Winter 2025 | Lecture: Tues/Thurs., 9:30am-10:50am, | Room: Harper Memorial 135 (Lecture) | Units: 100

PLSC 40502: Data Analysis with Statistical Models

Anton Strezhnev

Office: Pick Hall 328, 3rd floor
Office Hours: Tuesdays 4pm-6pm or schedule an appointment by e-mail
astrezhnev@uchicago.edu
http://www.antonstrezhnev.com

Last Updated: 01/6/2025

Course Overview

Statistical models provide a structure for the analysis of data. Often many scientific questions revolve around drawing statistical inferences about some parameter, such as a regression coefficient. Models also allow researchers to generate predictions on new or out-of-sample data. Understanding the fundamentals of how to define, estimate and validate a statistical model is essential to the process of quantitative empirical research.

This course is part of the second year of the Quantitative Methodology sequence in the Department of Political Science and builds on the first year sequence (PLSC 30500, 30600, 30700). It will introduce students to likelihood and Bayesian inference with a focus on multilevel/hierarchical regression models. The overarching framework of this class is model-based inference for description and prediction – a complement to the design-based framework of PLSC 30600 Causal Inference. Students will learn both the theory behind Bayesian modeling as well as how to implement common estimators (e.g. Expectation-Maximization, Markov Chain Monte Carlo (MCMC)) in the R statistical programming language. Applied examples will be drawn from across the political science literature, with a particular emphasis on the analysis of large survey data (e.g. the American

National Election Survey (ANES), the Cooperative Election Survey (CES), the European Social Survey (ESS)).

This course will involve a combination of lectures and problem sets. Lectures will focus on introducing the core theoretical concepts being taught in this course as well as providing illustrations through worked applied examples. Problem sets will contain a mixture of both theoretical and applied questions and serve to reinforce key concepts and allow students to assess their progress and understanding throughout the course. Primary evaluation will take the form of a take-home midterm and final exam.

Assignments will involve analysis of data using the R programming language. This is a free and open source language for statistical computing that is used extensively for data analysis in many fields. Prior experience with the fundamentals of R programming is required.

Prerequisites

This course assumes that you have both a background in the core concepts of probability, statistics and inference as well as prior exposure to linear regression models. Completing the first three courses in the political science graduate methodology sequence should prepare you for the material in this class. However, there are no strict, specific course pre-requisites as many different disciplines and departments offer introductory statistics classes that cover the relevant material.

If you are unsure of whether you meet the requirements, skim/read through the first six chapters of *Regression and Other Stories*, one of the books being used by this course. You should find most of the concepts behind the material relatively familiar, aside from the references to Bayesian models (which will be covered in this course).

Please contact the instructor at (astrezhnev@uchicago.edu) if you are interested in enrolling but are unsure of the requirements.

Logistics

Lectures: Tuesdays/Thursdays from 9:30am-10:50am – Location: Harper Memorial 135

Disucssion Forum: We will use the Ed platform as a course discussion board. See the Canvas page for more details.

Course Materials: Lecture materials, assignments and section code will be posted on the course GitHub page at https://github.com/UChicago-pol-methods/plsc-40502-statistical-models/.

Readings will be listed on the syllabus. I will also post links to any non-textbook readings on the Modules page on Canvas.

Textbooks

The course will involve readings from a variety of different textbook chapters and published papers. The class will not require the purchase of any textbook as they are available online. However, you may wish to obtain a paper copy for your own personal use or reference.

The two primary textbooks from which many readings will be drawn are:

- Gelman, A., Hill, J., & Vehtari, A. (2020). *Regression and other stories.* Cambridge University Press. (An introduction to regression and multilevel modeling from an applied perspective) https://avehtari.github.io/ROS-Examples/index.html

- Gelman, A., Carlin, J., Stern H., Dunson, D., Vehtari A., & Rubin, D. (2013). *Bayesian Data Analysis.* 3rd Edition. Chapman and Hall/CRC. (A more advanced text on Bayesian modeling) http://www.stat.columbia.edu/~gelman/book/

Requirements

Students' final grades are based on three components:

- **Problem sets** (25% of the course grade). Students will complete a total of three problem sets throughout the quarter. Problem sets will primarily cover topics from the lecture and section for that week and the previous week.

