CSCI 3022 Spring 2021
Intro to Data Science

See: Zoom poll!

Instructor: Dr. Zachary Mullen

Hello and Welcome!

Syllabus Material

The course Canvas page will house:

1. The course syllabus and schedule
2. Annotated lectures and their videos posted after completion of the lecture
3. Homework Assignments and their turn-ins locations
4. Grades
5. Links to In-Class Notebooks, Data sets, and course Piazza page (register!)
6. Whatever else is necessary

Piazza: `https://piazza.com/colorado/spring2021/csci3022`

1. Ask questions in Q & A forum (and answer other students' questions!)
2. Discuss work, but do not post solutions/vital code
3. Send private messages to faculty instead of email (keeps things organized)

do this to learn more, better!

Learning Goals

At the end of this class, students should be able to:

1. load a data set into Python, clean and munge the data, perform exploratory data analysis, and report on patterns and correlations in the data,
2. compute and interpret various measures of central tendency, such as the mean, median, and mode; and measures of dispersion, such as variance, and standard deviation,
3. write the axioms of probability theory, prove basic theorems of probability theory, and apply those theorems to solve "real-world" problems involving chance events,
4. estimate population parameters of interest by calculating point and interval estimates from a sample/data,
5. perform statistical hypothesis tests,
6. construct and perform diagnostics on simple linear, multilinear, and logistic regression models to make predictions and inferences about data, and
7. construct basic data visualizations in Python and organize analyses, findings, and recommendations into easily interpretable reports in Jupyter Notebooks.

"Other" Courses

This course functions as a survey-level course for material sometimes found in:

1. Introduction to Data Science and Programming
2. Applied Probability (APPM 3570)
3. Introduction to Mathematical Statistics (APPM 4520)
4. Statistical Modeling/Regression (APPM 4590)

Coding Overview

1. We will use Python 3 and in particular (Numpy and Pandas) packages/libraries
2. Lot's of great data science libraries and decent plotting
3. We'll exclusively work in Jupyter Notebooks. We strongly recommend you install local copy
4. If not, you can use Microsoft Azure or Google Colab notebooks
5. Remote Learning can be tough for this: I'll *partially* work through the "in-class" notebooks most Fridays, but you're best suited spending some time before/after class making your own implementations!

Coding Assignments

1. Homework will be done through Jupyter notebooks, submitted into Canvas assignments.
   To install Jupyter on your computer:
   Jupyter: `http://jupyter.org/install.html`
   Anaconda Python: easiest!
   `https://www.anaconda.com/download/`
2. Back up your work! Use a regularly updating Google Drive, Github, secondary hard drive, etc. If it's a cloud-based backup, make the repo/drive **private**.

Python

What:
`Python` is free high-level programming language built for flexibility and simple syntax. It is commonly used in statistical computing and graphics.
Why we're using it:
It's widely used - especially in industry - free, and has a healthy repository of packages.
Common Syntax
Function Syntax: Functions use indents to determine the stopping point after a colon. The function `def myfunction(x):` ends after the indenting stops.
Indexing: Python is 0-indexed, and uses square brackets. For an $n \, x \, m$ matrix named `mydata`, `mydata[3,2]` accesses the entry in the fourth row and third column.
Comments: # comment
Favorite Reference: Official Documentation

Jupyter

What:
`Jupyter notebook` is free web application to combine running live code and visualizations.
Why we're using it:
Statistics is inherently interdisciplinary, and communication of clear results is paramount. The notebook environment encourages replicabile results and a clear workflow.
Common Syntax
Cells: The notebook is divided into cells. For our purpose, expository material will be done in Markdown cells with LaTeXcompatibility. Computational work will be done in Python 3 code cells, which may also generate plots, histograms, tables, and other output.
Formatting: # (with varying numbers of # signs) can be used to create section headers in markdown cells.
Comments: % comment
Favorite Reference: "Cheat Sheet"

