MATH 435, SPRING 2024

- **Instructor:** David Aristoff

  `https://www.math.colostate.edu/~aristoff/`

- **Contact:** Find me in Weber 221 or email at `aristoff@colostate.edu`
- **Class meetings:** 9:30-10:45, T/Th, Weber 205, Jan 16 - May 5.
- **Class info:** This course is an introduction to inquiry-based research in mathematics based on open-ended projects, with an emphasis on problem identification and formulation, team approach, and reporting results. The projects will be on topics of your choosing. Projects can be inspired by scientific or more general interests. Students will have the opportunity to become comfortable with coding in Matlab or Python. Whenever possible, bring your laptops to class.

  Detailed class info is on the course overleaf page.

- **Prerequisites:** Some coding, linear algebra, differential equations.
- **Requirements:** Your final grade will be based on:
  - 5% attendance and participation
  - 20% project 1 report
  - 5% project 1 presentation
  - 25% project 2 report
  - 5% project 2 presentation
  - 25% final project report
  - 5% poster session
- **Academic Policies and Integrity:** Students are expected to adhere to the CSU Academic Integrity Policy as found on the Students' Responsibilities page of the CSU General Catalog and in the Student Conduct Code; see `https://resolutioncenter.colostate.edu/conduct/academic-integrity/`. For further policies which apply to all math department classes, please see `https://mathematics.colostate.edu/undergraduate-students/departmental-class-policies/`. You are heavily encouraged to use outside sources (papers, books, software, datasets, etc.); always cite the resources you use.

1 Matlab and Python

You are encouraged to use computing as part of your projects. Through the university, you can get a free Matlab license. Python is freely available online; the Anaconda distribution is probably the most popular and makes it easy to install packages and use notebooks.

Matlab is easier to get started with, and very nice (and very fast) for linear algebra-based code, when properly vectorized. Also, I am very familiar with Matlab and will be able to help you debug your code on a granular level. On the other hand, Python is the language of choice for machine learning, with packages like PyTorch and TensorFlow allowing for easy implementation of a wide range of techniques. If you want to work with artificial neural networks, or pursue a career with a significant coding element, or need to code collaboratively or use a wide variety of packages, Python is the better choice. Matlab is superior if you want to code quickly with less hassle and debugging, or if linear algebra routines are the bottlenecks in your code.

Some Matlab demos I have written are included in this overleaf project.

2 Topic ideas

Here are some possible project ideas, a few of which we discussed in class. Disclaimer: I am only knowledgeable about a few of these topics! Also, this is *not* a comprehensive list of possible projects. Indeed, essentially any topic is possible as long as you can relate it to mathematics. If you have an interest that you don't see how to make mathematical, ask me and maybe we can brainstorm something.

In no particular order:

- Problem banks:
  - Problems from the the Mathematical Contest in Modeling. (New problems are available on Feb 1.)
  - Coding projects in mathematics: see https://projecteuler.net and https://projecteuler.net/archives.
- Sports:
  - Can you predict a team's win-loss record from player stats?

  - How do you pick a team for a fantasy competition?

  See the Baseball Research Journal for other ideas.

- Gambling:
  - Model the *Bringing Down The House* strategy for making money in blackjack. How did the winning strategy work and what countermeasures could the casino have taken?

- Finance:
  - How do you predict the future price of a stock?
  - How do you calculate the value of an option?
  - How do you calculate the risk of an investment?
  - How big must a bank or insurance company's pool of money need to be in order to prevent them from going bankrupt with 99.9% certainty?

- Biology:
  - How do you model the spread of a disease?
  - How can you simulate the process of protein folding?
  - How does AlphaFold uncover folded protein structures?

- Machine learning:
  - How do transformers like ChatGPT work?
  - How do AI image generators work?
  - How are good feature mappings designed?
  - Could interpretable LLMs be based on Koopman theory?

