Discovering Colorado’s Freshwater Emergent and Forested Wetland Areas by Size and County

Abstract:

This project finds and focuses on freshwater emergent and forested wetland types within all counties of the state of Colorado. This project will use python – a programming language to determine which selected size categories of wetlands are nearest to state highways, and which of the specific wetland areas intersect state highways.

Keywords: *Wetlands, Colorado, Wetland Type, Freshwater, Emergent, Forested, State Highway*

Introduction:

Wetlands are some of nature’s most valuable ecosystems. Wetlands provide crucial ecosystem services such as flood protection, water quality improvement, erosion control, recreation, and provide habitat for thousands of species. Unfortunately, more than half of these ecosystems within the United States have been lost since 1780 (NWI, 2024). To conserve and protect these ecosystems and habitats the National Wetlands Inventory was founded by the U.S Fish and Wildlife Service so that the distribution and other attribute information could be provided to biologists around the nation.

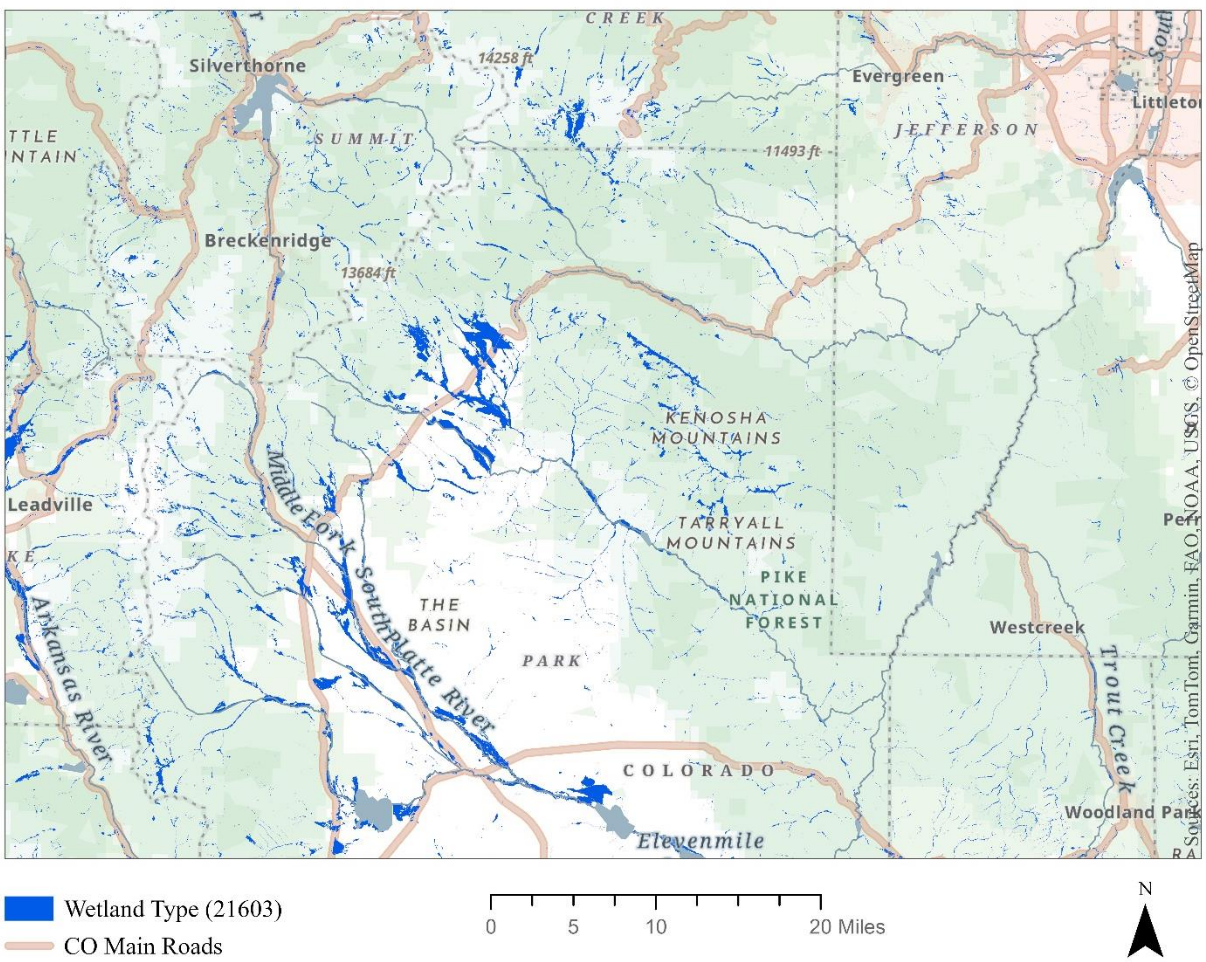
The United States Environmental Protection Agency provides the classification and types of wetlands that is utilized by the U.S Fish and Wildlife Service and the U.S Army Corps of Engineers. However, it is important to note that each of these agencies utilize a different classification system. For instance, the U.S Fish and Wildlife Service utilized the “Cowardin System”. The Cowardin system bases classification of wetlands by their position on the landscape, vegetation, and hydrologic regime (discharge or flow of related water system). This system has five major wetland types: marine, estuarine, lacustrine, palustrine, and riverine (EPA, 2024). The U.S Army Corps of Engineers on the other hand, utilizes the “Brinson System”. The Brinson system refers more to a wetlands dominant water source (precipitation, groundwater, etc.) and hydrodynamics. This system categorizes wetlands as riverine, slope, depressional, flat, and fringe.

