International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 10 No. 6s,2024
https://theaspd.com/index.php

Challenges In Sustainable Non-Convex Optimization For Deep Learning Applications

Harish Kunder[1*], Manjunath Kotari[2]
[1*]Department of Artificial Intelligence and Machine Learning, Alva's Institute of Engineering and Technology, Moodbidri, VTU Belagavi, India.
[2]Department of Computer Science and Engineering, Alva's Institute of Engineering and Technology, Moodbidri and VTU Belagavi, India.
Email ID's :- researchkunder@gmail.com*, mkotari@gmail.com.

Abstract: *Today, it is a common practice to use datasets to make prediction inference in data-driven worlds. Deep learning experiments success depends on the existence and diversity of datasets that can deliver accurate results in different domain. Of these, the primary datasets (e.g., time series data) are often of spectacular efficiency. So, NP-hard problems in this setting makes it also very challenging, often resulting in non-convex solutions. To solve these problems, the critical step is to convert NP-hard problems to P problems in order to reach the best results. In this research author focus on datasets where it is hard to obtain an optimal solution, pointing out that finding a global minimum might be difficult in many situations. It also touches on the issue of how common are non-convexity problems in DC-set systems and lists some possible directions a follow-up research would take to improve them to be more friendly to convex optimization. The field can better advance deep learning applications by tackling these obstacles to make predictive analytics more accurate and efficient. This research main goal is to increase understanding of the difficulties associated with non-convex calculations in time series and real-time problem-solving scenarios.*
Keywords: *Deep learning, Artificial Intelligence, non-convex, Convex, Optimization, Local minimum.*

INTRODUCTION

In today's data-focused research, digging into datasets to find useful predictions is not just common but really important. How well deep learning experiments work depends a lot on the different and detailed datasets that are in use. These datasets are important because they strongly impact how accurate, dependable, and useful the results are in many different areas. Some datasets, like ones that track data over time or those considered primary sources, have their built-in patterns [1]. These patterns help to get really good and fitting results that make sense in different situations. NP hard problems are a complex problem in the world of data exploration. Exploring methods to transform NP-hard problems into more manageable P problems is key when addressing these formidable challenges. (Figure 1). Non-convex optimization is a fundamental aspect of deep learning methods, addressing intricate problems across diverse datasets [2]. Non-convex optimization uses functions that have multiple local minima, which makes the optimization process much more difficult than convex optimization, that employs an objective function featuring a single global minimum. Figure 2 displays a convex graph where it's easy to spot the global minimum point. Figure 3 illustrates multiple local minimum points. Figure 4 provides explanations for the terms associated with the points on the graph. In some scenarios, the learning task's natural objective is a non-convex function [3]. This often happens when training deep neural networks or working on tensor decomposition issues. Non-convex objectives and constraints are useful for accurately representing learning problems, but they're tough for algorithm designers.

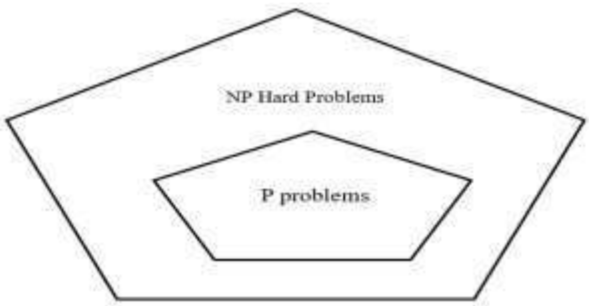

Figure 1. Relationship of NP hard and P problems.

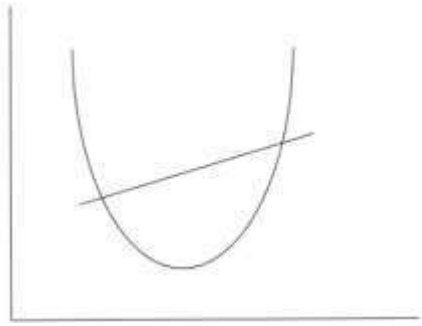

Figure 2. Convex optimization curve.

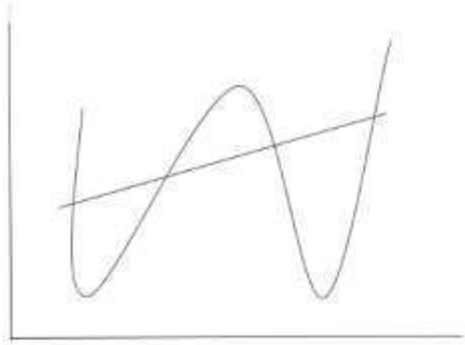

Figure 3. Non-convex optimization curve.

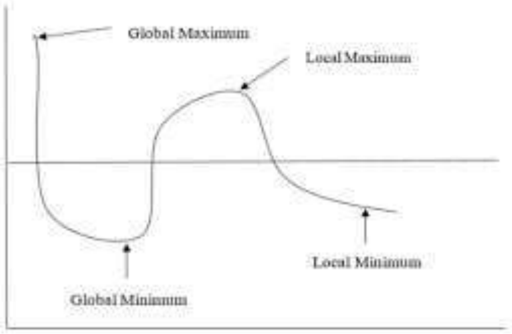

Figure 4. minimum/maximum points.

Unlike convex optimization, there isn't a ready-made toolkit for solving non-convex problems. Many of these issues are very difficult since they are known to be NP-hard. Also, some of these non-convex problems are not just hard to solve optimally, but also hard to solve approximately, which makes things even trickier. In the era of artificial intelligence, numerous real-world problems demonstrate non-convex characteristics owing to complex interactions and the nature of high-dimensional data [4]. The optimization of non-convex functions involves finding optimal solutions amidst multiple local optima, saddle points, and plateaus. The various datasets are available from the varieties of a field, and the nature of a real-time dataset provides non-convex optimization challenges. These issues come up in a number of domains, including reinforcement learning, computer vision, and natural language processing [1]. For example, maximizing the loss function a non-convex function is necessary while training deep neural networks for the computer vision application. The convergence and generalization of the model are affected when there are many minimum variables present. An additional example is: Because of the enormous dimensionality of the language, non-convexity is a difficult challenge in natural language processing. The problem is to find datasets that produce non-convex results and require additional techniques for optimization. This paper highlights the presence of non-convexity in three datasets and offers researchers insight into developing optimization methods.

