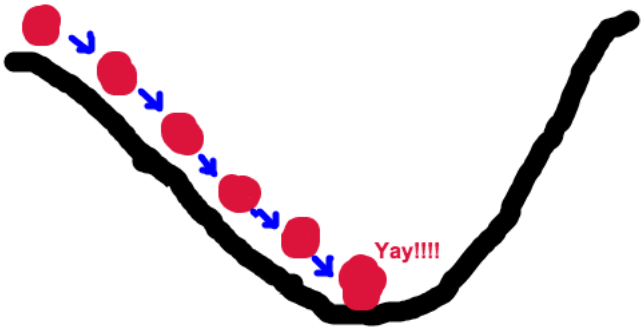
Gradient Descent - Problem of Hiking Down a Mountain

Udacity

Have you ever climbed a mountain? I am sure you had to hike down at some point? Hiking down is a great exercise and it is going to help us understand gradient descent.

Whats the goal when you are hiking down a mountain? - To have fun and to reach the bottom. Let's focus on reaching the bottom for now.



What is the red dot doing when it's hiking down? It's always going in the downward direction, until it hits the bottom. Let's call our friend calculus and see what she has to say about this.

Derivatives

Before we hop in, let me remind you a little bit about derivatives. There are different ways to look at derivatives, two of the most common ones are

- Slope of the tangent line to the graph of the function
- Rate of change of the function

Following are some of the common derivatives:

- $\frac{d(x^2)}{dx} = 2x$
- $\frac{d(-2y^5)}{dy} = -10y^4$
- $\frac{d(5-\theta)^2}{d\theta} = -2(5-\theta)$ (negative sign coming from $-\theta$)

Sounds great! What if, we have more than one variable in our function? Well, we will talk about partial derivatives then! Let's look at some examples:

- $\frac{\partial}{\partial x}(x^2y^2) = 2xy^2$
- $\frac{\partial}{\partial y}(-2y^5 + z^2) = -10y^4$
- $\frac{\partial}{\partial \theta_2}(5\theta_1 + 2\theta_2 - 12\theta_3) = 2$
- $\frac{\partial}{\partial \theta_2}(0.55-(5\theta_1 + 2\theta_2 - 12\theta_3)) = -2$ (Can you convince yourself where the $-$ is coming from?)

Now that we have familiarized ourselves with derivatives, let's start walking towards gradient descent algorithm. Wait, we are not there yet! Some more calculus!

What is Gradient?

Gradient is the generalization of derivatives in several variables. Let's use θ's as our variables in the following function $J(\theta_1, \theta_2, \theta_3)$

$$\begin{aligned} J(\Theta) &= 0.55 - (5\theta_1 + 2\theta_2 - 12\theta_3) \\ \nabla J(\Theta) &= \left\langle \frac{\partial J}{\partial \theta_1}, \frac{\partial J}{\partial \theta_2}, \frac{\partial J}{\partial \theta_3} \right\rangle \\ &= \langle -5, -2, 12 \rangle \end{aligned}$$

Here, ∇ is just a symbolic way of indicating that we are taking gradient of the function, and the gradient is inside $<$ and $>$ to denote that gradient is a vector.

Let's look at a slightly more complicated example. Make sure you really understand this, we will use this type of expression in Linear Regression with Gradient Descent.

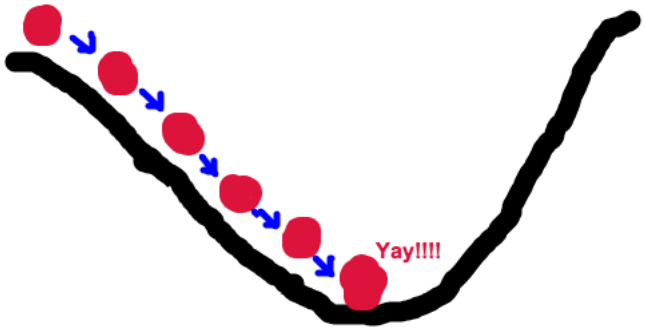
$$\begin{aligned} J(\Theta) &= \frac{1}{2}\left(0.55 - (5\theta_1 + 2\theta_2 - 12\theta_3)\right)^2 \\ \nabla J(\Theta) &= \left\langle \frac{\partial J}{\partial \theta_1}, \frac{\partial J}{\partial \theta_2}, \frac{\partial J}{\partial \theta_3} \right\rangle \\ &= \langle -5\left(0.55 - (5\theta_1 + 2\theta_2 - 12\theta_3)\right), -2\left(0.55 - (5\theta_1 + 2\theta_2 - 12\theta_3)\right), 12\left(0.55 - (5\theta_1 + 2\theta_2 - 12\theta_3)\right) \rangle \end{aligned}$$

Why do we care about gradient?