- Smorgasbord:
  - Can you write an algorithm to win as often as possible in snake?
  - How do you arrange a round-robin tournament where each match consists of three competitors (instead of two)? You could ask that each possible triplet competes exactly once, or you could instead ask that a competitor is matched in a triplet with every other competitor exactly once.

– Random walks in $\mathbb{R}^1$, $\mathbb{R}^2$, $\mathbb{R}^3$? There's a simple combinatorial argument for showing random walker in $\mathbb{R}^1$ returns to origin infinitely often. What tweaks can we add?

– Variants on the traveling salesperson problem.

– Voting theory. Arrow's impossibility theorem gives a set of basic axioms that one would want a voting system to satisfy, and then proves that (surprisingly) no such voting system exists. This leaves room for one to consider the pros and cons of various voting systems.

– What is the best way to fit spheres / tetrahedra / dodecahedra / footballs / etc into a box?

– Variants on the lion and contamination problem mentioned in class. See this paper or this blog post. If you have more than n lions moving randomly in an $n \times n$ grid, how long do they have to move until you expect them to have cleared the grid?

– Error correcting codes.

– Cryptography, for example RSA.

– Toppling sandpiles. See figures 1-4 in this article, or the figure on the wikipedia page.

2.1 Updates to list based on questionnaries

Some links:

– LWR traffic flow model (Jack's project)

– Network and topological neuroscience (Casey's project)

– exploring baseball data with R, in particular, using Markov chains to predict pitches, and using Markov Chain Monte Carlo to predict home run rates.

Here are some dates based on your reported interests that are also by no means exhaustive.

- *Statistics*: Financial forecast, geostatistics, environmental modeling, and natural language processing, using tools like Gaussian process regression and neural networks, the Karhunen–Loève transform in statistical signal processing.

  Sports analytics, consumer behavior, bioinformatics, public policy, healthcare, quality control, ... using tools like regression, k-means clustering, Monte Carlo simulation, stochastic calculus.

  In finance, the Black-Scholes equations.
- *Cryptography*: Encryption algorithms like RSA, post-quantum cryptography, random number generators like the Mersenne Twister, and group-theoretic tools like elliptic curves.
- *Information theory*: thermodynamics, entropy and order from disorder as in the hard sphere model, data compression including the JPEG algorithm, neural information processing and artificial neural networks, machine learning based on information gain.
- *Abstract algebra*: Analyzing symmetries of molecules in chemistry, algebraic signal processing, quantum mechanics, automata theory.
- *Math bio*: ODE models in systems biology, machine learning in biochemistry e.g. AlphaFold, Brownian motion and molecular dynamics simulation, Markov state models.
- *Traffic flow*: The LWR equations and Aw-Rascle-Zhang model, shallow water equations, network analysis.
- *Atmospheric science*: Prediction of sea surface temperature using Koopman mode decomposition, prediction of El Nino using evolutionary algorithms, building ODE climate models.
- *Population theory*: Population growth models, birth-death processes, Markov chain models, ODE models like SIR and Lotka-Volterra.
- *Numerical analysis*: Scaling up kernel methods to big data using randomized numerical linear algebra.
- *Modeling:* Diffusion models used for AI image generation, chemical reaction networks like the Schlögl model, Markov models in biochemistry, epidemiology, finance, climate, text generation, etc., and the many examples listed above.

3 Brainstorming groups: Feb 23 and Feb 25

Using your course surveys, I put together a couple sets of groups based on common interests. I picked a third set of groups from a random permutation. I'm looking forward to upcoming discussions!

Round 1:

membership	possible common interests
Olivia, Casey, & Irvilinda	statistics, entropy, information theory
Alyssa & Kyle	math bio, chemistry, medicine
Alex & Amanda	computational chem, math physics
Camden & Jack	FPGAs, GIS, CAD, programming

Round 2:

membership	possible common interests
Irvilinda & Amanda	group theory applications to chemistry
Casey & Alex	information theory in thermodynamics
Camden & Kyle	optimization, machine learning
Jack, Alyssa, & Olivia	calc-based bio and chem modeling

Round 3:

membership	possible common interests
Camden, Olivia, & Alex	??
Casey, Irvilinda, & Alyssa	??
Kyle, Amanda, & Jack	??

