JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Review on First-Order Iterative Optimization Techniques for Machine Learning

Mr. Himanshu Mishra
PG Student
Computer Engineering,
Sanghavi college of engineering,Nashik

Mr.Puspendu Biswas
HOD
Computer Engineering,
Sanghavi college of engineering, Nashik

Abstract-:

Optimization is a fundamental component of machine learning, as it enables models to learn parameter values that minimize prediction error. Among the wide range of optimization strategies, first-order iterative methods are the most extensively used due to their mathematical simplicity, low computational cost, and scalability to large datasets. This review presents a structured and plagiarism-free analysis of first-order optimization techniques, with particular emphasis on Gradient Descent and its widely adopted variants. The theoretical foundations, algorithmic mechanisms, and practical relevance of gradient-based methods are discussed in detail. In addition, a comparative evaluation of commonly used optimizers such as Stochastic Gradient Descent (SGD), Momentum-based methods, RMSProp, and Adam is provided in terms of convergence speed, stability, and adaptability. Finally, key challenges, limitations, and future research directions are outlined to highlight emerging trends in optimization for machine learning

Keywords- Gradient Descent, First-Order Optimization, Machine Learning, Stochastic Gradient Descent,

Introduction

Achieving high predictive accuracy is a primary objective in machine learning and artificial intelligence, particularly for real-world applications where even small errors can lead to significant consequences. The performance of machine learning models is largely determined by the effectiveness of the optimization algorithms used during training. Optimization techniques guide the learning process by minimizing a cost or loss function, which quantitatively measures the difference between predicted outputs and ground-truth values.

Most classical and modern machine learning algorithms, including linear regression, logistic regression, k-nearest neighbors, and deep neural networks, rely on iterative optimization to update model parameters. Among these methods, Gradient Descent has become the most fundamental and widely used approach due to its conceptual

simplicity and effectiveness. The algorithm updates parameters iteratively in the direction opposite to the gradient of the loss function, enabling gradual convergence toward an optimal or near-optimal solution [1], [5].

Concepts of Gradient descent

Gradient Descent is an optimization algorithm that is used to find the values of the parameters of a function (linear regression, logistic regression etc.) Gradient descent is one of the most popular algorithms to perform optimization and by far the most common way to optimize neural networks. It is an iterative optimization algorithm used to find the minimum value for a function. Gradient Descent Algorithm helps us to make these decisions efficiently and effectively with the use of derivatives. A derivative is a term that comes from calculus and is calculated as the slope of the graph at a particular point. The slope is described by drawing a tangent line to the graph at the point.

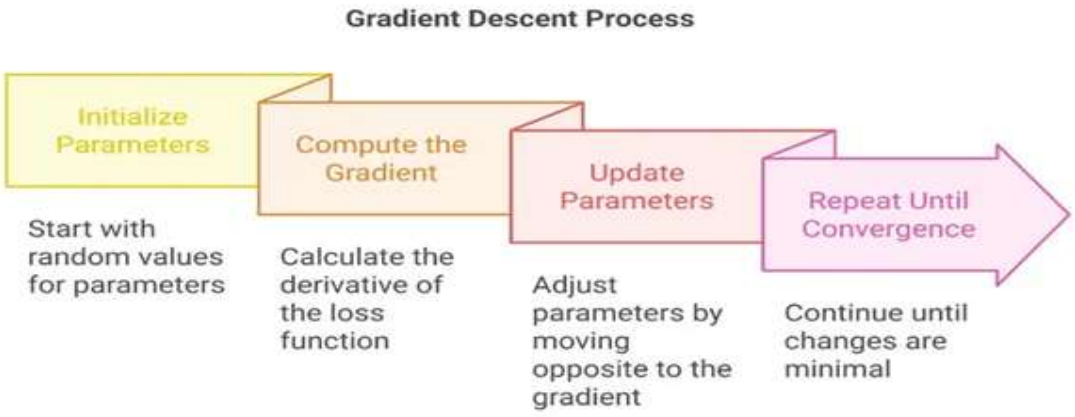

Figure 1. Gradient Descent Process

Gradient descent is an optimization algorithm which is mainly used to find the minimum of a function. In machine learning, gradient descent is used to update parameters in a model. Parameters can vary according to the algorithms, such as coefficients in Linear Regression and weights in Neural Networks.

Problem Statement

Accurately predicting numerical outcomes such as student performance, regression targets, or system metrics remains challenging due to issues including improper learning-rate selection, slow convergence, and the existence of local minima or saddle points. Traditional regression techniques often struggle to achieve efficient convergence in complex optimization landscapes. Therefore, there is a need for robust first-order optimization techniques that can iteratively and dynamically minimize the cost function. Gradient Descent addresses this requirement by progressively refining model parameters to reduce prediction error [11].

