MATH 50 (2023)

INSTRUCTOR: ETHAN LEVIEN

Contents

About the course 1
Course policies 1
Resources 2
Accessibility Needs 3

About the course

Prerequisites: There are many other paths to prepare you for this course. The import thing is that you have some exposure to probability theory and are comfortable coding.

Course objectives: This is an introductory/intermediate statistics class with an emphasis on using simulation to explore statistical models. You will learn how to build, fit and make predictions with regression models. Such models form the basis of many widely used data analysis techniques, including machine learning algorithms. This is an applied course, and we won't spend much time deriving equations or proving theorems (at least compared to MATH 40 and 70). Instead, we will learn by playing around with real and simulated data. We will challenge the underlying assumptions of a method and see when things "break". The goal is to help you develop an intuition about statistical inference, which can be generalized to real world settings where theorems and analytical formulas are not applicable.

Here is a list of some things I expect you to be able to do by the end of the course:

- Work with tabular data in python.
- Visualize data in python.
- Understand some basic probability models and theorems, such as: Binomial distribution, Normal distribution CLT, LLN.
- Translate probability models into python code and perform Monte Carlo simulations.
- Understand the concept of conditioning in probability.
- Interpret paramaters in regression models.
- Fit and perform hypothesis tests for regression models.
- Design and evaluate regression models based on knowledge of real word data.
- Understand what are and are not appropriate applications of regression models.

Structure of class time: Class time will involve a mixture of lecture and coding. On a typical day, I will lecture for the first 30-45 minutes, then there will be a 15-20 minute "problem solving" session during which you can work in groups on exercises (and take a break) before I resume lecture of the remainder of the class. The lecture component of class will involve both chalkboard work and live coding demonstrations. During the problem solving sessions and live coding exercises I encourage you to use a computer and follow along. However, during the chalkboard portion of the lecture the use of computers is forbidden unless you have been authorized by me before hand (for example, due to accommodations).

Note on coding: All coding for the course will be done in Python. Some advantages of python (over R) are that: (1) We can easily run everything directly in the browser using google colab notebooks, so there is no need to download anything on your machine. (2) Python is widely used in data science and machine learning, both in academia and in industry. (3) While some very basic things are more difficult to implement, I think you'll find it's easier to generalize to more advanced methods. (4) I know python and I've basically never coded in R.

My availability: I will hold office hours

- Tuesday 1:00pm-2:00pm
- Friday 3:30pm-4:30pm (xhours, if I'm not using them to lecture)

I will generally be available to answer questions on slack (please use slack over email for course related matters) throughout the week. I will occasionally answer question on the weekend, but there is no guarantee.

Course policies

Attendance: The course meets twice a week and attendance is mandatory.

Exams: There will be an (in class) midterm quiz and a final. The final will be a take-home and involve working with some data. I will provide more details in class.

Exercises: Your "homework" is to submit solutions to a set of exercises. I plan to incorporate problem solving sessions into the lectures, giving you more time to discuss problems with myself and your peers. You will submit you solutions to canvas (approximately) every week before I release the solutions. Then, the following week you will self-evaluate (i.e. grade) you solutions and submit the evaluation. You should use the following point scale, which I will elaborate on in class.

- 0 - no work was done, or barely any effort was made.
- 1 - You put down partial work, but didn't put much effort in and didn't reach out if you needed help.
- 2 - You put in effort, but didn't get the problem exactly correct. You reached out at least once if you needed help.
- 3 - You got the problem correct, or made a very significant effort, including attending office hours and asking questions on slack if needed.

The # of points you get for each self-evaluations are the score you've given yourself plus an additional point for providing an explanation of what you did wrong. The graders will review your self-evaluations.

Guidelines for turning in exercises: You must adhere to the following guidelines when turning in exercises

- They should be turned in in PDF form on canvas. You can produce this PDF directly from your Colab notebook by saving as a PDF or by copying screenshots of your code into a word document and exporting as a PDF. There are many other ways as well.
- All code and figures should be in the PDF.
- Exercises should be in order and clearly labeled.

Final grades See canvas for details on how your final grade will be computed.

Use of Large Language models (LLM) such as ChatGPT: I STRONGLY encourage you to use LLM to assist with exercises and the final project. However, it is important that you use it in a way that supports learning the the material and not as a crutch. Therefore, usage of ChatGPT and other AI programs is subject to the following guidelines.

(1) You can use LLM to help write code, proofread your writing, and answer conceptual questions.

(2) In some instances I may specifically ask you to use ChatGPT. In this case the above constraints may not apply.

(3) You may NOT use it to produce entire written answers.

(4) You may not copy and paste the exercises into ChatGPT prompt. In-fact, I have tested many of the problems and found this will give nonsensical or incorrect answers (some exceptions are inevitable of course).

(5) If you use LLM to answer a homework, question you must include an explanation of how you used it. You do not need to share the entire chat transcript, but you should summarize the arc of the conversion and key prompts you used in a sentence or two.

(6) You may only use LLM programs which are publicly available or accessible to ALL Dartmouth students free of charge.

Here are some examples of **acceptable** LLM usage:

- "How do I reformat a pandas dataframe to that the columns and rows are switched?"
- "Given numpy arrays x, y and z how do I make a scatter plot where x and y are the coordinates and z is the color of the point?"
- "In class my professor said [insert something I said]. I'm having trouble understanding this statement, can you help me?"
- "Proofread this paragraph from my project proposal [insert text from project proposal]"
- "When I run this code [insert your code] I get the following error [insert error], can you help me debug it?"

[Note: I reserve the right to change the guidelines above at any point during the term.]

Resources

Textbooks: The course will be self-contained in the python notebooks and class notes. However, I will often reference the following textbooks for required readings and have borrowed some of the exercises from these books. They are available for free as pdfs.

- Introduction to statistical learning (Python version): This is a textbook on statistics and machine learning which covers regression. It is helpful to get a different perspective. The approach I take is much more probabilistic. In particular, I prefer to make it more explicit how the models are grounded in underlying probability distributions.
- Probability and statistics – the science of uncertainty (second addition): This is a textbook on probability theory which is far more technical than the level of this course, however it contains many useful examples and exercises which are appropriate for this math 50.

Software: All coding will be done using python in colab notebooks. Within python there are a number of packages we will use throughout the course, including:

- numpy for working with arrays, linear algebra and generating random numbers.
- pandas for working with tabular data sets.
- statsmodels for classical statistics

Accessibility Needs

Students with disabilities who may need disability-related academic adjustments and services for this course are encouraged to see me privately as early in the term as possible. Students requiring disability- related academic adjustments and services must consult the Student Accessibility Services office (Carson Hall, Suite 125, 646-9900). Once SAS has authorized services, students must show the originally signed SAS Services and Consent Form and/or a letter on SAS letterhead to me. As a first step, if students have questions about whether they qualify to receive academic adjustments and services, they should contact the SAS office. All inquiries and discussions will remain confidential.