Neuron
NeuroView

The Mutual Inspirations of Machine Learning and Neuroscience

Moritz Helmstaedter[1,*]
[1]Department of Connectomics, Max Planck Institute for Brain Research, Max-von-Laue-Str. 4, 60438 Frankfurt, Germany
*Correspondence: mh@brain.mpg.de
http://dx.doi.org/10.1016/j.neuron.2015.03.031

Neuroscientists are generating data sets of enormous size, which are matching the complexity of real-world classification tasks. Machine learning has helped data analysis enormously but is often not as accurate as human data analysis. Here, Helmstaedter discusses the challenges and promises of neuroscience-inspired machine learning that lie ahead.

Introduction

Brains deal with high-dimensional sensory data when navigating an organism through the environment. In turn, neuroscientists investigating the brain face high-dimensional data sets of enormous complexity, which are increasingly difficult to analyze. A prominent example of this challenge is the new field of high-resolution connectomics, in which 3D electron microscopy (EM) data sets are breaking the petabyte-scale size barrier. Analysis of these data is a major challenge, and only with machine learning (ML) techniques has the reconstruction of such data sets even become plausible. Functional imaging or behavioral tracking data also require substantially automated analysis and constitute other examples where ML algorithms can be fruitful. This viewpoint illustrates how ML has helped neuroscience and how we are still falling short of devising algorithms as powerful as human data analysis in many relevant settings. I end with a discussion about strategies to unravel the algorithmic specializations of the sensory cerebral cortex, which could ultimately provide us with the missing insights about biological algorithms, in turn inspiring next-generation high-performance ML.

High-Dimensional Data Classification

Everyday tasks like the classification of house numbers or birds are almost trivial for humans but are rather impressive conceptually; identifying an image as "bird" or "fly" means collapsing the enormous number of possible images ($10^{66,583}$ in the case of colored 92 × 92 pixel images) into just a few dozen classes (Figure 1A). One can describe these images as high-dimensional data: each of the 92 × 92 pixels can be varied independently, and the apparent dimensionality of this data (92 × 92 = 8,464) is therefore rather high. On the other hand, the classification results are often just 1D (in the case of digits, one of the 10 possible classes [0,1,2,...]). Classification thus means finding structure in a very large space of possible data instantiations.

In neuroscience, with the widespread application of high-throughput recording techniques, data analysis has become a comparable challenge in, for example, the detection of synapses and neurites in 3D EM for connectomics (see Helmstaedter, 2013 for a review). Other examples include tracking flies, mice, rodent whiskers, or embryonic cells in high-resolution videography (Kabra et al., 2013; Clack et al., 2012; Amat et al., 2014), analyzing spike data from large-scale electrode arrays (Carlson et al., 2014; Vogelstein et al., 2004), or detecting action potentials from Ca^{2+}-based fluorescence transients (Greenberg et al., 2014).

In all of these cases, teaching computers to do the analysis is of substantial value, either to automate object detection in real-world settings or to increase the throughput of otherwise prohibitively time-consuming analyses in neuroscientific experiments, which then enables new technology (cell body detection and tracking, connectomics, animal tracking). In some cases, automated analysis is needed to provide consistent results when even expert neuroscientists struggle (for example, with spike detection, large-array spiking data, genetic sequence comparison).

Machine Learning

How can machines be enabled to analyze such high-dimensional data? One approach is to treat the conversion of high-dimensional input data to lower-dimensional output as a function, parametrize this function, and optimize the parameters to best approximate this transformation. This approach requires no knowledge about how images of, say, birds are generated from classes of birds, or what noise sources are relevant. The only requisite is labels, i.e., examples of images for which the class assignment is known. Then, one optimizes the transformation from high-dimensional input to lower-dimensional output (Figure 1A, red arrow) by adjusting (or "learning") the parameters based on these labels (therefore called training data). Given that some architectures to implement this optimization were inspired by neuronal networks, and since the parameter adjustment involves the presentation of example transformations, this approach is often termed "machine learning."

