COGS 108: Neural Analytics

Professor Ramesh Srinivasan, Cognitive Sciences

Version 0 09/22/22

Course meetings: TTh 2:00-3:20
Course location:SE2 1306 E-mail: r.srinivasan@uci.edu
Course Web: **https://canvas.eee.uci.edu/courses/49498**
Office Hours: **M 2 or by appointment**
Office: SBSG 2336

Course Description

Introduces the theoretical foundations and practical applications of neural data analysis. Topics include linear systems theory, signal processing for neural time series, supervised and unsupervised learning of structure in brain data, and machine learning applications in neuroscience. The course will take a hands on approach using simulation and data analysis with an emphasis on methods relevant to human EEG (MEG) and fMRI (fNIRS).

Prerequisites

COGS/PSYC 10A/B/C OR STATS 110 OR STATS 7
COGS/PSYC 14P or COGS/PSYC 14M or ICS 31 or PSYC 114M or MATH 9 or EECS 12

A Python or Matlab course is a prerequisite for this course. However, this course will make use of Python. I will provide materials to help Matlab users transition to Python, e.g., **https://numpy.org/doc/stable/user/numpy-for-matlab-users.html**

All of the Lecture Notes I will provide incorporate Python code for you to execute. I will spend more time at the start of the class explaining the programmming details which will facilitate the transition from Matlab to Python or reviving dormant Python skills.

In addition, I will provide Lecture Notes from my course COGS 14P, for your review.

Textbook

There is no textbook for this class. There will be articles to read from the scientific literature.

Weekly Lecture Notes and Exercises will be provided as Jupyter Notebooks using the Python programming language. Jupyter Notebooks combine text (written in markdown) and code to create **interactive** Lecture Notes that allow you to execute and modify computer programs to build your intuition about the simulation and data analysis methods covered in this course.

All the Lecture Notes in this class will provide example code which you will execute and create the figures within the Lecture Notes, matching the figures I present in the Lectures. Your homework Exercises will ask you to write Python code in order to solve practical simulation and data analysis problems modeled after the Lecture Note examples.

Data Analysis Project

The major component of this course (and your grade) is a data analysis project. The goal of this project is to take the methods developed in the course and apply them to new data and to interpret the results of these analysis in a final report, submitted at the end of the quarter.

Data Sets

I will provide some options for the data analysis projects consisting of data sets I have curated.

However, I recognize that my scientific interests are not the same as yours, so I am quite open to a final project that you define. This may come from data sets that you have access to because of your participation in research labs on campus. It may also come from data sets you identify from a search through data archives such as openneuro.org.

If you choose to use your own data, you must get your proposed data analysis plan approved by me to establish the reasonable scope of the project. If the data comes from a research lab, you must also ask for permission to use the data for this class, and for me to have access to the data for evaluating your project.

Scope of Work

The final project should incorporate some basic visualization of the data, and some of the techniques discussed in the class, and *some techniques not discussed in class*. One of the goals in this class is for you to have the ability and confidence to explore novel data analysis methods. I will suggest methods to explore for each project.

Final Report

The final report should be submitted in the form of a Jupyter notebook that provides: (1) text (formatted in markdown) that a brief background and explanation of the data (1-2 paragraph) (2) an explanation of the objective of each analysis block (3) CODE I can run on your data that runs the analysis and makes each graph or returns each quantitative result (4) an explanation of the meaning of the graph or result.

Collaboration

You are allowed to collaborate with others in the class in the data analysis project. However, I expect there to be independent work on programming and for you to submit independent

write-ups.

Software

Anaconda Python

All of the exercises and projects in this class make use of the Python programming language. Install Anaconda Python Libraries on your (Windows, Mac, or Linux) computer.

Anaconda Python - https://www.anaconda.com/products/individual

Why Anaconda Python?

Your computer may already have Python installed on it, which in principle you could configure to use for this class. It is much easier to simply install the Anaconda distribution of Python which is free and comes with an nearly complete library of software for scientific computing as well as a number of other useful tools like the *Jupyter Lab* IDE.

VS CODE

My preferred IDE for Jupyter Notebooks is Visual Studio Code.

https://code.visualstudio.com/

There is no requirement to use VS code. You can use Jupyter Notebooks or Jupyter Lab, if you are already used to them and like them. Some people like PyCharm. All of the materials in this class should work in any IDE.

GitHub

This course has a GitHub repository. My preference would be to use the GitHub repository for distribution of course materials versus using CANVAS which I kind of hate.

https://github.com/rameshsrinivasanuci/COGS108

What would make this the most effective is if you were to make use of git software to interact with the repository. This can either be done with commmand line tools (better) or using GitHub Desktop

The official instructions for setting up command line tools are available here -

https://git-scm.com/book/en/v2/Getting-Started-Installing-Git https://git-scm.com/book/en/v2/Getting-Started-Installing-Git

I actually thought this unofficial one is good too.

https://www.atlassian.com/git/tutorials/install-git

I would be up for trying to get this to work, and helping you do so. But, I also know there is overhead to doing it this way, so you might want to take an easier option of using GitHub Desktop, which I have been told is just as good.

https://desktop.github.com/

Course Structure

Weekly Course Topics

1. Orientation to Python, Jupyter Notebooks

2. Measurement Methods, Design of Experiments, Linear Systems
   Event Related Potentials, Hemodynamic Response Functions

3. Time Series Properties, Digital Filters
   How to Filter Data Properly

4. Spectrum of a Signal, Fourier Analysis
   EEG Spectrum and Spectrogram

5. Functional Connectivity in EEG and fMRI
   Geometry of Multivariate Neural Data

6. Unsupervised Learning Methods - Uncovering Latent Structure
   Principal and Independent Components Analysis

7. Supervised Learning Methods and Classification
   Linear Classification in neural data

8. Introduction to Machine Learning Methods
   Support Vector Machines and Ensemble Classifiers

9. Introduction to Neural Networks for Neural Data Analysis

10. Final Project in Class

Homework

Homework will be assigned weekly, due one week later. After submitting the homework, the answer key will be released, and you will be allowed to correct your homework. The only way you dont get full credit is if you don't submit an original homework, which is at least attempted and then a corrected homework.

Final Exam

There is no Final Exam for this course. **Final Project Draft is due Friday Dec 2. I will provide feedback on what else needs to be done. Final Project is due on Friday Dec 9, 5 pm.**

Grading Policy

Homework provided as Jupyter Notebooks: 50% of grade.
Data Analysis Project: 50% of grade.

Disability services, academic dishonesty, and copyright policy

Disability Services link: https://dsc.uci.edu/
Academic Dishonesty link: https://aisc.uci.edu/students/academic-integrity/index.php
Copyright policy link: http://copyright.universityofcalifornia.edu/use/teaching.html