GEOG 4860: Python Programming for GIS

3 credits, Online Delivery

Instructor: Tyler Hatch

UTF: Sarah Robertson (https://usu.instructure.com/courses/675626/)
tyler.hatch@usu.edu

1. Course Description and Prerequisites

1.1. Course Description: This weekly assignment-based course is designed for students aiming to learn how to write basic scripts in Python, and specifically, how to automate geoprocessing tasks in ESRI’s ArcGIS software. We’ll use ESRI geoprocessing for all of the exercises during this course, which will require ArcGis Pro (and maybe ArcMap) for all students. The course will begin with an introduction to Python, basic syntax, and writing effective code. The remainder of the class is devoted to manipulation of vector data, attribute/tabular datasets, and dealing with raster data, followed by automated map/figure generation.

1.2. Prerequisites, Formal and Informal. This course requires a working knowledge of spatial science and ArcGIS Pro. That is, USU’s NR 6910 (GIS for Natural Resource Applications), or my permission and the approval of your academic advisor, are required. Please contact me if you’re unsure whether you meet these requirements. No prior knowledge of Python is assumed.

A high degree of computer literacy is required, and will be assumed in exercise instructions. This includes file manipulation (i.e., cut/copy/paste, directory creation), the use of web browsers, and the ability to install software. If you do not have administrative privileges on your computer (that is, if you’re not allowed to install software on your own), you’ll need to work closely with any IT administrators who manage installations to stay up to speed on this course.

Interest in natural resources spatial science is helpful. Students are required to have access to a computer which meets the system requirements to run the ArcGIS software and must have reliable internet access. Please see Esri’s webpage on system requirements before registering for the course: https://pro.arcgis.com/en/pro-app/latest/get-started/arcgis-pro-system-requirements.htm. Note that ArcGIS Pro will not run on an Apple OS or Linux/Unix.

The pace of the class is rigorous in that students need to complete one assignment per week of the semester, along with providing collaborative feedback in the course forum. This is not a self-paced course.

2. Course Objectives

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

- Automate tasks in ArcGIS using Python
- Create custom toolboxes and add-ins for ArcGIS
- Make simple maps using Python

IDEA Objectives

- Learning fundamental principles, generalizations, or theories
Developing specific skills, competencies, and points of view needed by professionals in the field most closely related to this course

3. Course Materials

3.1. Canvas: The course will use Canvas (https://online.usu.edu/) for course announcements, assignment delivery and submission, quizzes, discussions, grades, etc. It is the student's responsibility to log in to and utilize the Canvas system. Help using the Canvas system can be obtained by contacting USU Information Technologies at 797-HELP or going to http://it.usu.edu (http://it.usu.edu/).

3.2. Software: The ESRI education edition software package (a one year free trial of ArcGIS Pro) will be made available to all students enrolled in the class. All other software that will be used is free, and students will be provided directions and support in installing needed software packages.

3.3. Hardware: Please see https://pro.arcgis.com/en/pro-app/latest/get-started/arcgis-pro-system-requirements.htm (https://pro.arcgis.com/en/pro-app/latest/get-started/arcgis-pro-system-requirements.htm) for more information or to test your computer for compatibility with ArcGIS.

3.4. Internet access: A reliable high-speed internet connection is required for all students enrolled in the course. Your success in this course will be determined at least in part by the effectiveness of your computer and internet connection. Over the years, the students who are dealing with computer issues after the start of the semester fall behind and have a very difficult time catching up. If you suspect you will have limited internet access, you might consider putting off this course until a time when you will have full-time reliable internet access.

3.5. Required Textbook: There's no required text, but I'll often refer students to documentation and background materials that are freely available and which can help with the course assignments. I will not be providing weekly reading assignments from the suggested texts.

