

Lab Report

Title: GIS 5571 – Lab 0
Notice: Dr. Bryan Runck
Author: Luke Zaruba
Date: 21 September 2022

Project Repository: <https://github.com/lukezaruba/GIS5571>

Google Drive Link: N/A

Time Spent: 4.0

Abstract

A simple buffer analysis is a common task in any GIS application, but in the Esri ecosystem there are a variety of different ways to perform the same analysis and achieve the same outcomes. The case-study of creating a 100-foot buffer around the road network for the City of Minneapolis is used to examine three distinct workflows, in which buffers can be created using Esri products/tools. The workflows differ via their environments (i.e., desktop versus cloud) and deployment methods (i.e., GUI versus raw code), which are explained in greater detail in the subsequent paragraphs. There are pros and cons of each workflow, but ultimately, they all have a time and place in which they should be utilized, depending on the use-case requirements and desired outcomes.

Problem Statement

The purpose of this analysis is to show one example of how Esri tools can perform the same functions via a variety of different workflows and tools. The example at hand, performing a simple buffer analysis on a roads dataset, will be completed three different ways, using ArcGIS Pro, Jupyter Notebooks in ArcGIS Pro (using ArcPy), and Jupyter Notebooks in ArcGIS Online (using the ArcGIS API for Python).

Table 1. Analysis requirements for buffer analysis road dataset.

| # | Requirement | Defined As | (Spatial) Data | Attribute Data | Dataset | Preparation |
|---|--------------|---|----------------|---|---|-------------------|
| 1 | Road network | Raw input dataset containing road network | Road geometry | No attributes are necessary in the analysis | City of Minneapolis Open Data | No prep necessary |

Input Data

The data used in this analysis is a roads dataset downloaded from the City of Minneapolis Open Data website, featuring all the roads in the city boundary. Although the data contains a wide variety of attribute fields, for the purpose of this analysis, only the feature geometries are needed to perform the buffer.

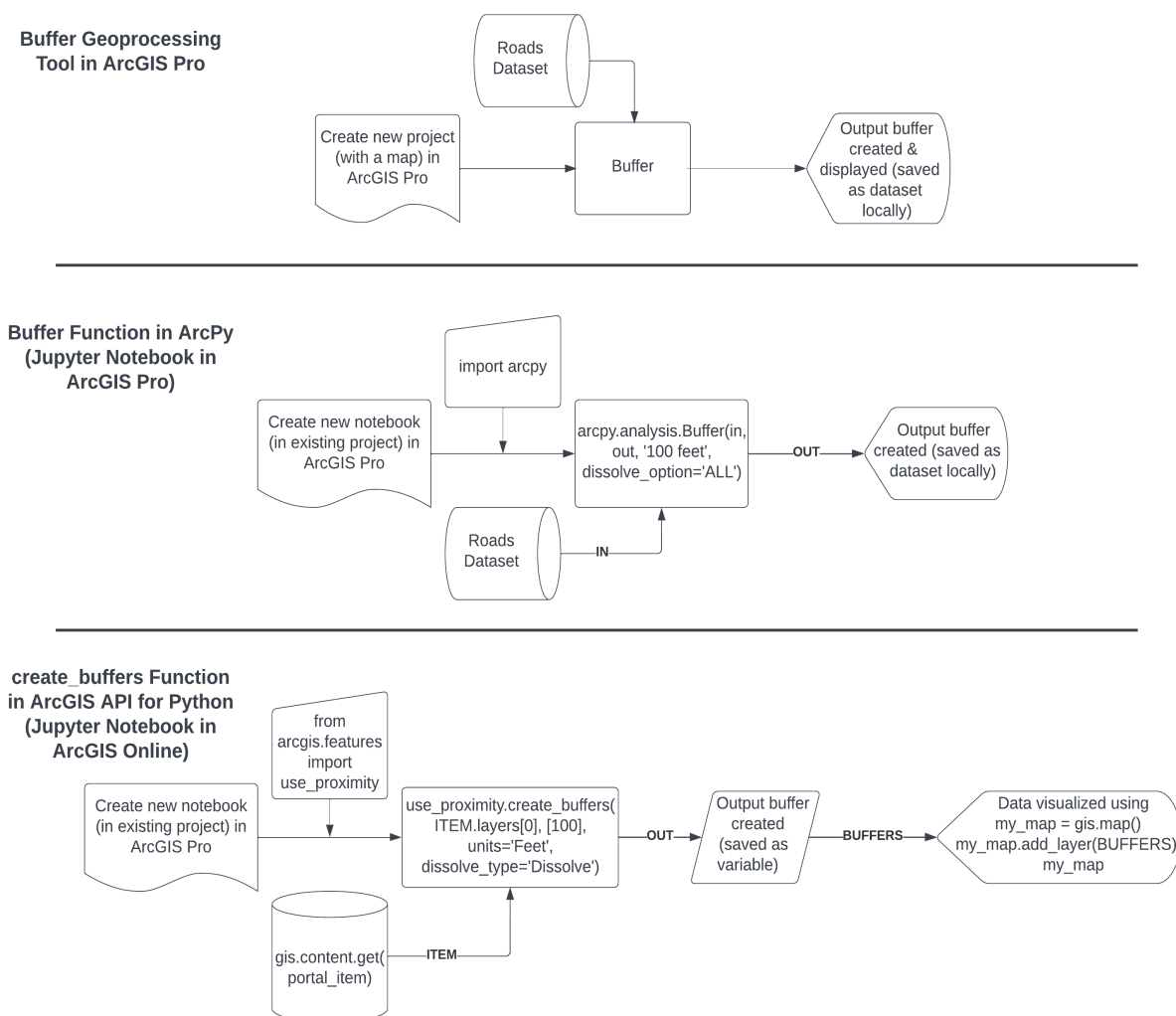
Table 2. Input dataset for buffer analysis