The state of Colorado is known as a typically dryer state with the average yearly rainfall being approximately 17 inches. However, it is important to note that this precipitation is highly varied due to the rouged terrain and altitude (Doesken, et al 2003). Colorado is not known for having a significant number of wetlands, but it does matter as the state is at risk for more drought conditions. Therefore, the wetlands that the state does have need to be protected to obtain the full and necessary benefits from this ecosystem type.

This project focuses on identifying freshwater emergent and forested/shrub wetlands within the state of Colorado. Colorado has a vast array of wetland types including freshwater pond, riverine, and lake. It is important to note that these names are based off the classification attributes, not wholly the Cowardin classification system. For example, "Lake" wetland types here refer to lacustrine wetland types.

Plant data is collected from a fen that sits at 11,000 feet near Mount Emmons on the Gunnison National Forest in Colorado. (U.S. Forest Service)

Freshwater Emergent and Forested Wetland type near Denver Colorado



(Figure 2: Map above shows areas that are classified across the landscape near Denver, Colorado).

Freshwater emergent wetlands are typically categorized as a transitional area between water and land. The name "emergent" refers to the fact that these wetlands are non – tidal. These wetland types are typically dominated with vegetation such as emergent plants, trees, and shrubs. Palustrine translates to living in a "marshy" environment. These palustrine wetlands are shallow and smaller in size and are seasonally flooded or permanently flooded (EPA, 2024). Forested wetland areas can also be palustrine wetlands, which is why these two types were chosen for this assignment. As the name suggests, forested wetland areas would include more trees than other vegetation.

The last portion of this project is to identify wetlands that are intersected by roads, or within a half mile of main roads. The largest threats to wetlands are degradation, pollution, invasive species, and disease (Kingsford, et al 2016). It is also important to consider that the fragmentation of the landscape due to agriculture expansion can divert wetland flow patterns to protect irrigated crops. Wetlands that become polluted with pesticides or run off can harbor an excessive amount of nutrients that can adversely affect the native biodiversity (Kingsford, et al 2016).

Materials and Methods:

Data Collection:

Data for this project was collected from the National Wetlands Inventory (NWI), run by the U.S Fish and Wildlife Service. The NWI provides and produces maps and geospatial data on wetland and deepwater habitats within the United States. The data that is collected helps provide information that supports numerous analyses related to research, land management planning, policy decisions, and mitigation efforts. Colorado is the state of interest for this project due to its rapidly expanding urban development that puts extra stress on water availability and management. This project uses a shapefile of all the Colorado wetlands from the east and western halves of the state. The shapefile data from the NWI also provides the type of wetlands, its classification, and size in acres. The second data source this project utilizes are main road and county boundary layers from the U.S Census Bureau. These data layers will provide the project with the ability to run geoprocessing tools to extract the desired attribute information.

