Info 2950
Intro to Data Science

Instructor: Paul Ginsparg (242 Gates Hall)

Tue/Thu 10:10-11:25
Kennedy Hall 116 (Call Auditorium)

Only permitted to use middle section of auditorium

https://courses.cit.cornell.edu/info2950_2017sp/
Info 2950: Introduction to Data Science

Spring 2017
Tue/Thu 10:10-11:25 PM, Kennedy Hall 116 (Call Auditorium)
4 credits, S/U Optional

This course teaches basic mathematical methods for information science, with applications to data science. Topics include discrete probability, Bayesian methods, graph theory, power law distributions, Markov models, and hidden Markov models. Uses examples and applications from various areas of information science such as the structure of the web, genomics, social networks, natural language processing, and signal processing. Assignments require python programming.

Announcements:

Professor: Paul Ginsparg (242 Gates Hall, ginsparg @at@ cornell.edu)
Office hours: Wed 1-2 PM (or by appointment)
TAs: Hala B., Carter B., Marc G., Haoyan L., Dean O., Qinru S., Alex W., Kai Y. (office hours on Piazza pages)
Course website: http://courses.cit.cornell.edu/info2950_2017sp/ (this page)
Prerequisites: An introductory statistics course (from approved list) and an introductory programming class (e.g., CS 1110), or permission of instructor
Highly Recommended: Math 2310 or a similar linear algebra class (but no longer a prereq)
Textbooks: Rosen, Discrete Mathematics and Its Applications, 6th edition, is a basic ref for some of the earlier non-programming material

Syllabus:

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture</th>
<th>Reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1</td>
<td>Thu 1/26</td>
<td>1. Course Introduction</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Most recent class syllabus: Spring '16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Python notebook class lecture: sec1.ipynb (here)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Instructions for downloading ananconda python are here</td>
</tr>
<tr>
<td></td>
<td></td>
<td>There will be some python resources listed on the Piazza course site, the python.org site has a tutorial, and there are other online resources, including the book Think Python.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rosen ch 2.1-2.2 (online here)</td>
</tr>
</tbody>
</table>

Grading:
50% Problem Sets
25% Final exam
20% Midterm exam (in-class)
5% Subjective

Academic integrity policy:
You are expected to abide by the Cornell University Code of Academic Integrity. It is your responsibility to understand and follow these policies. In particular, the work you submit for course assignments must be your own. You may discuss homework assignments with other students at a high level, by for example discussing general methods or strategies to solve a problem, but you must cite the other student in your submission. Any work you submit must be your own understanding of the solution, the details of which you personally and individually worked out, and written in your own words.

Advising Notes:
This course is a required course for the Information Science (IS) major in the College of Arts and Sciences and the College of Agriculture and Life Sciences. It can also be used to satisfy the math/stats course requirement for the Information Science minor/concentration.

