Quantitative Neuroscience Core

Instructor	Joshua Gold	jigold@pennmedicine.upenn.edu
	Bijan Pesaran	Bijan.Pesaran@Pennmedicine.upenn.edu
TA	Eren Kafadar	Eren.Kafadar@Pennmedicine.upenn.edu
Required meeting times	MWF	9–10 am
Optional discussion section/office hours	TBD	
Meeting location	Class of '62 Auditorium	

Introduction

This course is designed to be an overview of quantitative approaches used for rigorous and reproducible neuroscience research. This course does not cover statistics in a traditional way, in the sense that we will not provide a comprehensive survey of statistical tests, nor will we dive very deeply into formal mathematical derivations of those tests (information about such things can be found in textbooks and all over the web). Instead, we will focus on teaching you to apply quantitative approaches to your thinking about neuroscience research from beginning to end, including defining clear hypotheses; designing experiments to test those hypotheses; collecting, visualizing, analyzing, and interpreting data in reference to those hypotheses; and keeping effective and transparent records at each stage to ensure rigor and reproducibility.

There are two main components to the course. The first component consists of a series of modules, each of which is designed to use a specific example from neuroscience to illustrate a set of quantitative approaches and tools. The second component consists of group projects that focus on designing and implementing quantitative analyses for existing data sets (e.g., from your rotation project).

Learning Objectives

1) Develop good habits for transparent, reproducible science. Transparency is the idea that none of your data or methods should be hidden. Reproducibility is the idea that you should be designing, conducting, and analyzing experiments in a way that maximizes the probability that someone else doing the same experiments would come to the same conclusions. To support these ideas, we will incorporate into the course the use of several on-line tools that, even if you do not end up using these particular tools in your own research, will help establish good habits for record keeping (we will use LabArchives electronic notebooks, https://researchnotebooks.upenn.edu), version control for code (we will use GitHub, https://github.com), and data storage (we will use PennBox: https://upenn.app.box.com). Please register the names/links for your accounts here.

2) Learn to think about statistics in the context of good experimental design. The question "what statistical test should I use?" can be answered only after answering more basic questions, like "what are the alternative hypotheses that I am testing?" and "how well does my experimental design allow me to distinguish those hypotheses?"

3) Learn foundations of statistical reasoning, particularly how to think about randomness using probability distributions. Even the most sophisticated statistical procedures are ultimately about distinguishing signal from noise. This ability depends on understanding what is meant by "noise", or randomness. The primary mathematical tool for quantifying and manipulating randomness is the probability distribution, which describes the probability of obtaining all possible values of a quantity of interest (e.g., the outcome of an experiment). We therefore will spend some time learning about probability distributions and then build on those concepts to better understand how to use probability distributions to make inferences.

4) Learn to visualize your data effectively to lay bare your statistical reasoning. Ultimately your ability to convince other people that you have a robust finding will not depend on the results of a statistical test but rather on your ability to show the finding in a clear and compelling way; that is, in a way that is transparent in terms of what you measured, clearly reflects the experimental design, and illustrates both the signal and noise that you found. We will focus on specific ways to visualize data effectively throughout the course.

Course Resources

This course is just one component of a more comprehensive training program that we are developing that covers quantitative approaches in neuroscience. A centerpiece of this training program is a curated list of resources and collection of readings. Our goal in this course is to cover some of these materials in detail but more generally get you familiar with some of the key ideas so you can continue to return to these materials when you need them in your research.

Coding

Many of the topics covered in this course are implemented in either Matlab or Python (course materials are here). Ask anyone who uses either language and they will doubtless give you long, passionate explanations of why one is better than the other, but here suffice it to say that both are used in research labs and so we want to give you exposure to both. In class we will mostly run the Python-based demonstrations, because they are implemented in a web-based notebook format called Colaboratory that is easy for even non-programmers to use. You can choose which language(s) to use for your course-based exercises and projects.

