CS 6190: Probabilistic Modeling

Administrative Details and Syllabus
Spring 2018

Course Web Page: http://www.cs.utah.edu/~fletcher/cs6190/

Description. This course focuses on how to use probability theory to model and analyze data. Data in the real world almost always involves uncertainty. This uncertainty may come from noise in the measurements, missing information, or from the fact that we only have a randomly sampled subset from a larger population. Probabilistic models are an effective approach for understanding such data, by incorporating our assumptions and prior knowledge of the world. These ideas are important in many areas of computer science, including machine learning, data mining, natural language processing, computer vision, and image analysis.

Instructor. Tom Fletcher. Office: 4686 WEB. Email: fletcher@cs.utah.edu.

Class Meetings. Tuesdays and Thursdays, 3:40 – 5:00pm, WEB L103.

Getting Help. Take advantage of the instructor and TA office hours. We will work hard to be accessible to students. Please send me email if you need to meet outside of office hours. Don’t be shy if you don’t understand something: come to office hours, send email, or speak up in class!

Announcements. Important announcements, such as assignment corrections or deadline changes, will be sent to the class via the Canvas announcements for this class. Make sure that you have setup Canvas to send you announcements to an email address that you will check regularly, as they may be time-sensitive.

Textbook. There is no textbook for this class. See the course webpage for reading material, as well as optional, recommended textbooks to learn more.

Workload and Grading. There will be several (4-5) projects, involving programming and problem solving. They make up the entire grade for the course (and may be unequally weighted). Letter grades will be assigned as follows:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>93-100</td>
<td>87-89 B+</td>
</tr>
<tr>
<td>A-</td>
<td>90-92</td>
<td>83-86 B</td>
</tr>
<tr>
<td>B</td>
<td>87-79</td>
<td>77-79 C+</td>
</tr>
<tr>
<td>B-</td>
<td>80-82</td>
<td>73-76 C</td>
</tr>
<tr>
<td>C</td>
<td>77-69</td>
<td>67-69 D+</td>
</tr>
<tr>
<td>C-</td>
<td>70-72</td>
<td>63-66 D</td>
</tr>
<tr>
<td>D</td>
<td>67-69</td>
<td>60-62 D-</td>
</tr>
<tr>
<td>D-</td>
<td>63-66</td>
<td>0-59 E</td>
</tr>
</tbody>
</table>

All programming must be done in either R (http://www.r-project.org/) or Python (http://www.python.org). Submit all assignments electronically through the Canvas page for this course. All assignments must be formatted in R markdown (*.Rmd) or a Jupyter notebook (*.ipynb). Following the paradigm of literate programming, these documents will combine formatted text descriptions, math notation (using LaTeX), and working R code and output. Homeworks are due by midnight (11:59:59 PM) on the due date. Late assignments will not be accepted.

If you believe there is an error in grading (homeworks or quizzes), you may request a regrading within one week of receiving your grade. Requests must be made in writing, explaining clearly why you think your solution is correct.
Working Together. You are encouraged to form study groups and to discuss the concepts in a homework assignment with your fellow classmates. However, you must develop and write up your own solutions. Do not read another person’s answers or code, and do not show your answers or code to anyone else. Presenting someone else’s solutions as your own will be considered cheating.

Posting homework questions on the internet is not allowed and is considered cheating. This is cheating regardless of whether or not you use any of the posted answers or hints.

You may use other sources (Google, Wikipedia, papers, books, etc.) to find information to help answer problems. However, you must cite any source that you use. Failing to cite a source is considered cheating. Also, you may not use source code that you find online (unless it is explicitly permitted in the assignment).

If a student is caught cheating, they will receive a failing grade for the course. For a detailed description of the university policy on cheating, please see the University of Utah Student Code: http://www.regulations.utah.edu/academics/6-400.html.

Please also read the School of Computing Guidelines: https://www.cs.utah.edu/~germain/SoC_Guidelines_Spring_2017

Discussions in Canvas. Students are encouraged to use the Canvas discussion board to ask questions that may benefit the entire class. Questions may be answered by the professor, TAs, or other students. Do not post full or partial solutions to the homework assignments. It is appropriate to provide general advice on how to use R (for example, how to setup R Studio, how to use a particular R function, solving an error message, etc.) You may also discuss general mathematical concepts from class on the discussion board, again as long as you don’t provide answers to assignments. Also, feel free to share web links to sources of information that you have found useful.

Students with Disabilities. The University of Utah seeks to provide equal access to its programs, services, and activities for people with disabilities. If you need accommodations in this class, reasonable prior notice needs to be given to the Center for Disability Services, 162 Olpin Union Building, 581-5020 (V/TDD). CDS will work with you and the instructor to make arrangements for accommodations.

Syllabus. The following topics will be covered. See the course web page for a detailed schedule.

- Probability Crash Course
  - Probability distributions, conditional probability
  - Estimation, maximum likelihood
  - Linear regression
- Basics of Bayesian statistics
  - Priors and maximum a posteriori estimation
  - Exponential families and conjugate priors
  - Improper/invariant priors
  - Graphical models, Markov random fields
- Computational methods for Bayesian inference
  - The Expectation Maximization Algorithm
– Sampling methods (Markov Chain Monte Carlo)
– Variational methods

• Regression
  – Bayesian prediction
  – Model selection, sparsity, and model averaging
  – Nonparametric regression (Gaussian processes)

• Classification
  – Discriminative vs. generative learning
  – Naive Bayes
  – Logistic regression
  – Sparsity in classification
  – Ensemble learning (boosting, bagging, random forests)

• Re-sampling methods
  – Cross-validation
  – Bootstrapping
  – Permutation testing