Computational Cognitive Modeling

Brenden Lake

Brenden Lake

Assistant Professor, Data Science and Psychology
Research Scientist, Facebook AI Research

office hours: Wednesdays 4:30-5:30pm, zoom https://nyu.zoom.us/my/brenden

https://cims.nyu.edu/~brenden
https://lake-lab.github.io/

Yanli Zhou

PhD student, Data Science

office hours: TBD

Guy Davidson

PhD student, Data Science

office hours: TBD


LOGISTICS

Course website

https://brendenlake.github.io/CCM-site/

Computational cognitive modeling - Spring 2022

NYU PSYCH-GA 3405.004 / DS-GA 1016.003

View On GitHub

This project is maintained by brendenlake

Computational cognitive modeling - Spring 2022

Instructor: Brenden Lake

Teaching Assistants: Yanli Zhou and Guy Davidson

Meeting time and location:
Lecture. Our course will use a “flipped classroom” model. Lectures will be pre-recorded and available to watch on Vimeo (password on EdStem), which is followed by a live discussion of the material in person (or on zoom). Please watch the lecture *before* the date it is listed under, so that we have a productive live discussion. Please come ready with your questions.
Live discussion of the lectures is on **Mondays 10-11 AM**. You can attend in person at 60 5th Ave Room 150. There is also a zoom option (links available on the class brightspace and EdStem).

Labs. Tuesdays 1:30-2:20 PM. You can attend in person at the Silver Center Room 520 or on zoom (links available on class brightspace and Edstem).

Waitlist and auditors. You shouldn’t need access to the brightspace. All of the key info should be pinned on the class EdStem.

Course numbers:
DS-GA 1016 (Data Science)
PSYCH-GA 3405.004 (Psychology)

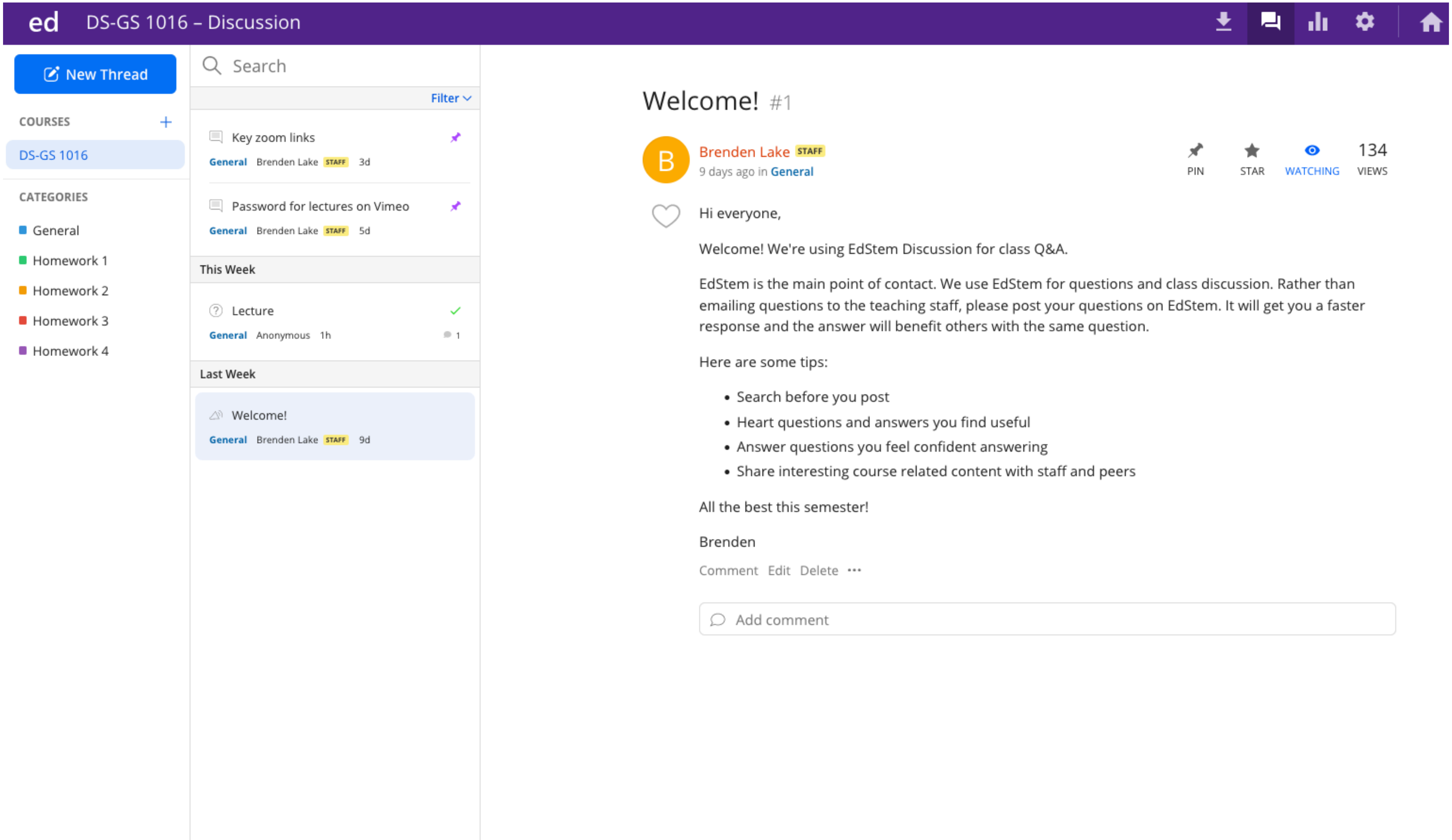
Contact information and EdStem:
EdStem is the main point of contact. We use EdStem for questions and class discussion. Rather than emailing questions to the teaching staff, please post your questions on EdStem. It will get you a faster response and the answer will benefit others with the same question.

