Large-Scale Neural Network Models for Neuroscience

CS 375/Psych 249 (Stanford University, Fall 2017)

Overview

This class will serve as an introduction to designing, building, and training large-scale neural networks for modeling brain and behavioral data, including: deep convolutional neural network models of sensory systems (vision and audition); recurrent neural networks for dynamics, memory and attention; integration of variational and generative methods for cognitive modeling; and methods and metrics for comparing such models to real-world neural data. Attention will be given both to established methods as well as cutting-edge techniques. Students will investigate conceptual bases for deep neural network models, while learning to implement and train large-scale models in Tensorflow using GPUs.

Basic Info

- Times: Monday & Wednesday, 4:30-6pm.
- Location: Lathrop Library, Room 299.
- Instructors: Daniel Yamins (`yamins@stanford.edu`) and Damian Mrowca (`mrowca@stanford.edu`)
- Course website: `http://cs375.stanford.edu`.
- Course code repository: `https://github.com/neuroailab/cs375`.

Modules

Through the course of the quarter, we will cover modules on the following topics:

- **Feedforward Convolutional Neural Networks (CNNs) and Visual System.** This will cover basic results using deep convolutional neural network models optimized for visual tasks as models for neural response patterns in the primate visual system. Useful references include [1, 2, 3, 4, 5, 6, 7, 8, 9].
- **CNNs and the Auditory System.** This will cover more recent work showing how deep convolution networks optimized to solve auditory tasks, including speech and music genre recognition, can be used as models of the human auditory system. References: [10, 11].
- **Autoencoders, unsupervised learning, and generative models.** This will cover the concept of autoencoders, a idea in unsupervised learning.
- **The retina.** [12].
- **Dynamics in the motor system.** [13, 14, 15]
- **Recurrence and feedback in vision.**
- **Reinforcement learning.**

Course Structure

The course will divided into a series of 3-session modules, during which students will work in small teams to re-implement an existing piece of work using neural networks to model neuroscience data. Each module will be organized in the following format:

- An introductory lecture on the topic (1hr in Session 1), describing key results in the area, with particular focus on the paper to be reproduced.
- Guided team work, with technical consultation from instructors (Sessions 2-3), with class-wide mini-lectures as needed.
- At the beginning of the next session, one or two teams will present results from their work over the course of the module (first 30 minutes of session 1 for next module).

The course will require work both during class meeting sessions, as well as significant team work time outside of the session. Student groups will be assigned at the beginning of the quarter by course staff, and are expected to remain as constituted throughout the quarter. Students are expected to attend the course every session to participate in work with their group.

Assumed Background

Students should possess a significant amount of background knowledge to ensure productive use of course time. Assumed background topics (and Stanford courses relevant to these topics) include:

- Unix shell commands and environment (CS 1U).
- Python programming language (CME 193).
- Differential equations and linear algebra (Math 51, 53, 113, CME 102, 104)
- Probability theory (Math 230A, CME 106, CS 109)
- Basics of machine learning, neural networks, including back-propagation and convolution (CS 131, 221, 229, 231N).

Although we will not make formal courses in any of the above hard prerequisites for enrollment in CS 375, we *really will* assume you have at least the majority of these skills, and will likely not devote time to formal instruction in them. The course will also assume that you are familiar with using Git, the open source versioning control system. Damian will provide help at his office hours for those who don't yet know Git and GitHub. You can also consult the Git Manual[1].

Lab Reports

At the end of each module, each team will be expected to submit a Lab Report containing their results on each module in the form an executable Jupyter Notebook[2]. Details of how Lab Reports should be composed will be given at the beginning of the quarter. A key feature of data organization for the class will be that students will store the results of their work in a MongoDB database[3]. Lab report submissions will be required to draw results live from this database.

Lab reports for a given module will be required to be committed to the CS 375 code repository the night before the beginning of the next module (e.g. if module 2 starts on Monday, the lab report for module 1 needs to be committed to the repo by Sunday night). For each module, each team should select a reporter who will make sure to commit the lab report; the role of reporter should rotate throughout the quarter.

Final Project

Computational Resources

Groups will be expected to work together using common resources. The course will provide credits on Google Cloud for access to GPU computing resources and data storage. We will provide short introduction to how to use Google Cloud as part of the course, and Damian can answer questions about this in his office hours. Students can also use other resources at their disposal for the project, including any local machines they have, or other cloud services such as AWS or Azure.

Grading Basis

The course will count for 3 units and may be taken only for a letter grade. Grades will be calculated based on: lab reports (40%), final project (35%), class participation (25%).

References

[1] Yamins*, D. *et al.* Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences* (2014).
[2] Yamins, D. L. & DiCarlo, J. J. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience* **19**, 356 (2016).
[3] Cadieu, C. F. *et al.* Deep neural networks rival the representation of primate it cortex for core visual object recognition. *PLoS computational biology* **10**, e1003963 (2014).
[4] Yamins, D., Hong, H., Cadieu, C. & Dicarlo, J. Hierarchical modular optimization of convolutional networks achieves representations similar to macaque it and human ventral stream. *Advances in Neural Information Processing Systems* (2013).
[5] Khaligh-Razavi, S. M. & Kriegeskorte, N. Deep supervised, but not unsupervised, models may explain it cortical representation. *PLOS Comp. Bio.* (2014).
[6] Güçlü, U. & van Gerven, M. A. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *The Journal of Neuroscience* **35**, 10005–10014 (2015).
[7] Serre, T., Oliva, A. & Poggio, T. A feedforward architecture accounts for rapid categorization. *Proc Natl Acad Sci U S A* **104**, 6424–9 (2007). 0027-8424 (Print) Journal Article.
[8] DiCarlo, J. J., Zoccolan, D. & Rust, N. C. How does the brain solve visual object recognition? *Neuron* **73**, 415–34 (2012).
[9] Hong*, H., Yamins*, D., Majaj, N. & JJ, D. Representation of non-categorical properties in inferior temporal cortex. *in prep* (2014).

[1] https://git-scm.com/documentation
[2] https://ipython.org/notebook.html
[3] https://www.mongodb.com/

[10] Kell, A., Yamins, D., Norman-Haignere, S. & McDermott, J. Functional organization of auditory cortex revealed by neural networks optimized for auditory tasks. In *Society for Neuroscience* (2015).
[11] Güçlü, U., Thielen, J., Hanke, M., van Gerven, M. & van Gerven, M. A. Brains on beats. In *Advances in Neural Information Processing Systems*, 2101–2109 (2016).
[12] McIntosh, L., Maheswaranathan, N., Nayebi, A., Ganguli, S. & Baccus, S. Deep learning models of the retinal response to natural scenes. In *Advances in Neural Information Processing Systems*, 1369–1377 (2016).
[13] Sussillo, D., Churchland, M. M., Kaufman, M. T. & Shenoy, K. V. A neural network that finds a naturalistic solution for the production of muscle activity. *Nature neuroscience* **18**, 1025–1033 (2015).
[14] Mante, V., Sussillo, D., Shenoy, K. V. & Newsome, W. T. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature* **503**, 78–84 (2013).
[15] Pandarinath, C. *et al.* Inferring single-trial neural population dynamics using sequential auto-encoders. *bioRxiv* 152884 (2017).