Linear Regression in Math

Linear Regression Expression

Expression 1(for single sample)

$$\hat{y} = \omega_1 x_1 + \omega_2 x_2 + \ldots + \omega_n x_n + b$$

Expression 2(for single sample)

$$\hat{y} = w^T x_n + b$$

Expression 3(for multiple samples)

$$\hat{y} = Xw + b$$
$$\hat{y} = Xw(X \; include \; b)$$

Analytical solution and Loss function

Math pipeline → Linear Regression Expression → Loss Function minimization → Gradient Descent

Question 1:

why MSE could be error function?

Answer 1: Perspertive of Normal Distribution

Normal Distribution

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Likelihood:

$$p(y \mid x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(y - w^T x - b\right)^2}{2\sigma^2}\right)$$

Maximum Likelihood:

$$p(y \mid X) = \prod_{i=1}^{n} p\left(y^i \mid x^i\right)$$

Maximum (-log)Likelihood:

$$-\log p(y \mid X) = \sum_{i=1}^{n} \frac{1}{2} \log\left(2\pi\sigma^2\right) + \frac{1}{2\sigma^2}\left(y - w^T x - b\right)^2$$

Now we can see that linear gression are about last part:

$$\frac{1}{2\sigma^2}\left(y - w^T x - b\right)^2$$

Looks as MSE

Question 2:

How to compute anlytical solution? And what's the cost?

Let loss function's partial derivative of w = 0

$$\nabla_w \|y - Xw\|^2 = 0$$

we get:

$$w^{*} = \left(X^T X\right)^{-1} X^T y$$

Vectorization for speed up

Use 'tensor' and 'numpy' for patch operation and execution
Do not use 'for' loop

Minibatch stochastic gradient descent

$$(w, b) \leftarrow (w, b) - \frac{\eta}{|\mathfrak{B}|} \sum_{i \in \mathfrak{B}} \partial_{w,b} l^i(w, b)$$

$$w \leftarrow w - \frac{\eta}{|\mathfrak{B}|} \sum_{i \in \mathfrak{B}} x^i \left(w^T x^i + b - y^i\right)$$

$$b \leftarrow b - \frac{\eta}{|\mathfrak{B}|} \sum_{i \in \mathfrak{B}} \left(w^T x^i + b - y^i\right)$$