Regression

Estimating the Combat Power (cp) of a pokemon after evolution.

$$\hat{y} = b + \sum w_i x_i$$

w_i and b are parameters

w_i → weight, b → bias

Loss function L:

Input: a function, output: how bad it is

$$L(f) = L(w, b)$$

Common Loss function

pay attention to the difference between cost fuction and Loss function.

① zero-one loss

$$L(y, f(x)) = \begin{cases} 1, & y \neq f(x) \\ 0, & y = f(x) \end{cases}$$

variant

$$L(y, f(x)) = \begin{cases} 1, & |y - f(x)| \geq T \\ 0, & |y - f(x)| < T \end{cases}$$

② absolute loss

$$L(y, f(x)) = |y - f(x)|$$

③ quadratic loss

$$L(y, f(x)) = (y - f(x))^2$$

④ Logarithmic loss

$$L(y, f(x)) = -\log p(y|x)$$

Here use the idea of maximum Likelihood. $p(y|x)$ represent the probability of predict tuple x correctly. Since it is a loss function, the better the model, the lower the function value will be. So we put a "−" in front of the function.

⑤ Hinge loss

$$L(w, b) = \max\{0, 1 - yf(x)\}$$

mostly used in classification, especially in SVM.

⑥ exponential loss $L(y, f(x)) = e^{-yf(x)}$

very sensitive to outliers and noise. mostly in AdaBoost.

Now, back to the lecture.

Suppose we have 10 points (10 pokemon), let the loss function be

$L(w,b) = \sum_{n=1}^{10} (\hat{y}^n - (b + w \cdot x_{cp}^n))^2$, where x_{cp}^n represents the cp value of the nth pokemon.

We want to find the best model where $f^* = \arg\min_f L(f)$

That is $w^*, b^* = \arg\min_{w,b} L(w,b)$

$$= \arg\min_{w,b} \sum_{n=1}^{10} (\hat{y}^n - (b + w \cdot x_{cp}^n))^2$$

Here, using Linear algebra, we can easily solve the problem.

$\beta_0 = b \quad \beta_1 = w. \qquad \hat{\beta} = (X'X)^{-1}X'Y = \begin{bmatrix} b \\ w \end{bmatrix}$

more at MH8131 — chapter 5 — 4 Estimation of Regression Coefficients.

Another way is <u>Gradient Descent</u>

1. consider $L(w)$ with 1 parameter w: $\Rightarrow w^* = \arg\min_w L(w)$

① (Randomly) Pick initial w^0

② moving $-\eta \frac{dL}{dw}\Big|_{w=w^0} \Rightarrow w^1 = w^0 - \eta \frac{dL}{dw}\Big|_{w=w^0}$

η → Learning rate.

⋮ iteration

⑥ reach local / global minimum. → No need to worry. In linear regression, the loss function L is convex. Only reach global optimal.

2. 2 parameters $w^*, b^* = \arg\min_{w,b} L(w,b)$

① initial w^0, b^0

② $w^1 = w^0 - \eta \frac{\partial L}{\partial w}\Big|_{w=w^0, b=b^0}$, $b^1 = b^0 - \eta \frac{\partial L}{\partial b}\Big|_{w=w^0, b=b^0}$

⋮

③ optimal

Regularization

$$y = b + \sum w_i x_i$$

No need to concer bias in the regularization.

$$L = \sum_n \left(\hat{y}^n - (b + \sum w_i x_i) \right)^2 + \underbrace{\lambda \sum (w_i)^2}_{\text{regularization term}}$$

to find function with smaller W_i

smooth function.

less sensitive to $\underbrace{\text{changes in } x_i}_{\text{noise}}$