Research | October 01, 2020

The Connection Between Applied Mathematics and Deep Learning

By Manuchehr Aminian

In recent years, deep learning (DL) has inspired a myriad of advances within the scientific computing community. This subset of artificial intelligence relies on multiple components of applied mathematics, but what type of relationship do applied mathematicians have with DL? This question was the subject of a plenary talk by Yann LeCun (Facebook and New York University) at the virtual 2020 SIAM Conference on Mathematics of Data Science, which took place earlier this year. LeCun provided a brief history of machine learning (ML), highlighted the mathematical underpinnings of the field, presented both his vision and several broad open questions for ML's future, and discussed applied math's current relation and potential impending contributions. A 2018 *SIAM News* article by Gilbert Strang, entitled "The Functions of Deep Learning," offers an introduction for those who are unfamiliar with neural networks, ML, and DL.

Stochastic Gradient Descent

LeCun immediately identified applied mathematicians' most fundamental connection to DL: gradient descent and optimization. The search for an optimal set of parameters for a nonlinear function—with the goal of succeeding in a practical task, such as classifying images or predicting text—comprises the heart of DL. Researchers use a special form of gradient descent to find this optimal set of parameters.

Applied mathematicians often encounter gradient descent in numerical linear algebra when they seek approximate solutions of a square linear system of equations $Ax = b$. Finding a solution x^* is equivalent to finding a minimum of the function $||Ax - b||_2^2$ over all choices of x, beginning with some initial guess x_0. Researchers understand gradient descent as a process wherein they "walk down the mountain" by going in the steepest direction at every step. These actions produce a sequence of approximations x_i that is guaranteed to converge to a unique minimum for symmetric positive definite A. This is admittedly a very special class of functions that allows users to take the theory quite far, which is why it is taught in the classroom.

In contrast, the functions that one must minimize in DL—known as "loss functions"—are typically nonlinear and nonconvex, which makes theoretical guarantees much more challenging. Nevertheless, practitioners utilize gradient-based approaches and typically employ a modified version called stochastic gradient descent (see Figure 1). This stochastic component relates to the loss function's evaluation; rather than using all training data to evaluate the loss, one uses a randomly selected subset of data on each iteration of gradient descent. LeCun refers to this as "walking down the mountain in a fog," wherein each sample provides a noisy estimate of the direction. This stochastic component has found enormous practical success. "Nobody even considers anything else," LeCun said. However, opportunities still exist for researchers to provide theory that explains this success.

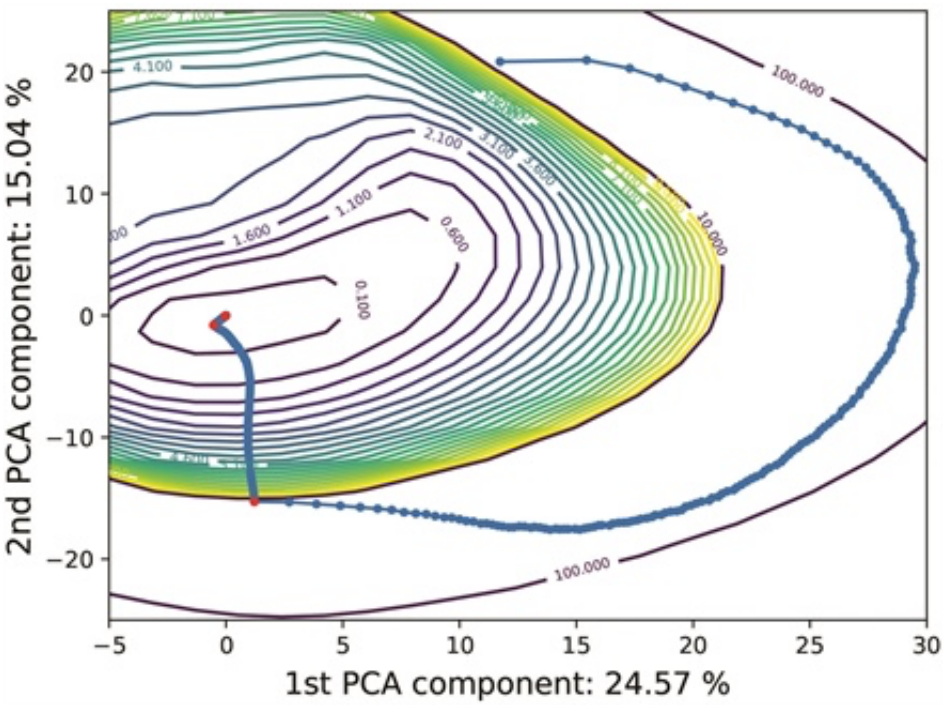

Figure 1. Visualization of the approximate loss surface when applying stochastic gradient descent to an image dataset. Figure courtesy of Tom Goldstein.

Overparameterization and Deep Nets

Turning to areas that are open to theoretical discovery, LeCun first highlighted a phenomenon that is contrary to traditional mathematical and statistical intuition: understanding overparameterized models in DL. Mathematician John von Neumann famously said, "*With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.*" This sentiment reflects a common attitude among mathematical modelers and suggests that one should be mindful of both the number of parameters in a mathematical model and the conclusions that result from fitting the model to data. Mathematicians are familiar with the perils of overparameterization from fitting high-degree polynomials. Similarly, parameter fitting and identifiability are common issues in differential equation models, especially when these models have many interacting components.

