Gov 50: Data

David Kane

Fall 2020

Description

Data matters. Learning to think critically about data is a fundamental skill. How much money is donated to political campaigns? Does exposure to Spanish-speakers affect attitudes toward immigration? What characteristics are associated with voting Republican? We need data to answer these questions – to describe, to predict, and to infer.

This course, an introduction to data science, will teach you how to *think with data*, how to gather information from a variety of sources, how to import that information into a project, how to tidy and transform the variables and observations, how to visualize, how to model relationships, how to assess uncertainty, and how to communicate your findings. Each student will complete a final project, the first entry in their professional portfolio. Our main focus is data associated with political science, but we will also use examples from education, economics, public health, sociology, sports, finance, climate and any other topic which students find interesting.

We use the R programming language, RStudio, Git and GitHub.

Prerequisites: None. You must have a laptop with R, RStudio and Git installed.

Logistics: There are two sessions: 7:30 AM to 8:45 AM and 12:00 PM to 1:15 PM, both on T/TH. I will teach both sessions. You must attend one (either morning or afternoon) consistently all semester, unless you have spoken to course staff about an exemption.

Figure 1: *Ulysses and the Sirens*, 1891, by John William Waterhouse. Homer's *Odyssey* recounts the decade-long journey home of Odysseus (known as Ulysses in Latin) after the Trojan War. Although Ulysses's ultimate goal is his kingdom of Ithaca, he does not shy away from adventure along the way. The Sirens use their enchanting voices to lure unwary sailors to their deaths. Ulysses wanted to hear their songs. He instructed his men to fill their ears with beeswax and to tie him to the mast.

Course Metaphor

The central metaphor for this class is *Ulysses and the Sirens*. You are Ulysses. Ithaca is where you hope to arrive after graduation. The Sirens are the many distractions of the modern world. *I am the rope.*

Course Promise

No course at Harvard does a better job of increasing your odds of getting the future — the internship, the job, the graduate school, the career — that you want. The best way to decide whether or not this class is for you is to look at our final projects. If you want to learn how to build something like that, take the class.

Course Staff

Preceptor David Kane; dkane@fas.harvard.edu; CGIS South 310; 646-644-3626. Office hours via Zoom on Wednesdays (sign up at Calendy) or by appointment. Please address me as "Preceptor," not "David," nor "Preceptor Kane," nor "Brah," nor "Professor Kane," nor "Mr. Kane," nor, worst of all, "Dr. Kane." I respond to e-mail within 24 hours. If I don't, e-mail me again.

Head Teaching Fellow: Tyler Simko (tsimko@g.harvard.edu, Zoom).

E-mail Tyler (and cc your assigned TF and Preceptor) if you have a complaint about a grade, or any other issue.

Teaching Fellows: Dan Baissa (dbaissa@g.harvard.edu, Zoom); Mitchell Kilborn (mkilborn@g.harvard.edu, Zoom); Shivani Aggarwal (saggarwal@college.harvard.edu, Zoom); Rucha Joshi (ruchajoshi@college.harvard.edu, Zoom); Beau Meche (beau_meche@college.harvard.edu, Zoom); and Wyatt Hurt (wyatthurt@college.harvard.edu, Zoom).

Head Course Assistant: Yao Yu (yaodongyu@college.harvard.edu)

Course Assistants: Evelyn Cai (evelyncai@college.harvard.edu), Kayla Manning (kaylamanning@college.harvard.edu), Kevin Wang (kpwang@college.harvard.edu), Lindsey Greenhill (lgreenhill@college.harvard.edu), Eliot Min (eliotmin@college.harvard.edu), Miroslav Bergam (mbergam@college.harvard.edu), Ishan Bhatt (ishanbhatt@college.harvard.edu), Emma Freeman (efreeman@college.harvard.edu), Sam Lowry (samlowry@college.harvard.edu), Erin Guetzloe (eguetzloe@college.harvard.edu), Chelsea Marlborough (chelseamarlborough@college.harvard.edu), and Amy Zhou (amyzhou@college.harvard.edu).

