

# Early findings from a cross-domain need-finding study with users of geospatial data

# Programming Systems Seminar // April 7, 2022

## UC Berkeley EECS

Parker Ziegler // @parkie-doo

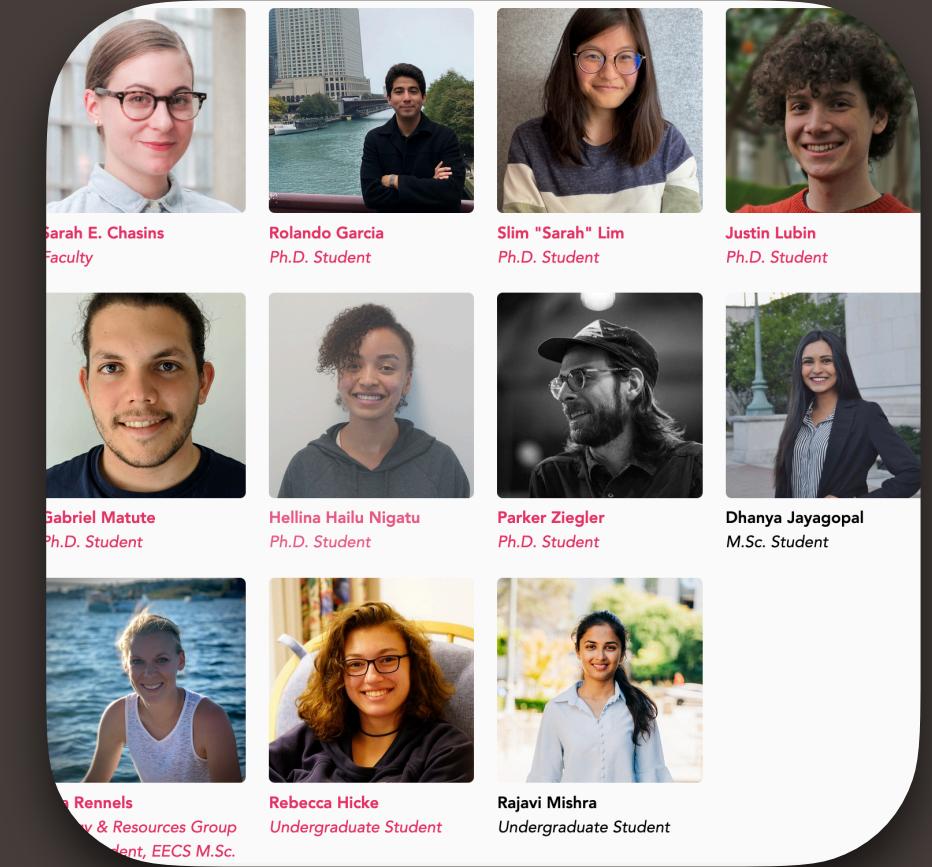
# Acknowledgments



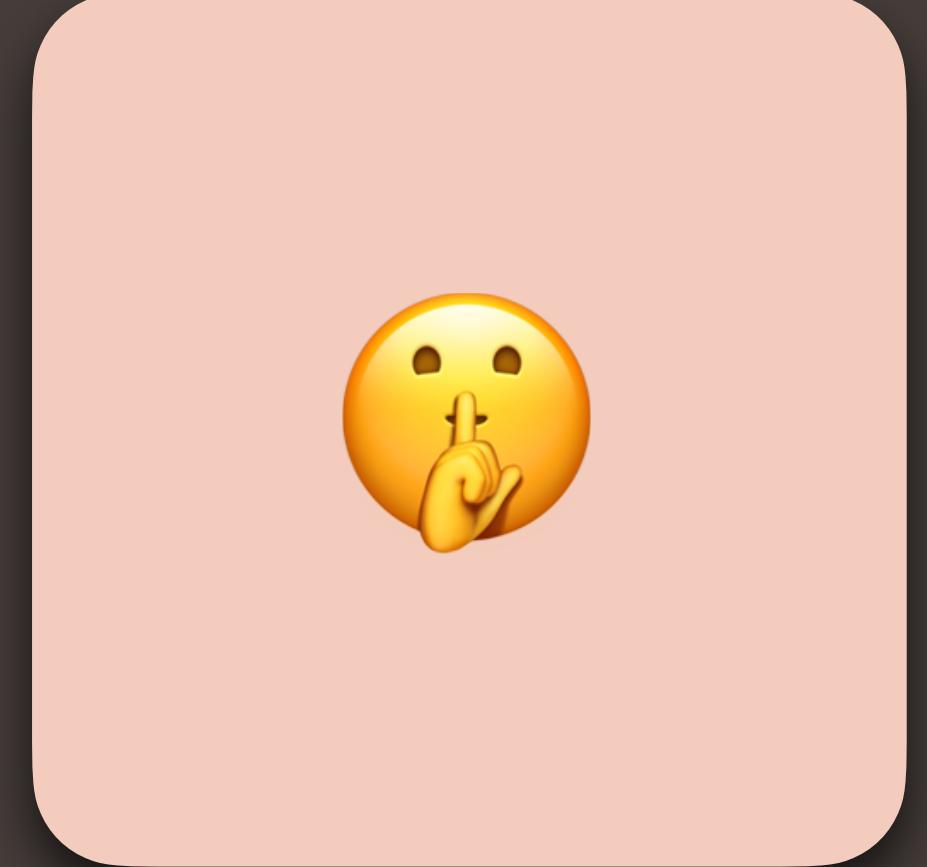
**Sarah E. Chasins**  
Ph.D. Advisor,  
Principal Investigator



**Rachel Leven**  
CDSS Media  
Communications  
Specialist



**PLAIT Lab**  
Kind, thoughtful friends  
who let me rant about  
maps, KEXP, and the Red  
Sox



**My Participants**  
Computing tools are  
built for those who  
participate!

# A Note on Feedback

1. Put on your {CHI, UIST, OOPSLA, PLDI} reviewer hat!
2. Pay special attention to the **Motivation** section

# What's this talk about?

Sharing **preliminary findings** from a **need-finding study** with users of geospatial data.

25

participants

3

domains of  
interest

30hrs 16mins

of video from observational  
interviews

# Roadmap



# Roadmap



# What is geospatial data?

GEOID	STATE	TOTAL_POP	NUM_SHOOTINGS	TOTAL_AREA
05	AR	3,011,524	27	134,770.0
27	MN	5,706,494	37	206,233.0
32	NV	3,104,614	34	284,331.5
09	CT	3,605,944	9	12,540.7
51	VA	8,631,393	41	102,278.6
22	LA	4,657,757	49	111,897.8
41	OR	4,237,256	34	248,607.8
18	IN	6,785,528	41	92,788.9
13	GA	10,214,860	69	148,958.0
39	OH	11,799,448	64	105,829.5
02	AK	733,391	13	1,477,953.4
49	UT	3,271,616	18	212,819.3
33	NH	1,377,529	5	23,188.2
21	KY	4,505,836	41	102,268.3
35	NM	2,117,522	43	314,160.4

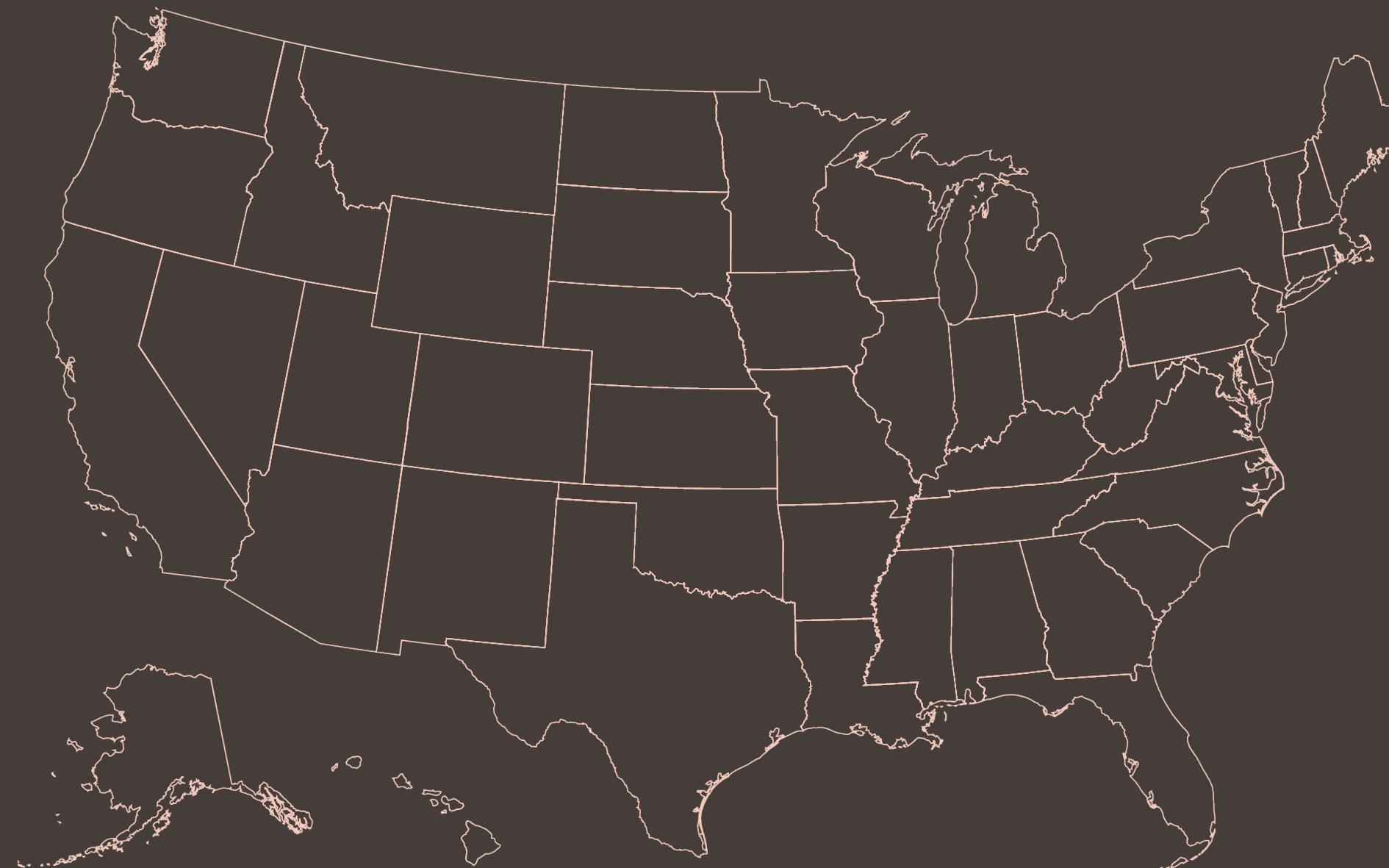
*Tabular Data*

# What is geospatial data?

Data in which **location** is encoded alongside **attributes**

GEOID	STATE	TOTAL_POP	NUM_SHOOTINGS	TOTAL_AREA
05	AR	3,011,524	27	134,770.0
27	MN	5,706,494	37	206,233.0
32	NV	3,104,614	34	284,331.5
09	CT	3,605,944	9	12,540.7
51	VA	8,631,393	41	102,278.6
22	LA	4,657,757	49	111,897.8
41	OR	4,237,256	34	248,607.8
18	IN	6,785,528	41	92,788.9
13	GA	10,214,860	69	148,958.0
39	OH	11,799,448	64	105,829.5
02	AK	733,391	13	1,477,953.4
49	UT	3,271,616	18	212,819.3
33	NH	1,377,529	5	23,188.2
21	KY	4,505,836	41	102,268.3
35	NM	2,117,522	43	314,160.4

*Attributes*



*Geometry*

# What is geospatial data?

Data in which **location** is encoded alongside **attributes**



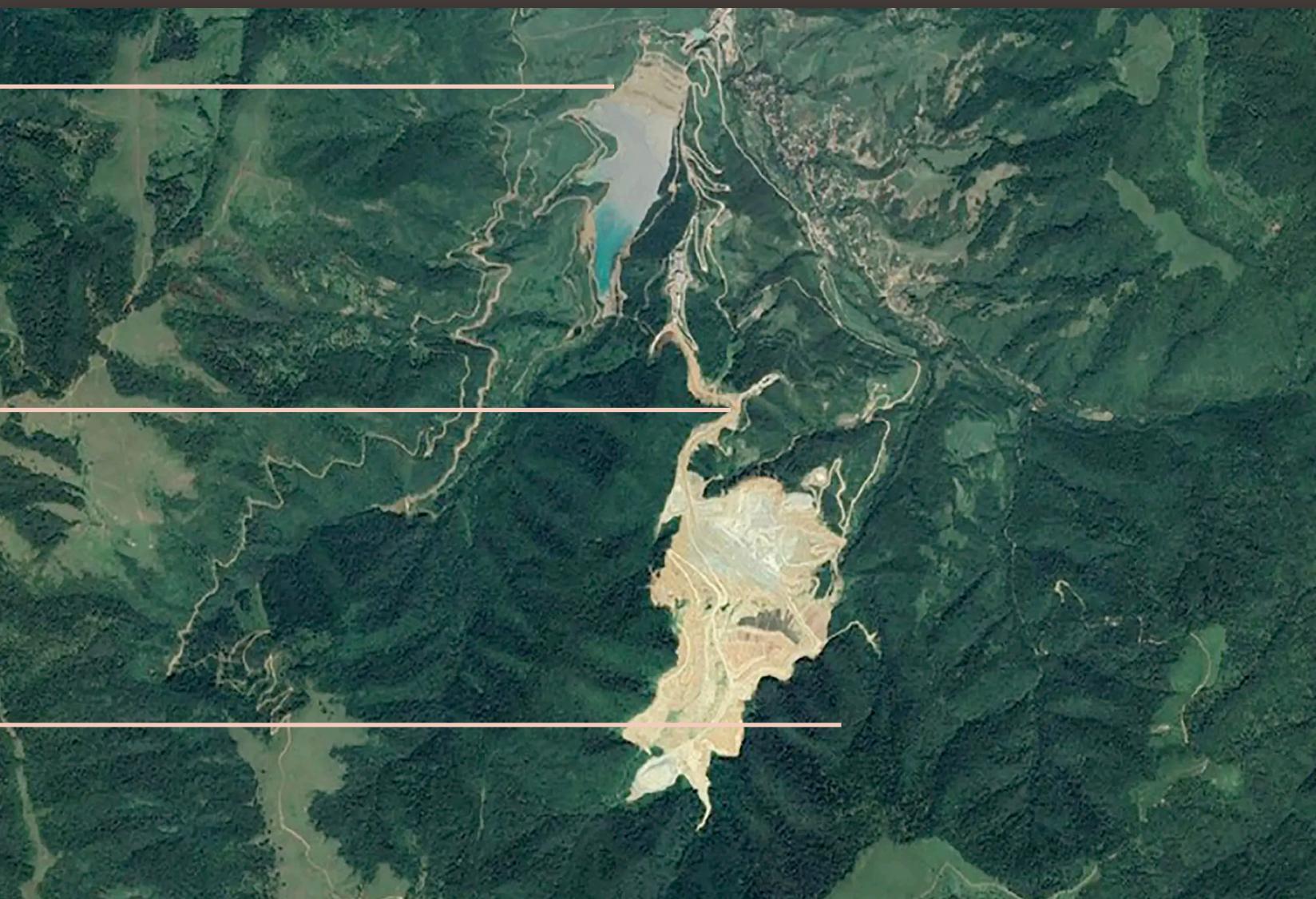
B4: 106, B3: 121, B2: 102



B4: 193, B3: 181, B2: 154



B4: 40, B3: 60, B2: 58



*Attributes*

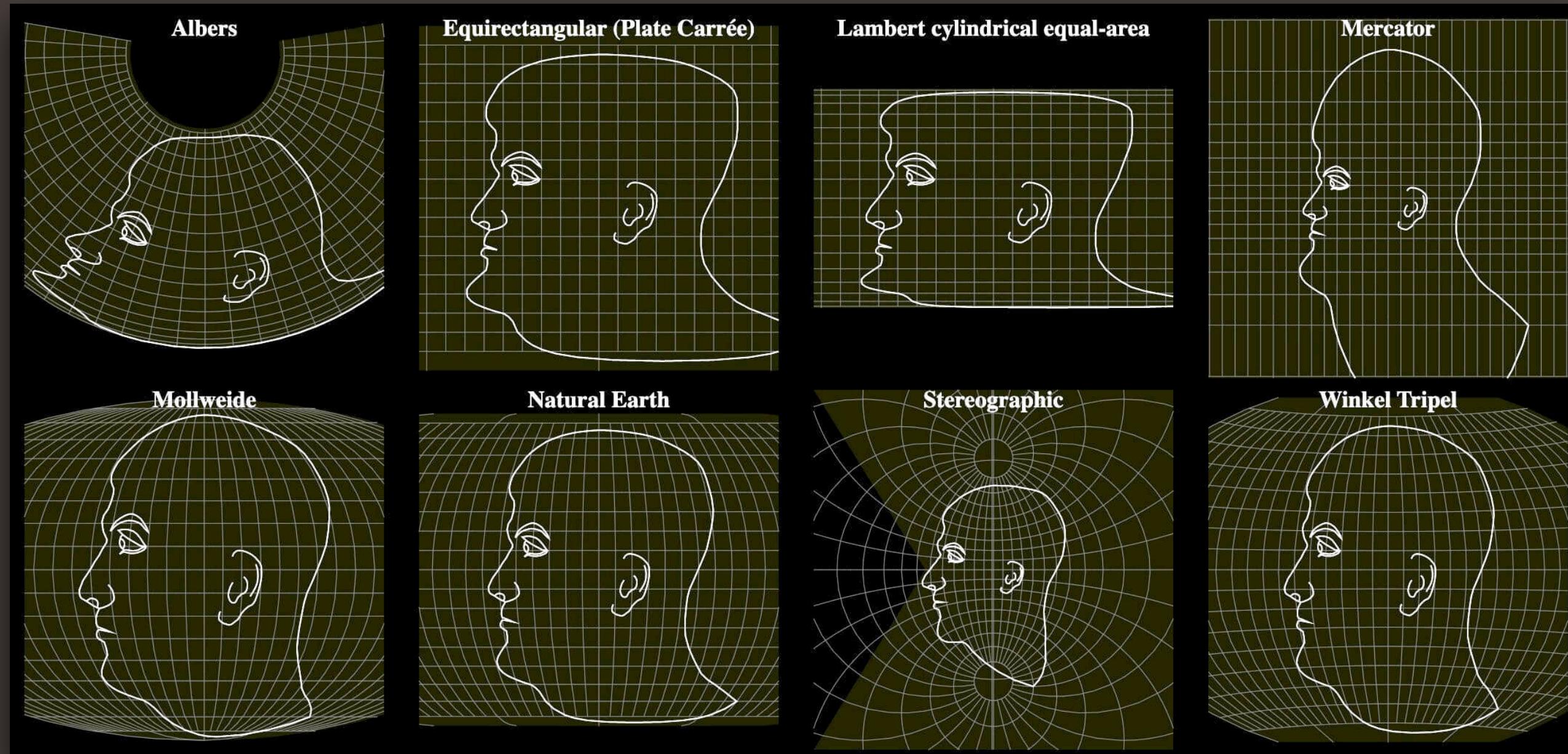
bbox: [  
[44.44, 40.49],  
[44.62, 40.45],  
[44.80, 40.49],  
[44.62, 40.53]]

*Geometry*

# What is geospatial data?

**Location** is encoded in reference to a **coordinate system**

All coordinate systems trade-off between **size**, **shape**, and **distance**



Andy Woodruff's projected heads, based off Charles Deetz's 1921 book  
*Elements of Map Projection with Applications to Map and Chart Construction*

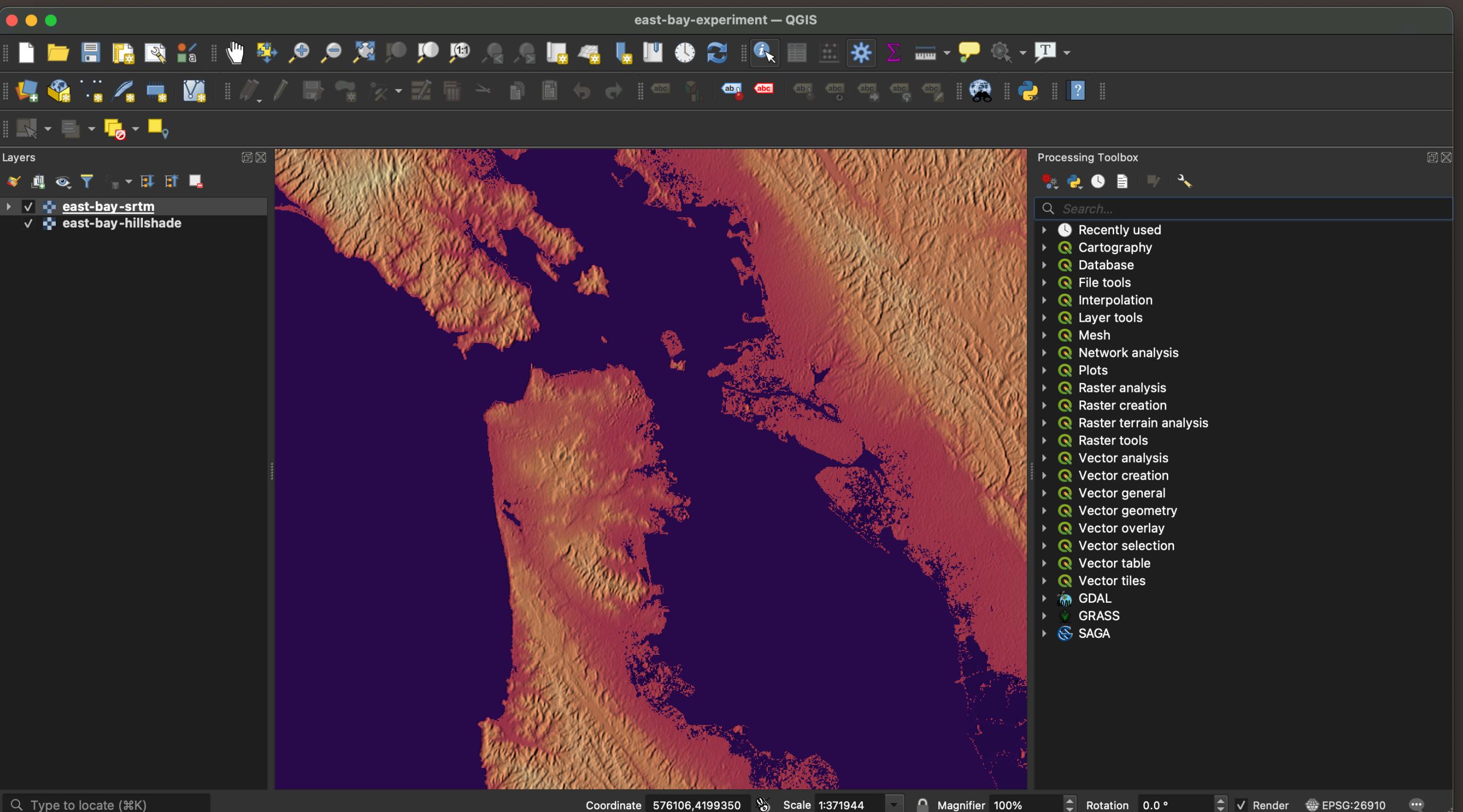
What is GIS?

Geographic  
Information  
System

# What is GIS?

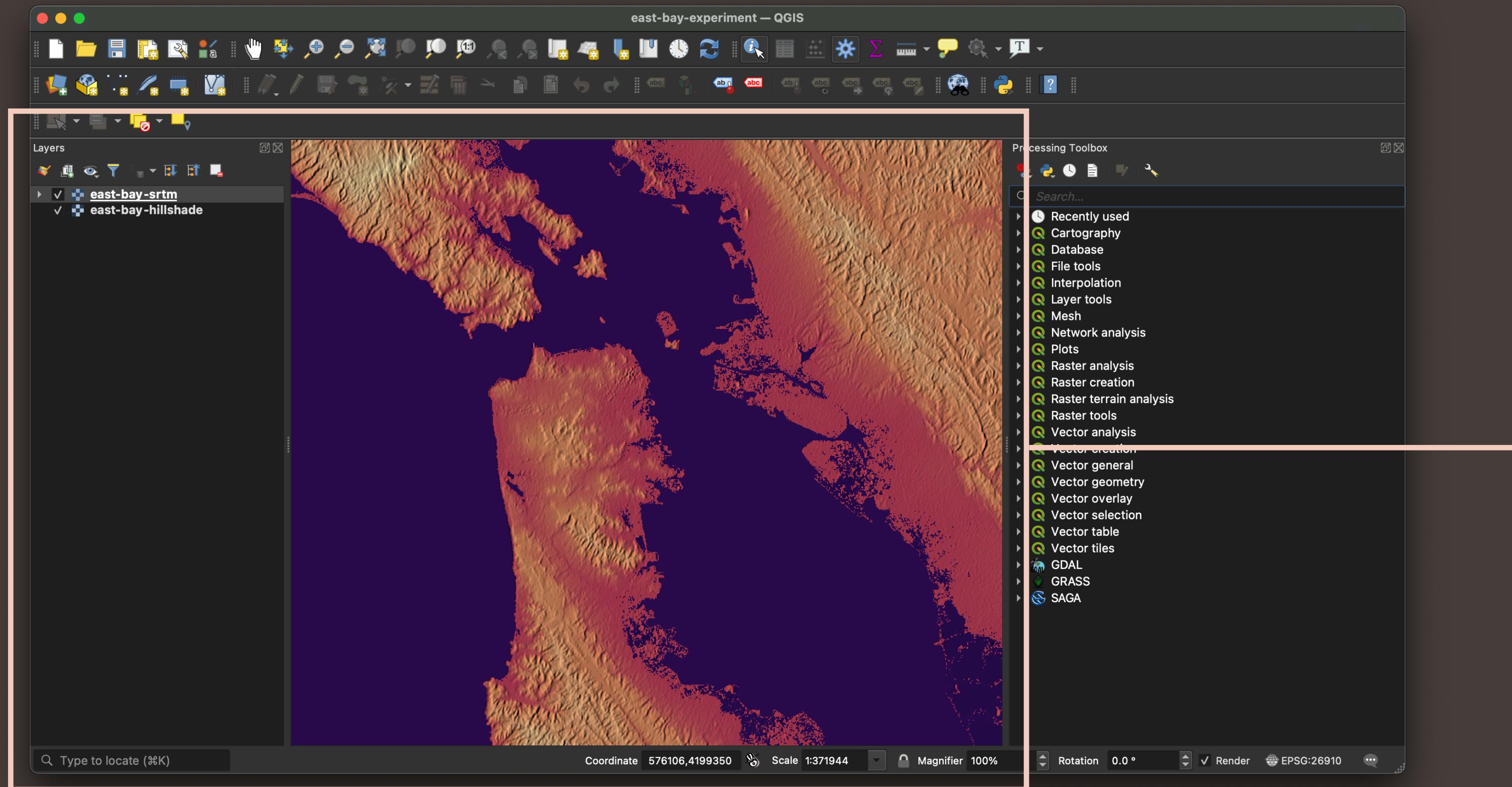
Geographic  
Information  
System

Software for **viewing**,  
**operating on**, and  
**managing** geospatial data.



QGIS

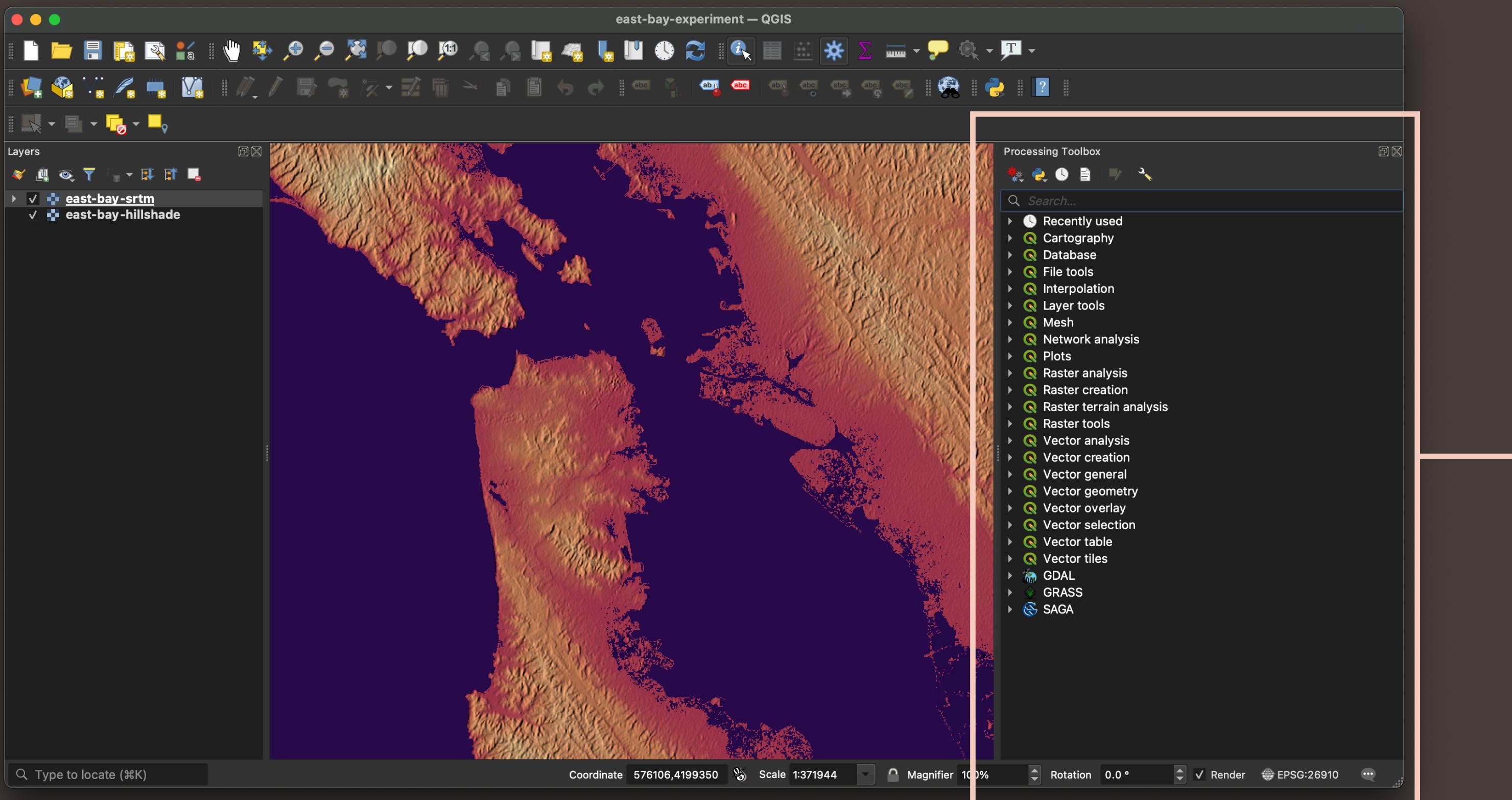
# What is GIS?



