Gradient Descent

Mahmoud Fathy

1 Introduction

Gradient descent is the backbone of the learning process for various algorithms, including linear regression, logistic regression, support vector machines, and neural networks which serves as a fundamental optimization technique to minimize the cost function of a model by iteratively adjusting the model parameters to reduce the difference between predicted and actual values, improving the model's performance.

2 The Core Algorithm

Gradient Descent is an algorithm used to find the best solution to a problem by making small adjustments in the right direction.

$$w = w - \alpha \frac{\partial J(w)}{\partial w}$$

Gradual steps used in descent is done by defining the learning rate (α). It is used to determine how quickly or slowly the algorithm converges towards the minimum value. A too small alpha significantly increases training time and computational cost especially for large datasets while a too large one leading overshooting the minimum of cost function without settling. For a positive gradient, subtracting it reduces w hence reducing the cost function and for a negative gradient it increases w reducing the cost function.

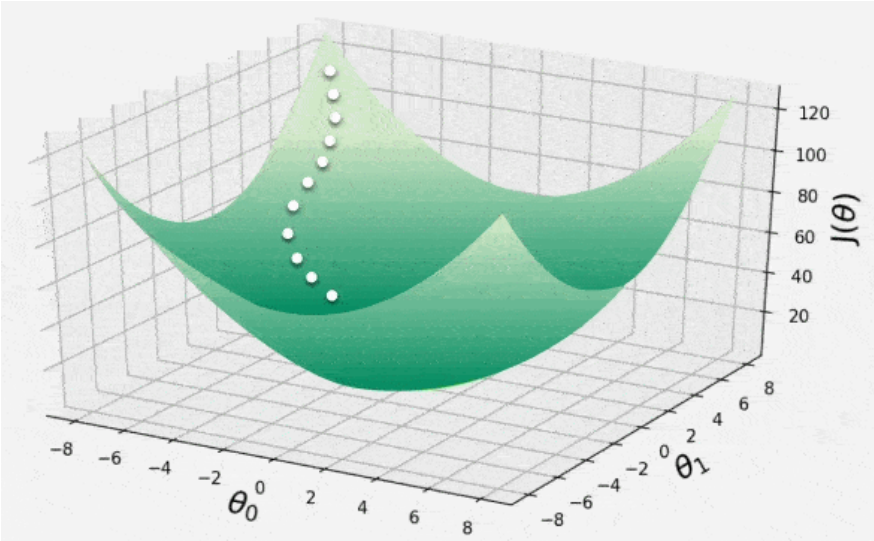

Figure 1: Cost Function

3 Types of Gradient Descent

3.1 Batch Gradient descent

Computes the gradient of the cost function using the entire training dataset for each iteration. This approach ensures that the computed gradient is precise, but it can be computationally expensive when dealing with very large datasets.

3.2 Stochastic Gradient Descent

Addresses the inefficiencies of Batch Gradient Descent by computing the gradient using only a single training example (or a small subset) in each iteration. This makes the algorithm much faster since only a small fraction of the data is processed at each step.

3.3 Mini-Batch Gradient Descent

Mini-batch gradient descent is a optimization method that updates model parameters using small subsets of the training data called mini-batches. This technique offers a middle path between the high variance of stochastic gradient descent and the high computational cost of batch gradient descent. They are used to perform each update, making training faster and more memory-efficient. It also helps stabilize convergence and introduces beneficial randomness during learning.

4 Applications in machine learning

4.1 Training Machine Learning Models

Neural networks are trained using Gradient Descent (or its variants) in combination with backpropagation. Backpropagation computes the gradients of the loss function with respect to each parameter (weights and biases) in the network by applying the chain rule.

4.2 Minimizing the Cost Function

The algorithm minimizes the cost function, which quantifies the error or loss of the model's predictions compared to the true labels. For example, in linear regression:

$$w = w - \alpha \frac{\partial J(w,b)}{\partial w}, b = b - \alpha \frac{\partial J(w,b)}{\partial b}$$