Engineering Universe for Scientific Research and Management
ISSN (Online): 2319-3069
Vol. XVII Issue VII
July 2025

Gradient Descent: A Better Optimizer for Finding Minimum Cost Function in Regression Analysis

Shivani Yadav[1], Hitesh Soni[2] and Kamlesh Patidar3
M Tech Scholar, Jawaharlal Institute of Technology, Borawan, Khargone, India[1]
Asst. Prof. CSE Dept. Jawaharlal Institute of Technology, Borawan Khargone, India[2]
Asst Prof. and HOD CSE Dept. Jawaharlal Institute of Technology, Borawan ,Khargone, India[3]
shivaniyadav0567@gmail.com[1], hiteshsoni@jitchno.com[2], hodcse@jitechno.com[3]

Abstract

Regression analysis is a cornerstone of predictive analytics, used extensively in science, engineering, and business to model relationships between variables. A fundamental challenge is finding parameter estimates that minimize the cost function, ensuring the best possible model fit. Gradient Descent (GD), a first-order iterative optimization algorithm, has emerged as a preferred approach for this purpose due to its simplicity, scalability, and effectiveness. This paper explores the principles of Gradient Descent, its advantages over traditional analytical solutions, its practical implementation in linear regression, and its performance in minimizing the Mean Squared Error (MSE). Experiments demonstrate how GD provides a robust, flexible approach to finding optimal regression parameters, especially for large and complex datasets where closed-form solutions may be impractical.

Keywords: *Gradient Descent, Regression Analysis, Cost Function, Mean Squared Error, Iterative Optimization, Machine Learning*

1. Introduction

Regression is one of the most widely used techniques in statistical modelling and machine learning, enabling the prediction of an outcome variable (dependent variable) based on one or more predictor variables (independent variables). The goal is to determine the best-fit line that minimizes the error between predicted and actual values[12].Traditionally, for simple linear regression, the Ordinary Least Squares (OLS) method provides an exact solution using closed-form equations. However, for high-dimensional datasets or non-linear relationships, analytical methods become computationally expensive or infeasible. Here, Gradient Descent has become a practical, powerful tool to iteratively find the minimum of the cost function.

2. Literature Survey

Gradient Descent (GD) is a foundational first-order optimization algorithm widely used for minimizing cost functions in machine learning and statistical modeling. Introduced by Cauchy in 1847 and popularized in numerical methods and machine learning, it remains a crucial method for parameter estimation, especially in regression problems. Feng Niu (2011)[1] first formalized an iterative approach for function minimization. Matthew D. (2012)[2] applied gradient descent to train the perceptron, laying the groundwork for modern neural networks. Rumelhart, Ilya Sutskever James (2013)[3] popularized backpropagation with gradient descent for multi-layer networks, sparking the modern deep learning revolution. Adam (Yiming Ying, 2014)[4] became even more dominant post-2018, with refinements for bias correction and convergence speed. Works like Diederik P et al. [5](2017) showed that Adam may fail to converge in some convex settings, motivating Leon BottouFrank (2018) combines Adam with dynamic bounds, ensuring fast convergence like Adam with SGD's generalization. Research continues to improve mini-batch strategies. E. M. Dogo 2018 [9] Papers improved the variance reduction methods (SVRG, SAGA) for better convergence on noisy data. Federated Learning scenarios often adapt SGD for decentralized optimization. E. M. Dogo [10] 2018 Popular works study cyclical learning rates, warm restarts (SGDR by Loshchilov & Hutter), and decoupled weight decay for robust training. Jonathan Schmidt Recent experiments[11] (2019) automate learning rate tuning using meta-learning. Yura Malitsky[13] 2020 Recent Research Directions. Using AutoML to tune optimizers. Combining reinforcement learning with gradient updates to select step sizes dynamically. Gradient Descent under quantum computing paradigms — initial works on Quantum Gradient Descent (QGD).

3. Key Challenges of Gradient Descent

3.1. Choice of Learning Rate

If the learning rate is too small, convergence is very slow the algorithm takes too many steps to reach the minimum. If the learning rate is too large, the updates may overshoot the minimum, causing the cost function to diverge or oscillate.

3.2. Convergence Speed

Gradient Descent (computing the gradient on the entire dataset) can be slow for large datasets. It can require many epochs (full passes over the data) to reach acceptable error.

