Lab Report

Title: Lab 2 - Part 1
Notice: Dr. Bryan Runck
Author: Greg Kohler
Date: October 31st, 2023

Project Repository: Lab 2 Repository
Google Drive Link: Link to Google Drive Folder
Time Spent: 13

Abstract

This lab report covers the first part of a two-part lab. This lab involved downloading a LAS file from the Minnesota DNR and converting it to both DEM and TIN formats. It also involved visualizing this LAS data in both 2D and 3D views in ArcGIS Pro. This lab also involved downloading BIL files from the PRISM Climate Group and converting them into a multidimensional raster layer, which allows visualization of precipitation over time. The methods used to download and work with the data are represented in flowcharts. The results are showcased in figures of the different raster layers and as an animation of the changes in precipitation. A comparison of the 2D and 3D visualizations is shown in a table. The results are verified by comparing the LAS figures to ensure they represent the same landscape and by checking the conversion code. The results of the multidimensional raster layer are verified by confirming the code ran successfully and visually inspecting the timestamps shown in the animation window of ArcGIS Pro. The final section of this report discusses lessons learned from this part of the lab.

Problem Statement

The first part of this lab was split into three different sections. The purpose of the first section of this lab is to build an ETL pipeline to download LAS data and transform it into TIN and DEM layers. These layers can then be visualized and compared. The purpose of the second section of this lab was to use ArcGIS Pro to visualize the LAS file in 2D and 3D. This allows for an exploration of the capabilities of ArcGIS Pro with this raster data. The purpose of the third section was to design another ETL pipeline that downloads BIL files, turns them into an animation, and also converts them into spacetime cubes.

Table 1 - Resources Used

#	Requirement	Defined As	(Spatial) Data	Attribute Data	Dataset	Preparation
1	ArcGIS Pro Notebook	Python Notebooks used to build the ETL Pipelines.	N/A	N/a	N/A	N/A
2	Minnesota Geospatial Commons	URL to download raster files from	Raster data	Elevation	LAS Files	Had to use correct url for LAS file.

3	PRISM	URL to download BIL files	Raster data	Precipitation	BIL Files	Had to find correct URL for BIL files ZIP

Input Data

For this section of the lab, all of the data used was raster data that was obtained from two sources. The first dataset was a LAS file from the Minnesota DNR, obtained from the Minnesota Geospatial Commons. This file represented the elevation of a small section of land in Minnesota. This data was represented in 2D and 3D, it was also converted into DEM and TIN layers. The other dataset came from the PRISM Climate Group. This was a collection of BIL files that represented 30-year normals for precipitation in the United States. This data ranged from 1991-2021. This data was used to create a multidimensional raster layer, which helped us visualize how precipitation differs for each month. This data was also used to produce spacetime cubes.

Table 2 - Data Used

#	**Title**	**Purpose in Analysis**	**Link to Source**
1	Minnesota DNR LAS Files	Used for 2D and 3D visualization. Also converted into DEM and TIN layers.	Mn GeoSpatial Commons
2	PRISM 30 year normals for precipitation	Used to create multidimensional raster layer to animate how precipitation changes. Also used to create spacetime cubes.	PRISM Climate Group

Methods

This part of the lab was split into three sections. For the first sections, the first step was to download the LAS files from the Minnesota Geospatial Commons (See Figure 1). This first involved requesting the file from the URL and writing it to a specified folder path. Next, I had to create an empty LAS dataset and add the newly downloaded LAS file to it (OpenAI, 2023).