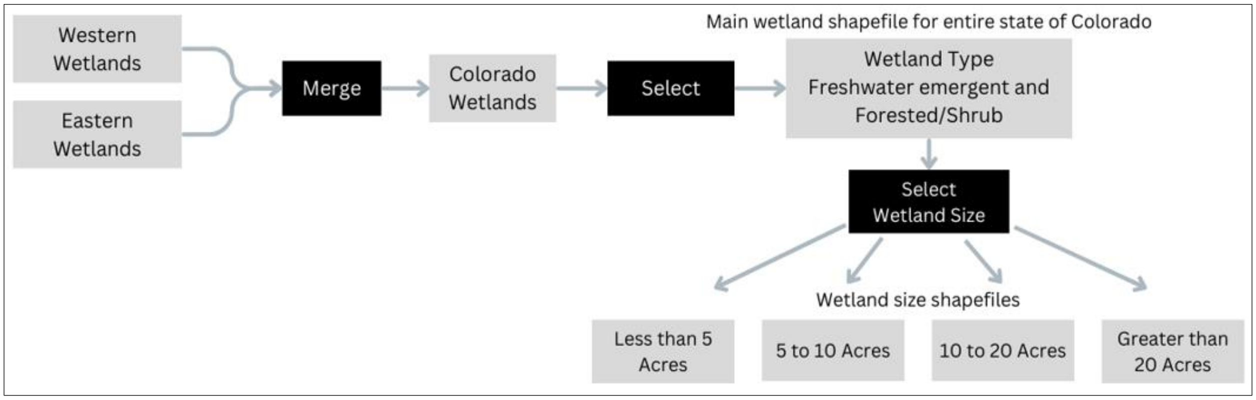
This project will utilize jupyter notebook for containing the code documentation for the various geoprocessing steps and will be used within ArcGIS Pro. Python IDLE will also be used to quickly create and import modules that are relevant to the user.

Methods:

<u>Step 1:</u>

The shapefile layers from the NWI consists of hundreds and thousands of polygons. The NWI also splits the state wetlands into western and eastern halves. The wetlands need to be merged into a single layer so it will not interrupt the later geoprocessing steps of clipping to a specific Colorado county of interest. This process will create a "Colorado Wetlands" layer, however the dataset at this point is incredibly large which would slow processing times. To extract the data of interest the select tool will be used to create a shapefile layer that consists of polygons that only have the attribute information regarding freshwater emergent wetlands, and forested shrub wetlands. These first geoprocessing steps are illustrated in the flow figure below.

(Figure 3: Separating and extracting data layers)