Model-Based Analysis and Unsupervised Machine Learning

Another approach for high-dimensional data analysis is the converse: instead of fitting the data-to-analysis transformation, one tries to model the generation of data from the known classes or objects (Figure 1A, black arrow). This requires substantial prior knowledge about the data generation and noise sources, but in the ideal case it requires only few, if any, labeled example data. In this approach, the model contains parameters that represent the analysis result of interest, and these parameters are optimized

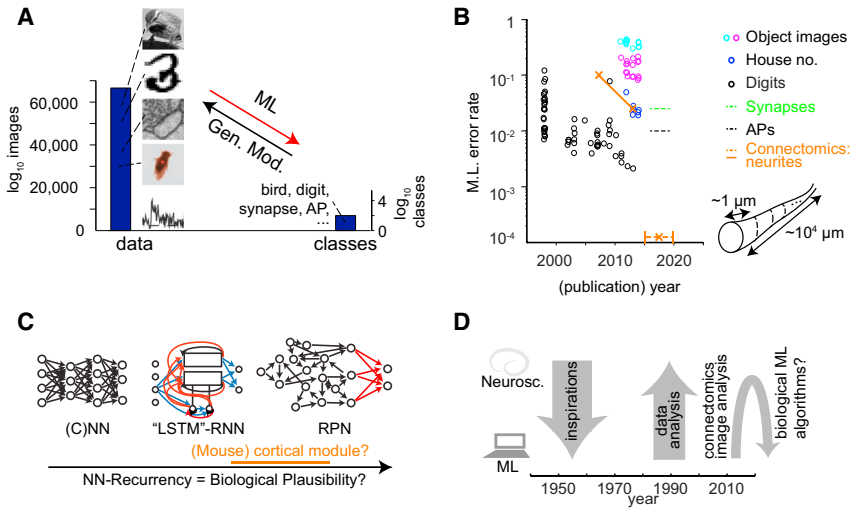

Figure 1. Machine Learning for Real-World and Neuroscientific Data Analysis

(A) Conformation space in a typical image analysis setting comparing the number of possible 92 × 92 pixel images at 24-bit color depth to the number of underlying object classes. Example images from STL-10 benchmark data set, the MNIST data set (LeCun et al., 1998), electron microscopy data for connectomics, behavioral tracking software for fly movement (Kabra et al., 2013), and intracellular Ca^{2+} transient recorded by 2-photon microscopy from live neocortex in rat. Machine learning (ML) aims to optimize the data-to-classification conversion, while generative models (Gen. Mod.) attempt to reproduce the image generation from the underlying classes.

(B) ML classification errors in benchmark data sets (STL-10; Coates et al., 2011), CIFAR-10, MNIST, SVHN, see text for references) compiled from http://yann.lecun.com/exdb/mnist/ and http://rodrigob.github.io/are_we_there_yet/build/ together with achieved (solid lines) and required neuroscientific ML error rates (dashed).

(C) Types of biologically inspired ML classifiers. Feedforward networks (left) initially proposed as two-layer perceptrons (Rosenblatt, 1958) and later developed into "deep" networks (e.g., Hinton et al., 2006; LeCun et al., 1998). Recurrent architectures with special recurrency rules to make them efficiently machine trainable (middle, LSTM networks; Hochreiter and Schmidhuber, 1997) and fully recurrent networks suggested as echo-state, random pool (RPN), or liquid-state machine-RNNs (right, e.g., Maass et al., 2002). We do not know yet whether any of these architectures are used and implemented in real biological networks in the mammalian cortex.

(D) Simplified timeline of the mutual inspirations of ML and neuroscience: can neuroscience instruct ML about the tricks of biological computing devices?

to fit the data, not the output. Take for example the analysis of functional imaging data of intracellular calcium transients (Figure 1A; Greenberg et al., 2014). The source of these calcium transients (and the analysis result of interest) is action potentials (APs) in the soma occurring at certain time points. The transformation of an action potential to the somatic calcium influx, calcium binding to the sensor proteins and its decay dynamics, as well as the imaging noise sources are relatively well understood. Therefore, the generative model describing AP-to-calcium data transformation is well constrained, and by optimizing the AP time points to generate data that best resembles the measured data, one obtains the AP time points as a result.