A. Literature survey

Identifying intrusions within a social media network, which is a novel approach presented in [8]. The soft computing method combines fuzzy clustering, “Particle Swarm Optimization” (PSO), and “Multi-Layer Perceptron” (MLP) neural networks to solve non-convex optimization problems. This application is proven only in the financial sector. To address the economic dispatch problem in power systems, the Water Cycle Optimization (WCO) algorithm is designed [9]. WCO combines global and local search characteristics, which are inspired by the water cycle. WCO produced a precise solution for the non-convex economic dispatch issue and contrasted it with other optimization techniques already in use. This paper enhanced and proved the PSO algorithm to address non-convex challenges more effectively.

The review explains distributed learning within non-convex optimization problems [10]. This paper explored stochastic gradient descent (SGD), coordinate descent, and proximal algorithms involving batch as well as streaming data. The work is limited to proving communication efficiency, fault tolerance, and privacy protection within distributed learning. But they identified unresolved problems and potential research directions for the future. The non-convex optimization methods are applied to signal processing and machine learning applications. These developments in the new sectors are explored in the paper [11]. The optimized techniques used in these sectors are gradient descent, alternating direction methods of multipliers, and proximal gradient methods. Furthermore, this study describes the difficulties in finding global minima using non-convex optimization.

To address the non-convex optimization problems within machine learning, a novel approach utilizing second-order optimization is introduced [12]. The author used the analysis of matrix factorizing and deep learning to demonstrate the effectiveness of second-order optimization over first-order optimization. This study says that second-order optimization techniques accurately solve the non-convex problems with highly complex and nonlinear objective functions. To determine the subsequent optimal solution in multi-objective optimization challenges, a ray tracing method is developed [13]. This method generates a set of random solutions, followed by the arrangement of random numbers. Subsequently, a ray extends from the present solution toward the non-dominated solutions, selecting the next optimal solution based on the intersection point of the ray with the Pareto front. The method's performance is assessed across various benchmark problems, revealing its superiority over existing techniques concerning solution convergence and diversity. The authors propose that it could be used to solve a variety of multi-objective optimization issues.

The study outlined in reference [14] proposes a two-step approach designed to solve minimal cost flow problems that are nonlinear and non-convex. This technique blends a local search strategy with a genetic algorithm. The genetic algorithm first generates an initial population, which is then refined using a local search algorithm that is based on quasi-Newton. The algorithm's performance is assessed on benchmark tasks and contrasted with a number of cutting-edge techniques. The results show that this suggested technique outperforms other methods in terms of both computing efficiency and solution quality. As a result, the authors propose this approach as a viable remedy for comparable nonlinear non-convex optimization issues.

To tackle the non-convex optimization challenge, PSO is proposed and addresses the complexities of optimizing wellbore trajectories [15]. This paper discussed the PSO algorithm and its diverse adaptations, including hybrid PSO methodologies. This article served as evidence for researchers and industry professionals to use PSO for solving non-convex optimization problems in petroleum engineering. The non-optimization problem is also present in the field of cyber. A novel method was created to identify irregularities in "Cyber Physical Production systems" (CPPS) by capturing the geometric structure and the data using a non-convex hull. The high-dimensional data used in this approach produced a result that was not optimal [16].

The gradient-based method involves projecting the gradient onto the feasible set, which leads to the non-smoothness and non-differentiability of the objective function [17]. The focus lies on non-convex optimization problems in control systems. Conditions for the differentiability of the projected trajectory are provided, demonstrating the method's robust convergence to a stationary point under certain assumptions, elucidated through numerical examples.

Another paper, [18], presents a new method for addressing sparse multiple instance learning (MIL) problems using a non-convex penalty function. They use the ADMM algorithm to minimize this non-convex penalty function, and empirically show its advantages for sparse MIL through experiments with various benchmark datasets, which are, in terms of classification accuracy and computational complexity, more efficient than existing techniques.

The work in [19] presents an online method specifically designed for powered descent guidance with non-convex problems. Introduced as a way to steer spacecrafts towards safe landing locations under computational and sensor resources constraints, they presented a modified branch-and-bound algorithm in the paper. Numerical example results show that the proposed algorithm can accurately and efficiently guarantee that spacecrafts can arrive in safe landing sites.

Paper [20] introduces a deterministic algorithmic structure for solving the non-convex phase retrieval problem. Correct Uniqueness of the solutions give rise to a sequence of non-convex optimization problems with convex constraints is demonstrated by the authors. This leads to almost sure exact recovery of the signal in a number of measurements that is sub-linear in the column-dimension of a certain matrix of the signal model, and with the number of measurements much smaller than existing approaches. The claims are verified by way of numerical simulations.

The authors in [21] introduce a new approach for salient region detection/segmentation based on non-convex non-local reactive flows. Presenting a detailed description of the methodology, including mathematical formulation and implementation, the studied shows that the proposed approach has a good accuracy with fast computation speed compared to existing methods. Using several well-illustrated case studies, it provides a good demonstration that the new methodology works well for saliency detection and segmentation, and discusses the limitations and difficulties of the existing methods.