Course Philosophy

No Lectures: The worst method for transmitting information from my head to yours is for me to lecture you. There are no lectures. We work on problems together during class. You learn soccer with the ball at your feet. You learn about data with your hands on the keyboard.

Bayesian: The philosophy of this class is unapologetically Bayesian.

R Everyday: Learning a new programming language is like learning a new human language: You will practice (almost) every day.

Community: You will probably learn the names of more students in this course than in all your other courses combined. *Awkwardness in the pursuit of community is no vice.*

Professionalism: We use professional tools. Your workflow will be very similar to the workflow involved in paid employment. Your problem sets and final project will be public, the better to impress others with your abilities. High quality work will be shared with your classmates. We will learn the "full cycle" of how to draw inferences from data and communicate those inferences to others. We will network.

Cold Calling: I call on students during class. This keeps every student involved, makes for a more lively discussion and helps to prepare students for the real world, in which you can't hide in the back row. Want to be left alone? Don't take this course.

Recitations: We do not have normal sections. Instead, you will spend 60 minutes each week meeting with your assigned TF in a small group in the first half of the semester. Recall what I just said about (not) being left alone in this course. In the second half of the semester, you will have one-on-one meetings with your TF for 30 minutes each week, focused on your final project.

Millism: Political disputes are not the focus of this class but, when such topics arise, I will insist that we follow John Stuart Mills' advice: "He who knows only his own side of the case, knows little of that. His reasons may be good, and no one may have been able to refute them. But if he is equally unable to refute the reasons on the opposite side; if he does not so much as know what they are, he has no ground for preferring either opinion."

Engagement: We require you to be engaged with the outside world. For example, you are required to email alumni and seek to meet with them about their careers. Several of the course assistants are Mindich Course Fellows, meaning that they work closely with the Mindich Program in Engaged Scholarship to help students connect their final projects to someone outside Harvard interested in what you are interested in.

Speaker Series: We host the Gov 50 Speaker Series: Data Scientists, Data Professionals, Data Dissidents on Fridays. Attendance is neither taken nor required. Members from the Harvard community are invited, in addition to students from Gov 50, although we do try to provide preferences to our students in the Q&A. Contact Preceptor if you are interested in helping out.

Teaching to Learn: My main goal is not to teach you how do X. That is easy! More importantly, in a few months, I won't be around to teach you Y. My goal is to teach you how to teach yourself, how to figure out X and Y and Z on your own. That is harder! Much of the pedagogy of the course — especially my insistence that you work on topics not covered in lecture — is driven by this goal. You will find it frustrating.

Course Policies

Book: The text for the class is *Preceptor's Primer for Bayesian Data Science*. We call it *The Primer* for short. The book is still a draft and contains many mistakes. Please help us by pointing them out!

Workload: The course should take about 10 hours per week, outside of meetings, exams and the final project. This is an expected average across the class as a whole. *It is not a maximum.* Some students will end up spending much less time. Others will spend **much, much more**.

Late Days: Assignments (Tutorials, Problem Sets and Final Project Milestones) are always due at 11:55 PM, unless specified otherwise. An assignment is a day late if it is turned in any time after it was due (even 5 minutes after) but within 24 hours. After that, it is two days late, and so on. You have 5 *late days* in total. Late days may be used for any assignment, except the four exams and the final project Demo Day. **You should save your late days.** If you use them early in the semester for no particularly good reason and then, later in the semester, have an actual emergency, we will not be sympathetic. We will not give you extra late days in such a situation. (That isn't fair to your classmates, and we are all about fairness.) We will just, mentally, move the late days you wasted so that they cover your actually emergency. You will now be penalized for being late earlier in the semester, when you did not have a good reason for tardiness. You may only use one late day on a given assignment. Hand it in after more than 24 hours and you get a zero on that assignment. **But you still must hand it in!** Everything must be completed. Late days accrue until you do. Each day late (beyond the five allowed) results in -1 point to your final score. This decrement is a *point* not *percentage* penalty. In other words, each additional late day used outside of the allotted five will drop your final class average by one point out of 100.