For such model-based approaches, almost no labels are needed, and the model is not "learned," but "known" (to the researcher). One can, however, consider the parameters of the generative model as an ideal code (the AP time points ideally encode the calcium transients). In this case, the knowledge of this code is the result of extensive biophysical research, and the compression is perfect if one is interested in APs, not calcium dynamics per se. So-called unsupervised ML approaches aim at "learning" such codes from the data. It is unlikely that brains have implemented all relevant models explicitly (maybe kinematic models, which have been present long enough to be genetically encoded, are explicitly implemented, but certainly not models about, say, smartphone behavior). The quest to learn such relevant encodings that either help classification or represent the relevant parameters directly is open, and unsupervised ML approaches are already being used when detecting extracellular spikes (dictionary learning, e.g., Carlson et al., 2014) or in typical benchmark competitions (combined unsupervised/supervised classifications; Ranzato et al., 2007).

How Good Are ML Classifiers?

To illustrate the performance improvement of ML classifiers, let's consider a data set of handwritten digits from mail envelopes (MNIST data set; LeCun et al., 1998). This has served as a key benchmark of accuracy improvements in ML research: error rates have dropped from about 1%–2% in 1998 to about 0.2% today (thus, an order of magnitude within about 15 years). While an error rate of 0.2% is already very impressive, the best ML methods perform still about 1 order of magnitude worse on more complex data sets (Figure 1B), such as house numbers from street view data or low-resolution real-world images, illustrating that machine analysis is still outperformed by human analysis in many real-world settings.

What about ML analysis in neuroscience? The ambition of mapping entire synaptic networks has spurred technological developments in 3D EM imaging, yielding high-resolution large-scale image data sets in connectomics. Human annotators (both experts and trained non-experts) can analyze this data faithfully (Helmstaedter et al., 2011), but analysis time for all-manual analysis would have made larger-scale circuit reconstruction impossible (see Helmstaedter, 2013 for a review). A key step forward was the usage of convolutional neuronal networks (CNNs) for image data analysis. While the reconstruction accuracy is still much worse than that of humans, the help of automated analysis techniques was crucial to make the first locally dense circuit reconstructions possible at all: Takemura et al. (2013) used a combination of prior-based filters and learned feature detectors; Helmstaedter et al. (2013) used CNNs and a sequence of segmentation procedures. Likewise, in behavioral tracking, ML analysis has resurfaced, being more flexible and general than purely model-based analysis (Kabra et al., 2013).

ML for Connectomics

Compared to other real-world and neuroscientific ML challenges, connectomics

turns out to be an especially hard problem (Figure 1B). Consider the reconstruction of neuronal wires (inset in Figure 1B): about every 1 μm, the classifier (human or machine) has to make a correct decision about how to and whether to continue the neurite. Considering that neurites are 10^4–10^5 μm in path length, for the faithful reconstruction of even a single neuron, the effective classifier error rate has to be on the order of 10^{-4}–10^{-5}. Currently, best classifiers have error rates of, at best, one in 10–40 μm neurite path length. Figure 1B illustrates what this means: connectomics needs an improvement in classifier accuracy of about 2 orders of magnitude to reconstruct even one neuron properly automatically and an improvement of another 7 orders of magnitude for the automated reconstruction of an entire mouse brain. Compare this to the automation improvements when classifying the rather restricted set of handwritten digits in the MNIST data set: it took about 15 years to gain 1 order of magnitude in error rate improvement. Thus, both in absolute numbers and with respect to the required rate of improvement, the challenges of connectomics are by orders of magnitude more daunting than other ML benchmarks.

The Need for Labels—Human versus Machine

ML requires large amounts of labeled examples ("training data"). In most settings, human data annotation is considered the gold standard, yet manually generated labels may contain errors. Some of these are attention related, but others may be more systematic. How can one deal with this? Human performance can be substantially augmented by consensus procedures, where labels are generated from multiple independent annotations, not a single manual choice. Such procedures are especially successful when a large fraction of single-annotator errors are unbiased and independent (such as when missing branches in neuron reconstructions; Helmstaedter et al., 2011). In all cases, human annotations need to be cross-validated to make sure that labels are interpreted with the required error margins.

This still assumes that manual annotation is superior to automated and that one has to train the computer, not the human. But is it conceptually possible to create classifiers that exceed human performance while training them on human-generated labels? In some settings, human annotation is already inferior; take the detection of APs from calcium transients, which has no plausible real-world detection analogy. Since the underlying model is biophysically well understood, computers are expected to be better analytic devices than humans, and manual annotation becomes irrelevant.

Thus, as soon as the automated analysis represents a sufficiently correct model, computer analysis can in principle exceed human performance. Proving that the learned encodings have sufficient descriptive power, however, is often difficult.

