STATS 507: Modern Data Analysis

Winter 2020, 3 Credits

Description

STATS 507 surveys the software tools that are currently popular among data scientists and machine learning practitioners in academia and industry. The course begins with an accelerated introduction to programming in Python. Next, we focus on Python's scientific computing stack: numpy, scipy, pandas, and scikit-learn. We also cover regular expressions, relational databases, git, and the UNIX/Linux command line. The final part of the course is an introduction to deep learning using PyTorch.

Prerequisites

All students should have some background in programming, preferably in Python.

Primary instructor

Prof. Jeffrey Regier, `regier@umich.edu`
Office hours: Wednesdays, 2 pm – 4 pm, 454 West Hall

Graduate student instructors

Zi Wang, `tlwangzi@umich.edu`
Office hours: Tuesdays, 4 pm – 5:30 pm, in the SLC Satellite (2165 Undergraduate Science Building)

Su I Iao, `iaosui@umich.edu`
Office hours: Thursdays, 2 pm – 3:30 pm, in the SLC Satellite (2165 Undergraduate Science Building)

Lectures

Fridays, 2:00 pm to 5:00 pm, 182 Weiser Hall

Textbook, Readings & Online Resources

The first part of the course is based on *Python for Everybody* by Charles Severance:
`https://www.py4e.com/book.php`

Students without prior Python experience may also benefit from this rapid introduction to Python:
`https://www.codecademy.com/learn/learn-python`

No textbook is required for the remainder of the course. Lecture slides will be posted on Canvas and articles to read may be assigned before lecture. The following resources may also be useful:

- *Python Machine Learning (3rd edition)* by Sebastian Raschka, `https://www.packtpub.com/data/python-machine-learning-third-edition`
- *Deep Learning* by Ian Goodfellow et al. `http://www.deeplearningbook.org/`
- *PyTorch tutorials*, `https://pytorch.org/tutorials/`

Course websites

Homework assignments and grades will be posted on Canvas:
`https://umich.instructure.com/courses/339333`

Questions about the homework or the course material should be asked through Piazza:
`https://piazza.com/umich/winter2020/stats507`
For questions about homework that cannot be asked without revealing a solution, please ask during office hours rather than on Piazza (and rather than by email).

For certain assignments we may also use

- GitHub classroom (`https://classroom.github.com/`),
- Kaggle InClass (`https://inclass.kaggle.com/`), and
- Google colab (`https://colab.research.google.com/`).

Grading

Final grades will be based on homework (45%), a midterm exam (25%), and a class project (30%). I expect the distribution of final grades will be similar to what it has been for previous offerings of STATS 507.

Homeworks & late days

Homework grades will be based on cumulative performance on approximately ten homework assignments. The exact number of homework assignments depends on factors such as lecture cancellations and how fast we cover material.

Each homework assignment is worth a given number of points. Assignments later in the semester tend to be worth more points than those earlier in the semester. Homework scores may be curved upwards if the class average is low.

Homework due dates are strict, and you may turn in work late only with the use of "late days", of which you have seven to use over the course of the semester. For each late day you spend, you extend the deadline of a homework by 24 hours. You may spend multiple late days per homework. Once you have turned in your homework you may not spend more late days to turn in your homework again. The purpose of this late day policy is to enable you to deal with unexpected circumstances (e.g., illness, family emergencies, job interviews) without having to come to me. Of course, if dire circumstances arise (e.g., long-term illness that causes you to miss multiple weeks of lecture), please speak with me as promptly as possible.

Due to the university grading schedule, you may not use late days to extend any deadline beyond the last day of winter term classes: Tuesday, April 21, 2020.

Midterm exam

During the first half of the course, students are expected to learn Python well enough that they can efficiently solve basic programming problems. Students with these skills are well prepared for technical interviews. Also, they have a solid enough grasp of Python programming that they can focus on the more high-level and creative aspects of programming for the remainder of the course.

The midterm exam will test that students have acquired these skills. Students will be provided with ample practice problems ahead of time (in addition to homework problems). Practice problems may be found at `https://hackerrank.com` and `https://codechef.com` too. We will work on some of these practice problems together during class. Though the exam will consist primarily coding questions, there may also be some short answer / conceptual questions based on the lectures.

The exam will take place during lecture on March 13, 2020. There are no alternative exam dates. No make up exams will be given. You must bring a laptop computer to the midterm exam. If you do not have access to a laptop computer that you can bring, please let the instructor and/or the GSIs know right away.

Class project

Students will work in groups of two or three on data analysis projects. Each group will select one or more large datasets that interest them, and formulate research questions that can potentially be answered using software tools from class.

The project has three aims: 1) to provide students with in depth experience with a real data analysis problem, 2) to familiarize students with collaborative work, which is widespread in industry, and 3) for students to begin developing a portfolio that showcases their skills, suitable for showing potential employers.

The project is graded based on four deliverables: a proposal, a preliminary report, an oral presentation during the last two weeks of class, and a final report. Scores are based on how thoroughly you analyze your dataset(s), how clearly you communicate your findings, and on the technical skills your analysis demonstrates.

Ethics and class policies

Academic misconduct includes such actions as copying code from the web or from your fellow students, providing code to your fellow students, looking up solutions online, turning in assignments from other classes or previous iterations of this course, and hiring others to complete your work for you. You are welcome to discuss homework with your classmates, but the work that you turn in must be yours and yours alone, and you must disclose the names of those you spoke with in your homework. From the LSA Community Standards of Academic Integrity:

> Academic dishonesty may be understood as any action or attempted action that may result in creating an unfair academic advantage for oneself or an unfair academic advantage or disadvantage for any other member or members of the academic community. Conduct, without regard to motive, that violates the academic integrity and ethical standards of the College community cannot be tolerated.

See `https://lsa.umich.edu/lsa/academics/academic-integrity.html` for more information. Violations of these or other university ethical standards surrounding academic honesty will be met with serious consequences and disciplinary action. Cheating on an assignment will result in a 0 for that assignment and the incident will be reported to the appropriate office. Additionally, a full letter grade may be deducted from the student's final grade in the course.

Accommodations for students with disabilities

If you need an accommodation for a disability, please let me know as promptly as possible. Some aspects of this course may be modified to better suit you. As soon as you make me aware of your needs, we can work with the Services for Students with Disabilities (SSD) office to determine appropriate academic accommodations. SSD (734-763-3000; `http://ssd.umich.edu`) typically recommends accommodations through a Verified Individualized Services and Accommodations (VISA) form. Any information you provide SSD is confidential.