Submissions: All tutorials, problem sets, milestones and exams are turned in via Canvas. Late days are assigned on the basis of the official Canvas submission time.

Missing Class: You expect me to be present for lecture. I expect the same of you. There is nothing more embarrassing, for both us, than for me to call your name and have you not be there to answer. But, at the same time, conflicts arise. It is never a problem to miss class if, for example, you are out of town or have a health issue. Simply put an X by your name and in the correct date column in the Google absence sheet **and** send me and your assigned TF an e-mail explaining the situation. Failure to do so will decrease your participation points, as will missing too many classes, even with notification. There is no need to put a reason in the sheet. An X is enough.

Major Emergencies: We are not monsters. If you are hit with a major emergency — the sort of thing that necessitates the involvement of your Resident Dean — we will be sympathetic. Speak with your TF.

Monologues: I give brief monologues, designed to explain specific topics that have confused students in the past. I hope to never talk for more than 5 minutes straight.

Speakers: We will have a speaker series on Fridays. These are 100% optional. More information to come.

No Cost: Every reading/tool we use is free. You don't have to spend any money on this class. Some activities, like Shinyapps and GitHub, have paid options which provide more services, but you never have to use them. Don't give anyone your credit card number.

Role of Teaching Fellows: The TFs run their Recitations, approve all grades, deal with emergencies and so on. Go to them first with any problems. You will be assigned to work closely with a specific TF — your "assigned" TF — but you may ask other TFs for help as well.

Role of Course Assistants: The CAs run Study Halls. They can make no commitments about how the TFs will assign the final grade on a problem set, milestone or exam. *Never ask a CA a question about grading.* Instead, ask on Slack and a member of the course staff will respond, or come to a TF privately with your question.

Exceptions: There may be a reason why you can't adhere to class policies. For example, severe social anxiety may make being cold-called problematic. A learning disability may make take-home tests unfair. Whatever the situation, please seek me out for conversation. I am sure we can work out something! I will do whatever it takes to allow every Harvard student to thrive in this class.

Use your Harvard e-mail: Please use your official Harvard e-mail address for all aspects of this class, especially things like signing up for services like shinyapps, GitHub, and so on. Doing so makes it much easier for us to figure out who is doing what. This may not be easy if you already connect with these services but, even in that case, you should be able to add your Harvard e-mail address to your account.

Slack: All general questions — those not of a personal nature — should be posted to Slack so that all students can benefit from both the question and the answer(s). Please post your question in a sensible channel.

Plagiarism: If you plagiarize, you will fail the course. See the Harvard College Handbook for Students for details.

Working with Others: Students are free (and encouraged) to discuss problem sets and their final projects with one another. However, you must hand in your own unique code and written work in all cases. Any copy/paste of another's work is plagiarism. In other words, you can work with your friend, sitting side-by-side and going through the problem set question-by-question, but you must **each type your own code**. Your answers may be similar (obviously) but they must not be identical, or even identical'ish.

Git and GitHub: Analyzing data without using source control is like writing an essay without using a word processor — possible but not professional. We will do all our work using Git/GitHub.

Readings: Assignments in a given week cover (approximately) the material that we will use that week. I will not hesitate to cold-call students with questions about the readings. Do them. Note that the readings must be done before Tuesday class or before your Recitation, whichever is earlier. For example, if you have Monday Recitation, the readings should be done before that.

Optional Activities: The syllabus includes background readings, videos and materials which students may find interesting. You do not have to do them.

Computer Emergencies: We are not sympathetic about computer emergencies. You should keep all your work on GitHub, so it won't matter if your computer explodes. If it does explode, you will lose only the work after your last push. You can then restart your work on a public computer (the basement of CGIS Knafel has machines with R/RStudio installed) or on your roommate's computer.