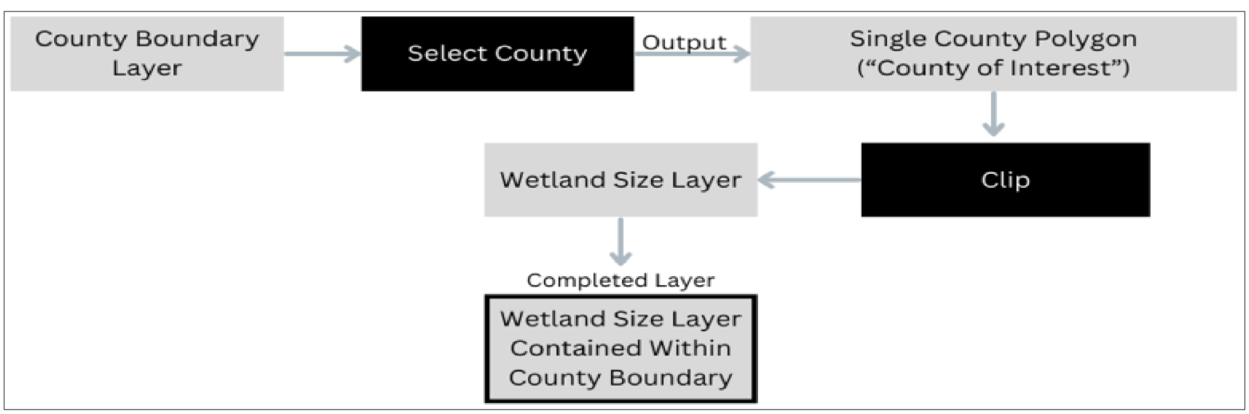


By creating different size layers, it allows analyzing steps to be more organized and processing times to be increased. If the user is using a map view, it also allows for the data to be represented visually in different ways and allows for future manipulation of layers if desired.

Step 2:

The next step of geoprocessing will consist of incorporating the tiger line shapefiles from the U.S Census Bureau. To obtain a list of county names the user can implement in later processing steps; a search cursor will be implemented within python. Once a list has been created, the select tool will again be used to select a specific county within the "Colorado County Boundary" shapefile layer. This will then create an output shapefile layer that shows only the county of interest. This new layer can then be used with wetland size category layers created in step 1. Once a county and a wetland size of interest are chosen, the clip tool will be implemented to show only the wetland size of interest within the county of interest.

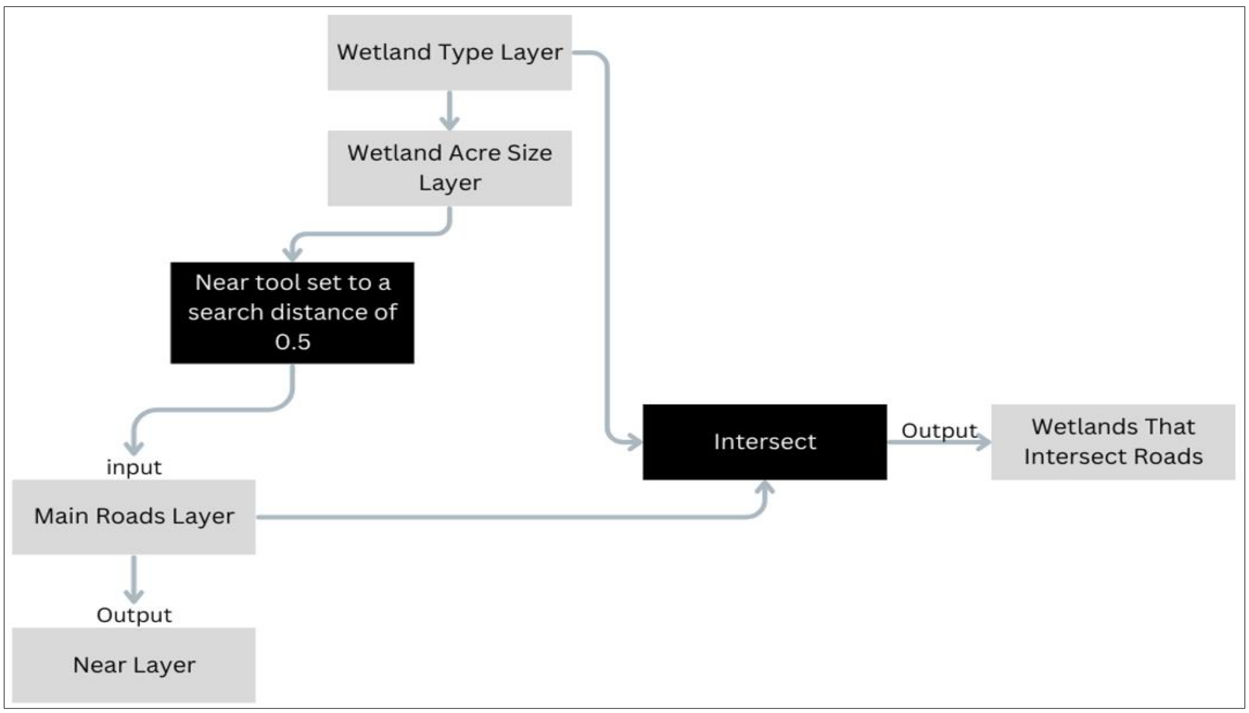
(Figure 4: Separating and clipping data layers from two different data sources)



