



Discriminative Models

Linear Model

$$f(x) = w_1 x_1 + w_2 x_2 + \dots + w_n x_n \rightarrow y$$

$$y = w_1 x_1 + w_2 x_2 + \dots + w_n x_n$$

(price) $\hat{y} = w^T x$ ← OLS

Linear Regression

Loss → MSE → $\min \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2$

$$w^*_{(optimal)} = (X^T X)^{-1} X^T y$$

$$\hat{y} = w^T x$$

MLE (estimate $\theta(w)$)

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} L(\theta) \rightarrow \text{Loss}$$

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}} P(D|\theta)$$

Assumption: IID

$$P(D|\theta) = \prod_{i=1}^N P(y_i | x_i, \theta)$$

$$= \prod_{i=1}^N \log P(y_i | x_i, \theta)$$

$$\hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}} \sum_{n=1}^N \log P(y_n | x_n, \theta)$$

$$NLL = -\underset{\theta}{\operatorname{argmin}} \sum_{n=1}^N \log P(y_n | x_n, \theta)$$

min(NLL)

→ MAP

MLE → Bayes Theorem

$$P(Y|X) = \frac{P(X|Y) P(Y)}{\text{evidence}}$$

MLE for Bernoulli

(PDF) $P(y_n | \theta) = \theta^{y_n} (1-\theta)^{1-y_n}$

$$NLL = -\log \sum_{n=1}^N \theta^{y_n} (1-\theta)^{1-y_n}$$

$$= -N_1 \log \theta - N_0 \log(1-\theta)$$

$$N = N_1 + N_0$$

$$\frac{d(NLL)}{d\theta} = -\frac{N_1}{\theta} + \frac{N_0}{1-\theta} \xrightarrow{\text{set } 0}$$

$$\hat{\theta}_{MLE} = \frac{N_1}{N_0 + N_1}$$

MLE for Normal distribution

$$NLL(\mu, \sigma^2)$$

$$\frac{\partial(NLL)}{\partial \mu} \xrightarrow{\text{set } 0}$$

$$\frac{\partial(NLL)}{\partial(\sigma^2)} \xrightarrow{\text{set } 0} \text{solve for } \sigma^2$$

Logistic Regression → classification

$$\hat{y} = w^T x$$

$$\hat{y} = \sigma(w^T x)$$

$$\sigma(a) = \frac{1}{1+e^{-a}}$$

$$\sigma'(a) = \sigma(a) \cdot (1 - \sigma(a))$$

$$\sigma(-a) = 1 - \sigma(a)$$

MLE

$$P(y|x, \theta) = \text{Ber}(y | \sigma(\underline{w}^T x))$$

$$NLL = -\frac{1}{N} \sum_{i=1}^N \log [\mu_n^{y_n} (1-\mu_n)^{(1-y_n)}]$$

$$\text{where } \mu_n = \sigma(\underline{w}^T x), \theta$$

$$= -\frac{1}{N} (y_n \log \mu_n + (1-y_n) \log(1-\mu_n))$$

(cross-entropy)

min(NLL)

Loss function → optimization problem

↓ solve
Gradient Descent