**Zoomable, pan-able  
canvas** for viewing  
geospatial data as  
**layers**



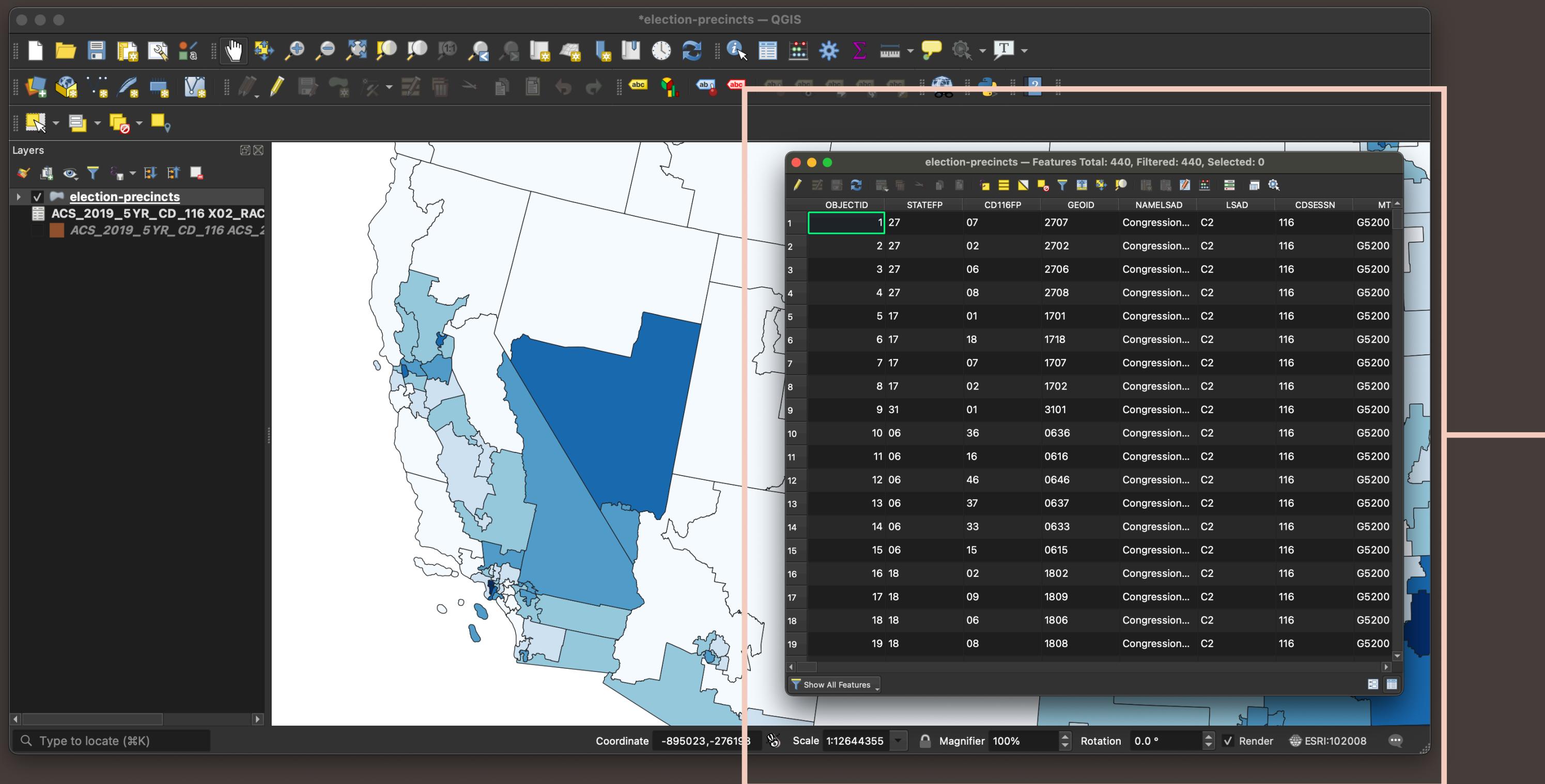
# What is GIS?



**Processing tools** for  
operating on  
geospatial data to  
produce new layers



# What is GIS?



Attribute tables for connecting tabular data to geometry



# Roadmap



# Roadmap



# Motivation

**Why study the challenges and needs of geospatial data users?**

1. There has been **relatively little focus** on tools for working with geospatial data in PL and HCI circles.
2. There is a **growing gap** between the **amount** of geospatial data and the **experts** capable of analyzing it.
3. Geospatial data is fundamental to understanding **climate change, public health, election integrity, racial and economic inequity**, and much more.

# Motivation

Why study the challenges and needs of geospatial data users?

1. There has been **relatively little focus** on tools for working with geospatial data in PL and HCI circles.

# Geospatial Data in HCI

**“Why Are Geographic Information Systems Hard To Use?”**

Traynor and Williams, CHI Short Papers, 1995

**“End Users and GIS: A Demonstration Is Worth a Thousand Words”**

Traynor and Williams, Your Wish is My Command:  
*Programming By Example*, 2001

**“Human-computer interaction and geospatial technologies – context”**

Mordechai Hackley, *Interacting with Geospatial Technologies*,  
2010

## CHAPTER **Six**

**End Users and GIS: A Demonstration Is Worth a Thousand Words**

**CAROL TRAYNOR**  
*Saint Anselm College*

**MARIAN G. WILLIAMS**  
*University of Massachusetts, Lowell*

Color profile: Generic CMYK printer profile  
Composite Default screen

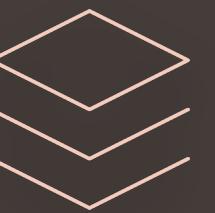
V:\002564\002564.VP  
Monday, December 18, 2000 1:30:41 PM  
TNT Job Number: [002564] • Author: [Lieberman] • Page: 115

S  
R  
L

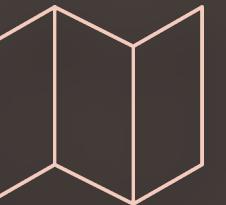
# Geospatial Data in HCI

## The Many Skillset Problem

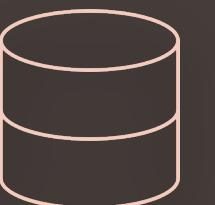
*Your Domain Knowledge*



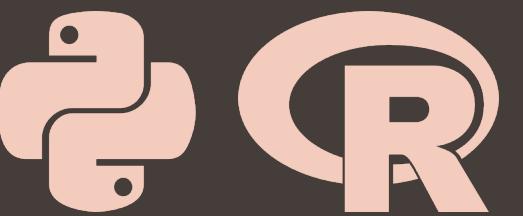
*Geography*



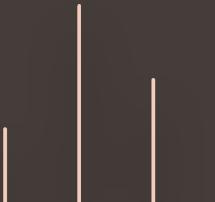
*Cartography*



*Databases*



*Programming*



*Statistics*

# Geospatial Data in HCI

## The Disconnected Toolbox Problem



ArcGIS

.....

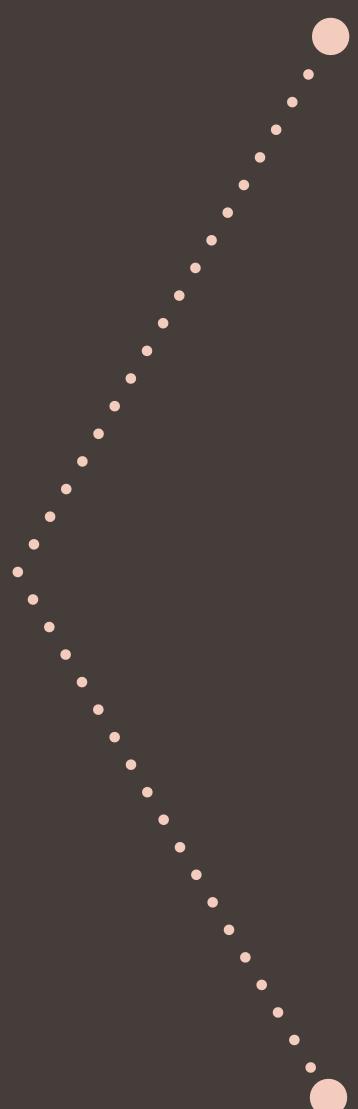
*Server Toolbox*

*Ready to Use Toolbox*

***Spatial Analyst Toolbox***

Spatial Statistics Toolbox

***... +35 More***



*Bitwise Left Shift*  
*Kriging*  
*Raster Calculator*  
*Iso Cluster Unsupervised*  
*Fuzzy Overlay*  
*Zonal Histogram*  
*Darcy Flow*  
***... +200 More***

# Geospatial Data in HCI

## What does this work miss?

This work relies (mainly) on **experts analyzing GIS software** rather than **directly observing GIS users**.

This work hasn't been revisited to address **modern geospatial tools** beyond desktop GIS software.

This work doesn't consider **domain-specific needs** of geospatial data users.

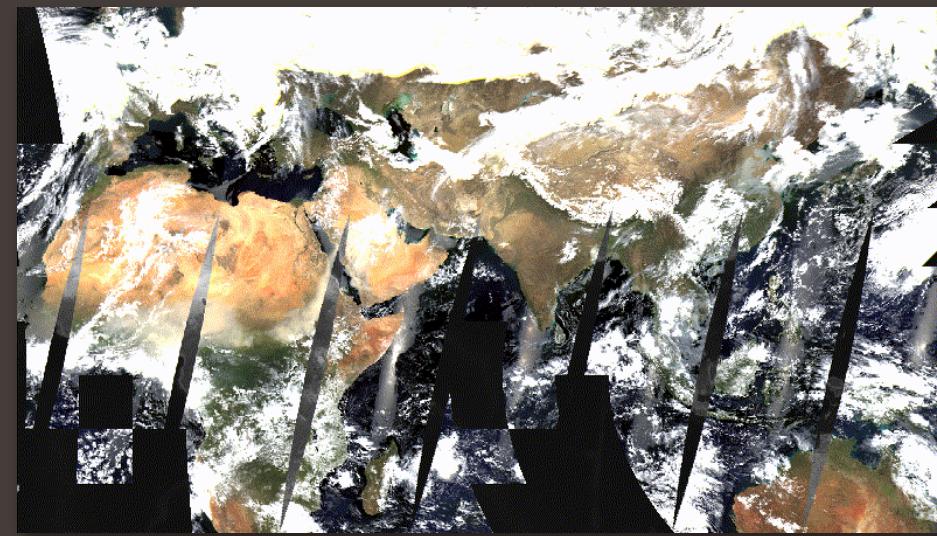
# Motivation

Why study the challenges and needs of geospatial data users?

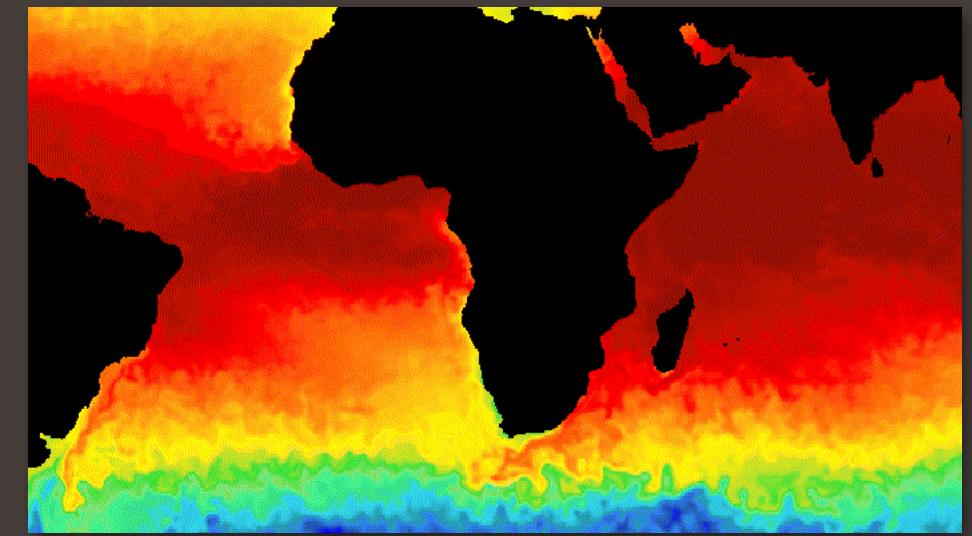
2. There is a **growing gap** between the **amount** of geospatial data and the **experts** capable of analyzing it.

# An Abundance of Geospatial Data

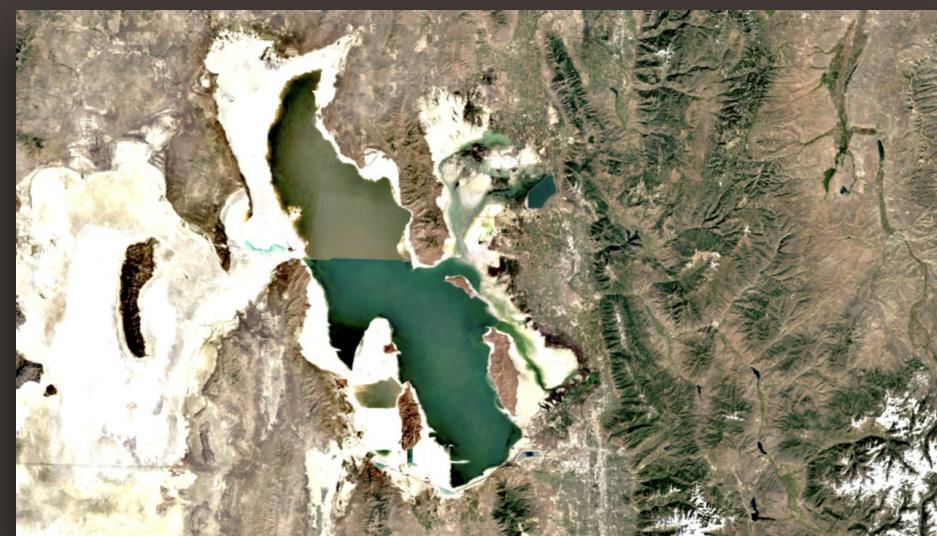
Rolf et al. estimate  
that global imagery  
data increases by  
***80TB per day.***



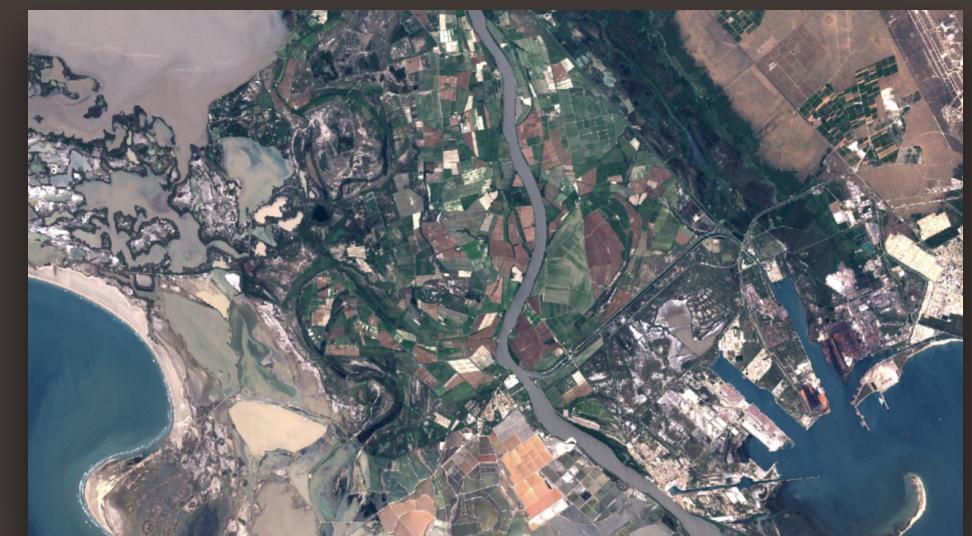
**NASA MODIS**



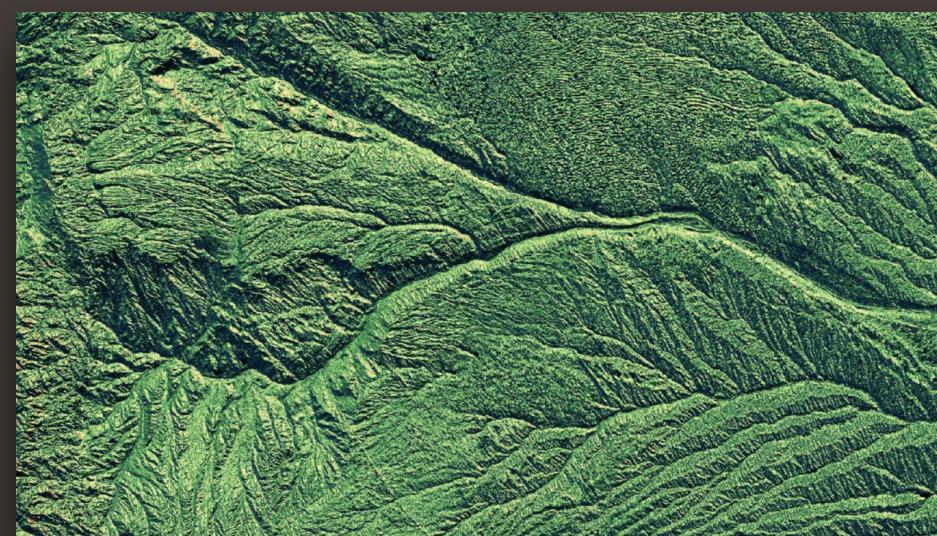
**NASA ASTER**



**ESA Sentinel-2**



**NASA / USGS Landsat**



**NASA SRTM**



**USDA NAIP**

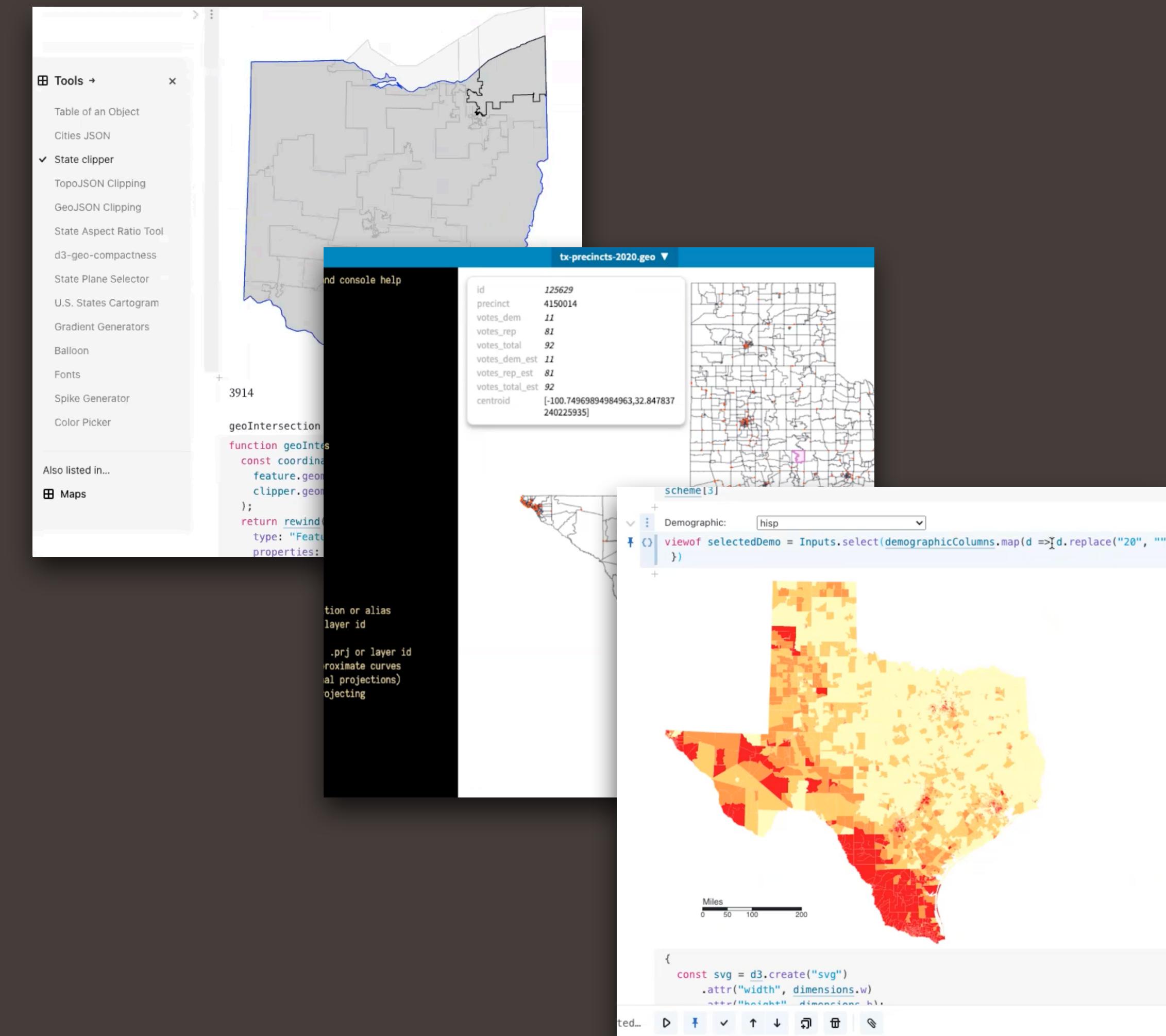
# Motivation

Why study the challenges and needs of geospatial data users?

3. Geospatial data is fundamental to understanding **climate change, public health, election integrity, racial and economic inequity**, and much more.

# The Importance of Geospatial Data

Participant 13 is a  
**data journalist**  
exploring whether Texas' new  
electoral precincts “**pack**” or  
“**crack**” racial minority  
groups.



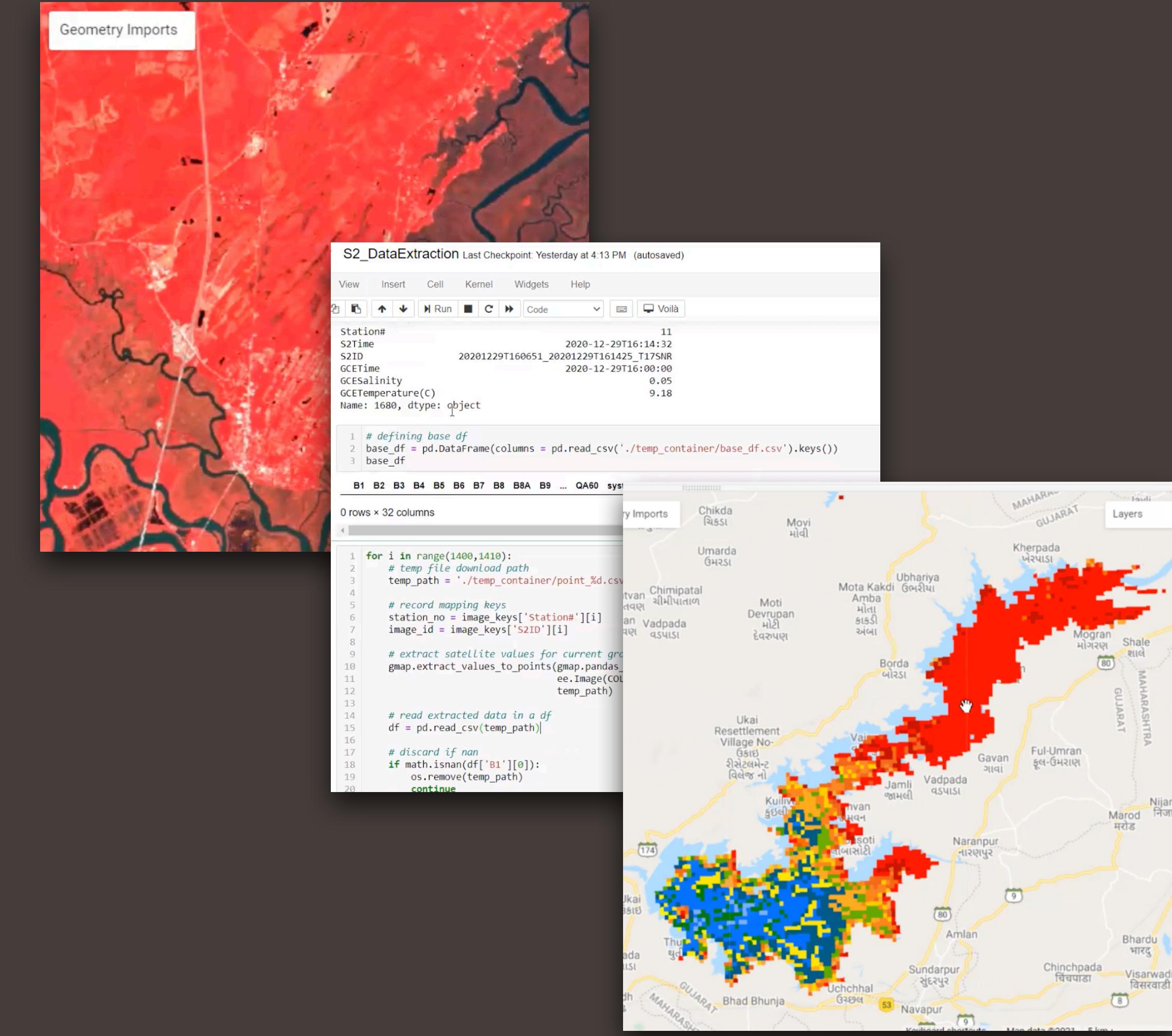
Observable



JavaScript

# The Importance of Geospatial Data

Participant 8 is a **climate scientist** building **ML models** to predict development of Cyanobacterial harmful algal blooms (CyanoHABs) from satellite imagery.



Google  
Earth  
Engine



Jupyter



Python

# Roadmap



# Roadmap



# Research Questions

1.

What are the **challenges** users face when attempting to **gather, analyze**, or **visualize** geospatial data?

2.

Which of these challenges are **shared** between **experts** and **non-experts** alike? Which are **unique** to one group?

3.

Which of these challenges are **shared** across our **domains of interest?** Which are **unique** to one group?

