

## PRML Review: Bias-Variance Tradeoffs.

2021. 12. 4.

1° For regression problems:

t: the target variables.

$$D = \{(x_1, t_1), (x_2, t_2), \dots, (x_n, t_n)\}$$

Cost Function:

(Recall: The expectation of a joint distribution:

$$\text{The average/expected Loss: } \mathbb{E}[L] = \iint L(t, y_{\text{obs}}) p\{x, t\} dx dt \quad \mathbb{E}[g(x, y)] = \iint g(x, y) f_{x,y}(x, y) dx dy$$

$\uparrow$   
The prediction of the model.

L is the Loss function.

i.e. the squared loss function:  $L(t, h(x)) = \{h(x) - t\}^2$

Our goal is to choose the best  $h(x)$  that minimize the Loss

$$\therefore \frac{\partial \mathbb{E}[L]}{\partial y_{\text{obs}}} = -2 \int \{h(x) - t\} p\{x, t\} dt = 0 \quad (1)$$

$$\Rightarrow h^*(x) = \frac{\int t p\{x, t\}}{p(x)} = \int t p(t|x) dt = \mathbb{E}_t[t|x] \Leftarrow \text{the best estimation is the conditional average of } t \text{ on } x.$$

let  $h(x)$  denotes the best estimation from the training data, which is:

$$h(x) = \int t p(t|x) dt = \mathbb{E}[t|x]$$

$$\begin{aligned} \Rightarrow \mathbb{E}[L] &= \iint \{y(x) - t\}^2 p\{x, t\} dx dt \quad [(y(x) - h(x) + h(x) - t)^2 = [y(x) - h(x)]^2 + 2[y(x) - h(x)][h(x) - t] + [h(x) - t]^2] \\ &= \iint \{y(x) - h(x) + h(x) - t\}^2 p\{x, t\} dx dt \\ &= \iint [y(x) - h(x)]^2 p\{x, t\} dx dt + 2 \iint [y(x) - h(x)] [h(x) - t] p(x, t) dx dt + \iint [h(x) - t]^2 p(x, t) dx dt \\ &= \int [y(x) - h(x)]^2 p(x) dx + 2 \int [y(x) - h(x)] \underbrace{\int [h(x) - t] p(x, t) dt}_{\text{By equation (1): } h(x) \text{ is chosen to set } \int [h(x) - t] p(x, t) dt \text{ to zero.}} dx + \int [h(x) - t]^2 p(x) dx \\ &= \int [y(x) - h(x)]^2 p(x) dx + \int [h(x) - t]^2 p(x) dx \end{aligned}$$

$$= \int \{y(x) - \hat{E}[\hat{t}(x)]\} p(x) dx + \int \{\hat{E}[\hat{t}(x)] - t\}^2 p(x) dx. \quad (2)$$

**Remark:**

$y(x)$ : If we draw  $n$  inputs from the target distribution, forming a training data  $D$ . Based on the data  $D$ , we want to train an algorithm  $A$  to learn a hypothesis. The process is  $y(x; D) = A(D)$ .

$h(x) = \int p(t|x) dt$  could be viewed as the intrinsic distribution of the target distribution that you want to model.

The second term in (2):

$\int \{h(x) - t\}^2 p(x) dx$  is the intrinsic noise in the data.

If we have unlimited supply of the data, we could have  $y(x) = h(x) = \hat{E}[\hat{t}(x)]$

$\Rightarrow \hat{E}[L] = \int \{h(x) - t\}^2 p(x) dx \Rightarrow$  the loss of our model is just the noise of the data.

However, in reality, we can only derive the algorithm  $A$  from a finite data set  $D$ :  $y(x; D) = A(D)$ .

$$\Rightarrow \hat{E}[L] = \int \{y(x; D) - \hat{E}[\hat{t}(x)]\}^2 p(x) dx + \int \{\hat{E}[\hat{t}(x)] - t\}^2 p(x, t) dx dt.$$

For the first term:

$$\begin{aligned} & \because \{y(x; D) - \hat{E}[\hat{t}(x)]\}^2 \\ &= \{y(x; D) - \hat{E}_D[y(x; D)] + \hat{E}_D[y(x; D)] - \hat{E}[\hat{t}(x)]\}^2 \\ &= \{y(x; D) - \hat{E}_D[y(x; D)]\}^2 + \{\hat{E}_D[y(x; D)] - \hat{E}[\hat{t}(x)]\}^2 + 2\{y(x; D) - \hat{E}_D[y(x; D)]\} \{ \hat{E}_D[y(x; D)] - \hat{E}[\hat{t}(x)] \} \\ &\Rightarrow \hat{E}_D[\{y(x; D) - \hat{E}[\hat{t}(x)]\}] = \underbrace{\{\hat{E}_D[y(x; D)] - h(x)\}^2}_{\text{Bias}} + \underbrace{\hat{E}_D[\{y(x; D) - \hat{E}_D[y(x; D)]\}^2]}_{\text{Variance}} \end{aligned}$$

**Remark:**

$\hat{E}_D[y(x; D)] = \int y(x; D) P_D(D) dD$ :  $\hat{E}_D[y(x; D)]$  can be viewed as the expected classifier:  $P_D(D)$  is the probability of drawing

$D$  from the target distribution  $P$ .

Bias:  $E_D[y(x; D)] - h(x)$  : the inherent error that one obtains from the model even with infinite training data  
(Often arises when you select the inappropriate model. i.e. model nonlinear models with linear model).

