

# Python and Other Technologies for Data Science

Stats 21

Miles Chen, PhD

Department of Statistics

Week 1 Monday



## Section 1

### Highlights from the Syllabus

# Welcome!

- My name is Miles Chen
- [mileschen@stat.ucla.edu](mailto:mileschen@stat.ucla.edu)
- <http://ccle.ucla.edu>
- Office Hours: Wednesdays 4-4:50 PM, Fridays Noon - 12:50PM, Saturdays 10AM-11AM, and by appointment
- You can call me: Miles, Professor Chen, or Dr. Chen, whatever name you are comfortable with. However, please don't start your emails without a name. Also, you'll probably get a faster response if you message me via Campuswire.

# Grade Breakdown

- 10% Lecture Viewing Quizzes
- 20% DataCamp Homework
- 36% Submitted Homework
- 30% Final Exam
- 4% Campuswire Participation

Letter grades are assigned on a straight scale as follows:

- 59.9 and below: F
- 60.0 - 69.9: D
- 70.0 - 76.9: C, 77.0 - 79.9: C+
- 80.0 - 82.9: B-, 83.0 - 86.9: B, 87.0 - 89.9: B+
- 90.0 - 92.9: A-, 93.0 and up A, top 5% of students: A+

I do not curve grades. Please do not ask me to round your grade.

# Lecture Viewing Quizzes / Attendance

I will not take traditional attendance but students are required to watch the lectures.

Viewing of lectures is enforced through the lecture viewing quiz. Quiz answers are given audibly during lecture. Answers to the quiz will not be found in lecture notes. Sharing quiz answers with another student is considered facilitating academic dishonesty.

If you can't attend live, you can watch the recorded video.

Viewing quizzes for each lecture will close before the next session begins. One quiz will be dropped. Do not ask me to open up the quiz for you if you forget to take the quiz or make a mistake when selecting your quiz answers.

# DataCamp Homework

Students enrolled in the course should have received an invitation from DataCamp. Students will have access to DataCamp's premium content through September 18, 2021. (Auditors will not receive an invitation.)

I will assign homework from DataCamp. Please check DataCamp regularly to see if homework has been assigned.

For many of the assignments, I will assign only a chapter or two from a course. I do this to make the workload manageable. If you have the time, I recommend completing the other chapters from the course.

# Regular Homework Assignments

I will assign “regular” homework that needs to be submitted as well.

Homework will be posted as an Jupyter Notebook (.ipynb file). Students will complete their exercises in the notebook, save it as a PDF and submit the PDF to Gradescope. You will need to have LaTeX installed on your computer to save the notebook as a PDF.

- Please read the late policy in the syllabus.
- A 72-hour extension granted if documentation is submitted with the homework. **No need to contact the professor if you will include documentation with the homework submission.**
- If you need an even longer extension, please visit professor in office hours.



## Personal note about office hours:

When you come to office hours, please introduce yourself. Say “Hi Miles, I’m Joe Bruin.” Do this **every** time you visit me until I start calling you by your name.

I **like** when students come to office hours with questions about material. I love to explain things and to help students understand.

I **like** when students come to office hours to tell me more about themselves and to seek counsel about classes to take or next steps. I am also happy to help students who face difficult circumstances and may need some exceptions to be made. Please do not hesitate to visit office hours if you need an extension longer than 72 hours, or if you need to delay an exam.

I am happy to correct mistakes that may have been made. I do not enjoy getting in arguments with students over homework points. I do not like arguing whether a particular mistake on an exam should be a 5 or 10 point deduction.

## Campuswire for questions. Office hours for requests.

Post your questions on Campuswire. You will likely get a quick response from classmates.

If it is a question you don't want public, you can DM me on Campuswire. You are likely to get a faster response via Campuswire than email. Keep your messages short.

If you need me to do something - e.g. grant an extension, schedule a make up, change a grade, etc. - please come to office hours.