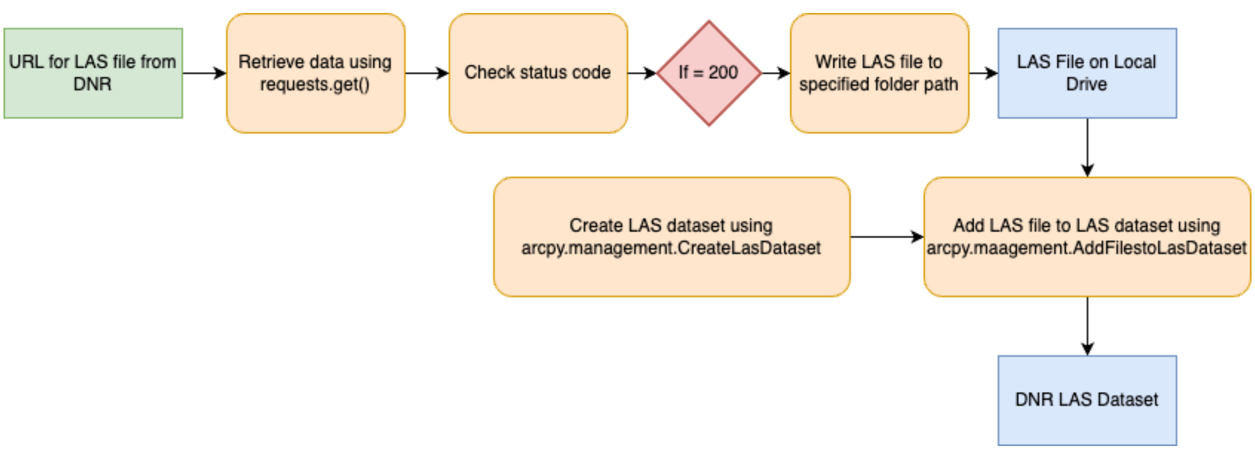

Figure 1 - Obtaining LAS Dataset

After the LAS dataset was ready, the next piece was to convert the LAS dataset into a TIN file, see Figure 2. The first piece was using the arcpy tool, LasDatasettoTin. This converts the LAS dataset into a TIN file and saves it to the project. With some guidance from OpenAI (2023), I was able to export the TIN layer to PDF. This process first involved creating a layout with a map frame, making the TIN layer the only visible one, and setting the map frame extent to the TIN layer. Once all of this was set, is exported to PDF.

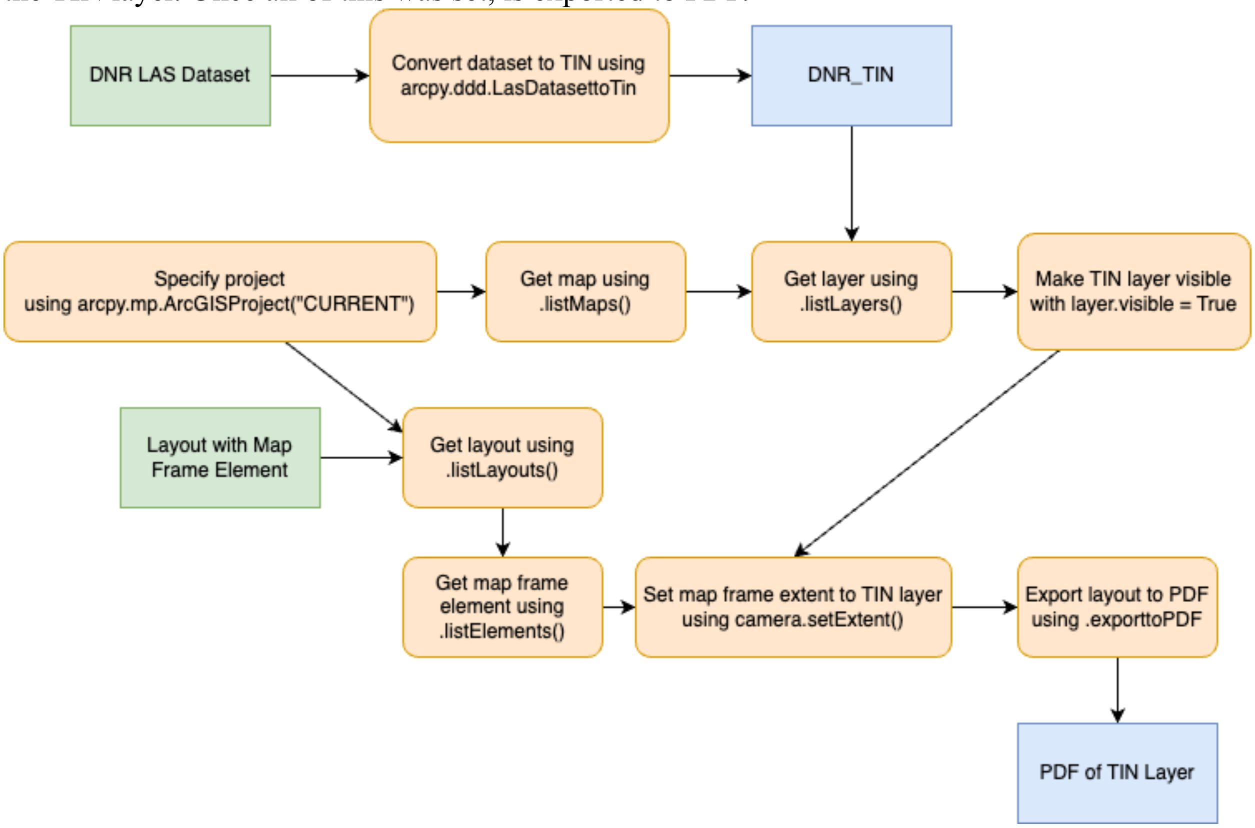

Figure 2 - LAS to TIN to PDF

After obtaining the PDF of the TIN image, I had to create and export an image of a DEM layer. This followed a very similar process as Figure 2. The main difference was the use of the arcpy function LasDatasettoRaster. This converted the LasDataset into the DEM raster. After this the process was the same as exporting the TIN to a PDF.

