Inverse Problems
Course AM216, Spring 2019

Instructors: Michael P. Brenner
Xiaojue Zhu
Teaching Fellow: Xi Chen
Class Schedule: F 12-3
Office hours: F 11-12 , or by appointment (Pierce 313)
Skype: A better way to talk to me is via skype – I'm mpbrenner.
E-mail: brenner@seas.harvard.edu

Class overview: The scientific endeavor is about solving inverse problems. Given data, our task is to invent a model that describes the data. Until recently, the predominant paradigm for solving this problem in the physical sciences was find a physical or mechanistic model of the phenomenon in question, and find the variant of the model that explained the data. This would be done by understanding in detail the features in the data that a class of models would predict, and then fitting these features to the data. The fits would allow narrowing down the classes of the model. A second paradigm dominated those cases where causal mechanisms were unknown – here the goal is to simply try to find a function that captures the correlation structure of the data.

Recent advances in computation and data analysis have changed things significantly. Given sufficient data it is now possible for computers to learn sophisticated features *themselves* using neural networks and large scale computation. In a range of practical examples, this has proven to surpass human ability to engineer features. This advance has had tremendous practical applications (think Amazon/Apple/Google/etc.) but it promises to make a significant impact on science and engineering practice. There seems to be enormous opportunity to change the way that science is done – but exactly how to go about using these methods to make science progress is not clear.

The goal of this class is think about these broad questions in a pedagogical fashion.This is on one hand going to be a research class, in that we are primarily motivated by understanding how current advances can improve scientific practice. On the other hand, we will do this in a pedagogical fashion: We will review classical methods for solving inverse problems – ranging from Bayesian parametric methods, to those based on the laws of physics and in this context, and will also introduce and study neural network based approaches. Theory will be mixed with computation and practical examples.

Class: Classes will meet on Friday from 12-245. Each class will be divided into roughly three parts: (i) Lecture; (ii) Computational instruction and lab; (iii) Student presentations. The laboratory section will be carried in python with Jupyter or Colaboratory notebooks. As the class progresses, the balance of these components will shift from instructor towards student presentations.

Requirements/Grading: Each student will be required to do the following:

1. Scribe report. (15 %) Everyone will be assigned a class for typing careful notes. Depending on the class size, a group of students might be assigned to each class. The scribe report should summarize the main ideas and calculations from the lecture, but *also* summarize the main ideas and results of the computer lab. We are trying to use this to get a holistic and analytical summary of what happens in this class.

2. Class presentations (15 %). During the first half of the semester, groups of students will sign up to present key papers that have aimed to apply machine learning to scientific problems. Depending on the class size, these presentations will be for about 15 minutes, and will summarize the papers for the class and lead a discussion on whether they succeeded. The papers you can present from will be chosen each week to coincide with the topic we are covering.

3. Homework. (30 %) For the first $\sim 2/3$ of the class, there will be regular homework, that will mainly involve computational implementations of the ideas of the class. Coding will be done in Python, where there are easy to use high level libraries for implementing the ideas of the class (for example, Keras for Neural networks.) The main problem in every problem set will ask you to solve an inverse problem – we will give you data and your task will be to figure out how we generated it.

4. Final projects: (40 %, including class presentations). A key part of the class is for each student to carry out a final project, applying the ideas we are discussing to some problem of particular interest. To prepare for a final project, students will make presentations on papers working up to the project throughout the class. Final projects can be carried out in groups of up to 3 people. We will start to choose them and work on them after about the first month of the class.

Feb 1	Introduction. What are inverse problems? Contrast two "orthogonal" approaches for solving inverse problems: Model/Mechanism based vs. Data driven approaches. Intro to Regression (when you know the right basis), PCA and SVD.
Feb 8	Deep dive into Linear problems (when you don't know the basis): Tikhonov Regularization; Bias variance tradeoff. SVM's and the Kernel trick.
Feb 15	Feedforward neural networks, and the use of (transfer learned) embeddings for classification and regression.
Feb 22	"Physical" inverse problems: Inferring equations of motion from data. Adjoint based optimization and automatic differentiation
Feb 29	Probabilistic methods: Bayes, maximum likelihood, etc. Inverse covariance matrix estimation.
March 8	Probabilistic methods, ctd: mixture models; the EM algorithm; monte carlo, etc. tensorflow probability
March 15	Sequence models: RNN's/LSTM's, etc.
March 22	Spring Break
March 29	Attribution methods: (How) can we deduce mechanism from a data driven model for a science problem? This is partly about the interpretability of neural networks, partly about formulation.
April 5	Control theory and Reinforcement Learning
April 12	TBD
April 19	TBD
April 26	TBD
May 3	Reading period – Class meets if needed.