# What we'll cover

- ① Week 1: Git, GitHub, conda, Jupyter
- ② Week 2: Python Basics: variables, expressions, statements
- ③ Week 3: Python Functions: encapsulation, recursion, return values, iteration
- ④ Week 4: Python Data types: strings, lists, dictionaries, tuples
- ⑤ Week 5: Python OOP: Classes and objects, methods, inheritance
- ⑥ Week 6: Pythonic code: List comprehensions, kwargs; Numpy
- ⑦ Week 7: Pandas: Importing, reshaping, and cleaning data
- ⑧ Week 8: Pandas: Wrangling, and Aggregation
- ⑨ Week 9: Data Visualization
- ⑩ Week 10: Fitting models with scikit-learn

## Section 2

# Academic Integrity and Plagiarism

# Let's talk about Plagiarism

Some truths:

- There is a lot of high quality code that does exactly what you need available out on the Internet. Some of it is available in ready-to-install packages and some are available as solutions on places like stackexchange and github.
- If the goal is to accomplish a task, you should use the readily available packages or code solutions out there.
- However, the goal of this class is not to accomplish some task. The goal is to help you learn how to write code.

# No Pain, No Gain

Think of the gym. The goal of lifting weights at the gym is not to lift weights. Lifting weights is a means to the real goal of gaining strength.

“No Pain, No Gain”: if your weight training does not result in some muscle soreness, you probably did not exert enough effort to expect muscle gain. Experiencing muscle soreness is a sign that your muscles will go through repairs and get stronger.

Your brain is similar: if your brain does not struggle when writing code, then it has no reason to create additional neuron connections that will improve your abilities as a coder. On the other hand, if your brain struggles with writing code, then your brain will try to create new connections between neurons so the next time will not be as hard. And thus you become a better coder.

# No Pain, No Gain

Plagiarizing code for a difficult assignment is like having a stronger person lift the weights that are too heavy for you.

This would be a good solution if the goal of lifting weights were to lift the weights. But it does not help achieve the goal of gaining strength.

Copying, pasting, and modifying a stronger programmer's code works if the goal is to accomplish a coding task. It does not help towards the goal of creating neuron connections in the brain that will make you a better coder.

# Course Goals

I believe students resort to plagiarism because they have confused the goals of the course.

Students who plagiarize believe the goal for them is to get good grades and avoid bad grades in the class. For these students, the goal of learning is secondary to the goal of getting the desired grade.

But this is wrong! The goal of the course is your learning.

I will admit, a major conflicting issue here is that I am not able to create individualized grading schemas that evaluate exactly how much each student learned over the course. All students are graded on the same criteria and evaluated on what they turn in for the assignments.

That said, I hope you can judge your performance in class based on what you learned and not your letter grade.



# My expectations

When you face a challenging homework assignment, I expect

- you work hard
- you will not seek out solutions online or view another (current or former) student's code
- if you are not able to complete everything required by the assignment by the deadline
  - ▶ you submit what you have and accept a grade that is less than 100%
  - ▶ you view this not as a failure of your coding abilities, but as a indication of areas for growth and improvement

I (and the statistics department) take issues of plagiarism seriously and will escalate cases to the Dean of Students. Full details regarding academic integrity are in the syllabus.

# Collaboration Policy

Read the course collaboration policy and be sure you understand it.

For all homework assignments, verbal collaboration but no code sharing.

You are allowed to collaborate verbally with other students but you are not allowed to look at or show someone else the code you are writing. This applies to discussions on Campuswire.

# You are encouraged to discuss code that is not part of an assignment!

This is a coding class! As long as the code is not part of a homework assignment, you can post and discuss code.

You can always post and discuss code that appears in lecture. You are encouraged to modify the examples that appear in lecture and discuss the effect of each change you make.

You can post and discuss code that is for the purpose of learning a particular concept or how a function works.