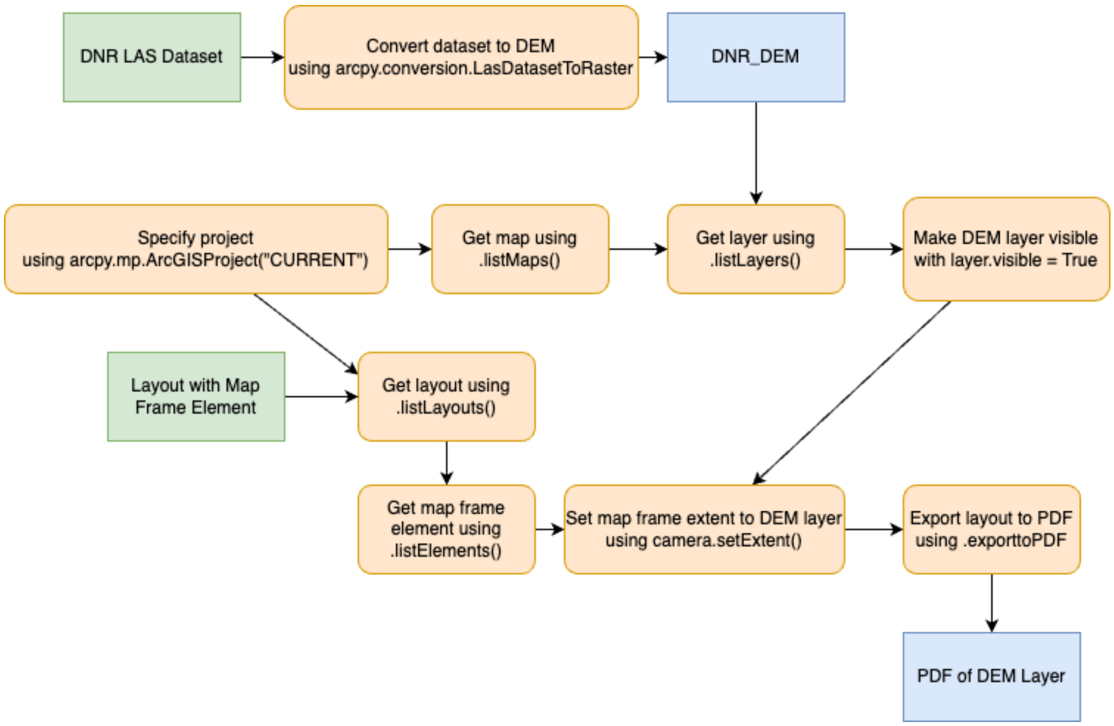

Figure 3 - LAS to DEM to PDF

After creating the DEM and TIN layers and exporting them to PDF, the next section of this lab required visualizing the LAS files in both 2D and 3D. This was a fairly straightforward process (See Figure 4). For the 2D visualization, the LAS needed to be thinned to allow for visualization at the zoomed-out scale (OpenAI, 2023). Once it was thinned, it was added to the map for a 2D visualization. For the 3D visualization, the LAS file had to be added to a 3D map scene.