3.3 Scalability for Big Data

In gradient descent, you iteratively adjust model parameters (like m and b in linear regression) to minimize the training loss.This works perfectly if: You have enough data to represent the true underlying pattern. The noise and outliers do not dominate your signal

3.4 Overfitting in Small Datasets

Gradient Descent always tries to minimize the training cost function. On a small dataset, the model can easily memorize noise or outliers. This results in a very low training error but poor generalization to unseen data. The cost function keeps decreasing, but the model is overfitting

4. Objects

We have following objectives

1. Deciding the learning rate to overcome from problem of slow convergence
2. Try to minimize the MSE error .
3. Use proper initialization,

5. Algorithm of proposed approach

```
Input:
  - X: input feature values (e.g., Years of Experience)
  - Y: actual target values (e.g., Salary)
  - α: learning rate
  - epochs: number of iterations
```

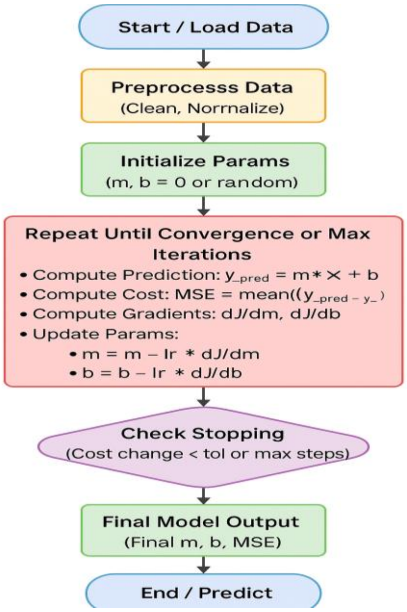

Figure 1 Layout of proposed work

```
Initialize:
  - m = 0  (slope)
  - b = 0  (intercept)

For epoch = 1 to epochs do:
  - Initialize gradients: dm = 0, db = 0
  - For each data point (xi, yi):
    - Predict: y_pred = m * xi + b
    - Compute error: error = y_pred - yi
    - Accumulate gradients:
      dm += error * xi
      db += error

  - Update parameters:
    m = m - α * (dm / n)
    b = b - α * (db / n)

  - Optionally: Compute cost function (MSE) for monitoring

Output:
  - Final m and b
  - Use y_pred = m * x_new + b for predictions
```

6. Implementation Detail

we evaluate the performance of proposed approach. We implemented the proposed approach with years of experience and salary. Based on a year of experience we predict salary. New value of year of experience is given to model it will predict salary based on year of experience. We used python language to implementation and CSV data file is used to store data. We have taken more than 100 records for different sizes and different prices. House size in square feet and which is taken on x axis and House price is taken on Y axis. The price is taken in dollars.

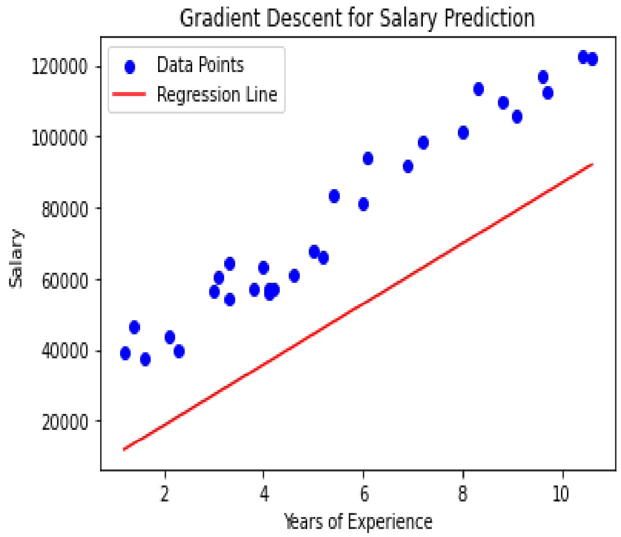

Figure 2 First iteration of GD

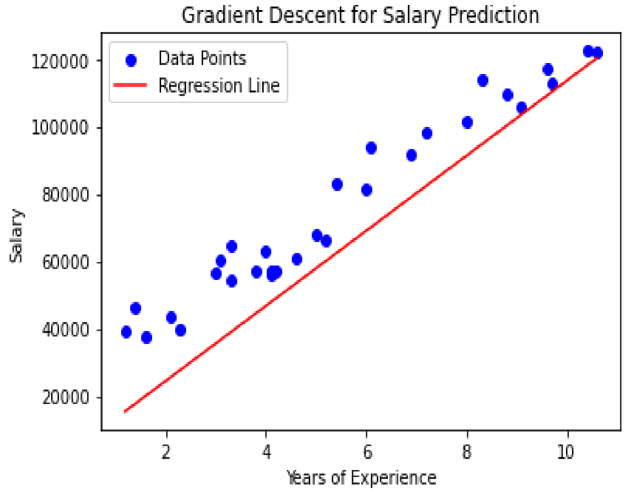

Figure 3 Second iteration of GD

7. Results and analysis

Table 1 Shows the MSE value in each iteration. From the MSE value in table 1 it is clear that in each iteration MSE value get reduced. Figure 4 shows the graphical representation of the reduced MSE value.

