Optimization in Machine Learning
An Annotated Bibliography

Danel Levi Bacarra (dlbacarra@up.edu.ph) Marfred James Deen (mpdeen@up.edu.ph) Arwin Delasan (avdelasan@up.edu.ph) Jed Edison Donaire (jjdonaire@up.edu.ph) Princess Mae Parages (pbparages@up.edu.ph) Sheldon Arthur Sagrado (smsagrado@up.edu.ph) James Torres (jtorres1@up.edu.ph)

21 October 2024

References

[1] J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization.” *Journal of machine learning research*, vol. 12, no. 7, 2011.

> The paper ”Adaptive Subgradient Methods for Online Learning and Stochastic Optimization” introduces a family of adaptive subgradient methods that dynamically adjust learning rates based on the observed geometry of the data. This allows the algorithms to focus on rare but highly informative features, improving predictive performance. The authors provide efficient algorithms for risk minimization problems and demonstrate through experiments that adaptive methods, such as ADAGRAD, outperform traditional non-adaptive methods. The paper also explores the use of block-diagonal matrices to further refine learning while maintaining computational efficiency.
>
> The paper is highly relevant to the study ”A Survey of Optimization Methods from a Machine Learning Perspective” because both focus on optimization techniques crucial for machine learning. The adaptive subgradient methods introduced in the paper address challenges such as high-dimensional data

and rare, informative features, which are essential in modern machine learning. Furthermore, optimization is a key theme in both works, as efficient methods directly impact the performance of machine learning algorithms in tasks like risk minimization and regularization.

[2] P. Jain and P. Kar, “Non-convex optimization for machine learning,” *Foundations and Trends in Machine Learning*, vol. 10, no. 3-4, pp. 142–363, 2017. [Online]. Available: http://dx.doi.org/10.1561/2200000058

The summary of ”Non-convex Optimization for Machine Learning” by Prateek Jain and Purushottam Kar presents an in-depth exploration of non-convex optimization techniques in machine learning, covering theory, algorithms, and applications. The authors emphasize the advantages of directly solving non-convex problems instead of relaxing them into convex ones, which often leads to faster and more scalable solutions. Key methods like projected gradient descent, alternating minimization, and the EM algorithm are discussed alongside their use cases in sparse recovery, matrix completion, and robust regression. These methods, despite the inherent NP-hardness of non-convex problems, have shown significant promise when applied to real-world tasks, particularly when these problems possess favorable structure.

The monograph is structured in three parts: introduction to non-convex optimization, core algorithms, and practical applications. Part I covers mathematical tools and basic concepts, while Part II delves into specific algorithms like non-convex projected gradient descent and stochastic optimization. Part III focuses on the practical applications of these methods in machine learning, such as low-rank matrix recovery and phase retrieval. The authors provide a comprehensive introduction to the theoretical underpinnings of non-convex optimization, making this work valuable for researchers and practitioners alike, particularly those seeking to solve complex problems in high-dimensional spaces.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” vol. 25, 2012.

In this groundbreaking paper, Krizhevsky et al. present a deep convolutional neural network architecture, commonly known as AlexNet, which dramatically improved the state of the art in image classification tasks. Trained on the large-scale ImageNet dataset, AlexNet won the 2012 ImageNet Large Scale Visual Recognition Challenge by a significant margin. Key innovations introduced include the use of Rectified Linear Units (ReLUs) to accelerate training convergence, GPU utilization for handling extensive computations, and techniques like dropout and data augmentation to mitigate overfitting. This work demonstrated the practical feasibility and superior performance of deep learning models in complex computer vision tasks, catalyzing a wave of research and application in neural networks.

The reference is directly related to Sun et al.'s ”A Survey of Optimization Methods from a Machine Learning Perspective” as it exemplifies the challenges and solutions in optimizing deep neural networks. Krizhevsky et al.'s work highlights critical optimization strategies—such as handling vanishing gradients with ReLUs and preventing overfitting with dropout—that are essential topics in the survey. By discussing the optimization methods employed in AlexNet, Sun et al. provide insights into practical approaches that have shaped modern machine learning. This connection underscores the importance of optimization techniques in advancing deep learning and supports the survey's exploration of optimization from both theoretical and practical perspectives.

[4] Y. Li, “Deep reinforcement learning: An overview,” *CoRR*, vol. abs/1701.07274, 2017. [Online]. Available: http://arxiv.org/abs/1701.07274

This study explains how deep reinforcement learning combines two powerful techniques: deep learning and reinforcement learning. Deep learning is a method where machines learn to recognize patterns, while reinforcement learning teaches machines to make decisions by interacting with their environment. Together, these methods allow machines to learn from complex data like images or videos to make better decisions. The study also highlights the success of this combination in games like Atari, where machines learned to play by observing the game. The authors also discuss future challenges, such as improving these techniques for real-world applications like robotics.