  The goal of the problem sets is to encourage exploration of the material and to provide you with a clear and credible means of assessing your understanding and progress through the course.

  Problem sets will be graded on a (+/✓/-) scale with a + awarded for complete and near-perfect work, a ✓awarded for generally good work with clear effort shown but with some errors, and a - awarded for significantly incomplete work with major conceptual errors and little effort shown.

  - *Collaboration policy*: I strongly encourage collaboration between students on the problem sets and highly recommend that students discuss problems with each other either in person or via Ed. However, each student is expected to submit their own write-up of the answers and any relevant code.
  - *Office hours and online discussion*: Students should feel free to discuss any questions about the problem sets with me during class and during office hours. I also strongly encourage students to post questions about both the problem sets and the readings on the course Ed board and respond to other students' questions. Responding to other students' questions will contribute to your participation grade.

  - *Submission guidelines*: Problem sets will be distributed as PDF and Rmarkdown files (.Rmd). You should submit your answers and any relevant R code in the same format: including an Rmarkdown file (.Rmd extension) and a corresponding rendered .pdf file as your submission. Rmarkdown combines the text formatting syntax of Markdown markup language with the ability to embed and execute chunks of R code directly into a text document. This allows you to present your code, graphical output, and discussion/write-up all in the same document. I highly recommend that you edit the distributed Rmarkdown assignment file for each problem set directly to make organization easier.

- **Take-home midterm and final** (30% and 35% of the course grade respectively). The take-home midterm and final exams will have the same format and structure as the problem sets but with one key difference. You are **not** permitted to collaborate with other students or any other individual on the exams. I will answer any clarifying questions on the Ed discussion board, but will not answer substantive questions.

- **Participation** (10% of the course grade). I expect students to take an active role in learning in lecture. Engagement with the teaching staff by asking and answering questions will contribute to this grade as will interaction on the Ed discussion board.

Computing

This course will use the R programming language. This is a free and open source programming language that is available for nearly all computing platforms. You should download and install it from http://www.r-project.org. Unless you have strong preferences for a specific coding environment, I also highly recommend that you use the free RStudio Desktop Integrated Development Environment (IDE) which you can download from https://rstudio.com/products/rstudio/download/#download. In addition to being a great and simple to use environment for editing code, RStudio makes it very easy to write and compile Rmarkdown documents: the format in which problem sets will be distributed. In addition to base R, we will be frequently using data management and processing tools found in the tidyverse set of packages along with basic graphics and visualization using ggplot2.

The course will also introduce the Stan language and software for specifying and estimating Bayesian models. Stan is written in C but has bindings for a variety of programming languages. We will use two interfaces for Stan in R: RStan and brms.

Policy on Generative Large Language Models

The rapid growth in both the capabilities and the accessibility of generative large language models (LLMs) such as the GPT series, PaLM, LLaMa, etc… has introduced some novel challenges to the classroom. On the one hand, generative text models can be used as a tool to improve the quality of students' writing. On the other hand, they can be readily used to represent another's work as one's own – that is, to commit plagiarism. Additionally, LLMs may appear to be useful for some tasks – such as summarizing a set of texts or finding new sources on a particular topic – when in fact the outputs are arguably sub-optimal relative to conventional research methods.

My view in short: Large language models are marvels of **engineering**. You should use them for **engineering** tasks, but the task of research is not purely engineering and LLMs are much less effective for the task of doing **science**.

By "engineering," I mean the the iterative task of solving a problem by brainstorming potential solutions, implementing those solutions, and then subsequently *evaluating* the solutions with respect to some clearly defined criteria. The key components here are both the existence of a well-defined problem and the ability to assess whether the proposed solutions are effective.

