STAT 339: PROBABILISTIC MODELING AND MACHINE LEARNING (SPRING 2017)

Instructor. Colin Reimer Dawson (*he/him/his*)
Office. King 204
Email. cdawson@oberlin.edu
Website. http://colindawson.net/stat339/
Office Hours. MTW 10:00-11:30am, F 4:30-5:30pm, or by appointment. *If your schedule prevents you from attending any of the posted hours, please let me know as soon as possible.*
Location. King 239

Course Description. This course is an introduction to the field of machine learning from the particular perspective of probabilistic reasoning. We will discuss fundamental principles of machine learning and probabilistic reasoning, as well as specific models and algorithms used to do classification, prediction, clustering, hidden variable modeling, and sequence learning. Rather than cover as many different models and algorithms as possible, we will focus on general principles of model-building and probabilistic reasoning as illustrated through a relatively small set of common models and algorithms. The overarching goal of the course is for your to strengthen your ability to reason probabilistically, and to equip you to learn about and/or develop new models and algorithms to solve problems. Applications will be selected from a variety of domains.

Who This Course is For. The course is designed to appeal to students interested in modern data science who have some prior experience with mathematical and/or statistical thinking. This likely includes (among others!) math majors with statistics interest, computer science majors with data science / artificial intelligence interest, natural / social science majors (e.g., psychology, neuroscience, economics, biology, physics) with a methodological/statistical bent, and even humanities majors who have some math background and who are interested in quantitative methods (for, e.g., text analysis). Note that this course is quite different in its approach to machine learning from CSCI 374, and it certainly makes sense to take both.

Date: Last Revised January 27, 2017.

Prerequisites. The main prerequisites for the class are familiarity with derivatives and integrals at the level of Calculus II (MATH 134), basic principles of data, statistical inference, and linear regression modeling at the level of an introductory statistics course (STAT 113, 114, 215, 237 or equivalent), and familiarity with basic programming concepts, such as variables, functions, and arguments (if you took a stat course where you used R, that should be enough).

In terms of additional mathematics, we will rely heavily on probability, but I am not expecting you to have taken a course in it; we will go through the main ideas that we will need. I have set as an additional prerequisite any MATH or STAT course numbered at the 200 level or above (MATH 220, MATH 231, MATH 232, STAT 213, STAT 215, or STAT 237). We will rely on a few concepts from each of these courses, and although we will introduce them as needed, being familiar with at least some of them coming in is helpful. Note that either of STAT 215 or 237 satisfy both the "intro stat" and "intermediate stat/math" requirements.

Materials

Main Textbook. The main textbook is

- *A First Course in Machine Learning*, by Simon Rogers and Mark Girolami.

You do need the 2nd edition, as it includes several new chapters not in the first edition. We will (attempt to?) cover the large majority of the material in the book.

Supplementary Texts. Supplementary readings will come from the following sources, all of which are freely available in pdf online from the authors' websites:

- *Bayesian Reasoning and Machine Learning*, by David Barber (`http://web4.cs.ucl.ac.uk/staff/D.Barber/textbook/171216.pdf`)
- *Introduction to Statistical Learning*, by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani (`http://www-bcf.usc.edu/~gareth/ISL/ISLR%20Sixth%20Printing.pdf`)
- Chapter 1 of *Machine Learning: A Probabilistic Perspective* by Kevin Murphy (`https://www.cs.ubc.ca/~murphyk/MLbook/pml-intro-22may12.pdf`)

Software. You may use any programming language(s) you like to write code; scientific computing languages with easy access to plotting capabilities, such as MATLAB/Octave, Python, and R are strongly recommended. Example code will be

provided that runs in MATLAB and its open-source counterpart Octave. If you have access to a MATLAB license, great! If not, you probably want to download Octave, which is free. See the "Resources" section of the course website for download links and other resources on various languages.

The first (optional) homework assignment will be a chance for those without (much) programming experience to learn some basics about functions, arguments, variables, and arrays and a refresher on those ideas for others who want it.

Miscellany

Laptops. I strongly discourage the use of electronic devices in class, unless there is an in-class activity where a laptop is needed. For note-taking, take hand-written notes if you are physically able to do so: there is evidence that writing by hand improves your cognitive processing of the material.

Email Etiquette. Email is the best way to reach me outside of a face-to-face meeting. You are welcome to address me by my first name, which is generally what I will use when signing emails. I try to respond to most email within 24 hours. If I have a lot, and it is not pressing, as well on my research day (Thursdays) and the weekend, it might be longer. **If you need to ask me about something due the following morning, don't wait until the night before**, as I have limited work time in the evenings due to family obligations, and what time I do have is often needed to prepare for class for the following day.

Accommodations. If you require accommodations of any kind in order for you to do your best work in this class, please let me know as early as possible, and consult as well with the Office of Disability Services (ODS). By college policy, **all requests for accommodation require documentation from ODS.**

Honor Code. The Oberlin College Honor Code formalizes the idea that all work that you submit is your own and that you have given credit to the ideas and work of others when you incorporate them. You will be asked to write and sign the honor pledge on each graded assignment that you hand in. The honor pledge reads: "I have adhered to the Honor Code in this assignment." What it means to adhere to the honor code depends on context. I describe what it means to follow the honor for each assignment type below.