# Roadmap



# Roadmap



# Study Design

25  
participants

## *Domain*



9

Earth and climate scientists



8

Data journalists



6

Social scientists



2

“Miscellaneous”

## *Tool Usage*



14

Programming Environments



8

GUI-based Software



3

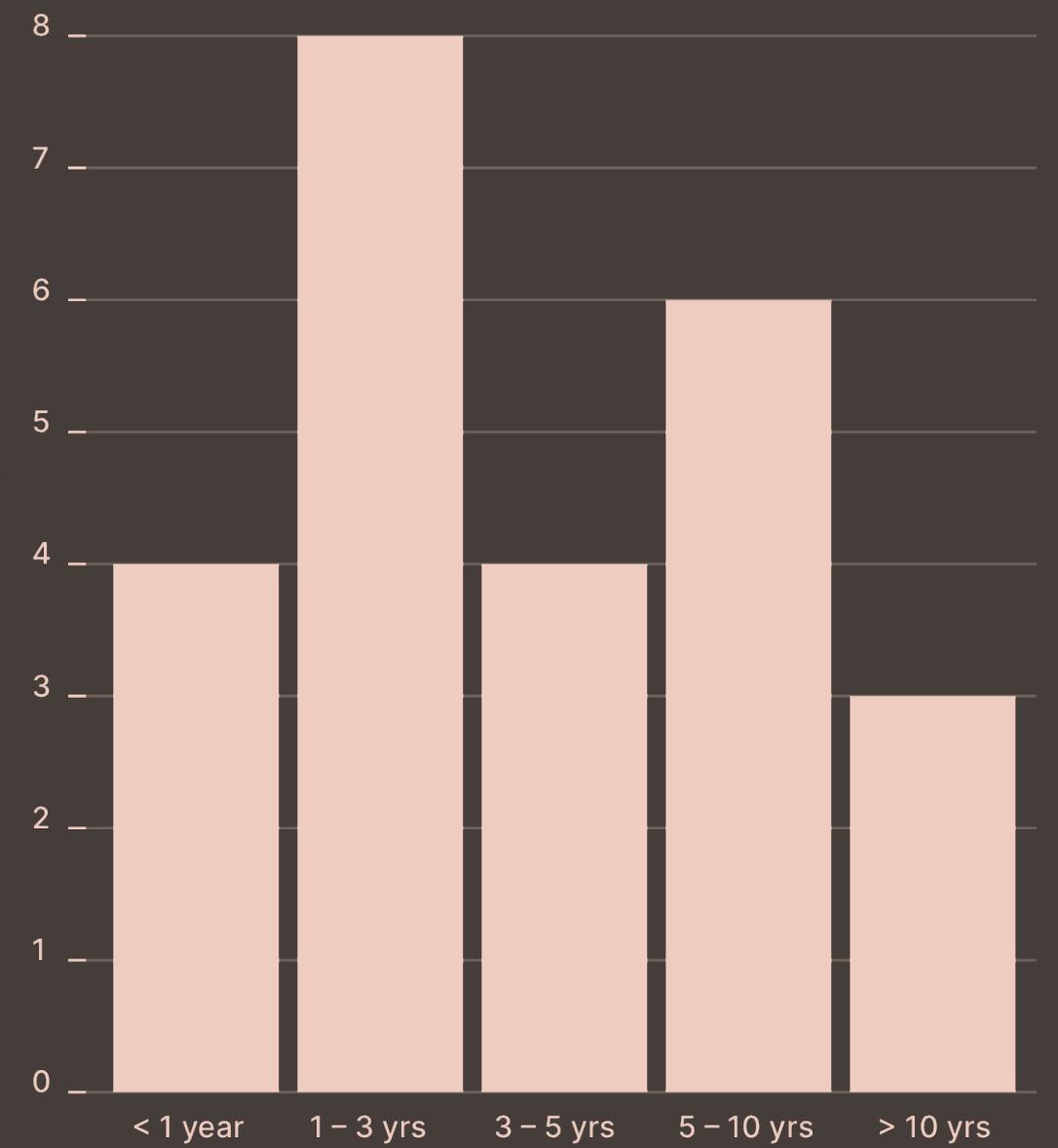
Both

# Study Design

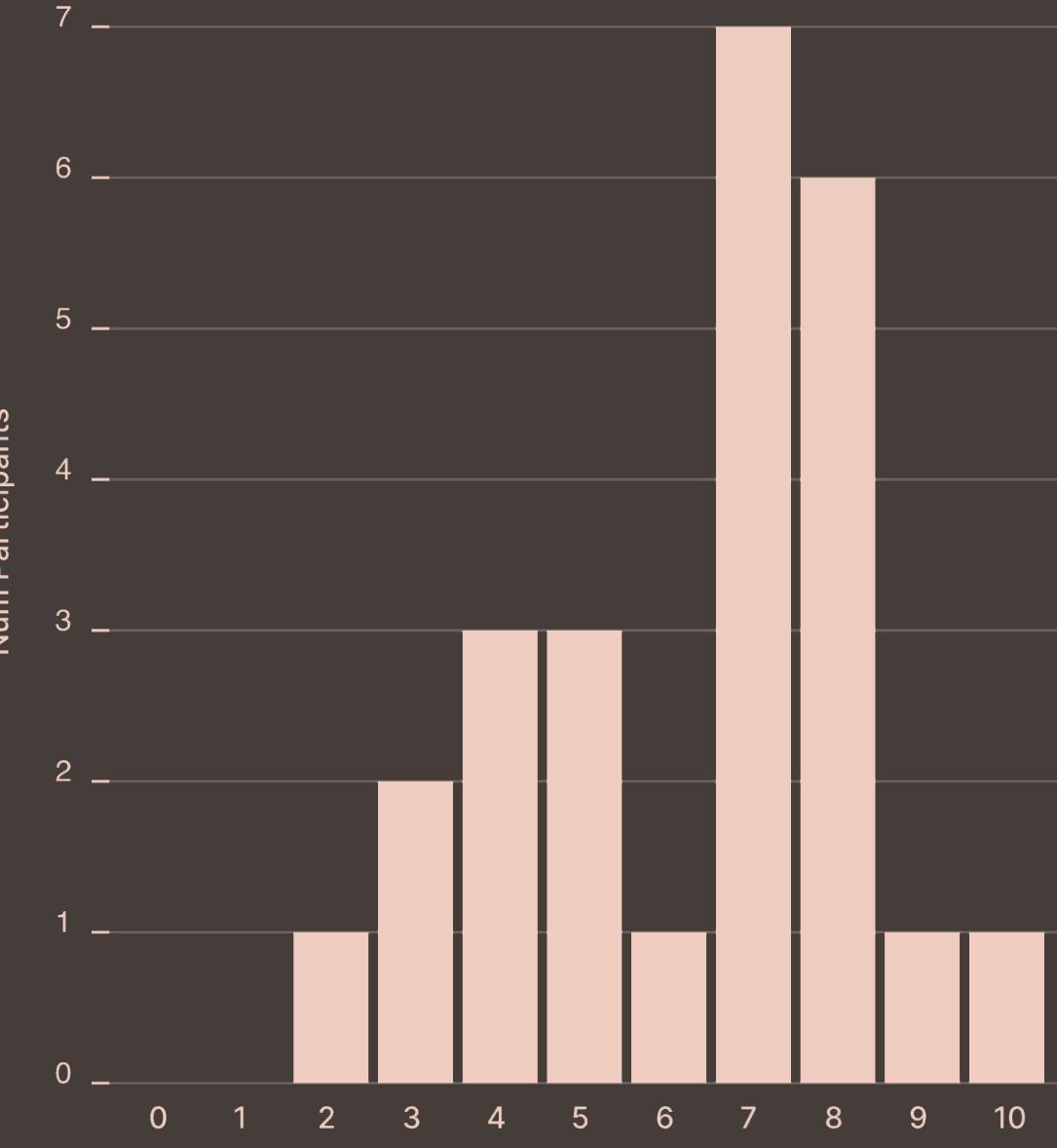
25

participants

*Expertise*



How long have you been  
working with geospatial data?



How would you assess your skill  
level in working with geospatial  
data?

# Methods

## *Contextual Inquiry*

“...go where the [user] works, observe the [user] as he or she works, and talk to the [user] about the work.”

Karen Holtzblatt and Hugh Beyer. 2017. 3 - Principles of Contextual Inquiry. In *Contextual Design (Second Edition)*.

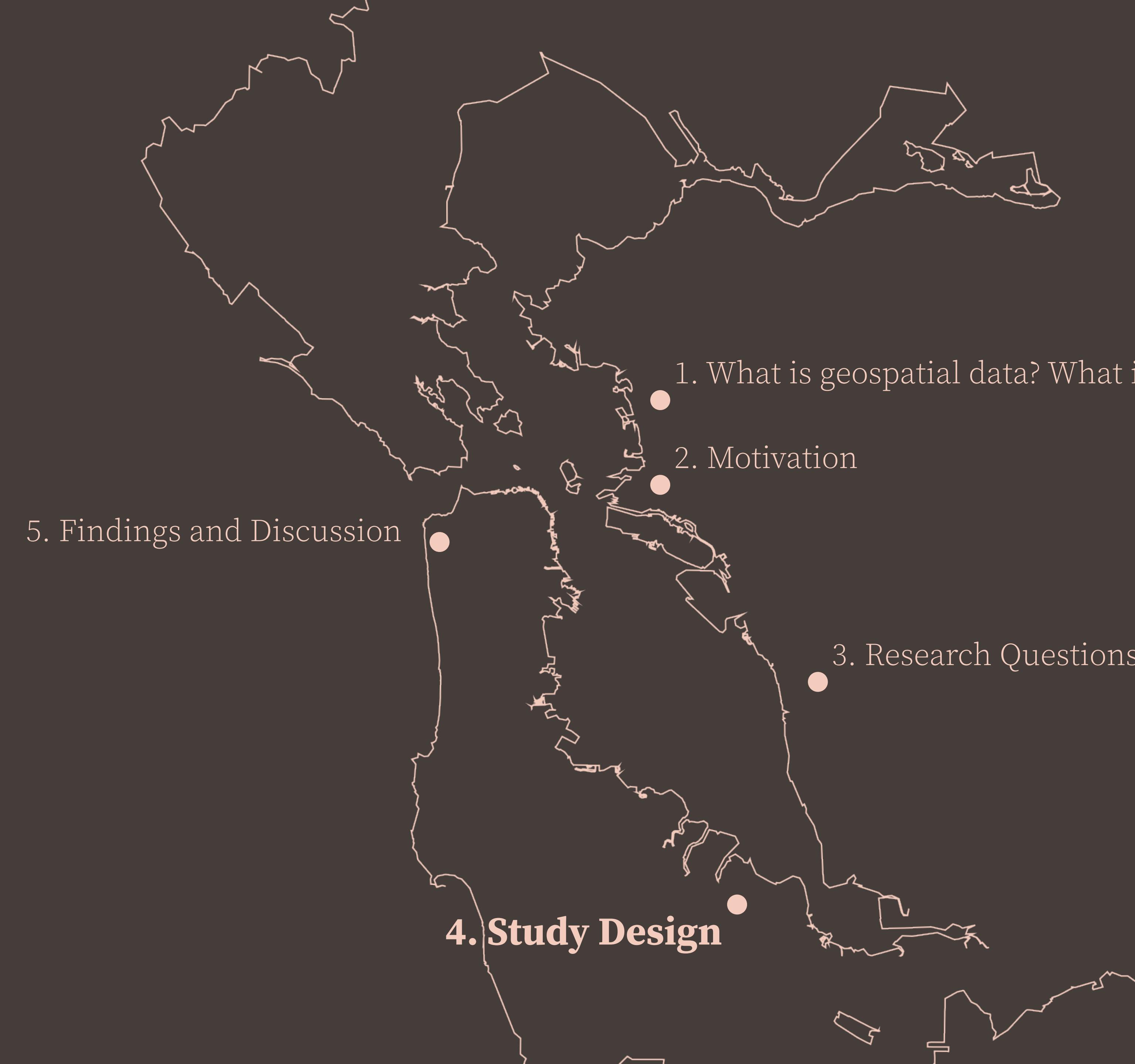
# Methods

## *Thematic Analysis*

“...a method for identifying, analysing and reporting patterns (themes) within data. It minimally organizes and describes your data set in (rich) detail.”

Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. Qualitative Research in Psychology 3, 2 (April 2006), 77–101.

# Roadmap



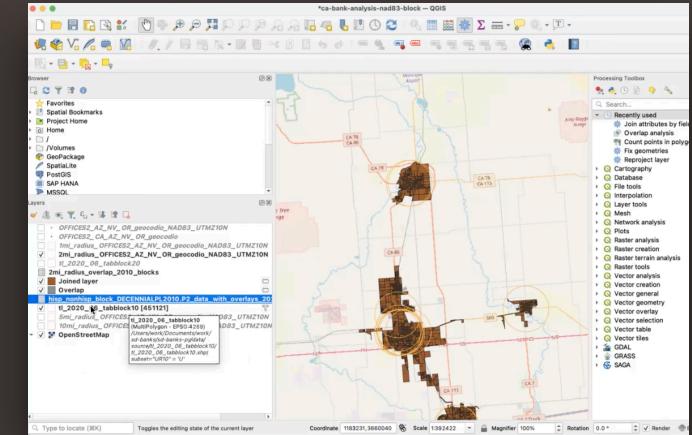
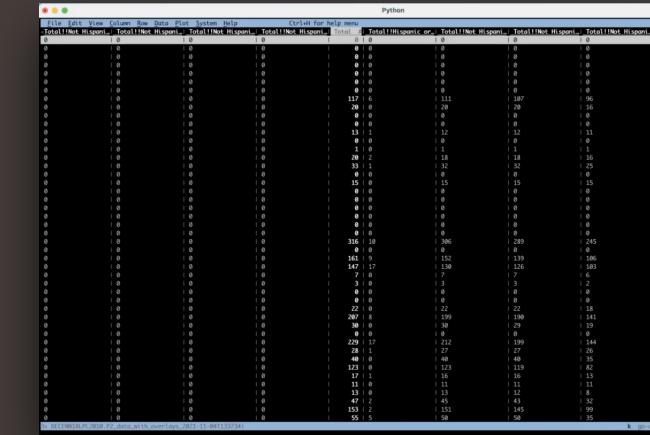
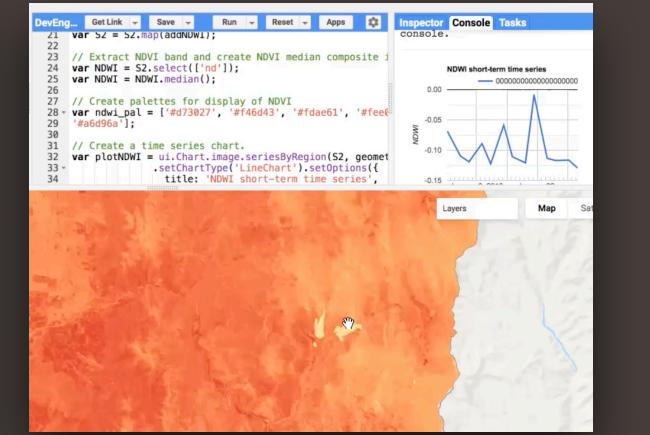
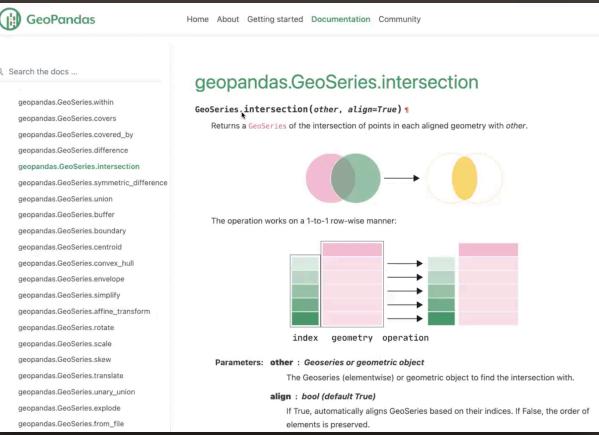
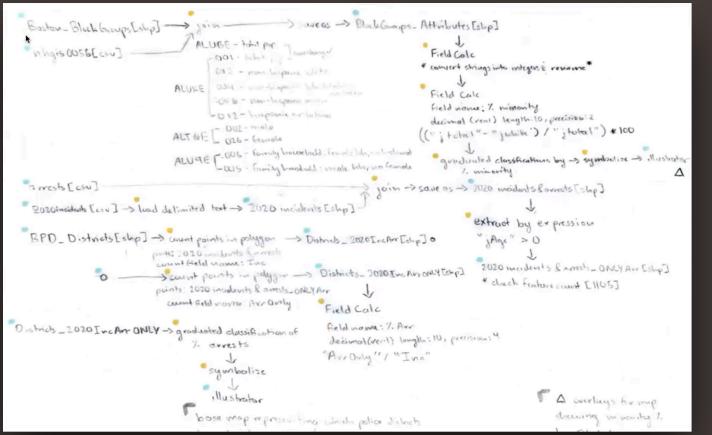
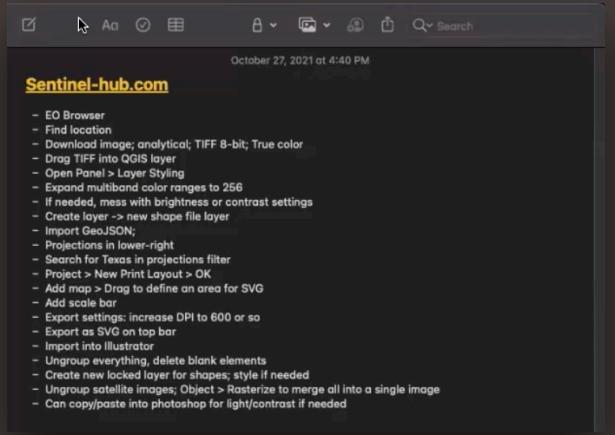
# Roadmap

## 5. Findings and Discussion



# Findings

 These findings are preliminary.



# Creating informal program representations



# GIS Software

# Reasoning about geospatial operator behavior



# IS Software



# Programming Environments

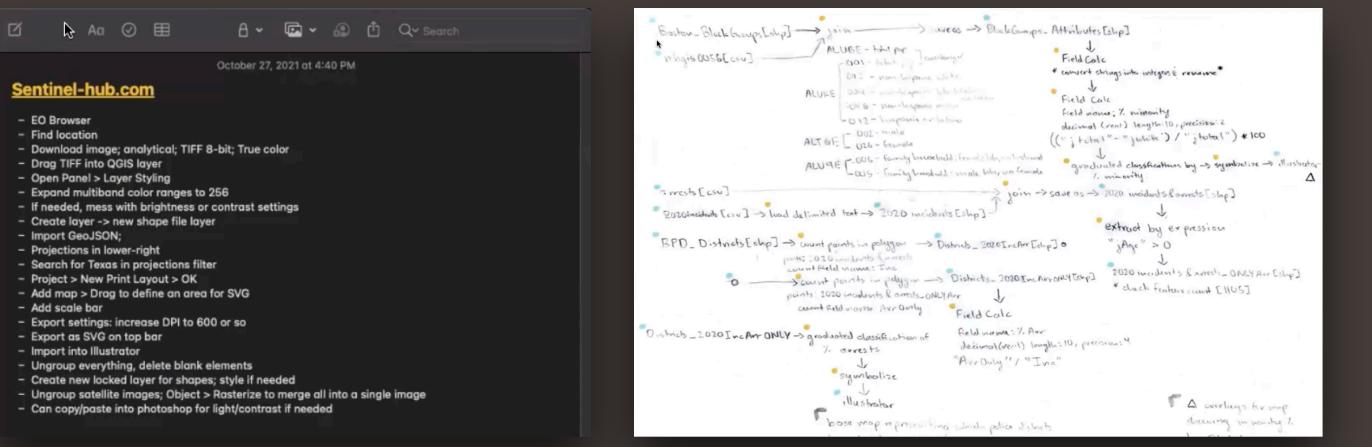


# Lack of spatial visibility in code-based tools



# Programming Environments

# Findings



## Creating informal program representations



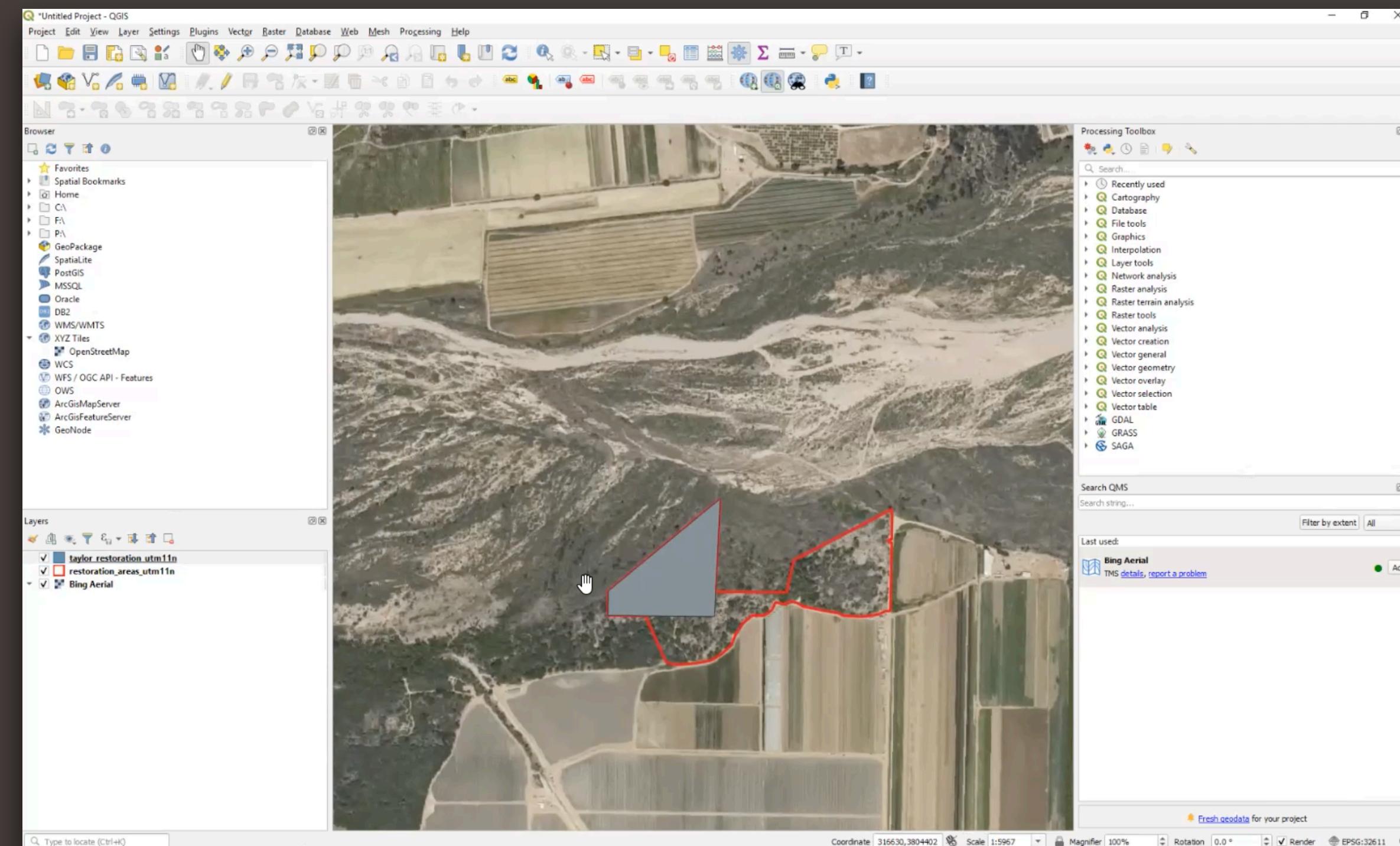
GIS Software

GIS software maintains only a minimal record of steps taken by a user in the interface, **but users want detailed process information.**

# Creating informal program representations

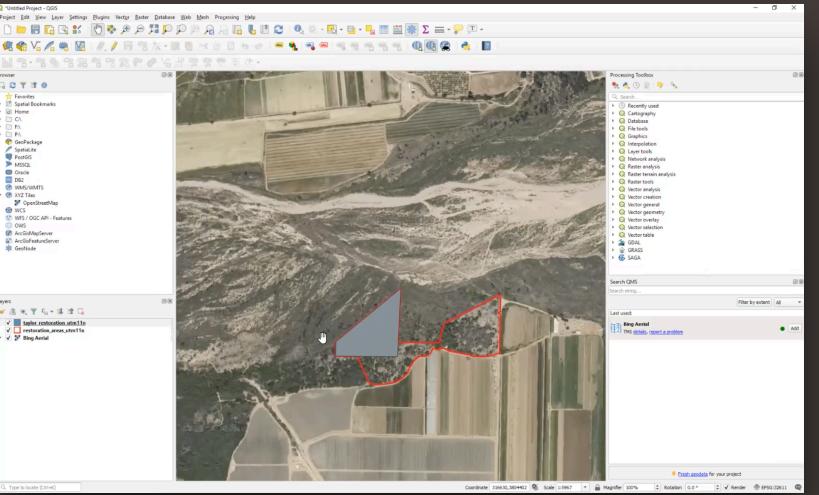
1.

**Load two shapefiles**  
into a new QGIS  
project. **Change Layer**  
**Symbology** to use red  
outline with no fill.



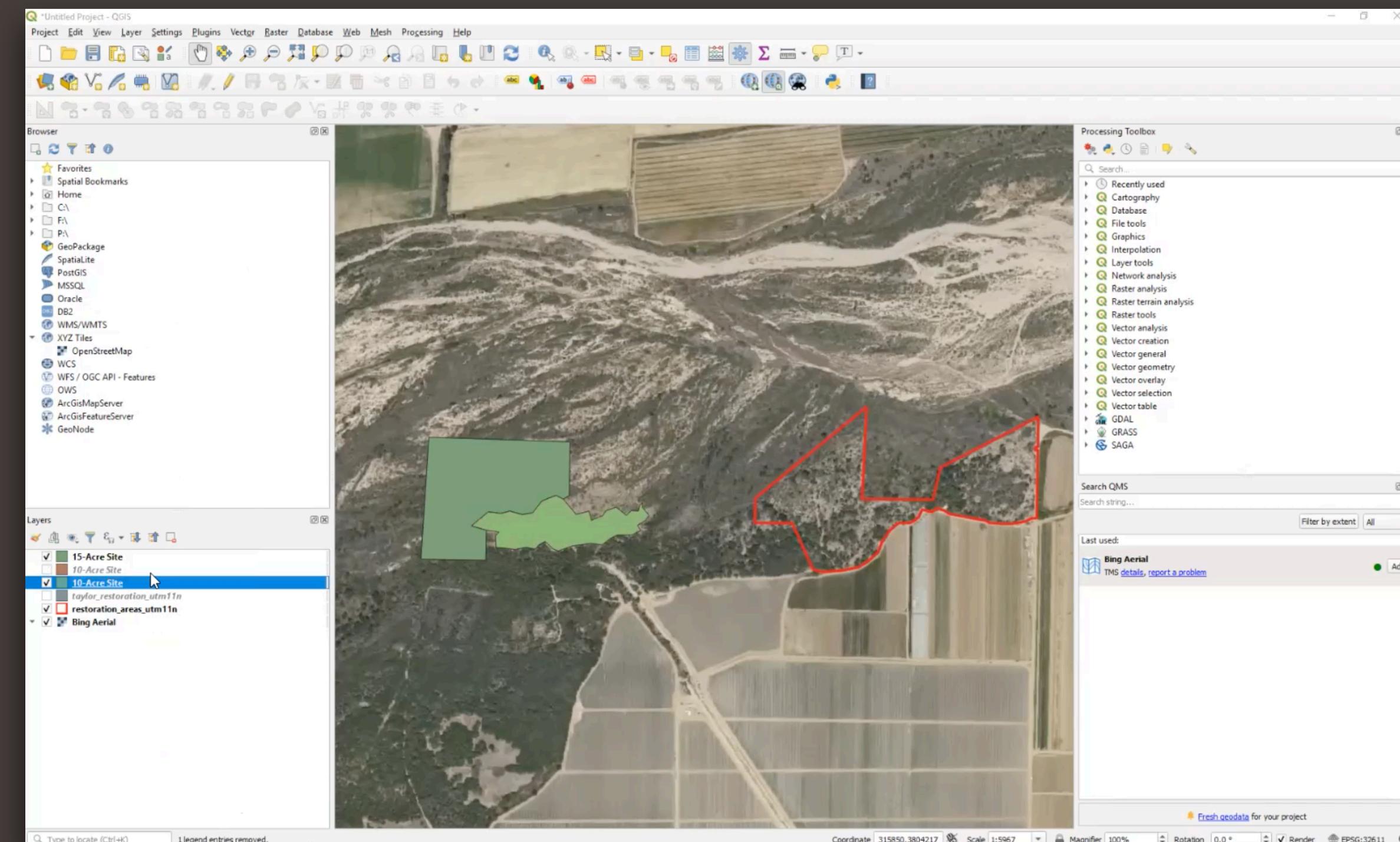
# Creating informal program representations

1.



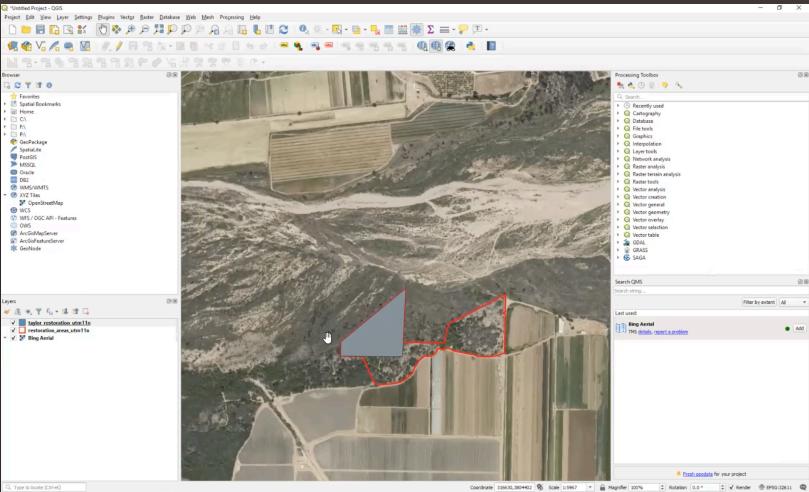
2.

**Load two additional shapefiles into the QGIS project. Toggle layer visibility of one previous layer.**

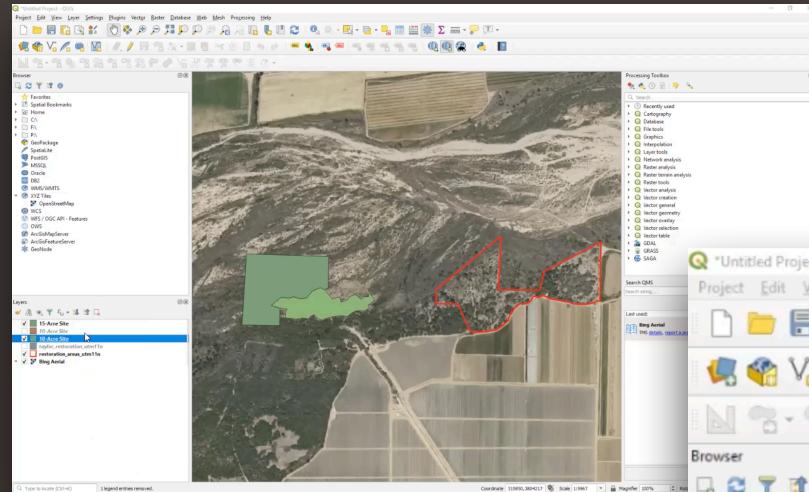


# Creating informal program representations

1.

