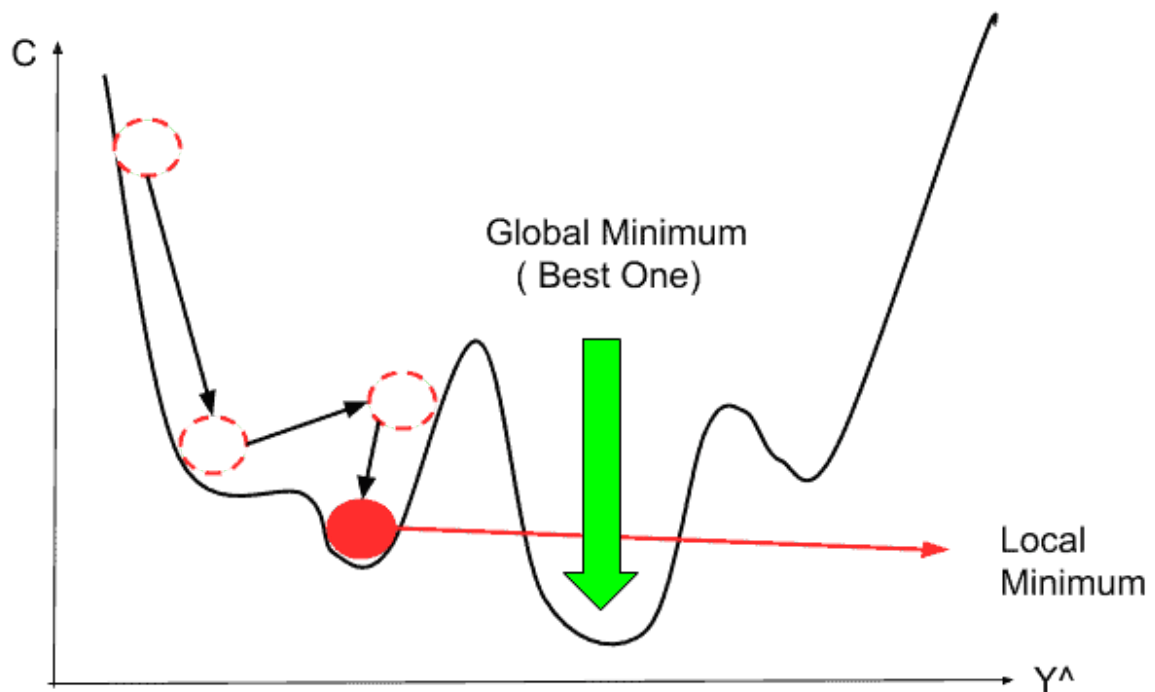




[Home](#) » [Blogs](#) » [Deep Learning](#) » Stochastic Gradient Descent- A Super Easy Complete Guide!

# Stochastic Gradient Descent- A Super Easy Complete Guide!

[Deep Learning](#)



Do you wanna know What is **Stochastic Gradient Descent**?. Give your few minutes to this blog, to understand the **Stochastic Gradient Descent** completely in a super-easy way. You will understand **Stochastic Gradient Descent** in a few minutes. So read this full article :-).

Hello, & Welcome!

In this blog, I am gonna tell you-

1. Brief of **Gradient Descent**.
2. What is **Stochastic Gradient Descent**?
3. How **Stochastic Gradient Descent** work and how it is different from Normal/Batch Gradient Descent?

I have explained all the details of Gradient Descent in my previous article. If you wanna read this article, you can read it [here](#). So in this article, I will give you a brief about Gradient Descent, so that you will understand **Stochastic Gradient Descent**.

So without wasting your time, let's get started-

## **Brief of Gradient Descent-**

A Gradient Descent is a very famous optimization technique that is used in machine learning and in deep learning. The main purpose of gradient descent is to minimize the cost function. The cost function is nothing but a method to find out the error between the actual output and the predicted output.

As in neural networks, the main goal is to predict the output which is similar to the actual output. In order to check the difference between actual output ( $y$ ) and the predicted output ( $\hat{y}$ ), we calculate the cost function. The formula of the cost function is-

$$\text{cost function} = \frac{1}{2} \text{square}(y - \hat{y})$$

The lower the cost function, the predicted output is closer to the actual output. So, to minimize this cost function we use Gradient Descent.

