CISC 5800 Machine Learning
Department of Computer and Information Science
Dr. Daniel D. Leeds, Spring 2018

Class times: Wednesday, 5:30 – 7:45pm, LL 307
Instructor: Prof. Daniel D. Leeds
Office: For office hours: LL 610H, Non-office hours: JMH 332 (Rose Hill)
E-mail: dleeds@fordham.edu
Office hours: Most usually Wednesday 4-5pm and by appointment

Course website: http://storm.cis.fordham.edu/leeds/cisc5800/

Texts: No text is required, but it will be useful to have one for reference.
"Pattern Recognition and Machine Learning", C. Bishop, 2007.
Bishop is a classic text in the field. I recommend Bishop for most students.

"Machine Learning: A Probabilistic Perspective", K.P. Murphy, 2013.
Murphy is a newer text. It provides more in-depth analyses we will not cover in class, but may be of interest to mathematically advanced students.

Our course site also links to course notes from Andrew Ng's Machine Learning course notes at Stanford University, which are particularly helpful.

Course description: Machine learning discovers and responds to patterns in the rich data sets of the world. It provides powerful tools for diverse fields, from software development to product marketing to scientific research. This course introduces a collection of prominent machine learning models. We will study both theory and implementation. Topics will include: Bayesian statistics, learning theory, support vector machines, dimensionality reduction, and graphical models.

Objectives: To understand the mathematical principles and algorithmic mechanics behind popular methods in machine learning. A student who successfully completes this course will be able to:

- Implement "basic" learning algorithms in the Matlab or Python programming environment
- Apply diverse learning algorithms to complex data
- Understand the advantages and limitations of diverse learning approaches

Pre-requisites: It is essential each student has a base level of computer science and math background to be able to succeed in this class.

- Completion or equivalent experience of at least one semester of a computer programming course, such as Java, C++, or Python, and be comfortable with the syntax for arrays/lists, loops, defining new functions, and evaluating multi-part logical conditions.
- Background in probability — a familiarity with conditional probabilities and independence, which we will quickly review at the start of the course.
- Comfort with algebra; prior background in calculus will be beneficial, but is not required as we will learn needed calculus topics as we go

Software: We will complete our programming assignments in Matlab or Python. There are multiple ways to use Matlab or Python. I highly recommend you follow option 1. I will provide support for programming difficulties you face in Matlab. Due to my own limited time, I most likely will **not** be able to provide support for programming difficulties in Python.

- **Recommended:** Download your own student license from MathWorks for $50-$100.
- **Second-choice:** Run Matlab in the computer lab or (without graphics) by remote login to our department machines.
- **Third-choice:** If you have a strong computational background/past Python experience and want to use Python, you can use numpy, scipy and matplotlib packages to write acceptable code for this class in Python. (You also may used the pandas package; you **may not** use packages like tensorflow for this class.)

Attendance and class participation: As we do not directly-follow any one textbook and, instead, rely heavily on lectures and lecture slides, it is important to attend every class, and to arrive on time. One unexcused/unexplained absence is permitted for the semester. Attendance will be taken regularly. Please *actively* participate in class since this will make the course more interesting for everyone! Ask questions if you are unsure about something.

Laptops during lecture: I encourage students **NOT** to use laptops during lecture. If you feel you need it, limited acceptable use includes writing notes and reading lecture material. If you do use a laptop, please sit near the back of the class. Keep in mind anyone behind you can see what is on your screen at any time and you may be distracting your classmates.

Course assignments: There will be 4-5 homeworks and a final project assigned for the course. The homeworks usually will be announced at least a week before they are due, e.g., a homework announced on Wednesday may be due the following Wednesday. All assignments must be turned in on time to receive credit.

Academic honesty: All work submitted in this course must be your own. You are very welcome to discuss class topics with other students generally, but you may not provide assignment solutions to one another. Copying of any part of an assignments is never acceptable and will be considered a violation of Fordham's academic integrity policy. Violations of this policy will be handled in accordance with university policy which can include automatic failure of the assignment and/or failure of the course. See Fordham's Graduate Policy on Academic Integrity for more information.

Exams: There will be one mid-term exam in February and a final in May. The exact dates will be announced at least 3 weeks in advance of the exams.

Timing conflicts: If you have a significant issue and cannot complete an assignment on time, or cannot attend class on a certain day, whenever feasible let me know beforehand -- I tend to be reasonable in such cases. Examples of significant issues include personal illness (with doctor's note) or a religious holiday (give me at least a week's notice) on an announced exam day. In general, let me know of any significant issues that affect your performance early on.

Grading: The percentages given below are guidelines for both the student and instructor and may be changed as needed to reflect circumstances in the course. Any changes that occur during the semester will be minor.

Participation	5%	Mid-term	25%
Homeworks	25%	Final exam	25%
Final project	20%		

Tentative schedule

Schedule is subject to change as the class progresses. The last few days are scheduled as "catch-up" days as it is likely we will run behind at some point as the semester progresses. If we are **ahead** of schedule, I will add some additional topics.

January 17	Background: Probability, calculus, classifier basics, MATLAB basics	
January 24	Bayes classifier	HW 0 due
January 31	Bayes classifier / Logistic classifier	
February 7	Logistic classifier / Support vector machines	HW 1 due
February 14	Support vector machines	
February 21	Dimensionality reduction	HW 2 due
February 28	Neural networks	
March 7	Midterm	
March 14	Neural networks	
March 21	Bayesian Networks	HW 3 due
April 4	Hidden markov models	
April 11	Expectation maximization	HW 4 due
April 18-25	Catch-up days	Final project due
May 2	Final exam	