## Section 3

### Grades and Life

# Your grades do not define you

You are here at UCLA. One of the reasons you got into UCLA was because you had good grades in high school and/or at community college. While you are in school, a lot of your energy is poured into your classes and I can understand why grades feel so important. That said,

## **Your grades do not define you**

It feels good to get good grades. Grades do play a role in graduate school admissions. But they are not the most important thing in life. No one on their death bed looks back and says “I wish I got an A- instead of a B+ in that one college class.”

# Work - Life Balance

I like to split where we put our energies of life into three broad categories:

- Work
  - ▶ Jobs and internships
  - ▶ School and academics
  - ▶ Other professional obligations
- Relationships
  - ▶ Family
  - ▶ Friends
  - ▶ Romantic partner
  - ▶ Other social obligations
- Self
  - ▶ Care of physical health (food, sleep, exercise)
  - ▶ Care of mental health (sleep, play, entertainment)
  - ▶ Care of spiritual health (if you are spiritual/religious)

There are 24 hours in a day. It is not possible to give 100% to all categories

# Work - Life Balance

Work-Life balance is achieved by consciously choosing what is important to you and devoting your time and energies accordingly.

In general, the more you put in, the more you get out.

Satisfaction can be found by accepting the natural consequences of what you have chosen to deprioritize.

# Work - Life Balance

Let's say you are part of a group of friends. Let's say that one day you become involved with a romantic partner.

If you choose to invest all of your relationship hours into your romantic partner, you will likely develop a very strong relationship with your romantic partner. However, because you now invest much less into your original group of friends, those relationships will naturally become more distant. When you see distance forming, it can initially feel hostile. This is not (necessarily) the result of your friends being angry that you have a romantic partner but the natural consequence of having less time to spend with them.

As people, we have to make a choice about what is important to us.

When you accept the natural consequences of investing less time into something, you can reduce your own feelings of bitterness and jealousy.



# Work - Life Balance

In the corporate and professional world, people who devote a lot of energy into the goals of the company are rewarded. The company is not necessarily punishing people who choose to have families and a life outside of work.

From the company's perspective, who would they rather promote?

- the person who did everything asked of them and then continued to stay at work and did even more
- the person who did everything asked of them and then immediately left to spend time with their family/friends/romantic partner

You have to choose what is important to you. If climbing the ranks within the company is more important, then you will spend your time accordingly. If spending time with your family/friends/romantic partner is more important, then spend your time accordingly.

# Self care is important

You must not neglect taking care of your physical and mental health.

If you neglect care of self, you will likely operate at less than 100% efficiency and the time you invest in work/school/relationships will not be as productive.

Examples:

- You don't get enough sleep. A friend invites you out. You choose to accept your friend's invitation instead of sleep, but you are a bit 'out of it' and are a drag to hang around. Maybe it would have been better to decline your friend's invitation and get sleep.
- Exams are coming up. You choose to skip a meal and minimize sleep to study. You end up getting sick. Your performance on the exam suffers. Maybe it would have been better to eat properly, sleep well, and study a bit less.

# Self care is important

When I tell you that your physical and mental health is important, I'm encouraging you to choose to invest your time into activities like exercise, sleep, and relaxation that will boost your physical and mental health.

Sometimes this means choosing not to complete your homework to 100%.

The natural consequence of this is a homework grade that is less than 100%.

When you can readily embrace this natural consequence of prioritizing your own physical and mental health over your homework grade, you can enjoy the quarter with less bitterness, more joy and better health.

# Beware of “fruitless” entertainment

Entertainment and fun activities are important for your mental well being. It's important to have fun.

I love hanging out with people I like, watching TV, movies, sports, playing board games, video games, going on hikes, browsing the Internet, reading a book, listening to podcasts, etc.

Participating in an entertainment activity should be a break from work and should give you mental energy so you can return to your work in a good mood.