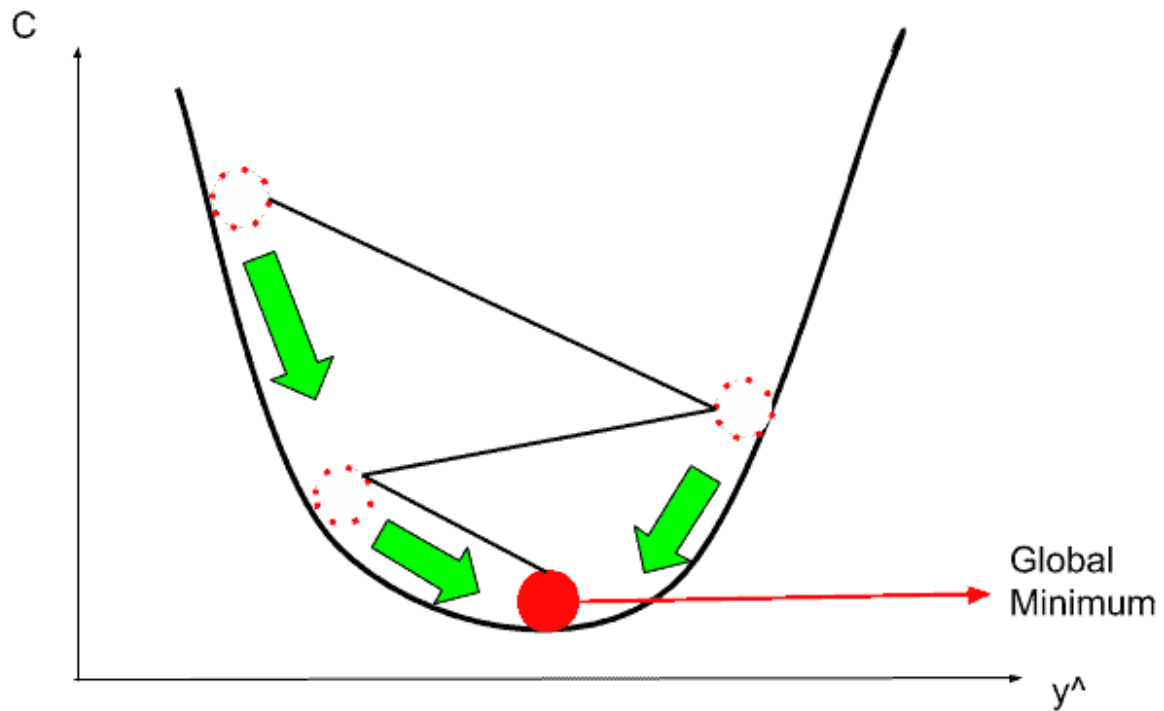
Gradient Descent is of three types-

1. Stochastic Gradient Descent.
2. Batch Gradient Descent.
3. Mini-Batch Gradient Descent.

In this article, I will explain to you how these three are different from each other.

