Lab Report

Title: Lab 3 - Part 2

Notice: Dr. Bryan Runck

Author: Mattie Gisselbeck

Date: November 30, 2022

Repository: <https://github.com/mgisselbeck/GIS5571>

Time Spent: 20 hours

## Abstract

The main objective of this lab is to build an ETL that scrapes the last 30 days of temperature data for all the NDAWN stations and compare three different interpolation methods (IDW, GPI, Kriging) using the NDAWN weather data. The data was sourced from Minnesota Geospatial Commons and was scraped through an ETL in ArcGIS Pro via a Python notebook.

Based on Dory’s preferences, the objective is to find a path that avoids farm fields, water bodies without a bridge, and has a gradual slope. The results are shown in the figures below (see Figure 2 through Figure 7). The data flow diagram above (Figure 1) shows all the variables and commands I applied in finding an optimal route for Dory. The results could be qualitatively verified by using the ‘ArcGIS Pro – Topographic’ by visually comparing the route with the hill shade of the map. In this lab, I was able to build off pre-existing knowledge with creating an ETL and building a cost path analysis model. The objectives of this lab helped me to gain practical applications of how I would create a cost path analysis through ArcPy or an open-source package.

## Problem Statement

The main objective of this lab is to build an ETL that scrapes the last 30 days of temperature data for all the NDAWN stations and compare three different interpolation methods (IDW, GPI, Kriging) using the NDAWN weather data (Runck, 2022). The analysis workflow should be able to collect real-time d

* Be able to run your notebook and create an interpolated temperature map for the highs and lows of the last 30 days from NDAWN in real-time.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Requirement** | **Defined As** | **(Spatial) Data** | **Attribute Data** | **Dataset** | **Preparation** |
| 1 | Weather Data | Average Temperature - Last 30 Days | XY Coordinates | Temperature (F) | [NDAWN Center](https://ndawn.ndsu.nodak.edu/weather-data-daily.html) | ETL |

*Table 1. Required Data*

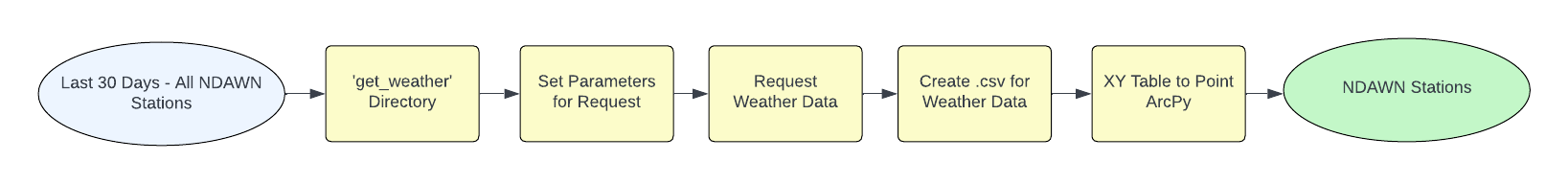
## Input Data

The table below is a collection of data from the North Dakota Agricultural Weather Network (NDAWN). Data was scraped through an ETL in ArcGIS Pro via a Python notebook. All the data described below will be used to analyze the average daily temperature recorded at all the NDAWN stations in the last 30 days.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Title** | **Purpose in Analysis** | **Link to Source** |
| 1 | NDAWN Weather (Average Temperature for the Last 30 Days) | The raw input dataset will be extracted into a .csv file to analyze the average daily temperature recorded at all the NDAWN stations in the last 30 days. | [NDAWN Center](https://ndawn.ndsu.nodak.edu/weather-data-daily.html) |

*Table 2. Input Data*

## Methods



*Figure 1.* [*Data Flow Diagram*](https://github.com/mgisselbeck/GIS5571/blob/main/Lab2/Part%202/Graphics/Lab2_Part2_DataFlowDiagram.png) *for Extracting Weather Data from NDAWN.*

### Part 2.1: Import Packages and Request Data from Minnesota Geospatial Commons

(See Lab 2 – Part 2 Python Notebook)

### Part 2.2: Create a Study Extent

To create the study extent, I used ‘Feature Class to Feature Class’ to apply a SQL expression that selects only Winona, Olmsted, and Wabasha and create a new feature class.

|  |
| --- |
| # Create Study Extent (Feature Class to Feature Class)  arcpy.conversion.FeatureClassToFeatureClass("mn\_county\_boundaries", r"C:\Users\gisse015\Documents\ArcGIS\Projects\Lab2\_2\Lab2\_2.gdb", "StudyExtent", "CTY\_NAME = 'Wabasha' Or CTY\_NAME = 'Winona' Or CTY\_NAME = 'Olmsted'", 'AREA "AREA" true true false 19 Double 0 0,First,#,mn\_county\_boundaries,AREA,-1,-1;PERIMETER "PERIMETER" true true false 19 Double 0 0,First,#,mn\_county\_boundaries,PERIMETER,-1,-1;CTYONLY\_ "CTYONLY\_" true true false 19 Double 0  # Dissolve County Boundaries  arcpy.management.Dissolve("StudyExtent", r"C:\Users\gisse015\Documents\ArcGIS\Projects\Lab2\_2\Lab2\_2.gdb\StudyExtent\_Dissolve", None, None, "MULTI\_PART", "UNSPLIT\_LINES") |