Gradient is a pretty powerful tool in calculus. Remember, in one variable, derivative gives us the slope of the tangent line. In several variables, **Gradient points towards direction of the fastest increase of the function**. This is extensively used in Gradient Descent Algorithm. Let's see how.

What is the idea behind Gradient Descent Algorithm?

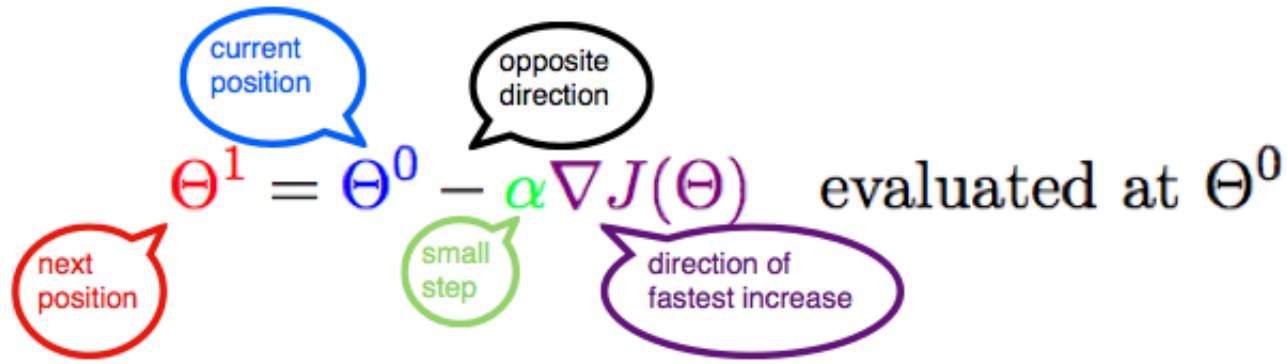
Gradient descent algorithm is an iterative process that takes us to the minimum of a function (This will not happen always, there are some caveats!). Let's look at the red dot example again:



If you want to reach the bottom, in which direction would you walk? In the downward direction, right? How do we find the downward direction? That's the direction opposite of the fastest increase. This means, if we are at point Θ^0 and want to move to lowest nearby point (this is why we say "local minimum") our next step should be at:

$$\Theta^1 = \Theta^0 - \alpha \nabla J(\Theta) \quad \text{evaluated at } \Theta^0$$

This needs some more clarification.



What's the deal with α?

α is called the **Learning Rate** or **step size**. Which means, we want to take baby steps so that we don't overshoot the bottom. This is particularly important when we are very close to the minimum. A smart choice of α is crucial. When α is too small, it will take our algorithm forever to reach the lowest point and if α is too big we might overshoot and miss the bottom.

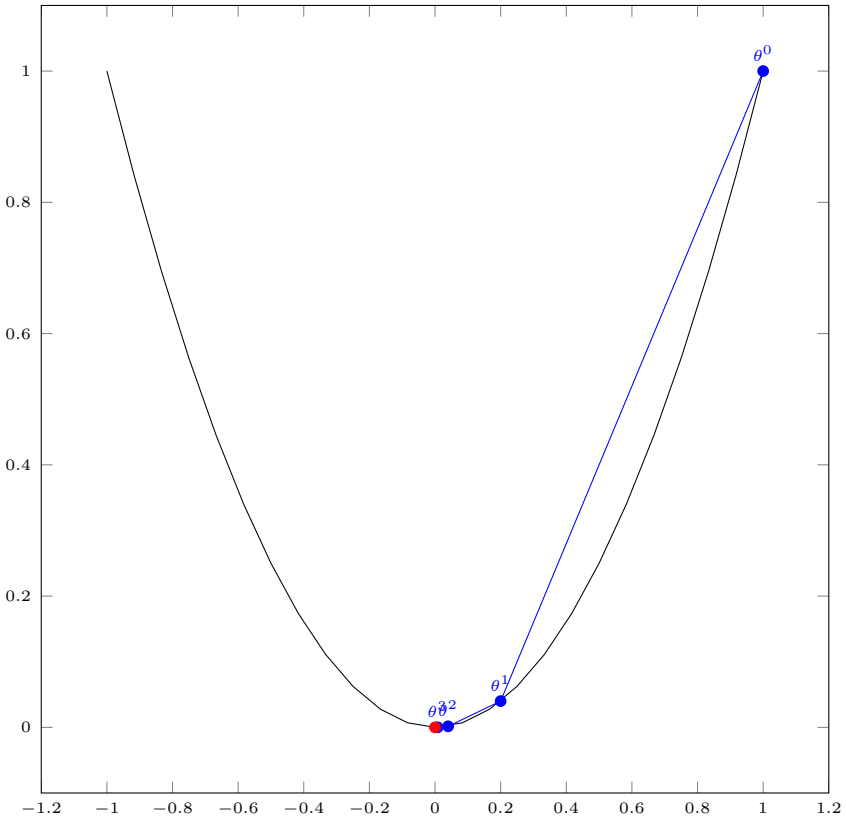
Why $-$ sign?

