Machine Learning I

Supervised learning framework - Optimal predictions

Souhaib Ben Taieb

March 10, 2022

University of Mons

Table of contents

Optimal prediction function

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

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

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

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

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

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

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

Table of contents

The bias-variance decomposition

- Previously, we considered the **unrealistic scenario** where we know p_{xy}. As a result, we were able to compute the optimal hypothesis/predictions for different loss functions.
- In practice, we only observe a **dataset** $\mathcal{D}$ where each data point is assumed to be an i.i.d. realization from p_{xy}.
- Overly simple models underfit and complex models overfit. There is an **approximation-generalization** tradeoff:

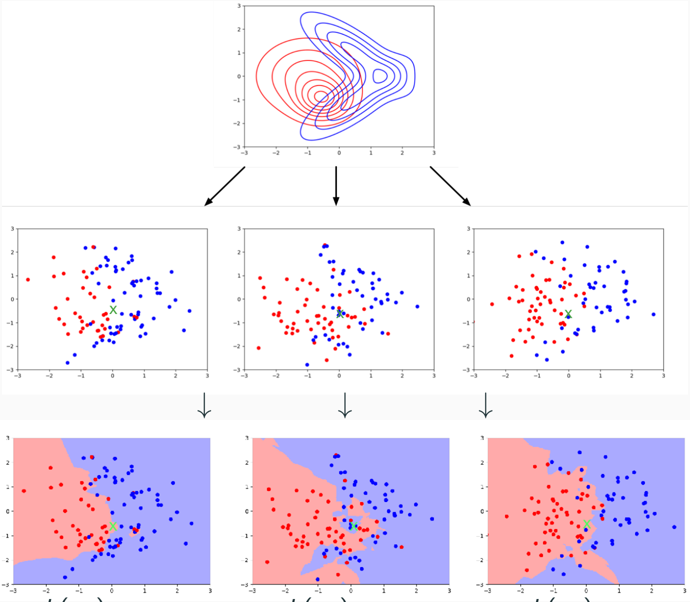
$$E_{\text{out}}(g) - E_{\text{out}}(f) = \underbrace{[E_{\text{out}}(g^*) - E_{\text{out}}(f)]}_{\text{Approximation error}} + \underbrace{[E_{\text{out}}(g) - E_{\text{out}}(g^*)]}_{\text{Estimation error}}$$

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

An experiment

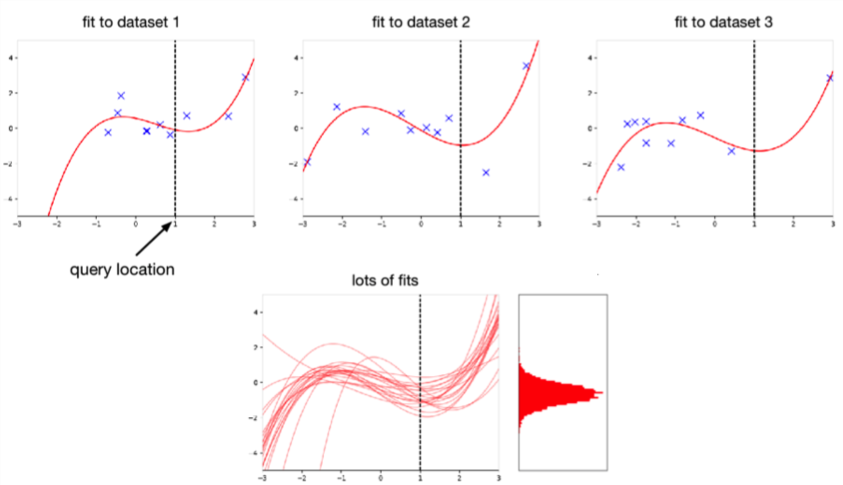
- Consider an experiment where we sample lots of training sets independently from p_{xy}.
- Pick a fixed query point x_*.
- Let's run our learning algorithm on each training set, and compute its prediction $g(x_*)$ at the query point x_*.
- We can view $g(x_*)(= g_{\mathcal{D}}(x_*))$ as a random variable, where the randomness comes from the training set $\mathcal{D}$.

