Computational Neuroscience NBIO136b Spring 2020 (Tue & Thu 2-3:20pm)

How can neurons in our brain produce spikes of electrical activity? How do the neurons communicate with each other? How do groups of neurons act in concert to produce the mental activity so important to our lives? If you have thought about these questions and want more than a hand-waving answer, then computational neuroscience is the course for you.

A rigorous understanding of any process requires us to be able to formulate the process in terms of mathematical equations. When a process is as complex as the voltage dynamics of a single-cell and even more so the coherent operation of many such cells in our brain, we must both simplify the underlying mathematical description and use a computer simulation to test the mathematical equations properly describe the behavior. In this course, you will learn these skills, which in principle can be generalized to studies of any complex system.

Our class is “half-flipped” with 50% presentation by the professor and 50% in-class tutorials. Required readings to gain course content will be done by students before each class. Students will comment or ask/answer questions on the readings and gain credit for doing so before each class.

Learning Goals

After taking this course you should be able to simulate model neurons and circuits and analyze basic spike train data. More generally you should gain the ability to produce a simple model of a dynamical biological system with appropriate differential equations, to write a computer code that will solve the model through time and interpret the meaning and relevance of the resulting outputs.

Instructor: Paul Miller, Volen 317 (via Volen 316) tel: 6-2890 email: pmiller@brandeis.edu
Office hrs: Mon 11-12, Thu 12:30-1:30pm.

TA: Jordan Breffle, Volen 316 tel: 781-736-3138 email: jbreffle@brandeis.edu
Office hrs: Bassine-Volen 3rd floor connector Wed 3:30-5pm

Course website on Latte. Also see http://http://people.brandeis.edu/~pmiller/TEXTBOOK/

Book: An Introductory Course in Computational Neuroscience (by P. Miller)
Each chapter of the book will be freely available as a pdf on NotaBene – email me to sign up if you enroll in the course after the first class.
You will be required to make online comments or ask questions on each reading section.
For more in-depth reading, see Theoretical Neuroscience by Dayan and Abbott.

Optional Introductory Matlab Tutorials (for those new to programming): Date TBD (8-10:30pm) (Farber Computer Classroom).

Please do **download Matlab + practice the tutorial found on Latte before class**!
(Evening tutorial dates will be based on student and classroom availability).

Grading: Tutorials 60%; NotaBene Comments 10%; Midterm 10%; Final Exam 20%.

If you are a student with a documented disability on record at Brandeis University and wish to have a reasonable accommodation made for you in this class, please see me immediately.
You are expected to be familiar with and to follow the University’s policies on academic integrity (see http://www.brandeis.edu/studentlife/sdc/ai). Faculty may refer any suspected instances of alleged dishonesty to the Office of Student Development and Conduct. Instances of academic dishonesty may result in sanctions including but not limited to, failing grades being issued, educational programs, and other consequences.