Github Classroom: We use Github Classroom to distribute problem sets and exams. You will receive an e-mail with a link. Click on that link and a repo, with instructions, will be created. *Do this as soon as you receive the e-mail.* We don't want GitHub problems to arise the night before the assignment is due.

Tardiness: We begin on time and end on time.

Credit: Gov 50 fulfills the QRD requirement. You may also get concentration credit. This is true, obviously, for Government. It is also true in Statistics, Psychology, Sociology, and Social Studies. I am happy to support students who want to petition other departments.

Announcements: You are responsible for any assignment/exam/deadline updates/changes which are either announced in class or promulgated via the course Canvas e-mail list. The official Preceptor's Notes, posted to the Slack channel #preceptors-notes, are important, but we will e-mail them to you. You are not responsible for every other random post on Slack. In fact, you can ignore Slack completely, if you want.

Grading

Solo Participation: 5 points. This category relates to things you do alone in class. Missing class (without notifying us) or missing too many classes will cost you points, as will a failure to participate in class activities. Note that I do not care if you know the answer when I cold-call you. This plays no part in your grade. The most common example in this category is required e-mails, which we do in class. Whenever you send such an e-mail, you must bcc both your assigned TF and gov50data@gmail.com.

Group Participation: 5 points. This category relates to activities you do with other students. Helping your fellow students, especially on Slack, is the best form of group participation, as is volunteering for a class role like scribe. Be a good class citizen. Help your classmates during Study Halls. Speak up during Recitations. Do not shirk on group projects.

Tutorials: 5 points. Tutorials are distributed in the **PPBDS.data** package. Make sure to install the latest version before you start a new tutorial. Grades are pass/fail only. Given the level of the questions and the hints provided, it is essentially impossible not to get full credit as long as you make an honest effort. If a question is too hard, just provide your best guess. If something seems broken, just skip it and go on. There are hundreds of questions. We don't care if you miss a few, or even a bunch. But, if you just give up or enter nonsense answers, then you and your TF are going to have an unpleasant conversation . . .

Problem Sets: 25 points. Follow these instructions. The first problem set is worth 1 point. The remaining 8 are worth 3 points each. Problem sets after the first are distributed on Thursday and then due the following Wednesday at 11:55 PM. You are welcome to work on them with your friends but, first, you must personally type in every character in the work you submit and, second, you must list all the people you worked with. We define "work with" very broadly, to include minor interactions. You would certainly list anyone you sat nearby during Study Hall, for example.

Exams: 35 points total. The four exams are take-home and unhackable. The first is worth 5 points and the others are each worth 10 points. They are open-book and open-web. Because students have different schedules, you can complete the exam any time within a four-day window starting after exam distribution. Late exams earn zero points. You may not seek or receive help on the exam from a person, e.g., asking a roommate or posting at RStudio Community. You may use any written materials from the class, including problem set answers. If you have a question, ask on Slack. Teaching (not other students) will answer it.

Final Project: 25 points. Students will present their projects publicly at the end of the semester. They will then have the opportunity to incorporate feedback before submitting the final version. There are eight milestones for the projects, each worth one point. Demo Day (which includes a review of your code) is worth 7 points. The final project submission is worth 10 points. Follow the course style guide.

Calculation: Each problem set, milestone and exam is graded out of a maximum of score of 20, regardless of its weight in the final grade calculation. For example, both Exam 2 and Milestone 2 are graded out of 20, but the former is worth ten times as much to your final grade.

Final Project

Do you love soccer or wine or NYC politics? The final project provides you with an opportunity to study that topic in depth. Your final project will be, for most of you, the first item in your professional portfolio, something so impressive that you will be eager to show it to graduate schools or potential employers. You must show this work publicly, both on the web (viewable by all) and in person at our Demo Day. You will host your final project using Shiny Apps. Make use of free statistical consulting from the Harvard Statistics Department and from IQSS. Read this advice if you are working with data larger than 100 megabytes. Consider scheduling an interview with Hugh Truslow (truslow@fas.harvard.edu), Head, Social Sciences and Visualization, Harvard University. No one at Harvard knows more about potential data sources. Visualization Specialist Jessica Cohen-Tanugi (jessica_cohen-tanugi@harvard.edu) is a great person to talk to about your graphics. Explore the final projects from past semesters.

