Overview

Welcome to **SML 310 – Research Projects in Data Science**! The course introduces core techniques in data science, in lectures and in mini-projects. The lectures and mini-projects will support students' work no their course project and future research work involving data. Students will select a dataset of interest to them and produce an analysis and a project report. Students will use domain knowledge and technical expertise to produce their analyses.

Website & Forum

Website: `http://guerzhoy.princeton.edu/310f19/`

Forum: `https://piazza.com/princeton/sml310/`

All course handouts will be posted on the course website. *Students are responsible for reading all announcements on the course forum on Piazza.*

References

There is no required textbook. Lecture slides will be posted on the course website. The following are recommended as references.

- Andrew Gelman and Jennifer Hill. **Data Analysis using Regression and Multilevel/Hierarchical Models**. Cambridge University Press, 2006. Available to PU students from `https://ebookcentral.proquest.com/lib/princeton/detail.action?docID=288457`
- Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. **Deep learning**. MIT press, 2016
- Cosma Rohilla Shalizi. **Advanced Data Analysis from an Elementary Point of View**. Cambridge University Press (forthcoming). Preprint available at `https://www.stat.cmu.edu/~cshalizi/ADAfaEPoV/`
- Fred Ramsey and Daniel Schafer. **The Statistical Sleuth: A Course in Methods of Data Analysis, 3rd ed**, Brooks/Cole 2013
- Kieran Healy. **Data Visualization: A Practical Introduction**. Princeton University Press, 2018
- Eugene Charniak. **Introduction to Deep Learning**. The MIT Press, 2018
- **PyTorch tutorials**. Available at `https://pytorch.org/tutorials/`

Instructors

Instructor	**Email**	**Office**	**Office Hours**
Michael Guerzhoy	`guerzhoy@princeton.edu`	CSML 202	TBA
Tony Ye	`zjye@princeton.edu`	TBA	TBA

Contact

Lectures take place on Mondays and Wednesdays 3:00pm-4:20pm in CSML 103. Students should attend one of the two **precepts** (Thursdays 3:00pm-4:20pm in CSML 103 or Wednesdays 7:30pm-8:50pm in CSML 103).

Grading

The grading scheme and the tentative schedule for the course is as follows.

	Worth	Due	Notes
Initial Project Proposal	2%	Sept. 30	
Mini-Project 1	7%	Oct. 9	Generative models
Mini-Project 2	6%	Oct. 21	Python warm-up; data preprocessing
Revised Project Proposal	10%	Nov. 12	
Mini-Project 3	7%	Nov. 25	NLP; the problem-solving process in data science
Mini-Project 4	7%	Dec. 9	Image data and PyTorch; interpretable models
Project prop. presentation	10%	Mid-Nov	Present prior work, current progress, and plans
Course Project	40%	Dean's date	
Participation	10%		
Python homework	1%		

Late work

Each student starts the semester with 7 "grace days." You can use grace days to submit late work with no penalty. For example, a student who submits their work 25 hours after the deadline will use up two grace days. You cannot use more than 3 grace days at a time.

Course outline

A tentative course outline is below. Since SML310 is a small seminar course, the initial plan may be adjusted during the semester.

Week 1	Review of probability. Maximum Likelihood and Bayesian inference.
Week 2	P-values and CIs review. Linear and logistic regression review. Hierarchical models.
Week 3	Hierarchical models continued. Causal inference.
Week 4	Generative models for classification. Naive Bayes.
Week 5	Neural networks I. Generalization and overfitting.
Week 6	Neural networks II: Intro to PyTorch and learning with gradient descent.
Week 7	Unsupervised learning. Time-series data and Recurrent Neural Networks.
Week 8	Convolutional neural networks.
Week 9	Student presentations.
Week 10	Fairness in machine learning.
Week 11	Topics/projects
Week 12	Topics/projects

Python will be introduced in lecture and precept in the first month. The initial plan is below.

Week 1	Variables, conditionals, functions, lists, loops. Using a Python IDE.
Week 2	Lists and loops
Week 3	Nested for-loops, files, dictionaries

Course project

Each student in the course will be working on a large-scale data science project. Students are encouraged to pick a dataset of interest to them. Students should aim to achieve one or several of the following.

- Obtain interesting insights about the dataset
- Apply interesting data analysis techniques to the dataset
- Demonstrate a new data analysis technique and apply it to the dataset
- Collect a novel dataset

Course projects will contain

- A description of the dataset being analyzed, and a summary of the relevant domain knowledge. For example, if working with audio data, a summary of how sound is generated, perceived, and processed would be appropriate. If working with hospital data involving length-of-stay times, it would be appropriate to summarize the relevant literature on hospital lengths of stay.
- A summary of previous work that is related to the data analysis project.
- An exploratory analysis of the dataset being analyzed.
- A description of the method being applied, and a summary of related work.
- A description of the outcomes of the analysis, or an analysis of of a predictive system that was built.

Students will submit an initial project proposal by Sept. 30 and a revised project proposal, which should contain a full description of the project and initial eploratory data analysis results, by Nov. 12.

Each student will make a 15-20-minute project proposal presentation to the class about their project in mid-to-late November.

Participation and attendance

Students are expected to actively participate in the seminar. In particular, students should not miss more than 15% of lectures and precepts (students who feel comfortable with Python may skip precepts in September with no penalty). Students should actively work during precepts and the active learning segments of lecture.

Students will post at least four pieces of project feedback for their peers in Piazza. Details on this will be announced.

Students are encouraged to ask questions and to participate in discussion in lecture. While attendance in lecture will be considered when assigning "participation" grades, asking questions and participating in lecture discussion will not be.

Piazza participation will count for 2% of the course grade. Work in precepts and lecture attendance will count for an additional 8%.

Students who are missing class for a valid reason, such as illness or athletics participation, can email the instructor. Students who miss classes for a valid reason will not be penalized.

Collaboration policy

At heart, the collaboration policy in SML 310 is simple: you are encouraged to discuss SML 310 work with other people, but the work you submit must be substantially your own, and you should acknowledge other people's ideas if they are important enough to your work.

Mini-projects. You may discuss general approaches and ideas with anyone, unless the mini-project handout specifies otherwise.

You may use other people's code, with acknowledgement of the source, as long as the code is not specific to the task assigned in the mini-project. For example, it is fine to google for code that converts text to lowercase; however, you cannot simply copy-and-paste code that constitutes a solution to an entire problem in the mini-project. You must never look at or use code that was written by students in SML 310.

You may consult any textbook or website for general information. However, you must not seek out or read solutions to entire problems in the mini-projects.

Course projects. You may discuss general approaches and ideas with anyone. The sources of substantial ideas that are specific to your project and that are not your own should be acknowledged as such. For example, if a friend suggested a particular non-obvious instrumental variable that can be used in your causal analysis, the friend should be acknowledged. On the other hand, if your professor suggested that you try controlling for the variables that you already had in your dataset, there is no need to akcnowledge that in your write-up since the idea is obvious.

Code that is not your own can be used in the course projects. The source of the code should always be acknowledged.