More information about the honor code can be found on the web at the Dean of Students site:
`http://new.oberlin.edu/office/dean-of-students/honor/students.dot`

Graded Work

The final grade will be based 40% on homework (approximately 6 assignments), 30% on a final project, 20% on a takehome midterm, and 10% on class participation, as assessed through frequent short in-class self-reflections.

Homework. There will be an optional primer assignment to get up to speed with coding (specifically scientific computing) basics, and approximately six more substantial assignments due roughly every two weeks. These will consist of a mix of written (conceptual) questions, mathematical calculations and derivations, and computer implementation. For more complex methods, code scaffolding, or sometimes fully functional code, will be provided.

Honor Code: You may freely discuss homework assignments with each other (and of course with me); however **each individual must turn in their own original code and writeups.**

Participation. Most days I will set aside 2 minutes at the end of class for you to write down a short reflection comment or question about the class. I will read these between classes to get a sense of how the course is going and what topics need more review time. These are graded for completion only.

Honor Code: You may not submit a reflection on anyone else's behalf.

Midterm Exam. There will be one take-home midterm exam covering the foundational material of the first six weeks, to be done the week before spring break (I will try to choose days during that week that work best for your midterm schedules in other classes). The questions will mainly be conceptual, but may also include simple derivations and on-paper calculations. The midterm will not involve coding.

Honor Code: The takehome exam must be done individually, but you may consult any assigned readings or notes.

Final Project and Presentations. For the final project, you will work in a small group (2-3 people), defining your own classification/clustering/prediction question(s), and implementing and evaluating at least three different machine learning models/algorithms on your problem. At least one of the three should be a standard "off the shelf" probabilistic method, and at least one of which should be either an original model or one of the "advanced" probabilistic models discussed in Part IV of the course. The third method may be an additional model from one of these two categories or may be a non-probabilistic method. The scheduled final exam period is Friday, May 12th from 9-11 A.M. and will be used for presentations of the projects. Final writeups are due at the presentation.

Honor Code: Free collaboration with your team members is of course required. Describing the work that other people did before you is an important part of intellectual inquiry, and you must give credit and cite sources for for any data, code, or ideas that did not originate within your team. This includes paraphrases as well as direct quotations. It is fine (and indeed you are encouraged) to call other people's open-source code in your own code. All members of the group must make approximately equal overall contributions to the project, though it is possible, for example, for one person to do more coding and another to do more writing, etc. All group members must participate in the presentation.

Approximate Topic Outline

In the reading list below, the following abbreviations are used:

- FCML: *A First Course in Machine Learning* (Rogers and Girolami)
- BRML: *Bayesian Reasoning and Machine Learning*
- ISL: `Introduction to Statistical Learning` (James, Witten, Hastie, Tibshirani)
- MLPP: *Machine Learning: A Probabilistic Perspective* (Murphy)

Classes	Topic	Readings
1	What is Machine Learning?	MLPP 1.1-1.3, BRML 13.1
	Types of Learning	
	[PART I: Basic Machine Learning ($\sim$ 2 weeks)]	
2	Nearest Neighbor Classification	BRML 14.1-14.2
	Generalization Error	
3-5	Linear and Logistic Regression	FCML Ch. 1.1-1.4, ISL Ch. 3
6-7	Utility, Loss, and Validation	FCML 1.5, BRML 13.2-13.3, ISL Ch. 5
	[PART II: Probabilistic Modeling ($\sim$ 4 weeks)]	
8-9	Probability Fundamentals	FCML 2.1-2.7, BRML 1.1, 8.1-8.4
10-11	Likelihood-Based Inference	FCML 2.8-2.11, BRML 1.2-1.3, 8.5-8.8
12-13	Bayesian Inference	FCML Ch. 3.1-3.7, BRML 9.1-9.2
14-15	Naive Bayes Classification	FCML 5.1-5.2, BRML 10.1-10.3
16-17	Bayesian Regression & Model Selection	FCML Ch. 3.8, BRML 12.1-12.4
18-20	Approximate Inference	FCML Ch.4, 9.1-9.4, BRML 27.1-27.4
21	Catch-up day	
	SPRING BREAK	
	[PART III: Unsupervised Learning and Latent Variable Models ($\sim$ 3 weeks)]	
22-24	Mixture Models and the EM algorithm	FCML Ch. 6, BRML 20.1-20.3
25-26	Belief Networks	FCML 3.6, BRML 3.1-3.3
27-29	Hidden Markov Models	BRML 23.1-23.3, 23.5
	[PART IV: Nonparametric Models ($\sim$ 2 weeks)]	
30-32	Gaussian Processes	FCML Ch. 8, BRML 19.1-19.2, 19.5
33-34	Infinite Mixture Models	FCML Ch. 10
35	Infinite Hidden Markov Model	notes
	PART V: Non-Probabilistic Methods ($\sim$ 1-2 weeks)	
36-37	Support Vector Machines	FCML Ch. 5.3, 5.5, ISL Ch. 9
38-39	Neural Networks	TBD