[Note for InfoSci majors: INFO2950 and CS2800 may not both be taken for credit towards the IS major.
If (and only if) for some reason you have already taken CS2110 and CS2800, that combination can be petitioned to be used in place of INFO2950.]
<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Lecture</th>
<th>Reading</th>
</tr>
</thead>
</table>
| 1     | Thu 1/28   | 1. Course introduction                          | Most recent class syllabus: Fall '14
Ipython notebook class demo: loc1.ipynb (intro)
Instructions for downloading anconda python are here
There will be some python resources listed on the Piazza course site, the python.org site has a tutorial, and there are other online resources, including the book Think Python.
Rosen ch 2.1-2.2 (online here) |
|       | Tue 2/2    | 2. Probability and counting                      | Notes (11) on set theory and loc2.ipynb
Notes (12-3) on probability
Rosen ch 6 'Counting' (ch 5 in earlier editions, link given in class) |
|       | Thu 2/4    | 3. Conditional Probability                      | Continue notes (12-3) on probability
(Rosen ch 7.1-7.2 (discrete probability, ch 6 in older editions, link available)) |
|       | Tue 2/9    | 4. Bayes' theorem and applications               | Finished these notes (12-3) for the birthday/jackpot problem, then started notes (14-5) on cond1prob and Bayes
(Rosen ch 7.2-7.3)
Re "birthday" problem, see loc4.ipynb, and as well nytimes popular version (Strogatz). |
|       | Thu 2/11   | 5. Probability conundrum + naive Bayes           | mentioned Wason selection task,
continued these notes (14-5), including Bayesian spam filters (Rosen 7.3).
Introduced first part of problem Set 2 (not due til a week after break) |
| 2     | Thu 2/16   | A Break                                           |                                                                                           |
|       | Thu 2/21   | 6. Expected value, Variance                     | A few more "naive Bayes" resources: Bayesian spam filtering, A plan for spam,
and nytimes popular Bayes (Strogatz).
mentioned Doomsday Argument
Then started these notes (Rosen 7.4) |
|       | Thu 2/23   | 7. Binomial and Normal distributions             | Finish these notes (Rosen 7.4),
and discuss loc7.ipynb, see also these simulations
(see riddler column for the probability problem mentioned at outset) |
|       | Thu 2/25   | 8. Regexp, more expectation/variance of Bernoulli processes | comments on regular expressions
more from loc7.ipynb, and started discussion of loc8.ipynb |
| 3     | Tue 3/1    | 9. Normal distributions and Election polls       | Clarification of Bernoulli Trial, introduced Galton machine (time-lapse video),
mentioned Bernoulli->normal derivation and What is Life?,
then deviations from normal: on-line dating and Poincare bread,
plus "sked on p=.05" and pre-cognition.
More on normal distribution from loc8.ipynb
(closed with continued discussion of addendum from that notebook.) |
|       | Thu 3/3    | 10. Data Wrangling, Visualizations, and PS3      | Most of the class was devoted to an interactive tutorial on the two parts of problem set 3 (since it was interactive, there are no notes other than about this question, so unfortunate if missed).
Also recalled UNIVAC data science foray during 1952 election. In relation to PS3 prob 5 mentioned Data janitoring (nytimes) and Data carpentry (Mimmo).
Also completed discussion of loc7.ipynb and recurred these simulations (linked from lecture).7 |
|       | Thu 3/8    | 11. Poisson distributions                       | notes on exponentials, see also popular logarithms (Strogatz nytimes, includes video link)
then some notes on Poisson distribution, and simulation of balls_in_bins.ipynb |
| 4     | Thu 3/10   | 12. Poisson, Multivariate Normal, and PS4        | Mentioned covariance matrix of multinormal distribution,
UTC (last leap second was 30 June 2015 23:59:60), json,
and more description of real data: armpit.ipynb (to provide framework for PS4 Q1) |
| 5     | Tue 3/15   | 13. Graph Theory                                 | start graph theory (Rosen ch 10.1-10.4, and these notes).
Bipartite and Hamiltonian circuits (Rosen ch 10.5, and notes2, see also Hamiltonian circuit and Seven Bridges of Königsberg),
then "Why your friends have more friends than you do" (Feld, 1991; slide, plus nytimes popular version (Strogatz)) |
discussed implications for vaccination (see also popular version)
Discussed properties of social graphs that distinguish them from random graphs, see Easley/Kleinberg Chpt 3 for "triadic closure" and "clustering coefficient" (section 3.1, p.48);
and discussed various counting problems on graphs involving (x choose m) and Poisson. |
| 6     | Tue 3/22   | 15. Graph Algorithms                             | distance matrix, adjacency matrix, spanning trees, notes3 (Rosen ch 10.7, 11.1), notes4 (Rosen 11.3-11.5) (DFS, BFS, topo sort) |
Description

Teaches basic mathematical methods for information science, with applications to data science. Topics include discrete probability, Bayesian methods, graph theory, power law distributions, Markov models, and hidden Markov models. Uses examples and applications from various areas of information science such as the structure of the web, genomics, social networks, natural language processing, and signal processing. Assignments require python programming.

