Labwork 1: Gradient Descent

Pham Gia Phuc

April 2024

1 Introduction

1.1 Definition

Gradient descent is an optimization algorithm which is commonly-used to train machine learning models and neural networks. It trains machine learning models by minimizing errors between predicted and actual results.

1.2 Usage

The equation below describes what the gradient descent algorithm does:

$$x = x_A - L * d(f(x_A))$$

where x is the next stage, while x_A is the current position. The minus sign refers to the minimization part of the gradient descent algorithm, L is the learning rate and $d(f(x_A)$ is the derivative of $f(x_A)$.

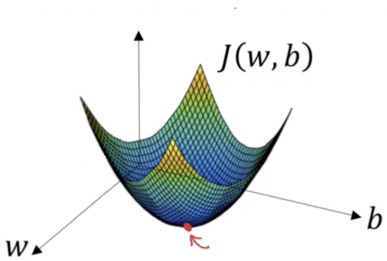
This formula basically tells the next position we need to go, which is the direction of the steepest descent.

Overall, gradient descent algorithm is used to minimize your cost-function $J(w, b)$ and reach its local minimum by tweaking its parameters (w and b). The image below shows the horizontal axes representing the parameters (w and b), while the cost function J(w, b) is represented on the vertical axes. Gradient descent is a convex function.

2 Implementation

To implement the gradient descent, it is able to follow the steps:

* - Define the function and its derivative. In the context of this report, first is to define function $f(x) = x^2$, and its derivative $df(x)$ which returns 2x. The derivative of a function gives the slope of the function at a given point.



* - The next step is to define to gradient descent function, where the algorithm is implemented. It takes three arguments: initial value of x, learning rate which is the step size for each iteration and the number of iterations to perform.

* - The function will be entered to a loop that runs for number of iterations. On each iteration, it calculates the gradient of the function at the current point, then updates the input value by subtracting the gradient times the learning rate. This is the key step of the gradient descent algorithm.

* - After all iterations are complete, the function returns the final value of initial value x. This is the estimate for the minimum of the function $f(x)$

3 Evaluation

Time	x	f(x)
1	6.40	40.96
2	5.12	26.21
3	4.10	16.78
4	3.28	10.74
5	2.62	6.87
6	2.10	4.40
7	1.68	2.81
8	1.34	1.80
9	1.07	1.15
1	0.86	0.74

Minimum value occurs at x = 0.86

The result above are provided by gradient descent function applied to function:

$$f(x) = x^2$$

with learning rate of 0.1 and starting from an initial value of $x = 8$.

The 'Time' column represents the iteration number. It starts from 1st to 10th iteration.

The ’x’ column represents the value of ’x’ at each iteration. x starts at 6.4 and gradually decreases at each step.
The $f(x)$ column represents the value of $f(x) = x^2$ function at the corresponding value x. This value is also decreasing at each step.

After 10 iterations, the value of the function occurs at $x = 0.86$. This is the minimum value the algorithm has found.

4 Conclusion

In conclusion, I have created a simple gradient descent algorithm that can find the minimum of function $f(x) = x^2$, where the implementation and results provide a demonstration of how the algorithm works. However, it is crucial for choosing the parameters of initial value, learning rate and number of iteration as they might affect the effectiveness of the algorithm.