Suggested texts (not required):

Python for Kids, Jason R. Briggs. ISBN: 1593274076
https://www.amazon.com/Python-Kids-Playful-Introduction-Programming/dp/1593274076

Don't laugh. This is the book that got me into programming way back in middle school. The most important part of this class is that you understand the fundamentals really, really well. This book explains it in extremely simple terms, and makes learning Python fun and interactive! I promise that if you work though this book, you'll get a firm grasp on Python quicker than anyone else in the class.

This is a simple, accessible text that introduces writing python code to work with ArcGIS and is ESRI-centric. If I were to have required a textbook for class, this would be the one.

If you're interested in doing many of the same geoprocessing tasks that we'll do in class, except using completely free and open-source software, this is the book for you.

4. Course Structure and Grade Breakdown
Each week is built around a central theme. There will be three components to each week’s assignment. Below I’ve listed each of the components and the percentage of the final grade that they’ll comprise.

4.1. Jupyter Notebooks. These are interactive blocks of python code that can be viewed in ArcGIS Pro, and they’ll be provided for students to read through and run sample code (and in some cases, slightly modify). These notebooks will introduce the core concepts for each week. Completion of these notebooks will comprise 30% of the grade.

4.2. Problem Sets. This course will require students to write original python scripts each week to solve problems related to that week’s theme. There will be, on average, 3-5 problems assigned each week. Students will create their own Jupyter notebooks that contain the code they’ve written, and these will be turned in via Canvas. These problem sets will comprise 30% of the grade.

4.3. Instructor Feedback. This is only my second time teaching this course! I have full confidence I’ll be able to teach you well, but I want to make this best experience possible for you, your peers, and future students who take this class. Each week you’ll fill out a short quiz describing what could be improved, things you liked, thing’s you didn’t like, and other similar things. This is super helpful to me, so please do your best to complete these each week! These feedback surveys will comprise 10% of the grade.

4.4. Final Quiz. Despite the name, I don't want you to think of this as a test. This is a sort of checkpoint towards the end of the class, so I can check your comprehension on topics you should be comfortable with by that point. Don't stress this. My goal is for everyone to get 100% on this quiz, even if it means retaking it a few times. This will also be available over several days, and open book. This quiz will comprise 10% of your total grade.

4.5. Final Project. This project is a chance for you to show me and the world all that you've learned. Your topic can be anything that you want, so start thinking of ideas now. As long as it involves automating some GIS process in Python, that counts! This project will comprise 20% of your total grade.

4.6. Collaborative Troubleshooting (Questions and Answers). Developing code is rarely a solo endeavor, and is more often than not done within a collaborative group. Problems will arise, and you'll inevitably hit a roadblock with your scripts. I'll do my best to help with these, but your first line of defense will be one another. With this in mind, I've set up Canvas Discussions, where you’ll be able to post questions, provide feedback to other students, and get help from classmates. I’ll also be active on these boards. Because I understand that a few of you want to breeze through this class, and a few of you are super busy, these aren't required. However, if you post two questions, answers, novel coding ideas, or interesting approaches to solving problems, you'll get 5 points of extra credit each week. That adds up to an extra 50 points throughout the semester. In a class worth 1000 points, that's a decent difference.

5. Other Items

5.1. Workload: This course will be a good deal of work. It will, at times, be challenging, and will undoubtedly expose students to topics and skills they haven’t come across before. You should expect to spend an average of 5-10 hours per week on this course.

5.2. Course assessment: Assessment of the class will occur during week 15 utilizing the IDEA course assessment tool.

6. Grading Scale
7. Communication: what you can expect from me

This course is a lot of work, but I won't leave you high and dry. I'll be available via email to answer any questions or to clarify issues that arise throughout the semester. That being said, I can’t promise to act as on-call tech support for all issues that arise! It's much more effective if we all collaborate to solve issues. When you run into trouble, here's what to do, in this order:

1. **Google.** This will quickly become second nature. If you’re having a problem, the overwhelming odds are that someone else out there has had the exact same issue.
2. **Post your question on the Questions and Answers page for the course.** I’ll contribute to this regularly too, and my hope is that we’ll be much more effective in troubleshooting as a team than as individuals.
3. **Email me.** I’ll do my best to help you out in a timely manner, provided I know that you’ve tried other options to get to the bottom of the issue. For me, email is a faster mode of communication than Canvas.