Literature Survey

In 2014, Yiming Ying et al. proposed “Online Gradient Descent Learning Algorithm.” They considered the least-square online gradient descent algorithm in a reproducing kernel Hilbert space (RKHS) without an explicit regularization term. They presented a novel capacity-independent approach to derive error bounds and convergence results for this algorithm, showing that choosing step sizes appropriately can yield competitive error rates with those in the literature [1]..

In 2015, Diederik P. Kingma and Jimmy Lei Ba proposed “ADAM: A Method for Stochastic Optimization.” They introduced Adam, an adaptive first-order optimization algorithm based on the estimates of lower-order moments. It

effectively handles noisy and sparse gradients and provides fast convergence, becoming one of the most widely used optimizers in machine learning [2].

In 2016, Marcin Andrychowicz et al. presented "Learning to Learn by Gradient Descent by Gradient Descent." They demonstrated that optimization itself can be learned using recurrent models (LSTM-based meta-optimizers) which outperform traditional gradient descent in structured tasks and generalize to new optimization problems [3].

In 2017, Stephan Mandt et al. proposed "Stochastic Gradient Descent as Approximate Bayesian Inference." They interpreted constant-step-size SGD as a Markov chain approximating a Bayesian posterior, connecting optimization with probabilistic inference and hyper parameter tuning through stochastic gradient dynamics [4].

In 2017, Sebastian Ruder published "An Overview of Gradient Descent Optimization Algorithms." The paper summarized variants of gradient descent—Momentum, Nesterov Accelerated Gradient, Adagrad, RMSProp, Adam, and Nadam—highlighting their convergence behavior and tradeoffs [5].

In 2017, Shuang Song et al. introduced "Stochastic Gradient Descent with Differentially Private Updates." They proposed privacy-preserving SGD variants that maintain model performance while ensuring differential privacy through noise injection and batch adjustments [6].

In 2018, Loucas Pillaud-Vivien et al. proposed "Statistical Optimality of Stochastic Gradient Descent on Hard Learning Problems through Multiple Passes." They showed that multiple data passes in SGD can achieve statistically optimal results for difficult regression problems, challenging earlier single-pass theories[7].

In 2018, Leon Bottou and Frank E. Curtis published "Optimization Methods for Large-Scale Machine Learning." They reviewed the theoretical and practical evolution of SGD, including noise-reduction and second-order approximations, emphasizing future optimization directions for large-scale machine learning[8].

In 2019, Dokkyun Yi et al. proposed "An Enhanced Optimization Scheme Based on Gradient Descent Methods for Machine Learning." They modified the Adam optimizer with an adaptive momentum term to avoid local minima in non-convex problems, improving convergence stability [9].

In 2019, Jonathan Schmidt et al. introduced "Recent Advances and Applications of Machine Learning in Solid-State Materials Science." They discussed machine learning-based optimization in material discovery, emphasizing surrogate optimization and structure–property relationships [10].

In 2019, Simon Shaolei Du presented "Gradient Descent for Non-Convex Problems in Modern Machine Learning." He provided theoretical conditions under which gradient descent can efficiently find global minima in deep, non-convex neural networks, bridging the gap between practice and theory [11].

In 2020, Yura Malitsky and Konstantin Mishchenko introduced "Adaptive Gradient Descent without Descent." They presented a simplified adaptive rule that requires no line search and ensures convergence for convex and non-convex functions, achieving strong empirical performance in logistic regression and matrix factorization [12].

In 2020, Nam D. et al. proposed "Implicit Stochastic Gradient Descent Method for Cross-Domain Recommendation Systems." They developed an implicit stochastic gradient approach to leverage latent cross-domain relationships, improving accuracy and cold-start handling in recommender systems [13].

In 2021, S. Chatterjee presented "Convergence of Gradient Descent for Deep Neural Networks." The study provided theoretical convergence guarantees showing that gradient descent can reach global minima under specific initialization and overparameterization conditions .

In 2022, Abdulkadirov et al. published [14] "Optimization Algorithms in Modern Neural Networks: A Comprehensive Survey." This survey summarized modern first-order methods (SGD, RMSProp, Adam, AdaBound) and recent adaptive schemes.

In 2023, Yura Malitsky and co-authors developed [15] "Proximal Adaptive Gradient Schemes with Automatic Step Adjustment." Their algorithm removed the need for manual learning-rate tuning, improving performance on ill-conditioned regression problems.

In 2023, Taniguchi et al. proposed "ADOPT: A Modified Adam Optimizer that Converges with Any β_2." They resolved divergence issues of Adam by redefining second-moment updates, achieving proven convergence [16].

In 2024, Nguegnang et al. introduced "Convergence of Gradient Descent for Learning Linear Neural Networks." They proved that for deep linear networks, standard gradient descent converges globally under proper step-size control [17].