## What is Stochastic Gradient Descent?

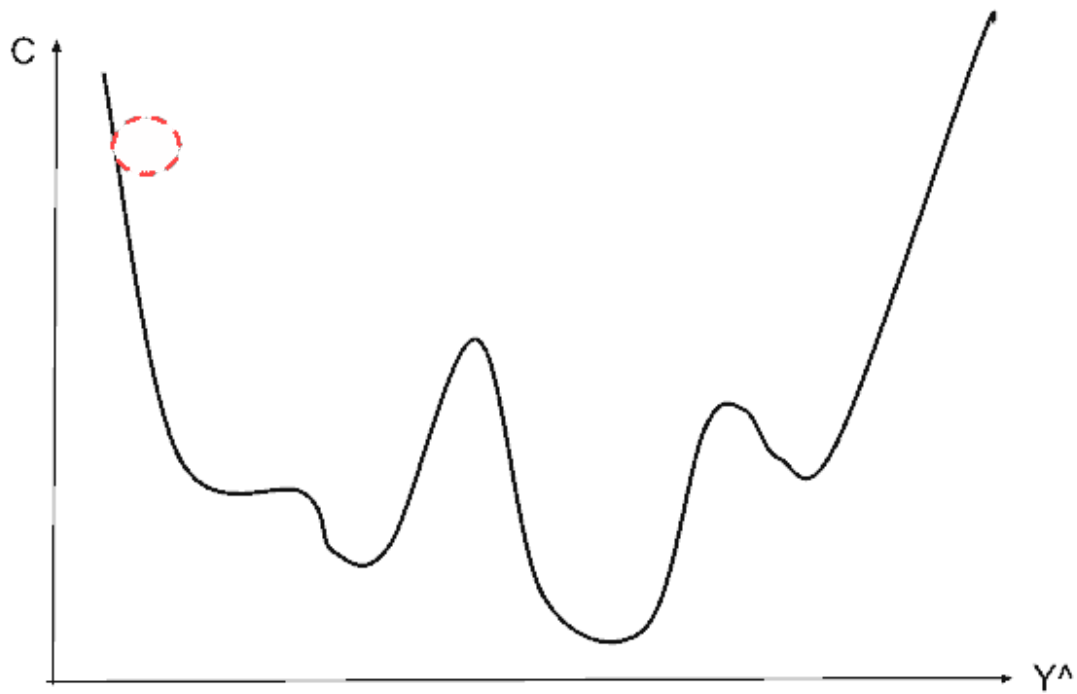
In Gradient Descent the cost function should be convex. The convex function looks something like that-



So, in the convex function, there is one global minimum as you can see in the image.