Feedforward versus Recurrent Network Architectures

I now turn to the comparison of ML techniques to biologically plausible neuronal networks. Early neuronal networks were suggested as perceptrons with an input and an output layer. More than 5 decades later, a key improvement is to work with "deep" networks, which contain many more hidden layers (Figure 1C) and have become trainable with new learning strategies from computer science (see Schmidhuber, 2015 for a review), winning today's ML competitions, including connectomic data classification. All of these are still strictly feedforward architectures. However, neuronal networks in the brain are highly recursive. Thus, in addition to the push for deep networks, the usage of recurrent neuronal networks (RNNs) has resurfaced in ML after training them has become routine (Schmidhuber, 2015). Currently, RNNs are most successful in the automated analysis of visual or acoustic sequences.

Is it likely that any of these brain-inspired but man-conceived network architectures are in fact at the core of the brain's classifiers? Have we already obtained the relevant architectural insights, such that all it would take is better and more-efficient machine implementation to match and outperform human analysis (as the more optimistic ML proponents would argue), or are there still tricks missing that evolution has found but human thought hasn't yet?

A Strategy to Discover the Brain's Classification Tricks

The key circuits for sensory classification beyond hard-wired genetically determined reactions are most likely located in the cortex of mammals. Can we investigate these circuits such that we can determine the algorithmic solutions developed during evolution of the biological computing devices (Figure 1D)?

Input from the sensory periphery arrives in primary sensory cortex via thalamic afferents. In the case of rodent primary somatosensory cortex (S1), the largest fraction of this innervation targets neurons in layers (L) 4, 5, and 3 (Meyer et al., 2010). L4 is a particularly clustered circuit in mouse and rat barrel cortex, where the input from whiskers on the animal's snout is encoded. This circuit contains about 2,000 neurons in a sudden representational expansion when compared to the about 200 neurons responsible for the same main sensory input in the brainstem and thalamus. L4 neurons are highly interconnected (pairwise connectivity of 20%–30%; Feldmeyer et al., 1999), and their main output is neurons in L2, L3, and L5. What is the function of this first-stage cortical circuit in L4? It has been maintained that amplification of thalamic inputs is the main purpose of L4 (e.g., Feldmeyer et al., 1999; Lien and Scanziani, 2013). However, it may seem unlikely to build an elaborate 10-fold circuit expansion just for signal amplification.

A sensible strategy is therefore to start by mapping the circuit structure of a barrel in mouse L4 as the first cortical computational module. Are any of the proposed ML architectures actually implemented in such a computational module? If so, where are the main readouts: L2/3 or L5? How is top-down input processed? Are hypotheses projected down to the primary sensory cortex and, if so, to where? Most L4 neurons are too local to receive long-range input via L1, but L2/3 and L5 pyramidal neurons extend their dendrites into L1 and can therefore in principle receive top-down input.

Mapping one such cortical processing module alone already constrains the range of ML architectures that can be implemented. In order to disambiguate individual circuit patterns from general circuit principles, it will be necessary to screen for the invariants between different cortex

modules within individuals, between cortex modules of different individuals, and for the circuit principles that may be invariant between different sensory cortices serving the key sensory modalities: sensation, audition, and vision (S1, A1, V1, respectively).

This "search for invariants" in one species, mouse, is almost doable today. Recent improvements in imaging speed and analysis throughput promise to make the reconstruction of one cortical module doable, and the need for screening techniques is evident. One may wonder, though, whether this is already good enough for algorithmically understanding superior human classification abilities. Indeed, the classification performance of a mouse can be matched by today's computer algorithms, while on the other hand, a detailed mapping of human cortical networks is still unrealistic because of their sheer size.

Therefore, I propose that comparative mapping along the species axis should be the next goal. This would progress from mouse to rat with its higher learning performance, a cortex 3-fold larger by number of neurons, yet with the same modular structure in S1 (with larger columns). Can we find algorithmic, principled improvements in rat cortex when compared to mouse?

What about cat, non-human primates, and ultimately human cortex samples? With progress in imaging and reconstruction throughput, we will attain these volumes. With proper algorithmic preparation (insights into mouse cortex classifiers as a baseline, and stepwise extrapolation along the species axis), we may be able to use just a few of the higher-species samples to discover which new algorithmic inventions are present compared to the simpler animals' cortex.

