Course Syllabus: COMPSCI 688
Probabilistic Graphical Models – Spring 2016

Instructor: Prof. Brendan O'Connor (http://brenocon.com)

TA: Tao Sun

Web: http://people.cs.umass.edu/~brenocon/pgm2016/

Course Description: Probabilistic graphical models are an intuitive visual language for describing the structure of joint probability distributions using graphs. They enable the compact representation and manipulation of exponentially large probability distributions, which allows them to efficiently manage the uncertainty and partial observability that commonly occur in real-world problems. As a result, graphical models have become invaluable tools in a wide range of areas from computer vision and sensor networks to natural language processing and computational biology.

The aim of this course is to develop the knowledge and skills necessary to effectively design, implement and apply these models to solve real problems. The course will cover (a) Bayesian and Markov (MRF) networks; (b) exact and approximate inference methods; (c) estimation of both the parameters and structure of graphical models. Students entering the class should have good programming skills and knowledge of algorithms. Undergraduate-level knowledge of probability and statistics is recommended.

This course emphasizes the versatile compositionality of PGMs' core building blocks—intuitive representations and algorithms which can be combined to derive a huge variety of models and inference/learning procedures. Probabilistic graphical models unify a very broad range of statistical and machine learning methods, and allow the invention of new ones too. These concepts are essential for advanced work in machine learning, artificial intelligence, and statistical modeling.

Textbook: Kevin Murphy's *Machine Learning: a Probabilistic Perspective* is the course textbook. We will provide additional readings throughout the course.

Course Website: The course website, materials, and discussions are hosted on Piazza, and assignments will be submitted on UMass' Moodle website. Links are available from the public webpage at http://people.cs.umass.edu/~brenocon/pgm2016/

Announcements: Official announcements for the course will go out through Piazza. We will use your *official UMass email*. If you want to redirect this email elsewhere, login to the UMail Post Office (https://postoffice.oit.umass.edu/umpo/exec/login) and setup mail forwarding to an alternate address, or change your Piazza settings.

Grading Plan: The coursework will consist of homework assignments, quizzes, in-class discussions, and a final exam. The assignments will include some written questions, but will mostly consist of mathematical derivations, programming and experimentation. Assignment solutions will be written-up as short reports. The grading scheme is shown at the right.

Homework Assignments	50%
Midterm Exam	15%
Final Exam	15%
Quizzes	15%
Course Participation	5%

Course Policies:

- **Homework Submission:** Homework assignments will generally consist of a written component consisting of derivations, short answer questions and analyses as well as a programming component. The written component must be handed in at the start of class on the date it is due. Code for the programming component must be uploaded to Moodle before class on the date it is due. Code must be submitted as a single **zip** file (not rar, gz, tgz, bz2, etc.). In the event that you cannot attend class on the date an assignment is due, you may submit a PDF version of the written portion of your assignment on Moodle, before class.
- **Late Homework:** To allow some flexibility to complete assignments while working on research or other coursework, you have a total of five free late days. You will be charged one late day for handing in an assignment within 24 hours after it is due, two late days for handing in an assignment within 48 hours after it is due, etc. Your assignment is considered late if either the written or code portions are submitted late. The late homework clock stops when both the written and code portions are submitted. The late written portion of your assignment must be submitted to Moodle. It is your responsibility to scan any written work. The code portion must be uploaded to Moodle. After you have used up your late days, late homework will not count for credit except in special circumstances. Any unused late days will be credited as bonus points on your final grade at a conversion rate of 0.5% per late day.
- **Homework Collaboration and Academic Honesty:** You are encouraged to discuss assignments and course material with other students in person or on the course forums. However, you may not share solutions and you may not look at at others' solutions. You must produce all solutions yourself: you must write your own code, run your own experiments, and write your own write-up or derivations as appropriate for the problem.
- You are required to list the names of anyone you discuss problems with on the first page of your solutions. Copying any solution materials from external sources (books, web pages, etc.) or other students is considered cheating. Sharing your code or solutions with other students is also considered cheating. Any detected cheating will result in a grade of 0 on the assignment for all students involved, and potentially a grade of F in the course.
- **External Resources:** For programing assignments, you are allowed and encouraged to use libraries and existing code at *one level of abstraction lower than the material we are covering in this course*. For example, if you are experimenting with inference in Bayesian networks, you can use an existing code library implementing data structures for graphs, but not an existing library implementing Bayesian networks. If you are experimenting with learning multivariate Gaussian models, you can use an existing code library implementing matrix operations, but not a library implementing Gaussian models. If you are in doubt about whether using a particular code library would be permitted for an assignment, ask. You must provide citations for all code and libraries that your programming assignment solutions build on. In formulating solutions to written homework problems, you may use background material other than the textbook and the lecture notes, but you must provide citations.
- **Re-grading Policy:** Errors in grading of assignments and exams can occur despite the best efforts of the course staff. If you believe you've found a grading error, complete the online re-grade request form. Write "Regrade" on your original solutions and hand them back in to the instructor. Re-grade requests must be submitted no later than one week after the assignment is returned.
- **Course Participation:** Students are expected to attend each class and participate in discussions, which will count toward your classroom participation grade.

Approximate Schedule: (Subject to change)

Lecture	**Topics**
Lecture 1	Introduction: Course logistics. Probability theory.
Lecture 2	Bayesian Network Representation
Lecture 3	Learning in Bayesian Networks
Lecture 4	Markov Random Fields
Lecture 5	MRF Inference
Lecture 6	Message Passing I
Lecture 7	Message Passing II
Lecture 8	Learning in Markov Networks
Lecture 9	Numerical Optimization
Lecture 10	MRFs and BNs
Lecture 11	Monte Carlo Methods
Lecture 12	Markov Chain Monte Carlo I
Lecture 13	Markov Chain Monte Carlo II
Lecture 14	Markov Chain Monte Carlo III
Lecture 15	Midterm Review
	(Spring Break)
Lecture 16	Expectation-Maximization
Lecture 17	Variational Inference
Lecture 18	Variational Learning
Lecture 19	Inference Case Study
Lecture 20	Optimization for Inference
Lecture 21	Structured Perceptron and Support Vector Machines
Lecture 22	Neural Networks I
Lecture 23	Neural Networks II
Lecture 24	Bayesian Modeling
Lecture 25	Bayesian Modeling
Lecture 26	Advanced Topics
Lecture 27	Final Review