**Lab Report**

Title: Decomposing interfaces and build ETL pipeline

Notice: Dr. Bryan Runck

Author: Zhaoxu Sui

Date: 10/2/2021

**Project Repository: https://github.com/KennethSui/GIS5571/tree/main/lab1**

**Google Drive Link: https://drive.google.com/drive/folders/1u6x\_nOy58I1rDu0W8drRQY0Cuax9BOgo?usp=sharing**

**Time Spent:** 7 hours

**Abstract**

This report can be split into two parts. Firstly, this report describes how the web API can be broken down into conceptual models. Secondly, this report performs tasks of getting data from Minnesota Geospatial Commons, Google Places, and NDAWN by using Jupyter notebooks, integrating them, and saving them into a single dataset.

**Problem Statement**

Web API is a common resource every time we have to check and download data. However, Jupyter notebooks can help us getting these resources by accessing these APIs programmatically. My object is to test downloading these data from three different web APIs and perform simple integrations in the Jupyter Notebook.

*Sheet 1: Required Datasets for this lab task*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Requirement** | **Defined As** | **(Spatial) Data** | **Attribute Data** | **Dataset** | **Preparation** |
| 1 | MN Weather Related Data | Meteorological Data retrieved from MN Geospatial Commons | Minnesota Climate Divisions | FID, Shape, STATE, …, Shape\_Leng,  Shape\_Area | MN Geospatial Commons | Derived from Notebook |
| 2 | MN Weather Related Data | Meteorological Data retrieved from MN Geospatial Commons | Normal Annual Precipitation Average | FID, Shape, GRIDCODE, MINVALUE, MAXVALUE, Shape\_Leng, Shape\_Area | MN Geospatial Commons | Derived from Notebook |

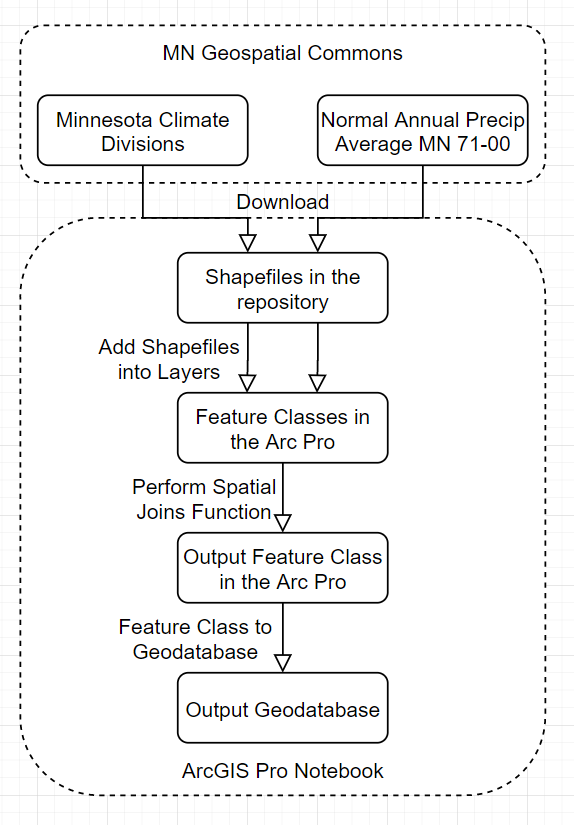
**Comparing different web APIs**

For Minnesota Geospatial Commons, developers can build searches in their own codes to get the dataset directly from the Minnesota Geospatial Commons. This website follows the framework of CKAN, with a built-in REST-ful API. For Google Places, there are a variety of filters and keywords the users can be benefited from. However, the users have to follow the Googel Maps Platform Terms of Service, with a API key to access the data. For NDAWN, the procedure will be simpler, just doing request command, and get the csv files on the webpage.

**Input Data**

*Sheet 2: Input datasets*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Title** | **Purpose in Analysis** | **Link to Source** |
| 1 | Minnesota Climate Divisions | Raw input dataset for spatial join from MN Geospatial Commons | [Mn GeoSpatial Commons](https://gisdata.mn.gov/dataset/trans-roads-mndot-tis) |
|  | Normal Annual Precipitation Average, Minnesota, 1971-2000 | Raw input dataset for spatial join from MN Geospatial Commons | [Mn GeoSpatial Commons](https://gisdata.mn.gov/dataset/trans-roads-mndot-tis) |

**Methods**

The whole process is programmed in the ArcGIS Pro notebook with the utilization of Arcpy functions.

Firstly, I will try to locate the data I want to download by using tags, and after locating the dataset, I will copy the url and try to download it through the notebook function. I will download two datasets and add them into a new map in the ArcGIS Pro through notebook.

Then, I will perform spatial joins function, in which I will utilize two feature layers imported and downloaded from the Minnesota geospatial Commons, into a new output feature layer.

Lastly, I will utilize the function “Feature Class To Geodatabase” to convert the output feature class into the default geodatabase in my working repository.

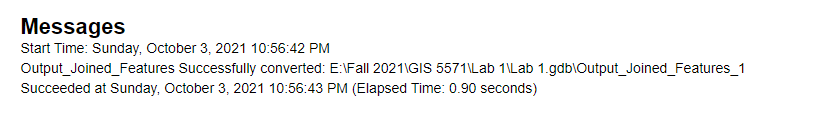
**Results**

The Output Feature Class has been successfully converted into the geodatabase, which stores the spatial joins data of both Minnesota Climate Divisions and the Normal Annual Precipitations of Minnesota 1971-2000. The whole process is performed in ArcGIS Pro Notebook, with the utilization of Arcpy functions.

**Results Verification**

The Spatial Join function is performed successfully since its new attributes are a collection of the attributes from two different classes.

The Feature Class to Geodatabase function is also performed successfully as what the messages implies.



**Discussion and Conclusion**

I learned how to access and download the data from the service API, and successfully inquired and downloaded the data from Minnesota Geospatial Commons. By using spatial join and geodatabase functions, this experiment concluded with a new merged dataset in the geodatabase.

In my future academic or career paths, I will perform some other data gathering through Google Place API, which can help me better understand the whole process of the decomposing interfaces and data pipelining.

**References**

**Self-score**

*Fill out this rubric for yourself and include it in your lab report. The same rubric will be used to generate a grade in proportion to the points assigned in the syllabus to the assignment.*

|  |  |  |  |
| --- | --- | --- | --- |
| **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 | **23** |
| **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 | **98** |