**MA665: Introduction to Modeling and Data Analysis in Neuroscience (Fall 2018)**

**Instructor:** Mark Kramer (mak@math.bu.edu)

**Course Hours:** September 6, 2016 – October 20, 2016 [7 weeks]

Tuesday & Thursday, 12:30-1:45 PM, CAS 330

**Office Hours:** TBD, Math & Stats Room 224, or by appointment

**Textbook:** None

**Course Website:**      (github)

**Prerequisites:** Graduate standing or consent of instructor.

This course is intended to introduce neuroscience graduate students to mathematical concepts in neuroscience.  We will use experimental observations in neuroscience to motivate the study of mathematics, both to quantify the observed data and build biophysical models. The course will focus on five fundamental topics in mathematical neuroscience, with emphasis on quantifying neurophysiological time series and developing mathematical models of the activity observed. An important component of the course will include an introduction to scientific computing. Students completing this course will develop computational skills essential in interdisciplinary neuroscience research and in more advanced neuroscience courses offered at BU.

**Course goals**

To introduce some mathematical concepts encountered in neuroscience research and more advanced neuroscience graduate courses. To teach basic programming skills. To think about neuroscience in quantitative ways.

**Course requirements**

The main requirement in this course is effort. I expect your full effort during our course meetings, and outside of the course, to meet the course objectives. As part of this course, you will work together in teams of 3-4. We’ll establish those teams during the first week of MA665. You will work with your team to complete a series of examinations in this course.

**Grades**

To earn an A in MA665, your team must pass oral examinations in five topics:

* Programming proficiency
* The integrate and fire neuron, and its extensions
* The Hodgkin-Huxley neuron
* The power spectrum
* The coherence

You may choose to take one exam during any lab. During an exam, I will ask your team a series of questions about your chosen topic. These questions will include: basic facts, conceptual questions, and practical programming implementation questions. Each question will be posed to one team member – the selected team member must answer the question without assistance from other team members. Please do your best to answer each question. If your team successfully answers each question, then your team completes the exam. If not, I will recommend topics for additional study. You may repeat an exam as many times as you like during the course, but you may only take one exam per lab.

**Schedule**

In this course, you choose your own path. A suggested schedule is provided below. You may undertake this material in any order (e.g., start with Week 4), but I recommend starting with Week 1, and progressing in the order of weeks. You will complete this course at your own pace, although I recommend you complete all 5 oral exams by the end of MA665. I recommend viewing online lectures and completing reading outside of lab. During lab, I recommend attempting suggested assignments in MATLAB, asking questions, and completing the oral exams.

Completing the exams will provide you with the minimum requirements for an introduction to a subset of topics in computational neuroscience. After you complete these exams, you are free to pursue topics of your choice. This might include a survey of different topics in computational neuroscience, or a deep dive into a specific topic. For example, if you are new to MATLAB, you might continue to build your knowledge in this area. Or, if you are interested in computational modeling, you might investigate other models or methods of modeling. Or, if you are interested in data analysis, you might explore traditional or novel methods, and their practical application. When you complete all five exams, let’s chat.

**Suggested Course Schedule**

Readings: These papers provide a general context and motivation for the application of mathematics and statistics to problems in neuroscience. Please read these papers at your leisure during the next 7 weeks.

* Gerstner, W., Sprekeler, H., & Deco, G. (2012). *Theory and simulation in neuroscience*. Science, 338(6103), 60–65.
* Marder, E. (2015). *Understanding brains: details, intuition, and big data*. PloS Biology, 13(5), e1002147
* Churchland, A. K., & Abbott, L. F. (2016). *Conceptual and technical advances define a key moment for theoretical neuroscience*. Nature Neuroscience, *19*(3), 348–349.

Advanced Reading: This paper provides a big-picture discussion of computational neuroscience and related topics. Important ideas to consider for your future career in neuroscience.

* Brain 2025: A Scientific Vision, 2014

Depressing reading:

* Saving Science: Science isn’t self-correcting, it’s self-destructing. Daniel Sarewitz, 2016.
* Button, K., et al (2013). *Power failure: why small sample size undermines the reliability of neuroscience.* Nature Reviews Neuroscience, 14(5), 365–376.

**Week 1**

Sept 4: Introduction & course goals, computer set-up, interviews, teams.

Sept 6: Create teams, Programming setup and basics.

Materials

Reading:   ~~(pdf) Chapter 1 @ Kramer & Eden,~~ *~~Case studies in neural data analysis~~*~~, 2016.~~

~~(lib) Chapters 1 and 2 @ P. Wallisch et al,~~ *~~MATLAB for neuroscientists~~*~~, 2009.~~

~~(pdf) Chapter 4 @ M. X. Cohen,~~ *~~Analyzing neural time series data~~*~~,~~~~2014.  
 Code:~~ [~~http://www.mikexcohen.com/book/index.html~~](http://www.mikexcohen.com/book/index.html)

Tutorial: (pynb) “[01. Introduction to Python](https://github.com/Mark-Kramer/Case-Studies-Python/tree/master/01.%20Introduction%20to%20Python)”

**Week 2**

Sept 11 & 13: The integrate and fire neuron, and its extensions.

