squared error loss)

$$
\begin{aligned}
& E_{\text{out}}(h, x) && (5)\\
& = \mathbb{E}_{y|x}[(y - h(x))^2 | x] && (6)\\
& = \mathbb{E}[y^2 - 2yh(x) + h(x)^2 | x] && (7)\\
& = \mathbb{E}[y^2 | x] - 2h(x)\mathbb{E}[y|x] + h(x)^2 && (8)\\
& = \mathrm{Var}(y|x) + (\mathbb{E}[y|x])^2 - 2h(x)\mathbb{E}[y|x] + h(x)^2 && (9)\\
& = \mathrm{Var}(y|x) + (\mathbb{E}[y|x] - h(x))^2 && (10)
\end{aligned}
$$

- The second term is non-negative, and will be equal to zero if $h(x) = \mathbb{E}[y|x]$.
- The first term corresponds to the inherent unpredictability, or noise, of the output, and is called the **Bayes error**. It is the smallest error any learning algorithm can achieve.

Table of contents

Optimal predictions in regression (zero-one loss)

For a multi-class classification problem with K categories, i.e. $y \in \mathcal{C} = \{C_1, \ldots, C_K\}$ and the zero-one loss function $L(y, \hat{y}) = \mathbb{1}\{y \neq \hat{y}\}$, the optimal prediction function is given by

$$f(x) = \underset{h:\mathcal{X}\to\mathcal{C}}{\operatorname{argmin}} \mathbb{E}_{y|x}[\mathbb{1}\{y \neq h(x)\}|x] \tag{11}$$

$$= \underset{h:\mathcal{X}\to\mathcal{C}}{\operatorname{argmax}} \mathbb{P}(y = h(x)|x). \tag{12}$$

The optimal classifier is called the **Bayes classifier**, which has the following error rate at x:

$$1 - \max_{k=1,\ldots,K} \mathbb{P}(y = C_k|x),$$

also called the **Bayes error rate**, which gives the lowest possible error rate that could be achieved <u>if we knew</u> $\mathbb{P}(y|x)$.

Optimal predictions in regression (zero-one loss)

$$
\begin{aligned}
E_{\text{out}}(h, x) &= \mathbb{E}_{y|x}[\mathbb{1}\{y \neq h(x)\}|x] \\
&= \sum_{k=1}^{K} \mathbb{1}\{C_k \neq h(x)\}\ \mathbb{P}(y = C_k|x) \\
&= \sum_{k:C_k \neq h(x)} 1 \times \mathbb{P}(y = C_k|x) + 0 \times \mathbb{P}(y = h(x)|x) \\
&= \sum_{k:C_k \neq h(x)} \mathbb{P}(y = C_k|x) \\
&= \sum_{k:C_k \neq h(x)} \mathbb{P}(y = C_k|x) + \mathbb{P}(y = h(x)|x) - \mathbb{P}(y = h(x)|x) \\
&= \sum_{k=1}^{K} \mathbb{P}(y = C_k|x) - \mathbb{P}(y = h(x)|x) \\
&= 1 - \mathbb{P}(y = h(x)|x).
\end{aligned}
$$

Optimal predictions in classification

Using the fundamental bridge, we can directly write

$$\begin{aligned}
&\mathbb{E}_{y|x}[\mathbb{1}\{y \neq h(x)\}|x] \\
&= \mathbb{P}(y \neq h(x)|x) \\
&= 1 - \mathbb{P}(y = h(x)|x).
\end{aligned}$$

In conclusion, we have

$$\begin{aligned}
f(x) &= \operatorname*{argmin}_{h:\mathcal{X}\to\mathcal{C}} \mathbb{E}_{y|x}[\mathbb{1}\{y \neq h(x)\}|x] && (13) \\
&= \operatorname*{argmin}_{h:\mathcal{X}\to\mathcal{C}} 1 - \mathbb{P}(y = h(x)|x) && (14) \\
&= \operatorname*{argmax}_{h:\mathcal{X}\to\mathcal{C}} \mathbb{P}(y = h(x)|x). && (15)
\end{aligned}$$

Table of contents

Quantifying the approximation-generalization tradeoff

The difference between the out-of-sample error of g and f can be decomposed as follows

$$E_{\text{out}}(g) - E_{\text{out}}(f) = \underbrace{[E_{\text{out}}(g^*) - E_{\text{out}}(f)]}_{\text{Approximation error}} + \underbrace{[E_{\text{out}}(g) - E_{\text{out}}(g^*)]}_{\text{Estimation error}}$$

Recall that simple models underfit the data and complex models overfit. There is an **approximation-generalization** tradeoff.

The **bias-variance** decomposition allows to quantify the approximation-generalization tradeoff for the **squared error** loss function.

The bias-variance tradeoff

See board.

The bias-variance tradeoff