Currently, the most obvious and effective use-case for large language models is in coding. I am perfectly happy for you to experiment with using LLMs in debugging code. The interactivity is great for beginning programmers who may have an idea of what they want their code to do, but are unfamiliar with the syntax of a particular language. Likewise, it's an incredibly valuable tool for experienced programmers who want to quickly generate some prototype code that is customized to their particular problem.

Why is programming an ideal use case? Programming is fundamentally an engineering task. There is a clearly defined problem that a programmer needs to solve via code and there is a straightforward way to evaluate whether a block of code works. As a result, mistakes are easy to catch – if the code throws an error, something needs to be changed. There is always a human in the loop who is capable of evaluating the output.

Outside of coding, I do not think LLM outputs are too useful, especially for generating text that is to be submitted without further refinement. In general, you should be cautious about any LLM outputs that you are not able to verify or evaluate yourself.

Irrespective of whether LLM outputs are "good" or not, it is absolutely clear that presenting LLM-generated output as one's own ideas is clearly plagiarism and will be treated as such. This does not rule out all uses of LLM-generated text, but it does rule out most. One use that I would consider acceptable is cleaning up original text that you have written to eliminate grammar mistakes or to rephrase the text to have a clearer style. We already accept the use of spellcheckers and thesauruses that are embedded in most word processors and I don't see this use case as substantively different as long as your original writing is the input. It is important, however, that you are able to evaluate the output and determine that it is conveying exactly what you want to say in exactly the way that you want to say it, just as you would when using any other writing tool.

Beyond this particular use, **submitting LLM-generated text as a substitute for your own thinking is not permitted in this class and will be considered plagiarism**. This includes prompting an LLM to compose all or part of your writing and submitting that output either verbatim or with some editing. This policy also applies to generating posts on the Ed discussion board.

In general, I do not think that presently there are too many good uses for LLMs for the particular tasks that you will be doing in this class. Although these models can be utilized for things like brainstorming, summarizing text, and search - acting as something of a personalized tutor - and the quality of the model outputs does appear to be steadily growing, I think that you will find significant value to working through the course material directly and asking questions to the teaching staff and to your colleagues in the class.

Schedule

A schedule of topics and readings is provided below. Each week will cover a single topic or group of topics. You should treat the readings as a reference and as a more detailed exposition of the topics discussed in lecture. Consult the readings when you want to know more or want a slightly different approach to explaining a particular topic.

Week 1: Introduction to Likelihood Inference (January 7)

- What are statistical models good for?
- What is a "parametric" model?
- The likelihood function
- Maximum likelihood estimation

Readings

- **Review**: “Regression and Other Stories” - Chapters 1-7

Problem Set 1 Assigned January 7, Due January 20

Week 2: Generalized Linear Models (January 14)

- Properties of maximum likelihood estimators
- Binary outcome models, Event count models, Duration models

Readings

- “Regression and Other Stories“ - Chapters 13-14
- Box-Steffensmeier, J. M., & Jones, B. S. (1997). Time is of the essence: Event history models in political science. *American Journal of Political Science*, 1414-1461.
- Wooldridge, J. M. (1999). Chapter 8: “Quasi-likelihood methods for count data.” In *Handbook of applied econometrics*, 2, 35-406.

Week 3: Bayesian Inference (January 21)

- Principles of posterior inference
- How to write a bayesian model
- Quantities of interest: Posterior Mode, Posterior Mean, Credible Intervals
- Estimation and inference via Markov Chain Monte Carlo

Readings

- “Regression and Other Stories“: Chapter 9
- “Bayesian Data Analysis“ Chapters 10-11

Problem Set 2 Assigned January 21, Due February 3

Week 4: Multilevel regression models (January 28)

- “Hierarchical” regression models – random slopes/random intercept models
- Estimation via MCMC in `Stan`
- Interpreting and analyzing results