Some activities can have the opposite effect - they drain you. Some video games, apps, and social media sites are designed to be addictive - giving your brain immediate dopamine pleasure hits while you use them so you play round after round or continue scrolling forever (and keep coming back) . . . but after spending hours of doing the activity, you don't feel good about yourself.

Be mindful and selective about your entertainment activities.

## Section 4

### Dinner with Classmates

# Dinner with Classmates

Last quarter, I hosted a few “Zoom Dinner” meetings. I enjoyed them and received positive feedback for the events.

I plan on hosting a few of these again. I'll post more information on CCLE and Campuswire.

## Section 5

### Anaconda

The Python world is a bit messy. There's a bunch of different versions of Python and different packages that require certain other packages and if you have the wrong version, code can break.

Conda is tool for managing packages and environments.

Anaconda is collection of commonly used Python libraries along with Conda to facilitate installing and maintaining Python libraries and installations.

Included with Anaconda are:

- Jupyter Notebook and Jupyter Lab
- NumPy
- SciPy
- Matplotlib
- Pandas
- Scikit-Learn



# Download and Install Anaconda

<https://www.anaconda.com/products/individual>

## Section 6

### Shell Basics

# Opening Terminal on MacOS

Probably the quickest: Open spotlight with Command + space. Start typing “Terminal”. Terminal will appear as the top hit after you type the first few letters. Hit Enter to start.

Another method: You can open Launchpad from the dock. Click “Other”. Click “Terminal”.

I suggest adding the terminal to the dock.

# Opening Anaconda PowerShell on Windows

Use the Start Menu to find “Anaconda Powershell Prompt”. It is found under the Anaconda3 folder.

You can find it quickly by hitting the Windows Key and then immediately typing “anaconda pow” then enter

I suggest pinning the shortcut to the Start Menu.

If you start regular PowerShell instead of Anaconda Powershell, it will not have environmental variables set up correctly and you will not be able use conda or start Jupyter.

# Getting Help

Windows PowerShell

`help commandname`

Mac OS and git bash

`commandname --help`

Type q to exit help. Hit the space bar to scroll to the next page.

# Shell Basics: Navigation

`~` is your home directory.

`pwd` will tell you where you are currently located. (present working directory)

`cd` is the command to change your directory

Wherever you are, you can switch to your home directory with `cd ~`

# Shell Basics: Navigation

Directories are listed in a hierarchy. For example, you may decide to store content for this class in:

```
home/classes/stats21
```

Let's say this is your present working directory.

`cd homework` will change to the directory `homework` if it is available in your current directory. If there is a `homework` folder inside the `stats21` folder, it will take you to `home/classes/stats21/homework`

`cd ..` will take you to the parent directory. If you are currently in `stats21`, `cd ..` will take you to `home/classes`

`cd ../../` will take you two levels up.

Shell support tab completion. If you have the folder `homework` inside `stats21`, you can begin by typing `cd ho` and then hit TAB. Shell will try to complete what you are typing. If there are multiple items that start with `ho` then you can hit TAB multiple times until it finds the item you are looking for.

`ls` will list the contents of your current directory.

`mkdir name` will create a new directory called `name` inside your current working directory.

`clear` will clear the screen.



# conda

Conda is a package and environment manager. When you install Anaconda, conda will be a utility that will be installed. You can use conda from the terminal or Anaconda PowerShell.

Most of the packages you need will come pre-installed with Anaconda. However, you may need to install a new package in the future or update your packages.

Installation is done with

```
conda install packagename
```

Updating can be done with

```
conda update packagename
```

or you can update all of them with

```
conda update --all
```

If you are using Windows, I recommend starting Anaconda Powershell Prompt with Administrator rights before running `conda update --all`

# Jupyter Lab and Jupyter Notebook

I will discuss this further in the coming days. But if you just want to start playing with Python, you can launch Jupyter Lab or Jupyter Notebook.

```
jupyter lab
```

```
jupyter notebook
```