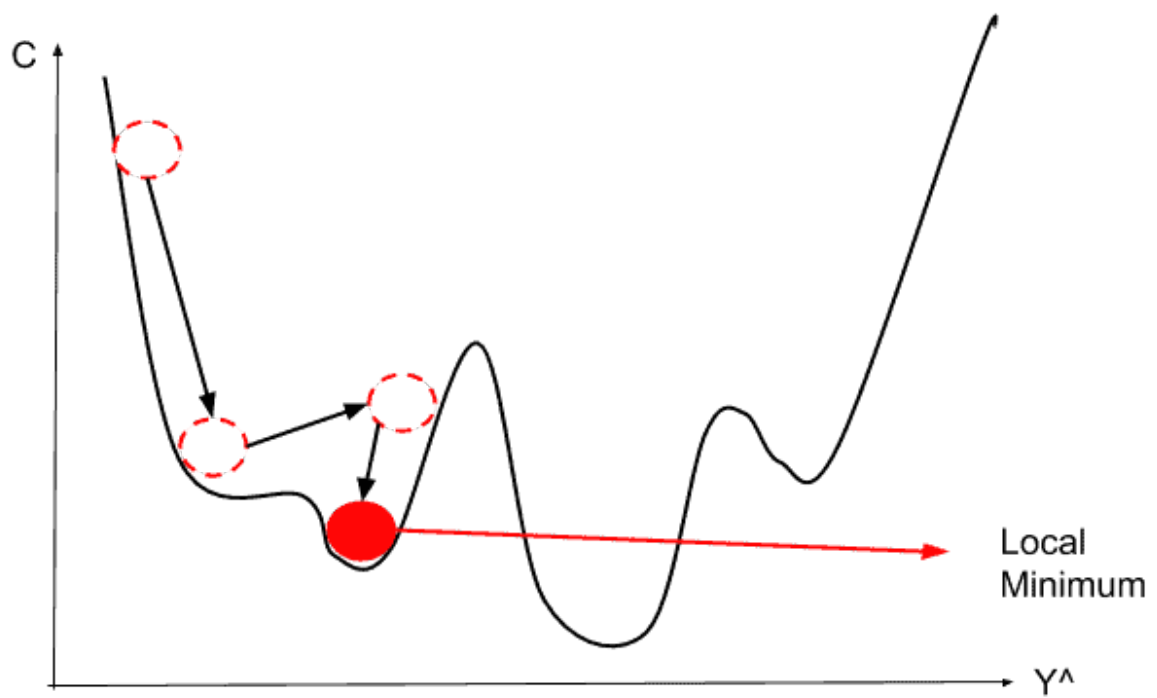
But,

When our cost function is not convex? and it looks something like that-

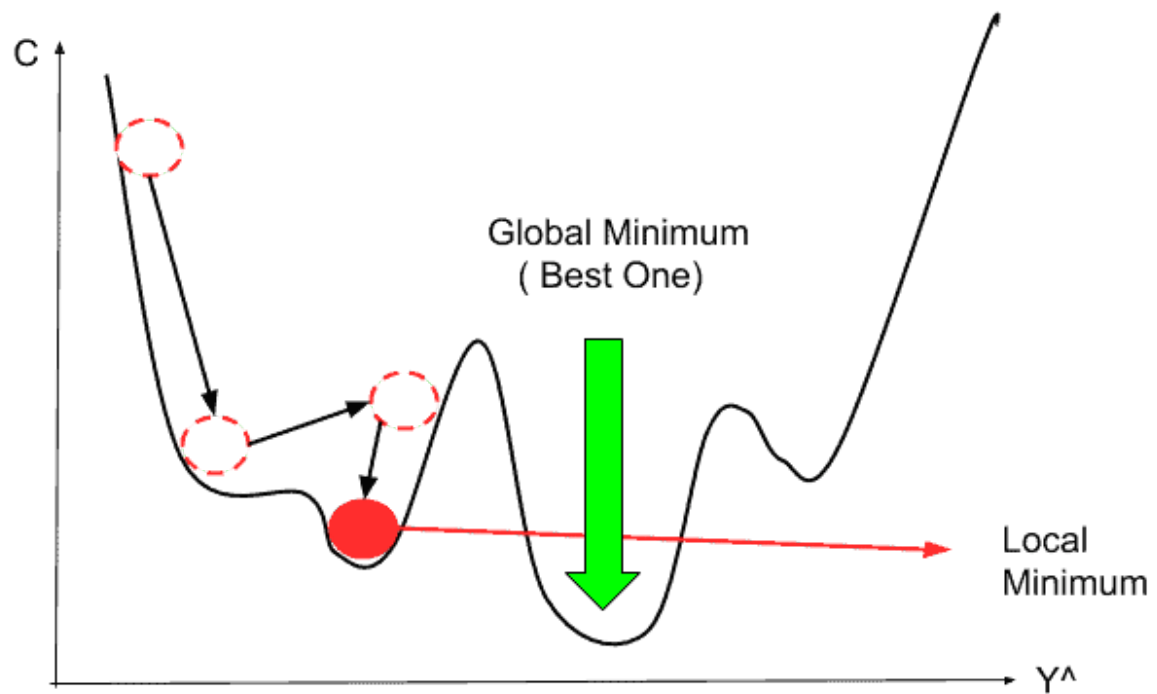


We get non-convex function when we choose a cost function which is not the square difference between  $\hat{y}$  and  $y$ .

So let's see what will happen if we apply normal gradient descent to non-convex function.



So, We found a local minimum instead of a global minimum. And this is not the correct one, because this is not the global minimum. The global minimum is that one-



As we have found the wrong one, therefore we don't have the correct weights. And we don't have an optimized neural network.

To solve this local minimum problem in the neural networks, the **Stochastic Gradient Descent** is used. It doesn't require the cost function to be convex. It easily works on a non-convex cost function.

Now, let's see-

## How Stochastic Gradient Descent work and how it is different from Normal/Batch Gradient Descent?

Suppose this is our dataset where neural network has to predict the percentage that person will buy the house or not, based on his/her salary, spending, and

saving. We have actual output- Buy House as  $y$ .

Row Id	Salary (per annum)	Spending (per annum)	Saving (per annum)	Buy House( Percentage)
1	\$ 100K	\$ 40K	\$ 60K	90%
2	\$ 160K	\$ 80K	\$ 80K	95%
3	\$ 120K	\$ 90k	\$ 30K	70%
4	\$ 130K	\$ 60K	\$ 70K	91%

So in normal gradient descent, we take all of our rows and plug them into a neural network. After plugging into neural network, we calculate the cost function with the help of this formula- cost function =  $\frac{1}{2} \text{square}(y - y^{\wedge})$ . Based on the cost function, we adjust the weights. This process is of normal Gradient Descent. It is also known as **Batch Gradient Descent**.

