Optimizers and Activation Functions

February 2022

1 Optimizers

1.1 Momentum

$$v_t = \beta v_{t-1} + \eta \nabla J(\theta)$$
$$\theta_t = \theta_{t-1} - v_t$$

Here η and β are two hyperparameters, controlling the learning rate and the amount of accumulated gradients, respectively. The idea is to use a fraction of the accumulated gradient and add it to the current gradient so that loss could be descented more smoothly.

1.2 A better type of momentum: Nesterov Accelerated Gradient(NAG)

The momentum function first computes first computes the gradient at the current direction, then takes a huge step in the direction of the accumulated gradient.
The momentum method that Nesterov proposed is to first take the huge step in the direction of the accumulated gradient, then make a correction.
The NAG is defined as:

$$v_t = \beta v_{t-1} + \eta \nabla J(\theta - \beta v_{t-1})$$
$$\theta_t = \theta_{t-1} - v_t$$

1.3 Root Mean Square Propagation(RMSProp)

RMSProp is a optimization method introduced by Geoff Hinton: The issue is that Adagrad accumulates the squares of the gradient into a state vector. As a result the accumulated gradient keeps on growing without bound due to the lack of normalization. So RMSprop normalizes the accumulated squares of the gradients.

$$s_t = s_{t-1} + \gamma \nabla J(\theta)^2$$

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{s_t + \epsilon}}$$

1.4 Combining RMSProp and Momentum: Adam

The Adam optimizer combines RMSProp and Momentum, first it computes the accumulated gradients

$$v_t = \beta_1 v_{t-1} + (1 - \beta_1)\nabla J(\theta)$$

Then it computes the accumulated square of the gradients:

$$s_t = \beta_2 v_{t-1} + (1 - \beta_2)J(\theta)$$

Then corrects the bias by dividing v_t and s_t by $1 - \beta_1$ and $1 - \beta_2$ respectively

$$\hat{v}_t = \frac{v_t}{1 - \beta_1}$$

$$\hat{s}_t = \frac{s_t}{1 - \beta_2}$$

At last, Adam updates its parameters

$$\theta_t = \theta_{t-1} - \frac{\hat{v}_t}{\sqrt{\hat{s}_t + \epsilon}}$$

2 Activation Functions

2.1 Sigmoid

The sigmoid function usually refers the function define below:

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

The sigmoid σ function transforms the original input into a real number between 0 and 1, it is usually used to calculate a probability. Its derivative is given by $\frac{d}{dx}\sigma(x) = \sigma(x)(1 - \sigma(x))$

2.2 Rectfied Linear Unit(ReLU)

The ReLU activation function is defined as below:

$$ReLU(x) = \max(0, x)$$

The ReLU function introduces non-linearity to the model. However, the ReLU function is not differentiable at $x = 0$. To find the gradient of the ReLU function, we use the subgradient of the ReLU function, which is any real number between zero and one as the gradient at $x = 0$.