Machine Learning - Notes

François Lux

22 August, 2017

Contents

Machine Learning - Notes

1) Logistic Regression - Classification

Given input x, find the Probability that y = 1, where 0 and 1 denote the two possibilities of a binary classification problem.

Logistic Regression is still a linear classifier (Decision boundary is linear).
Linear: $z = wx + b$

Linear Regression references Gaussian distribution while Logistic regression references Binomial distribution.

So as we want a probability between 0 and 1 we use sigmoid-function.

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

Loss-function (for one example):

$$L(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$$

But optimization problem becomes non-convex (many local optima). So this causes problems with Gradient Descent.

Instead we use:

$$L(\hat{y}, y) = -(y\ log\hat{y} + (1 - y)\ log(1 - \hat{y}))$$

First case: y = 1: $L(\hat{y}, y) = -\ log\hat{y}$ wants a large $\hat{y}$ in order to minimize loss-function.

Second case: y = 0: $L(\hat{y}, y) = -log(1 - \hat{y}))$ wants a small $\hat{y}$ in order to minimize loss-function.

Cost-function(for all test-data):

$$J(w, b) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y}^i, y^i)$$

Gradient Descent:

Initialize w and b, most often with 0 or random (does not matter too much because Loss-function is convex).

Repeat {
$w = w - \alpha\frac{dJ(w,b)}{dw}$
$b = b - \alpha\frac{dJ(w,b)}{db}$
}

If gradient is negative then update is positive, if gradient is positive the update is negative. Alpha is the learning rate.

Backpropagation: Using the chainrule to calculate the derivative of the input-parameters with respect to the final output (J or L).