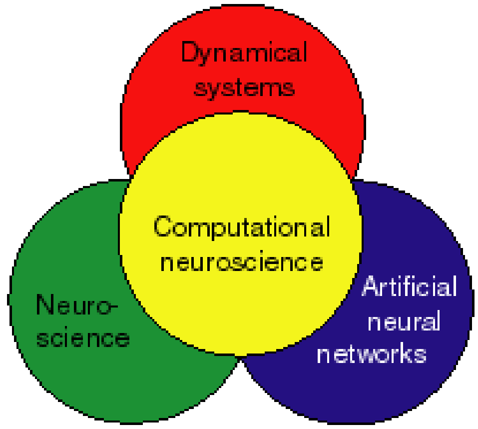
SCHEDULE OF CLASSES

Jan 14	Introduction to course, Matlab, and differential equations. (Ch. 1, Sections 1.4-1.5)
Jan 16	Matlab Tutorial (Ch. 1 Sections 1.5-1.6)
Jan 21	*The leaky integrate-and-fire model (Tutorial 2.1, Ch. 2)
Jan 23	Modeling the refractory period (Tutorial 2.2, Ch. 2)
Jan 28	*Extensions of the LIF model (Tutorial 2.3, Ch. 2)
Jan 30	Generating receptive fields with spike-triggered averages (Tutorial 3.1, Ch. 3)
Feb 4	*Statistical properties of simulated spike trains (Tutorial 3.2, Ch. 3)
Feb 6	Receiver-operating characteristic of a noisy neuron (Tutorial 3.3, Ch. 3)
Feb 11	*The Hodgkin-Huxley model (Tutorial 4.1, Ch. 4)
Feb 13	Post-inhibitory rebound (Tutorial 4.2, Ch. 4)
Feb 25	*A two-compartment model of an intrinsically bursting neuron (Tutorial 4.3, Ch. 4)
Feb 27	Synaptic responses to changes in inputs (Tutorial 5.1, Ch. 5)
Mar 3	Detecting circuit structure and non-random features in a connectivity matrix (Tutorial 5.2, Ch. 5)
Mar 5	MIDTERM
Mar 10	*Bistability and oscillations from two LIF neurons (Tutorial 5.3, Ch. 5)
Mar 12	Bistability and oscillations in a firing-rate model with feedback (Tutorial 6.1, Ch. 6)
Mar 17	*Frequency of excitatory-inhibitory coupled unit oscillator and PING (Tutorial 6.3, Ch. 6)
Mar 19	Dynamics of a decision-making circuit with two modes of operation (Tutorial 6.2, Ch. 6)
Mar 26	*Orientation selectivity in a ring model (Tutorial 6.4, Ch. 6)
Mar 31	The inhibition-stabilized circuit (Tutorial 7.1, Ch. 7)
Apr 2	*Diverse dynamical systems from similar circuit architectures (Tutorial 7.2, Ch. 7)
Apr 7	Pattern completion and pattern separation by Hebbian learning (Tutorial 8.1, Ch. 8)
Apr 14	*Competition via STDP (Tutorial 8.2, Ch. 8)
Apr 21	Learning the weather-prediction task in a neural circuit (Tutorial 8.3, Ch. 8)
Apr 23	*Principle Component Analysis (Tutorial 9.1, Ch. 9)
Apr 28	A model of eye-blink conditioning (Tutorial 8.4, Ch. 8) + Review

* = tutorial for students to complete over two successive classes – tutorial due by the start of class usually exactly one week later, but in the case of breaks 2 classes later.

Tutorials: Each Tutorial scored out of 20,with 2pts deducted per day late until a score of 10 is reached. Half of points lost for other than lateness can be regained by redoing the tutorial.
Only your best 9 Tutorials (of 11) are kept! But 1% of grade subtracted for each unexcused absence/non-submission/lack-of-fair-effort.

Computational Neuroscience: Overview



from http://www.scholarpedia.org/article/Encyclopedia_of_computational_neuroscience

Original definition of Computational Neuroscience:
"Computational neuroscience is the study of brain function in terms of the information processing properties of the structures that make up the nervous system"
Schwartz, Eric (1990). *Computational neuroscience*. Cambridge, Mass: MIT Press.

At the intersection of:

Neuroscience: The study of neurons, including their internal properties as well as their interconnected circuitry, and the associated biological processes that maintain the function of neural circuits. Some subfields of neuroscience: (1) neurogenetics, (2) cellular neuroscience, (3) systems neuroscience, (4) cognitive neuroscience. At present computational neuroscience aims to link (2), (3) and (4).

Dynamical systems theory: A branch of mathematics, describing how a system changes over time in response to inputs as a function of its current state. Typically, this can be the result of analyzing coupled nonlinear differential equations. Since all living things contain processes that change in time, interact with each other, and are nonlinear, dynamical systems theory is the required math to understand life.

Artificial Neural Networks: A field that studies how extremely simplified models, such as neural units which are either "on" or "off", can lead to the solution of computational or cognitive problems. Solutions depend upon the connections between units so the models are also called "Connectionist Models".

Within NBIO136:

Single cell models – how does a neuron work (and produce spikes) and how can we model it?

Models range in complexity as more channels or conductances are added to a neuron and/or more spatial complexity is added.

Small circuits – if 2 or more neurons are connected by a synapse, we model the synaptic interaction. The coordinated behavior of just 2 neurons can be qualitatively different from one alone. Small circuits are usually modeled with more realistic single-cell models and synapses.

Large circuits – tens to thousands of model cells, often aimed at producing behavior that matches either neural measurements in animals or can lead to intelligent behavior. Level of modeling ranges from simple models of spiking neurons to "firing-rate" models similar to those used in artificial neural networks.

Plasticity – the term used for changes in synapses connecting neurons, or the neurons themselves. Plasticity is essential for learning and long-term memory.

Spike train analysis – when spike times are recorded, what information can we derive from them?