Why Machine Learning (ML) Works

Team: Jesse Hoogland & Karel Zijp

We are in the clutches of a machine learning (ML) moment, the third, and so far the biggest, of three waves of artificial intelligence furor. Whereas the first two eventually slowed down, this third round, driven by artificial neural networks (ANNs) that mimick the biological neural networks in our own brains, shows no sign of abating. Whether this will stay true remains to be seen.

In particular, a class of ANNs known as "deep neural networks" (DNNs), has captivated the ML community's attention. "Deep" is somewhat of a murky distinction, but we typically use it refer to neural networks with at least 100 internal, or hidden, layers. It is these kinds of systems which are responsible for the flashiest headlines of recent years: the high performances in the image-classification competitions that started this ML wave, autonomous driving, and the superhuman performances in Go and now even in (the much harder) Dota 2, Starcraft, and other e-sports.

Despite the success of these techniques, there still is no consensus on *why* DNNs work as well as they do. Excess experimental success has not begotten a standard theoretical framework. This could become a problem. At its most innocent, better theoretical understanding could prompt better experimental implementation, and we might make quicker progress than we otherwise would have. But on a more pernicious level, machine learning techniques are increasingly finding adoption in critical applications: determining who is eligible for parole, deciding how to respond in automobile accidents, and even in bolstering wartime decision-making. In these kinds of applications, we cannot accept neural networks as black-boxes. We need a rigorous understanding of their inner-workings so that we might navigate the riskiest edge cases.

It (could be) is our goal to provide an, necessarily incomplete, overview of competing explanations for DNNs success. These explanations range as much in optimism (from DNNs are curve-fitting to something revolutionarily new) as they do in disciplines (from mathematical and physical to information-theoretic and biological explanations).

(Insert introduction to DNNs here)

The Mathematical Reasons: Probabilistic Interpretation

It's an optimization problem and you're minimizing some cost function. Dropout et al. increase the likelihood of reaching better minima.

The Mathematical Reasons: DNNs as fancy curve-fitting

The most cynical take on DNNs is that they are simply a souped-up version of curve-fitting: tricks for higher-dimensional regression.

In this analysis, DNNs are a particularly effective (but not much more) means of approximating arbitrary functions and probability distributions.

There are even formal proofs to back this up, ideas like: 1. A sufficient number of hidden units paired with the alternating of linearity and non-linearity allow NNs to approximate arbitrary functions arbitrarily accurately. => The Universal Approximation Theorem (Lu et al.) 2. Increasing the depth of a network serves to increase the neuron-efficiency (logarithmically). 3. Sparsity (often accompanying deeper-networks) increases the efficiency of the training process.

Papers to look at: - Why does Deep and Cheap Learning Work so Well (Lin, Tegmark, Rolnick) - [Neural Networks, Manifolds, and Topology (colah, a blog post)] (https://colah.github.io/posts/2014-03-NN-Manifolds-Topology/)

The Physical Reasons: Energy Landscapes and Simulated Annealing

Maybe a quick commenet about this interpretation (follows more from the probabilistic interpretation)

The Physical Reasons: The Manifold Hypothesis and Symmetries

Where the cynics perhaps undersell DNNs is that their explanation does not provide a satisfying answer to why DNNs can generalize so well. State-of-the art DNNs can have millions or even billions (see GPT-2) of parameters without guaranteeing an outcome of "overfitting." This is when a model learns (almost) perfectly to reconstruct the data in its training set, but fares poorly when attempting to generalize to new, not-before-seen examples. How come DNNs can generalize at all?

The Manifold Hypothesis states that most real-world, high-dimensional data rests near a much lower-dimensional manifold embedded in that space. It offers a first-order answer: DNNs can generalize because the testing data has the same lower-dimensional structure as the training data. In light of the previous section (see colah), then, the network does not care about individual points as much as learning the boundaries between manifolds. This procedure works for all possible points on that manifold. Then, higher dimensionality of layers gives us more directions in which we can distort and manipulate the space and ultimately draw satisfactory boundaries.

Symmetry As for why the manifold hypothesis is true to begin with, the answer might have to do with symmetry. The universe's symmetries greatly constrain the allowed states and configurations.

Papers to look at: - Both of the above papers

The Physical Reasons: Markovian Data

Another explanation uses the fact that many data generation processes are Markovian and hierarchical. Then, the (hierarchical) structure of NNs allows them to run this Markovian process in reverse.

Maybe say something about Boltzmann machines, maxEnt models would also be relevant somewhere.

Papers to look at: - See Lin et al. above

The Physical Reasons: Renormalization

We can take a slightly different interpretation. Instead of deforming spaces and linearly separating them, we can interpret a neural network (in the supervised context) as an iterative procedure whose fixed points correspond to the classes of interest.

You can do a lot with the machinery from 1-D maps.

From this perspective, NNs begin to look a lot like renormalization. Indeed, you can make this a bit more precise.

Sources: - Mehta and Schwab - I have a few others if you want, but this one is seminal.

The Information Theoretic Reasons: The Information Bottleneck

The information bottleneck method (IBM) formalized the notion of "relevant information", using concepts from rate-distortion theory. This offers a framework for analyzing how neural networks process information.

Sources: - The original paperon the information bottleneck method - Tishby & Zaslavasky (turned out to be wrong but still interesting) - Koch Janusz & Ringel (using IBM to improve renormalization techniques)

The Biological Reasons: Convnets and feature accumulating in the visual system

Convnets accumulate low-level features to more complicated features and have their origins in our understanding of the retina and visual lobe.

- Krizhevsky et al.
- Overview of convnets from Stamford

The Biological Reasons:

Contrast supervised and unsupervised learning; feed-forward and recurrent networks.

Proposals of backpropagation in the brain.

Say something about genetic algorithms.

Comment on the energy efficiency of neural networks: need for 100-1000x improvement to parallel the efficiency of human brains.

Discrete systems versus continuous systems.

What we could show practically:

- Analysis of a simple (2 or 3-hidden-states) system (of fixed points and stability). Show some kind of bifurcationo
- Rederive results of Tishby and Zaslavasky (Mutual information flows)
-