So, in Batch gradient descent, we take the whole batch from our sample.

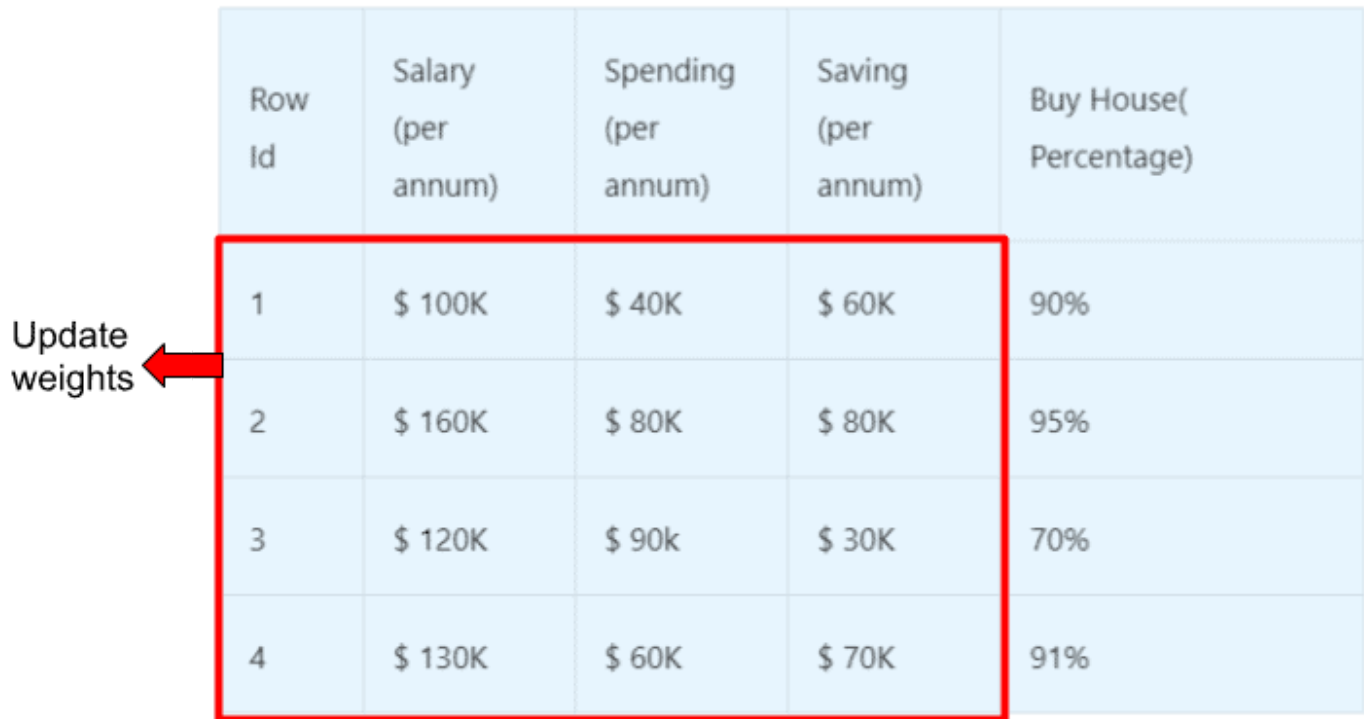
But,

In **Stochastic Gradient Descent**, we take the row one by one. So we take one row, run a neural network and based on the cost function, we adjust the weight. Then we move to the second row, run the neural network, based on the cost function, we update the weight. This process repeats for all other rows. So, in stochastic, basically, we are adjusting the weights after every single row rather than doing everything together and then adjusting weights.