Round 4 and beyond:

Your choice! You can work alone if you prefer, or in a group. Due to the small class size, a group of 2 would be great, but if 3 of you mesh really well, I think that is OK too. Your groups can also change between projects. We'll spend this Tuesday and Thursday (Jan 23 and 25) brainstorming topics, and you can hopefully decide on who you'd like to work with by next week.

4 Presentation guidelines

After projects 1 and 2 you will give a short presentation to the class. Here are some guidelines:

- *Length:* Your presentation should at most 10 minutes long, plus extra time for questions.
- *Content:* Since you only have 10 minutes, focus on the main ideas and most interesting aspects of your project. You won't have time to explain every subtlety of your project, but you do have time to mention a few interesting or unexpected details.
- *Visuals:* Visuals are not strictly required but are usually helpful. Feel free to use the whiteboard or project onto a screen using your laptop.
- *Tone:* Maintain a positive tone while discussing both the pros and cons of your project. I recommend not saying something along the lines of "I'm sure you've all heard of XXX before..." since it's likely someone in the audience hasn't!
- *Participation:* As an audience member, ask questions during your classmates's presentations!

5 Schedule

Below is a tentative schedule. A lot of class time will be spent working on projects. During this time I will wander around discussing things, e.g., debugging code, sorting out mathematical arguments, plans about what to try next, and thinking about interpreting results of the things you've already tried. This discussion aspect is the most important part of the course. During some class periods, you will present the results of your projects. Your presentations should be at most 10 minutes long, not including extra time for questions. I will also give a few presentations as a guide.

The topics listed below are subject to change based on your interests and project topics as the semester progresses.

date	topic	topics
Jan 16	course overview & questionnaire	selected examples
Jan 18	rotate in groups to brainstorm	
Jan 23	brainstorm; work on project 1	
Jan 25	work on project 1	
Jan 30	introduction to Overleaf	
Feb 1	using Latex and Overleaf	
Feb 6	work on project 1	
Feb 8	work on project 1	
Feb 13	work on project 1	
Feb 15	topics in machine learning	PCA, SVMs
Feb 20	topics in machine learning	neural networks
Feb 22	presentations	project 1 due
Feb 27	presentations; work on project 2	
Feb 29	presentations; work on project 2	
March 5	presentations; work on project 2	
March 7	presentations; work on project 2	
March 9-18	spring break	
March 19	work on project 2	
March 21	work on project 2	
March 26	work on project 2	
March 28	applications in science	Markov models
April 2	applications in science	diffusion, MD
April 4	presentations; work on final project	
April 9	presentations; work on final project	
April 11	presentations; work on final project	
April 16	presentations; work on final project	
April 18	work on final project	
April 23	group discussions	
April 25	group discussions	
April 30	out for lunch!	
May 2	poster session	

6 Questionnaire

1. Where are you from?

2. What's your major/concentration/minor? If you're choosing between majors, which ones are you considering?

3. Why are you taking this class? ("It's required for my degree" is an acceptable answer, but feel free to say more.)

4. What is your planned next step? That is, what are the next classes you plan to take, or job you plan to start / apply for, or grad program you plan to start / apply for, or place you want to live, etc?

5. Regarding Math 435, what are your feelings about group projects? Love them? Hate them? Put me in a random group to start? Let me choose my group? Prefer to work alone?

6. What are some of your academic interests or past classes you've liked?

7. Regarding Math 435, what are potential project areas of interest for you? Don't confine yourself. These do not need to be well-formed ideas yet, or even mathematically related ideas yet!

8. Any potential interest in projects related to machine learning?

9. Any potential interest in projects related to biology or chemistry?