An experiment - Classification



$h(x_*) = \textcolor{red}{\bullet}$ $h(x_*) = \textcolor{blue}{\bullet}$ $h(x_*) = \textcolor{red}{\bullet}$

An experiment - Regression



An experiment (continued)

- The experiment
  - Fix a query point x_*.
  - Repeat:
    - Sample a dataset $\mathcal{D}$ i.i.d. from $p_{x,y}$
    - Run the learning algorithm on $\mathcal{D}$ to obtain g
    - Compute the prediction for x_*, i.e. $g(x_*)$
    - Sample the (true) output y_* from $p_{y|x}(\cdot|x = x_*)$
    - Compute the loss $L(y_*, g(x_*))$
- $L(y_*, g(x_*))$ contains two sources of randomness: $\mathcal{D}$ and y_*. This gives a distribution over the loss at x_*.
- Let us compute

$$\mathbb{E}_{\mathcal{D}}\left[\underbrace{\mathbb{E}_{y|x}\left[L(y, g(x))|x\right]}_{E_{\text{out}}(g,x)}\right]$$

for the squared error loss $L(y, g(x)) = (y - g(x))^2$.

The bias-variance decomposition

Previously, we proved that

$$E_{\text{out}}(g, x) = \mathbb{E}_{y|x}[(y - g(x))^2 | x] = \text{Var}(y|x) + (f(x) - g(x))^2,$$

where $f(x) = \mathbb{E}[y|x]$.

We can write

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}}[E_{\text{out}}(g, x)] \\
&= \text{Var}(y|x) + \mathbb{E}_{\mathcal{D}}[(f(x) - g(x))^2] \\
&= \text{Var}(y|x) + f(x)^2 - 2f(x)\mathbb{E}_{\mathcal{D}}[g(x)] + \mathbb{E}_{\mathcal{D}}[g(x)^2] \\
&= \text{Var}(y|x) + f(x)^2 - 2f(x)\mathbb{E}_{\mathcal{D}}[g(x)] + \text{Var}(g(x)) + \mathbb{E}_{\mathcal{D}}[g(x)]^2 \\
&= \underbrace{\text{Var}(y|x)}_{\text{Bayes error at } x} + \underbrace{(f(x) - \mathbb{E}_{\mathcal{D}}[g(x)])^2}_{\text{Bias at } x} + \underbrace{\text{Var}(g(x))}_{\text{Variance at } x}
\end{aligned}$$

The bias-variance decomposition

$$\mathbb{E}_{\mathcal{D},y|x}[(y-g(x))^2|x] = \underbrace{\mathrm{Var}(y|x)}_{\text{Bayes error at } x} + \underbrace{(f(x) - \mathbb{E}_{\mathcal{D}}[g(x)])^2}_{\text{Bias at } x} + \underbrace{\mathrm{Var}(g(x))}_{\text{Variance at } x}$$

We split the expected error at x into three terms:

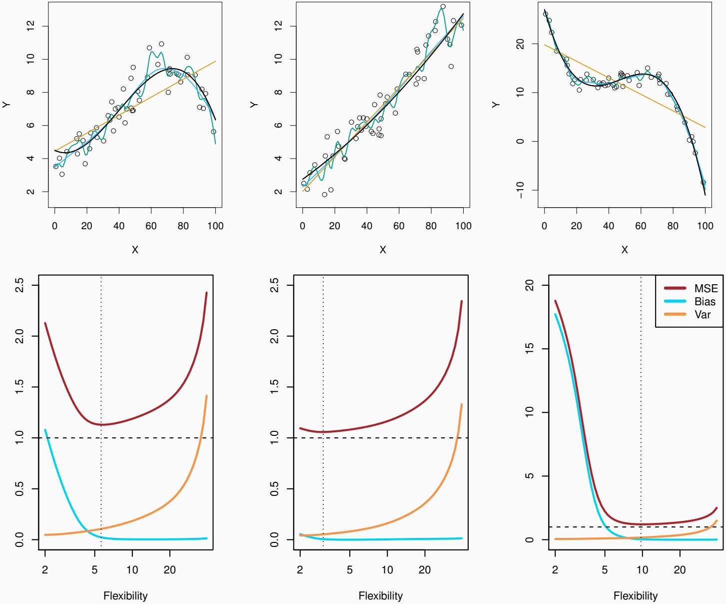
- Bayes error: the inherent unpredictability of the output
- **bias**: how wrong the expected prediction is (underfitting)
- **variance**: the variability of the predictions (overfitting)

If we take the expectation with respect to x, we obtain

$$\mathbb{E}_{\mathcal{D},y,x}[(y-g(x))^2] = \underbrace{\mathrm{Var}(y)}_{\text{Bayes error}} + \underbrace{\mathbb{E}_x[(f(x) - \mathbb{E}_{\mathcal{D}}[g(x)])^2]}_{\text{Bias}} + \underbrace{\mathbb{E}_x[\mathrm{Var}(g(x))]}_{\text{Variance}}$$

While the analysis only applies to squared error, we often use "bias"/'variance" as synonyms for "underfitting"/"overfitting".

The bias-variance tradeoff



The bias-variance tradeoff

Throwing darts = predictions for each draw of a dataset