Variance:  $E_D[\{y(x; D) - E_D[y(x; D)]\}^2]$  : Captures how much your model changes if training on different data set.  
(If we have the best model from the training data, how far are we from the average model?)

Combine these terms into equation (2):

$$E[L] = \int \{E_D[y(x; D)] - h(x)\}^2 p(x) dx \rightarrow \text{Bias}^2 + \int E_D[\{y(x; D) - E_D[y(x; D)]\}^2] p(x) dx \rightarrow \text{Variance} + \int \{E[h(x)] - f\}^2 p(x, \epsilon) dx \rightarrow \text{noise.}$$

For classification problems:

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} \quad x_i \in X \quad y_i \in Y, \quad X \text{ and } Y \text{ are two countable spaces.}$$

$y_i \in \{c_1, \dots, c_k\} \leftarrow \text{classification problems.}$

The target  $f: f = P\{Y_F = y_F | x\}$

one-hot:  $P\{Y_F = y_F | x\} = 1$  for one value of  $y_F$ , and 0 for all others.

The hypothesis  $h: c$  generated from a learning algorithm

$$h = P\{Y_H = y_H | x\}$$

Similarly:  $P\{Y_H = y_H | x\} = 1$  for one value of  $y_H$ , and 0 for all others.

Cost Function,

$$\mathbb{E}[L] = \sum_{y_H, y_F} l(y_H, y_F) P\{y_H, y_F\}$$

We use zero-one loss.

$$\Rightarrow l(y_F, y_H) = 1 - \delta(y_F, y_H)$$

$$\delta(y_F, y_H) = \begin{cases} 1 & \text{if } y_F = y_H \\ 0 & \text{otherwise.} \end{cases}$$

$$\Rightarrow \mathbb{E}[L] = \sum_{y_H, y_F} [1 - \delta(y_H, y_F)] P\{y_H, y_F\}$$

$$= 1 - \underbrace{\sum_{y_F, y_H} \delta(y_H, y_F) P\{y_H, y_F\}}$$

The misclassification rate

$$\Rightarrow L = 1 - \sum_{y \in Y} P\{Y_H = y_F = y | x\}$$

$$= \sum_{y \in Y} -P\{Y_H = y_F = y | x\} + \sum_{y \in Y} P\{Y_H = y\} P\{Y_F = y | x\} + \sum_{y \in Y} [-P\{Y_H = y\} P\{Y_F = y | x\} + \frac{1}{2} P\{Y_F = y | x\}^2]$$

$$+ \left[ \frac{1}{2} - \sum_{y \in Y} P\{Y_H = y | x\}^2 \right] + \left[ \frac{1}{2} - \sum_{y \in Y} P\{Y_F = y | x\}^2 \right]$$

$$= \sum_{y \in Y} \left[ P\{Y_H = y | x\} P\{Y_F = y | x\} - P\{Y_F = y | x\} \right] \quad \text{covariance}$$

$$+ \sum_{y \in Y} \left[ P\{Y_F = y | x\} - P\{Y_H = y | x\} \right]^2 \quad \text{bias}^2$$

$$+ \frac{1}{2} \left[ 1 - \sum_{y \in Y} P\{Y_H = y | x\}^2 \right] \quad \text{variance}$$

$$+ \frac{1}{2} \left[ 1 - \sum_{y \in Y} P\{Y_F = y | x\}^2 \right] \quad \text{noise.}$$

Proposition:  $Y_F$  and  $Y_H$  are conditionally independent given  $f$  and test points  $x$

Sketch of proof:  $\because Y_F$  depends only on the target  $f$  and the test point  $x$

$$\begin{aligned} P\{Y_F, Y_H | f, x\} &= P\{Y_F | Y_H, f, x\} P\{Y_H | f, x\} \\ &= P\{Y_F | f, x\} P\{Y_H | f, x\} \end{aligned}$$

$\Rightarrow$  The covariance term.

$$\sum_{y \in Y} \left[ P\{Y_H = y | x\} P\{Y_F = y | x\} - P\{Y_F = y | x\} \right] = 0$$

$$\Rightarrow E[L] = \sum_{x \in X} P(x) L(x)$$

$$= \sum_{x \in X} P(x) \left\{ \frac{1}{2} \sum_{y \in Y} \left[ P\{Y_F = y | x\} - P\{Y_H = y | x\} \right]^2 \right\} \Rightarrow \text{bias}_x^2$$

$$+ \frac{1}{2} \left[ 1 - \sum_{y \in Y} P\{Y_H = y | x\}^2 \right] \Rightarrow \text{variance}_x$$

$$+ \frac{1}{2} \left[ 1 - \sum_{y \in Y} P\{Y_F = y | x\}^2 \right] \} \Rightarrow \text{noise}_x$$

Remark:

$P\{Y_F=y|x\}$  : the intrinsic distribution of the target data

$P\{Y_H=y|x\}$ : The full expression is,

$$P\{Y_H=y|f, m, x\}$$

$$= \sum_d P\{d|f, m, x\} P\{Y_H=y|d, f, m, x\}$$

$$= \sum_d \underbrace{P\{d|f, m\}}_{\text{generate training set } d} \underbrace{P\{Y_H=y|d, x\}}_{\text{to make a guess to the } Y \text{ value.}}$$

$\Rightarrow$  the learning algorithm trained on data  $d$   
from the target distribution  $f$