If you do not already know how to program in either language, you should still be able to follow along with the logic and learn a bit as you go. We will provide some instruction, and it is reasonable to assume that you could go from a complete novice to someone who can write some basic code. However, do not expect the course alone to teach you how to program. Here are some resources that can help you:

Resources for Learning Matlab

- From Mathworks: Getting started
- Coursera: Learn Matlab
- Wallisch et al, *Matlab for Neuroscientists*
- The summer Matlab course offered by BGS

Resources for Learning Python

- From Python.org: Python for beginners
- Coursera: Python courses
- Kaggle: Introduction to Programming in Python, Learning Python
- A useful refresher for basic Python syntax
- The summer Python course offered by BGS

Use of Generative AI

For this course, the use of generative AI tools like ChatGPT are explicitly encouraged (see the Handbook, section 1.7.4, for guidelines for the use of these tools in the NGG). The idea is that writing code is one case where these tools have the potential to be quite useful, particularly for those with minimal coding experience. There is so much code out there that has been used to train these models that in many cases, good solutions to well-specified coding problems can be obtained quickly and easily. Moreover, these tools can be used to convert code between different languages (e.g., Matlab and Python) and provide the kind of thorough commenting and documentation that is often lacking in custom research code. All that said, there are also major potential pitfalls to using these tools, the main one being that there is absolutely no guarantee that any code they provide actually does what you want it to do. Our goal this semester is to work together to see if and how these tools can best be used to allow even non-expert coders to produce reliable, readable code for quantitative data analysis.

Homework

Most class sessions in the first component of the course ("Foundations") have homework assignments, listed in the right-most column below. Please complete each assignment in advance of the session for which it is listed – they are designed to give you enough of a starting point to get the most out of the in-class lectures and be a full participant in the in-class discussions.

The assignments highlighted in **red** are the ones for which we will dedicate extra in-class sessions to go over the answers as a group, with a focus on if and how each of you used generative AI to produce your answers. Think of this as the crowd-sourcing component of the course: we think that the best way to figure out best practices for using these tools is to get all of you to use them in different ways and then together discuss advantages and disadvantages of those approaches.

Grading

Grades are based on: 1) completion of homework assignments, including posting your results for the exercises in **red** to the appropriate notebook and/or repository (20%); 2) class participation, including engagement in discussions (20%); and 3) a final project involving two in-class presentations (20% each) and electronic records of analysis strategies and code (20%).

For our philosophy of grading, see here.

Syllabus

PART 1: FOUNDATIONS

Wed	**27-Aug**	**Introduction I: Overview, Goals, and Record Keeping**	**Read and be prepared to discuss:** Record Keeping: Introduction Record Keeping: Laboratory Notebooks Record Keeping: Algorithms Record Keeping: Data **Sign up for the following accounts (if you haven't already) and comfirm on this spreadheet:** QNX 2025 Record Keeping LabArchives (through Penn) GitHub PennBox (through Penn) **Come prepared to discuss this article:**

			Six tips for better coding with ChatGPT

Fri	**29-Aug**	**Introduction II: Inference and Statistics**	**Readings:** 1. Platt, J.R. (1964) Strong Inference: Certain systematic methods of scientific thinking may produce much more rapid progress than others. Science146, 347-353. **Add to Canvas an example from your own lab experiences, or from a study you have learned/read about, of either: 1) strong inference, or 2) not strong inference** 2. Pelham, B.W. (2012). Intermediate Statistics: A Conceptual Course, Chapter 1: A Review of Basic Statistical Concepts. **Come prepared to describe from the chapter: a) a topic that you have already learned/understand well, and b) a topic that is new to you and/or is not clear from the description in the paper.**

Mon	**1-Sep**	**LABOR DAY -- NO CLASS**	

Wed	**3-Sep**	**Introduction III: Frequentist versus Bayesian Approaches**	**Read and be prepared to discuss (focus on Figures 1 and 2):** Keysers et al (2020), Using Bayes factor hypothesis testing in neuroscience to establish evidence of absence **Go through the following tutorial and complete exercises 1 and 2.** Frequentist versus Bayesian approaches

Fri	**5-Sep**	**Probability Distributions I: Concepts and examples (Bijan)**	**Go through the following tutorials, then: 1) find a paper that shows data thought to come from one of these distributions, and 2) write code to simulate data that (roughly) match the distribution shown in the paper.** Samples and Populations Probability Distributions Overview

			Bernoulli Distribution Binomial Distribution Gaussian (Normal) Distribution Student's t Distribution

Mon	**8-Sep**	**Probability Distributions II: Distributions of events that unfold in time (Bijan)**	**Go through the following tutorials, then: 1) find a paper that shows data thought to come from one of these distributions, and 2) write code to simulate data that (roughly) match the distribution shown in the paper.** Binomial Distribution Exponential Distribution Poisson Distribution

Wed	**10-Sep**	**Probability Distributions III: Confidence Intervals and Bootstrapping (Bijan)**	**Go through the following tutorial, (including exercises):** Confidence Intervals and Bootstrapping

