

Lab Report

Title: Lab 0

Notice: Dr. Bryan Runck

Author: Jarred Paquin

Date: 9/22/2021

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

Time Spent: 4 Hours

Abstract

In this lab, I configured GitHub and created a repository for storing all of my class data. I then compared three different GIS tools – ArcPro, Jupyter Notebooks in ArcPro, and Jupyter Notebooks in ArcOnline – by running a simple buffer analysis in all three. I found that the process was fairly similar across all three, with the exception that both Jupyter Notebooks methods required several intermediary steps to prepare the data for the buffer analysis.

Problem Statement

There were two purposes to this lab. The first one was to download and familiarize myself with Git and GitHub. The second purpose was to compare the ways of accomplishing the same basic task – a buffer – in three different ways using Esri tools – ArcPro, Jupyter Notebooks in ArcPro, and Jupyter Notebooks in ArcOnline.

Input Data

The data I used was a shapefile depicting transit routes in the Twin Cities MSA. I ran into some access issues when using this for the ArcOnline portion, so I ended up using a layer of the Los Angeles transit stops.

Table 1. Data Used

#	Title	Purpose in Analysis	Link to Source
1	TC Transit Routes	Use to practice conducting a buffer	MN Geospatial Commons
2	LA Transit Stops	Use to practice conducting a buffer	ArcOnline Layer

Methods

My process for all three tools followed the same general structure. I started by importing the data into my workspace, did any intermediary preparation, and then ran the buffer analysis. In ArcPro, there was no intermediary preparation required, and I was able to move immediately into running my buffer tool. In Jupyter Notebooks for ArcPro, I had to first set the environment to the correct geodatabase and then set a variable to the correct feature class in that geodatabase before I was able to conduct my buffer. For Jupyter Notebooks in ArcOnline, I also had to set a variable to the correct layer before being able to run by buffer.

Figure 1. Flow Diagram of My Process in ArcPro

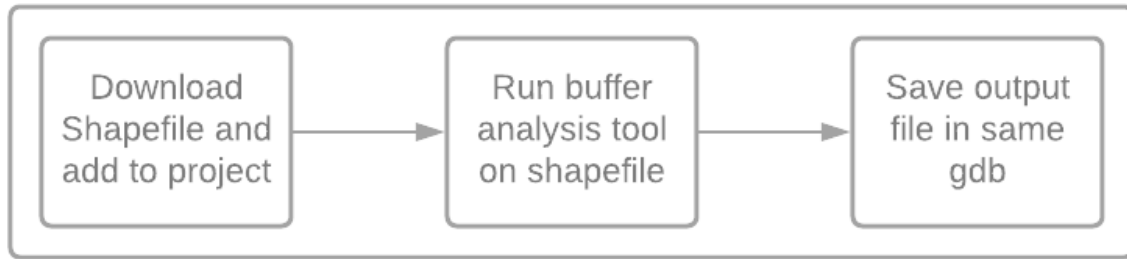


Figure 2. Flow Diagram of My Process in Jupyter Notebooks in ArcPro

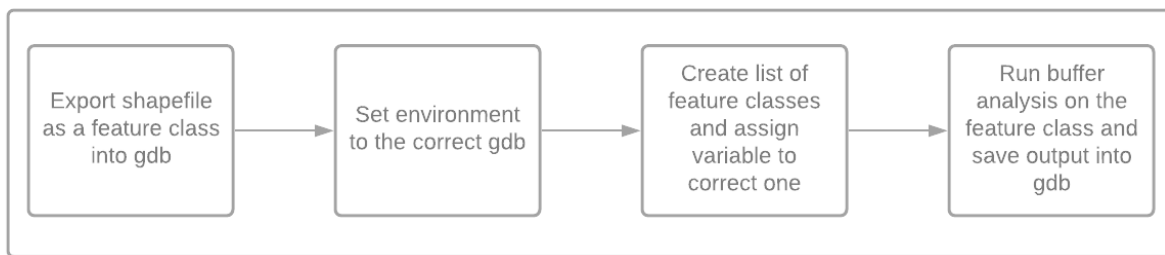
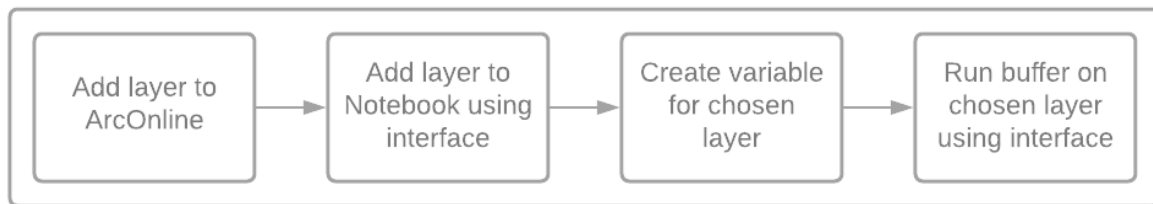


Figure 3. Flow Diagram of My Process in Jupyter Notebooks in ArcOnline



Results

While the overall result of each buffer analysis from the ArcPro tools was the same – and the result from the buffer from the ArcOnline tool would have been the same if not for the access issue – the process varied slightly between each. As can be seen in the table below, each tool had several steps in common – namely adding the data and running the buffer – but both Jupyter Notebook methods involved intermediary steps. The difference in adding data to the tool is rather arbitrary, as I could have chosen to add my data as a live link with the two ArcPro tools, just as I could have chosen to upload my data as a shapefile to the ArcOnline tool.

Table 2. Comparison of Specific Tasks Needed in Each Tool

Tool	Download as shapefile/feature class	Import as link to updating data	Set environment	Assign variables	Run Buffer
ArcPro	X				X
Jupyter - ArcPro	X		X	X	X
Jupyter - ArcOnline		X		X	X

Results Verification

The resulting buffer from the two ArcPro tools was identical, indicating the buffer was done correctly. I verified the results of the buffer from the ArcOnline tool by using that same data source with the normal ArcOnline interface and achieved the same results.

Discussion and Conclusion

One thing I found frustrating while completing this lab was that Jupyter Notebooks in ArcPro and Jupyter Notebooks in ArcOnline use different Python modules, which means familiarizing oneself with the Arcpy module in ArcPro does transfer well to the GIS module in ArcOnline.

For the GitHub aspect, most of that was review for me as I have used Git in the past. I had never downloaded and configured Git on my own machine before, so that part was a bit tricky. Once I had it set up, creating repositories and pushing/pulling data to and from Git felt very simple for me.

References

No references aside from those provided in the Lab Handout

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	20
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	24
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	20
		100	92