Step 3:

The final step of the geoprocessing components will be implementing the near tool with the acre size category shapefile layers created in step 1. The goal of this step is to identify wetlands of a particular size that are near major roads, and counties of interest. This step will also implement the intersect tool but will use the "Wetland Type" layer created in step 1. There are two reasons why this layer is chosen instead of the size category layers. The first reason is the output dataset from this tool is more manageable in size (smaller). Secondly, this layer is chosen so that the original dataset can be manipulated and utilized in different ways for future use and analysis.

(Figure 4: Further geoprocessing to extract related information of interest)



Results:

This project focused on the Colorado county of Larimer as an example of how this data can be processed. However, the data could be run with any county of interest with this method being applied to other states if needed. Using the merge tool to merge the datasets provided to be useful as many of the counties (including Larimer) would have been split between western and eastern wetlands. While it may have not been as necessary to run the select tool four times for the wetland acre size layers, it still proved to be beneficial to processing times and the visual display of data if using map view.

Within the code that was ran for the geoprocessing elements, conditional statements were implemented to provide information to the user on successful processing. The figure below shows an example of how this code was stored and organized with the given output. When running geoprocessing elements, it is important to add user input and provide an output using conditional statements. For example, if the user had an input of yes and the code ran successfully the output would let the user know that selecting the county of interest was completed. This is important when considering the time documentation of the code and to ensure that the code can be copied and have other elements added.

(Figure 5: Example of a *select* geoprocessing code with a conditional statement)

```python
#After printing the list of counties from the search cursor the user can clip the data layer to county of interest

import arcpy

#Set workspace
arcpy.env.workspace = r"C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19"

Select = input("Perform select yes or no: ")


#Set local variables

in_features = "Colorado_County_Boundaries"
out_feature_class = r"C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\Larimer"

#Please only change the county name itself
where_clause = "COUNTY = 'LARIMER'"

if Select == "yes":
    print("Selecting county . . . ")
elif Select == "no":
    print ("No clip performed.")
else:
    print ("Check user input.")


arcpy.analysis.Select(in_features, out_feature_class, where_clause)
```

```
Perform select yes or no: yes
Selecting county . . .
```

Geoprocessing shapefiles and data layers requires file organization to be clear and concise. The file paths and names for this project are stored within modules so the user can refer to different layers as needed. Providing the paths to the folders in module format allowed for quicker manipulation of data layers, especially for the wetland size categories.

(Figure 6: Utilizing modules to store information)

```python
# import modules that contain the shapefile layers for wetland size categories

import FiveAcresorLess
FiveAcresorLess.FiveAcresorLess_path()

import FivetoTenAcres
FivetoTenAcres.FivetoTenAcres_path()

import TentoTwentyAcres
TentoTwentyAcres.TentoTwentyAcres_path()

import GreaterthanTwentyAcres
GreaterthanTwentyAcres.GreaterthanTwentyAcres_path()
```

```
C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\Less than or Equal to 5 Acres.shp
C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\5 to 10 Acres.shp
C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\10 to 20 Acres.shp
C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\Greater than 20 Acres.shp
```

A cursor tool was implemented to obtain row information from the shapefile data. As mentioned previously, this would be useful if one did not have access to a map view or wanted to record the row data within the notebook for documentation purposes.