The signup link for our EdStem page is available here (https://edstem.org/us/join/KPspc2).

Once signed up, our class EdStem page is available here (https://edstem.org/us/courses/18607/discussion/).

If you have a question that isn't suitable for EdStem and there is a need to email the teaching staff

Course discussion: EdStem



Readings posted on EdStem

ed DS-GA 1016.003/PSYCH-GA 3405.004 – Resources

Search

Bayesian Modeling | Categorization | Computational Cognitiv... | Model Comparison | Neural Networks | Probabilistic Graphical ... | Program Induction/Lan... | Rational versus Mechan... | Reinforcement Learning

Upload Resource | Add Link

Bayesian Modeling

Ghahramani_-_2015_-_Probabilistic_machine_learning_and_artificial_intelligence	PDF
Russell_Norvig_AIMA_Ch13	PDF
Tenenbaum,_Griffiths_-_2001_-_Generalization,_similarity,_and_Bayesian_infer...	PDF
Tenenbaum_et_al._-_2011_-_How_to_Grow_a_Mind_Statistics,_Structure,_and_...	PDF
mackay_monte_carlo	PDF

Categorization

Love,_Medin,_Gureckis_-_2004_-_SUSTAIN_A_Network_Model_of_Category_Lea...	PDF
Marr__Vision__Ch_1	PDF

Brightspace

We won't use it for much, unless you want to watch cloud recording

If needed, auditors and folks on the waitlist can get added to brightspace. Please add your email to spreadsheet on lab website (link also here,)

Getting in touch

EdStem should be your main point of contact. If you have a question, and you think there is a possibility that someone may have the same question, please post it to EdStem for everyone's benefit.

If you need to send an individual message,

Email address for instructors and TAs:
instructors-ccm-spring2022@googlegroups.com

Flipped classroom

Pre-recorded lectures on Vimeo

(Links on course website) Please *watch before the date they are listed*, so we can have a robust discussion!
(some videos by Prof. Todd Gureckis, who I usually teach this class with)

Blended discussion of lecture:

Mondays 10-11 AM

Zoom (https://nyu.zoom.us/j/98018046361)
OR
60 5th Ave Room 150

Labs

Tuesdays 1:30-2:20 PM
Blended (Silver Center Room 520) or
Zoom: https://nyu.zoom.us/j/97635485906

