Optimization

Friday, 30 April 2021 07:42

Gradient Descent

- In general used to find a minimum of any function
- In ML mostly used to find a minimum of the error function (typically MSE) dependent on model parameters

General formulation: $\min_{\vec{x} = (x_1, \dots, x_n)} f(x_1, \dots, x_n)$

ML formulation (special case of the above): $\min_{\vec{\theta} = (\theta_1, \dots, \theta_n)} \sum_{(\vec{x}, y) \in D} (y - f(\vec{x} \mid \vec{\theta}))^2$

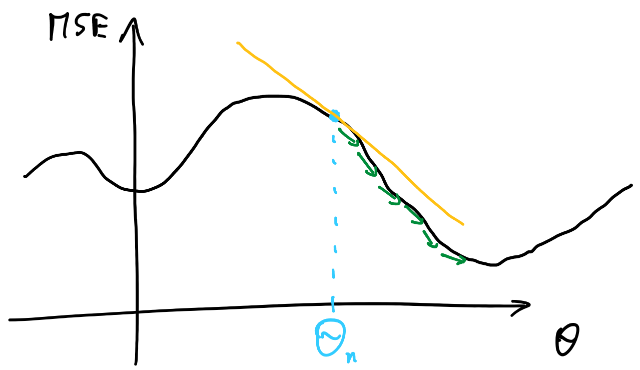
training data

Example

$$\min_{(\theta_0, \theta_1)} \sum_{(x, y) \in D} (y - (\theta_1 x + \theta_0))^2$$

Idea of GD

1. Start with any $\vec{\theta}$
2. Iteratively move $\vec{\theta}$ in the direction opposite to the derivative



The slope of the tangent line is equal to the derivative of the function with respect to θ

$$\frac{\partial MSE}{\partial \theta}$$

$$\frac{\partial MSE}{\partial \theta}$$

Derivative calculation for MSE

For simplicity we assume a linear model

$$f(x \mid \theta_0, \theta_1) = f(x, \theta_0, \theta_1) = \theta_1 x + \theta_0$$

$$MSE = \frac{1}{|D|} \sum_{(x,y) \in D} (y - (\theta_1 x + \theta_0))^2$$

Since $(f+g)'(x) = f'(x) + g'(x)$ and $(a f(x))' = a f'(x)$ we can make the derivative calculations for a single element in the sum (single datapoint) and then average.

$$\frac{\partial}{\partial \theta_0} (y - (\theta_1 x + \theta_0))^2 = 2(y - (\theta_1 x + \theta_0)) \cdot \frac{\partial}{\partial \theta_0} (y - (\theta_1 x + \theta_0))$$

$$= 2(y - (\theta_1 x + \theta_0))(-1)$$

$$= -2(y - (\theta_1 x + \theta_0))$$

$$\frac{\partial}{\partial \theta_1} (y - (\theta_1 x + \theta_0))^2 = 2(y - (\theta_1 x + \theta_0)) \cdot \frac{\partial}{\partial \theta_1} (y - (\theta_1 x + \theta_0))$$

$$= 2(y - (\theta_1 x + \theta_0))(-x)$$

$$= -2x(y - (\theta_1 x + \theta_0))$$

Updating rule

$$\boxed{\vec{\theta}^{\,n} = \vec{\theta}^{\,n-1} - \alpha \frac{\partial}{\partial \theta} MSE(\vec{\theta})}$$

$$\theta_0^n = \theta_0^{n-1} - \alpha \frac{\partial}{\partial \theta_0} MSE(\theta_0^{n-1}, \theta_1^{n-1})$$
$$= \theta_0^{n-1} - \alpha \frac{1}{|D|} \sum_{(x,y)\in D} (-2)(y - (\theta_1 x + \theta_0))$$

$$\theta_1^n = \theta_1^{n-1} - \alpha \frac{\partial}{\partial \theta_1} MSE(\theta_0^{n-1}, \theta_1^{n-1})$$
$$= \theta_1^{n-1} - \alpha \frac{1}{|D|} \sum_{(x,y)\in D} (-2x)(y - (\theta_1 x + \theta_0))$$

Stochastic Gradient Descent (SGD)

1. Start with any $\vec{\theta}$
2. Iteratively:
   a) take a datapoint $(\vec{x}, y) \in D$
   b) calculate the derivative of MSE on this single datapoint with respect to $\vec{\theta}$
   c) shift θ in the direction opposite to the derivative

Problem: SGD can be unstable and diverge

Batch Gradient Descent

Remark: most of the time when people say SGD they mean Batch GD

1. Start with any $\vec{\theta}$
2. Iteratively:
   a) take a datapoint $(\vec{x}, y) \in D$
   b) calculate the derivative of MSE on this single datapoint

b) calculate
on this single datapoint
with respect to $\vec{\theta}$

c) shift θ in the direction
opposite to the derivative

There are many other variants of SGD used in practice:

- SGD with momentum
- Rmsprop
- NAG
- Adam (the most popular)
- AdaGrad
- AdaDelta