

## Assignment B-4

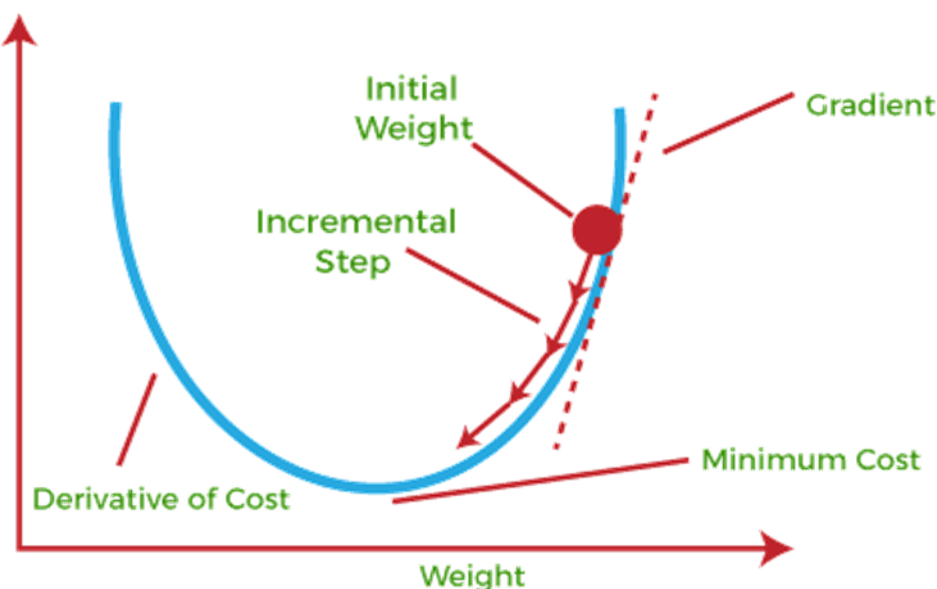
Problem Statement:

Implement Gradient Descent Algorithm to find the local minima of a function.

For example, find the local minima of the function  $y=(x+3)^2$  starting from the point  $x=2$ .

Gradient Descent:

Gradient Descent is an iterative algorithm that is used to minimize a function by finding the optimal parameters. Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms. It is basically used for updating the parameters of the learning model. We start with a random point on the function and move in the **negative direction** of the **gradient of the function** to reach the **local/global minima**.



Gradient Descent algorithm

Step 1 : Initialize  $X_0 = 2$ . Then, find the gradient of the function,  $dy/dx = 2*(x+3)$ .

Step 2 : Move in the direction of the negative of the gradient . the step size is determined by learning rate. Let us assume the learning rate  $\rightarrow 0.01$

Step 3 : Let's perform 2 iterations of gradient descent

Iteration 1

$$X_1 = X_0 - (\text{learning rate}) * dy/dx;$$

$$X_1 = X_0 - (0.01) * (2(X_0+3))$$

$$X_1 = 2 - (0.01) * (2(2+3))$$

$$X1=2-0.1$$

$$X1=1.9$$

Iteration 2

$$X2= X1 -(\text{learning rate}) * dy/dx;$$

$$X2= X1 -(0.01)* (2(X1+3))$$

$$X2=1.9-(0.01)*(2(1.9+3))$$

$$X2=1.9-0.098$$

$$X2=1.802$$

Step 4 : We can observe that the X value is slowly decreasing and should converge to -3 (the local minima) for the function given.

To stop the iterations two conditions are set

1. Select a precision variable in algorithm which calculates the difference between two consecutive “x” values. If the difference between x values from 2 consecutive iterations is lesser than the precision set, then stop the algorithm!
2. Set maximum count of iteration and exit the iterations.

Pseudo code for Gradient decent algorithm

while previous\_step\_size > precision and iters < max\_iters:

    prev\_x = cur\_x #Store current x value in prev\_x

    cur\_x = cur\_x - rate \* df(prev\_x) #Grad descent

    previous\_step\_size = abs(cur\_x - prev\_x) #Change in x

    iters = iters+1 #iteration count

Database Used: Inbuilt

Python: Colab, spider or similar flatform

Code (As attached) & Graphs (wherever applicable) \_\_\_\_\_

You tube link: <https://www.youtube.com/watch?v=kEKwJ14uEbI>

Metrics used for performance measurement: \_\_\_\_\_

Conclusion

Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms. The steps of gradients are controlled by learning rate, gradient descent and precision. It used in ANN and deep learning.