Such a research program, which we have been pursuing in my laboratory for a few years, is ambitious and may take decades to be successful. Its goals have received major attention by a recently proposed program of the US Intelligence Advanced Research Projects Activity (IARPA) (MICRoNS, http://www.iarpa.gov/index.php/research-programs/microns). Even if progress will be slow in the beginning, the quest to crack the classification tricks of the biological computing devices in our brains is open, and we may finally return the favor to computer science by providing algorithms that may be better than all the algorithms that human thought has come up with so far. On the way, we will need massive help from machine learning.

ACKNOWLEDGMENTS

I thank members of my laboratory for discussions, literature analysis, comments on the manuscript, and figure generation. The author is a shared stakeholder of the patent Method and apparatus for image processing. Published Patent Application No. 20100183217.

REFERENCES

Amat, F., Lemon, W., Mossing, D.P., McDole, K., Wan, Y., Branson, K., Myers, E.W., and Keller, P.J. (2014). Nat. Methods *11*, 951–958.

Carlson, D.E., Vogelstein, J.T., Qisong Wu, Wenzhao Lian, Mingyuan Zhou, Stoetzner, C.R., Kipke, D., Weber, D., Dunson, D.B., and Carin, L. (2014). IEEE Trans. Biomed. Eng. *61*, 41–54.

Clack, N.G., O'Connor, D.H., Huber, D., Petreanu, L., Hires, A., Peron, S., Svoboda, K., and Myers, E.W. (2012). PLoS Comput. Biol. *8*, e1002591.

Coates, A., Lee, H., and Ng, A.Y. (2011). An analysis of single-layer networks in unsupervised feature learning. In Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS) (Ft. Lauderdale, FL).

Feldmeyer, D., Egger, V., Lubke, J., and Sakmann, B. (1999). J. Physiol. *521*, 169–190.

Greenberg, D.S., Wallace, D.J., and Kerr, J.N. (2014). Cold Spring Harb Protoc *2014*, 912–922.

Helmstaedter, M. (2013). Nat. Methods *10*, 501–507.

Helmstaedter, M., Briggman, K.L., and Denk, W. (2011). Nat. Neurosci. *14*, 1081–1088.

Helmstaedter, M., Briggman, K.L., Turaga, S.C., Jain, V., Seung, H.S., and Denk, W. (2013). Nature *500*, 168–174.

Hinton, G.E., Osindero, S., and Teh, Y.W. (2006). Neural Comput. *18*, 1527–1554.

Hochreiter, S., and Schmidhuber, J. (1997). Neural Comput. *9*, 1735–1780.

Kabra, M., Robie, A.A., Rivera-Alba, M., Branson, S., and Branson, K. (2013). Nat. Methods *10*, 64–67.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Proc. IEEE *86*, 2278–2324.

Lien, A.D., and Scanziani, M. (2013). Nat. Neurosci. *16*, 1315–1323.

Maass, W., Natschläger, T., and Markram, H. (2002). Neural Comput. *14*, 2531–2560.

Meyer, H.S., Wimmer, V.C., Hemberger, M., Bruno, R.M., de Kock, C.P., Frick, A., Sakmann, B., and Helmstaedter, M. (2010). Cereb. Cortex *20*, 2287–2303.

Ranzato, M.A., Poultney, C., Chopra, S., LeCun, Y. (2007). Efficient Learning of Sparse Representations with an Energy-Based Model. Advances in Neural Information Processing Systems - Proceedings of the 2006 Conference 19.

Rosenblatt, F. (1958). Psychol. Rev. *65*, 386–408.

Schmidhuber, J. (2015). Neural Netw. *61*, 85–117.

Takemura, S.Y., Bharioke, A., Lu, Z., Nern, A., Vitaladevuni, S., Rivlin, P.K., Katz, W.T., Olbris, D.J., Plaza, S.M., Winston, P., et al. (2013). Nature *500*, 175–181.

Vogelstein, R.J., Murari, K., Thakur, P. H., Diehl, C., Chakrabartty, S., Cauwenberghs, G. (2004). Spike Sorting with Support Vector Machines. Proceedings of the 26th Annual International Conference of the IEEE EMBS, San Francisco, CA, USA 0-7803-8439-3/04 2004 IEEE.