## Difference between Stochastic Gradient Descent & Batch Gradient Descent-



So, these are two different approaches. To understand it more clearly, let's see in this image-



Row Id	Salary (per annum)	Spending (per annum)	Saving (per annum)	Buy House( Percentage)
1	\$ 100K	\$ 40K	\$ 60K	90%
2	\$ 160K	\$ 80K	\$ 80K	95%
3	\$ 120K	\$ 90k	\$ 30K	70%
4	\$ 130K	\$ 60K	\$ 70K	91%

## Batch Gradient Descent

So this is Batch Gradient Descent, but Stochastic Gradient Descent works with a single row. Let's understand with this image-

	Row Id	Salary (per annum)	Spending (per annum)	Saving (per annum)	Buy House( Percentage)
Update weights ←	1	\$ 100K	\$ 40K	\$ 60K	90%
Update weights ←	2	\$ 160K	\$ 80K	\$ 80K	95%
Update weights ←	3	\$ 120K	\$ 90k	\$ 30K	70%
Update weights ←	4	\$ 130K	\$ 60K	\$ 70K	91%

## Stochastic Gradient Descent

So after seeing both images, I hope you understand the difference between Batch Gradient Descent and Stochastic Gradient Descent.

The stochastic GD helps you to avoid the problem of a local minimum. Because in Stochastic GD, fluctuations are higher. It's doing one row at a time and therefore the fluctuations are much higher. It is also more likely to find the global minimum rather than just a local minimum.

Another important point in Stochastic Gradient Descent is that it is much faster than batch gradient descent. If you look at the process of the stochastic GD, you might think that it is slower because it is adjusting weights after every row. But actually, it is faster because it doesn't have to load all the data into memory, run,

and wait until all of these rows run together. In stochastic gradient descent, you run row one by one, that's why it is much lighter and faster.

The main advantage of Stochastic Gradient Descent is that it is a deterministic algorithm. That means it is random. In Batch gradient descent, you start with some weights and every time you run the batch gradient descent you will get the same results to update the weights. But in stochastic gradient descent, you will not get the same results, because you are picking at random. So if you have the same weights at the start, but after running a neural network you will get different results.

## Mini-Batch Gradient Descent-

There is one more method that is- mini-batch Gradient Descent. In the mini-batch gradient descent method, you can run batches of rows. The batch may be of 5 rows, 10 rows, or anything else. You can decide the number of batches. So after running one batch, the weights will be updated.

So that's all about **Stochastic Gradient Descent**. I hope now you understand it clearly. If you have any questions, feel free to ask me in the comment section.

Enjoy Learning!

All the Best!

If you wanna know about the neural network learning process? read it from [here](#).

Thank YOU!

## Thought of the Day...

*' It's what you learn after you know it all that counts.'*

*– John Wooden*

**Read Deep Learning Basics Here.**

---

[← Previous Post](#)

[Next Post →](#)

---

## Leave a Comment

Your email address will not be published. Required fields are marked \*

Type here..

Name\*

Email\*

Website

☐ Save my name, email, and website in this browser for the next time I comment.

Post Comment »

Search ...



## Coursera

## Recent Posts

[Best Way to Learn AWS DevOps in 2022- \[Bestseller Courses\]](#)

[25 Best Python Courses You Must Know in 2022 \[Free Courses Included\]](#)

[Best Resources to Learn Data Structures and Algorithms- \(YouTube, Tutorials, Courses, Books, etc\)- 2022](#)

[Udacity Computer Vision Nanodegree Review 2022- Is It Worth it?](#)

[15 Best Data Science Programs Online in 2022- \[Free Programs Included\]](#)

## Categories

[Artificial Intelligence](#)

[Data Mining](#)

[Data Science](#)

[Deep Learning](#)

[Machine Learning](#)

[Python](#)

## Learn Data Science

# Subscribe for Latest Articles!

Get the Latest Articles Weekly!

**Subscribe**

Follow Us on Twitter-

Follow Our Quora Space



Join Our Facebook Group-

Subscribe Our Youtube Channel

[Contact Me](#) [About Me](#) [Privacy Policy](#) [Disclaimer for MLTut](#)  
[Terms of Service](#) [Write For Us](#)

www.mltut.com is a participant in the Amazon Services LLC Associates Program,  
an affiliate advertising program designed to provide a means for sites to earn

advertising fees by advertising and linking to amazon.com. Additionally, [www.mltut.com](http://www.mltut.com) participates in various other affiliate programs, and we sometimes get a commission through purchases made through our links.