Gradient Descent

February 26, 2021

Abstract

In the Neural Network, we try to find one point that can minimize the loss function. During the process of finding the local minimization, we use gradient descent techniques. The negative gradient descent $-\nabla f$ follow the direction of steepest descent.

1 Why gradient descent

Any function can be estimated by Taylor Series:
$f(x) = f(a) + \frac{f^{'}(a)}{1}(x-a) + \frac{f^{''}(a)}{2!}(x-a)^2 + \frac{f^{'''}(a)}{3!}(x-a)^3 + \dots$
Then we can use root finding to find local minimization. Gradient descent can help us find the root.

2 Gradient of Multivariable function

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$

3 Update Rule

$\boldsymbol{X}^{k+1} = \boldsymbol{X}^k - \eta \nabla f$
Note:$\eta \nabla f$ follow the direction of steepest descent(if $\nabla f$is positive,$-\eta \nabla f$ is negative, if ∇f is negative,$-\eta \nabla f$ is positive), which update $\boldsymbol{X}^{k+1}$ to local minimization.
We want to update $\boldsymbol{X}^k$ to $\boldsymbol{X}^{k+1}$ such that $f(\boldsymbol{X}^{k+1}) < f(\boldsymbol{X}^k)$

4 Example

In order to verify this we can use one example, given $f(x) = x_1^2 + x_1x_2$, we can write f in Taylor Series

$f(x_1, x_2) \simeq f(a, b) + f^{'}_{x_1}(a, b) * (x_1 - a) + f^{'}_{x_2}(a, b) * (x_2 - b)$ (assume remove higher order terms)

$f(x_1, x_2) - f(a, b) = \left[\begin{array}{c} (x_1 - a), (x_2 - b) \end{array}\right] \left[\begin{array}{c} f^{'}_{x_1}(a, b) \\ f^{'}_{x_1}(a, b) \end{array}\right]$

Let $\Delta x_1 = x_1 - a = -f^{'}_{x_1}$ and $\Delta x_2 = x_2 - b = -f^{'}_{x_2}$ ($-\nabla$ f follow the steepest gradient descent)

$f(x_1, x_2) - f(a, b) = \left[\begin{array}{c} -f^{'}_{x_1}, -f^{'}_{x_2} \end{array}\right] \left[\begin{array}{c} f^{'}_{x_1} \\ f^{'}_{x_2} \end{array}\right] = -({f^{'}_{x_1}}^2 + {f^{'}_{x_2}}^2) < 0$

so that the update rule can make sure find the local minimum.