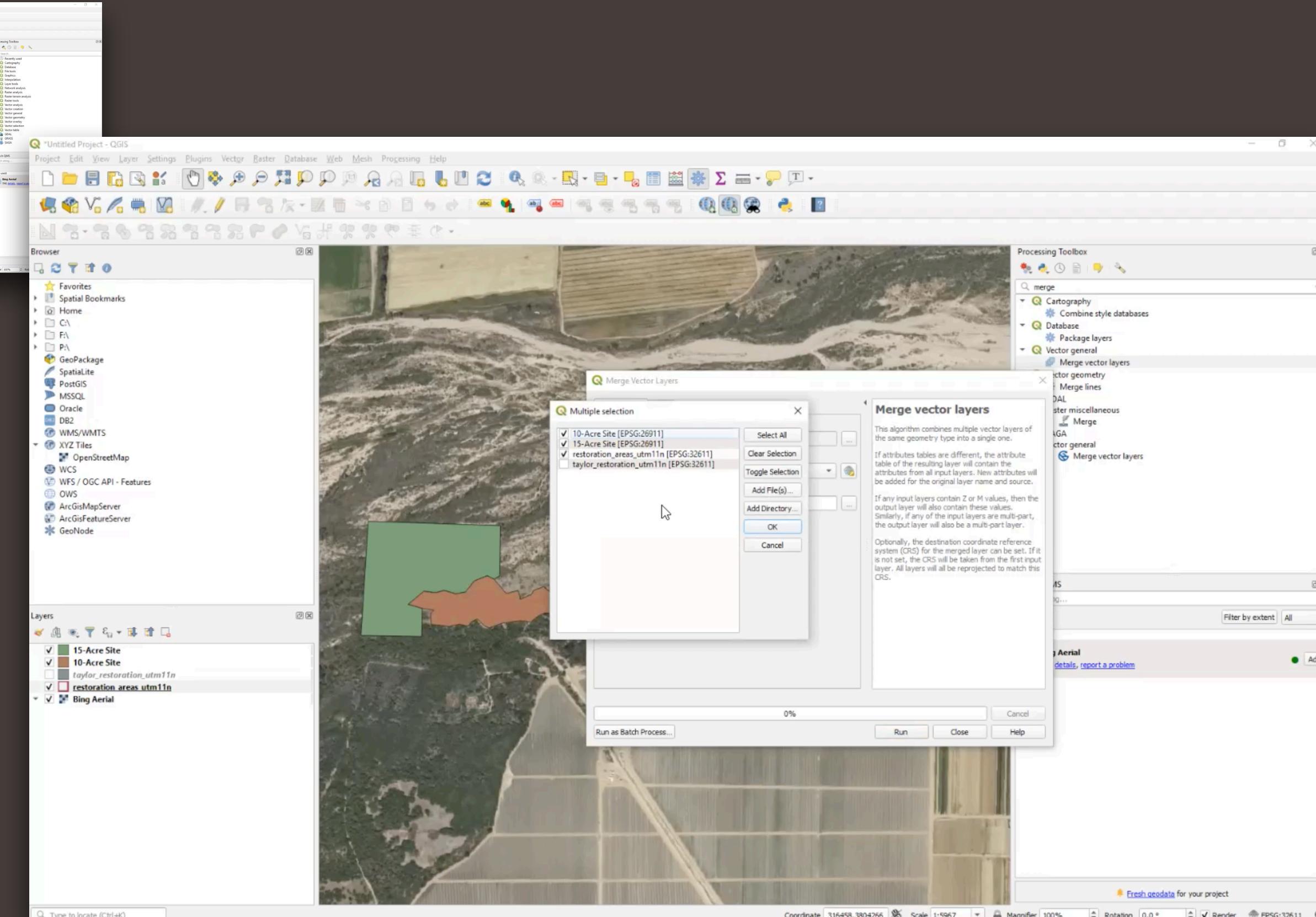


2.



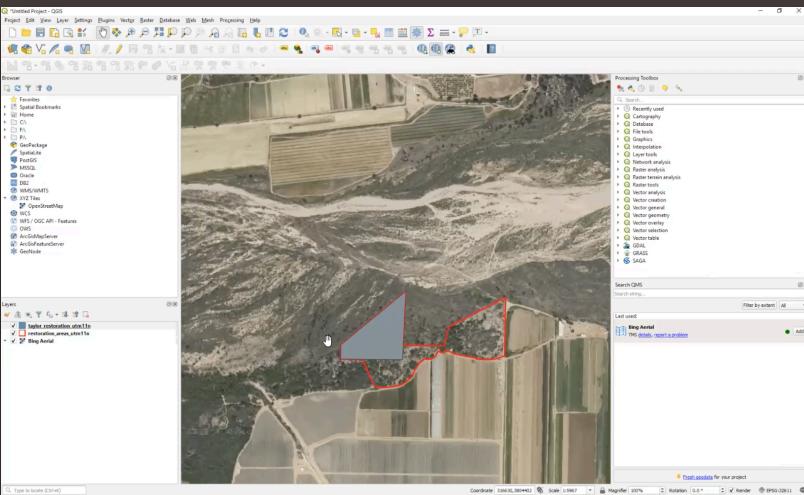
3.

**Attempt a Merge operation** on the three visible shapefiles. **Create a temporary layer** for the result.

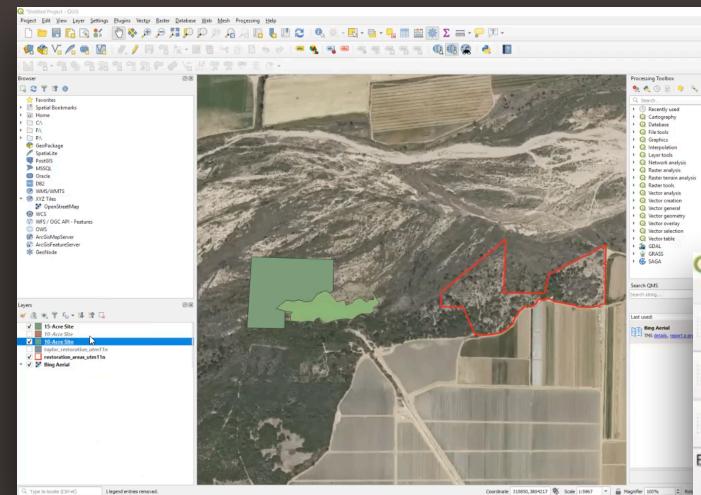


# Creating informal program representations

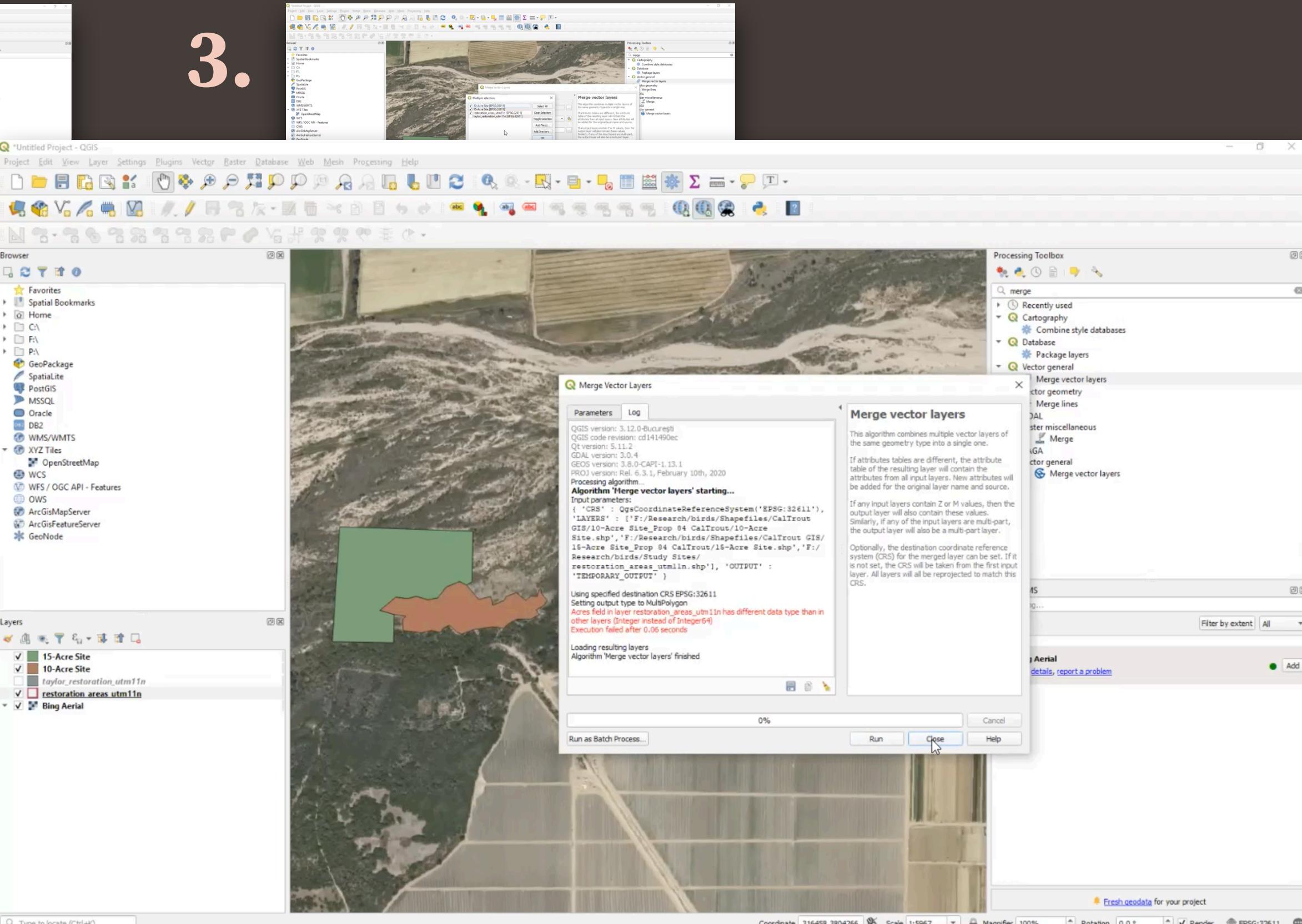
1.



2.



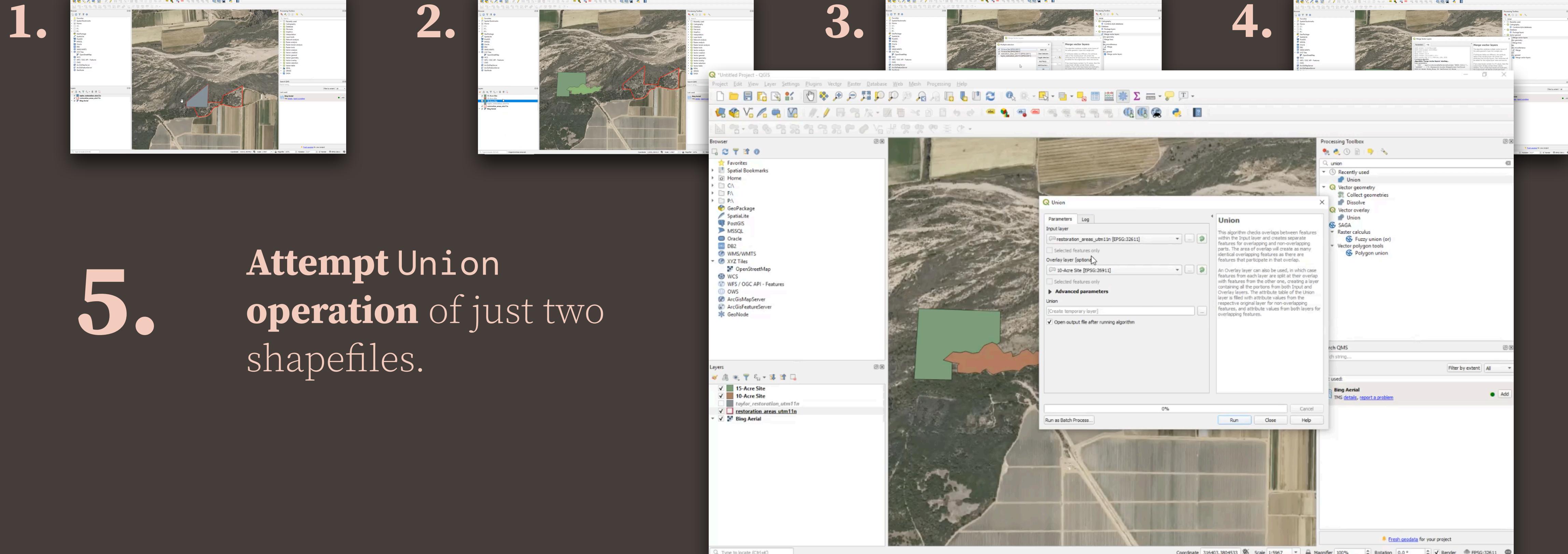
3.



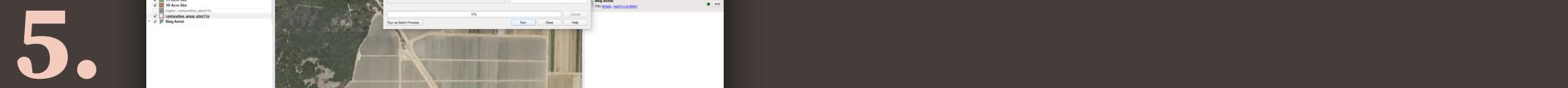
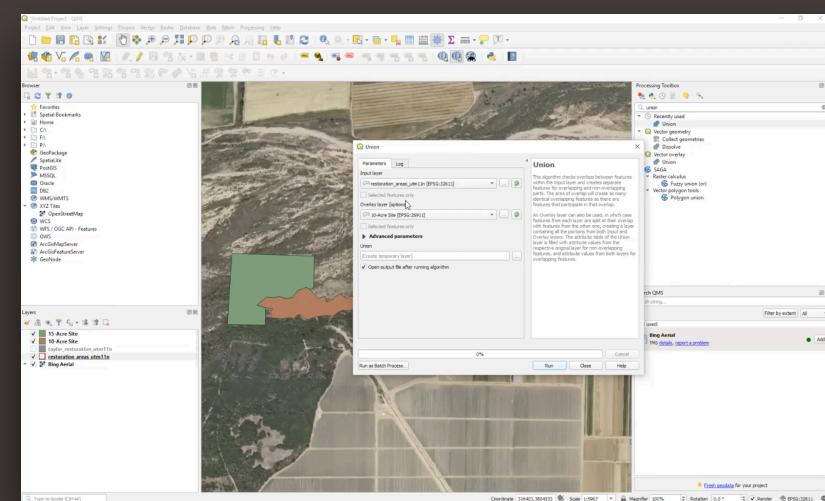
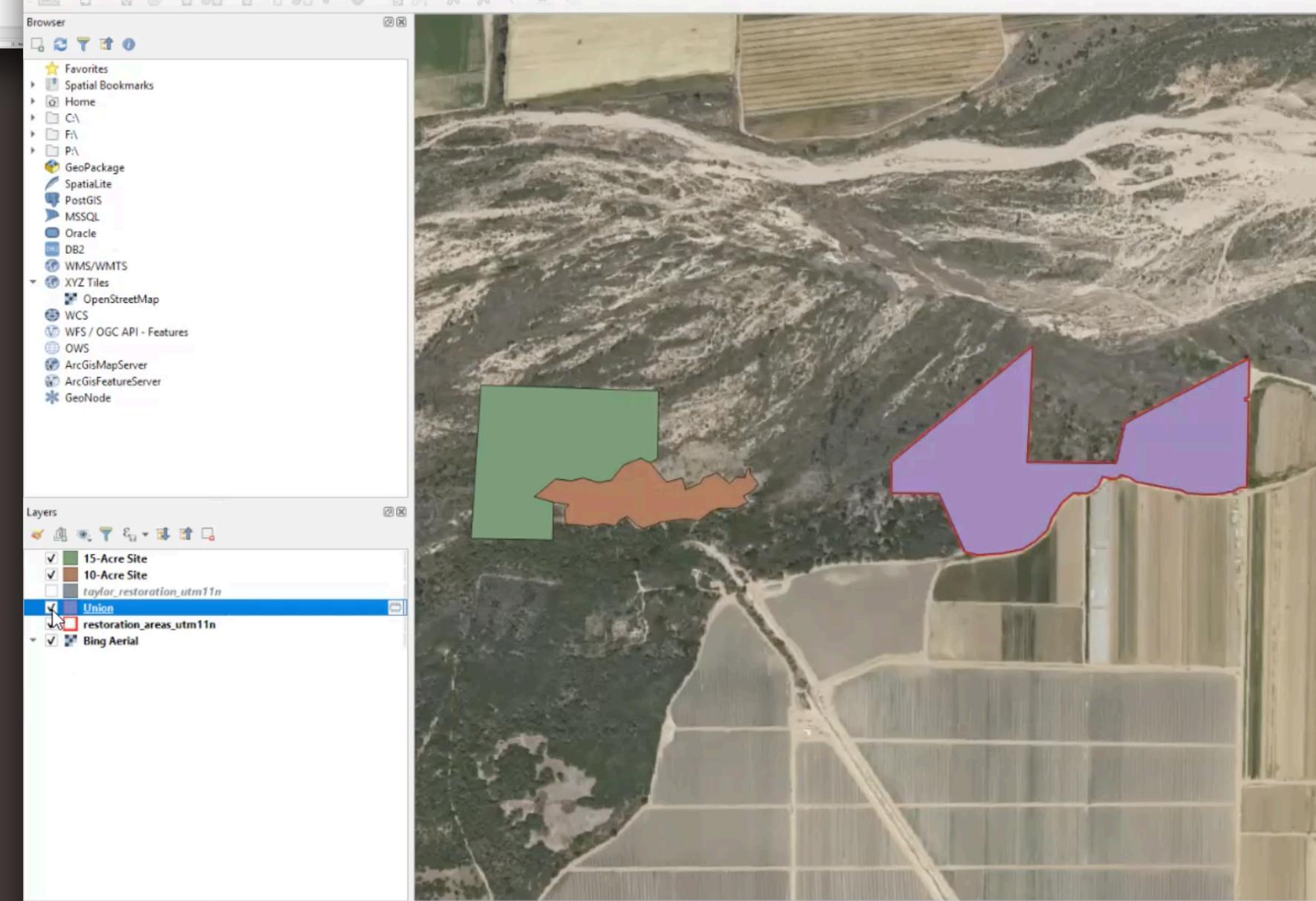
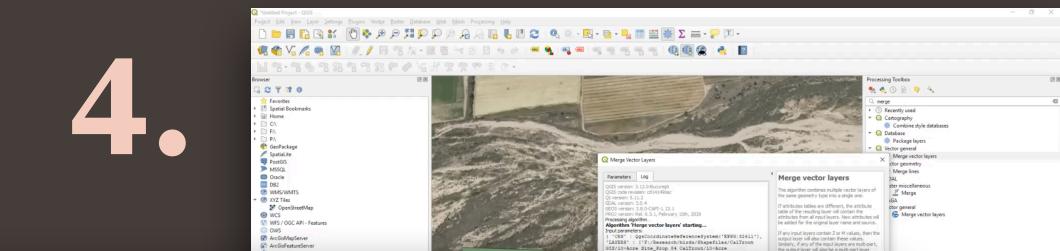
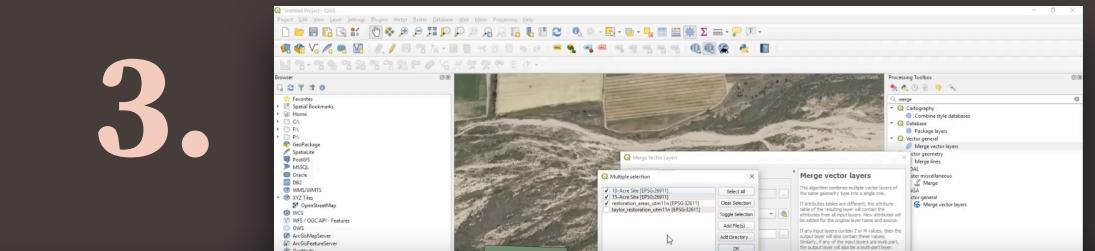
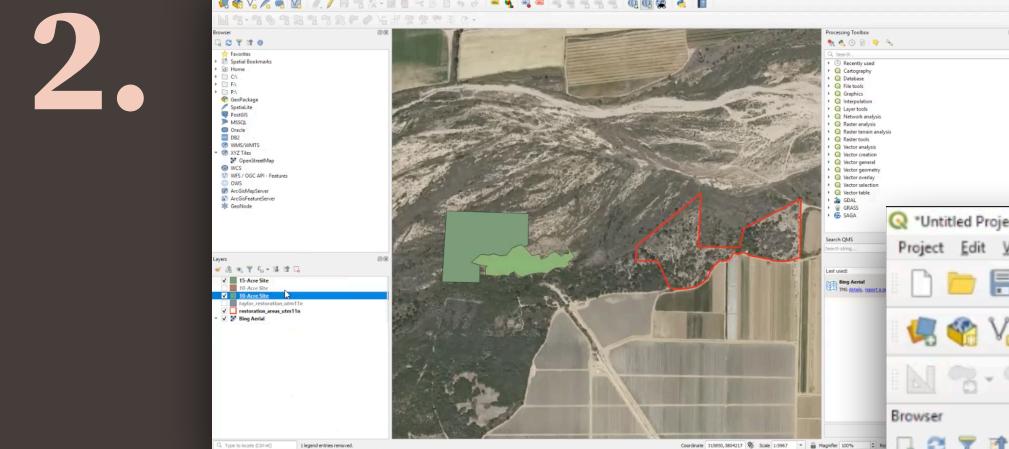
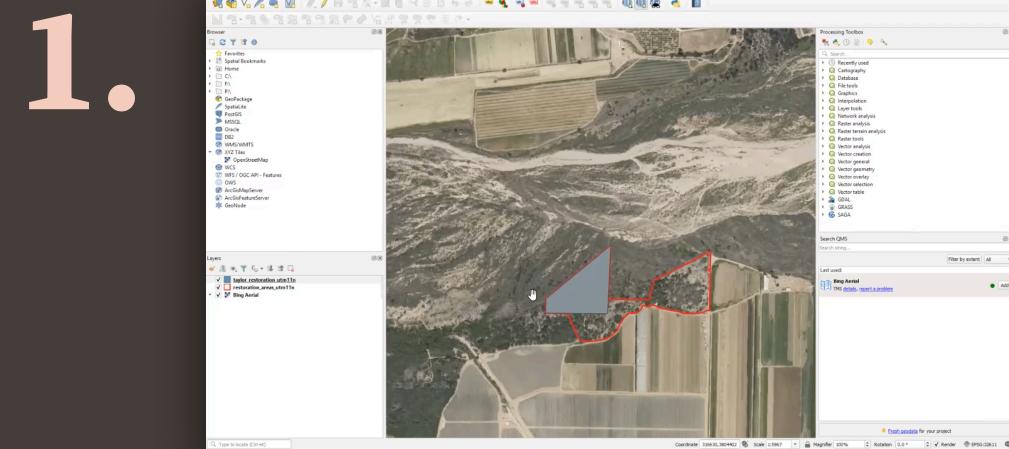
4.

Merge operation fails  
due to mismatching  
data types in attribute  
tables of source  
layers.

# Creating informal program representations

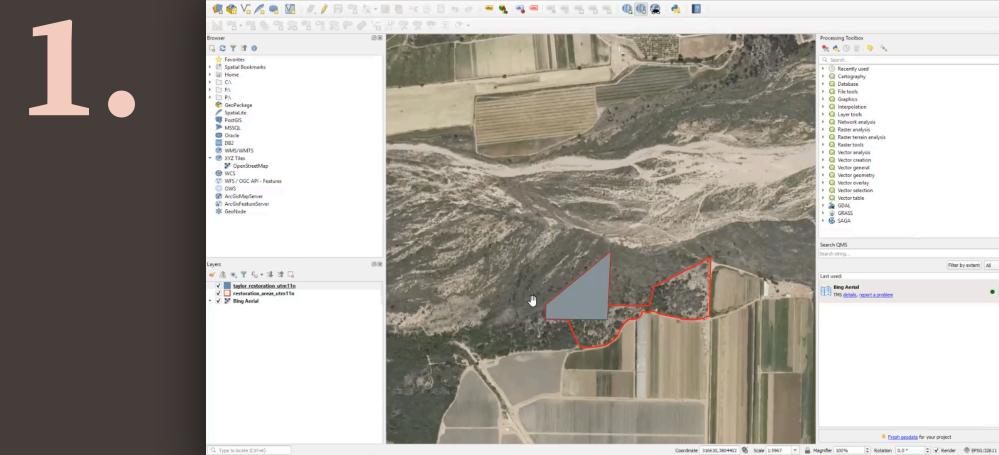


# Creating informal program representations

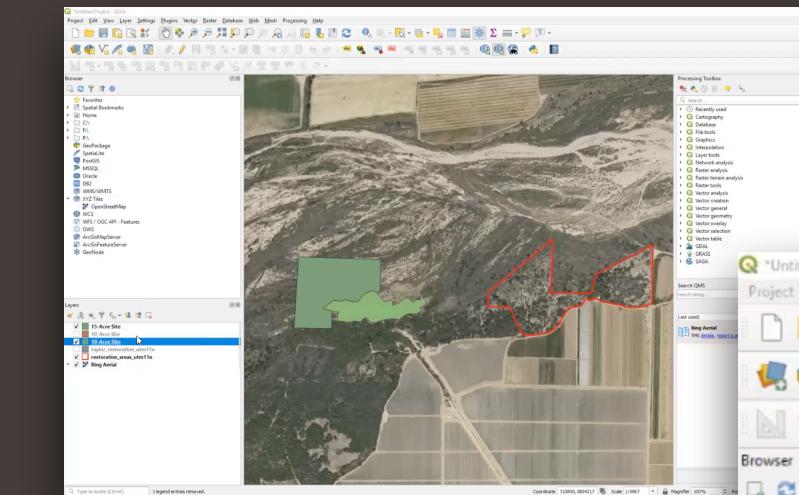


**Union operation  
doesn't error but  
produces an  
unexpected output.**

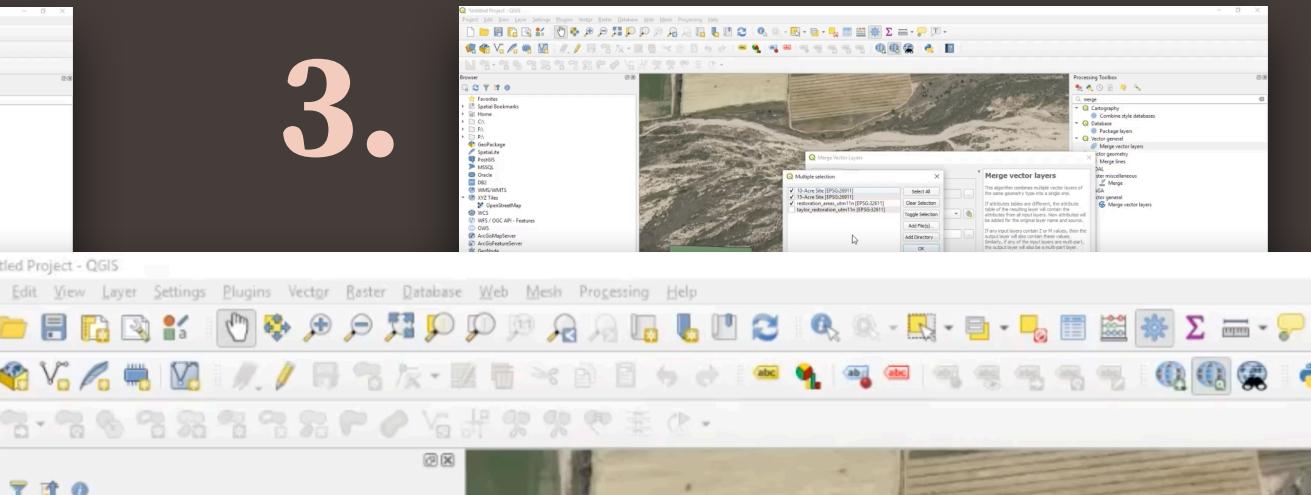
# Creating informal program representations



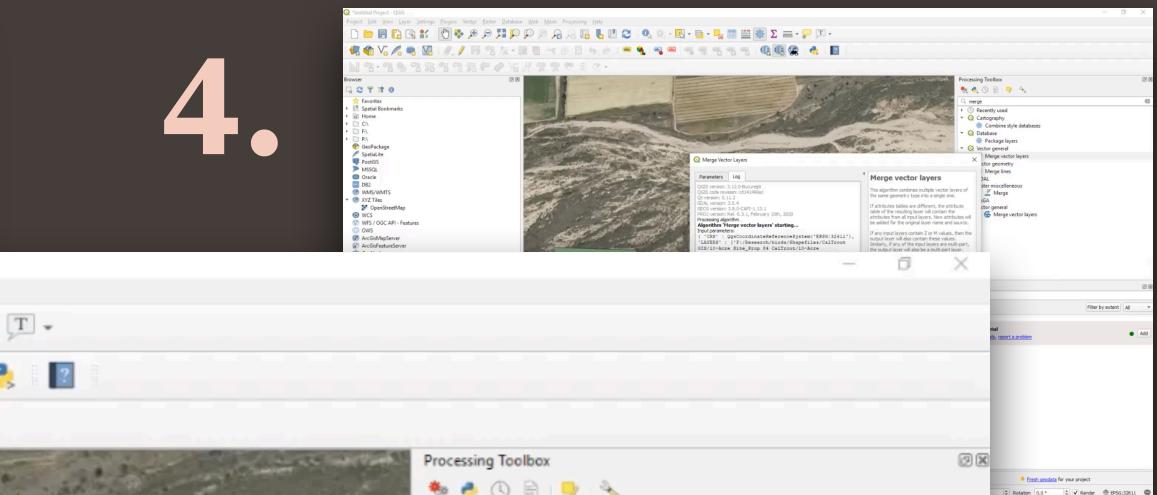
1.



