Lab Report 2.1

Title: Processing and Interpreting Elevation Data using arcpy
Notice: Dr. Bryan Runck
Author: Evening Hade
Date: 29 October 2024

Project Repository: https://github.com/eveningsaria/gis5571/tree/main/Lab2
Google Drive Link: N/A
Time Spent: 40 hours (not including looking for stock photos)

Abstract
Using API queries, different types of terrain data were downloaded using an ETL pipeline for manipulation in ArcGIS Pro via its Jupyter Notebooks integration. Within ArcPro, new visualizations of the data were created and compared. When analyzing ArcPro's 2D and 3D visualization capabilities, it is clear that this is a professional tool for a reason. The wide array of data types accepted and display options for that data were impressive. At the same time, the formatting of the data had to be specific in order for the 3D analysis tools to work with it. All in all, this was an exercise in patience and in the give and take that is necessarily a part of having a wide array of options within ArcPro.

Problem Statement
The purpose of this exercise is to examine different types of elevation related data. The focus of this exercise was less on the actual data and more on its extraction, transformation, and exportation. Skills making API requests and ETL pipelines were combined with exploratory spatial analysis.

Table 1. Types of data used.

#	Requirement	Defined As	(Spatial) Data	Attribute Data	Dataset	Preparation
1	Elevation	Sample .las data	Point feature	Terrain elevation	MNDNR	Uploaded to MNDNR
2	Rainfall	Monthly 30yr normals as .bil files	Raster image	Precipitation	PRISM	Uploaded to PRISM

Figure 0. A resident scientist surveying elevation.

Input Data
A vector .las file was downloaded from the MNDNR, while raster .bil files were downloaded from PRISM to represent two functionally different ways of presenting elevation data. The MNDNR data was a .las file created as a sample of MNDNR's wide collection of LiDAR data available for public use. A sample size was chosen due to the large size of .las data files. The .las data contained point feature data, but it needed to be turned into a .las dataset before it could be used. The PRISM data were .bil files of the 30 year normals of precipitation data in the continental United States of America. These were saved as .zip files and needed to be extracted before they could be used. Along with the .bil files, attribute data was included in these .zip files to facilitate ease of use.

Table 2. Datasets used in this exercise.

#	Title	Purpose in Analysis	Link to Source
1	4342_12_06	LiDAR .LAS sample elevation data from the Minnesota Department of Natural Resources	MNDNR
2	PRISM_ppt_30yr_normal_4kmM4_all_bil	Collection of .BIL files containing the 30 year precipitation normals for the 12 months of the year	PRISM

Methods
The .las data from MNDNR was manipulated first using Python and the modules `arcpy`, `requests`, and `os`. To start, the data was accessed by using `requests.get()` on a url endpoint for the data and saved to a file folder in `os`. Next, the data was converted to an LAS dataset in ArcGIS Pro and added to the active map. This LASD was then converted into a DEM layer and a TIN layer. These layers were each added to their own unique maps, which were added as map frames to their own unique layouts, then exported as PDFs. Outside of the Jupyter Notebooks coding environment within ArcPro, a 3D map scene was created from the LASD and visually compared to the 2D LASD layer on the map.

Once done with the MNDNR data, a new ArcPro project was created using the GUI and a new Notebook made to manipulate the PRISM data. The modules `arcpy`, `requests`, and `os` were once again used along with `zipfile`. Again, the data was accessed by using `requests.get()` on a url endpoint for the data and saved to a file folder in `os`. Uniquely, this process was repeated twelve times using loops to get data for each of the months of the year. While a single .ZIP file containing all twelve months' data was available through the PRISM website's GUI, I struggled to find the url endpoint for this folder and found a workaround. Next, the data was extracted from the .ZIP files and the .BIL files were identified and added to a list. These were added to a mosaic dataset, which was turned into a multidimensional raster layer with a time dimension based on the months of the year. After creating a new, distinct multidimensional raster layer, this new layer was then turned into a space time cube.

Figure 1. Data flow diagram for the DNR data.