$-$ sign indicates that we are stepping in the direction opposite to that of ∇J i.e. we are stepping in the direction opposite to that of fastest increase.

Example in one variable

Let's see what this looks like in one variable. Suppose $J(\theta) = \theta^2$, then derivative $J'(\theta) = 2\theta$. Our initial choices are $\theta^0 = 1$ and $\alpha = 0.4$ Then,

$$\begin{aligned}
\theta^0 &= 1 \\
\theta^1 &= \theta^0 - \alpha * J'(\theta^0) \\
&= 1 - 0.4 * 2 \\
&= 0.2 \\
\theta^2 &= \theta^1 - \alpha * J'(\theta^1) \\
&= 0.04 \\
\theta^3 &= 0.008 \\
\theta^4 &= 0.0016
\end{aligned}$$



Example in two variables

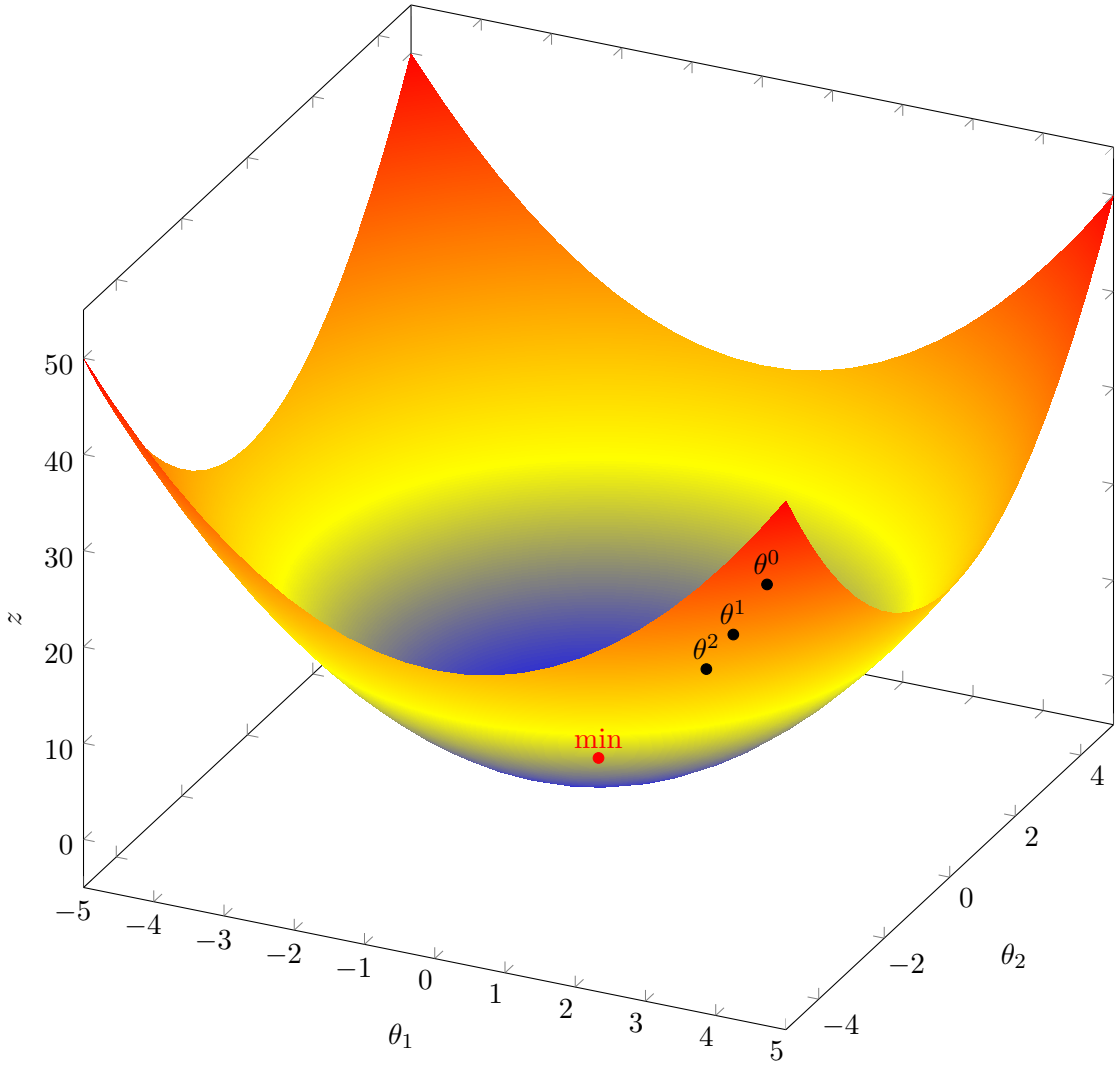
Suppose $J(\Theta) = \theta_1^2 + \theta_2^2$. By simple, observation we can see that $(0, 0)$ gives the minimum value for J. Let's see, if that's what we get by our gradient descent method.