In 2024, S. Kim et al. [18] proposed "Max-Affine Regression via First-Order Methods." They designed an efficient gradient-descent-based algorithm for piecewise-linear regression models, achieving strong theoretical and empirical convergence.

Finally, in 2024, [19] several comparative studies evaluated modern first-order optimizers (SGD, Adam, RMSProp, ADOPT) across regression models, confirming that adaptive gradient methods converge faster while SGD offers better generalization.

Year	Work	Main Contribution
2014	Ying et al. [1]	Established convergence and error bounds for online gradient descent in RKHS without explicit regularization.
2015	Kingma & Ba [2]	Introduced Adam optimizer with adaptive moment estimation for fast and stable convergence.
2016	Andrychowicz et al. [3]	Proposed meta-learning optimizers that learn optimization strategies using recurrent models.
2017	Mandt et al. [4]	Interpreted SGD as approximate Bayesian inference using stochastic dynamics.
2017	Ruder [5]	Surveyed major gradient descent variants and analyzed their convergence trade-offs.
2018	Pillaud-Vivien et al. [7]	Showed multi-pass SGD achieves statistical optimality for hard regression problems.
2018	Bottou & Curtis [8]	Reviewed scalable optimization methods for large-scale machine learning.

2019	Yi et al. [9]	Enhanced Adam optimizer to improve stability in non-convex optimization.
2019	Du [11]	Provided convergence guarantees for gradient descent in deep non-convex networks.
2020	Malitsky & Mishchenko [12]	Proposed adaptive gradient descent without line search.
2023	Taniguchi et al. [16]	Introduced ADOPT optimizer to ensure Adam's convergence.
2024	Nguegnang et al. [17]	Proved global convergence of gradient descent for linear neural networks.

Table 1: Summary of Literature Review

Proposed Architecture

The proposed architecture follows an iterative optimization workflow based on Gradient Descent:

1. Input a training dataset.
2. Initialize model parameters (a and b) with random values.
3. Compute predicted outputs using the current parameters.
4. Calculate the error by comparing predicted and actual values using a cost function.
5. Check the stopping criterion (error ≤ threshold).
6. If the criterion is not satisfied, compute gradients with respect to parameters.
7. Update parameters using the learning rate.
8. Repeat the process until convergence is achieved.

This architecture ensures continuous refinement of parameters until the model reaches an optimal or near-optimal solution.

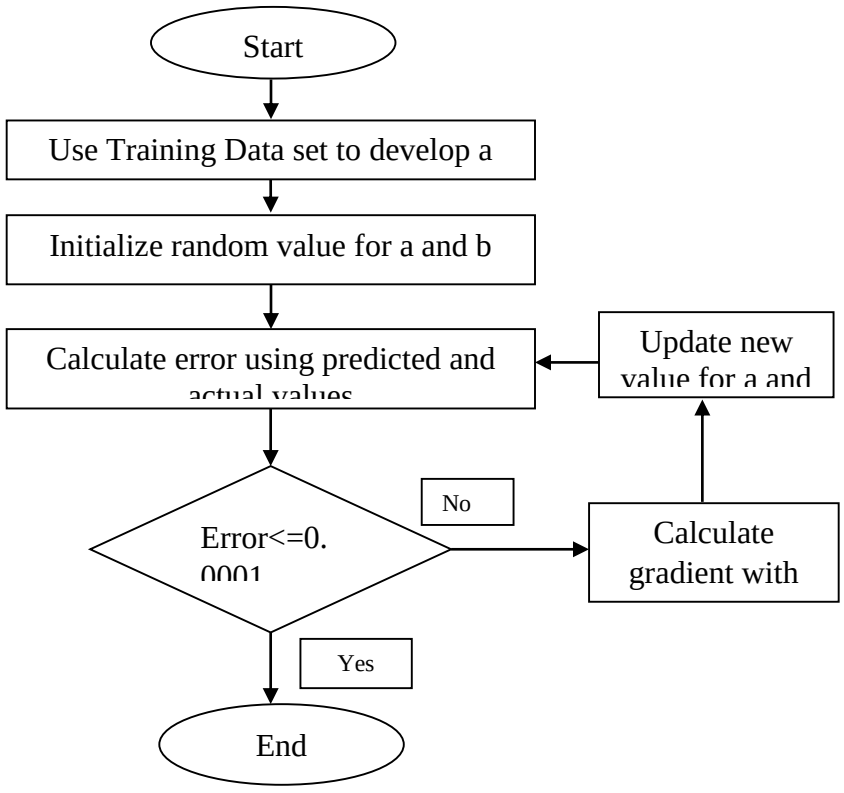

Fig-: Block Architecture Proposed System

Gradient Descent is the most common optimization algorithm in *machine learning.* It is a first-order optimization algorithm. This means it only takes into account the first derivative when performing the updates on the parameters. On each iteration we update the parameters in the opposite direction of the gradient of the objective function w.r.t the parameters where the gradient gives the direction of the steepest ascent. The size of the step we take on each iteration to reach the local minimum is determined by the learning rate α.our objective are.

