Available online at www.sciencedirect.com

ScienceDirect

Current Opinion in Neurobiology

# Open source tools for large-scale neuroscience

Jeremy Freeman

New technologies for monitoring and manipulating the nervous system promise exciting biology but pose challenges for analysis and computation. Solutions can be found in the form of modern approaches to distributed computing, machine learning, and interactive visualization. But embracing these new technologies will require a cultural shift: away from independent efforts and proprietary methods and toward an open source and collaborative neuroscience.

**Address**
HHMI Janelia Research Center, 19700 Helix Drive, Ashburn, VA 20147, United States

Corresponding author: Freeman, Jeremy (freemanj11@janelia.hhmi.org)
*URL:* http://www.jeremyfreeman.net/

**Current Opinion in Neurobiology** 2015, **32**:156–163

This review comes from a themed issue on **Large-Scale Recording Technology**

Edited by **Francesco P Battaglia** and **Mark J Schnitzer**

For a complete overview see the Issue and the Editorial

http://dx.doi.org/10.1016/j.conb.2015.04.002



''I am absolutely convinced that in a few decades, historians of science will describe the period we are in right now as one of deep and significant transformations to the very structure of science. And in that process, the rise of free openly available tools plays a central role.''

—Fernando Perez, creator of iPython [1].

Understanding the brain has always been a shared endeavor. But thus far, most efforts have remained individuated: labs pursuing independent research goals, slowly disseminating information via journal publications, and when analyzing their data, repeatedly reinventing the wheel.

New experimental technologies are forcing a paradigm shift. Data sets are getting both larger and more complex. Many labs have more data than they have time to analyze, even for basic processing, let alone rich data exploration. The scale and complexity of the problems we want to tackle demands shared solutions.

Large-scale, high-resolution optical recordings of neural activity present a particularly exciting and challenging case study, and will be the focus of this essay. As encapsulated in an earlier review, the ''operational principles of a neural circuit must be deduced through analysis of its structure and function'' [2]. Crucial to this effort is monitoring neural activity: at single-neuron resolution, in large populations, across multiple brain areas, or even the entire brain, during behavior.

Imaging methods – including two-photon laser scanning microscopy[3], light-sheet imaging [4,5], and light-field imaging [6,7] – monitor neural activity via protein sensors that convert changes in neural state, like changes in $Ca^{2+}$ concentration, into changes in fluorescence (the case of two-photon imaging in head-fixed behaving mice is described in detail in another review in this same issue, Peron et al.).

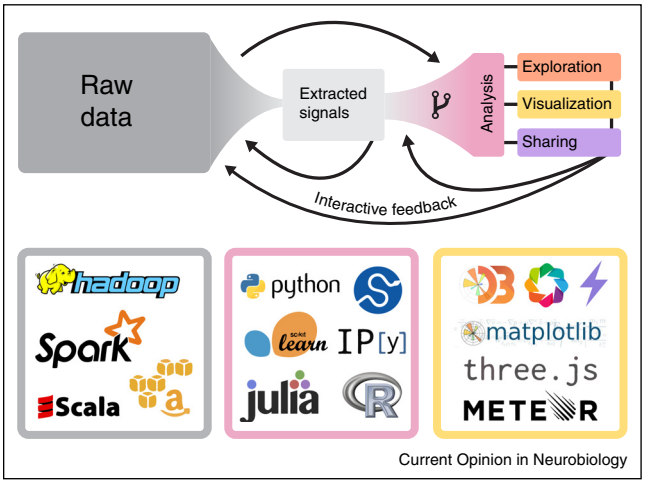
The raw recorded data are time-varying images. Currently, a two-photon imaging experiment monitoring a region of mouse visual cortex can yield $512 \times 512 \times 4$ pixel at 8 Hz, resulting in ~60 GB per hour, while a whole-brain light-sheet imaging experiment in a larval zebrafish can yield $1000 \times 2000 \times 40$ pixel at 2 Hz, resulting in ~1.2 TBs per hour. These numbers describe one recording session from one animal, whereas most experiments involve many of each. Improvements in the spatial extent and temporal resolution of these technologies [8,9] will only make these data sizes larger.