Fri	**12-Sep**	**Probability Distributions: Quantal Release exercise review (Eren)**	**Complete the exercises from the Neuroscience Example ("Quantal Release") case study in the following tutorial and post your answers to GitHub. Be prepared to discuss your code, including if/how you used generative AI.** Binomial Distribution

Mon	**15-Sep**	**Two-Sample Inference I: Experimental Design and Power Analysis**	**Read and be prepared to discuss:** Button et al (2013), Power failure: why small sample size undermines the reliability of neuroscience **Go through the following tutorial, then complete the Exercises and post your answers to GitHub:** Error Types, P-Values, False-Positive Risk, and Power Analysis

Wed	**17-Sep**	**Two-Sample Inference II: Parametric Tests and Multiple Comparisons**	**Complete and be prepared to discuss this Colab tutorial:** t-tests
Fri	**19-Sep**	**Two-Sample Inference III: Nonparametric Tests**	**Complete and be prepared to discuss these Colab tutorials:** Simple Non-Parametric Tests Pearson's chi-square test (in Proportions)
Mon	**22-Sep**	**Two-Sample Inference: Multiple Comparisons exercise review (Eren)**	**Complete the exercises from the following tutorial and post your answers to GitHub. Be prepared to discuss your code, including if/how you used generative AI.** Multiple comparisons
Wed	**24-Sep**	**Measures of Association I: Correlation**	**Go through the following tutorials, then complete the parametric correlation coefficient exercises and post your answers to GitHub.** Measures of association Parametric correlation coefficient Nonparametric correlation coefficient **Optional: Review the code in the NGG GitHub Repository under "Examples/LC-Pupil/" that was used to generate Fig. 3 of Joshi et al.**
Fri	**26-Sep**	**Measures of Association II: "Nonsense correlations"**	**Read and be prepared to discuss:** Nonsense Correlations in Neuroscience Code to generate figures is here
Mon	**29-Sep**	**Measures of Association III: Simple Linear Regression**	**Go through the following tutorials, then complete the linear regression exercises and post your answers to GitHub.** Measures of association Simple linear regression

Wed	**1-Oct**	**Measures of Association: Linear Regression exercise review**	**Be prepared to discuss your Linear Regression code, including if/how you used generative AI**
Fri	**3-Oct**	**Modeling I: Overview of computational modeling**	**Read and be prepared to Discuss (focus on sections "What is computational modeling of behavioral data?", "Design a good experiment!", "Design good models", and "Fit the parameters"):** Wilson and Collins (2019)
Mon	**6-Oct**	**Modeling II: LATER case study**	**Read and be prepared to discuss (focus on Figures 1–3):** Noorani (2014) **Some more readings just for fun:** RT at Penn I RT at Penn II RT at Penn III
Wed	**8-Oct**	**Modeling II: RT Data Visualization**	**Run and be prepared to discuss these Matlab tutorials:** LaterTutorial_plotExampleData.m LaterTutorial_dependenceOnModelParameters.m
Fri	**10-Oct**	**BGS REUNION -- NO CLASS**	
Mon	**13-Oct**	**Modeling: Model Fitting exercise Review**	**Complete the exercises from the following tutorial and post your answers to GitHub. Be prepared to discuss your code, including if/how you used generative AI.** LaterTutorial_fitModelToDataExercise.m
Wed	**15-Oct**	**Data Visualization I: Principles (Eren)**	
Fri	**17-Oct**	**Data Visualization II: Examples (Eren)**	**Find a figure/graph from a paper you think displays the distribution of their data well or poorly. Post it in the Canvas course discussion.**

PART 2: APPLICATIONS (STUDENT PRESENTATIONS)

Mon	20-Oct	PRESENTATION 1: HYPOTHESES AND EXPERIMENTAL DESIGN
Wed	22-Oct	
Fri	24-Oct	
Mon	27-Oct	
Wed	29-Oct	
Fri	31-Oct	
Mon	3-Nov	
Wed	5-Nov	
Fri	7-Nov	
Mon	10-Nov	

Wed	12-Nov	PRESENTATION 2: VISUALIZATION AND HYPOTHESIS TESTING
Fri	14-Nov	
Mon	17-Nov	SFN -- NO CLASS
Fri	19-Nov	SFN -- NO CLASS
Fri	21-Nov	
Mon	24-Nov	
Fri	26-Nov	THANKSGIVING -- NO CLASS
Fri	28-Nov	THANKSGIVING -- NO CLASS
Mon	1-Dec	
Wed	3-Dec	
Fri	5-Dec	
Mon	8-Dec	
Wed	10-Dec	
Fri	12-Dec	