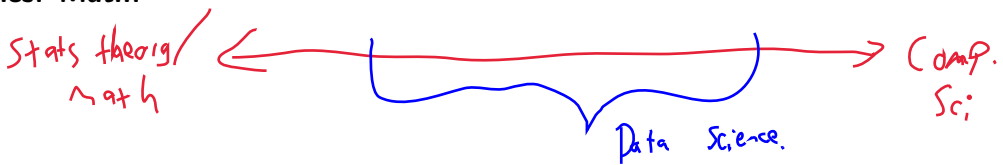
LaTeX

What:
LaTeXis a typesetting software with a particular emphasis on mathematics, including matrices, greek letters, etc.
Why we're using it:
Microsoft Equation Editor is a pain. It's included in `Jupyter` for Markdown cells.
Common Syntax
Function Syntax: Functions use curly brackets; "\textit{arg}" would italicize the argument
Math mode: inputting "$ arg $" will apply mathematical typesetting to the argument.
Comments: % comment
Favorite Reference: https://en.wikibooks.org/wiki/LaTeX

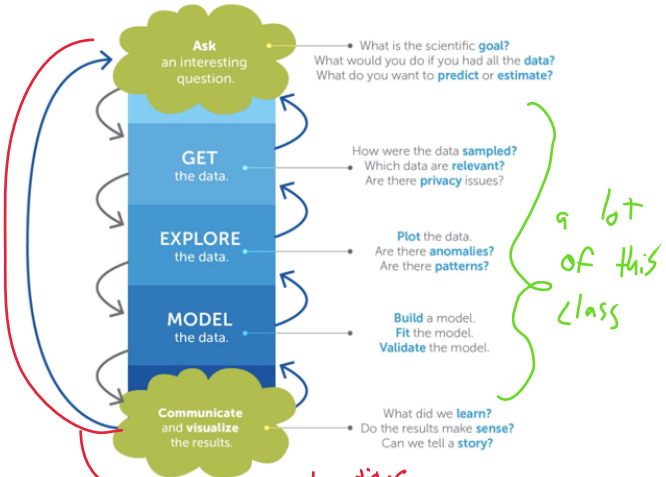
What is Data Science?

1. Making the invisible visible
2. Recovering insights/trends hiding within the data
3. Using data to answer interesting questions
4. Catch-all: using data to understand the world around us

Warning Label: we will do a lot of the "science" side of "data science" **Probability! Statistics! Math!**



Science!



From towardsdatascience

1. Hypothesis
2. Observations
3. Analysis
4. Conclusions
5. Refinements (repeat!)

Foundations

Realms	Topics
Probability	EDA, null models and hypotheses, Markov models
Statistical Inference	averages, regression models, MLEs
Optimization and Calculus	model fitting, computational shortcuts
Linear Algebra	Many many matrices...
CS	data structures, rapid estimations, simulation

"best" { maximums, minimums

Simulations are the last 15-20 yrs of Stat. theory.

The Plan

Week	Date	nb	txt	Topic	Slides	Hmwk
1	8.27			Course & Computing Introduction		
	8.29		16.1-3	EDA and Summary Statistics		
	1.26		2	Introduction to [illegible]		
2	9.03			LABOR DAY - NO CLASS		
	9.05		15	EDA and Data Visualization		hw1 posted
	9.08			**Data Wrangling**		
3	9.10		2,3	**How to Python**		
	9.12		6	Axioms and Theorems of Probability		
	9.14		3	**Stochastic Simulation**		hw1 due
4	9.17		4	Bayes' Rule and Intro to PDFs		hw2 posted
	9.19		4,5	Discrete RVs: PMFs, CMFs		
	9.21			Discrete RVs Strike Back		
5	9.24		5	**Return of the Discrete RVs**		
	9.26			Continuous RVs Awaken, PDFs, CDFs		
	9.28		7	**The Last Continuous RVs**		hw2 due
6	9.31		7	Expectation		hw3 posted
	10.03			Variance		
	10.05		5.5	**More Expectation & Variance**		
7	10.08			The Normal Distribution		
	10.10		14	**MIDTERM EXAM REVIEW**		
	10.10			The Central Limit Theorems		
	10.12			**MIDTERM EXAM (PM)**		hw3 due
8	10.15		23,24	**The Central Limit Theorem and You**		hw4 posted
	10.17		23,24	Inference and CI Intro		
	10.19			Two-Sample [illegible]		
9	10.22		25,26	[illegible] the Wild		
	10.24			Hypothesis Testing Intro		
	10.26			p-Values		hw4 due
10	10.29		27	**Practical HT & p**		hw5 posted
	10.31			Small-sample HT		
	11.02			TBD		
11	11.05		18,23.3	Bootstrap Intro		
	11.07			**Bootstrap and Small n HT**		
	11.09		27	OLS/SLR Regression		hw5 due
12	11.12			Inference in SLR		hw6 posted
	11.14			*Hands on inference in SLR*		
	11.16			MLR		
13	11.19			**FALL BREAK - NO CLASS**		
	11.21			**FALL BREAK - NO CLASS**		
	11.23			**FALL BREAK - NO CLASS**		
14	11.26		ISL Ch3	Inference in MLR		practicum posted
	11.28		ISL Ch3	More MLR and ANOVA I		
	11.30		ISL Ch3	ANOVA II		hw6 due
15	12.03			**ANOVA + Inference in MLR**		
	12.05			Logistic Regr. & Classification		
	12.07			**Logistic Regr. & Classification**		
16	12.10			Solution Techniques and SGD		
	12.12			**FINAL EXAM REVIEW**		practicum due
X	12.XX			**FINAL EXAM **		