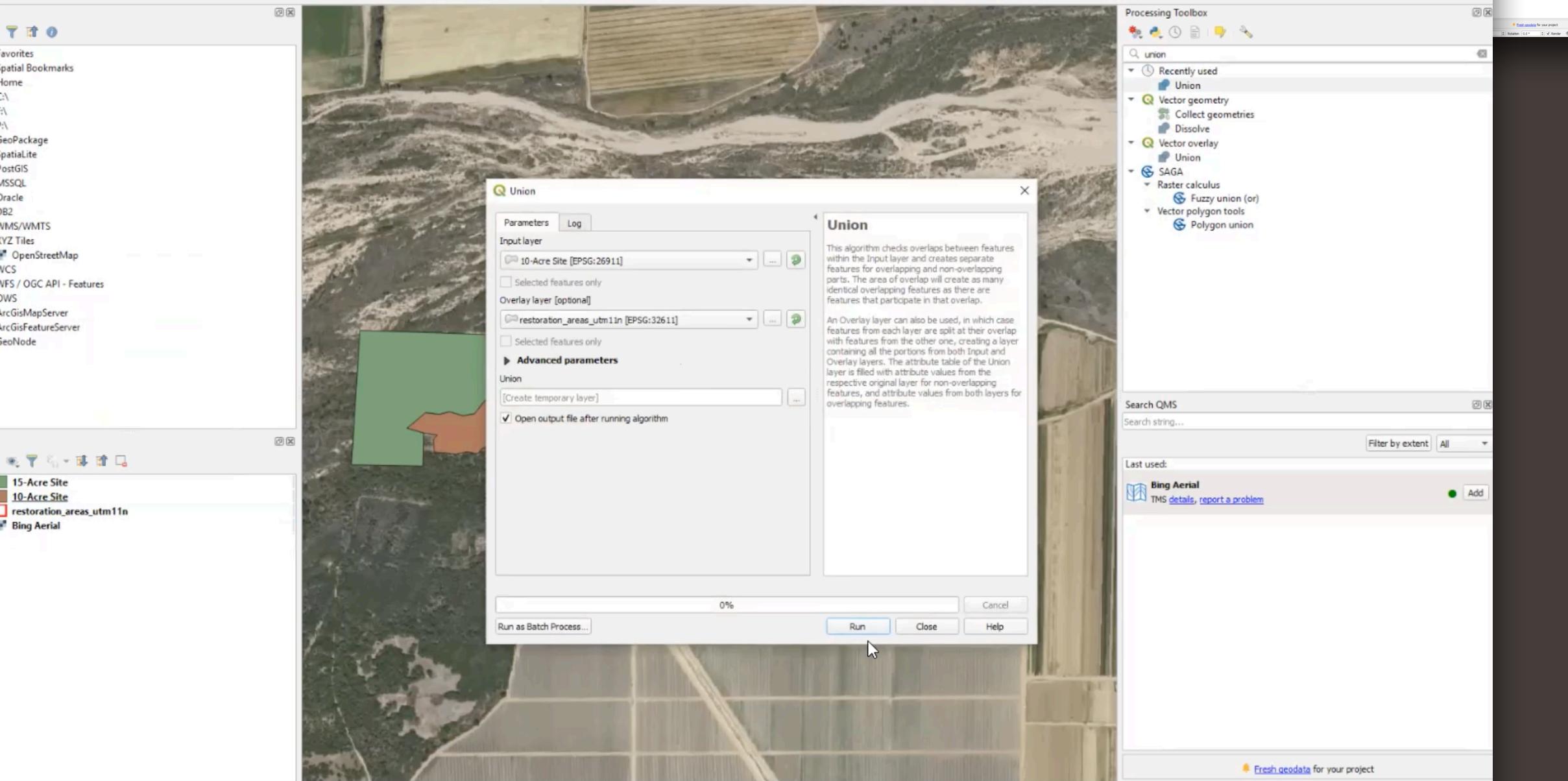
2.



3.

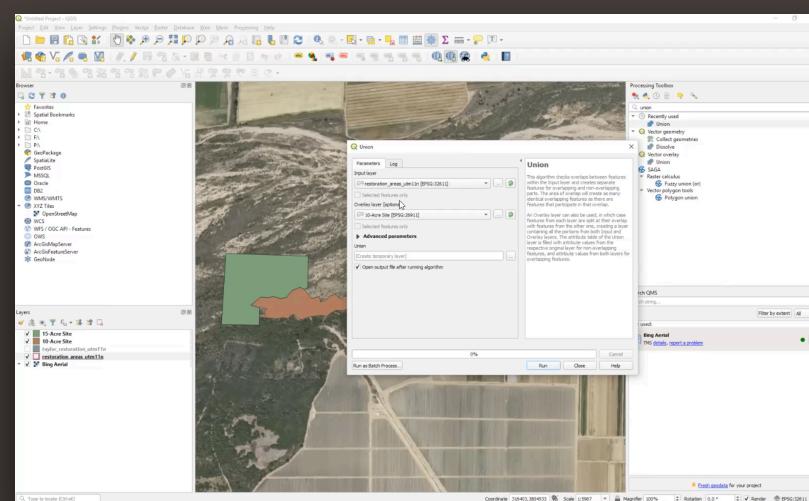


4.

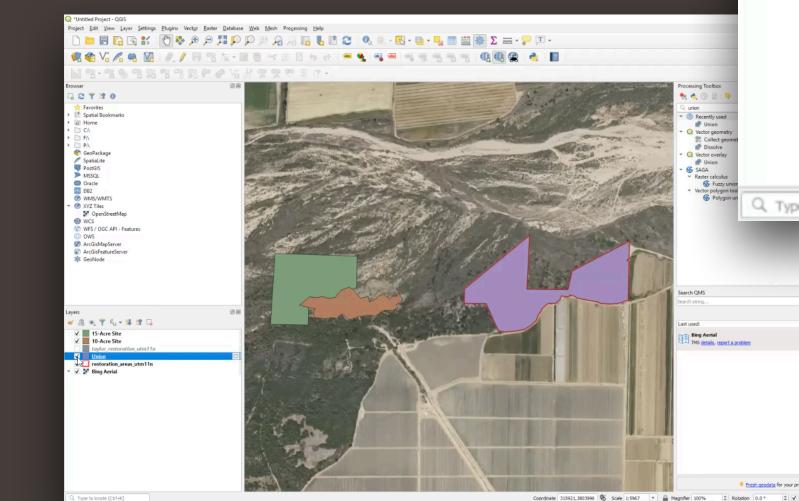


7.

Retry Union  
operation but flip  
layer parameter order.

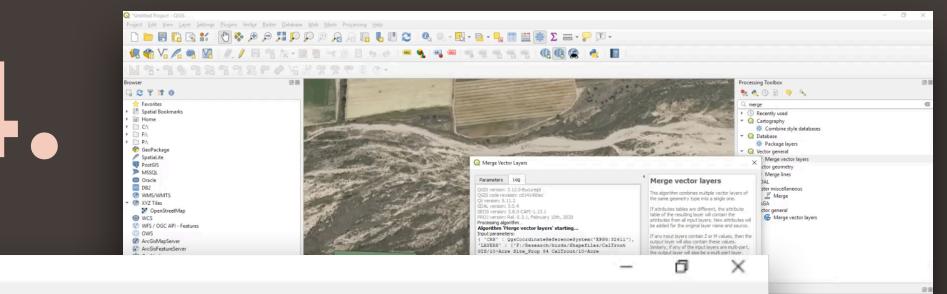
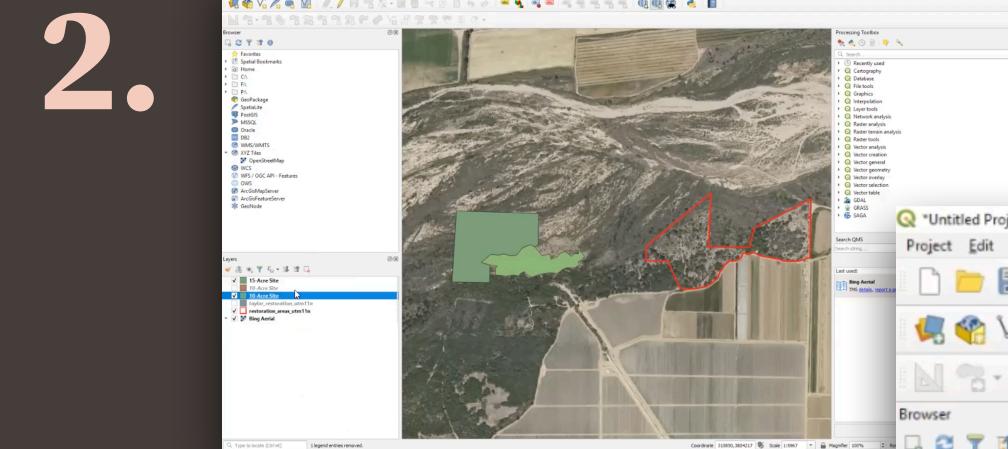
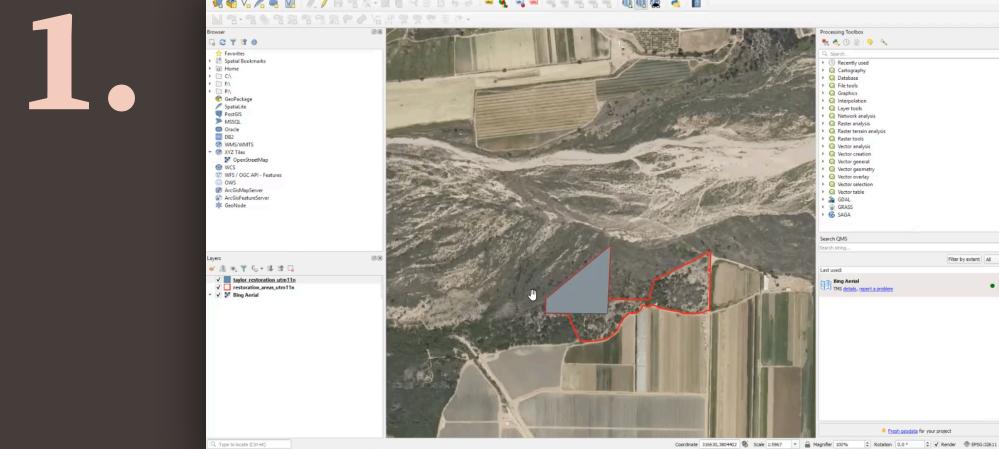


5.

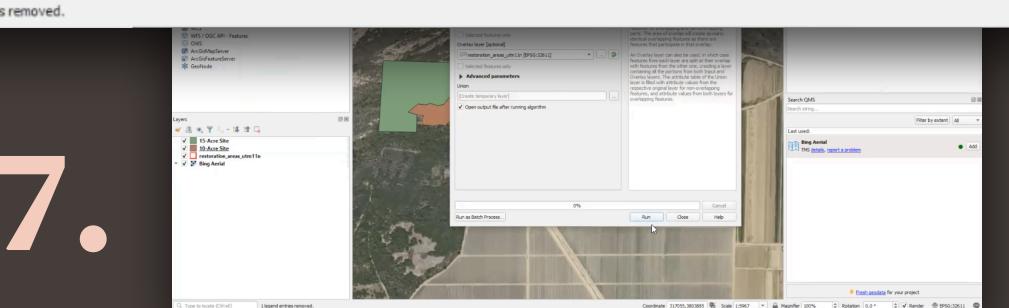
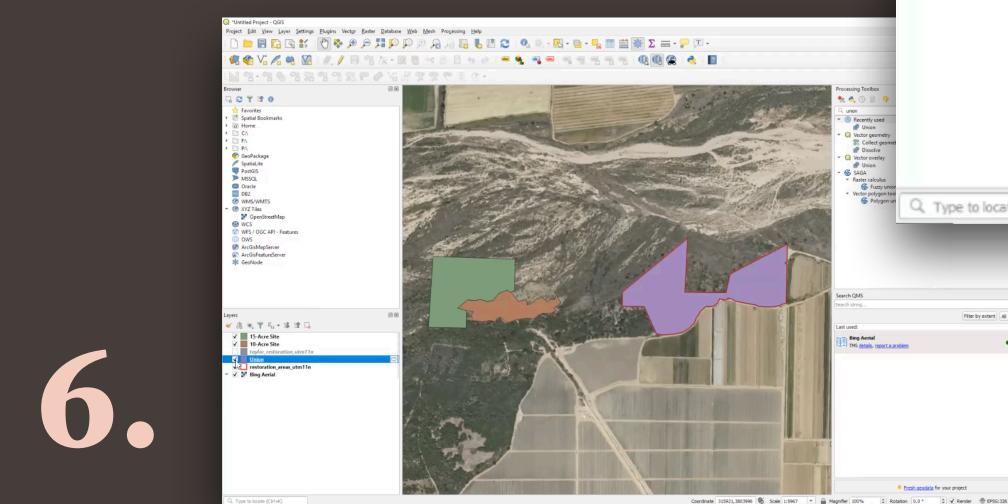
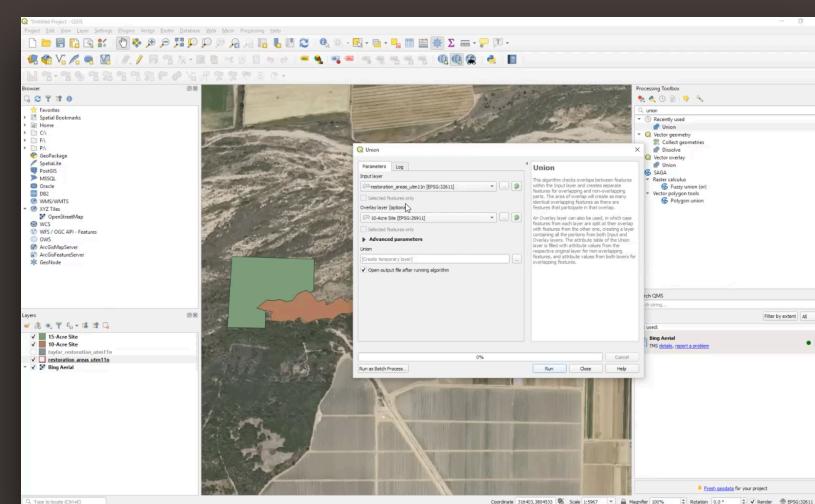
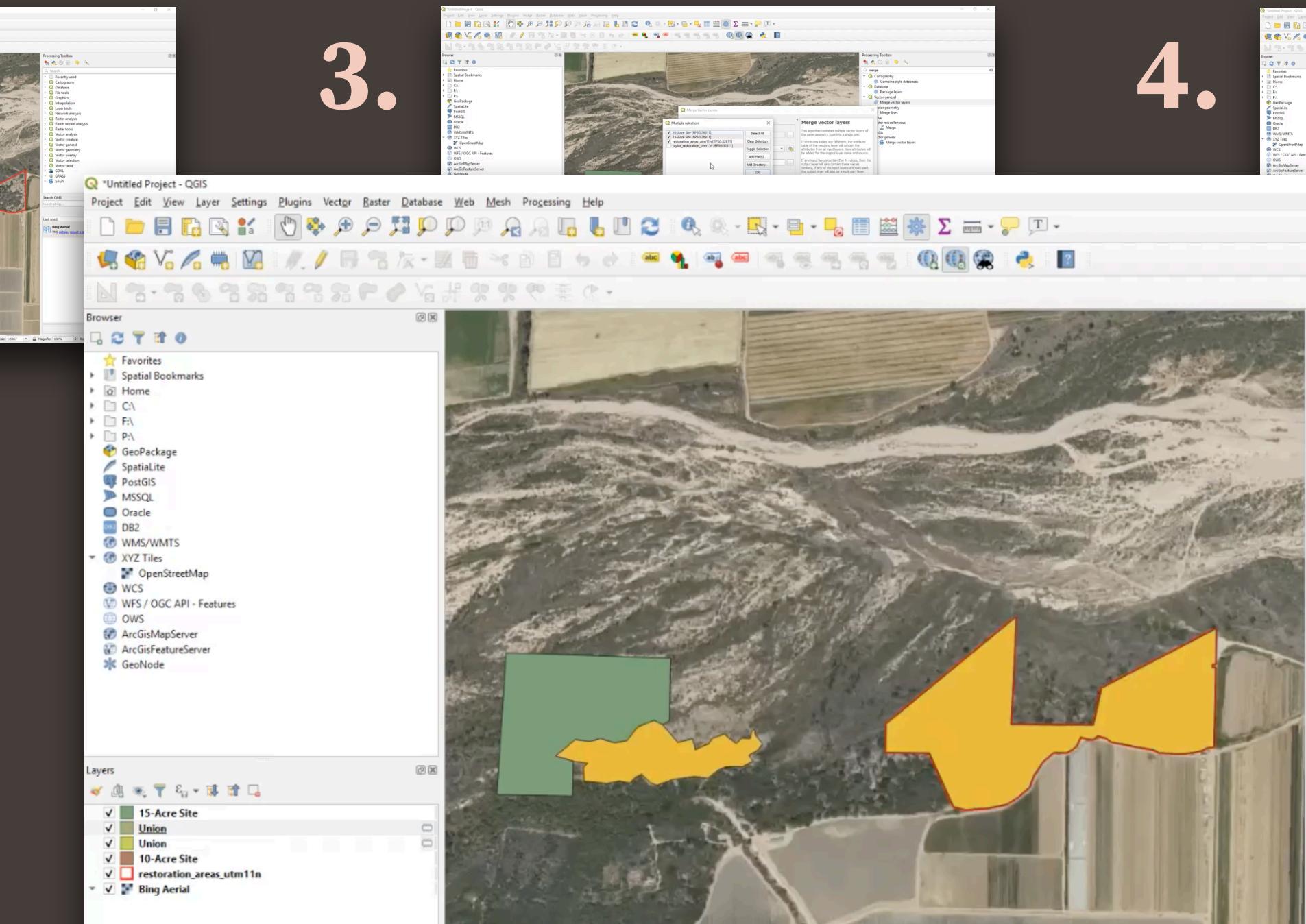


6.

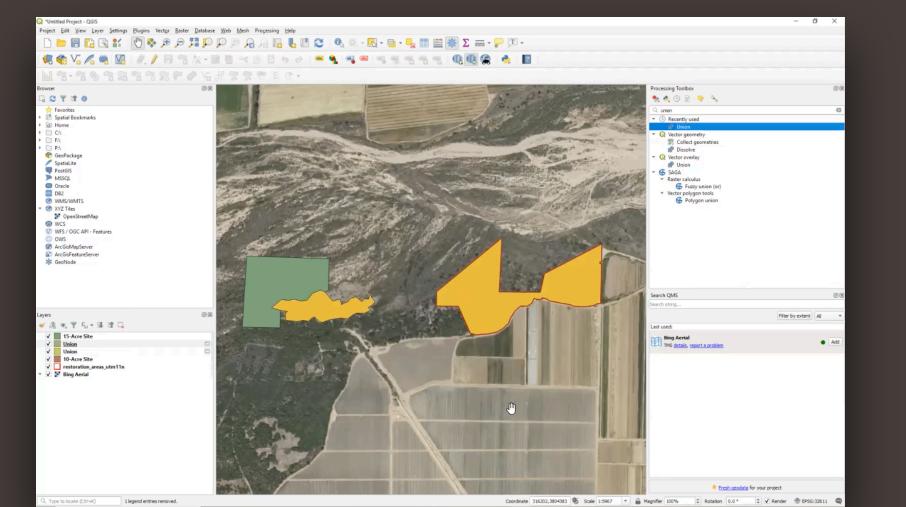
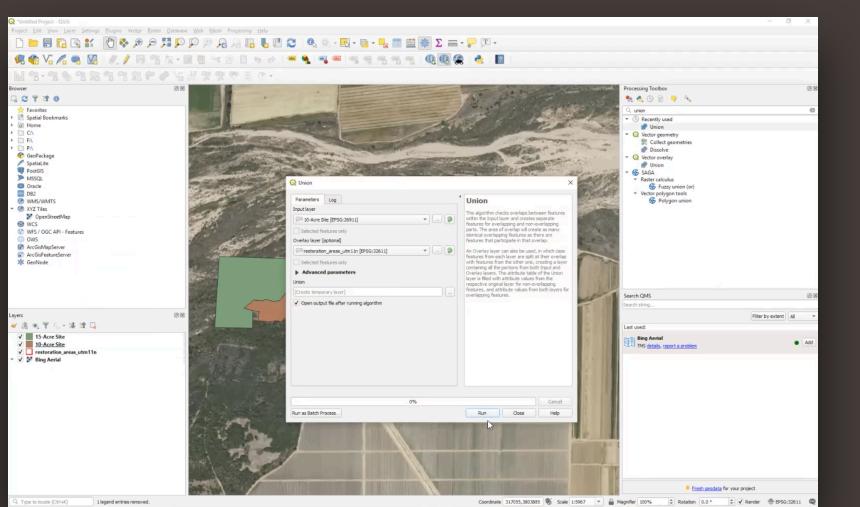
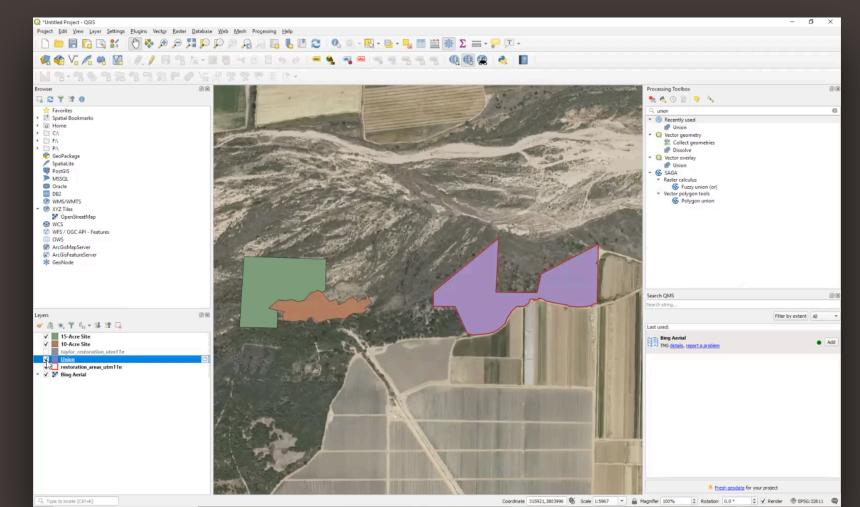
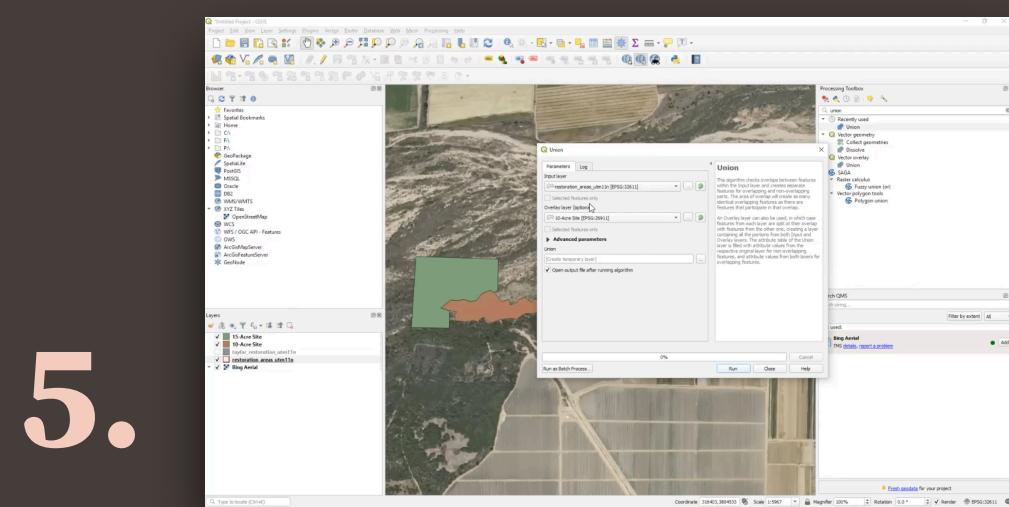
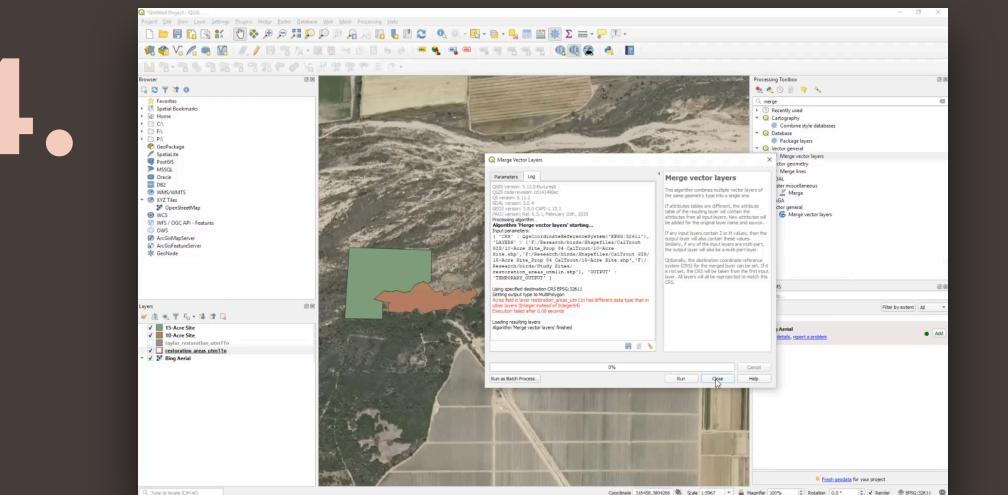
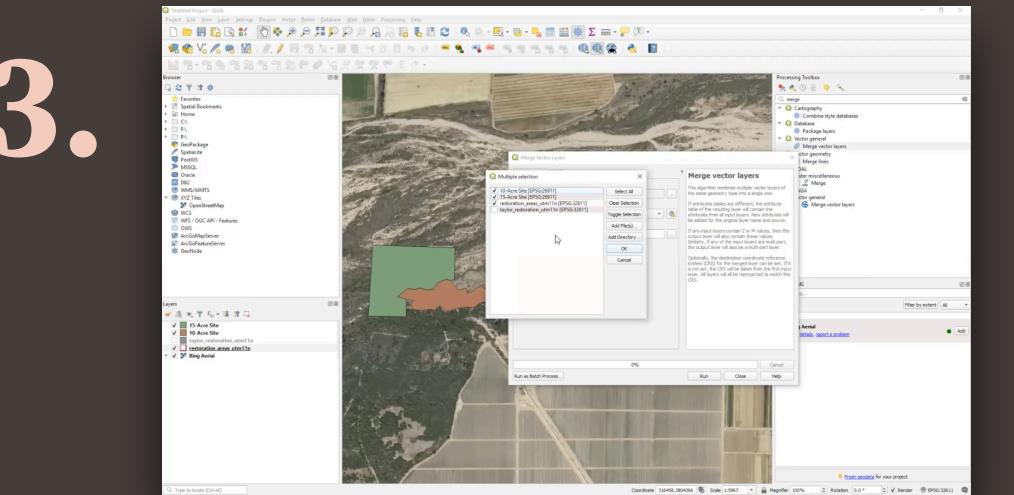
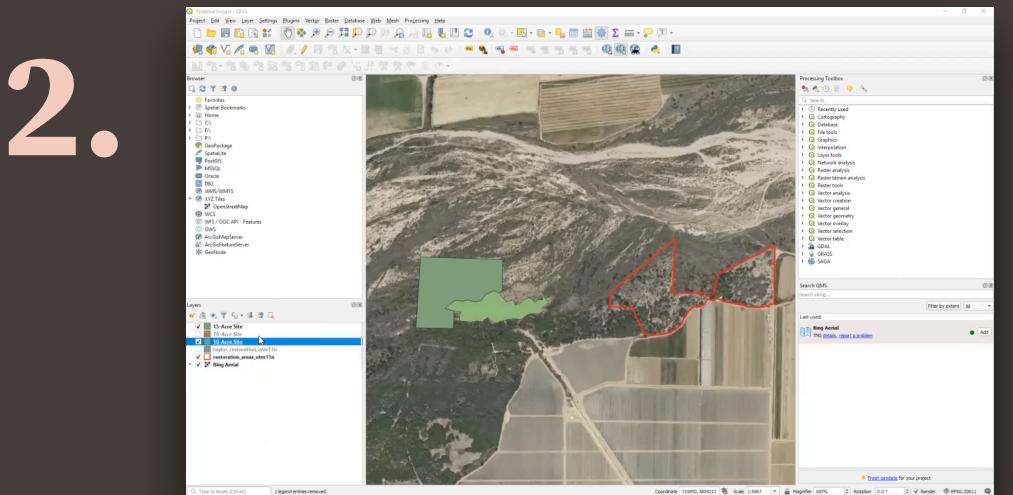
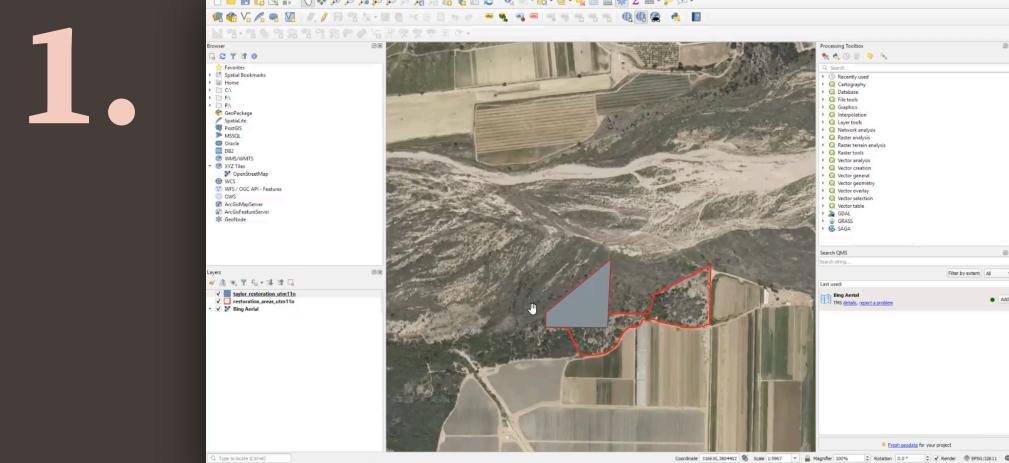
# Creating informal program representations