B. Data for analysis

- Dataset 1 [24]: The database of the "Modified National Institute of Standards and Technology" (MNIST). Every 28x28 pixel picture contains a caption and a handwritten number between 0 and 9. As an influential and widely-used benchmark, MNIST plays an important role for researchers in designing and comparing models operating on digit/pattern recognition and handwriting analysis purposes.
- Dataset 2 [25]: The 6,000 images included in the CIFAR-10 dataset are split into 10,000 images for testing and 50,000 images for training. This dataset is widely used in academic studies to investigate methods like CNNs and data augmentation, which propel improvements in performance of models and recognition of images.
- Dataset 3 [26]: M4 forecasting Dataset is a comprehensive collection of 100,000-time series data points from various domains like demographic, finance, and industry, each varying in length and frequency (yearly, quarterly, monthly, weekly, daily, hourly), created for the M4 Competition organized by the Makridakis Open Forecasting Center to advance time series forecasting.

METHOD

Figure 5 illustrates the utilization of various datasets that were modelled to assess the attainment of local minimum results. The following optimizers are considered.

- “Adaptive Moment Estimation” (ADAM)
- SGD
- “Root Mean Square Propagation” (RMSPROP)

All the above-mentioned optimizers are tested under the following learning rate,

- 0.001
- 0.01
- 0.1

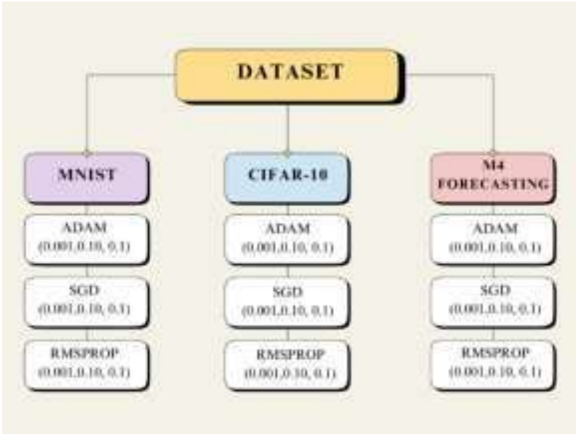

Figure 5. Block diagram for Analysis.

Algorithm 1: Visualizing Loss Landscape with CNN

1. Load the MNIST dataset and normalize pixel values.
2. Define a CNN model architecture.
3. Train the CNN model multiple times, collecting loss values.
4. Plot loss curves for each training run.
5. Apply PCA to reduce loss values to 2 dimensions.
6. Visualize the 2D PCA projection of the loss landscape.

Algorithm 1, based upon the CNN idea, is used with two datasets: CIFAR-10 and MNIST. The M4 forecasting dataset, a time series dataset, is used with deep learning, as shown in Algorithm 2. To visualize local minima, the graph is reduced from high dimensions to low dimensions using Algorithm 3.

Algorithm 2: Visualizing Loss Landscape with CNN

1. Incorporate in data processing, model-building, and visualization libraries..
2. Load dataset (e.g., M4 time series subset).
3. Normalize data using MinMaxScaler.
4. Create sequences from time series for LSTM input.
5. Split the dataset into train and test sets.
6. Reshape data to fit the LSTM model input.
7. Define an LSTM model with dynamic optimizer and learning rate parameters.
8. Train the model multiple times with different optimizers (Adam, SGD, RMSProp) and learning rates.
9. Store loss values after each training run.
10. Apply PCA to reduce the loss data to 2 dimensions.
11. Visualize PCA results for each optimizer and learning rate using scatter plots.

Algorithm 3: Principal Component Analysis (PCA)

Input: Dataset X
Output: Reduced dimensionality representation of X

1. To normalize X, divide by the standard deviation after eliminating the mean.
2. Determine the standardized X covariance matrix.
3. Apply an Eigen disintegration to the matrix of covariance.
4. The top k eigenvectors that match the largest eigenvalues should be selected.

5. Convert the standardized dataset to the fresh subspace that the chosen eigenvectors have generated.
6. Output reduced dimensionality representation of the dataset.

RESULT AND DISCUSSION

The outcomes are organized into three sections based on the dataset used. These findings demonstrate the presence of non-convex outcomes by revealing multiple local minima on the use of various optimizers and learning rates.

A. Result on the MNIST dataset.

When we apply a CNN model to find the correct handwriting text from the available image set, we end up with multiple local minima. ADAM, SGD, and RMSProp consisted the three optimizers used to analyze the MNIST dataset. They were each examined with three distinct learning rates: 0.001, 0.01, and 0.1. To visualize the behaviour of the optimizers and the loss landscapes, PCA was applied. The PCA reduces the dimensionality of the loss values into two components. This technique helped identify local minima across different configurations. Across the nine 2D PCA projections, multiple local minima were observed for each optimizer at different learning rates, indicating that the optimization process does not converge smoothly to a single minimum. Instead, the landscape is characterized by a non-convex nature, where the model's loss fluctuates through various minima before potentially settling into a region of lower loss.

- ADAM (Figure 6 to Figure 8): At lower learning rates (0.001), the model shows more concentrated and stable local minima, suggesting efficient convergence. As the learning rate increases to 0.01 and 0.1, the dispersion of local minima increases, indicating more exploration of the loss landscape and greater difficulty in finding stable solutions.
- SGD (Figure 9 to Figure 11): Across all learning rates, SGD exhibits a more scattered pattern of local minima. At the highest learning rate (0.1), the optimizer shows significant fluctuations, suggesting instability and an inability to settle into deeper minima consistently.
- RMSProp (Figure 12 to Figure 14): Similar to ADAM, RMSProp demonstrates stable local minima at lower learning rates, with a more moderate dispersion as the learning rate increases. This indicates that RMSProp is also effective in navigating the loss landscape, though not as aggressive in exploring the surface as SGD at higher learning rates.