There are four key components to every final project. First, you must have made a meaningful effort to collect and clean your data. Typing `library(fivethirtyeight)` is not enough. Second, your Shiny app must, on at least one panel, be interactive. It must provide the viewer with a choice of some sort and then report results which depend on that choice. Third, there must a statistical model, along with an associated discussion of its creation and interpretation. This generally occurs in a panel named "Model." It should include a nicely formatted table, along with a written interpretation which explains the meaning of (at least) some of the more important parameters. Fourth, there must be an "About" panel which provides an overview of the project, including a discussion of data sources. It must also provide a link to your Github repo for the project.

Other instructions: Follow the Style Guide. Use an informative name for both your repo and the app itself. (Do not name your repo final-project or gov-50. Call it something informative.) When opened, the Shiny app should default to an "interesting" panel, presumably something with graphics. You want to grab the reader's attention.

Possible Approaches

Most students will gather some data, estimate some models, and create a Shiny App. Good stuff! But there are other possible approaches:

Original Data Collection

Students interested in a topic about which there is no publicly available data are welcome to collect their own data. This must be something much more substantive than just asking 100 students outside Annenberg about their favorite salad. Two categories of data work best. First, pick a topic which you truly care about. Second, pick something Harvard-specific. This *Crimson* article and these class projects — spring 2019 and fall 2019 — are great examples of the latter.

Work with Other Classes

You are welcome to use data from your other classes in the creation of your final project. This includes thesis work. You automatically have permission from us to do this, but you must also obtain permission from the instructor of the other class.

Others?

Interested in doing a project which seems different from what we describe above? Come talk to me! The best projects involve topics which students are passionate about. If you really care about X, then we are eager to help you create a final project about X. Examples: participation in the NFL Big Data Bowl, submitting Numerai forecasts or entering a Kaggle competion.

Conclusion

If you had tried to complete a data analysis project before taking this class, you would have done X well. Now that you have taken the class – now that you know how to describe, predict and infer – you will do Y well. The success (or failure) of the class can be measured by comparing Y with X.

Organization

Everything — Tutorials (Mondays), Problem Sets (Wednesdays), Milestones (Fridays) and Exams (Sundays) — is due at 11:55 PM, unless otherwise specified.

Rhythm of the Class

The class follows a steady weekly rhythm:

Monday 11:55 PM. Tutorials are due.
Tuesday 7:30 – 8:45 AM or 12:00 PM – 1:15 PM. Class.
Wednesday 11:55 PM. Problem sets are due.
Thursday 7:30 – 8:45 AM or 12:00 PM – 1:15 PM. Class.
Thursday evening. Problem set due next week will be distributed.
Friday 11:55 PM. Final project milestones are due.
Sunday 11:55 PM. Exams, if distributed, are due.

Key Dates

Part 1: Tools and Framework

Tutorial #1 due Monday, September 7.
Tutorial #2 due Monday, September 14.
Problem Set #1 due Wednesday, September 16.
Milestone #1 due Friday, September 18.
Tutorial #3 due Monday, September 21.
Problem Set #2 due Wednesday, September 23.
Milestone #2 due Friday, September 25.
Tutorial #4 due Tuesday, September 29.
Problem Set #3 due Wednesday, September 30.
Milestone #3 due Friday, October 2.

Part 2: Sampling and Inference

Tutorial #5 due Monday, October 5.
Problem Set #4 due Wednesday, October 7.
Exam #1 distributed on Thursday, October 8 and due Sunday, October 11.
Tutorial #6 due Tuesday, October 13.
Milestone #4 due Friday, October 16.
Tutorial #7 due Monday, October 19.
Problem Set #5 due Wednesday, October 21.
Milestone #5 due Friday, October 23.
Tutorial #8 due Monday, October 26.
Problem Set #6 due Wednesday, October 28.
Exam #2 distributed Thursday, October 29 and due Sunday, November 1.