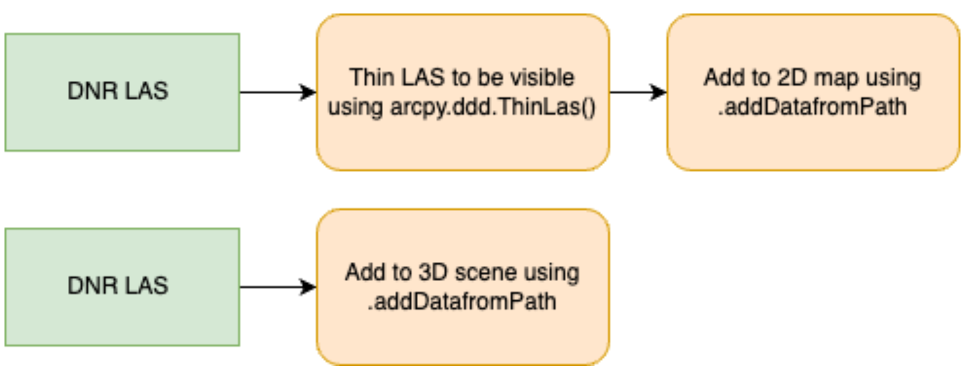

Figure 4 - LAS Visualization

For the third section of this part of the lab, the goal was to obtain data from the PRISM Climate Group, use it to visualize temporal precipitation data and convert it to spacetime cubes. The first step was to obtain a zip file containing the BIL files from the PRISM website, and then to extract the BIL files from the zip file. Next, I had to convert the BIL files into a usable format using an arcpy conversion function, RasterToOtherFormat (OpenAI, 2023). This converted the

BIL files and put them into a working folder. Then I created a mosaic dataset using the arcpy management function CreateMosaicDataset. I added the converted raster files into this empty mosaic dataset. This mosaic dataset had three parts, the boundary, the footprint, and the image. Within the footprint table, I had to calculate both the Variable and Timestamp fields. For the Timestamp field, I calculated the field to correspond with the month of each of the 30-year normals for precipitation (Buie, 2020).

With these fields calculated. Used the arcpy function BuildMultidimensionalInfo to build info for the mosaic dataset. Once this multidimensional info was built, I was able to create the multidimensional raster layer using the arcpy function MakeMultidimensionalRasterLayer. This created a raster layer that was temporal and changed based on the month. This was used to create and export an animation of the 30-year normals for precipitation. It was also used with the arcpy function CreateSpaceTimeCubeMDRasterLayer to convert the multidimensional raster layer into spacetime cubes and write it to my disk (Buie, 2020).