| # | Title | Purpose in Analysis | Link to Source |
|---|-------------------|--|--|
| 1 | Minneapolis Roads | Raw input dataset for buffer analysis from City of Minneapolis | City of Minneapolis Open Data Site |

Methods

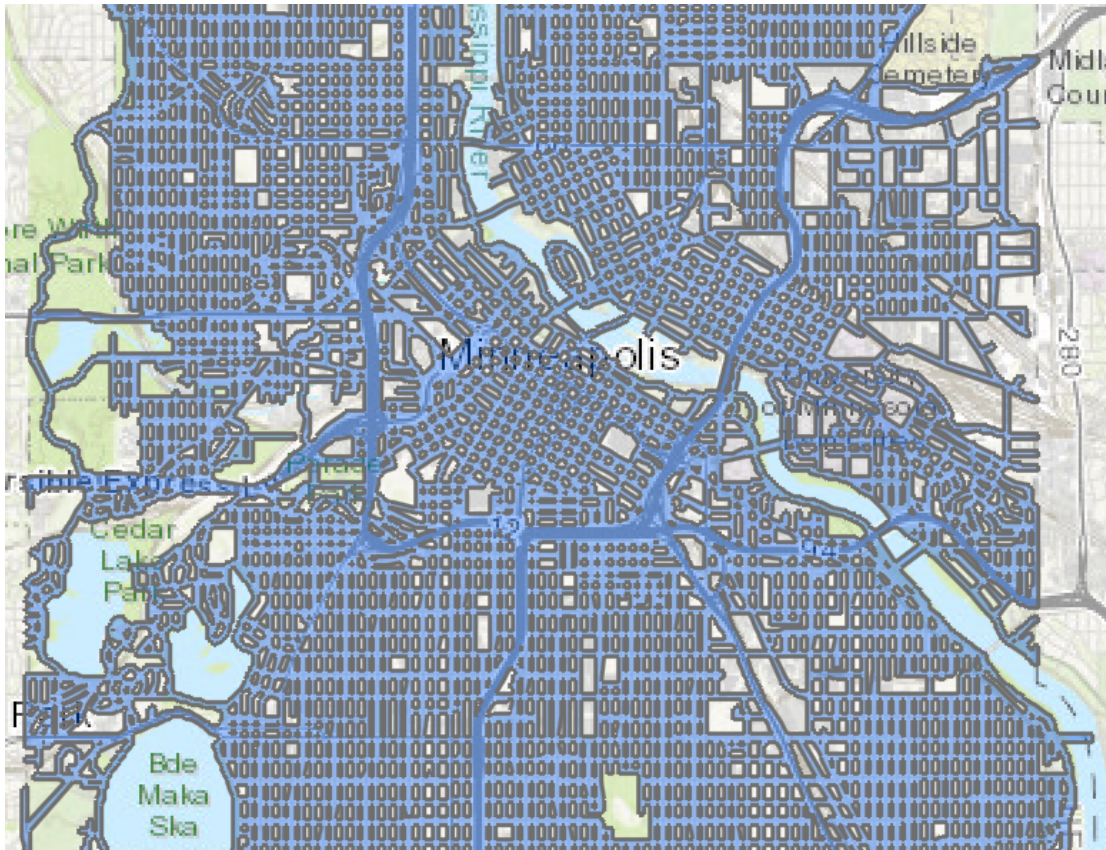
The analysis contained three distinct workflows, all resulting in the same resulting dataset, a 100-foot buffer around the input road network. As seen in Figure 1, the simplest and most intuitive workflow was to perform the buffer using the GUI of ArcGIS Pro, which essentially means that users can use an interface in the software to interact with ArcPy, without coding. This option is reliable and easy to use, especially if you are not comfortable utilizing ArcPy or the ArcGIS API for Python, which are two Python modules created by Esri to streamline and automate geospatial analyses in the Esri suite of GIS software and tools. For tasks that require automation or optimization, this is where using Python-based tools become far superior in terms of the power that they provide. Although the two workflows for using ArcPy and the ArcGIS API for Python look different, they are not very different in terms how they can be thought of computationally. However, they do vary syntactically and, in this case, vary in the environment in which they were deployed (in ArcGIS Pro versus in the cloud). There are some differences in how/where/if output data is saved and how the outputs can be displayed, between the modules, but ultimately the decision of which module to use depends on the specific requirements of a given scenario.