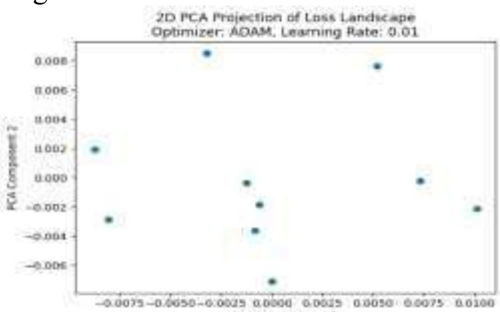

Figure 6. ADAM with 0.01 learning rate on the MNIST dataset.

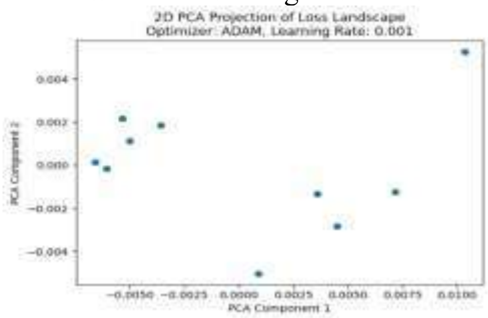

Figure 7. ADAM with 0.001 learning rate on the MNIST dataset.

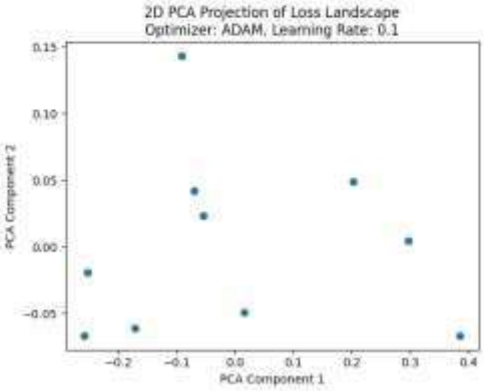

Figure 8. ADAM with 0.1 learning rate on the MNIST dataset.

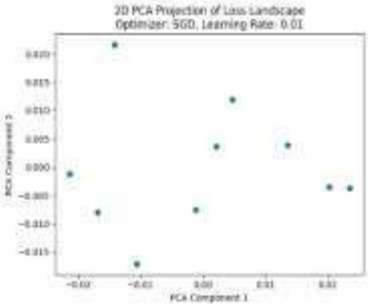

Figure 9. SGD with 0.01 learning rate on the MNIST dataset.

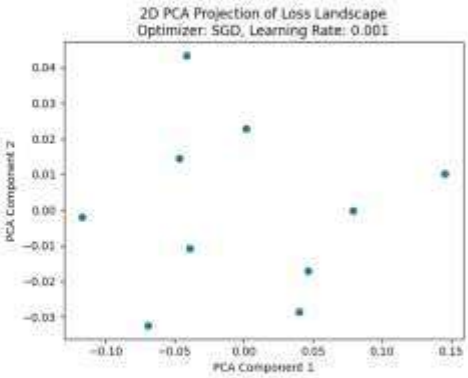

Figure 10. SGD with 0.001 learning rate on the MNIST dataset.

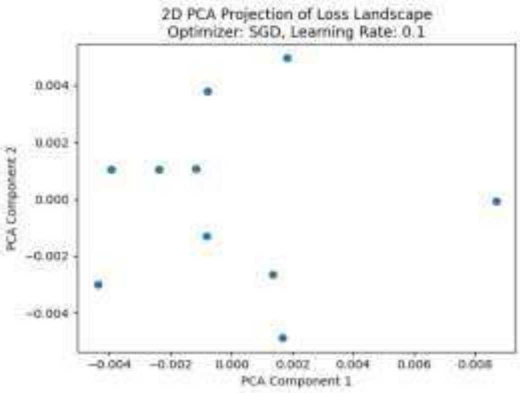

Figure 11. SGD with 0.1 learning rate on the MNIST dataset.

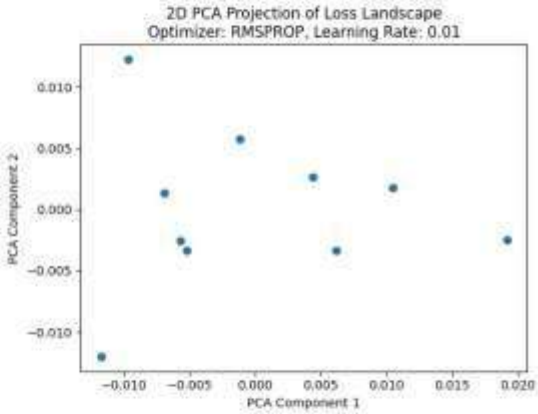

Figure 12. RMSProp with 0.01 learning rate on the MNIST dataset.

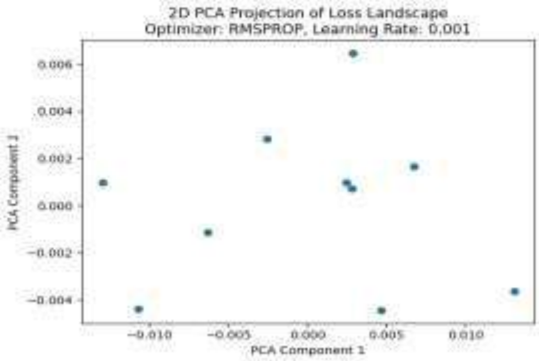

Figure 13. RMSProp with 0.001 learning rate on the MNIST dataset.

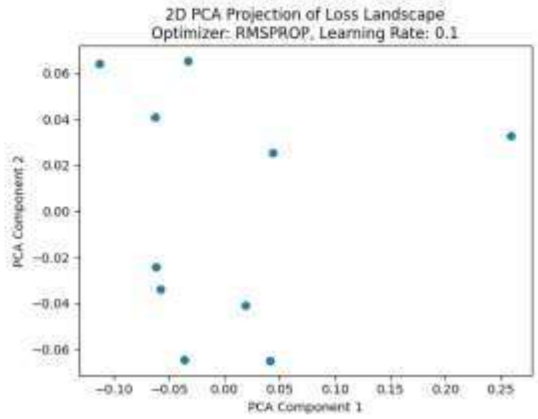

Figure 14. RMSProp with 0.1 learning rate on the MNIST dataset.