(Figure 7: Implementation of a cursor to pull data from tables within shapefile layers, some of the output data is posted below.)

```
import arcpy

#Select feature classes of interest

FeatureClass1 = r"C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\Colorado_County_Boundaries.shp"

#Ask user input for search area interest
Field_name1 = "COUNTY"

#Create a search cursor for both FeatureClass1 and FeatureClass2

with arcpy.da.SearchCursor(FeatureClass1, [Field_name1]) as cursor1:
    for row in cursor1:
        print(row[0].format(row[0], = ))
```

```
LARIMER
LAS ANIMAS
FREMONT
GUNNISON
CONEJOS
EAGLE
```

(Figure 8: Shows converting the cursor list to a string list for organization.)

```
county_list = ['LARIMER', 'LAS ANIMAS', 'FREMONT', 'GUNNISON', 'CONEJOS', 'EAGLE', 'OTERO',
              'LA PLATA', 'SUMMIT', 'CUSTOR', 'PITKIN', 'CROWLEY', 'CHEYENNE', 'BENT', 'ADAMS', 'ELBERT',
              'YUMA', 'LAKE', 'DELTA', 'COSTILLA', 'GARFIELD', 'MORGAN', 'PROWERS', 'MONTEZUMA',
               'MINERAL', 'LINCOLN', 'JEFFERSON', 'RIO BLANCO', 'SEDGWICK', 'SAN MIGUEL', 'ALAMOSA', 'PHILLIPS',
               'OURAY', 'MESA', 'SAGUACHE', 'DOUGLAS', 'DOLORES', 'RIO GRANDE', 'PUEBLO', 'KIT CARSON', 'BACA',
               'GRAND', 'LOGAN', 'CLEAR CREEK', 'MOFFAT', 'TELLER', 'BOULDER', 'KIOWA', 'CHAFFEE', 'HINSDALE',
               'JACKSON', 'WELD', 'SAN JUAN', 'MONTROSE', 'BROOMFIELD', 'WASHINGTON', 'ROUTT', 'ARCHULETA',
               'GILPIN', 'DENVER', 'PARK', 'EL PASO', 'ARAPAHOE', 'HUERFANO']

CO_counties = ", ".join(county_list)
print(CO_counties)
```

```
LARIMER, LAS ANIMAS, FREMONT, GUNNISON, CONEJOS, EAGLE, OTERO, LA PLATA, SUMMIT, CUSTOR, PITKIN, CROWLEY, CHEYENNE, BENT, ADAM
S, ELBERT, YUMA, LAKE, DELTA, COSTILLA, GARFIELD, MORGAN, PROWERS, MONTEZUMA, MINERAL, LINCOLN, JEFFERSON, RIO BLANCO, SEDGWIC
K, SAN MIGUEL, ALAMOSA, PHILLIPS, OURAY, MESA, SAGUACHE, DOUGLAS, DOLORES, RIO GRANDE, PUEBLO, KIT CARSON, BACA, GRAND, LOGAN,
CLEAR CREEK, MOFFAT, TELLER, BOULDER, KIOWA, CHAFFEE, HINSDALE, JACKSON, WELD, SAN JUAN, MONTROSE, BROOMFIELD, WASHINGTON, ROU
TT, ARCHULETA, GILPIN, DENVER, PARK, EL PASO, ARAPAHOE, HUERFANO
```

The implementation of the “where” clause allows the user to extract information from a shapefile layer that is specific to that layer. In this example, the shapefile layer was wetlands that were greater than twenty acres and located in Larimer. It is therefore important to note that the “where” clause allowed the user to search for wetlands greater than or equal to fifty acres. This step is important to note because it is processing the data down to a significantly smaller set and is specific to user search.

(Figure 9: Shows selecting the layer by attribute tool to obtain data information using a where clause statement.)

```python
import arcpy

#Set workspace
arcpy.env.workspace = r"C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\20AcresLarimer.shp"
arcpy.env.overwriteOutput = True

#further select layers from previous "20 Acres_Larimer"


arcpy.management.SelectLayerByAttribute(
in_layer_or_view = "20Acres_Larimer",
selection_type = "NEW_SELECTION",
where_clause = "ACRES >= 50.0",
invert_where_clause = None)

print(arcpy.GetCount_management("20Acres_Larimer"))
```

```
102
```

