1 Linear regression

乘以$\frac{1}{2}$是为了后续计算导数时能够刚好抵消。

$$h(x) = \sum_{i=0}^{n} \theta x_i = \boldsymbol{\theta^T x}$$

$$J(\theta) = \frac{1}{2} \sum_{i=1}^{n} (h_\theta(x^{(i)}) - y^{(i)})^2$$

目的是找到一个θ使得$J(\theta)$的值最小。

1.1 LMS algorithm

梯度下降算法

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$$

α为学习率

For these reasons, particularly when the training set is large, stochastic gradient descent is often preferred over batch gradient descent.

1.2 The normal equations

$$\nabla_\theta J(\theta) = X^T X\theta - X^T \boldsymbol{y}$$

$$\theta = (X^T X)^{-1} X^T \boldsymbol{y}$$

这里假设$X^T X$为可逆矩阵。

1.3 Probabilistic interpretation

$$y^{(i)} = \theta^T x^{(i)} + \epsilon^{(i)}$$

$$p(y^{(i)}|x^{(i)};\theta) = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2})$$

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

找到一个θ使得$L(\theta)$最大。

$$\begin{aligned}
\ell(\theta) &= \log L(\theta) \\
&= \log \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y^{(i)} - \theta^T x^{(i)})^2}{2\sigma^2}) \\
&= n \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{\sigma^2} \cdot \frac{1}{2} \sum_{i=1}^{n} (y^{(i)} - \theta^T x^{(i)})^2
\end{aligned}$$

需要找到一个θ最小化。

$$\frac{1}{2} \sum_{i=1}^{n} (y^{(i)} - \theta^T x^{(i)})^2$$

又得到了相同的结论。

1.4 Locally weighted linear regression

需要找到一个θ最小化。

$$\sum_{i=1}^{n} w^{(i)}(y^{(i)} - \theta^T x^{(i)})^2$$

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

越接近x，w越大。

2 Logistic regression

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

$$g(z) = \frac{1}{1 + e^{-z}}$$

2.1 Multi-class classification