

BIO 610: Introduction to Neural Computation

This is a graduate level course on Neural Networks. Students with either quantitative or biological backgrounds are all welcome to attend.

When: Tuesday/Thursday 10-11:20am

Where: McKenzie 473 (<https://library.uoregon.edu/classrooms/473MCK>)

Instructor: ; Office 238 Huestis.

Hours by appointment.

Prerequisite: Calculus, Linear Algebra. It helps to know Statistical Mechanics and Probability, but the course will begin with a summary of useful techniques. No previous experience with neural data is required.

Programming: A basic familiarity with Python is recommended. A nice tutorial from the Allen Institute can be found [here](https://github.com/AllenInstitute/SWDB_2018/tree/master/PythonBootcamp) (https://github.com/AllenInstitute/SWDB_2018/tree/master/PythonBootcamp).

Evaluation: The final grades will be based on: homework assignments (exercises), one third; class participation, one third; each student will give a final lecture/seminar based on one or more research papers on a special topic (one third).

Course goals: We will learn the fundamental concepts and techniques behind recurrent and feedforward neural networks, with a special emphasis on neuroscience applications. The course will be a mix of theoretical methods for analyzing neural networks and practical simulations run in Python using Jupyter notebooks. We will work out many examples in full details, with more emphasis on the problem solving strategy rather than on the formal constructions and proofs. Students will familiarize with a variety of ideas from mathematics, physics, neurobiology, machine learning and information theory. The computational tools we will explain in the course are broadly used in very diverse areas of research, from system biology to high frequency trading in finance

Syllabus

Class	Topic
Tue 4/2	Overview of the course and current research.
Thurs 4/3	Recap of probability theory: conditional probabilities, Bayes' theorem with several examples.

Tue 4/9	Statistical inference. Supervised learning: the perceptron.
Thurs 4/11	AND, OR and XOR functions. The Hebb rule. Neurobiological evidence for synaptic plasticity.
Tue 4/16	The perceptron learning rule, gradient descent learning. Learning as inference. Multi-layer networks: XOR function; error back-propagation.
Thurs 4/18	Unsupervised learning: Principal Component Analysis, Oja's rule.
Tue 4/23	Coding session: Perceptron.
Thurs 4/25	Coding session: Deep learning.
Tue 4/30	Auto-associative memory and attractor neural networks. Hebbian learning rule, synaptic plasticity. Memory as an Ising model: the Hopfield network.
Thurs 5/2	Coding session: Playing with the Hopfield network.
Tue 5/7	Coding session: Learning in the Hopfield network.
Thur 5/9	Recap of statistical mechanics: ensembles, statistical entropy, Ising model with nearest neighbor interactions.
Tue	Ising model with long range interactions and phase

5/14	transitions. Mean field theory for the Ising model.
Thurs 5/16	Statistical mechanics of the Hopfield network.
Tue 5/21	Computation of the capacity of the Hopfield network using mean field theory. Absence of spurious retrieval states; phase diagram in the temperature-storage plane.
Thurs 5/23	The brain as an anticipating machine: Boltzmann machines, Helmholtz machines learn probability distributions. Wake and sleep learning rule.
Tue 5/28	Rate networks. Backpropagation through time.
Thurs 5/30	Coding session: training recurrent networks using backpropagation through time.
Tue 6/4	Reinforcement learning. The temporal credit assignment problem and its solution using temporal difference learning.
Thurs 6/6	The actor/critic model.

Books

- Hertz, Krogh & Palmer, *Introduction to the theory of neural computation* The Hopfield network and the supervised learning part of the class are partly based on this book.
- David MacKay, *Information Theory, Inference, and Learning Algorithms* (<http://www.inference.phy.cam.ac.uk/itprnn/book.html>). Learn while having fun! Fantastic book, and it's freely available online. The inference and the Boltzmann machine parts of the course are

based on this book. It's going to become your favorite book, resistance is futile.

- Mezard, Parisi & Virasoro, *Spin glass theory and beyond*. Classic statistical mechanics book for the theory of spin glasses and the Hopfield network. Includes reprints of all the original papers, a must-read for everybody. I can't stress this enough: you will really enjoy reading the original papers. Back then, people put a lot of efforts in writing clear papers...
- Amit, *Modeling brain functions*. An exhaustive treatment of attractor neural networks as of the early nineties, starting from the Hopfield model. If you are looking for a more didascallic approach than the previous book.
- Abbott & Dayan, *Theoretical neuroscience*. The main textbook to get acquainted with the field of neuroscience. You can't live without it.
- Trappenberg, *Fundamentals of computational neuroscience*. Textbook in computational neuroscience. Parallel to the explanations of each topic, you can find *Matlab* code examples to start your own neural network simulations. Super fun.
- Dalvit et al., *Problems on statistical mechanics*. Collection of exercises on statistical mechanics, with solutions. Very useful as a general recap, explains many cool tricks to compute partition functions.

Exam

The exam is in the form of a student project, whose list of topics can be found below. Projects include a theoretical part and a numerical part. The items in the list below contains a few articles and book references on a specific problem, plus a coding problem. They are a mixture of classic results in the Neuroscience literature, that we could not discuss in class due to time constraints, or advanced research articles on recent or current open problems, whose theoretical background was discussed in class. The student will prepare a one-hour blackboard seminar on one topic, including theory

and simulation parts, to be held after the end of the semester.