8. Union operation succeeds and gives expected output.

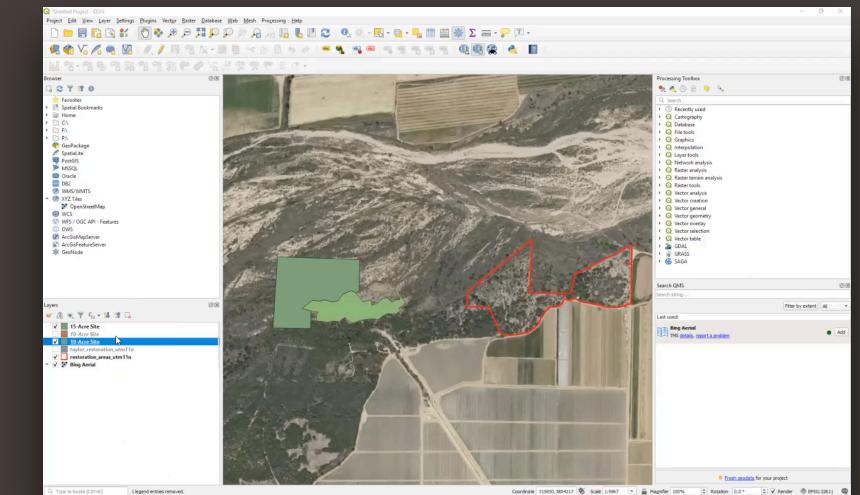
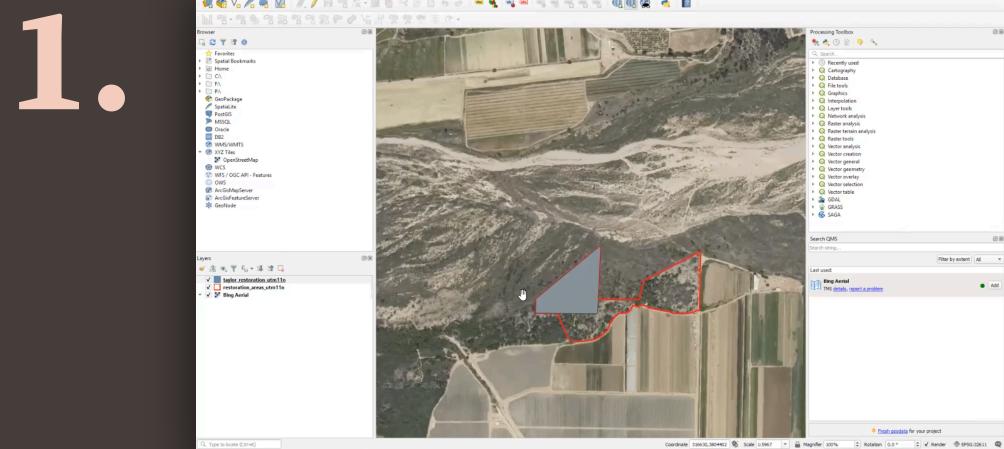


# Creating informal program representations

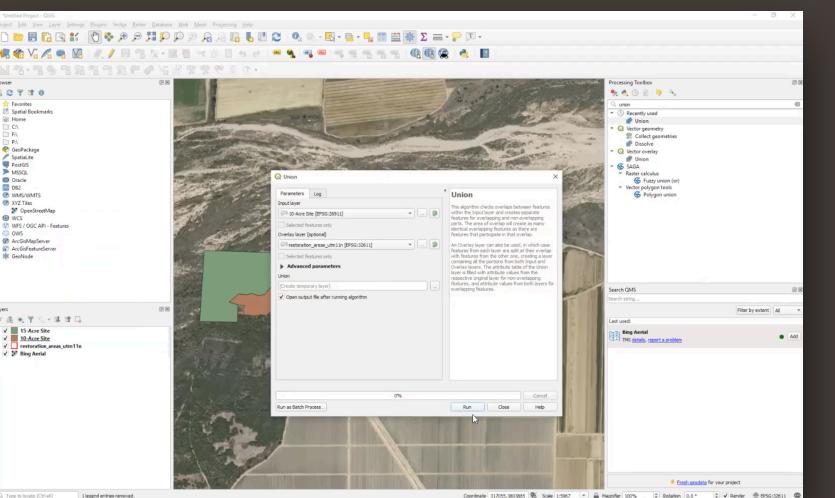


Full trace of user actions

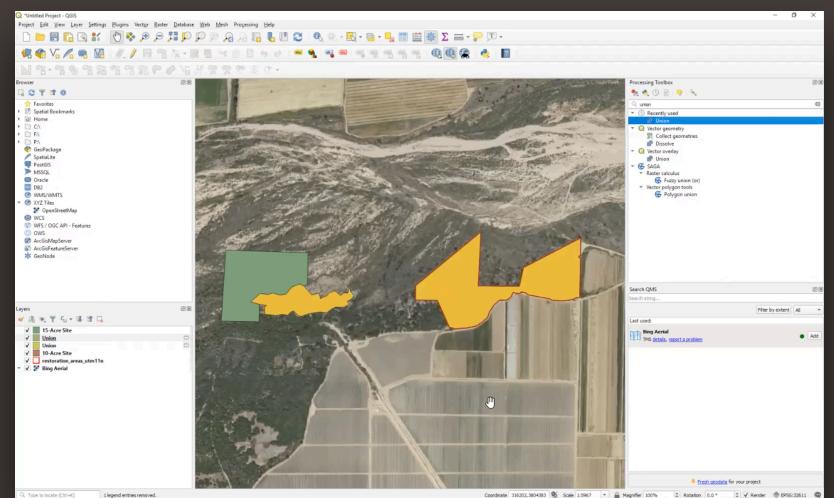
# Creating informal program representations



7.



8.

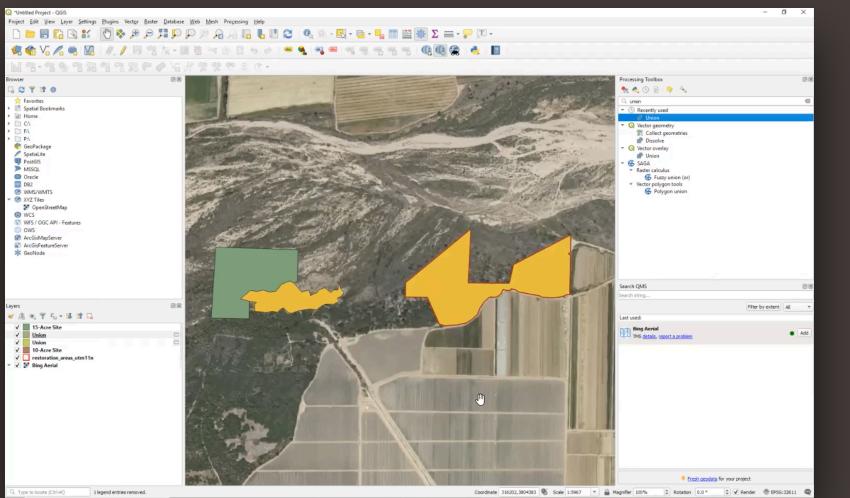


Full trace of desired program

# Creating informal program representations

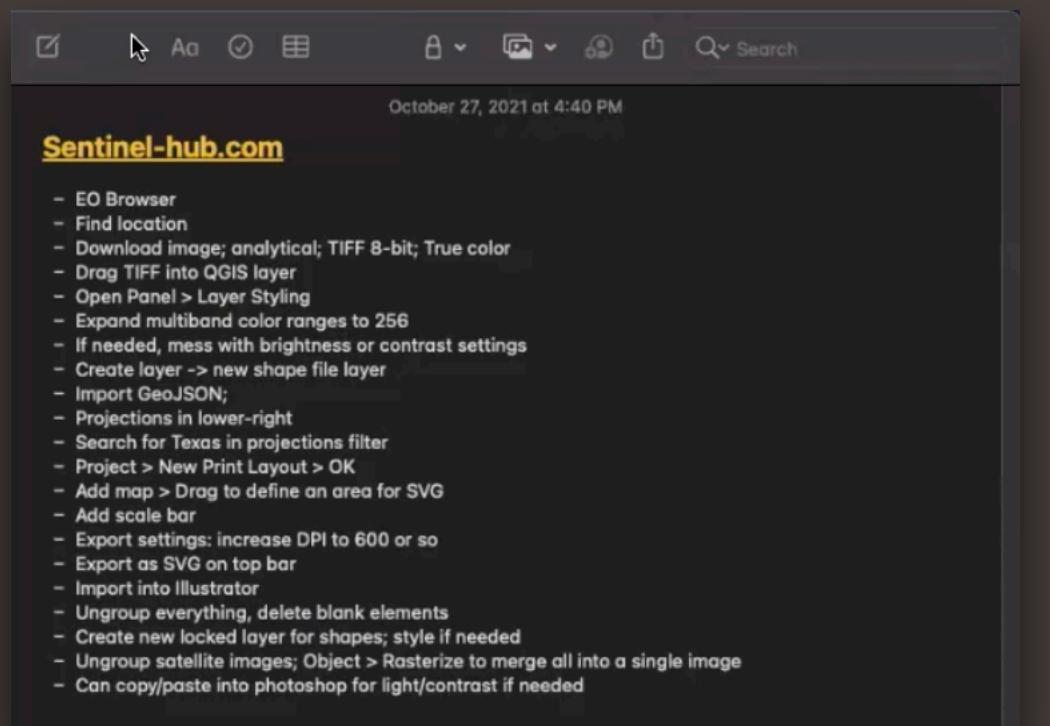
What QGIS actually gives us

8.

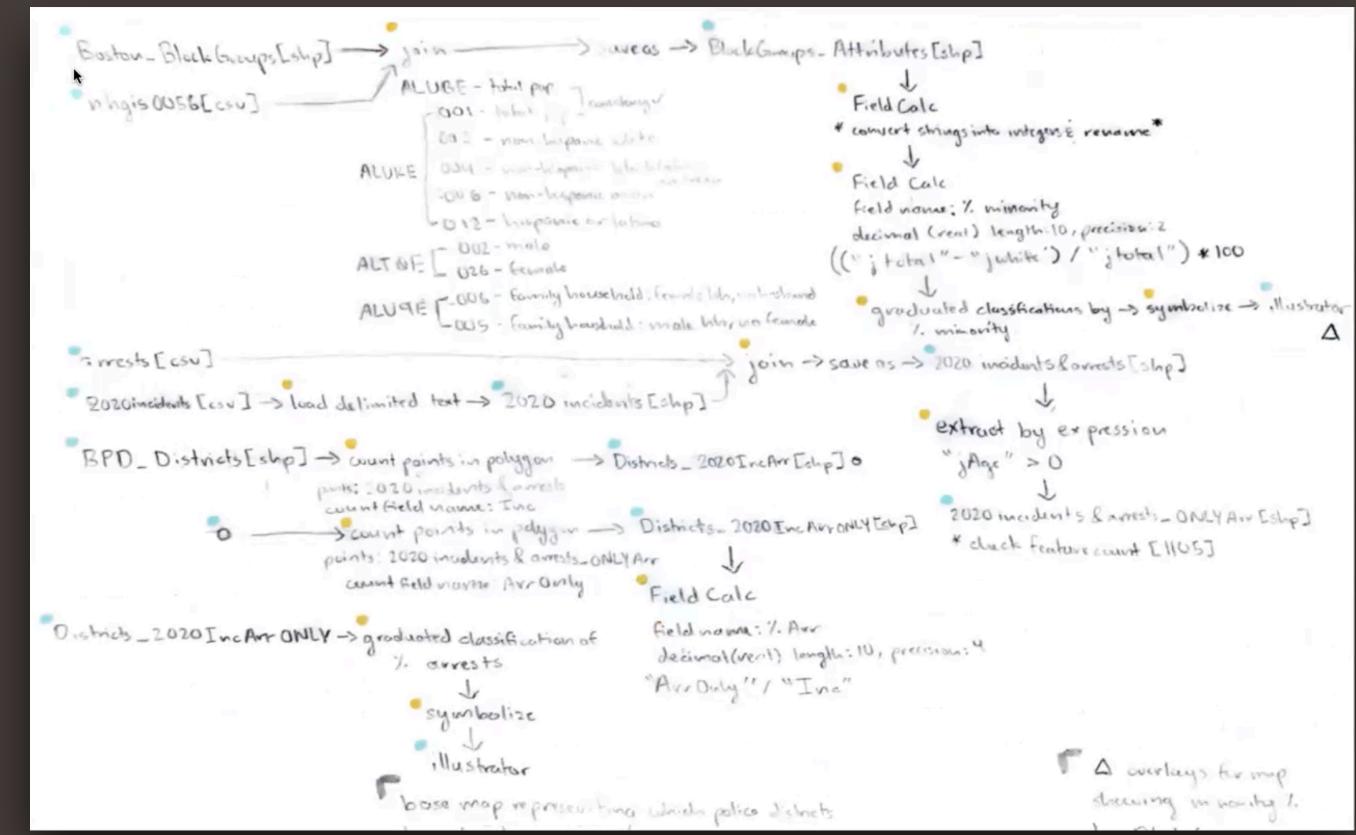


# Creating informal program representations

Participants using GIS software persisted process information about their analysis **outside of GIS software.**



Participant 19 used macOS Notes.



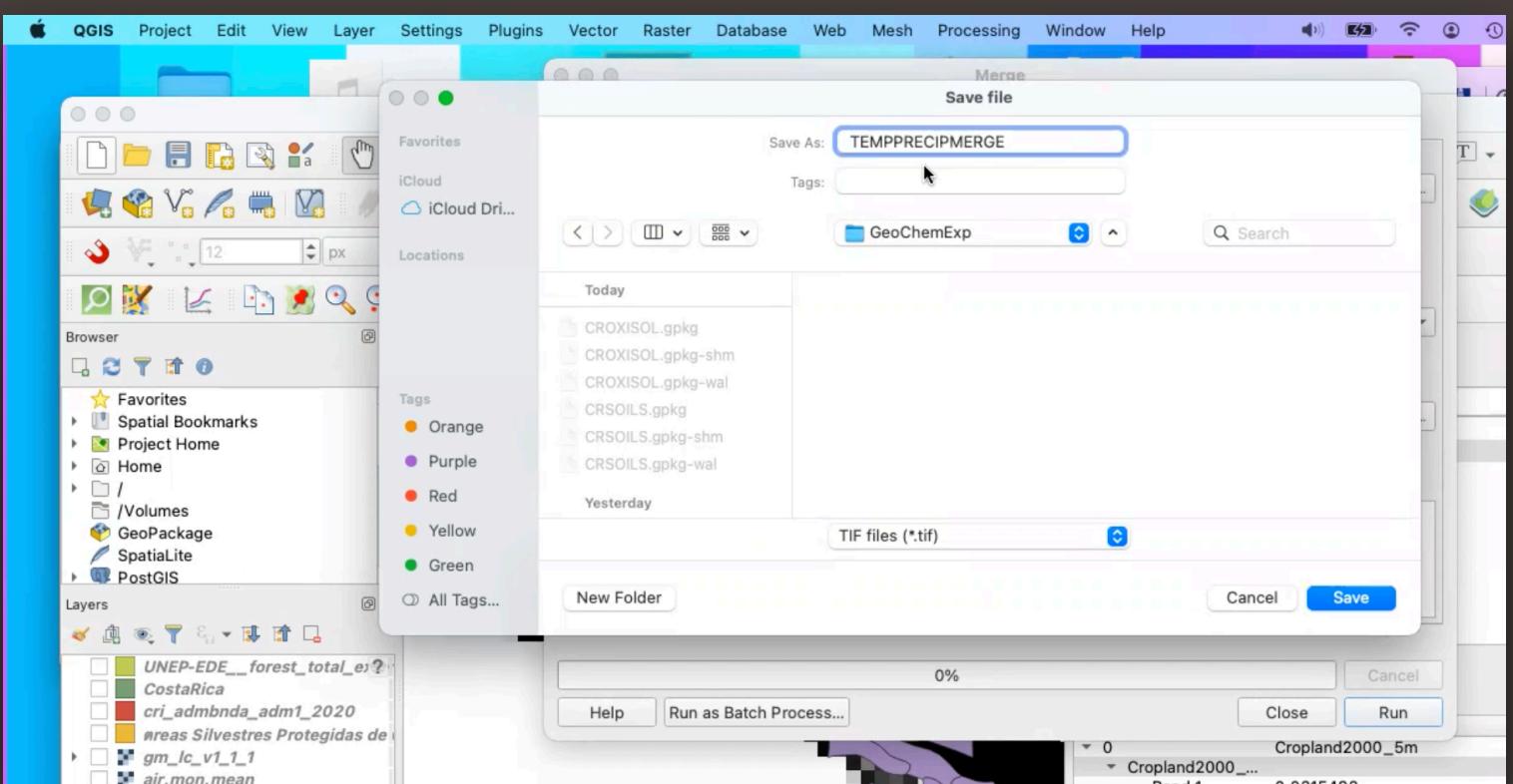
Participant 4 used a hand-drawn diagram, with color distinguishing layers vs. operators.

A1	B	C	D	E	F	G	H	I	J	K
2	Shapefile for Costa Rica									
3	<a href="https://data.humdata.org/dataset/costa-rica-subnational-administrative-boundaries">https://data.humdata.org/dataset/costa-rica-subnational-administrative-boundaries</a>									
4	some climate data									
5	<a href="http://climate.geog.udel.edu/~climate/html_pages/download.html#P2014">http://climate.geog.udel.edu/~climate/html_pages/download.html#P2014</a>									
6	<a href="https://www.ncdc.noaa.gov/data/noaa-global-surface-temperature/v5/access/gridded/">https://www.ncdc.noaa.gov/data/noaa-global-surface-temperature/v5/access/gridded/</a>									
7	<a href="https://www.ncdc.noaa.gov/data/noaa-global-surface-temperature/v5/access/timeseries/">https://www.ncdc.noaa.gov/data/noaa-global-surface-temperature/v5/access/timeseries/</a>									
9	Climate database									
10	<a href="https://climate-arcgis-content.hub.arcgis.com/search?tags=precipitation%2Cstandardized%20precipitation%20index&amp;type=feature%20layer">https://climate-arcgis-content.hub.arcgis.com/search?tags=precipitation%2Cstandardized%20precipitation%20index&amp;type=feature%20layer</a>									
11	<a href="https://www.worldclim.org/data/worldclim21.html">https://www.worldclim.org/data/worldclim21.html</a>	<- used for temp. data, 12 tiff files averaged in raster calculator								
12		number of raster band = grey for $i$ ( $A+B+C+D+E+F)/6$								
13	soil profiles									
14	<a href="https://www.isric.org/explore/soil-geographic-databases">https://www.isric.org/explore/soil-geographic-databases</a>									
15	use this one	<a href="https://data.apps.fao.org/map/catalog/srv/eng/catalog.search#/metadata/446ed430-8383-11db-b9b2-000d939bc5d8">https://data.apps.fao.org/map/catalog/srv/eng/catalog.search#/metadata/446ed430-8383-11db-b9b2-000d939bc5d8</a>								
16										
17	costa rica climate data									
18	not helpful	<a href="https://library.noaa.gov/Collections/Digital-Docs/Foreign-Climate-Data/Costa-Rica-Climate-Data">https://library.noaa.gov/Collections/Digital-Docs/Foreign-Climate-Data/Costa-Rica-Climate-Data</a>								
19										
20	precip	<a href="https://psl.noaa.gov/data/gridded/data.cmap.html#detail">https://psl.noaa.gov/data/gridded/data.cmap.html#detail</a>								
21										
22										

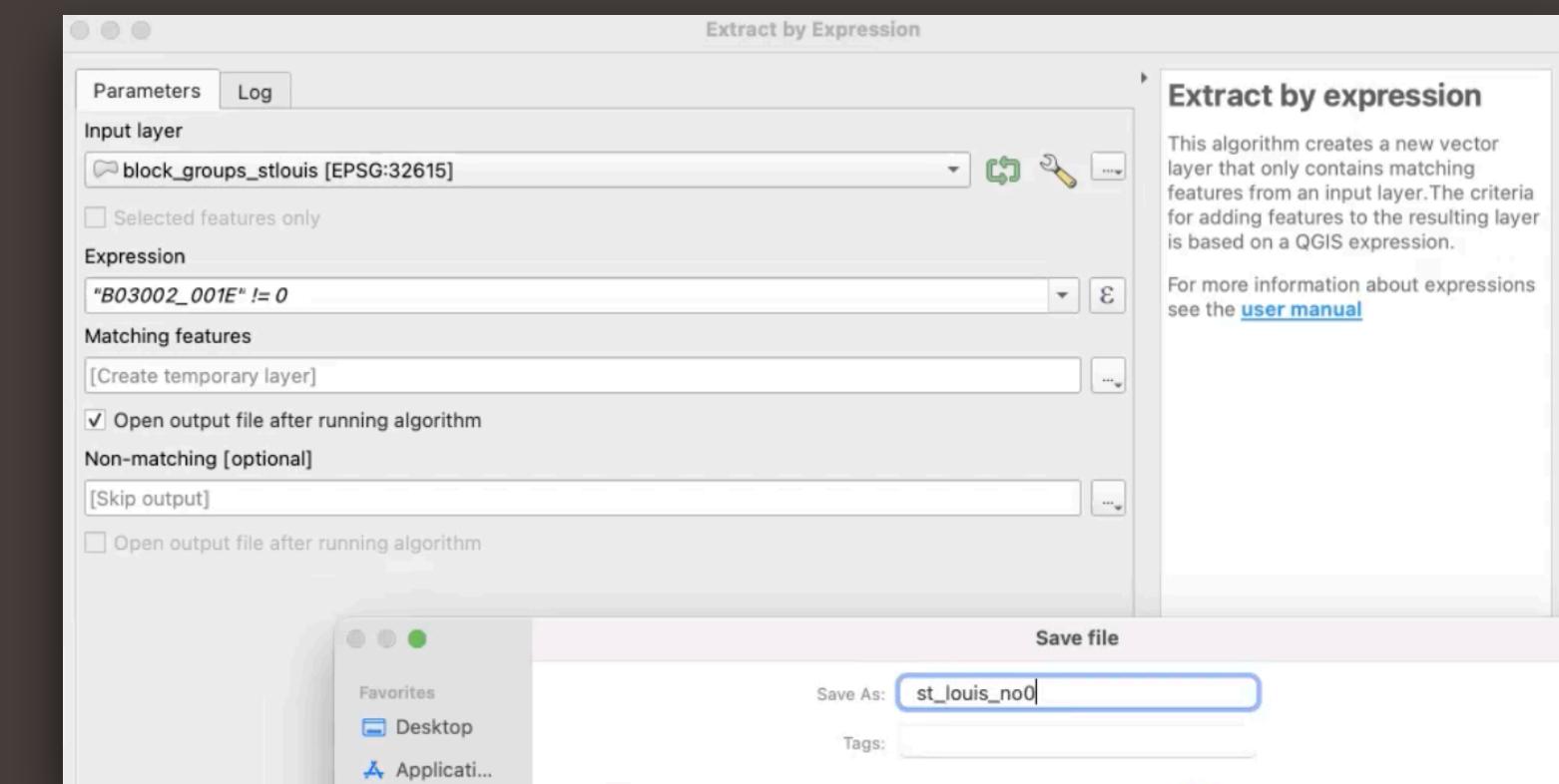
Participant 3 used Google Sheets, adding annotations on data source quality and expressions to use in the **Raster Calculator** tool in QGIS.

# Creating informal program representations

Participants using GIS software left hints to their analysis steps in **layer names**.



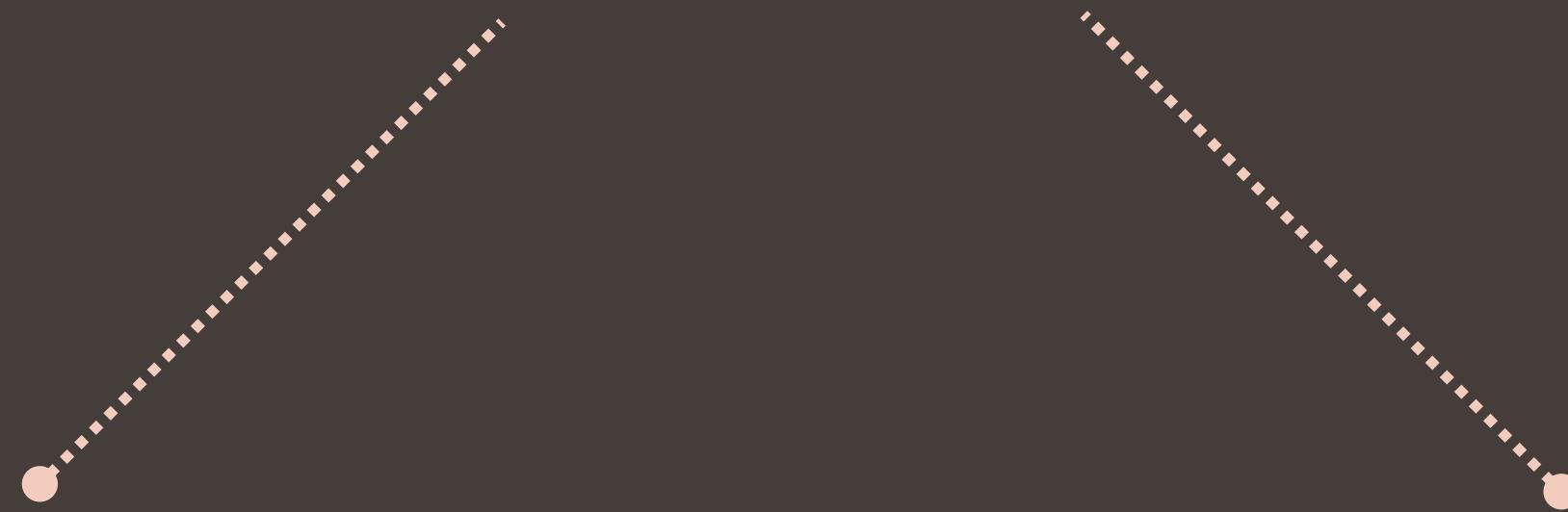
Participant 3 added the suffix “**MERGE**” to the layer generated by merging temperature and precipitation layers.



Participant 4 added the suffix “**\_no0**” when filtering out Census block groups with no population from his dataset.

# Creating informal program representations

Why do GIS software users create informal representations of their processes?

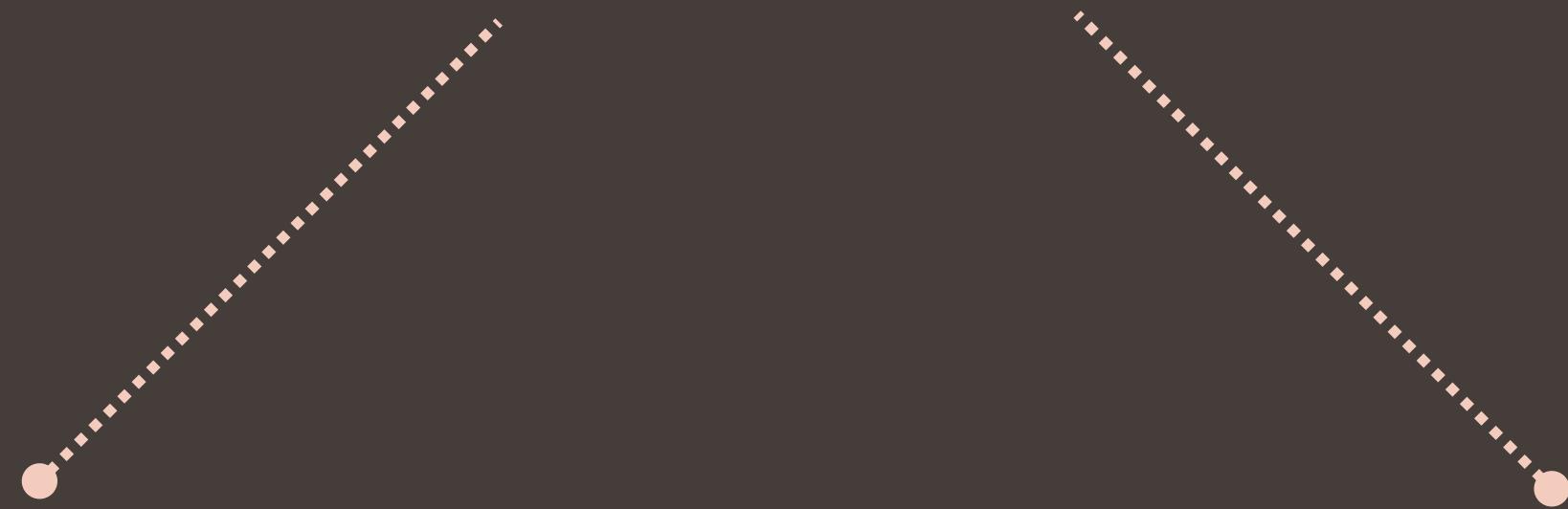


Assists in **recall** of how  
the current program state  
was achieved

Assists in **manual  
reproduction** of the full  
analysis at a future time

# Creating informal program representations

What are the **pitfalls** of working from  
these informal program representations?