Part 3: Models

Tutorial #9 due Wednesday, November 4.
Milestone #6 due Friday, November 6.
Tutorial #10 due Monday, November 9.

Problem Set #7 due Wednesday, November 11.
Milestone #7 due Friday, November 13.
Tutorial #11 due Monday, November 16.
Problem Set #8 due Wednesday, November 18.
Exam #3 distributed Thursday, November 19 and due Sunday, November 22.
Thanksgiving, November 26.

Part 4: Conclusion

Milestone #8 due Wednesday, December 2.
Demo Days
Final project due Friday, December 11.
Exam #4 distributed Saturday, December 12 and due Sunday, December 20.

Study Halls

Friday 7:00 PM – 10:00 PM, Study Hall with Kevin Wang via Zoom
Saturday 4:00 PM – 7:00 PM, Study Hall with Kayla Manning via Zoom
Sunday 7:00 PM – 10:00 PM, Study Hall with Evelyn Cai via Zoom
Monday 9:00 PM – 12:00 AM, Study Hall with Emma Freeman via Zoom
Tuesday 1:00 PM – 4:00 PM, Study hall with Miro Bergam via Zoom
Tuesday 8:00 PM – 11:00 PM, Study Hall with Eliot Min via Zoom
Wednesday 7:00 PM – 10:00 PM, Study Hall with Ishan Bhatt via Zoom

Schedule

Part 1: Tools and Framework

Week 0: September 3

Shopping Week

Purpose of this week is to help you decide whether or not you want to take the class. No one is required to be here. Only you can decide whether the goals of the class, and the workload associated with meeting those goals, make sense for you. We will spend most of our time in breakout rooms. One breakout session will be devoted to Tutorial 0, your answers to which you must submit via Canvas. Another breakout room session will begin Tutorial 1, which is due on Monday.

Readings: Shopping Week

Workflow: you should have one. – Jenny Bryan

Reminder: Tutorial 1 is due Monday 11:55 PM

Week 1: September 7

Visualization

You are Ulysses. I am the rope.

You will have your first Recitation with your Teaching Fellow this week. In class, we will learn how to start and knit and R markdown file. We will also connect to Github and discuss the importance of source control. We send a thank-you e-mail.

Readings: Chapter 1 Visualization and Appendix on Tools.

Assignment: Tutorial 1 due Monday at 11:55.

Optional: Read and watch the videos from *Getting Used to R, RStudio, and R Markdown* by Chester Ismay and Patrick C. Kennedy.

Week 2: September 14

Tidyverse

You can never look at your data too much. – Mark Engerman

The first problem set will be distributed on Tuesday, via Github Classroom, and completed during class. We will submit that problem set via Canvas. For those who don't finish in class, the problem set is due September 16 at 11:55 PM. We send an e-mail to an alum.

Assignment: Tutorial 2 due Monday at 11:55.

Readings: Chapter 2 Wrangling and Maps.

Final Project Milestone #1 due Friday, September 18. Speak with your Teaching Fellow about your final project during your Recitation session or outside of it. Google Dataset Search is a good way to find data. See also these resources. Write one paragraph in an Rmd file about your current thoughts/ideas/plans. Knit the Rmd file into an html, and then submit the html via Canvas.

Optional: RStudio Essentials Videos. Most relevant for us are "Writing code in RStudio", "Projects in RStudio" and "Github and RStudio". Again, these are optional! But they are very useful for students who find traditional lectures to be a helpful supplement to classroom practice. See also *GitHub Classroom Guide for Students*.

Week 3: September 21

Rubin Causal Model

The best data science superpower is knowing how to ask a question. – Mara Averick

We will introduce the "potential outcomes" framework and review the fundamental problem of causal inference. We will discuss the slogan "no causation without manipulation." We will learn how to produce a **repr**oducible **ex**ample — a "reprex" — in order to help strangers to help us. We send another e-mail to an alum.