### Part 2.3: Impervious Routes

The impervious roads were extracted by mask to match the study extent. Reclassification of impervious roads was executed by using the reclassify command (See Table 3).

|  |
| --- |
| # Extract by Mask  Extract\_Roads = arcpy.sa.ExtractByMask("NLCD\_2019\_Land\_Cover\_Impervious\_Descriptor.tif", "StudyExtent\_Dissolve"); Extract\_Roads.save(r"C:\Users\gisse015\Documents\ArcGIS\Projects\Lab2\_2\Lab2\_2.gdb\Extract\_Roads")  # Reclassify Roads (Scale: 1-10) (See Table 3)  arcpy.ddd.Reclassify("Extract\_Roads", "Class\_Name", "Unclassified 10;'Primary road' 1;'Secondary road' 1;'Tertiary road' 1;'Non-road non-energy impervious' 2;'LCMAP impervious' 3;'Wind turbines' 7", r"C:\Users\gisse015\Documents\ArcGIS\Projects\Lab2\_2\Lab2\_2.gdb\Reclass\_Roads", "DATA") |

## Results

The results are shown in the figures below (see Figure 2 through Figure 7). The main themes of the lab were preparing data in an ETL pipeline, creating a cost surface to find an optimal route. The data flow diagram above (Figure 1) shows all the variables and commands I applied in finding an optimal route for Dory.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Category |  | IDW | GPI | Kriging |
| Type of Information | One Prediction per Location | ✓ | ✓ | ✓ |
| Quantile Value |  |  | ✓ |
| Many Predictions per Location |  |  |  |
| Predicted Values and Errors |  |  | ✓ |
| Measurement of Spatial Autocorrelation | Yes |  |  | ✓ |
| No |  | ✓ |  |
| Implicit | ✓ |  |  |
| Output Type | Prediction | ✓ | ✓ |  |
| Prediction Error |  |  | ✓ |
| Probability |  |  | ✓ |
| Level of Assumptions | Few | ✓ | ✓ |  |
| Intermediate |  |  |  |
| Many |  |  | ✓ |
| Type of Interpolation | Exact | ✓ |  |  |
| Inexact |  | ✓ | ✓ |
| Smoothness | Smooth |  | ✓ |  |
| Intermediate |  |  | ✓ |
| Not Smooth | ✓ |  |  |
| Uncertainty of Predicted Values | Yes |  |  | ✓ |
| No | ✓ | ✓ |  |
| Processing Speed | Slow |  |  |  |
| Intermediate |  |  |  |
| Fast | ✓ | ✓ | ✓ |

*Table 3.*

Diagram, map

Description automatically generated with medium confidence

*Figure 2. Results for Part 2.2.4: Inverse Distance Weighting (IDW) – Minimum Temperature.*

Diagram

Description automatically generated

*Figure 2. Results for Part 2.2.4: Inverse Distance Weighting (IDW) – Maximum Temperature.*

Diagram

Description automatically generated with low confidence

*Figure 2. Results for Part 2.2.4: Inverse Distance Weighting (IDW) – Average Temperature.*

*Chart

Description automatically generated*

*Figure 3. Results for Part 2.2.3: Global Polynomial Interpolation (GPI) – Minimum Temperature.*

*Chart

Description automatically generated*

*Figure 3. Results for Part 2.2.3: Global Polynomial Interpolation (GPI – Maximum Temperature.*

*Chart

Description automatically generated*

*Figure 3. Results for Part 2.2.3: Global Polynomial Interpolation (GPI) – Average Temperature.*

*Map

Description automatically generated*

*Figure 4. Results for Part 2.2.2: Kriging – Minimum Temperature.*

*Chart, surface chart

Description automatically generated*

*Figure 4. Results for Part 2.2.2: Kriging – Maximum Temperature.*

*Diagram

Description automatically generated*

*Figure 4. Results for Part 2.2.2: Kriging – Average Temperature.*

## Results Verification

The results could be qualitatively verified by using the ‘ArcGIS Pro – Topographic’ by visually comparing the route with the hill shade of the map. There are many route outputs an analyst can get based off how they chose to reclassify. Using reliable data sources like Minnesota Geospatial Commons helps to ensure quality results if used correctly.

# Discussion and Conclusion

In this lab, I was able to build off pre-existing knowledge with creating an ETL and building a cost path analysis model. The objectives of this lab helped me to gain practical applications of how I would create a cost path analysis through ArcPy or an open-source package. The biggest roadblocks for this lab were creating the cost distance output for the cost path analysis and how computationally intensive it was for my computer. In the future, I hope to apply what I learned in this lab to learn how to learn how to code using packages like numpy and rasterio.

What does the literature recommend be used for interpolating temperature data? Why? (Find one or two articles to support your claims and reference them in the lab writeup)

## References

## 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 | **27** |
| **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 | **27** |
| **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** |