**Lab Report**

Title: Lab01-ETL Data Pipelines

Notice: Dr. Bryan Runck

Author: Megan Marsolek

Date: 10/3/2021

**Project Repository:**<https://github.com/mmarsole/GIS5571>

**Google Drive Link:** NA

**Time Spent:** 9 hours ?

**Abstract**

For this lab, I extracted data from three separate websites via python code (a.k.a. web scraped data). Each website required different extraction methods, for example, extracting shapefiles from MN Geospatial Commons involved using CKAN and requests package, while data from Google Places required an API key (needed to make an account to attain the key). I found NDAWN the easiest to understand and extract data from amongst all three websites.

After extracting data, I then spatially joined the two shapefiles from MN Geospatial Commons using python code. I corroborated the results by spatially joining the same data within ArcPro GUI. I found it easier to spatially join the data via open-sourced packages instead of Esri’s ArcPy, since I am still unfamiliar with its functions and syntax.

**Problem Statement**

Acquiring data from three different websites (NDAWN, MN Geospatial Commons, and Google Places) by coding with Python ETLS to extract the data directly from the web. You’ll notice based on Table 1 that in all I extracted four separate datasets, two shapefiles from MN Geospatial commons, and html based data that was scraped directly from the website. From NDAWN I extracted data within a table and converted it to a csv, while from Google I extracted ‘place details’ that was then stored as a dictionary.

*Table 1. Data Scraped via ETL Pipeline*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Requirement** | **Defined As** | **(Spatial) Data** | **Attribute Data** | **Dataset** | **Preparation** |
| 1 | **MNDNR Native Plant Communities** | Shapefiles | Polygon geometry | NPC\_descri (gives the plant description for each observation) | [Mn GeoSpatial Commons](https://gisdata.mn.gov/dataset/trans-roads-mndot-tis) | Import ‘requests’ package |
| 2 | **State Parks, Recreation Areas, and Waysides** | Shapefiles | Polygon geometry | UNIT\_NAME (name of park), UNIT\_TYPE (classification for type of public area) | [Mn GeoSpatial Commons](https://gisdata.mn.gov/dataset/trans-roads-mndot-tis) | Import ‘requests’ package |
| 3 | **ND Current Weather observations** | A table from a html website (converted to csv) | None (each station name did indicate what city within ND each measurement was from) | Station, Data of Acquisition, air temp, Wind direction, Current and Peak Wind Speed, Relative Humidity | From: [NDAWN](https://ndawn.ndsu.nodak.edu/current.html) | Import ‘requests’ and ‘BeautifulSoup’ package |
| 4 | **Bockley Art Gallery google reviews and ratings (‘place details’)** | Information pulled form google maps (loads as a dictionary using ‘requests’ package) | Written address | Ratings, reviews, address, phone number, etc. | From:  [Google Maps](https://www.google.com/maps/place/Bockley+Gallery/@44.9616569,-93.3102897,17z/data=!3m1!4b1!4m5!3m4!1s0x52b33325630e18c3:0xfeed41971150cce!8m2!3d44.9616764!4d-93.3081006) | Created a Google places account to access an API key |

**Input Data**

For this assignment we also had to create a spatial join and then export the data to a GDB (GeoDataBase). Using the Shapefiles from MN Geospatial Commons, I spatially joined MN State Parks to MN Plant Communities. The intention was to produce a subsetted version of the Plant Communities Data that only retained Plant observations that occurred within MN State Parks and indicated what park they belonged to with a new attribute column (UNIT-NAME). Please see Table 2 for further details about the data used and acquired.

*Table 2. Data used in Spatial Join*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Title** | **Purpose in Analysis** | **Link to Source** |
| 1 | MN Parks | An additional attribute added to the Plant Communities data that will tell us within what park each plant observation occurred | [Mn GeoSpatial Commons](https://gisdata.mn.gov/dataset/bdry-dnr-lrs-prk) |
| 2 | MN Plant Communities | Subsetted to include only plant observations within MN State Parks | [Mn GeoSpatial Commons](https://gisdata.mn.gov/dataset/biota-dnr-native-plant-comm) |

**Methods**

*Figure 1: Python Extraction*

Diagram

Description automatically generated

**Results**

*Image 1: State Wide view of Spatially Joined data and a Zoomed in view to the right*

Map

Description automatically generated Map

Description automatically generated

*Image 2: Initial inspection of csv converted NDAWN data*

Table

Description automatically generated

*Image 3: Inspection of the Places details for Bockley Art Gallery (formatted as a dictionary)*

Graphical user interface, text, application, email

Description automatically generated

In all, I managed to extract and save a csv version of some of the NDAWN weather data, store the reviews for Bockley Art Gallery, extract shapefiles via python ‘requests’, spatially join the shapefiles, and save the data to a GDB.

**Results Verification**

I performed the spatial join within Jupyter Notebooks using ‘.sjoin()’ function, and visually inspected the output (using folium maps), I further performed the Spatial Join within ArcPro GUI and found visual results upon inspection.

For verification on the extraction of data I compared the output for the NDAWN data to what was present on the page, and via this cursory glance I found no obvious errors (page updated every five minutes, which maked it hard to do more than a cursory glance). The same visual comparison was performed on the data I extracted about Bockley Art Gallery (I looked at the Google Maps to compare reviews, and other extracted data).

**Discussion and Conclusion**

Overall, I found the most challenging part of this assignment trying to use Arcpy functions to spatially join data. I am not familiar with this package’s format, syntax, or its function names, and have trouble using it when I am more familiar with other open-sourced packages that can perform the same functions (all except export data to a GBD, my research found that since this is an Esri proprietary product I cannot save my shapefiles using open source means).

Otherwise, the next biggest challenge was learning to extract data from any web source via coded ETLs. This is the first time I have ever built or coded such commands, and I am grateful for open-sourced tutorials that guided me through extracting from open domain websites (Goggle was trickier since you have to use an API key). I still found it very intimidating, I don’t know JavaScript, inspecting a webpage takes time to locate my desired data, and CKAN still feels awkward, given I don’t know its other functions or attributes.

I think with time I might get better at ETL Pipeline coding, but recognize I am severely hampered by my lack of understanding and experience in JavaScript, API keys, and general extraction packages (‘requests’, ‘BeautifulSoup’, etc.)

**References**

Clever Programmer'. (2019, April 4). *20 - web scraping with python using beautiful soup & requests (Python tutorial for beginners 2019)*. Www.youtube.com. https://www.youtube.com/watch?v=E5cSNSeBhjw

**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 | **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 | **22** |
| **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 | **94** |