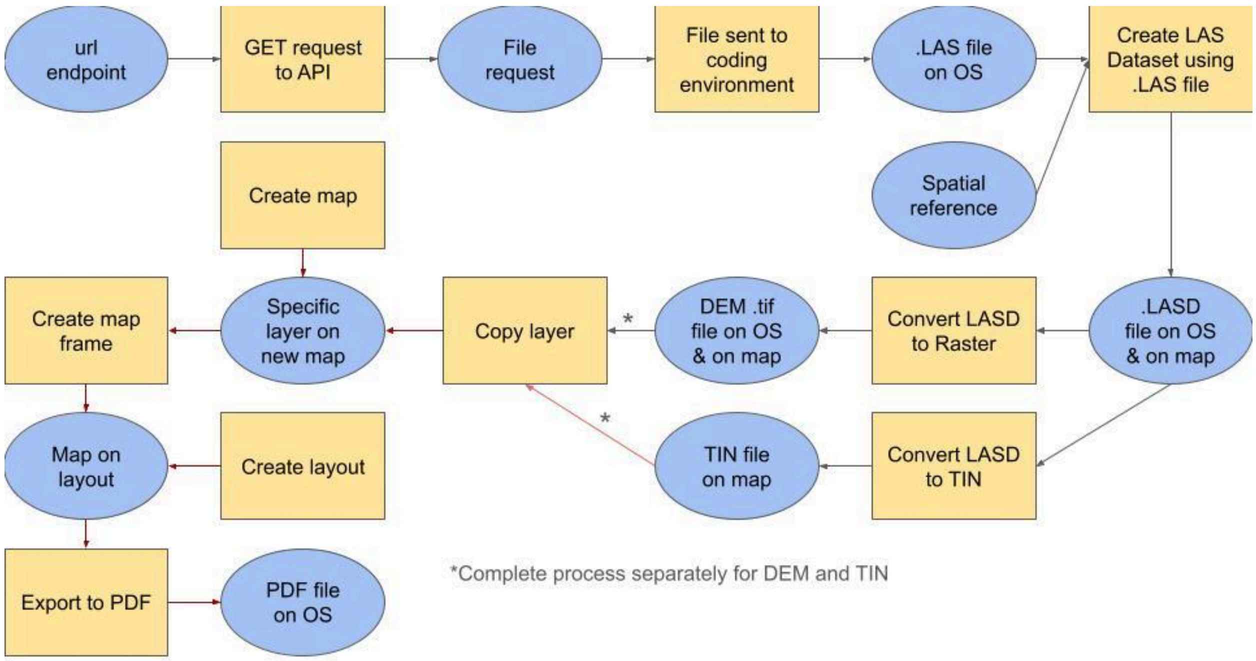
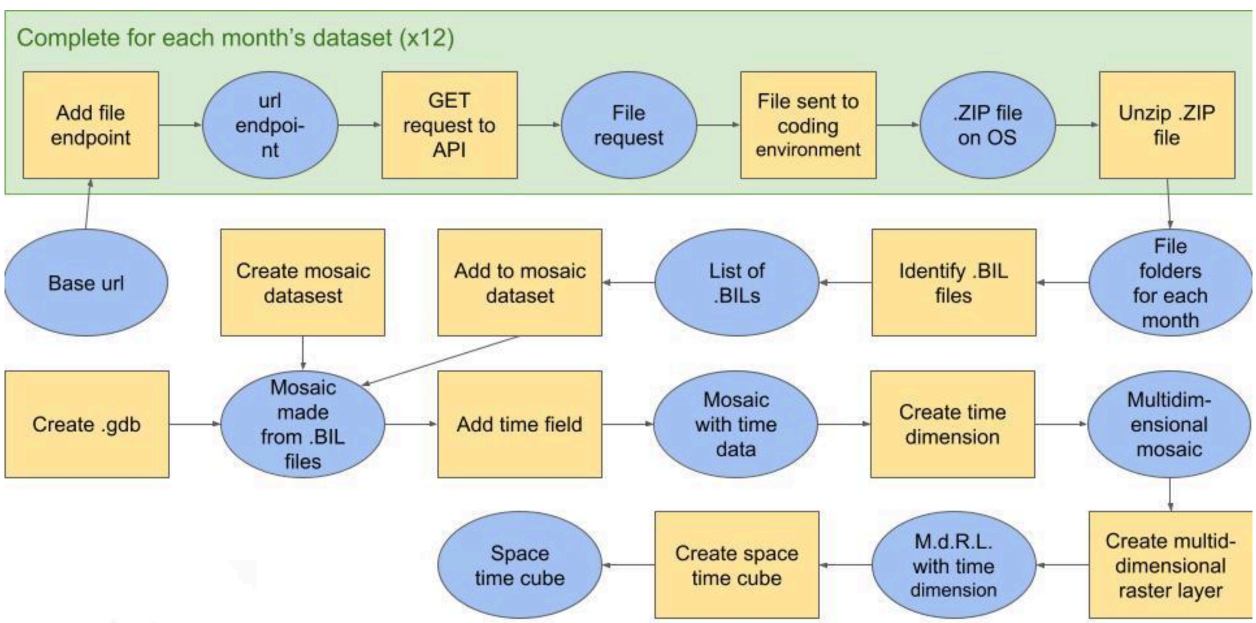


Figure 2. Data flow diagram for the PRISM data.



Results
Attempts to extract, transform, load, and manipulate vector .las data from MNDNR and raster .bil data from PRISM were fruitful. While the LASD was beautiful, almost like a pointellism painting, the large file size meant that I found the DEM (Figure 3) and TIN (Figure 4) more useful representations of the data. That said, the LASD is valuable because it can be transformed into so many different representations and it absolutely has a place in spatial data analysis using ArcGIS Pro. Seeing the map scene made from the LASD felt almost like something out of a scifi movie. I would say that the map scene is useful for more detailed analysis because a geographer can fully visualize the 3D nature of the terrain, but the 2D LASD is already quite precise and stunning in its own way. 3D map scenes might be especially useful to the construction industry or to those making intense weather models. Unique to the Esri suite, the TIN file seemed to struggle slightly with the minutiae of the river valley I was modeling, but this could have been a problem with my computer. Overall, I would say the DEM is (still) my favorite representation of the three (Figure 3). Moving on to the PRISM data, my laptop couldn't even load anything beyond the BIL files (Figure 5) and the mosaic dataset. While multidimensional raster layers and space time cubes could have their place in GIS and obviously create rather powerful visuals, the nature of these visualizations means they are not available to all geographers, including me. This sours my opinion of the visualizations, especially of the space time cube. Even creating it was difficult. The process was computationally intensive and required me to use a different computer than my personal laptop. Finally, a timeseries of the multidimensional raster was meant to be created, yet there were problems exporting it as anything but a still image of just January's 30 year normals. This timeseries has been excluded from this report.