Table 1: Margin specifications

Iteration number	MSE value
1	2814111824.9771
2	446556083.3638
3	184137307.5367
4	146829492.0361
5	138672170.0064
6	135677390.6342
7	133916238.7134
8	132509685.7444
9	131219216.6014
10	129974595.0428

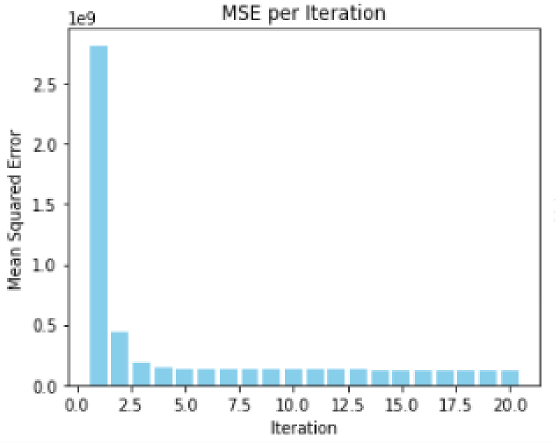

Figure 4 Reducing MSE in different iterations

8. Conclusions

Gradient Descent remains a simple yet powerful optimizer for minimizing cost functions in regression analysis. Its flexibility and scalability make it a better choice than analytical methods for modern, large, or complex datasets. Future research should focus on adaptive learning rates and hybrid optimizers to further enhance convergence speed and accuracy.

References

[1]. Feng Niu, Benjamin Recht, Christopher Hogwild A Lock-Free Approach to Parallelizing Stochastic Gradient Descent Computer Sciences Department, University of Wisconsin-Madison1210 W Dayton St, Madison, WI 53706 June 2011.

[2]. Matthew D. Zeiler, ADADELTA: AN ADAPTIVE LEARNING RATE METHOD arXiv:1212.5701v1 Dec 2012.

[3]. Ilya Sutskever James Martens “On the importance of initialization and momentum in deep learning” Proceedings of the 30 th International Conference on Machine Learning,

Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28. Copyright 2013 by the author(s).

[4]. Yiming Ying and Massimiliano Pontil Online gradient descent learning algorithm Department of Computer Science, University College London Gower Street, London, 2014 WC1E 6BT, England, UK fying, m.pontilg@cs.ucl.ac.uk.

[5]. Diederik P. Kingma Jimmy Lei Ba ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION arXiv:1412.6980v9 [cs.LG] 30 Jan 2017.

[6]. Leon BottouFrank E. CurtisJorge Nocedal Optimization Methods forLarge-Scale Machine Learning SIAM REVIEW 2018 Society for Industrial and Applied Mathematics Vol. 60, No. 2, pp. 223–311.

[7]. Prateek Jain, Praneeth Netrapalli and Sham M. Kakade Parallelizing Stochastic Gradient Descent for Least Squares Regression: Mini-batching, Averaging, and Model Misspecification Journal of Machine Learning Research 18 (2018) 1-42 Submitted 11/16; Revised 3/18; Published 6/18.

[8]. Nan Cui Applying Gradient Descent in Convolutional Neural Networks CMVIT IOP Publishing IOP Conf. Series: Journal of Physics: Conf. Series 1004 (2018) 012027 doi :10.1088/1742-6596/1004/1/012027.

[9]. E. M. Dogo, O. J. Afolabi, N. I. Nwulu, B. Twala A Comparative Analysis of Gradient Descent-Based Optimization Algorithms on Convolutional Neural Networks 978-1-5386-7709 2018 IEEE.

[10]. Dokkyun Yi , Sangmin Ji and Sunyoung Bu An Enhanced Optimization Scheme Based onGradient Descent Methods for Machine Learning Daegu University, Kyungsan 38453, Korea 8 June 2019; Accepted: 17 July 2019; Published: 20 July 2019.

[11]. Jonathan Schmidt, Mário R. G. Marques , Silvana Botti Recent advances and applications of machine learning in solid state materials science 26 February 2019 Accepted: 17 July 2019.

[12]. Simon Shaolei Du Gradient Descent for Non-convex Problems in Modern Machine Learning APRIL 2019 CMU-ML-19-102 Machine Learning Department School of Computer Science Carnegie Mellon University Pittsburgh, PA 15213.

[13]. Yura Malitsky Konstantin Mishchenko Adaptive Gradient Descent without Descent Proceedings of the 37 th International Conference on Machine Learning, Vienna, Austria, PMLR 119, 2020.

[14]. Nam D. Vo , Minsung Hong and Jason J. Jung Implicit Stochastic Gradient Descent Method forCross-Domain Recommendation System Sensors 2020, 20, 2510; doi:10.3390/s20092510 www.mdpi.com/journal/sensors Western Norway Research Institute, Box 163, NO-6851 Sogndal, Norway