This referenced study is highly relevant to the study on derivative-free optimization because both explore advanced methods for solving complex problems. Deep reinforcement learning (DRL) focuses on improving decision-making processes by learning optimal strategies through interactions, while this addresses optimization without using gradient information, which is often impractical in real-world problems. DRL's ability to handle high-dimensional data and complex environments is particularly useful when optimization problems lack clear mathematical formulations. The overview study demonstrates how neural networks can model complex input spaces, a capability that can complement derivative-free methods by offering alternative ways to represent and explore solution spaces. Combining DRL with derivative-free optimization could enhance problem-solving capabilities in scenarios where traditional optimization fails, making this a valuable foundation for expanding upon the techniques in the new research.

[5] J. Martens *et al.*, "Deep learning via hessian-free optimization." in *Icml*, vol. 27, 2010, pp. 735–742.

In this paper, Martens discusses the challenges of training deep auto-encoders using standard first-order optimization methods, like gradient descent. He highlights how these methods often struggle with issues such as under-fitting and problematic curvature in the objective functions, which can hinder effective learning. To tackle these problems, Martens introduces a new approach called Hessian-free (HF) optimization. This method utilizes a sparse initialization strategy, which limits the number of non-zero incoming connection weights for each unit in the network. By doing so, it prevents saturation of the units, allowing them to work more effectively and differentiate better from one another. This is important for deep learning tasks, where clear distinctions between units are crucial for success.

Martens supports his claims with empirical results showing that HF optimization outperforms traditional methods on various datasets, including synthetic curves and MNIST. His findings reveal that HF can achieve lower training errors without requiring pre-training, which is often necessary with other methods. This suggests that HF optimization can efficiently handle more complex architectures, such as recurrent neural networks and asymmetric auto-encoders. Overall, Martens' study provides important insights into optimizing deep learning models, showing that efficient learning algorithms can significantly improve the training of deep neural networks.

[6] J. Pajarinen, H. L. Thai, R. Akrour, J. Peters, and G. Neumann, "Compatible natural gradient policy search," *Machine Learning*, vol. 108, pp. 1443–1466, 2019.

The paper introduced a few novel key concepts in optimizing policy search for reinforcement learning. These concepts were (A) the equivalence between natural gradient and trust region optimization, and (B) the compatible policy search (COPUS) which utilized that equivalence. Concept A was possible due to natural parameterization of standard exponential policy distributions (e.g. Boltzmann, Gaussian). Moreover, to cover the case in complex models (e.g. neural networks), Concept A was combined with compatible value function approximation. Finally, Concept B was the product of the realization in concept A which produced satisfying results equal to the current state-of-the-art policy search algorithms in the paper's experiment. The experiment was running the algorithm in OpenAI's roboschool.
The findings of this paper brought about a big optimization win in simplifying the update loop and the reduction of the runtime for policy search and even further optimizations can be achieved in the cases where there are discrete actions. Furthermore, the parameterization and approximation methods applied have little to no effect on COPUS's performance in the experiment. Further works can leverage the realizations/concepts of the paper for a myriad of reasons (e.g. added optimization step in building models, applying concept A on other optimization problems). However, the only qualm

one can have on this paper is the sheer amount of prerequisites the reader is assumed to have.

[7] L. M. Rios and N. V. Sahinidis, “Derivative-free optimization: a review of algorithms and comparison of software implementations,” *Journal of Global Optimization*, vol. 56, no. 3, p. 1247–1293, July 2012. [Online]. Available: http://dx.doi.org/10.1007/s10898-012-9951-y

This paper reviews derivative-free optimization (DFO) algorithms, focusing on their use in solving bound-constrained optimization problems without relying on derivatives. The authors create a large scale test set comprising 502 problems, and evaluate 22 software implementations of the 22 algorithms into direct search methods and surrogate model based methods. The study also compares DFO solvers with the global solvers TOMLAB/MULTIMIND and TOMLAB/LGO for problems of higher dimension and non-convexities. This paper is of particular use to researchers and practitioners wishing to gain a systematic and comprehensive overview of the DFO methods. It is a fundamental resource for the field of optimization and particularly for problems where derivative information is not readily available or reliable. Extensive computational results are given on the basis of problem characteristics to offer a practical guide to the selection of an appropriate DFO solver. Derivative-free optimization (DFO) is a highly relevant topic of study to the survey of optimization methods of machine learning since it is to address the key challenges that arise from optimizing complex, non-smooth, or noisy functions when gradient based methods fail. In practice, high dimensional, noisy, or black-box functions are often encountered in machine learning algorithms such as deep neural networks and reinforcement learning for which obtaining derivatives is usually very difficult or even impossible to be computed directly. In such contexts it is especially useful that DFO provides an alternative focusing solely on objective function evaluations, without relying on gradients. Evaluation of DFO algorithms in large scale, nonconvex problems supports the application to machine learning optimization where successful algorithms need to achieve a good balance between computational efficiency and performance. DFO methods provide valuable tools for optimization in machine learning when traditional first or second order methods cannot be employed, as the survey shows how important optimization is in ML.