(Figure 10: A completed geoprocessing example that utilized processed layers and a search cursor to display the desired data in a list format; FID, wetland type and acre size within a county.)

```python
import arcpy

#Select feature classes of interest

FeatureClass1 = r"C:\Users\chris\OneDrive\Documents\ArcGIS\Projects\MyProject19\20Acres_Larimer.shp"

#Ask user input for search area interest
Field_name = "FID"
Field_name1 = "WETLAND_TY"
Field_name2 = "ACRES"


#Create a search cursor for both FeatureClass1 and FeatureClass2

with arcpy.da.SearchCursor(FeatureClass1, [Field_name, Field_name1, Field_name2]) as cursor:
    for row in cursor:
        Field_name = row[0]
        Field_name1 = row[1]
        Field_name2 = row[2]

        print(Field_name, ' ', Field_name1, ' - ',Field_name2)
```

```
0   Freshwater Forested/Shrub Wetland  -  42.4933579239
1   Freshwater Forested/Shrub Wetland  -  42.0617772081
2   Freshwater Emergent Wetland  -  80.6247895301
3   Freshwater Emergent Wetland  -  32.7567889894
4   Freshwater Emergent Wetland  -  22.4255089728
5   Freshwater Emergent Wetland  -  36.6902177386
6   Freshwater Emergent Wetland  -  80.1302902445
7   Freshwater Forested/Shrub Wetland  -  21.9250034434
8   Freshwater Forested/Shrub Wetland  -  42.3185318856
9   Freshwater Forested/Shrub Wetland  -  228.288276873
10   Freshwater Forested/Shrub Wetland  -  74.0075256724
```

(Figure 11: Map view of desired processing extent.)

Discussion:

Identifying wetlands is of critical importance, especially in a more arid climate. The wetlands in these regions provide crucial habitat for a multitude of species and provide necessary ecosystem functions for humans. As fragmentation of the landscape continues, wetlands need proper identification and documentation. By utilizing this data within ArcPro and other related geospatial software, the user should quickly be able to organize, sort, filter, and analyze Colorado's wetland data. By accounting for acre size, type, and relation to major roads, the user can inform policy makers, stakeholders, landowners, or organizations.

Utilizing the jupyter notebook within ArcPro provides a useful way to understand how geoprocessing elements are used. Arcpy (Python package) can be used to automate and analyze geographic data in ways that some tools cannot achieve and would therefore need to be created. This project focused on beginner code documentation that is necessary to run some basic geoprocessing tools to understand spatial relationships. This project also aimed to use several arcpy functions to list and organize data that is related to several common geoprocessing tools.

It should be noted that running the intersect for this project is likely not as useful as the near geoprocessing tool. Several of the polygons representing various wetlands are cut at the main road's shapefile naturally as these polygons likely originate from raster data. By adjusting the near distance to be smaller than a half mile, one is likely to get a much more accurate representation of intersecting wetlands although near will suffice in most cases. Another important consideration specific to this project is that user input was attempted to be implemented when "selecting" a county layer. If this code can be completed in other projects it would allow for smoother, faster, and more user-friendly automation of the code by providing a search within the county list.

References:

Doesken, N. J., Pielke, R. A., Bliss, O. A., & Center, C. C. (2003). *Climate of Colorado.* Accessed on December 4th, 2024. https://api.mountainscholar.org/server/api/core/bitstreams/15aca9ba-1a96-4aa7-bd3d-da80d310c9a3/content

EPA, United States Environmental Protection Agency (2024). *Classification and Types of Wetlands*. Accessed on December 1, 2024. https://www.epa.gov/wetlands/classification-and-types-wetlands#undefined

Kingsford, T. R., Basset, A., & Leland, J. (2016) Wetlands: conservation's poor cousins: *Aquatic Conservation: Marine and Freshwater Ecosystems.* 26(5), 892-916 https://doi.org/10.1002/aqc.2709

NWI, National Wetlands Inventory. (2024) *About us.* Accessed on December 1, 2024. https://www.fws.gov/program/national-wetlands-inventory/about-us

Link to photo on page 2:

https://www.usda.gov/media/blog/2014/07/15/rocky-mountain-wetland-provides-fen-tastic-habitat-high-altitude-plants

This work can be shared and used as an example for future courses.