Readings

- Park, David K., Andrew Gelman, and Joseph Bafumi. ”Bayesian multilevel estimation with poststratification: State-level estimates from national polls.” Political Analysis 12.4 (2004): 375-385.
- “Regression and Other Stories“: Chapter 9, 11, Appendix A
- “Bayesian Data Analysis“ Chapter 15

Week 5: Working with survey data (February 4)

- How to approach population inference from non-probability samples: constructing and using weights

Readings

- Caughey, D., Berinsky, A. J., Chatfield, S., Hartman, E., Schickler, E., & Sekhon, J. S. (2020). Target estimation and adjustment weighting for survey nonresponse and sampling bias. Cambridge University Press.
- Hanretty, Chris. ”An introduction to multilevel regression and post-stratification for estimating constituency opinion.” Political Studies Review 18.4 (2020): 630-645.
- Park, David K., Andrew Gelman, and Joseph Bafumi. ”Bayesian multilevel estimation with poststratification: State-level estimates from national polls.” Political Analysis 12.4 (2004): 375-385.

Midterm Exam: Assigned February 4, Due February 10

Week 6: Mixture Models and the EM Algorithm (February 11)

- Exploratory data analysis and clustering models
- MLE and MAP estimation via the “Expectation-Maximization” algorithm

Readings

- Imai, Kosuke, and Dustin Tingley. ”A statistical method for empirical testing of competing theories.” American Journal of Political Science 56.1 (2012): 218-236.
- McLachlan, Geoffrey J., Sharon X. Lee, and Suren I. Rathnayake. ”Finite mixture models.” Annual review of statistics and its application 6 (2019): 355-378.
- ”Bayesian Data Analysis” Chapters 13, 22

Week 7: Item Response Theory and Ideal Point Models (February 18)

- Latent variable models from a bayesian perspective
- “Ideal point” models for voting
- Extensions to models of networks

Readings

- Clinton, Joshua, Simon Jackman, and Douglas Rivers. ”The statistical analysis of roll call data.” American Political Science Review 98, no. 2 (2004): 355-370.
- Treier, Shawn, and Simon Jackman. ”Democracy as a latent variable.” American Journal of Political Science 52, no. 1 (2008): 201-217.
- Martin, Andrew D., and Kevin M. Quinn. ”Dynamic ideal point estimation via Markov chain Monte Carlo for the US Supreme Court, 1953–1999.” Political analysis 10, no. 2 (2002): 134-153.
- Burkner, Paul-Christian. ”Bayesian item response modeling in R with brms and Stan.” arXiv preprint arXiv:1905.09501 (2019).

Problem Set 3 Assigned February 18, Due March 3

Week 8: Regularization and Model Selection (February 25)

- Variable selection and penalized regression (Ridge, LASSO)
- Cross-fitting and out-of-sample validation

Readings

- Chapter 6: James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. An introduction to statistical learning. Vol. 112. New York: Springer, 2013.
- Stanescu, Diana, Erik Wang, and Soichiro Yamauchi. ”Using LASSO to assist imputation and predict child well-being.” Socius 5 (2019): 2378023118814623.

Week 9: TBA (March 4)

Final Exam: Assigned March 4, Due March 14

Assignment Schedule

- Problem Set 1: Assigned January 7, Due January 20
- Problem Set 2: Assigned January 21, Due February 3
- **Midterm Exam**: Assigned February 4, Due February 10
- Problem Set 3: Assigned February 18, Due March 3
- **Final Exam**: Assigned March 4, Due March 14

Acknowledgments

This course is indebted to the many wonderful and generous scholars who have developed causal inference curricula in political science departments throughout the world and who have made their course materials available to the public. This course in particular has been heavily inspired by Gov 2001 and Gov 2003 at Harvard University as well as Quant III at MIT. In particular, I thank Matthew Blackwell, Brandon Stewart, Erin Hartman, Molly Roberts, Kosuke Imai, Teppei Yamamoto, Jens Hainmueller, Adam Glynn, Gary King, Justin Grimmer, and In Song Kim whose lecture notes and syllabi have been immensely valuable in the creation of this course.