B. Result on CIFAR-10

The CIFAR-10 dataset was evaluated using three optimizers: ADAM, SGD, and RMSProp, each tested with three learning rates: 0.001, 0.01, and 0.1. To gain insight into the behaviour of the optimizers and the underlying loss surfaces, PCA was employed to project the high-dimensional loss values into two principal components. In the resulting nine 2D PCA projections, multiple local minima were observed across different optimizers and learning rates. This indicates that the optimization process does not follow a smooth path to a single global minimum but instead navigates through a non-convex surface characterized by numerous local minima. The presence of these multiple minima demonstrates the complexity of optimizing deep learning models on the CIFAR-10 dataset.

• ADAM (Figure 15 to Figure 17): At the lowest learning rate (0.001), the optimizer displays concentrated and well-formed local minima, suggesting efficient convergence with minimal fluctuations. However, as the learning rate increases to 0.01 and 0.1, the optimizer's trajectory becomes more dispersed, with wider fluctuations and more scattered local minima, indicating a greater exploration of the loss surface and challenges in consistently finding stable minima at higher learning rates.

- SGD (Figure 18 to Figure 20): Across all learning rates, SGD shows a more varied and scattered distribution of local minima, particularly at the highest learning rate (0.1). This indicates that SGD can become unstable with high learning rates as evidenced by high fluctuations and getting stuck at non-deep minima. For low learning rates, the SGD has gradual behavior, although, still has much fluctuation in local minima, due to its challenging to navigate the non-convex space model of CIFAR-10.
- RMSProp (Figure 21 to Figure 23): Similar to ADAM, RMSProp demonstrates more concentrated and stable local minima at lower learning rates (0.001), but its behavior becomes more dispersed at higher learning rates. However, compared to SGD, RMSProp maintains a better balance between exploration and stability, with less erratic behavior, especially at intermediate learning rates (0.01). This indicates that RMSProp is able to handle the non-convexity of the CIFAR-10 loss surface without an overly high level of instability.

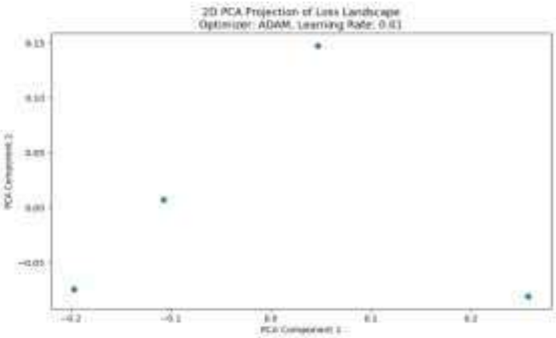

Figure 15. ADAM with 0.01 learning rate on the CIFAR-10 dataset.

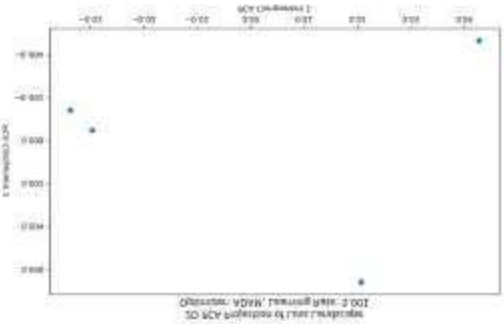

Figure 16. ADAM with 0.001 learning rate on the CIFAR-10 dataset.

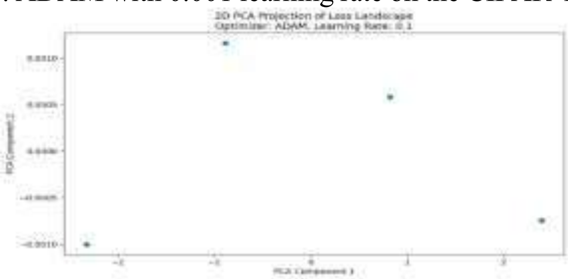

Figure 17. ADAM with 0.1 learning rate on the CIFAR-10 dataset.

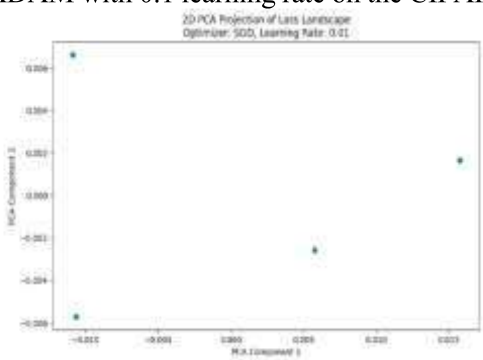

Figure 18. SGD with 0.01 learning rate on the CIFAR-10 dataset.

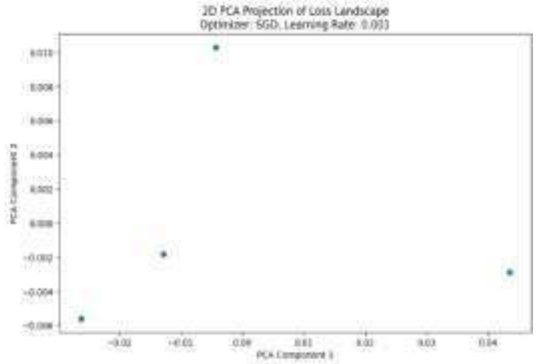

Figure 19. SGD with 0.001 learning rate on the CIFAR-10 dataset.