Results Verification
While observations about the efficacy of certain Esri programs are necessarily opinionated, all of the processes of this exercise can be easily replicated. The data is freely and publicly available online. So long as one has access to ArcGIS Pro and an Internet connection, they may download the same data I did and manipulate it in the same way.

Discussion and Conclusion
Prior to this lab, I had not worked with half of these file formats. It was educational to explore the different ways one can represent elevation data in ArcGIS Pro. As I explored the different 3D analysis tools and the map scene creator, I could tell that there were even more tools for terrain analysis beyond the scope of this lab. I was pleasantly surprised at how easy the ETL pipelines were for me to create. My only trouble was trying to find the correct endpoint for the PRISM data. This was a welcome surprise because it made me a little more confident in my coding skills. A major difficulty of this lab was how long everything took to load. I don't think my computer is equipped for these computationally intense exercises. To be honest, I think ArcGIS Pro is barely equipped for them, and users should approach with a healthy level of caution. It would be fun to see what GIS-ers could do in ArcGIS Pro on a more advanced computer. As Esri tends to market its products as top of the line, I would not be surprised if these geographers were Esri's target audience anyway. As far as my experience with the lab, I was happy to work through everything here (minus the days and days of crashing due to the space time cube) and I found this a valuable exercise (minus the space time cube).

Figure 3. DEM layer created from the DNR's .las data.

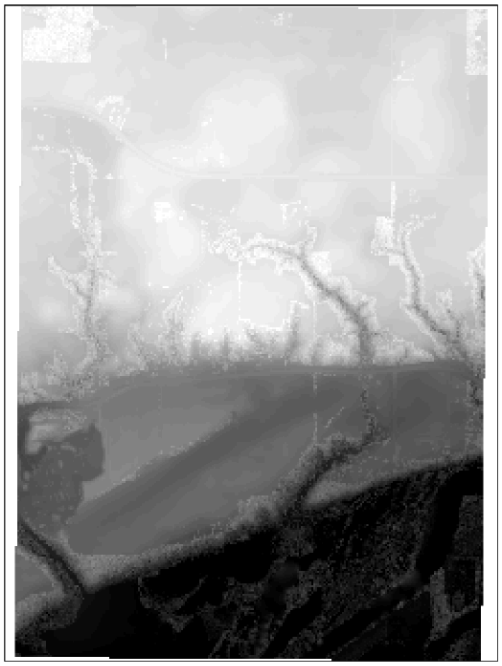


Figure 4. TIN layer created from the DNR's .las data.

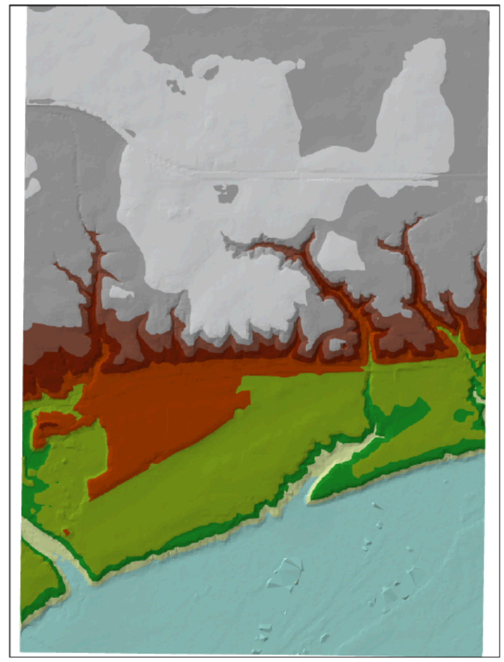


Figure 5. BIL raster image accessed via PRISM's API.

References

Esri. (2020). *Explore your raster data with Space Time Pattern Mining*. ArcGIS Blog. https://www.esri.com/arcgis-blog/products/arcgis-pro/analytics/explore-your-raster-data-with-space-time-pattern-mining/

Minnesota Department of Natural Resources. (2012). *LiDAR Elevation Data Examples*. Minnesota GIS Data Resources. https://resources.gisdata.mn.gov/pub/data/elevation/lidar/examples/

Northwest Alliance for Computational Science and Engineering. (2024). *30-Year Normals*. PRISM Climate Group at Oregon State University. https://prism.oregonstate.edu/normals/

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	**24**
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	**20**
		100	**100**