Prerequisite: An introductory statistics course (from the approved list of accepted statistics courses found at [http://infosci.cornell.edu/academics/degrees/ba-college-arts-sciences/degree-requirements/core-requirements](http://infosci.cornell.edu/academics/degrees/ba-college-arts-sciences/degree-requirements/core-requirements)) and an introductory programming class (e.g., CS 1110), or permission of instructor.

Highly Recommended: Math 2310 or a similar linear algebra class.

Website

https://courses.cit.cornell.edu/info2950_2017sp/

Lecture Location

Kennedy Hall 116 (Call Auditorium, Way too big ...)

Midterm

TBD

Final Exam

TBD

Graded problem Set Pickup

Homework Handbook Room Gates 216 (roughly 12:30-4:30pm)

Academic Integrity policy

You are expected to abide by the Cornell University Code of Academic Integrity ([http://cunix.cornell.edu/aic.cfm](http://cunix.cornell.edu/aic.cfm)). It is your responsibility to understand and follow these policies. (In particular, the work you submit for course assignments must be your own. You may discuss homework assignments with other students at a high level, by for example discussing general methods or strategies to solve a problem, but you must cite the other student in your submission. Any work you submit must be your own understanding of the solution, the details of which you personally and individually worked out, and written in your own words.)
0. Review of basic python / jupyter notebook
1. Counting and probability (factorial, binomial coefficients, conditional probability, Bayes Theorem
   Real Data: text classifier, etc. [baby machine learning]
2. Statistics: mean, variance; binomial, Gaussian, Poisson distributions
3. Graph theory (nodes, edges), networks (c.f. Info 2040), graph algorithms
4. Power Law data (need exponential and logarithms …)
5. Linear and Logistic regression, Pearson and Spearman correlators
6. Markov and other correlated data

Rosen chapters 2,6,7,10,11
Easley Kleinberg chpts 3,18
+ many other on-line resources
   [mentioned, e.g. Berkeley “Foundations of Data Science”
   https://data-8.appspot.com/sp16/course ]
Problem sets will involve both programming and non-programming problems.

**Problem sets are not group projects.**

You are expected to abide by the Cornell University Code of Academic Integrity. It is your responsibility to understand and follow these policies. (In particular, the work you submit for course assignments must be your own. You may discuss homework assignments with other students at a high level, by for example discussing general methods or strategies to solve a problem, but you must cite the other student in your submission. Any work you submit must be your own understanding of the solution, the details of which you personally and individually worked out, and written in your own words.)

You’ll be penalized if you copy an iPython notebook, OR if yours is copied.
“Problem Set 0”, due in one week
to be posted later today:

will include instructions for installing anaconda,
we'll standardize on python 3
due to minor python 2.7/3.5 compatibility issues
(though welcome to use python 2)

known problem with python installations:
cs 1110 unfortunately recommends misconfigured software that
violates standard practice by adding environment variables to
~/.bashrc file. (link to instructions for removing)
Definition. A set is a collection of objects.

The objects of a set are called elements of the set.

\( x \in S \), or \( x \notin S \)

Examples:

\( X = \{1, 2, 3, 4, 5\} \)

\( C = \{\text{Ithaca, Boston, Chicago}\} \)

\( \text{Stuff} = \{1, \text{snow, Cornell, y}\} \)

empty set = \( \emptyset \)

Cardinality \( |S| \) is the number of elements of \( S \)

Examples: \( |X| = 5, |C| = 3, |\text{Stuff}| = 4, |\emptyset| = 0 \)

Can also be defined by rule or equation:

Example: \( E \) is the set of even numbers. \( E = \{x | x \text{ is even}\} \)
Examples:

\[ X = \{1, 2, 3, 4, 5\} \]

\[ C = \{ \text{Ithaca, Boston, Chicago} \} \]

\[ \text{Stuff} = \{1, \text{snow, Cornell, y} \} \]

empty set = \emptyset

Can also be defined by rule or equation:

Example: \( E \) is the set of even numbers. \( E = \{ x \mid x \text{ is even} \} \)