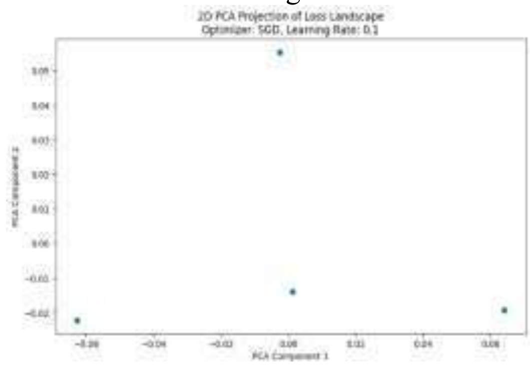

Figure 20. SGD with 0.1 learning rate on the CIFAR-10 dataset.

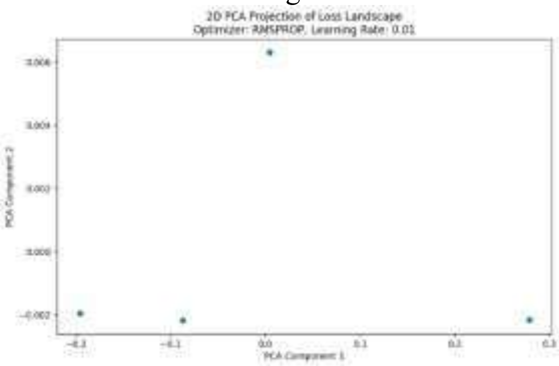

Figure 21. RMSProp with 0.01 learning rate on the CIFAR-10 dataset.

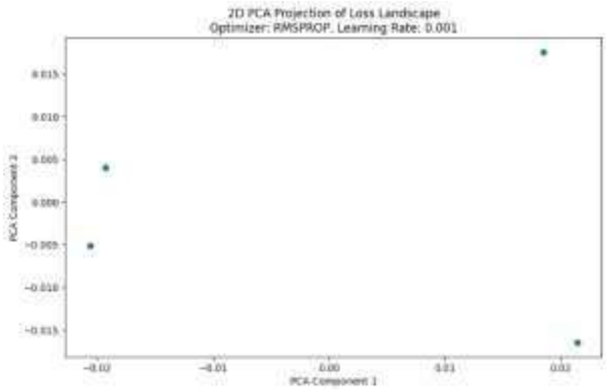

Figure 22. RMSProp with 0.001 learning rate on the CIFAR-10 dataset.

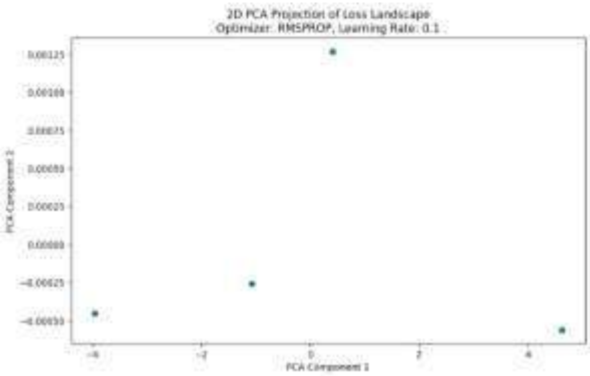

Figure 23. RMSProp with 0.1 learning rate on the CIFAR-10 dataset.

C. Result on M4 forecasting.

Three optimizers ADAM, SGD, and RMSProp were used to calculate the M4 statistics dataset having learning rates of 0.001, 0.01, and 0.1. To gain an insight into the optimization dynamics and the landscape of the loss, PCA was used to map the high-dimensional loss value to 2 principle components. This approach enabled a clearer view of the local minima that emerged during the optimization process for each configuration. Across the nine 2D PCA projections, multiple local minima were identified for all three optimizers at the different learning rates. In the M4 forecasting problem, the optimization process traverses a landscape with several local minima rather than convergently to a global minimum, confirming the non-convex character of the loss surface.

- ADAM (Figures 24 to 26): At the smallest learning rate (0.001), ADAM shows tightly clustered local minima, indicating efficient convergence with limited variation. However, as the learning rate increases to 0.01 and 0.1, the optimizer explores a wider area, with more scattered local minima and greater fluctuations. This suggests that higher learning rates lead ADAM to explore the loss surface more extensively, but may also increase instability in finding stable solutions.
- SGD (Figures 27 to 29): At all learning rates, SGD displays a more dispersed and unstable pattern of local minima. Particularly at the highest learning rate (0.1), the optimizer exhibits significant fluctuations, making it difficult to settle into deeper minima consistently. Even at lower learning rates, SGD's behavior remains less stable compared to ADAM and RMSProp, indicating the challenges it faces in navigating the complex loss landscape of the M4 forecasting dataset.
- RMSProp (Figures 30 to 32): At a low learning rate (0.001), RMSProp shows stable and concentrated local minima, similar to ADAM. As the learning rate increases, the optimizer exhibits moderate dispersion, though it still maintains better stability than SGD. At higher learning rates, RMSProp explores a wider area of the loss surface, but remains more consistent than SGD, striking a balance between exploration and convergence.

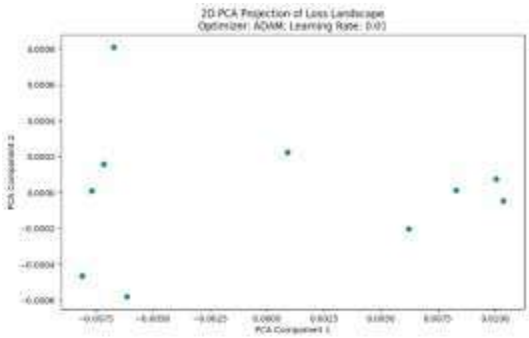

Figure 24. ADAM with 0.01 learning rate on the M4 Forecasting dataset.