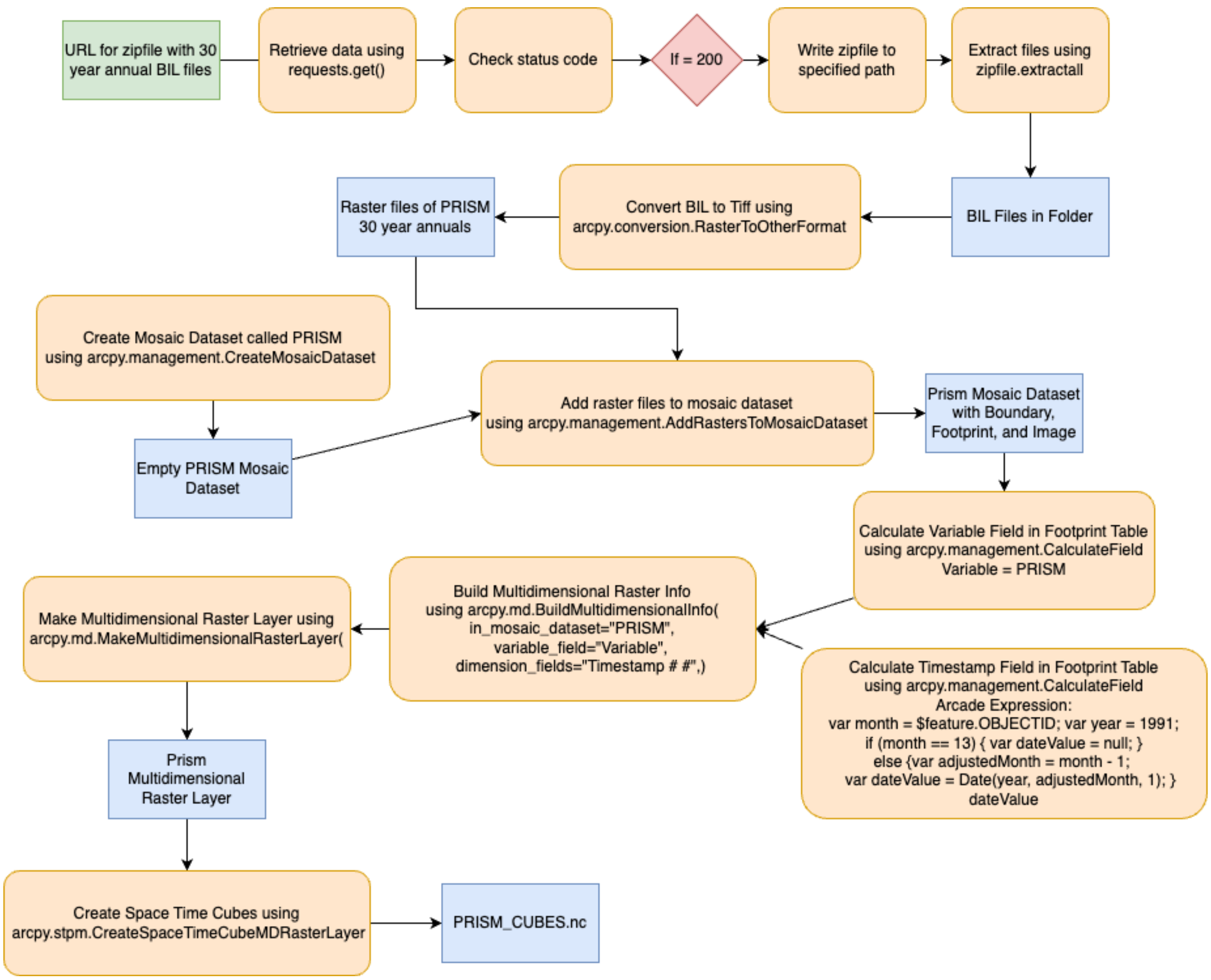

Figure 5 - PRISM Data and Visualization

Results

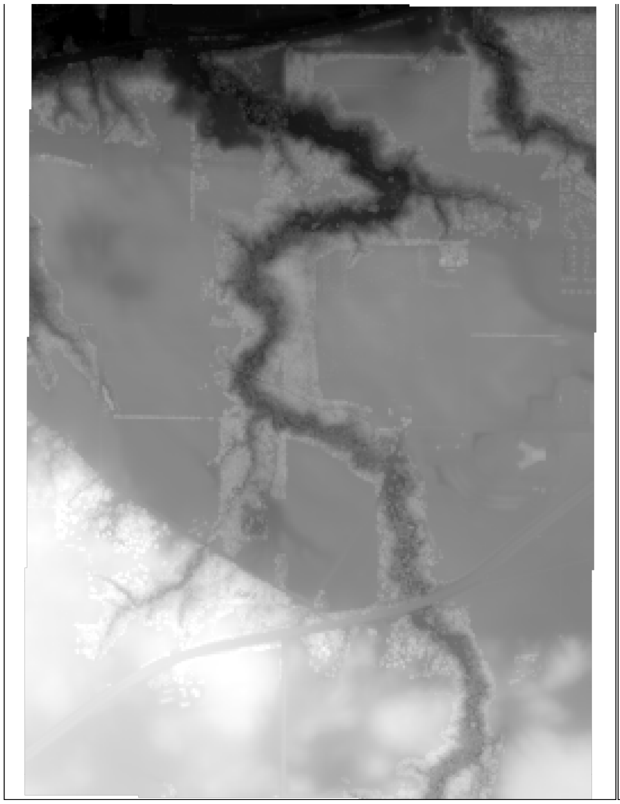

Figure 6 - DEM Layer

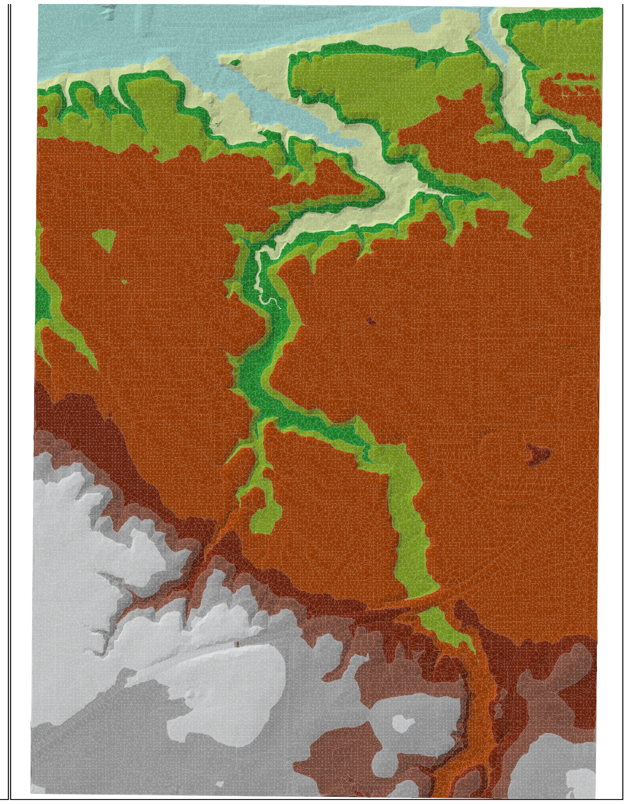

Figure 7 - TIN Layer

The first section of this part of the lab generated visualizations of the raster data as both a DEM, see Figure 6 and a TIN, see Figure 7. These were both converted from the LAS files obtained from the Minnesota DNR and exported to PDF.