*Note on asking questions:* if trouble should arise, make a habit of always providing as much information as possible on what's going wrong when you ask your question. The best practice is to just include your script, whether as a file or just copy/pasting the text, right in your question.

It’s MUCH, MUCH better to say “on line 37 of the script below, the error I’m getting is _________. Here’s what I’ve tried to fix this already, but to no avail” than it is to say “my script won’t work and is giving me an error.” Help us help you!

Students are expected to monitor Canvas announcements and emails in order to receive communications about the class in a timely manner. Canvas will be utilized for class-wide announcements. In addition to Canvas, I’ll use screen capture demonstration videos and other technologies to ensure that adequate interaction and assistance is available.

8. Policies

8.1. Late work:
Generally speaking, I am very generous with late work, as long as you let me know before hand. Please, if you have any issues, or you’re worried you won’t be able to finish, email me before the deadline, and I can guarantee we’ll work something out. Please don't wait until after the assignment is due.

Any work submitted 1 day past the due date will receive an automatic 5% reduction unless due to a pre-approved excused absence. 5% will be deducted for each subsequent day after that, down to a minimum of 50%. Partial credit can be received for work turned in late assuming the student has sought approval from the instructor. No work will be accepted more than two weeks after the original deadline regardless of the circumstances.

8.2. Cheating: This one's pretty simple - there's a zero tolerance policy on cheating, or plagiarism. That is, you will fail the course and I will report you to the office of student conduct. Do the assignments to learn, not to 'get them done'. It's okay, and encouraged, to solicit help from each other on coding issues; I want this to be a collaborative course. But don't blanket copy someone else's code and turn it in as your own. If you don't understand something, ask. If you are running out of time, just contact me!

8.3. Honor Pledge: Students will be held accountable to the Honor Pledge, which they have agreed to: “I pledge, on my honor, to conduct myself with the foremost level of academic integrity.”

8.4. Academic Dishonesty: The Instructor of this course will take appropriate actions in response to Academic Dishonesty, as defined the University’s Student Code:

Acts of academic dishonesty include but are not limited to:

1. Cheating: (1) using or attempting to use or providing others with any unauthorized assistance in taking quizzes, tests, examinations, or in any other academic exercise or activity, including working in a group when the instructor has designated that the quiz, test, examination, or any other academic exercise or activity be done “individually”; (2) depending on the aid of sources beyond those authorized by the instructor in writing papers, preparing reports, solving problems, or carrying out other assignments; (3) substituting for another student, or permitting another student to substitute for oneself, in taking an examination or preparing academic work; (4) acquiring tests or other academic material belonging to a faculty member, staff member, or another student without express permission; (5) continuing to write after time has been called on a quiz, test, examination, or any other academic exercise or activity; (6) submitting substantially the same work for credit in more than one class, except with prior approval of the instructor; or (7) engaging in any form of research fraud.

2. Falsification: altering or fabricating any information or citation in an academic exercise or activity.

3. Plagiarism: representing, by paraphrase or direct quotation, the published or unpublished work of another person as one's own in any academic exercise or activity without full and clear acknowledgment. It also includes using materials prepared by another person or by an agency engaged in the sale of term papers or other academic materials.

Full text of the Student Code available at available at available at http://www.usu.edu/studentservices/pdf/StudentCode.pdf:

8.5. Special needs: Students with ADA-documented physical, sensory, emotional or medical impairments may be eligible for reasonable accommodations. Veterans may also be eligible for services. All accommodations are coordinated through the Disability Resource Center (DRC) in Room 101 of the University Inn, (435)797-2444 voice, (435)797-0740 TTY, (435)797-2444 VP, or toll free at 1-800-259-2966. Please contact the DRC as early in the semester as possible. Alternate format materials (Braille, large print or digital) are available with advance notice.