***

To understand the analytical challenges posed by imaging data, it is worth first understanding the typical data analysis steps. In its abstract form, this sequence shares much in common with data analytics in many industry settings (Figure 1).

Images must first be preprocessed by registering across time to compensate for motion, the form of which may differ across experimental preparations and imaging modalities. Typically, this is followed by some kind of extraction of identified neuronal signals; for example: segmentation through morphological analysis of image structure [10•,11], activity-based identification and demixing of correlated fluorescence patterns [12•,13•], or some combination of the two [14]. Which methods are most appropriate will depend on the model system, resolution and sampling in both space and time, the indicator of neural activity, and the area imaged. For large data sets covering diverse morphological structures,

**Figure 1**




Most large-scale analytics, whether in industry or neuroscience, involve common patterns. Raw data are massive in size. Often, they are processed so as to extract signals of interest, which are then used for statistical analysis, exploration, and visualization. But raw data can be analyzed or visualized directly (top arrow). And the results of each successive step informs how to perform the earlier ones (feedback loops). Icons below highlight some of the technologies, discussed in this essay, that are core to the modern large-scale analysis workflow.

voxel-wise analyses may provide a complementary alternative [4,15••]. In either case, temporal filtering is required to remove artifacts (e.g. trends), and deconvolution can be used to try to identify spikes [16]. Having identified neurons and their responses, we want to understand them. This process is more exploratory, and can include relating neural responses to properties of the stimulus or behavior of an animal [17•], identifying topological or low-dimensional structure in the data [18], or inferring functional coupling [19].

The first challenge is that there is currently little agreement as to how to solve these problems. Many existing approaches are ad-hoc, especially for basic data processing. Analyses are often more focused on suiting the needs of individual labs than the community, and algorithmic sophistication is valued above ease of implementation—both unsurprising given the ordinary incentive structures in academia. Little is available in the way of vetting or benchmarking or standardization, partly due to the lack of curated ''ground-truth'' data sets, in formats that are easily accessible from modern, distributed computing environments.

The second challenge is that all aspects of analysis must scale to potentially massive data sets, but single workstation solutions designed for smaller datasets remain the norm. To process raw data efficiently, we need to both load the data and operate on it in parallel. Many operations are ''embarrassingly parallel'' – we apply the exact same function to different portions of the data – but require different strategies for splitting up the data (''partitioning'') depending on whether operations act locally in time, space, or both. Distributing a complete sequence of steps from data to result can quickly become complex. Some algorithms are also more scalable than others. For example, parallelizing an image registration algorithm that applies an operation to the image at each time point might be trivial, but parallelizing an algorithm that examines pairs of time points, and updates parameters after examining each pair, might be a significant challenge. Even for algorithms that scale well, complete processing pipelines usually require multiple passes over the same data—e.g. image filtering, registration, temporal filtering, factorization etc. Especially when data do not fit in the memory of a single machine, it becomes essential to minimize unnecessary reloading, and to efficiently combine sequences of operations.

After reducing a data set to, say, the time series of hundreds or many thousands of neurons, scalability remains a challenge, but in a different form. The data can be loaded into the memory of one machine, but fitting a complex model to every neuron becomes frustratingly slow, and fitting network models with coupling across neurons becomes intractable because they can create – in the process of analysis – objects that no longer fit in memory.

In approaching these challenges, we must balance the need for standardization and scalability with the importance of flexibility and interactivity. Different stages of analysis inform one another, with the results of one step suggesting a change to another (indicated by the feedback loops in Figure 1). It may prove beneficial to focus less on particular algorithms, and more on the access patterns and forms of useful distribution common to *all* algorithms, yielding modular frameworks into which new algorithms can be incorporated and compared.

Solving these challenges will not only require new tools, but also a new culture. Most labs develop custom analysis strategies, using proprietary tools like Matlab that are poorly suited to collaborative development, inventing creative algorithms but only applying them to data from the lab in which they were developed, because they are hard to reproduce, require complex configuration, and barely run on single workstations.

Imagine, instead: fast open-source libraries for common analyses, available to anyone and developed by all, with intuitive, modular code bases supporting customization, comparison, and benchmarking of pipelines and parameters, implemented in distributed systems that can run in cloud computing environments, with web-based interfaces for interactively exploring data and visualizing results. An exciting new ecosystem of open-source