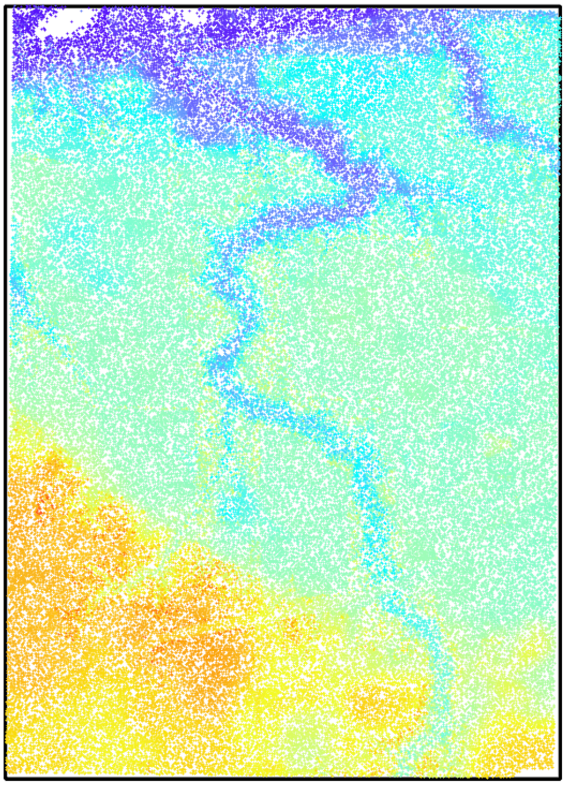

Figure 8 - 2D LAS

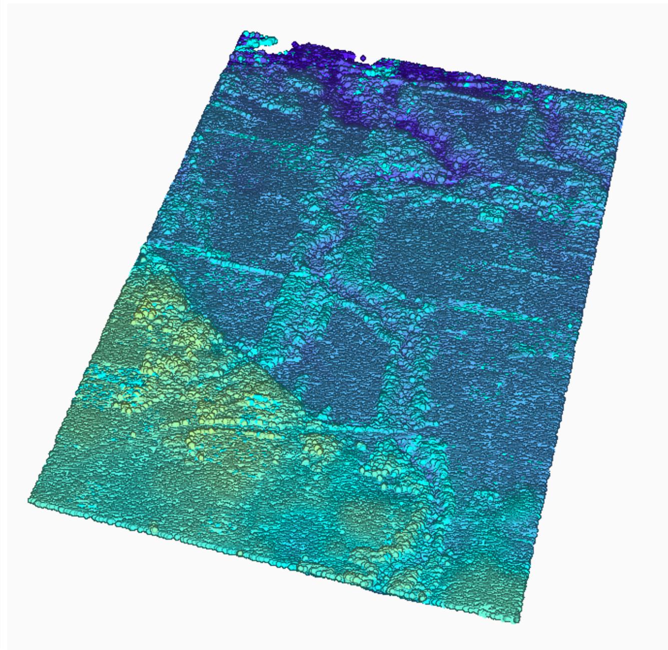

Figure 9 - 3D LAS

For the next step of this lab, two different visualizations of the LAS data were presented. A 2D visualization, see Figure 8, and a 3D visualization, see Figure 9. These show the elevation data from different perspectives. See Table 3 for a comparison of the features provided by ArcGIS Pro for 2D and 3D visualization.

Table 3 - Comparing 2D and 3D Visualization Tools

Feature	**2D Visualization**	**3D Visualization**
Display	Displays LAS Files	Displays LAS Files
Enviroment	Map	Scene
Required Point Thinning	Yes	No
Simplicity	Easier to visualize	Richer detail
Elevation	Harder to interpret elevation	Easier to visualize elevation
Navigation	Basic Controls	Allows for different perspectives and angles
Symbology	Draw based on elevation	Draw based on elevation
Data Display	Points	Points