Difficult to record a  
**sufficient level of detail**  
for precise reproduction

Not **independently  
verifiable** by other  
researchers or journalists

# Creating informal program representations

“I don’t do any of my processing in [QGIS], mainly because **I like that you can track what you did, the traceability of doing it in Python.** Versus there’s like **none of that if you do it in QGIS.** It’s like you use a plugin or a function, but there’s **no track record of it.**”

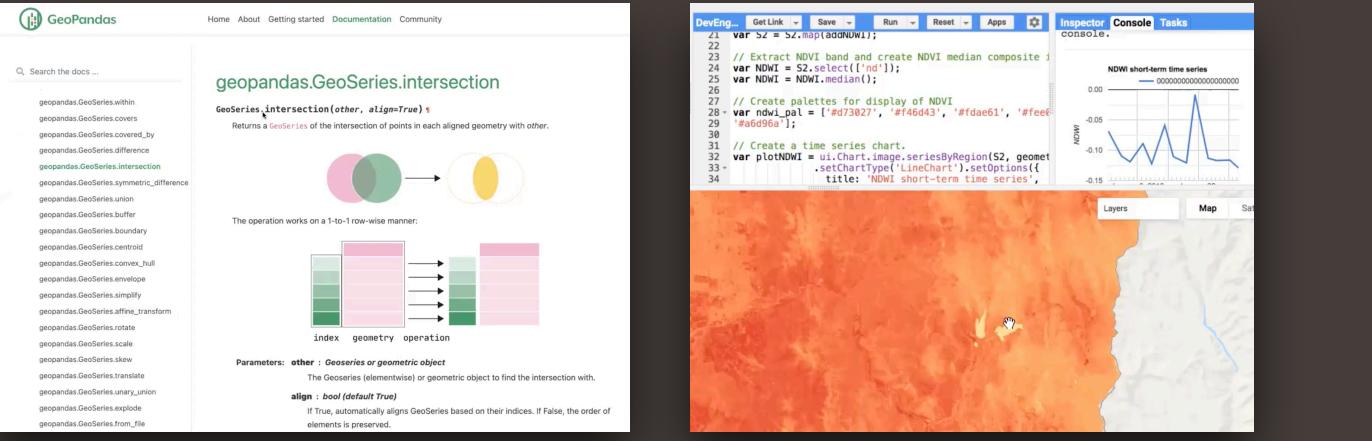
— Participant 17

# Creating informal program representations

“If someone wants to go back and look at my code — ‘Oh, he got this shapefile from here and this shapefile from here and he's pushing them together.’ Whereas if you do that in Arc **you can't really replicate that workflow in the same way.**”

— Participant 15

# Findings



## Reasoning about geospatial operator behavior



GIS Software



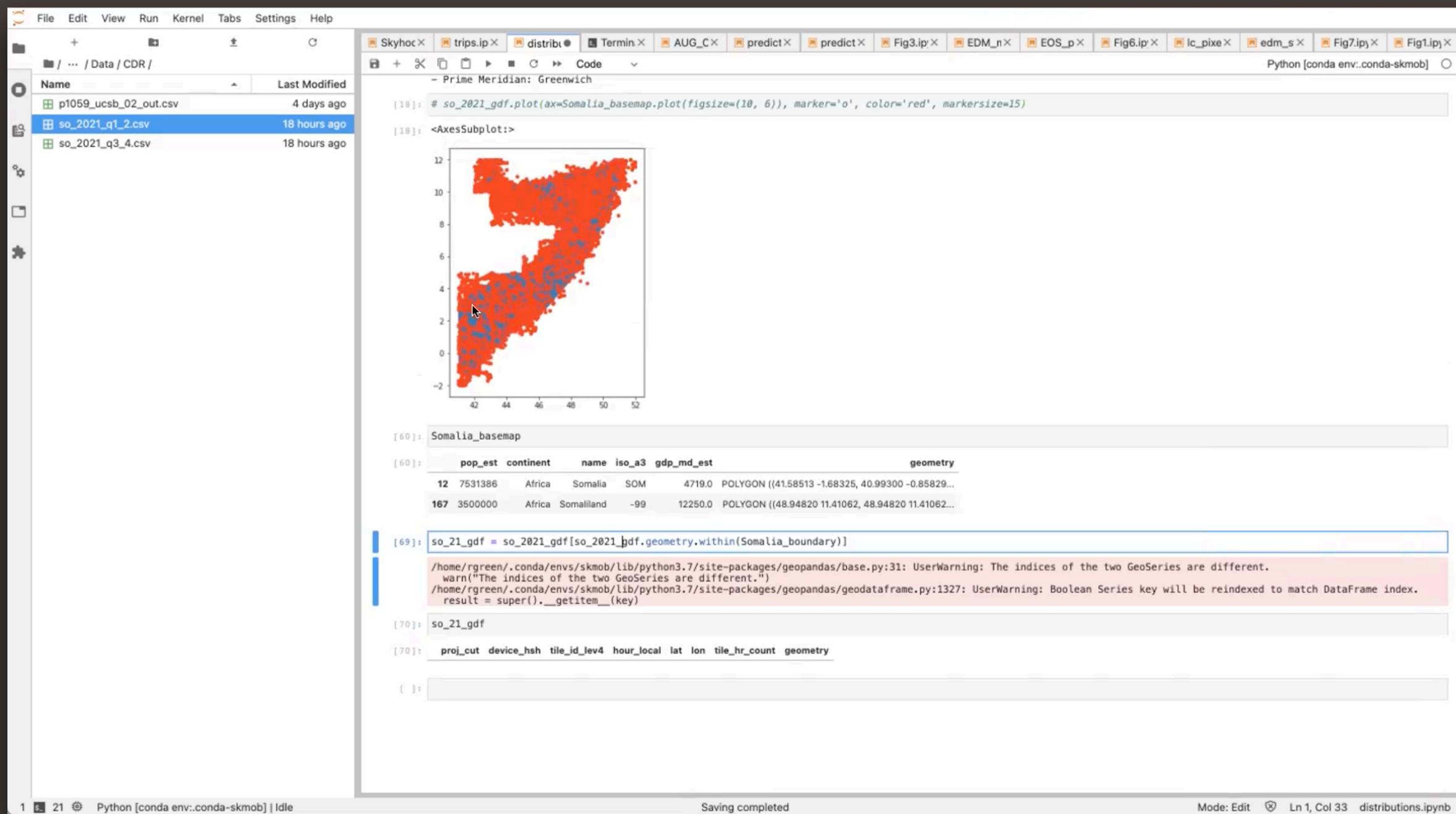
Programming Environments

Geospatial tools have hundreds of operators, and **finding the right operator for an analysis context is difficult.**

# Reasoning about geospatial operator behavior

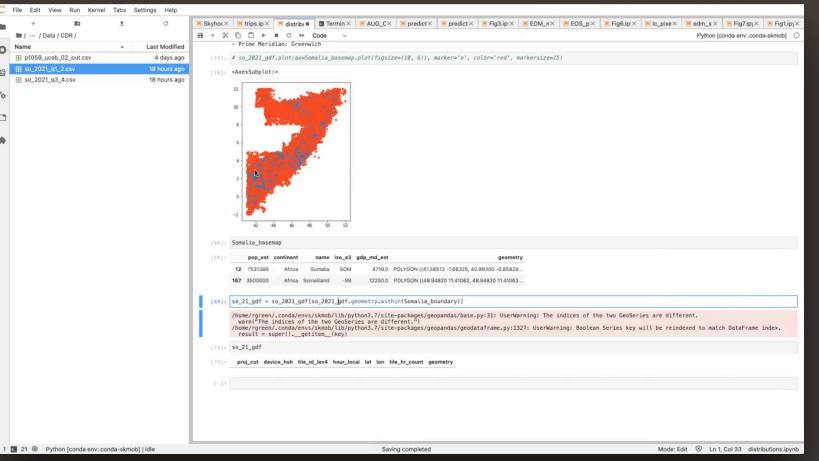
1.

“I want to subset to just the points that are within the polygon.”



# Reasoning about geospatial operator behavior

1.



2.

Copy `.within`  
example code from a  
blog post.

Overview

- Point in Polygon & Intersect
  - How to check if point is inside a polygon?
  - Intersect
- Point in Polygon using Geopandas
  - Spatial join
  - Nearest Neighbour Analysis
  - Exercise 4
  - Exercise 4 hints
- LESSON 5
  - Lesson 5 Overview
  - Static maps
  - Interactive maps with Bokeh
  - Advanced plotting with Bokeh
  - Sharing interactive plots on GitHub
  - Exercise 5
- LESSON 6
  - Overview
  - Automatize data download
  - Reading raster files with Rasterio
  - Visualizing raster layers
  - Masking / clipping raster
  - Raster calculations
  - Creating a raster mosaic
  - Zonal statistics

• Let's first enable shapely.speedups which makes some of the spatial queries running faster.

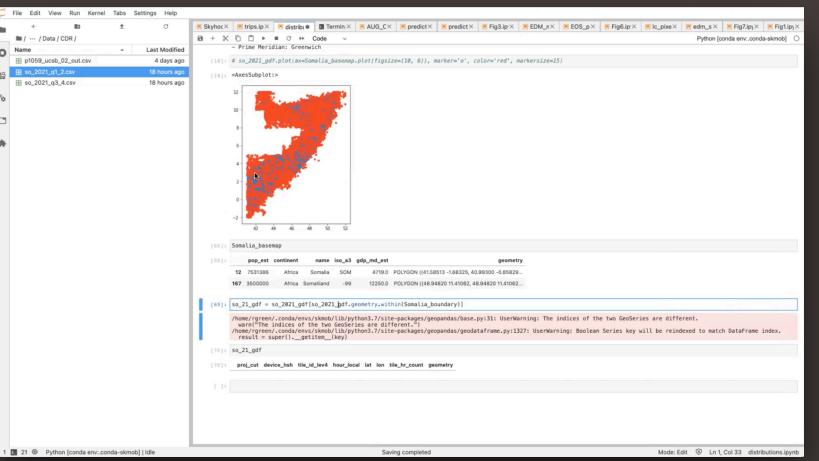
```
In [28]: import shapely.speedups
In [29]: shapely.speedups.enable()
```

• Let's check which Points are within the `southern` Polygon. Notice, that here we check if the Points are `within` the `geometry` of the `southern` GeoDataFrame. Hence, we use the `loc[0, 'geometry']` to parse the actual Polygon geometry object from the GeoDataFrame.

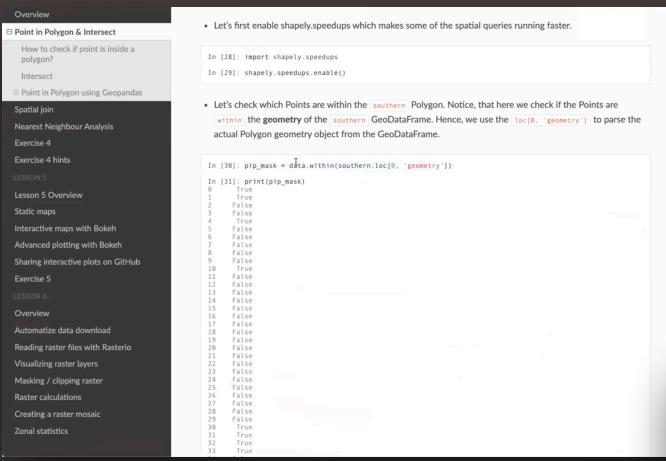
```
In [30]: pip_mask = data.within(southern.loc[0, 'geometry'])
In [31]: print(pip_mask)
0    True
1    True
2   False
3   False
4    True
5   False
6   False
7   False
8   False
9   False
10   True
11  False
12  False
13  False
14  False
15  False
16  False
17  False
18  False
19  False
20  False
21  False
22  False
23  False
24  False
25  False
26  False
27  False
28  False
29  False
30   True
31   True
32   True
33   True
```

# Reasoning about geospatial operator behavior

1.



2.



3.

When that fails, check  
the **geopandas** API  
documentation  
for `.intersection`.

GeoPandas Documentation

geopandas.GeoSeries.intersection

`GeoSeries.intersection(other, align=True)` ↗

Returns a `GeoSeries` of the intersection of points in each aligned geometry with `other`.

The operation works on a 1-to-1 row-wise manner:

Parameters:

- `other : Geoseries or geometric object`
- `align : bool (default True)`

If `True`, automatically aligns `GeoSeries` based on their indices. If `False`, the order of elements is preserved.

v. stable

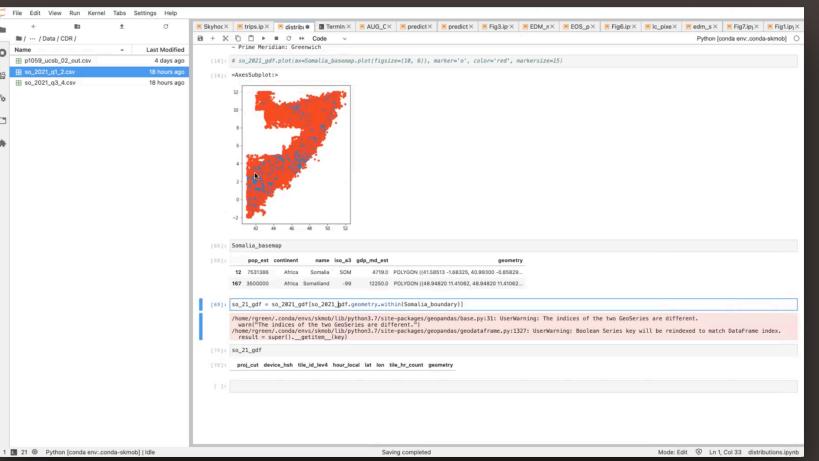
Home About Getting started Documentation Community

Search the docs ...

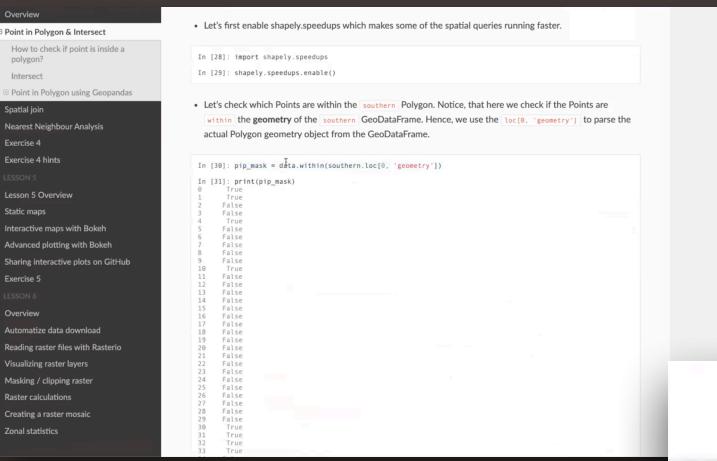
- geopandas.GeoSeries.within
- geopandas.GeoSeries.covers
- geopandas.GeoSeries.covered\_by
- geopandas.GeoSeries.difference
- geopandas.GeoSeries.intersection
- geopandas.GeoSeries.symmetric\_difference
- geopandas.GeoSeries.union
- geopandas.GeoSeries.buffer
- geopandas.GeoSeries.boundary
- geopandas.GeoSeries.centroid
- geopandas.GeoSeries.convex\_hull
- geopandas.GeoSeries.envelope
- geopandas.GeoSeries.simplify
- geopandas.GeoSeries.affine\_transform
- geopandas.GeoSeries.rotate
- geopandas.GeoSeries.scale
- geopandas.GeoSeries.skew
- geopandas.GeoSeries.translate
- geopandas.GeoSeries.unary\_union
- geopandas.GeoSeries.explode
- geopandas.GeoSeries.from\_file

# Reasoning about geospatial operator behavior

1.



2.



3.

The screenshot shows the GeoPandas documentation for the `GeoSeries.intersection` method. It includes a brief description, a diagram illustrating the operation on two GeoSeries, and a code example. The code example demonstrates how to find the intersection of two polygons:

```
>>> s.intersection(Polygon([(0, 0), (1, 1), (0, 1)])
0  POLYGON ((0.00000 0.00000, 0.00000 1.00000, 1...
1  POLYGON ((0.00000 0.00000, 0.00000 1.00000, 1...
2  LINESTRING (0.00000 0.00000, 1.00000 1.00000)
3  POINT (1.00000 1.00000)
4  POINT (0.00000 1.00000)
dtype: geometry
```

Below the code example, there is a note about aligning GeoSeries for comparison:

We can also check two GeoSeries against each other, row by row. The GeoSeries above have different indices. We can either align both GeoSeries based on index values and compare elements with the same index using `align=True` or ignore index and compare elements based on their matching order using `align=False`:

Two diagrams illustrate the difference between `align=True` and `align=False`. In the `align=True` diagram, arrows point from corresponding indices of two stacked GeoSeries to a single pink hexagon labeled "shapely geometry". In the `align=False` diagram, arrows point from all indices of one GeoSeries to all indices of the other, indicating a many-to-many comparison.

Code examples for both cases are provided:

```
>>> s.intersection(s2, align=True)
0  None
1  POLYGON ((0.00000 0.00000, 0.00000 1.00000, 1...
2  POINT (1.00000 1.00000)
3  LINESTRING (2.00000 0.00000, 0.00000 2.00000)
4  POINT EMPTY
```

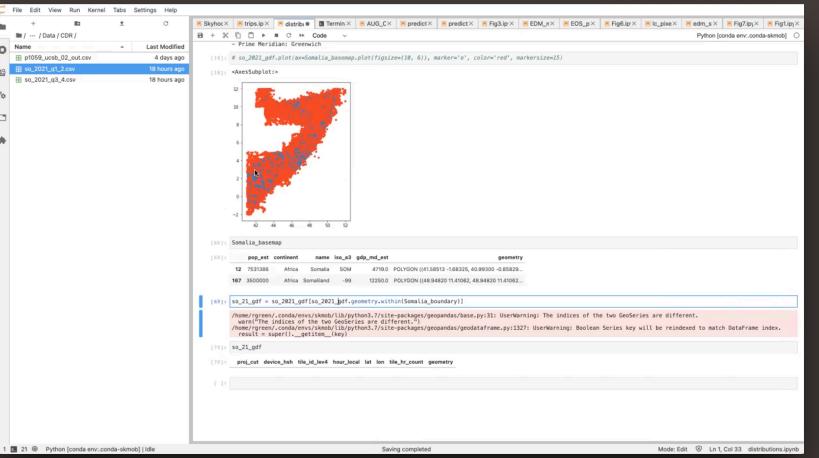
```
>>> s.intersection(s2, align=False)
0  None
1  POLYGON ((0.00000 0.00000, 0.00000 1.00000, 1...
2  POINT (1.00000 1.00000)
3  LINESTRING (2.00000 0.00000, 0.00000 2.00000)
4  POINT EMPTY
```

4.

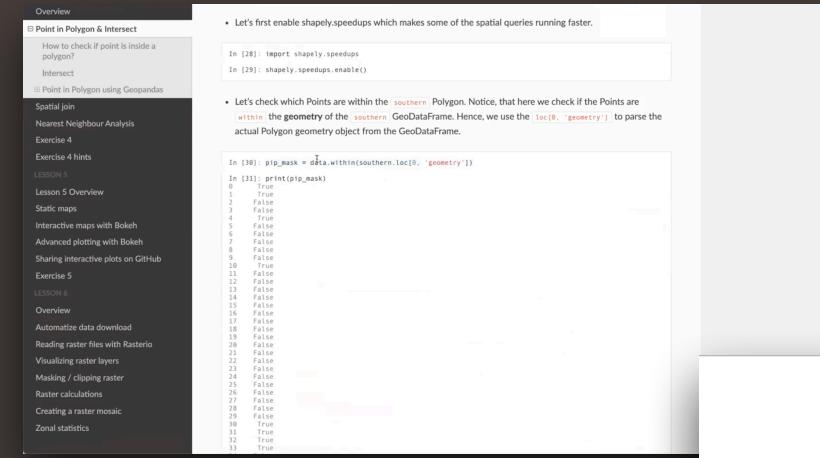
Get confused by the  
geopandas API docs'  
use of tables in lieu of  
geometries.

# Reasoning about geospatial operator behavior

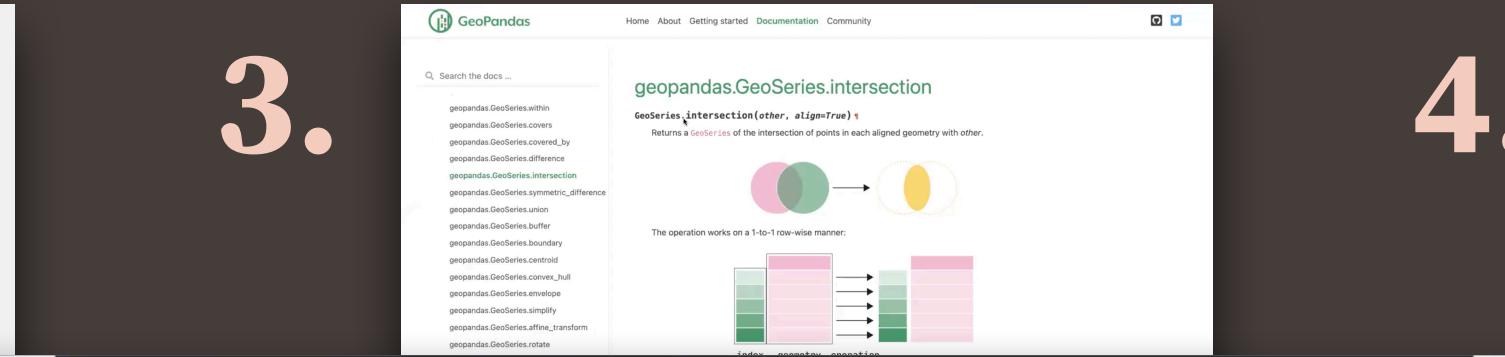
1.



2.



3.



```
p.random.uniform(-126,-64,N)

# Create a geodataframe from numpy arrays
df = pd.DataFrame({'lon':lon, 'lat':lat})
rds' = list(zip(df['lon'],df['lat']))
rds' = df['coords'].apply(Point)
points = gpd.GeoDataFrame(df, geometry='coords', crs=counties.crs)

# Perform spatial join to match points and polygons
Polys = gpd.tools.sjoin(points, counties, op="within", how='left')
```

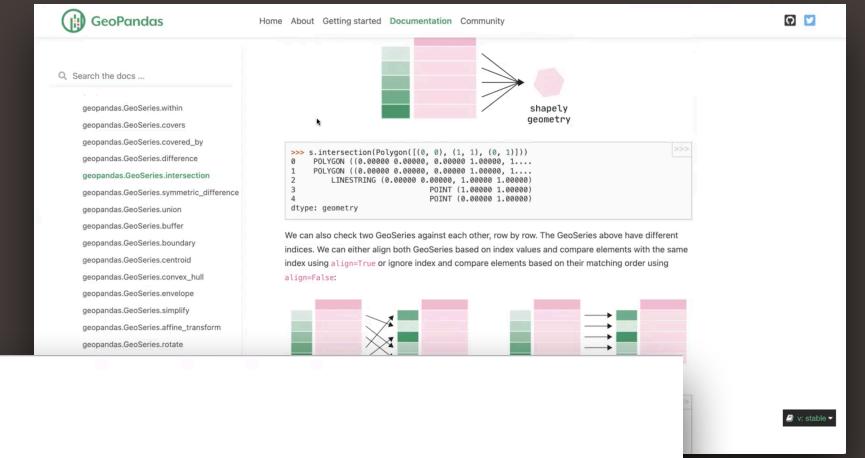
That's it, we now have matched points and polygons. For example use in how to retrieve the points from a specific polygon, we can continue as follows:

```
# Example use: get points in Los Angeles, CA.
pnt_LA = points[pointInPolys.NAME=='Los Angeles']

# Plot map with points in LA in red
base = counties.boundary.plot(linewidth=1, edgecolor="black")
points.plot(ax=base, linewidth=1, color="blue", markersize=1)
pnt_LA.plot(ax=base, linewidth=1, color="red", markersize=8)
plt.show()
```

This will produce the following image:

4.



On this page:

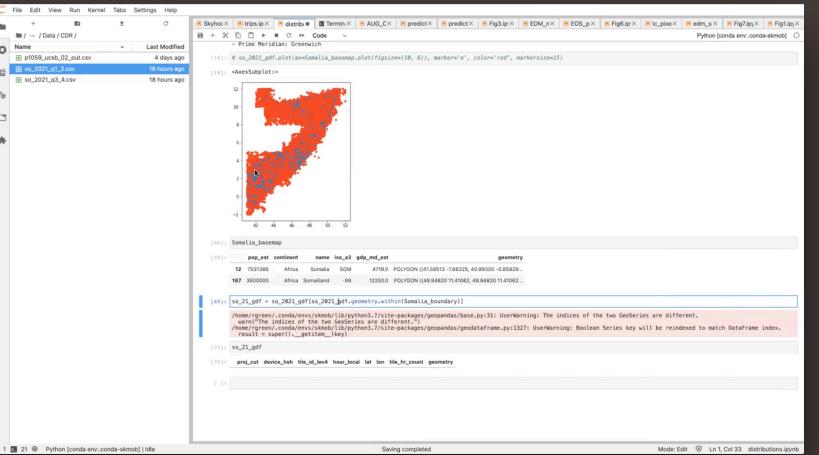
- How to check if a point is inside a polygon in Python
- How to get a list of points inside a polygon in Python
- Speeding up Geopandas point-in-polygon tests
- Use Rtree
- Use the optimized PyGEOS library
- Final remarks

5.

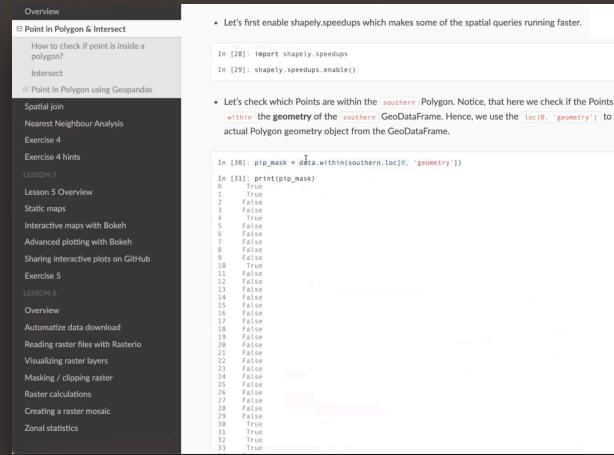
Copy `.sjoin` example code from a different blog post.

# Reasoning about geospatial operator behavior

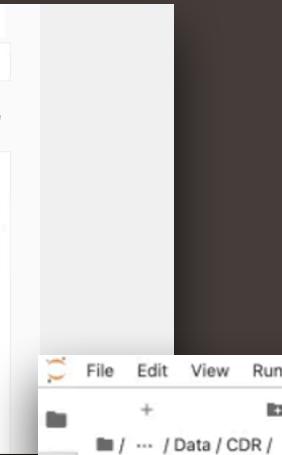
1.



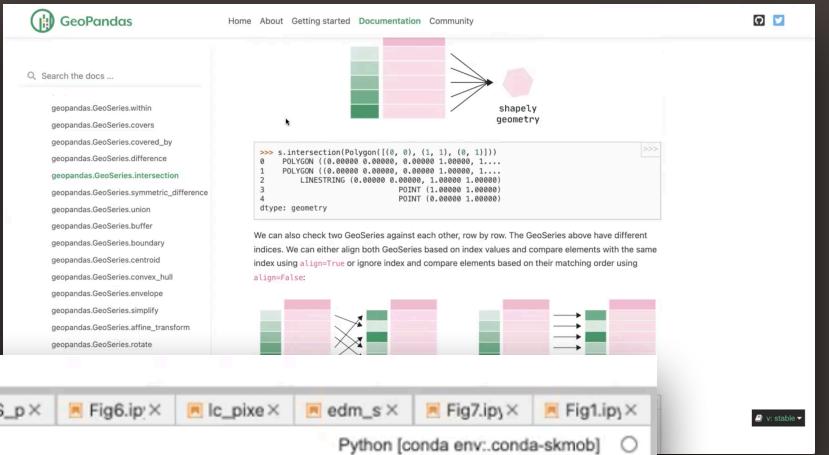
2.



3.