Assignment: Tutorial 3 due Monday at 11:55

Readings: Rubin Causal Model and Animation.

Problem Set #2 due Wednesday, September 23.

Final Project Milestone #2 due Friday, September 25. Github repo with Rmd (and knitted html) which discusses pros and cons two projects from past years. At least one project should be one which did extensive data gathering/cleaning. You should not select the same projects for commentary as your friends have. Students generally write about a paragraph for each project. The Rmd/html file should include the url for your repo. The only thing you are submitting is the html, via Canvas.

Optional: RStudio Webinar on Reprex. Again, these are optional! But they are very useful for students who find find traditional lectures to be a helpful supplement to classroom practice. Causality, Chapter 2 of *Quantitive Social Science* by Kosuke Imai, especially pages 46 – 63.

Week 4: September 28

Functions

Readings: Functions and Shiny.

Tutorial #4 due Tuesday, September 29.

Problem Set #3 due Wednesday, September 30.

Milestone #3 due Friday, October 9. Create a Shiny App. It will be mostly empty. But it must have at least two tabs, one of which will be the "About" tab. The About tab should include the url to your repo, should we want to examine it. Discuss all your proposed data sources. (If you are gathering Harvard data, you should have a draft of your survey questions.) Remember: You must gather data from two or more different sources. Learning how to source, clean and combine data is one of the goals of the project. On almost any topic, there are useful tables of information on Wikipedia. See here and here for advice. Submit the url for your Shiny App to Canvas. UPDATE: This assignment is canceled. All students receive full credit. These details become part of Milestone #4 due in two weeks.

Optional: The Unix Workbench, chapters 1 – 6.

Part 2: Sampling and Inference

Week 5: October 5

Probability

I stopped teaching frequentist methods when I decided they could not be learned. – Donald Berry

THERE IS NO SUCH THING AS PROBABILITY. — Bruno de Finetti

Readings: Probability.

Tutorial #5 due Monday, October 5.

Problem Set #4 due Wednesday, October 7.

Exam #1 distributed on Thursday morning, October 8 and due Sunday, October 11.

Optional: Statistical Rethinking: A Bayesian Course with Examples in R and Stan (pdf) by Richard McElreath. Chapter 1.

Week 6: October 12

One Parameter

Lot of points were taken off for small errors that I did not see as pedagogically important. – Gov 50 student

Tutorial #6 due Tuesday, October 13.

Readings: One Parameter.

Final Project Milestone #4 due Friday, October 16. Improve your Shiny App by adding a tab which does something with you data. With luck, you will have gathered all your data and placed it in the repo. (This

will generally be done with a different Rmd, like gather.Rmd, in your repo which contains the code which actually downloads your data.) You should have processed your data. (It is OK if you have not gotten quite this far as long as you discuss your progress and your plan in the About page.) This can be as simple as running `summary()`. You should still have an About tab. You may change your project completely, all the way until Demo Day. But you are still responsible for meeting these milestones, even if you know you are going to pivot. Your data can not be from a single source. Typing *library(fivethirtyeight)* is not enough! Submit the url for your Shiny App to Canvas. (This might be the same url as last week (with new material) or a new url at which you have started over.)

Optional: RStudio Webinar titled "How to Work with List Columns" by Garrett Grolemund. Background reading about anonymous functions in R.

Week 7: October 19

Two Parameters

Comment as a service to the dumbest possible version of your future self. – Alex Albright

Readings: Two Parameters and "Causal effect of intergroup contact on exclusionary attitudes" by Ryan Enos. PNAS March 11, 2014 111 (10) 3699-3704.

Tutorial #7 due Monday, October 19.

Problem Set #5 due Wednesday, October 21.

Final Project Milestone #5 due Friday, October 23. Add a beautiful graphic to your Shiny App, using **ggplot2** or another package of your choice, which uses some of your data. This can still be very rough. We just want some evidence that you have some data and that you are doing something with it. If your data plans are behind schedule, just let your TF know. But you still turn in something. Submit the url for your Shiny App to Canvas.