However, years of practice in DL reveal a different picture. Consider two neural networks with the same goal. The network with *more* parameters—often with orders of magnitude more parameters than data points—will numerically converge in the loss function, fit the training data, and still successfully predict unseen data. In contrast, the "smaller" net frequently gets trapped in local minima in the loss surface and struggles to converge. LeCun indicated that researchers commonly understand that these overparameterized nets will automatically reduce their "rank" during training by implicit regularization, but admitted that this aspect is still a theoretical mystery. He thus suggested that the applied math community could perhaps contribute to the knowledge surrounding this problem.

The Value of an Applied Mathematics Background

DL has found particular success in the area of image processing via convolutional neural nets (CNNs). LeCun provided many examples of these triumphs during his talk, including medical image analysis, self-driving vehicles, and automated emergency braking systems. However, convolutional approaches

in DL succeed with image-based applications because images are rectangular lattices; they face challenges when generalizing to other graph-based applications that lack this structure.

A generalization of the convolutional net to an arbitrary graph is called a graph convolutional net. LeCun understands the extension of CNN tools to arbitrary graphs in terms of Fourier transforms. To employ the convolution on the graph, one applies Fourier transforms to the data and filter, multiplies, and then applies the inverse Fourier transform. Unlike computer scientists, most applied mathematicians are already quite familiar with these tools. They are consequently in a strong position to understand how application problems fit in a more general theoretical landscape and are thus useful in the DL community.

However, the relationship between the applied mathematics and DL communities is not one-directional. LeCun emphasized that practitioners are interested in developing DL approaches that accelerate the numerical solution of partial differential equations (PDEs). Traditional approaches to solving PDEs rely on the solution of finite difference or finite element discretizations of the differential operator. However, issues related to discretization for the time and space variables arise for stiff and/or high-dimensional problems. In areas where careful obedience of the physics makes numerical solution difficult, one may attempt to replace the solution operator with a neural net that is trained to produce a solution based on a class of examples. While accuracy, precision, and preservation of conserved quantities are issues for such neural networks, the potential speed-ups are quite promising. During his talk, LeCun alluded to applications in lattice quantum chromodynamics, fluid dynamics, and astrophysics.

A Unifying Perspective

In the past, neural networks were primarily motivated by a desire to understand the living brain. Therefore, LeCun's presentation also touched on DL approaches that mimic the ways in which humans learn, reason, and plan complex tasks. To quote LeCun, humans are "barely supervised and rarely reinforced." Why is this? Success with deep networks in image processing, for instance, requires thousands or even millions of labeled examples and an enormous amount of computational power for training purposes. The neural net may find success in the same class of images with which it was trained, but the process must begin again upon the introduction of a fresh class of images that were not seen during training. The user must also alert the machine to the new class of objects.

This is in stark contrast to the way that babies learn. They can recognize new objects after only seeing them a few times, and do so with very little effort and minimal external interaction. If ML's greatest goal is to understand how humans learn, one must emulate the speed at which they do so. This direction of research is exemplified by a variety of techniques that may or may not fit into an existing paradigm; LeCun classified these tasks under the umbrella of "self-supervised learning."

While supervised learning and reinforcement learning have shown success in isolated tasks, LeCun believes that these paradigms will never lead to so-called "artificial general intelligence," regardless of the scale-up of hardware capabilities. Throughout his talk, he alluded to some of the fundamental challenges that are associated with these approaches. In particular, LeCun feels that reinforcement learning will struggle to explore state space, especially when one imposes the "rarely reinforced" aspect of natural intelligence. He stated that reinforcement learning will thus make it difficult for

researchers to even develop a system that has “cat-level intelligence,” much less human-level intelligence. As an alternative, LeCun discussed self-supervised learning via energy-based models during the second half of his lecture. This approach foregoes neural nets altogether and instead learns an energy surface that captures dependencies between inputs and can allow for multiple output predictions. He briefly highlighted many approaches but predicted that regularized latent-variable energy-based models will be the winning framework.

LeCun concluded by addressing a pervading question: Is DL a natural science or an engineering science? Is it *truly* a science, or more like alchemy? He implicitly conceded that it is mostly the latter but argued that “just because we don’t understand it doesn’t mean we shouldn’t use it.” LeCun noted that the gap between discovery of a machine and the subsequent identification of its theory can historically be quite lengthy. For example, the telescope was developed in 1608, but it took 50 years for optics to explain why it works. The steam engine appeared in 1695, but more than 100 years passed before thermodynamics could describe its function. LeCun hopes to “find the equivalent of thermodynamics for machine intelligence, or intelligence in general.” Mathematicians will likely play a significant role in this endeavor before the DL community can reach anything that resembles a unified theory.

This article is based on Yann LeCun’s invited talk at the 2020 SIAM Conference on Mathematics of Data Science (MDS20), which occurred virtually earlier this year. LeCun’s presentation is available on SIAM’s YouTube Channel.

Manuchehr Aminian is an assistant professor in the Department of Mathematics and Statistics at California State Polytechnic University, Pomona. His interests include mathematical modeling, visualization, and mathematical methods in data science.