Lecture schedule

Mon. Jan 24: Introduction
Mon. Jan 31: Neural networks / Deep learning (part 1)
Mon. Feb. 7: Neural networks / Deep learning (part 2)
Mon. Feb. 14: Reinforcement learning (part 1)
Mon. Feb. 21: No class, Presidents' Day
Mon. Feb. 28: Reinforcement learning (part 2)
Mon. Mar. 7: Reinforcement learning (part 3)
Mon. Mar. 14: No class, Spring break
Mon. Mar 21: Bayesian modeling (part 1)
Mon. Mar 28: Bayesian modeling (part 2)
Mon. Apr 4: Model comparison and fitting, tricks of the trade
Mon. Apr 11: Categorization
Mon. Apr 18: Probabilistic Graphical models
Mon. Apr 25: Information sampling and active learning
Mon May 2: Program induction and language of thought models
Mon May 9: Computational Cognitive Neuroscience

Lab schedule

Tue Jan 25, Python and Jupyter notebooks review
Tue Feb 01, Introduction to PyTorch
Tue Feb 08, HW 1 Review
Tue Feb 15, No lab
Tue Feb 22, No lab
Tue Mar 01, Reinforcement learning
Tue Mar 08, HW 2 review
Tues Mar 15, No lab, spring reak
Tue Mar 22, Probability Review
Tue Mar 29, HW 3 Review
Tue Apr 5, TBD
Tue Apr 12, TBD
Tue Apr 19, HW 4 Review
Tue Apr 26, TBD
Tue May 3, TBD
Tue May 10, TBD

Pre-requisites

- *Math:* We will use concepts from linear algebra, calculus, and probability. If you had linear algebra and calculus as an undergrad, or if you have taken Math Tools in the psychology department, you will be in a good position for approaching the material. Familiarity with probability is also assumed. We will review some of the basic technical concepts in lab.

- *Programming:* Previous experience with Python is required. Previous IN CLASS experience with Python is strongly recommend—it's assumed you know how to program in Python. The assignments will use Python 3 and Jupyter Notebooks (http://jupyter.org)

Grading:

- The final grade is based on the homeworks (65%) and the final project (35%). Class participation may be used in cases in borderline grades.

Final project:

- The final project will be done in groups of 3-4 students. A short paper will be turned in describing the project (approximately 6 pages). The project will represent either an substantial extension of one of the homeworks (e.g., exploring some new aspect of one of the assignments), implementing and extending an existing cognitive modeling paper, or a cognitive modeling project related to your research. We provide a list of project ideas (see website), but of course you do not have to choose from this list.

Homeworks — programming requirements

Programming: We assume you are familiar with programming in Python

Homeworks use this setup:

- Python 3
- Jupyter notebooks
- Standard Python packages for scientific computing
  - numpy
  - scipy
  - pandas
  - matplotlib
- PyTorch >=1.4 library for neural networks

Using your laptop setup is encouraged!

Jupyter notebooks

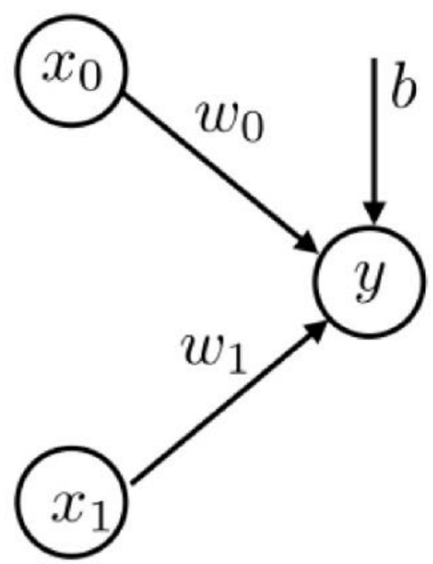
Homework - Neural networks - Part B (20 points)

Gradient descent for an artifical neuron

by *Brenden Lake* and *Todd Gureckis*
Computational Cognitive Modeling
NYU class webpage: https://brendenlake.github.io/CCM-site/
email to course instructors: instructors-ccm-spring2019@nyuccl.org

This homework is due before midnight on Monday, Feb. 25, 2019.

This assignment implements the gradient descent algorithm for a simple artificial neuron. As covered in lecture, the neuron will learn to compute logical OR. The neuron model and logical OR are shown below, for inputs x_0 and x_1 and target output y.



logical OR

x_0	x_1	y
0	0	0
0	1	1
1	0	1
1	1	1

This assignment requires some basic PyTorch skills, which were covered in lab. You can also review two basic PyTorch tutorials, "What is PyTorch?" and "Autograd", which have the basics you need.