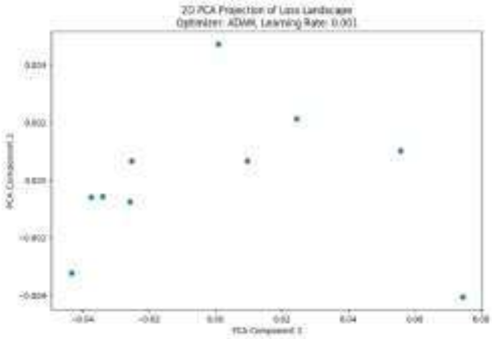

Figure 25. ADAM with 0.001 learning rate on the M4 Forecasting dataset.

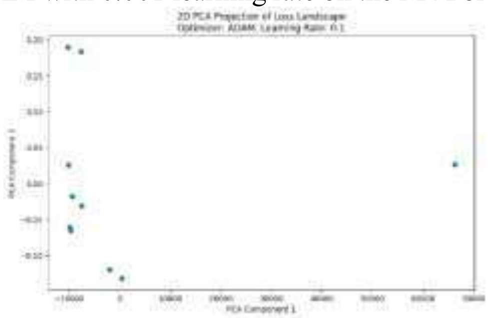

Figure 26. ADAM with 0.1 learning rate on the M4 Forecasting dataset.

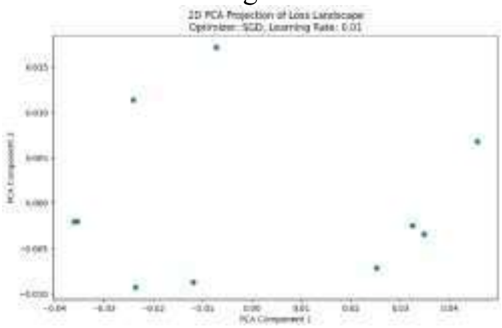

Figure 27. SGD with 0.01 learning rate on the M4 Forecasting dataset.

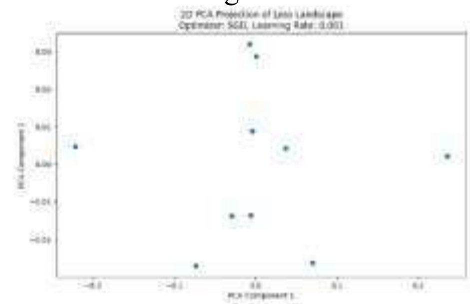

Figure 28. SGD with 0.001 learning rate on the M4 Forecasting dataset.

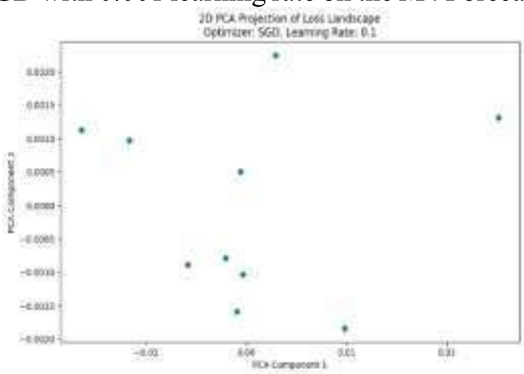

Figure 29. SGD with 0.1 learning rate on the M4 Forecasting dataset.

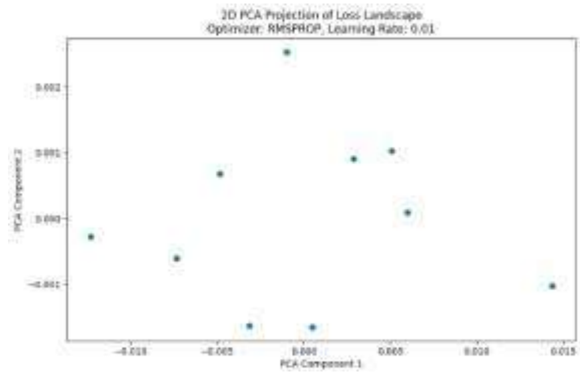

Figure 30. RMSProp with 0.01 learning rate on the M4 Forecasting dataset.

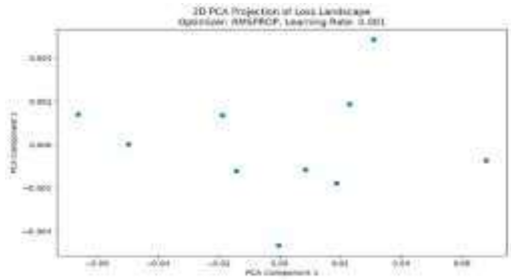

Figure 31. RMSProp with 0.001 learning rate on the M4 Forecasting dataset.

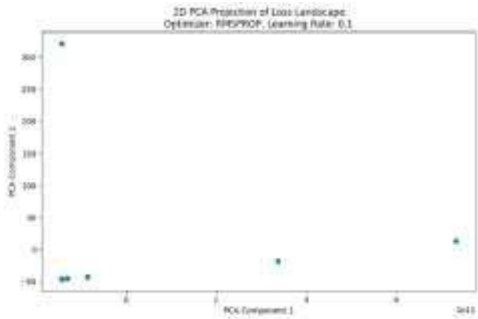

Figure 32. RMSProp with 0.1 learning rate on the M4 Forecasting dataset.

CONCLUSION

Real-time data sets are integral to numerous applications, offering valuable insights and solutions to complex problems. However, they often pose non-convex optimization challenges, characterized by the presence of multiple local minima, necessitating the selection of a global minimum for optimal results. This paper undertakes the task of substantiating the existence of non-convex solution spaces within real-time datasets. To achieve this, a diverse range of datasets from deep learning domains is examined, collectively illustrating the pervasive nature of non-convexity across various data types and learning scenarios. Exploring these non-convex METPS presents a significant challenge, especially considering the limitations of classic optimization methods. Even with the best optimizers such as ADAM, SGD and RMSProp, it is difficult to reach micro-optimal solutions, which shows that traditional techniques struggle to deal with dataset optimizations that are performed simultaneously by end users. Therefore, specialized optimization techniques designed to deal with non-convex scenes become necessary to achieve an efficient optimization for real-time data sets. In the end, we want to minimize the loss function and find the global minimum. The optimization process of any method attempts to find its optimal solution to solve the problem, it depends on the dataset and the hyper-parameters of the optimal optimization method.