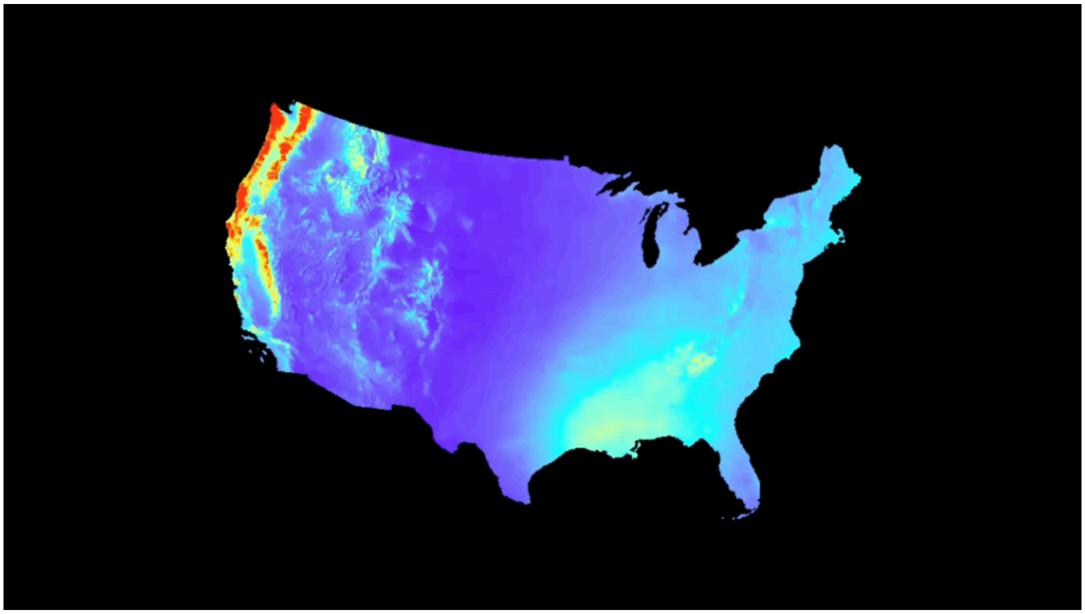

Figure 10 - 30 Year Normals for Precipitation

The final result for part one of this lab was an animation showing the 30-year normals for precipitation in the United States. See Figure 10 for a still image and see the *Google Drive Link* for the full working animation. This animation shows a different raster display for each month of the year. Each month corresponds to 30-year precipitation normals for that month.

Results Verification

For results verification, we can verify the raster data was correctly visualized by looking at the TIN (Figure 7), the DEM (Figure 6), the 2D LAS (Figure 8), and the 3D LAS (Figure 9). We can see that all of these visualizations have a similar appearance, showing a winding river/creek bed going through the middle of the figures. This verifies that all of the visualizations are of the same space.

Convert LAS to TIN

```
arcpy.ddd.LasDatasetToTin(
    in_las_dataset=r"C:\Users\gregkohler1\GIS\Lab2\Lab2\DNR.lasd",
    out_tin=r"C:\Users\gregkohler1\GIS\Lab2\Lab2\DNR_TIN",
    thinning_type="WINDOW_SIZE",
    thinning_method="",
    thinning_value=None,
    max_nodes=500000,
    z_factor=1,
    clip_to_extent="CLIP"
)
```

Figure 11 - TIN Conversion Code

Convert LAS File to DEM

```
arcpy.conversion.LasDatasetToRaster(
    in_las_dataset="DNR.lasd",
    out_raster=r"c:\Users\gregkohler1\GIS\Lab2\Lab2\Lab2.gdb\dnr_DEM",
    value_field="ELEVATION",
)
```

Figure 12 - DEM Conversion Code

Looking at Figure 11, we can see the function LasDatasettoTin was used and ran successfully, verifying the LAS dataset was turned into TIN. Furthermore, for the TIN visualization, we can verify it was correctly converted by looking at the triangles in the image (Figure 7). These are a key part of the TIN visualization. For the DEM figure, looking at Figure 12, the function LasDatasetToRaster was used and ran successfully, showing that the LasDataset was turned into a DEM file.

Make Multidimensional Raster Layer

```
arcpy.md.MakeMultidimensionalRasterLayer(
    in_multidimensional_raster="PRISM",
    out_multidimensional_raster_layer="PRISM_MultidimLayer",
    variables="PRISM",
    dimension_def="ALL",
    dimension_ranges=None,
    dimension_values=None,
    dimension="",
    start_of_first_iteration="",
    end_of_first_iteration="",
    iteration_step=None,
    iteration_unit="",
    template='-125.020833333333 24.0624999997935 -66.4791666661985
    dimensionless="DIMENSIONS",
    spatial_reference=None
)
```

Figure 13 - Successful Multidimensional Raster Layer Code

For the third part of this lab, the first way we can verify our results is by looking at the animation generated, see Figure 10. This animation shows that the BIL files were successfully used to create a multidimensional raster layer. This raster layer successfully displays how 30-year precipitation normals change from month to month. To further verify this, we can look at the code (See Figure 13). This code is built off of earlier steps in the ETL pipeline, mainly building multidimensional info and calculating the variable and timestamp fields. This code ran successfully without error and generated the multidimensional raster layer. We can also verify the multidimensional raster layer is correct by looking at the animation and time bar, see Figure 14. This figure shows the time bar at the top, with the correct dates that correspond with the timestamp field calculated.