In [ ]:
```
# Import libraries
from __future__ import print_function
%matplotlib inline
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
import torch
import torch.nn as nn
```

Let's create `torch.tensor` objects for representing the data matrix `D` with targets `Y`. Each row of `D` is a different data point.

In [ ]:
```
# Data
D = np.zeros((4,2),dtype=float)
D[0,:] = [0.,0.]
D[1,:] = [0.,1.]
D[2,:] = [1.,0.]
D[3,:] = [1.,1.]
```

Pre-configured cloud environment

Students registered for the course have the option of completing homework assignments on their personal computers (encouraged if know how to set it up!), or in a cloud Jupyter environment with all required packages pre-installed (see website).

Collaboration and honor code

We take the collaboration policy and academic integrity **very seriously**. Violations of the policy will result in zero points and possible disciplinary referral.

You may discuss the homework assignments with your classmates, but you must run the simulations and complete the write-ups for the homeworks on your own. Under no circumstance should students look at each other's code or write ups, or code/write-ups from previous years of this course. Do not share your write up or code with any of your classmates under any circumstances.

Course policies

Late work:

- We will take off 10% for each day a homework or final project is late.

See policy on extensions, regrading, extra credit, etc. on syllabus

Background survey

- Currently enrolled in what type of program:
  - Psychology Ph.D.? Psychology Masters? Data Science Masters? DS Ph.D.? Other graduate program? Undergraduate?

- Previous coursework:
  - Cognitive Psychology? Programming? Probability, statistics, MathTools? Machine learning? AI? Deep learning?

- Who knows about:
  - Prototype vs. exemplar models?
  - Categorical perception?
  - Semantic networks?
  - Logistic regression?
  - Backpropagation algorithm?
  - Simple recurrent network?
  - Model-based vs. model-free reinforcement learning?
  - Bayes' rule?
  - Conditional independence?
  - Conjugate prior?
  - Metropolis-Hastings?
  - Explaining away?
  - Probabilistic programming?

What you will come away with…

1. Experience with the major paradigms for computational cognitive modeling

2. An introduction to key technical tools (in Python and Jupyter notebooks):

- Neural networks / deep learning (in PyTorch)
- Reinforcement learning
- Bayesian modeling
- Model comparison and fitting
- Probabilistic graphical models
- Program induction and language of thought models

3. How to build computational models to test and evaluate psychological theories, and to understand behavioral data by modeling the underlying cognitive processes.

4. Ideally, students will leave the course with a richer understanding of how computational modeling advances cognitive science, and how computational cognitive modeling can inform research in data science, machine learning, and artificial intelligence

Is this course a substitute for machine learning?

- **No. It’s not a substitute, it’s complementary.**
- This course does survey various computational paradigms (deep learning, reinforcement learning, Bayesian modeling, classification, graphical models, etc.), and there is some overlap with ML classes in terms of technical content.
- But unlike ML classes, this is also a cognitive science class. **Our examples and applications aim to understand human learning, reasoning, and development, and to understand intelligent behavior more generally.**
- We get into some mathematical background, but ML courses take a more formal approach than we do here. We aim for a more accessible introduction.
- You will get hands on experience with running and analyzing complex models, implementing some (but not all) models, and analyzing behavioral data with computational models. Extensive final project.

For next time....

Readings for the next two lectures (available on EdStem)

- McClelland, J. L., Rumelhart, D. E., & Hinton, G. E. The Appeal of Parallel Distributed Processing. Vol I, Ch 1.
- LeCun, Y., Bengio, Y. & Hinton, G. (2015). Deep learning. Nature 521:436–44.
- McClelland, J. L., & Rogers, T. T. (2003). The parallel distributed processing approach to semantic cognition. Nature Reviews Neuroscience, 4(4), 310-322.
- Elman, J. L. (1990). Finding structure in time. Cognitive Science, 14(2), 179-211.
- Peterson, J., Abbott, J., & Griffiths, T. (2016). Adapting Deep Network Features to Capture Psychological Representations. Presented at the 38th Annual Conference of the Cognitive Science Society.

Homework 1 on neural networks will be released next class

Questions?