REFERENCES

[1]. Jain, Prateek, and Purushottam Kar. "Non-convex optimization for machine learning." arXiv preprint arXiv:1712.07897 (2017).
[2]. Alomoush, Muwaffaq I. "Optimal Combined Heat and Power Economic Dispatch Using Stochastic Fractal Search Algorithm." Journal of Modern Power Systems and Clean Energy 8, no. 2 (2020): 276-286.
[3]. Yonel, Bariscan, and Birsen Yazici. "A Deterministic Convergence Framework for Exact Non-Convex Phase Retrieval." arXiv preprint arXiv:2001.02855 (2020).
[4]. Galiano, G., I. Ramírez, and E. Schiavi. "Non-convex non-local reactive flows for saliency detection and segmentation." Journal of Computational and Applied Mathematics (2020): 112873.
[5]. Hauswirth, Adrian, Florian Dörfler, and Andrew Teel. "On the Differentiability of Projected Trajectories and the Robust Convergence of Non-convex Anti-Windup Gradient Flows." IEEE Control Systems Letters (2020).
[6]. Zhang, Yuqi, Haibin Zhang, and Yingjie Tian. "Sparse multiple instance learning with non-convex penalty." Neurocomputing (2020).
[7]. Reynolds, Taylor, Danylo Malyuta, Mehran Mesbahi, Behcet Acikmese, and John M. Carson. "A Real-Time Algorithm for NonConvex Powered Descent Guidance." In AIAA Scitech 2020 Forum, p. 0844. 2020.
[8]. Sathesh, A. "Enhanced Soft Computing Approaches for Intrusion Detection Schemes in Social Media Networks." Journal of Soft Computing Paradigm (JSCP) 1, no. 02 (2019): 69-79.
[9]. Chaudhary, Vishal, Hari Mohan Dubey, and Manjaree Pandit. "Solution of Non-convex Economic Dispatch Problems by Water Cycle Optimization." In Advanced Engineering Optimization Through Intelligent Techniques, pp. 819-829. Springer, Singapore, 2020.
[10]. Chang, Tsung-Hui, Mingyi Hong, Hoi-To Wai, Xinwei Zhang, and Songtao Lu. "Distributed Learning in the Non-Convex World: From Batch to Streaming Data, and Beyond." arXiv preprint arXiv:2001.04786 (2020).
[11]. So, Anthony Man-Cho, Prateek Jain, Wing-Kin Ma, and Gesualdo Scutari. "Non-Convex Optimization for Signal Processing and Machine Learning." (2020).
[12]. Xu, Peng, Fred Roosta, and Michael W. Mahoney. "Second-order optimization for non-convex machine learning: An empirical study." In Proceedings of the 2020 SIAM International Conference on Data Mining, pp. 199-207. Society for Industrial and Applied Mathematics, 2020.
[13]. Nowak, Dimitri, and Karl-Heinz Küfer. "A Ray Tracing Technique for the Navigation on a Non-convex Pareto Front." arXiv preprint arXiv:2001.03634 (2020).
[14]. Ghasemishabankareh, Behrooz, Melih Ozlen, Xiaodong Li, and Kalyanmoy Deb. "A genetic algorithm with local search for solving single-source single-sink nonlinear non-convex minimum cost flow problems." Soft Computing 24, no. 2 (2020): 1153-1169.
[15]. Biswas, Kallol, Pandian M. Vasant, Moacyr Batholomeu Laruccia, José Antonio Gámez Vintaned, and Myo M. Myint. "Review on Particle Swarm Optimization Approach for Optimizing Wellbore Trajectory." In Deep Learning Techniques and Optimization Strategies in Big Data Analytics, pp. 290-307. IGI Global, 2020.
[16]. Li, Peng, and Oliver Niggemann. "Non-convex hull based anomaly detection in CPPS." Engineering Applications of Artificial Intelligence 87 (2020): 103301
[17]. Hauswirth, Adrian, Florian Dörfler, and Andrew Teel. "On the Differentiability of Projected Trajectories and the Robust Convergence of Non-convex Anti-Windup Gradient Flows." IEEE Control Systems Letters (2020).
[18]. Zhang, Yuqi, Haibin Zhang, and Yingjie Tian. "Sparse multiple instance learning with non-convex penalty." Neurocomputing (2020).
[19]. Reynolds, Taylor, Danylo Malyuta, Mehran Mesbahi, Behcet Acikmese, and John M. Carson. "A Real-Time Algorithm for NonConvex Powered Descent Guidance." In AIAA Scitech 2020 Forum, p. 0844. 2020.
[20]. Yonel, Bariscan, and Birsen Yazici. "A Deterministic Convergence Framework for Exact Non-Convex Phase Retrieval." arXiv preprint arXiv:2001.02855 (2020).
[21]. Galiano, G., I. Ramírez, and E. Schiavi. "Non-convex non-local reactive flows for saliency detection and segmentation." Journal of Computational and Applied Mathematics (2020): 112873.
[22]. Tianyi Lin, Chi Jin and Michael I Jordan, "On gradient descent ascent for nonconvex-concave minimax problems", 2019.
[23]. Robert Mansel Gower, Nicolas Loizou, Xun Qian, AlibekSailanbayev, Egor Shulgin and Peter Richtárik, "Sgd: General analysis and improved rates", International Conference on Machine Learning, pp. 5200-5209, 2019.
[24]. Online available at https://www.kaggle.com/datasets/hojjatk/mnist-dataset
[25]. Online available at https://www.kaggle.com/c/cifar-10/
[26]. Online available at https://www.kaggle.com/datasets/yogesh94/m4-forecasting-competition-dataset/code