Syllabus
STAT 606: Computing in Data Science & Statistics
Spring 2024, 3 Credits

Description
A survey of some of the tools and frameworks that are currently popular among data scientists and statisticians working in both academia and industry. The focus will be on complementing the tools that are already familiar from previous courses on R. The course will begin with an accelerated introduction to the Python programming language and brief introductions to object-oriented and functional programming. We will then cover some of the scientific computing platforms available in Python, including numpy, scipy and scikit-learn, as well as visualization using matplotlib. We will then turn to discussing collecting data from the web both by scraping and using APIs. The course will conclude with a brief survey of distributed computing platforms, focusing on Hadoop and Google Cloud Platform.

Prerequisites
Declared in Statistics MS. There are no strict course prerequisites, but I assume your background is equivalent to STAT605 or similar. In practice, that means being comfortable using R, familiarity with the UNIX/Linux command line, and prior experience with distributed computing resources (e.g., using a scheduler on a cluster). Students with no prior programming experience are discouraged from enrolling.

Instructor
Keith Levin, `kdlevin@wisc.edu`
Office: Medical Sciences Center 6170 **Office hours:** To be determined

Meetings and Key Dates
Lecture: Mondays and Wednesdays, 9:30am-10:45am in 120 Ingraham Hall

- First lecture: Wednesday, January 24th, 2024
- Last lecture: Wednesday, May 1st, 2024
- Last homework due: Friday, May 10th 2024 by 11:59 p.m.

Course Learning Outcomes
By the end of this course, you will be able to:

- Understand and apply the basics of the Python programming language and basic programming patterns in both the object-oriented and functional programming frameworks.
- Collect and clean data from a variety of data sources including markup languages from the web, databases, and APIs
- Analyze and summarize data using `matplotlib`, `numpy`, `scipy` and `scikit-learn`.
- Use Google Cloud Platform and similar cloud computing resources to run large-scale computations in a distributed environment using `Hadoop`, `mrjob` and `PySpark`.
- Build and fit statistical models on large datasets in the Google TensorFlow and Keras frameworks.

Course Topics

- **Introduction to Python**. Variables and data types. Programming patterns. Classes and objects. Functional programming.
- **Visualization with** `matplotlib`. Basic plotting.
- **Scientific computing in Python**. Introduction to `numpy`, `scipy` and `scikit-learn`.
- **Processing Structured Data**. Regular expressions. Markup languages. Databases and SQL.

- **Retrieving Data with APIs**. HTTP request methods. Installing and using APIs.
- **Big data and distributed processing**. Basics of parallel/cloud computing. The MapReduce framework. Hadoop and Spark.
- **Specifying and training models with** `TensorFlow`. Basics of Google TensorFlow. Function graphs. Symbolic differentiation.

Textbook, Readings & Online Resources
There is no physical textbook required for this course. In the first half of the course, we will make frequent reference to Allen B. Downey's *Think Python*, available at `https://greenteapress.com/wp/think-python-2e/` and to Charles Severance's *Python for Informatics*, available at `https://www.py4e.com/book`. Other required readings will be made available as we cover relevant material, and supplemental readings will be suggested for those who are interested in learning more.

All resources will be made available on the course web page, `https://pages.stat.wisc.edu/~kdlevin/teaching/Spring2024/STAT606/index.html`. and on the course Canvas page. Please contact the instructor if any resources are missing from either of these websites. The instructor will make an effort to post slides and demo code a few days ahead of time so that they are available before lecture. It is recommended, though not required, that students complete assigned readings before lecture.

Homeworks & Late Days
Homework assignments will be submitted via Canvas, as `.zip` files, with the name `NetID_hwX.zip`, where `NetID` is your NetID and `X` is your homework number. Detailed instructions on homework submission can be found at `https://pages.stat.wisc.edu/~kdlevin/teaching/Spring2024/STAT606/hw_instructions.html`
Student collaboration. Students are permitted to discuss homeworks with one another, but students must complete assignments on their own, and must disclose in their homework the names of those with whom they collaborated. Details are available in the instructions linked above.
Late days. Homework due dates are strict, and you may turn in work late only with the use of "late days", of which you have seven (7) to use over the course of the semester. For each late day you spend, you may extend the deadline of a homework by up to 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 after the deadline (you may, of course, turn in multiple versions of your homework assignment through Canvas prior to the deadline). There is no need to notify the instructor that you wish to use a late day. Simply turn in your homework late, and your late days will be deducted upon grading. Please note that once you have run out of late days, homeworks turned in late will not receive credit.
The purpose of this late day policy is to give you a way to deal with unexpected circumstances (e.g., illness, family emergencies, job interviews) without having to contact the instructor. 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. **Note:** owing to the university grading schedule, you may not use late days to extend any deadline beyond the due date of the last homework.
Contesting scores. Students may contest a score on an assignment up to two (2) weeks from when scores are released, after which scores may not be changed. To comply with the registrar's grading schedule, students may not contest any grade more than one (1) week past the scoring of the final homework.

Grading
There are no exams in this course. Final grades will be based on cumulative performance on a set of approximately thirteen homeworks. Note that the exact number of homework assignments is subject to change depending factors such as lecture cancellations and the speed with which we cover material. Each homework assignment is worth a given number of points, and final grades will be based on a percentage out of the total possible points through the whole semester. Assignments later in the semester will be worth more points, on average, than those earlier in the semester. Grades will be assigned based on the scheme

outlined in the table below. The instructor reserves the right to relax this grading scheme in the event of skewed class performance, but pledges not to curve grades downward. That is, if you have an AB under this grading scheme, your curved grade will not be worse than an AB.

$\geq 93\%$	A
88% to 93%	AB
83% to 88%	B
78% to 83%	BC
70% to 78%	C
60% to 70%	D
$< 60\%$	F

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 homeworks with your classmates, but the work that you turn in must be yours and yours alone, and you must disclose in your homework the names of those with whom you collaborated. Use of AI or other software outside of manners explicitly permitted in an assignment is not permitted.

From the Office of Student Conduct and Community Standards:

> [A]cademic misconduct is behavior that negatively impacts the integrity of the institution. Cheating, fabrication, plagiarism, unauthorized collaboration, and helping others commit these previously listed acts are examples of misconduct which may result in disciplinary action.

See `https://conduct.students.wisc.edu/academic-misconduct/` for more information.

Violations of these or other university ethical standards surrounding academic honesty will be met with serious consequences and disciplinary action. At a minimum, cheating on an assignment will result in a 0 for that assignment and the incident will be reported to the appropriate office. At the instructor's discretion, depending on the circumstances, an additional full letter grade may be deducted from the student's final grade in the course.

Accommodations for Students with Disabilities

The University of Wisconsin-Madison supports the right of all enrolled students to a full and equal educational opportunity. The Americans with Disabilities Act (ADA), Wisconsin State Statute (36.12), and UW-Madison policy (Faculty Document 1071) require that students with disabilities be reasonably accommodated in instruction and campus life. Reasonable accommodations for students with disabilities is a shared faculty and student responsibility. Students are expected to inform faculty [me] of their need for instructional accommodations by the end of the third week of the semester, or as soon as possible after a disability has been incurred or recognized. Faculty [I], will work either directly with the student [you] or in coordination with the McBurney Center to identify and provide reasonable instructional accommodations. Disability information, including instructional accommodations as part of a student's educational record, is confidential and protected under FERPA.