Exploratory data analysis

Prob Thy

Probability theory & simulation

Modeling, classification, prediction

Hypothesis testing, inference

stat theory

Evaluations

Workload:

- (48%) Homework assignments (every 1-2 weeks, lowest dropped, late days)
- (13%) Midterm exam
- (13%) Final exam (cumulative)
- (10%) Practicum 1 (midterm)
- (10%) Practicum 2 (final)
- (6%) Participation (Canvas each Sunday)
- $\geq 55\%$ exam average required to earn a C- or higher in the class Let me know about any special needs in a timely manner Read the syllabus! More details can be found there regarding course policies (see: Late days!)

Remote Materials

All lectures for this class will be hosted remotely via Zoom. Our section is section 1, and meets from 10:20am-11:10am on MWF. The zoom link is `https://cuboulder.zoom.us/j/96586645524`, and will be open around 10:15am most days. All lectures will be recorded and posted to the course schedule.

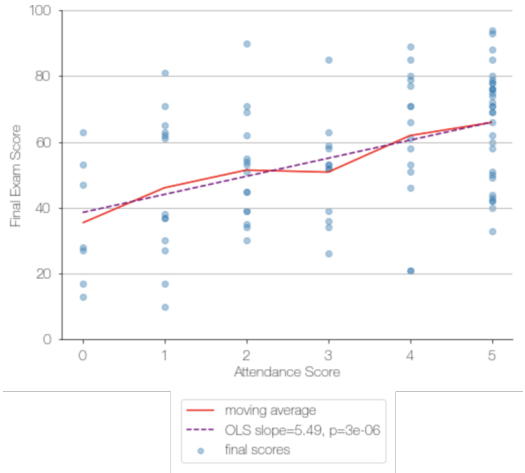
1) There will often be a warmup/intro problem to complete if you arrive between 10:15-10:20pm.

2) I will try to make Zoom as interactive as possible: use Zoom reactions, raise your hand if you have questions, and answer polls as I put them out.

3) It is my *strong* preference to have cameras on if your bandwidth can support it. It helps people feel invested and engaged in the process!

4) Fridays will typically be coding/application based - you are highly encouraged to follow along and attempt the exercises *before* class. M/W will be heavier on theory: pen-and-paper exercises and annotations on slides.

Attend!



Correlation...

Try to stay engaged! Take minute papers seriously, and ask questions through any/all mediums available (Zoom, Piazza, minute papers)

The curse of Laptops

"Results showed that students who used laptops in class spent considerable time multitasking and that the laptop use posed a significant distraction to both users and fellow students. Most importantly, the level of laptop use was negatively related to several measures of student learning, including self-reported understanding of course material and overall course performance."

I know it's a challenge learning remotely! Try to stay focused and hold yourself accountable to a routine with minimal distractions!

http://www.sciencedirect.com/science/article/pii/S0360131506001436
Also: http://journals.sagepub.com/doi/pdf/10.1177/0956797616677314
And: http://www.sciencedirect.com/science/article/pii/S0272775716303454

If at first you don't succeed...