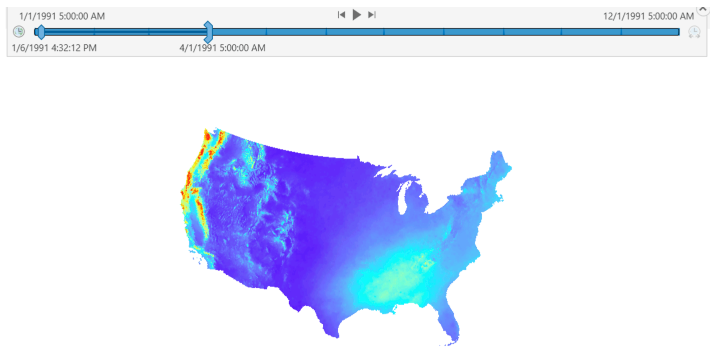

Figure 14 - Multidimensional Raster with Time Bar

Finally, we can verify that the space-time cubes were created by looking at my code, see Figure 15. We can see the use of the arcpy function CreateSpaceTimeCubeMDRasterLayer was ran successfully and exported the PRISM_CUBES.nc to the lab working folder.

Create Space Time Cubes

```
arcpy.stpm.CreateSpaceTimeCubeMDRasterLayer(
    in_md_raster="PRISM_MultidimLayer",
    output_cube=r"C:\Users\gregkohler1\GIS\Lab2\Lab2\PRISM_CUBES.nc",
    fill_empty_bins="ZEROS"
)

```

Figure 15 - Creating Space Time Cube

Discussion and Conclusion

This first part of this two-part lab helped me gain a more solid foundation when working with raster data. I have seldom worked with LAS or BIL files and this lab allowed me to explore the different file formats used to produce raster layers. This lab also helped me gain some patience with ArcGIS Pro, especially when working with animations and large raster data.

The first section of this lab was fairly easy. My first issue was figuring out how to turn the LAS file into a usable format. Once I discovered creating the LAS dataset, converting it to DEM and TIN was fairly easy thanks to the arcpy tools. When visualizing the LAS data in 2D, I got stuck on making the points visible when zoomed out. I eventually figured out how to thin the points, making it easier for my computer and ArcGIS Pro to display the point data to its full extent. With this point thinning figured out, it was simple to compare the 2D and 3D visualizations. I learned a few differences between them and how powerful a 3D visualization can be.

The last section of this part of the lab came with the most learning and new information for me. One problem was finding the correct link to the zip file that contained the BIL files, as it was not directly connected to the download button like in previous labs. Through the use of the web inspector, I was able to find it. Once I had the data, I was grateful for the (Buie, 2020) article, as it provided thorough guidance through the process of creating the multidimensional raster layer. Deciding on the timestamps to use for this layer was difficult at first, considering the data was 30-year normals for each month. I eventually decided to use the first of each month in the year when the data starts, 1991. Once generated, I found out how computationally taxing these animations can be, but it was worth the wait to see the visualization of precipitation changing as the year progresses.

In doing this lab, I discovered how versatile raster data can be. I used different raster data types to visualize the same dataset and saw how these different visualizations can be useful. I also discovered the value of using 2D and 3D layers to analyze data in ArcGIS Pro. Finally, I saw how creating a multidimensional raster layer can help us visualize how spatial data can change over time.

References

References

Buie, L. (2020, February 11). *Explore your raster data with Space Time Pattern Mining*. ArcGIS Blog; ESRI. https://www.esri.com/arcgis-blog/products/arcgis-pro/analytics/explore-your-raster-data-with-space-time-pattern-mining/

ChatGPT. (2023). *ChatGPT*. Chat.openai.com; OpenAI. https://chat.openai.com/

draw.io. (n.d.). Www.drawio.com. Retrieved October 9, 2023, from http://www.drawio.com

MN DNR. (2012, March 21). Minnesota Geospatial Commons. https://resources.gisdata.mn.gov/pub/data/elevation/lidar/

PRISM Climate Group. (2023). PRISM Climate Group; Northwest Alliance for Computational Science & Engineering. 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	**19**
		100	**99**