- A subset \( T \) of a set \( S \) is a set of elements all of which are contained in \( S \).

\[ T \subset S \] (proper subset) or \( T \subseteq S \)

empty set \( \emptyset \in S \) for all \( S \)

Examples:

\[ \text{C'} = \{ \text{Ithaca, Chicago} \} \]

\[ \text{C'} \subset C \]

\[ X' = \{ x \mid x \text{ is a whole number between 2 and 5} \} \] is a subset of \( X \)
For two sets to be the same, must have the same elements.

\[ A = B \text{ means that } \forall x \text{ we have } x \in A \text{ iff } x \in B \]

(Equivalently \( A = B \) means that \( A \subseteq B \) and \( B \subseteq A \))
• The power set \( \mathcal{P}(S) \) of a set \( S \) is the set of all subsets of \( S \).

Example: For the set \( A = \{1, 2, 3\} \), \( \mathcal{P}(A) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\}\} \)

For a set \( S \) with \( n \) elements, what is \( |\mathcal{P}(S)| \)?

• Cartesian product of two sets \( A \times B = \{(x, y) \mid x \in A \text{ and } y \in B\} \)

Example:

\[
A \times A = \{(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3)\}
\]
Definition. A set is a collection of objects. The objects of a set are called elements of the set.

\[ x \in S, \text{ or } x \notin S \]

Examples:

\[ X = \{1, 2, 3, 4, 5\} \]
\[ C = \{\text{Ithaca, Boston, Chicago}\} \]
\[ \text{Stuff} = \{1, \text{snow, Cornell, y}\} \]
\[ \text{empty set} = \emptyset \]

Can also be defined by rule or equation:

Example: \( E \) is the set of even numbers. \( E = \{x \mid x \text{ is even}\} \)

\[ A = \{1, 2, 3\} \]

**Set Operations**

- union of two sets \( A \cup B = \{x \mid x \in A \text{ or } x \in B\} \)

Examples:

\[ X \cup \text{Stuff} = \{1, 2, 3, 4, 5, \text{snow, Cornell, y}\} \]
\[ C \cup \emptyset = \{\text{Ithaca, Boston, Chicago}\} \]
\[ A \cup X = \{1, 2, 3, 4, 5\} \text{ (In this case, } A \cup X = X). \]

- intersection of two sets \( A \cap B = \{x \mid x \in A \text{ and } x \in B\} \)

Examples:

\[ X \cap \text{Stuff} = \{1\} \]
\[ C \cap \emptyset = \emptyset \]
\[ X \cap E = \{2, 4\} \]
\[ A \cap X = \{1, 2, 3\} \text{ (In this case } A \cap X = A) \]
Definition. A set is a collection of objects. The objects of a set are called elements of the set.

Examples:

\( X = \{1, 2, 3, 4, 5\} \)

\( C = \{\text{Ithaca, Boston, Chicago}\} \)

\( \text{Stuff} = \{1, \text{snow, Cornell, y}\} \)

empty set = \( \emptyset \)

Can also be defined by rule or equation:

Example: \( E \) is the set of even numbers. \( E = \{x \mid x \text{ is even}\} \)

\( A = \{1, 2, 3\} \)

- difference of two sets \( A - B = \{x \mid x \in A \text{ and } x \notin B\} \)

Examples:

\( X - \text{Stuff} = \{2, 3, 4, 5\} \)

\( \text{Stuff} - X = \{\text{snow, Cornell, y}\} \)

\( C - \emptyset = \{\text{Ithaca, Boston, Chicago}\} \)

\( X - E = \{1, 3, 5\} \)

- symmetric difference \( A \triangle B = \{x \mid x \in A \text{ or } x \in B, \text{ and } x \notin A \cap B\} \)

Examples:

\( X \triangle \text{Stuff} = \{2, 3, 4, 5, \text{snow, Cornell, y}\} \)

\( C \triangle \emptyset = \{\text{Ithaca, Boston, Chicago}\} \)

\( X \triangle A = \{4, 5\} \)