08.2_Final_Project

October 11, 2023

1 Introduction to Python for Open Source Geocomputation

- Instructor: Dr. Wei Kang

Content:

- Final Project
- Q & A
- Discussion

2 Course Project Requirements

- Individually design, implement, and write about a project which intensively uses Python programming for problem solving.
- The topic can be GIS related or unrelated.
- Three deliverables (45 points):
  - Project Idea/Proposal Presentation (5 mins) - 11/08 Class
  - Project Presentation (10-15 mins) - 12/04 or 12/06 class
  - Project Report (Jupyter Notebook) - by 12/12

2.1 Choosing Project Topic

- Project Objective/Goal:
  - To apply the lesson material to your own tasks and make them easier, faster, or more accurate
  - Up to you, but it is preferably one that relates to your current field of work or a field in which you have a personal interest

- Scale of the project: How many lines of code?
  - Not as important as the problem you solved
  - Encourage proposing a project that meets or slightly exceeds your level of experience with programming.
- Project Duration: How much time do I have to complete the project?
  - You will have 1+ classes at the end of the semester (11/29) to dedicate toward the project. This is your chance to apply what you've learned about Python to a problem that really interests you.
  - On 12/4 or 12/6, you are expected to have complete your project so that you will be able to give a presentation on it.
  - By 12/12, you need to submit a written report on the project.
- Token system:
  - cannot apply a token to presentations
  - can apply no more than one token to extend the submission date for the final report

2.2 Project ideas - What is allowed and what is not?

- Can be GIS related or not related.
- You may work on a similar project from another course or your work, or part of your thesis project, but you will need to intensively use python programming for the whole project of your choice (including (geo)data input/output, wrangling, analysis and (geo)visualization). (Double submission is not allowed and faculty members in the department work together to prevent such issues.)

2.3 Sample project ideas

- Compare dataset statistics: Make a tool or script that takes two data sets (csv or shapefile) as input, along with a column name. The tool should check whether the field is numeric and exists in both datasets. If both these conditions are met, the tool should calculate statistics for that field for both feature classes and report the difference. Statistics could be sum, average, standard deviation, etc.
- suitability analysis (reclassify, overlay,…)
- quick function to read work files and make format consistent (e.g., same length of id).
- Analysis and visualization: select your study area and data set, perform proper data manipulation, analysis (e.g., spatial analysis - hot/cold spot analysis, point cluster analysis or statistical analysis - regression, species distribution analysis, machine learning, etc), and visualization.
  - data from your work/research
  - Census data: unemployment, income, poverty, etc. https://www.census.gov/programs-surveys/acs/data.html
  - Project ideas from online posts that can help you brainstorm

2.4 Project design - An example

2.4.1 Objective

Suppose you are responsible for selecting sites for a chain restaurant. The selection criteria are:

- the land parcels need to be along a major road
- the parcels need to be no smaller than 0.25 acres

- other selection criteria

2.4.2 Implementation/ Problem solving

For each selection criterion, there is one tool in ArcGIS for accomplishing it.

If this selection process were limited to a small area, it would probably make sense to perform the work manually.

However, let's suppose you're responsible for carrying out the same analysis for several areas around the country. Because this scenario involves running the same sequence of tools for several areas, it is one that lends itself well to automation with programming.

- faster
- easier
- more accurate

2.4.3 Design

- a function for the whole site selection process by connecting the needed operations in order
- iteration for all site selections

2.5 Project proposal presentation (5 mins)

In our project idea/proposal presentation, you must clearly explain:

- the task you intend to accomplish using Python;
- how your proposed solution will make the task easier, faster, and/or more accurate
- why your task could not simply be accomplished using the "out-of-the-box" tools (e.g., from ArcGIS/QGIS), or why your script gives a particular advantage over those tools (e.g., iterations or conditionals are required)
- What other packages/libraries you expect to use for the project?
- the deliverables you will submit for the project: preferably a zip file containing the following items:
    - a well-documented python module (functions, classes, etc)
    - the data set used for this project (if the data set is very large, please only include a small sample of it)
    - a Jupyter Notebook with Markdown cells for narratives, and code cells for python code (for importing modules and run functions/statements/classes/visualization).

2.5.1 Caveats about the proposal presentation

- Submit your proposal presentation slide to Canvas before the presentation
- The proposal will be used to **help grade the rest of your project.**
- Your proposal must be **approved** by the instructor before you move forward with coding the project.

2.6 Project Evaluation criteria

- Based on both project presentation and report

- Python program:

- The project intensively use python programming for the whole project of your choice (including (geo)data input/output, wrangling, analysis and (geo)visualization).
- The python program needs to be carefully designed (modularity is important with functions, and classes), cleaned (no redundant code lines), and documented (proper docstrings).
- The submitted Jupyter Notebook runs without any error.

- Writing:

  - Clearly describe your project's objectives, plans, results, deliverables, and conclusions.
  - Free of grammar mistakes.
  - Proper usage of Markdown syntax in the Jupyter Notebook.
    * headings
    * lists
    * links