Figure 1. Data flow diagram containing the three workflows used in the analysis.



Results

Figure 2. An example of what the analysis results in, when using the ArcGIS API for Python



Results Verification

I was able to confirm that the results were correct by visually scanning and comparing the datasets to one another. In ArcGIS Pro, this was very simple to confirm via the use of the Measure tool, which I used to measure the distance between the centerline and the edge of the buffers, and confirm that it was indeed a 100-foot distance.

Discussion and Conclusion

Buffer Analysis

Although I have done many buffers using both ArcGIS Pro and ArcPy, I had never used the ArcGIS API for Python, so it was interesting to compare the workflow with ArcGIS Pro's and ArcPy's. I think that it is very clear to see how the two packages were designed for different use-cases (e.g., ArcPy for integration with desktop GIS and ArcGIS API for Python for cloud native/web-based GIS).

GitHub

I have used GitHub quite a bit over the last year or so, as I have started to dig deeper into the field of spatial data science, so that made this process straightforward and simple, since I have done many of the same tasks before. I did find it interesting to learn a little more about pull requests and branching, since those are two things that I don't have much experience with.

References

Esri. (2022). *arcgis.features.use_proximity module*. ArcGIS API for Python - API Reference. Retrieved from https://developers.arcgis.com/python/apireference/arcgis.features.use_proximity.html

Self-score

| Category | Description | Points Possible | Score |
|----------------------------|--|-----------------|-----------|
| Structural Elements | All elements of a lab report are included (2 points each): Title, Notice: Dr. Bryan Runck, Author, Project Repository, Date, Abstract, Problem Statement, Input Data w/ tables, Methods w/ Data, Flow Diagrams, Results, Results Verification, Discussion and Conclusion, References in common format, Self-score | 28 | 28 |
| Clarity of Content | Each element above is executed at a professional level so that someone can understand the goal, data, methods, results, and their validity and implications in a 5 minute reading at a cursory-level, and in a 30 minute meeting at a deep level (12 points). There is a clear connection from data to results to discussion and conclusion (12 points). | 24 | 22 |
| Reproducibility | Results are completely reproducible by someone with basic GIS training. There is no ambiguity in data flow or rationale for data operations. Every step is documented and justified. | 28 | 28 |
| Verification | Results are correct in that they have been verified in comparison to some standard. The standard is clearly stated (10 points), the method of comparison is clearly stated (5 points), and the result of verification is clearly stated (5 points). | 20 | 19 |
| | | 100 | 97 |