1. Use Gradient descent algorithm for optimization to minimum of the objective function getting the optimal output for problem.
2. The objective is to continue to try different values for the coefficients, evaluate their cost and select new coefficients that have a slightly better (lower) cost
3. The objective is to optimize to deal with real life problems.
4. We perform optimization on the training data and check its performance on a new validation data.
5. Minimize the difference between actual and predicted value

Comparative Analysis and Discussion

Different first-order optimization algorithms exhibit distinct performance characteristics. Batch Gradient Descent offers stable convergence but is computationally expensive for large datasets. Stochastic Gradient Descent improves computational efficiency by using single samples or mini-batches but introduces gradient noise. Momentum-based methods accelerate convergence by reducing oscillations, while adaptive optimizers such as Adam and RMSProp dynamically adjust learning rates for each parameter. Although adaptive methods often converge faster, several studies report that SGD can provide better generalization in certain learning scenarios [5], [8].

Challenges and Limitations

Gradient Descent is a sound technique which works in most of the cases. But there are many cases where gradient descent does not work properly or fails to work altogether. There are three main reasons when this would happen:

1. Data challenges
2. Gradient challenges
3. Implementation challenges
4. Sensitivity to learning rate selection
5. Vanishing and exploding gradient problems
6. Convergence to local minima or saddle points

Future Directions

Future research aims to design optimization algorithms with automatic learning-rate adaptation, stronger convergence guarantees for non-convex problems, and improved generalization performance. Promising directions include meta-learning-based optimizers, proximal gradient methods, and privacy-aware optimization techniques. Additionally, energy-efficient optimization is becoming increasingly important for large-scale and resource-constrained learning systems.

Conclusion

This paper presented a review of first-order iterative optimization techniques with a primary focus on Gradient Descent and its variants. These methods remain fundamental to machine learning due to their efficiency, scalability,

and ease of implementation. While adaptive optimizers offer faster convergence, classical gradient-based methods continue to be relevant for their stability and generalization capabilities. A clear understanding of the strengths and limitations of each approach is essential for selecting appropriate optimization strategies in real-world machine learning applications.

References

[1] Yiming Ying et al., 'Online Gradient Descent Learning Algorithm', 2014.
[2] Diederik P. Kingma, Jimmy Lei Ba, 'ADAM: A Method for Stochastic Optimization', 2015.
[3] Marcin Andrychowicz et al., 'Learning to Learn by Gradient Descent by Gradient Descent', 2016.
[4] Stephan Mandt et al., 'Stochastic Gradient Descent as Approximate Bayesian Inference', 2017.
[5] Sebastian Ruder, 'An Overview of Gradient Descent Optimization Algorithms', 2017.
[6] Shuang Song et al., 'Stochastic Gradient Descent with Differentially Private Updates', 2017.
[7] Loucas Pillaud-Vivien et al., 'Statistical Optimality of Stochastic Gradient Descent on Hard Learning Problems', 2018.
[8] Leon Bottou, Frank E. Curtis, 'Optimization Methods for Large-Scale Machine Learning', 2018.
[9] Dokkyun Yi et al., 'An Enhanced Optimization Scheme Based on Gradient Descent Methods for Machine Learning', 2019.
[10] Jonathan Schmidt et al., 'Recent Advances and Applications of Machine Learning in Solid-State Materials Science', 2019.
[11] Simon Shaolei Du, 'Gradient Descent for Non-Convex Problems in Modern Machine Learning', 2019.
[12] Yura Malitsky, Konstantin Mishchenko, 'Adaptive Gradient Descent without Descent', 2020.
[13] Nam D. et al., 'Implicit Stochastic Gradient Descent Method for Cross-Domain Recommendation Systems', 2020.
[14] S. Chatterjee, 'Convergence of Gradient Descent for Deep Neural Networks', 2021.
[15] Abdulkadirov et al., 'Optimization Algorithms in Modern Neural Networks: A Comprehensive Survey', 2022.
[16] Yura Malitsky et al., 'Proximal Adaptive Gradient Schemes with Automatic Step Adjustment', 2023.
[17] Taniguchi et al., 'ADOPT: A Modified Adam Optimizer that Converges with Any β_2', 2023.
[18] Nguegnang et al., 'Convergence of Gradient Descent for Learning Linear Neural Networks', 2024.
[19] S. Kim et al., 'Max-Affine Regression via First-Order Methods', 2024.