1. When you're asking for help, be sure to explain...
2. what you're trying to do
3. what you think should happen
4. what you get instead (copy/pastes or screenshots work well)
5. what all you have tried
6. if you haven't tried anything, try something first

Learning New Software

There are 3 major tools to use in learning new software:

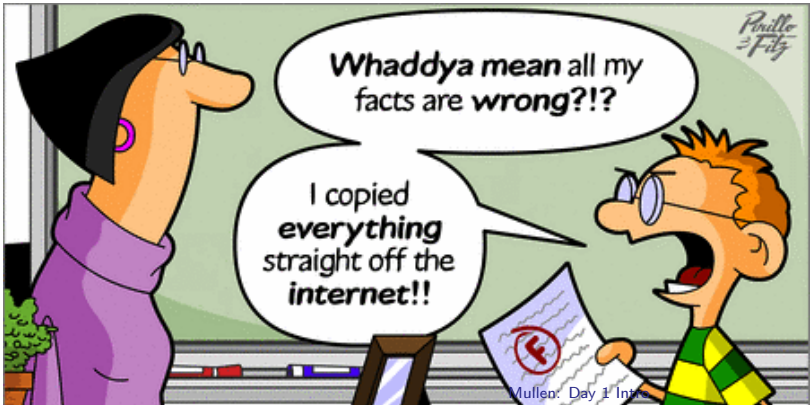
1. Pirating similar code found **from course materials**, etc.
2. Official documentation
3. Google searches, often directed to sites like stackexchange. (Don't Copy/Paste! Write from pseudo code, and *cite any sources* if you use them!)

Use (1.) and (2.) often, but be very careful with #3..., and don't hesitate to

1. Ask your instructor or peers for ideas on how to write specific routines, or for their syntax knowledge. Piazza is made for exactly this sort of thing!

Academic Integrity

1. See the CU Academic Integrity Policy for more details. Here are some highlights. "Examples of cheating include: copying the work of another student during an examination or other academic exercise (includes computer programming)"
2. "Examples of plagiarism include: . . . copying information from computer-based sources"



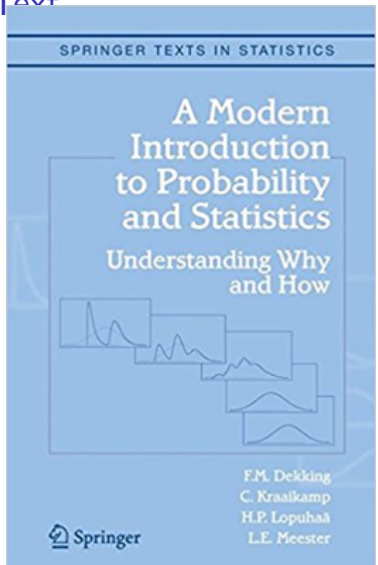
Integrity Examples

Example 1: For an assignment, Chris searches the internet for relevant codes and copy-pastes them into his Jupyter Notebook. He properly cites the source of the codes.

Example 2: For an assignment, Maciej and Felix work together to figure out how to implement the codes, but each works on their own computer and develops their own software.

Example 3: For an assignment, Rhonda has a plan for how to implement an algorithm, but isn't sure how to manipulate a Python list in a particular way that she needs to. She searches the internet, finds a fix, and implements it in her code without copying it.

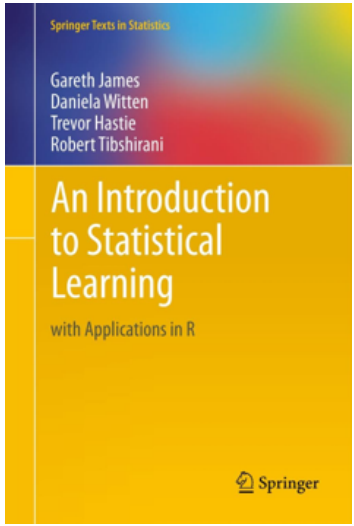
Text



A Modern Introduction to Probability and Statistics (MIPS)
by Dekking (et al.)
International, older, and PDF editions will work: just make sure to match any section numbers that changed.
Free PDF edition through CU (CU network, or VPN):
`https://www.springer.com/us/book/9781852338961`
Additional reading will be linked to the course calendar as needed

Other Texts

Think Stats by Downey ("TS")



An Introduction to Statistical Learning ("ISL")

Moving Forward

Let's get to work!

- Before next class:
  1. Make sure you can access the Canvas page and read the syllabus
  2. Set up some way to back up your work
  3. Install Anaconda (or other reliable Jupyter notebook method)
  4. Review and complete Numpy/Pandas tutorial

nb 00
nb 00a