Materials

Reading: (web) W. Gerstner, the integrate-and-fire model.

Link: <http://lcn.epfl.ch/~gerstner/SPNM/node26.html>

(pdf) Abbott, Brain Res Bull (1999) vol. 50 (5-6) pp. 303-4  
(pdf) Chapter 8, pages 267-269 @ E. Izhikevich, *Dynamical Systems in Neuroscience*, 2007.

(pdf) Chapter 1, pages 5-12 @ C. Koch, *Biophysics of computation*, 1998.

(pdf) Chapter 14, pages 330-341 @ C. Koch, *Biophysics of computation*, 1998.

(lib) Chapter 22 @ P. Wallisch et al, *MATLAB for neuroscientists*, 2009.

Videos: (lec) M. Kramer, *Introduction to the integrate and fire model* (Neural Spike Train Analysis 4)  
 NOTE: Slides available as PDF on github.

Link: <https://www.samsi.info/news-and-media/27-jul-drs-m-kramer-and-u-eden-samsi/>

(lec) W. Gerstner, *Passive membrane and Integrate-and-Fire model (a)*

Link: <http://klewel.com/conferences/epfl-neural-networks/index.php?talkID=1>

Code (pynb) Week\_2\_MA665.m

**Week 3**

Sept 18 & 20: The Hodgkin-Huxley neuron

Materials

Reading: (pdf) Chapter 2, pages 25-42 @ E. Izhikevich, *Dynamical Systems in Neuroscience*, 2007.

(pdf) Chapter 6, pages 142-159 @ C. Koch, *Biophysics of computation*, 1998.

(pdf) Hodgkin-Huxley 1-page cheat sheet

(web) W. Gerstner, *Detailed Neuron Models* (sections 2.1 & 2.2)

Link: <http://lcn.epfl.ch/~gerstner/SPNM/node12.html>

(pdf) Advanced: Hodgkin and Huxley, J Physiol (Lond) (1952) vol. 117 (4) pp. 500-44.

Videos: (lec) M. Kramer, *Introduction to the Hodgkin-Huxley neuron* (**Neural Spike Train Analysis 5**)

NOTE: Slides available as PDF on github.

Link: <https://www.samsi.info/news-and-media/27-jul-drs-m-kramer-and-u-eden-samsi/>

(lec) W. Gerstner, *Detailed Neuron Model (a)*

Link: <http://klewel.com/conferences/epfl-neural-networks/index.php?talkID=4>

(lec) W. Gerstner, *Detailed Neuron Model (b)*

Link: <http://klewel.com/conferences/epfl-neural-networks/index.php?talkID=5>

Code: (pynb) Week\_3\_MA665.m

(mat) HH0.m

**Week 4**

Sept 25 & 27: The power spectrum

Materials

Readings: (pdf) Chapter 3 @ Kramer & Eden, *Case studies in neural data analysis*, 2016.

(pdf) Kramer, SFN Short Course Document.  
 (pdf) Chapter 10 @ M. X. Cohen, *Analyzing neural time series data*,2014.

(pdf) Chapter 11 @ M. X. Cohen, *Analyzing neural time series data*,2014.

(lib) **Advanced**: Chapter 4 @ Percival & Walden, *Spectral Analysis for Physical Applications.*

Tutorial: (pynb) “[03. Analysis of Rhythmic Activity](https://github.com/Mark-Kramer/Case-Studies-Python/tree/master/03.%20Analysis%20of%20Rhythmic%20Activity)”

Code: (mat) Week\_4\_MA665.m

**Week 5**

Oct 2 & 4:        The coherence

Materials

Readings: (pdf) Chapter 5 @ Kramer & Eden, *Case studies in neural data analysis*, 2016.

(pdf) Kramer, SFN Short Course Document.

(pdf) Chapter 25 @ M. X. Cohen, *Analyzing neural time series data*,2014.

(pdf) Chapter 26 @ M. X. Cohen, *Analyzing neural time series data*,2014.

Videos: (MXC) “Intro to connectivity, volume conduction, and time- vs. trial-based connectivity”

“Phase-based connectivity analyses”  
 <http://www.mikexcohen.com/lectures.html>

Code: (mat) Week\_5\_MA665.m

**Week 6**

Oct 11: Topic exams

(Note: No class Oct 9 – Columbus Day)

**Week 7**

Oct 16: Special topics lecture: *Introduction to github* (Dr. Louis-Emmanual Martinet, Harvard/MGH)

Oct 18: Topic exams

**MA666: Advanced Modeling and Data Analysis in Neuroscience (Fall 2016)**

**Instructor:** Mark Kramer (mak@math.bu.edu)

**Course Hours:** October 23, 2018 – December 11, 2018 [7 weeks]

Tuesday & Thursday, 12:30-1:45 PM, CAS B25B

**Office Hours:** TBD, Math & Stats Room 224, or by appointment

**Textbook:** None

**Course Website:**by invitation (github)

**Prerequisites:** Graduate standing or consent of instructor.

**Course goals**

The goal of this course is a deep dive into a topic in computational neuroscience of your choice. Please work in teams of 2-4. You may have already started this project in MA665. If so, then keep going. There are three broad paths: (1) computer programming, (2) data analysis, (3) modeling. Your team will create a deliverable and present its work to the class at the end of the semester.

