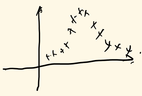
Logistics Regression

Monday, 19 March 2018 08:33

underfitting / overfitting.

Parametric learning algorithm / non parametric learning.

Locally weight regression: (Loess)



LWR: Fit θ to minimize

$$\sum_i w^{(i)} \left(y^{(i)} - \theta^T x^{(i)}\right)^2$$

where $w^{(i)} = \exp\left(-\frac{(x^{(i)} - x)^2}{2\tau^2}\right)$

τ: bandwidth parmetric.

Short coming: training everytime when querying.

* Fix: Andrew Moore : KD Tree.

Probabilistic Interpretation:

Assumption: $y^{(i)} = \theta^T x^{(i)} + \varepsilon^{(i)}$

$\varepsilon^{(i)}$ = error, $\varepsilon^{(i)} \sim N(0, \sigma^2)$

$$P(\varepsilon^{(i)}) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\varepsilon^{(i)})^2}{2\sigma^2}\right)$$

$$P(y^{(i)} \mid x^{(i)}; \theta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}\right)$$

$$\Rightarrow y^{(i)} \mid x^{(i)}; \theta \sim N(\theta^T x^{(i)}, \sigma^2)$$

CLT permit erro to be normal distribution.

assume: $\varepsilon^{(i)}$ are IID (Independently and Identically Distibute)

$L(\theta) = p(\vec{y} \mid X; \theta)$: likelihood.

$$= \prod_{i=1}^{m} P(y^{(i)} \mid x^{(i)}; \theta)$$

$$= \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}\right)$$

Maximum likelihood:

choose θ to maximize $L(\theta)$

$$l(\theta) = \log L(\theta) = \sum_{i=1}^{m} \log\left[\frac{1}{\sqrt{2\pi}\sigma} \exp(\dots)\right]$$

$$= m \log \frac{1}{\sqrt{2\pi}\sigma} - \sum_{i=1}^{m} \frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}$$

to maximize $l(\theta)$, we need minimize $\sum_{i=1}^{m} \frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}$

Variance σ will take no effect on θ.

Classification:

Linear regression works not very well on classification.

Choose: $h_\theta(x) \in [0, 1]$

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}} \quad // \; g(x) = \frac{1}{1 + e^{-x}}$$

logistic / Sigmoid function.

$P(y=1 \mid x; \theta) = h_\theta(x)$

$P(y=0 \mid x; \theta) = 1 - h_\theta(x)$

$P(y \mid x; \theta) = h_\theta(x)^y (1 - h_\theta(x))^{1-y}$ more elegant way

$$L(\theta) = P(\vec{y} \mid X; \theta) = \prod_{i} h_\theta(x^{(i)})^{y^{(i)}} (1 - h_\theta(x))^{1 - y^{(i)}}$$

$$\max l(\theta) = \sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log(1 - h_\theta(x^{(i)}))$$

// Gradient assent

$$\theta := \theta + \alpha \nabla_\theta l(\theta)$$

$$\frac{\partial}{\partial \theta_j} = \sum_{i=1}^{m} (y^{(i)} - h_\theta(x^{(i)})) x_j^{(i)}$$

$$\theta_j := \theta_j + \alpha \sum_{i=1}^{m} (y^{(i)} - h_\theta(x^{(i)})) x_j^{(i)}$$

$ Preception Algorithm:

$$g(z) = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & \text{else} \end{cases}$$

$h_\theta(x) = f(\theta^T x)$

$$\Rightarrow \theta_j := \theta_j + \alpha (y^{(i)} - h_\theta(x^{(i)})) \cdot x_j^{(i)}$$