Let's choose $\Theta^0 = (1, 3)$ and $\alpha = 0.1$.

$$\nabla J(\Theta) = \langle 2\theta_1, 2\theta_2 \rangle$$

Evaluated at $(1, 3)$ this gradient vector is $< 2, 6 >$

$$\begin{aligned}
\Theta^0 &= (1, 3) \\
\Theta^1 &= \Theta^0 - \alpha \nabla J(\Theta) \\
&= (1, 3) - 0.1(2, 6) \\
&= (0.8, 2.4) \\
\Theta^2 &= (0.8, 2.4) - 0.1(1.6, 4.8) \\
&= (0.64, 1.92) \\
\Theta^3 &= (0.512, 1.536) \\
\Theta^4 &= (0.4096, 1.2288000000000001) \\
&\vdots \\
\Theta^{10} &= (0.10737418240000003, 0.32212254720000005) \\
&\vdots \\
\Theta^{50} &= (1.1417981541647683\text{e}^{-05}, 3.425394462494306\text{e}^{-05}) \\
&\vdots \\
\Theta^{100} &= (1.6296287810675902\text{e}^{-10}, 4.888886343202771\text{e}^{-10})
\end{aligned}$$

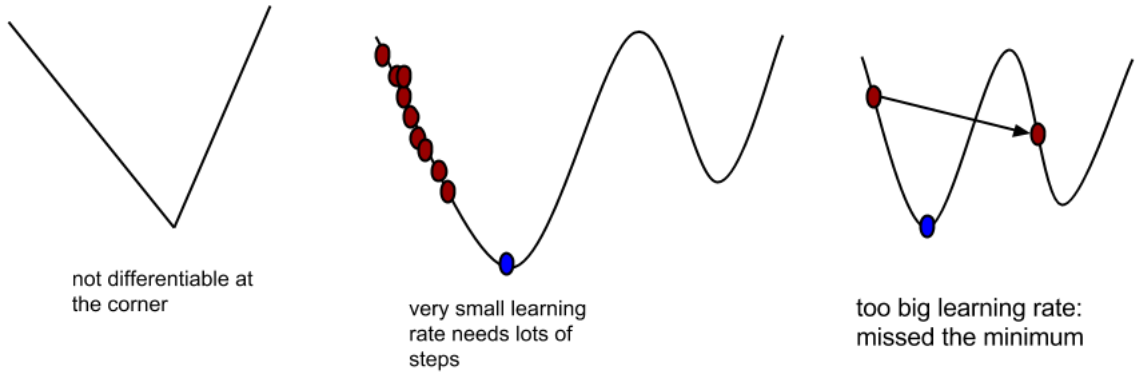


Yay! we see that we are indeed approaching the minimum, which we know is (0, 0).

Caution

Following are few of the most important things to keep in mind:

- Function must be differentiable.
- Learning rate should not be too small or too large.



Application in Linear Regression

Gradient Descent algorithm is one of the methods used in linear regression to find the minimum of the error or cost function. Remember that your error or cost function must be differentiable to be able to use gradient descent. This is one of the reasons behind squaring the differences. Using absolute values to calculate errors may lead to "corners" as illustrated in the picture above.