**Course requirements**

The main requirement in this course is effort. I expect your full effort during our course meetings, and outside of the course, to meet the course objectives. As part of this course, you will work together in teams of 2-4. You will work with your team to investigate a topic in computational neuroscience of your choice. To earn an A in MA666, please complete the following items:

* By Thursday October 27 (Week 8) your research team is required to discuss your research topic with me.
* By Thursday November 3 (Week 9) your team is required to submit a one-page summary of your project. Please discuss in your summary: (1) Background of project. (2) Motivation for choosing your topic. (3) Plan to complete project. (4) Expected project deliverable (e.g., a new piece of code, a how-to guide for your peers, an informative wiki or web video, etc).
* December 6 and 8 (Week 14) – Presentation. Your team will present its project to the class. You will have (approximately) 20 minutes for the talk + 5 minutes for questions. You must create an electronic version of your talk (e.g., PowerPoint, Google Slides, etc). You must also share your “deliverable” – and make it accessible – to the entire course.
* December 6 and 8 (Week 14) – Deliverable. As part of your project, please submit your “deliverable”. The deliverable could be a github repository with MATLAB or Python code, an informative web/Wikipedia page, an instructional web video, a challenging series of programming exercises with answers, etc. The goal of the deliverable is to develop something useful to your peers and future classes of MA665/6. You must make your deliverable accessible to the entire course.

**Project suggestions**

You are free to select a project of your choice. Here are some suggestions:

* Define and compare two or more coupling measures or cross frequency coupling measures or another set of measures commonly used in neuroscience. Explain each measure, and develop MATLAB code to compute each measure. Apply these measures to example data sets and compare the results. Create a publically accessible github repository to share your results. Develop a problem set to test your peers’ knowledge of the examined measures.
* Define and compare two or more reduced neural models commonly used in neuroscience. Explain each model, and develop MATLAB code to simulate each model. Compare these models and how they behave under different simulation circumstances. Create a publically accessible github repository to share your model and results. Develop a problem set to test your peers’ knowledge of the simulated models.
* Implement a biophysically realistic neural model. Describe all features of the model and its implementation in a wiki. Determine how the model behaves when different biophysical features are altered.
* Develop an online tutorial for the software package XPPAUT. The tutorial should provide step by step instructions (including code, text, and images) to allow your peers enough background to implement a neural model, simulate it in XPPAUT, and display useful dynamical quantities (e.g., the phase plane, nullclines, and fixed points). Develop a problem set to test your peers’ knowledge of the examined measures. Advanced: As part of your tutorial, include a discussion and example of AUTO.
* Develop an instructional web video discussing a topic in computational neuroscience suitable for an open online course (OOC). Your video and accompanying text should define the topic (in words and equations), and include questions to test the learner’s knowledge.
* If you are new to computer programming, develop a series of useful introductory programming tutorials for MATLAB. These tutorials should include detailed examples and explanations of MATLAB code. Produce both a written document, web videos (posted on YouTube), and example questions that can be directly incorporated into the OOC from Week 1.

**Schedule**

You will spend most of your time working on your team’s project. I recommend using the time in lab to ask me and the TF questions, and discussing areas where you’re stuck. In addition, your team (or group of teams) may request optional mini-tutorials. Some topics are listed below. Please notify me one week in advance if you’d like me to organize a mini-tutorial.

**Optional mini-tutorial topics**

An introduction to:

* cross frequency coupling.
* spike-field coherence.
* the multi-taper method.
* bootstrapping.
* fixed points and nullclines in a simple neuronal model.
* the software XPPAUT.
* modeling the gamma rhythm.
* adding funky currents to neural models.
* other?

**Detailed Schedule**

**TOPICS**

* cross correlation and coherence
* CFC
* spike-field coherence
* ING, PING, sparse PING
  + beta?
* Neural networks and backpropagation

**Week 8**

Oct 23 & 25: Cross-frequency coupling

**Week 9**

Oct 30 & Nov 1: Spike-field coherence

**Week 10**

~~Nov 6: No class SFN~~

Nov 8: Neural networks and learning

**Week 11**

Nov 13 & 15: Neural networks and learning**,** Backpropagation

**Week 12**

Nov 20: Backpropagation

~~Nov 22: No class Thanksgiving~~

**Week 13**

Nov 27 & 29: Real-life models: gamma (ING, PING, sparse PING).

**Week 14**

Dec 4 & 6: ?

Week 15

Dec 11: ?