Week 8: October 26

Three Parameters

Amateurs test. Professionals summarize.

Readings: Three Parameters and Animation

Tutorial #8 due Monday, October 26.

Problem Set #6 due Wednesday, October 28.

Exam #2 distributed Thursday morning, October 29 and due Sunday November 1.

Optional: How to Start Shiny video tutorial.

Part 3: Models

Week 9: November 2

N Parameters

Fitting is easy. Prediction is hard. – Richard McElreath

Readings: N Parameters

Tutorial #9 due Wednesday, November 4.

Final Project Milestone #6 due Friday, November 6. You must have a working Shiny App. It can be a mess, but it must have at least one attractive graphic with your data. Submit the url for your Shiny App via Canvas.

Optional: Shiny tutorials.

Week 10: November 9

Model Choice

Readings: Chapter 9

Tutorial #10 due Monday, November 9.

Problem Set #7 due Wednesday, November 11.

Final Project Milestone #7 due Friday, November 13. Cleaned up Github account. Submit the url for your Github account to Canvas.

Optional: "The Bayesian New Statistics" by John K. Kruschke and Torrin M. Liddel.

Week 11: November 16

Continuous Response

Readings: Chapter 10

Problem Set #8 due Wednesday, November 18.

Exam #3 distributed Thursday, November 19 and due Sunday, November 22.

Week 12: November 23

Discrete Response

Put your work on the web. – David Sparks

Thanksgiving week. No class Thursday.

Readings: Chapter 12

Optional: Video lectures of generalized linear models with binary data, parts 1, 2 and 3.

Part 4: Projects

Week 13: November 30

Wrap Up

A public portfolio of high quality work is better than a Harvard degree.

Final Project Milestone #8 due Wednesday, December 2. Working rough draft of your Shiny App. You must have a fairly complete version of your current project: a Shiny app with your About page, your data and your model. Write a four sentence elevator pitch for your project and e-mail it to your TF. This pitch is how

you will begin each presentation during Demo Day. Submit the url for your Shiny App via Canvas. You must also fill out the final project information form.

Last day of classes. Make memes, provide course feedback, discuss final projects and have fun!

Optional: Mastering Shiny by Hadley Wickham.

Important: Check your grades on Canvas, including your calculated late days. Any questions/complaints must be made before the last day of classes. After that, no changes will be made.

Demo Day Sessions Follow our detailed instructions. Although your presentation is not, itself, graded, we will take off points for showing up late or otherwise messing up the process. Demo Days are public events. We welcome all who are interested in your work. You, also, must send out two e-mail invitations (bcc'ing your assigned TF), one to your non-Harvard family/friends and one to your Harvard friend(s). We strongly urge you to invite your parents and/or extended family, but this is not required.

Assignment Details

Participation

Slack: Answering your classmates questions on Slack is the best way to earn participation points. Be a good class citizen! If you find a (meaningful!) typo in a problem set or exam, please post it to Slack. The first student to do so earns many participation points.

Useful Links

Google sheets for Final Projects and Absences.

Overview of, and grading rubrics for, the problem sets and exams.

How we conduct Study Halls.

How to improve your Github account.

Possible data sources for final projects.

Technical advice which students should follow. Read this at least once before submitting Problem Set #2.

Follow this advice if you have computer problems.

List of my friends/acquaintences from the world of data science, all of whom are happy to talk with students in my class. Reach out to them!

Free R Books.

Acknowledgements

This course is inspired by STAT 545, created by the legendary Jenny Bryan. The pedagogical goals follow Don Rubin's vision. Some of the classroom exercises come from (*Statistical Inference via Data Science: A moderndive into R and the tidyverse*) by Chester Ismay and Albert Y. Kim. Slides were created via the R package **xaringan** by Yihui Xie. Many thanks to all the folks responsible for R, RStudio, Git and GitHub. This course would not be possible without their amazing contributions.