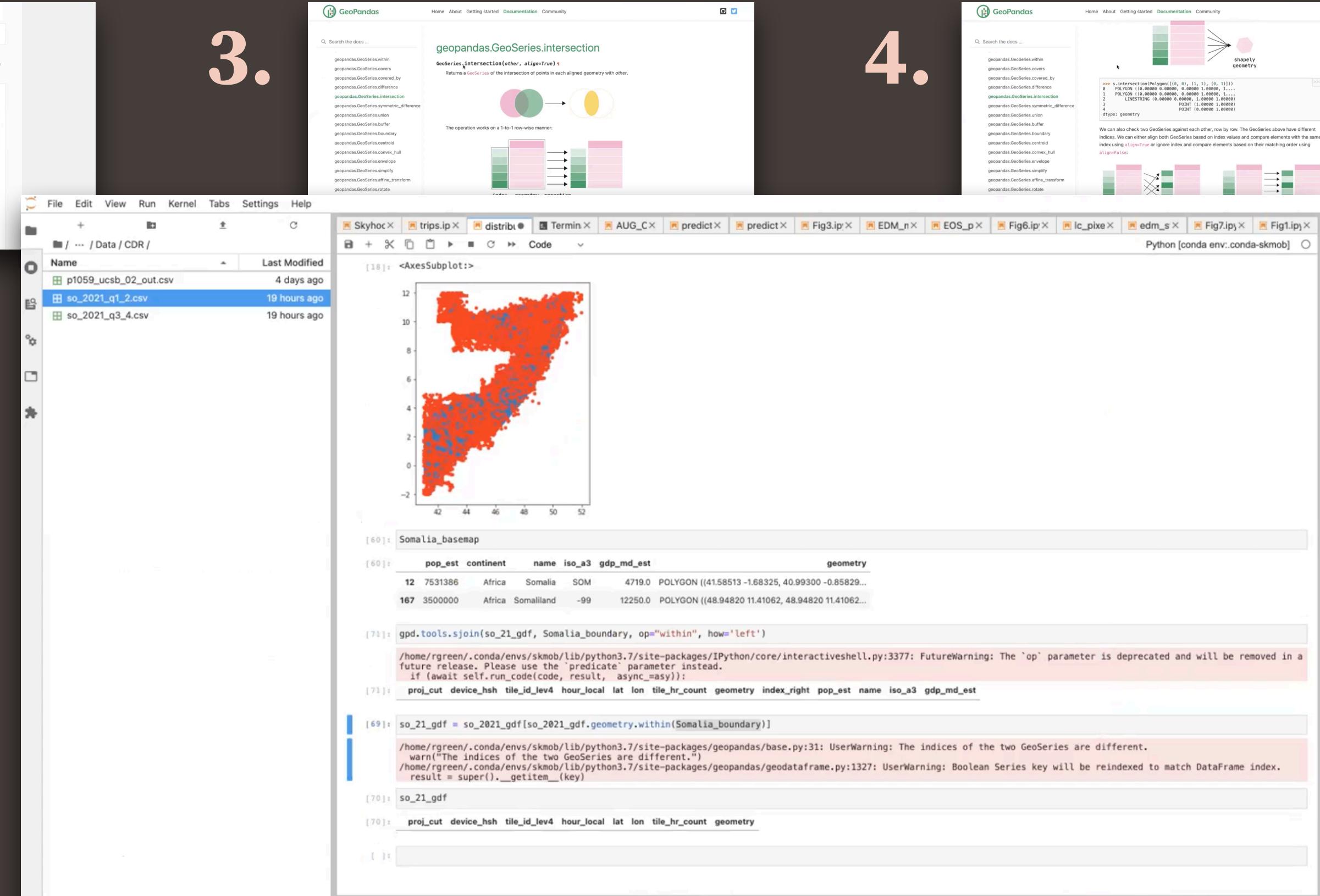


4.

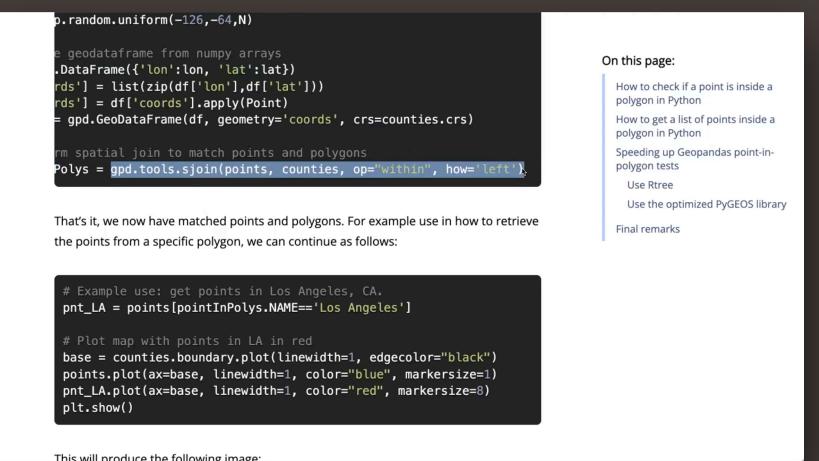


6.

Adapt copied .sjoin  
for the notebook's  
context. Notice a  
runtime warning.

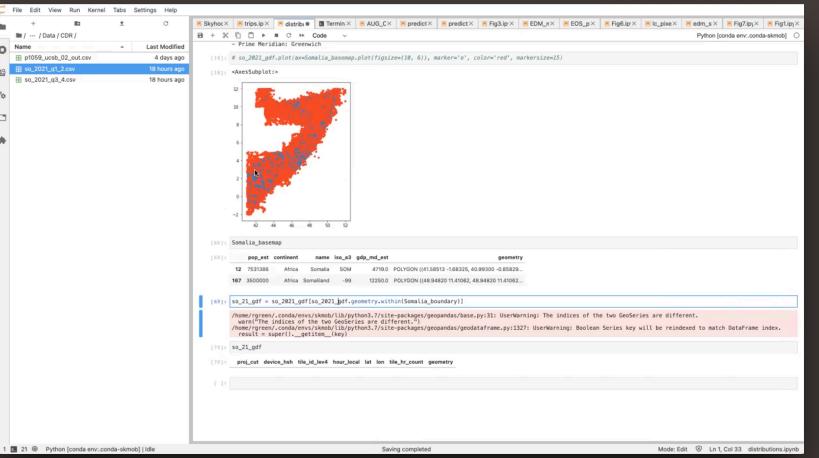


5.

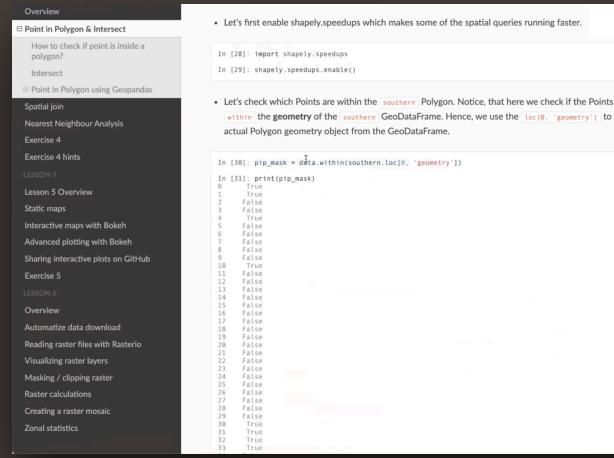


# Reasoning about geospatial operator behavior

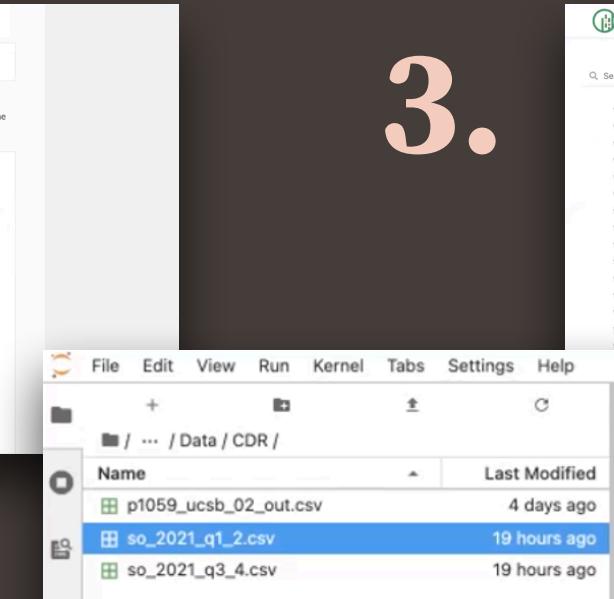
1.



2.



3.



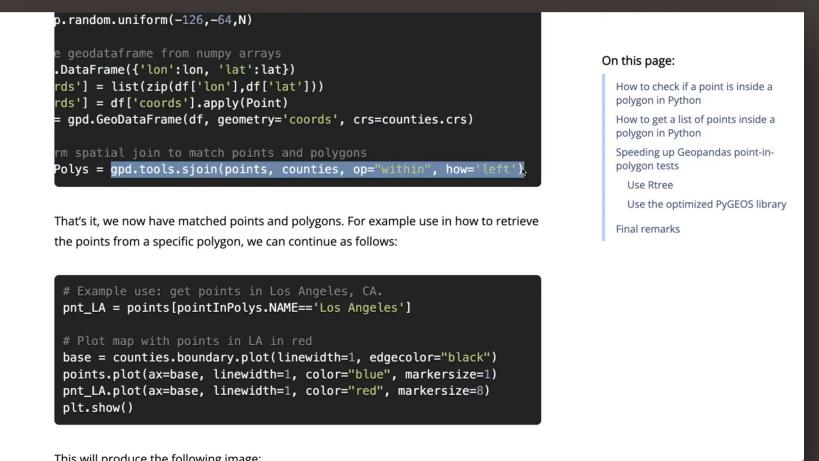
4.



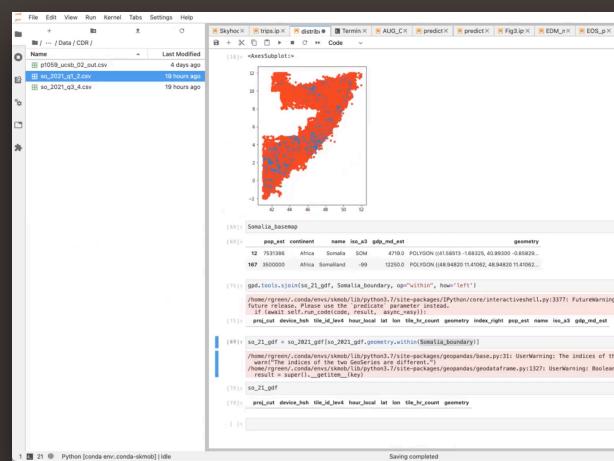
7.

Change op to predicate  
to fix warning. The  
result is an empty  
GeoDataFrame 😱.

5.

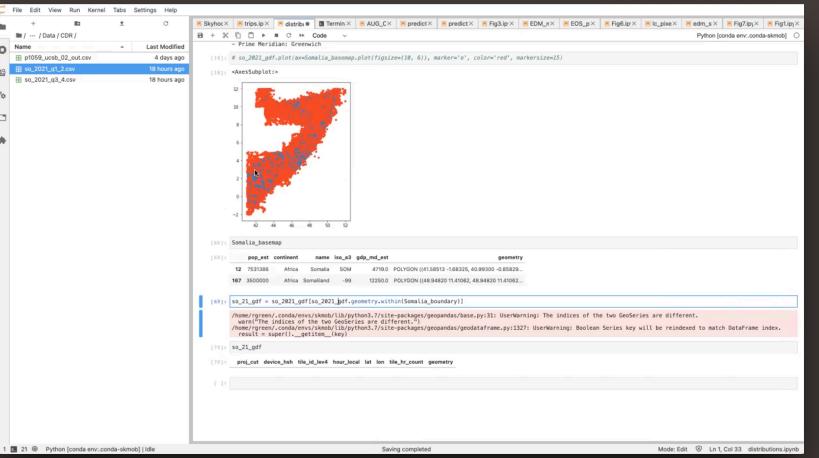


6.

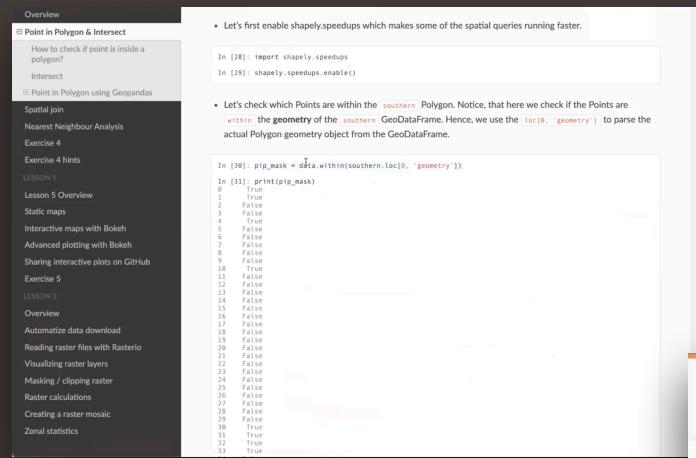


# Reasoning about geospatial operator behavior

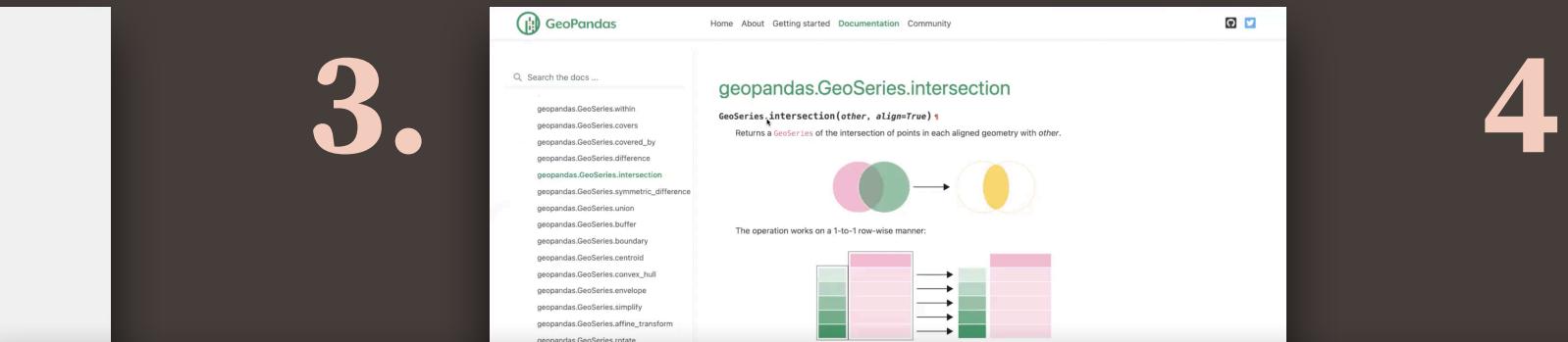
1.



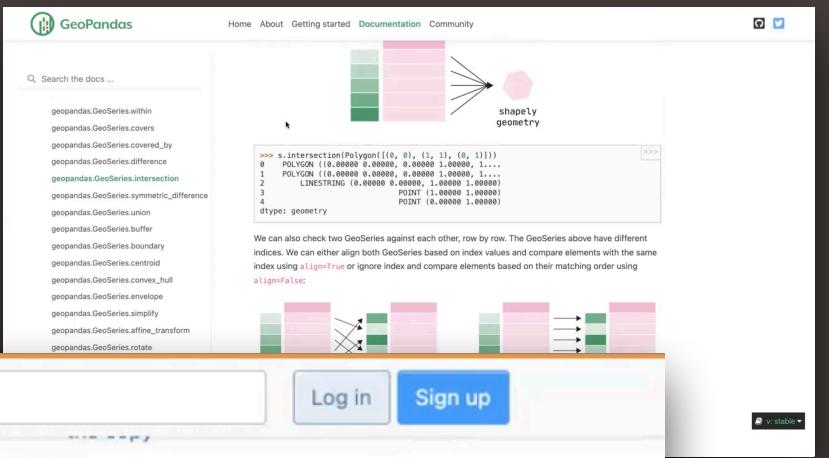
2.



3.



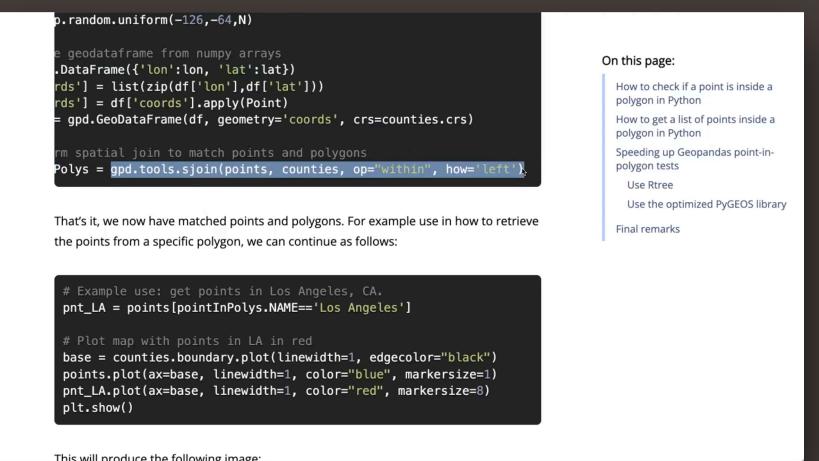
4.



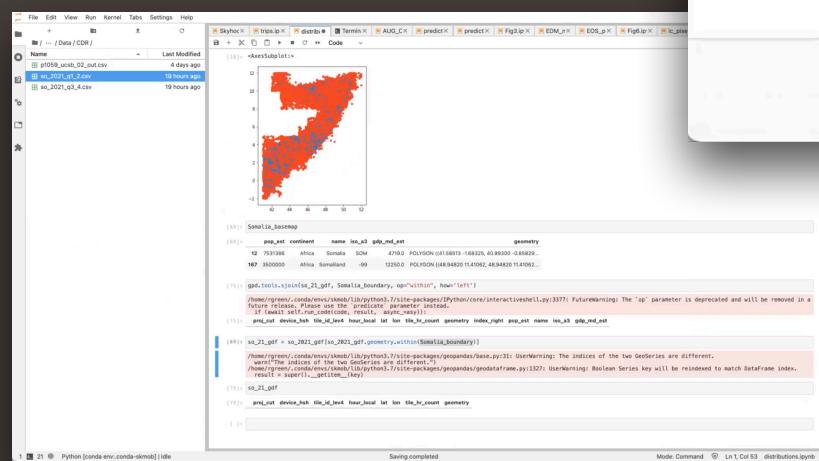
8.

Go to StackOverflow post. “I feel really silly. I waste a lot of time doing things like this.”

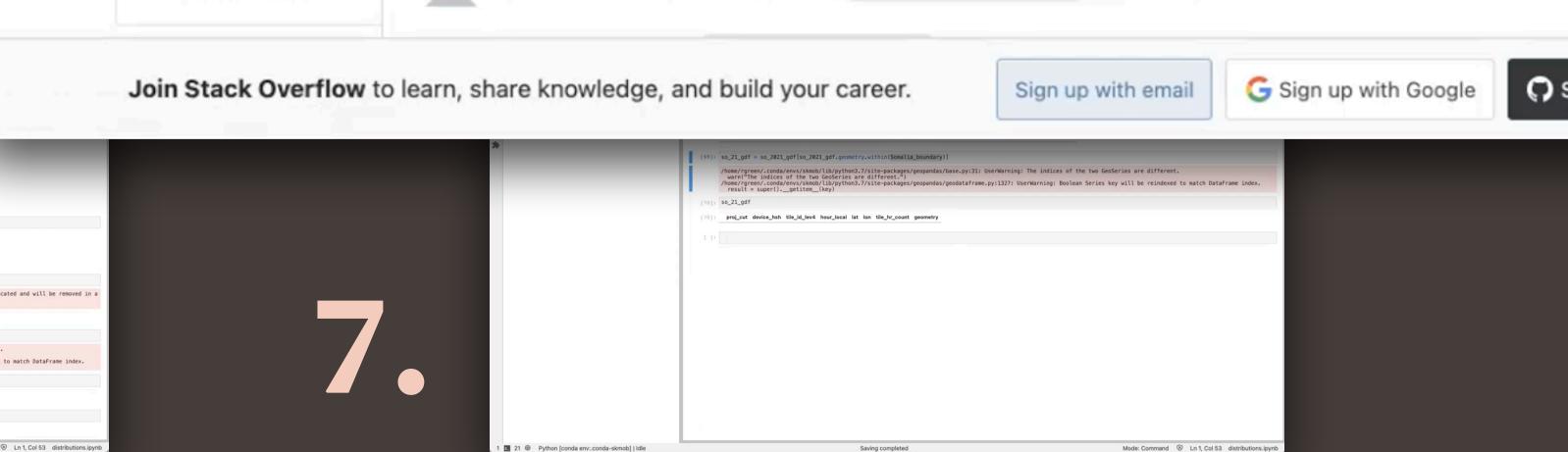
5.



6.



7.

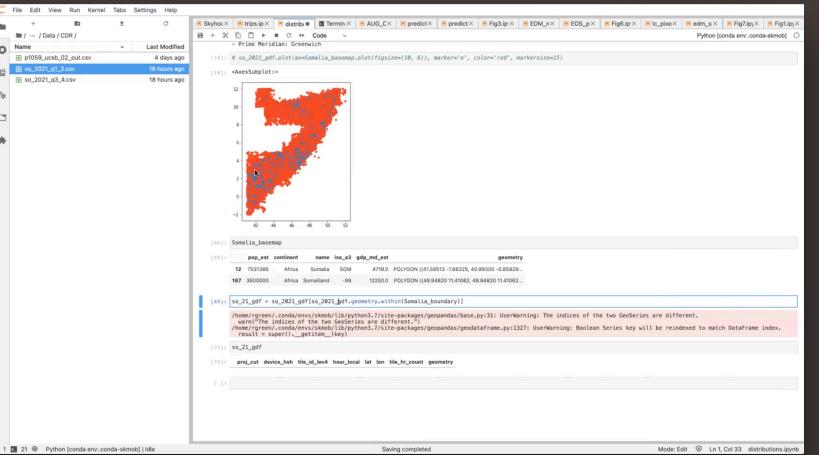


A screenshot of a StackOverflow post by user emax. The question asks how to extract LineString data from a GeoDataFrame and match it with a Polygon. The accepted answer provides a solution using the .within() method.

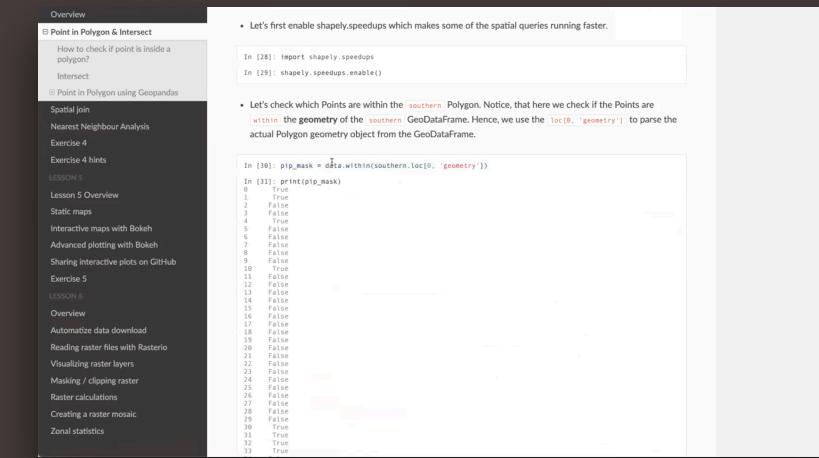
A screenshot of a StackOverflow post by user emax. The question asks what part of speech "freezing" is in the sentence "He will not see again the freezing kitchenhouse...". The accepted answer provides a detailed explanation of the grammatical analysis.

# Reasoning about geospatial operator behavior

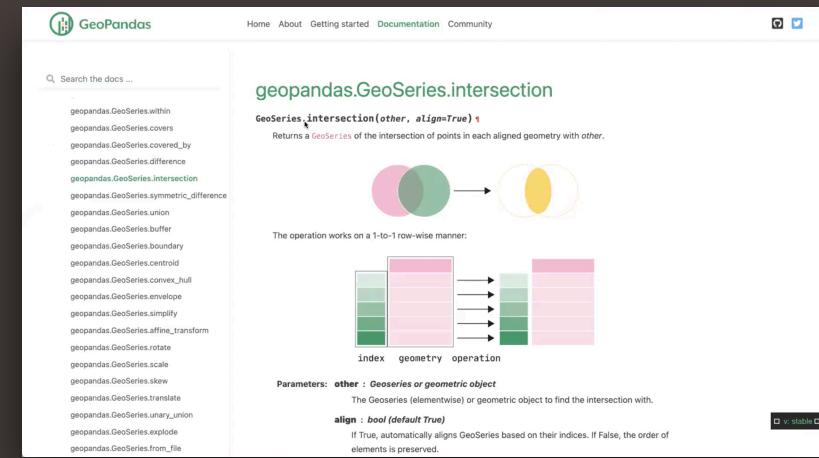
1.



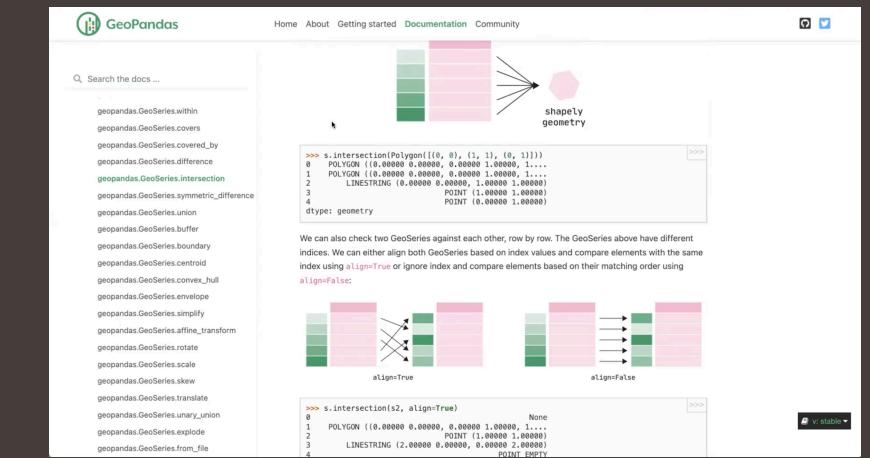
2.



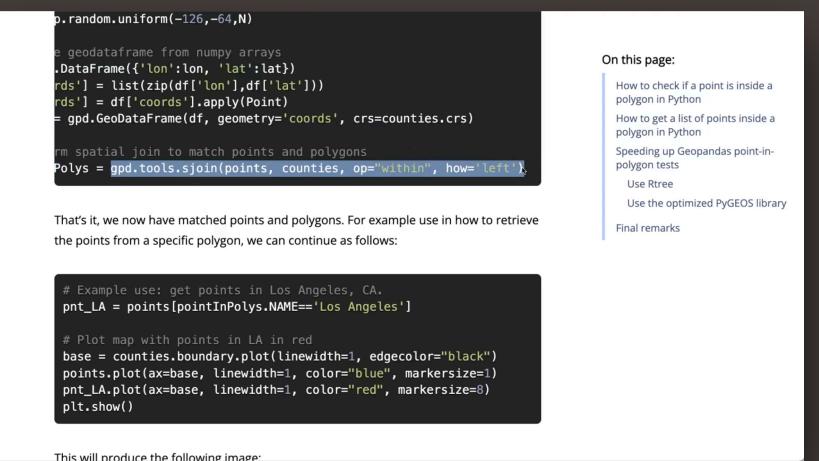
3.



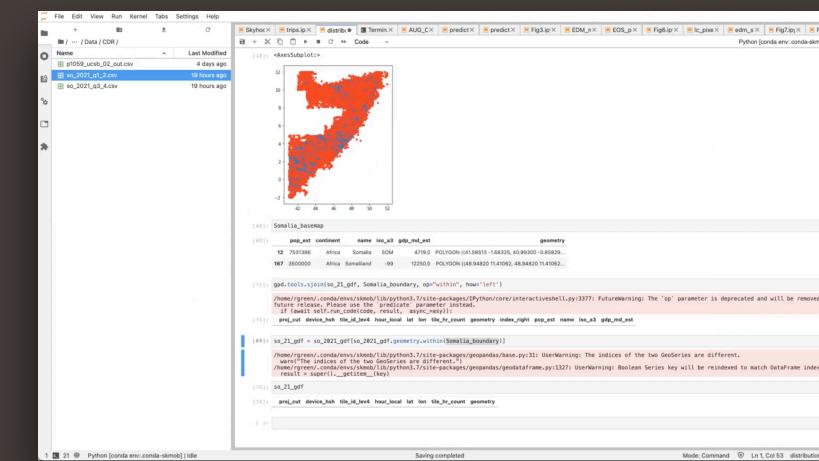
4.



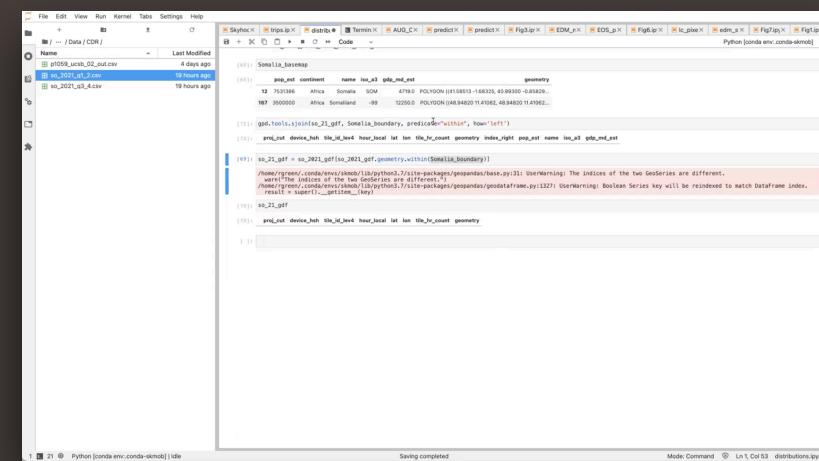
5.



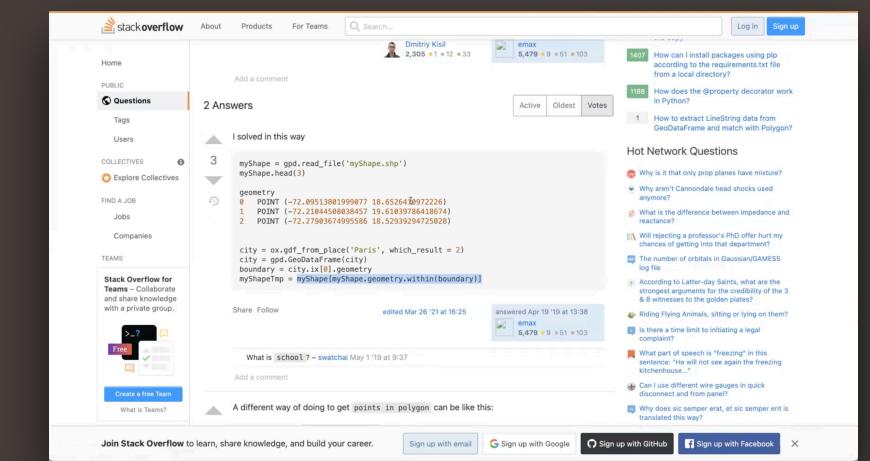
6.



7.



8.



The above sequence took **6 minutes**.

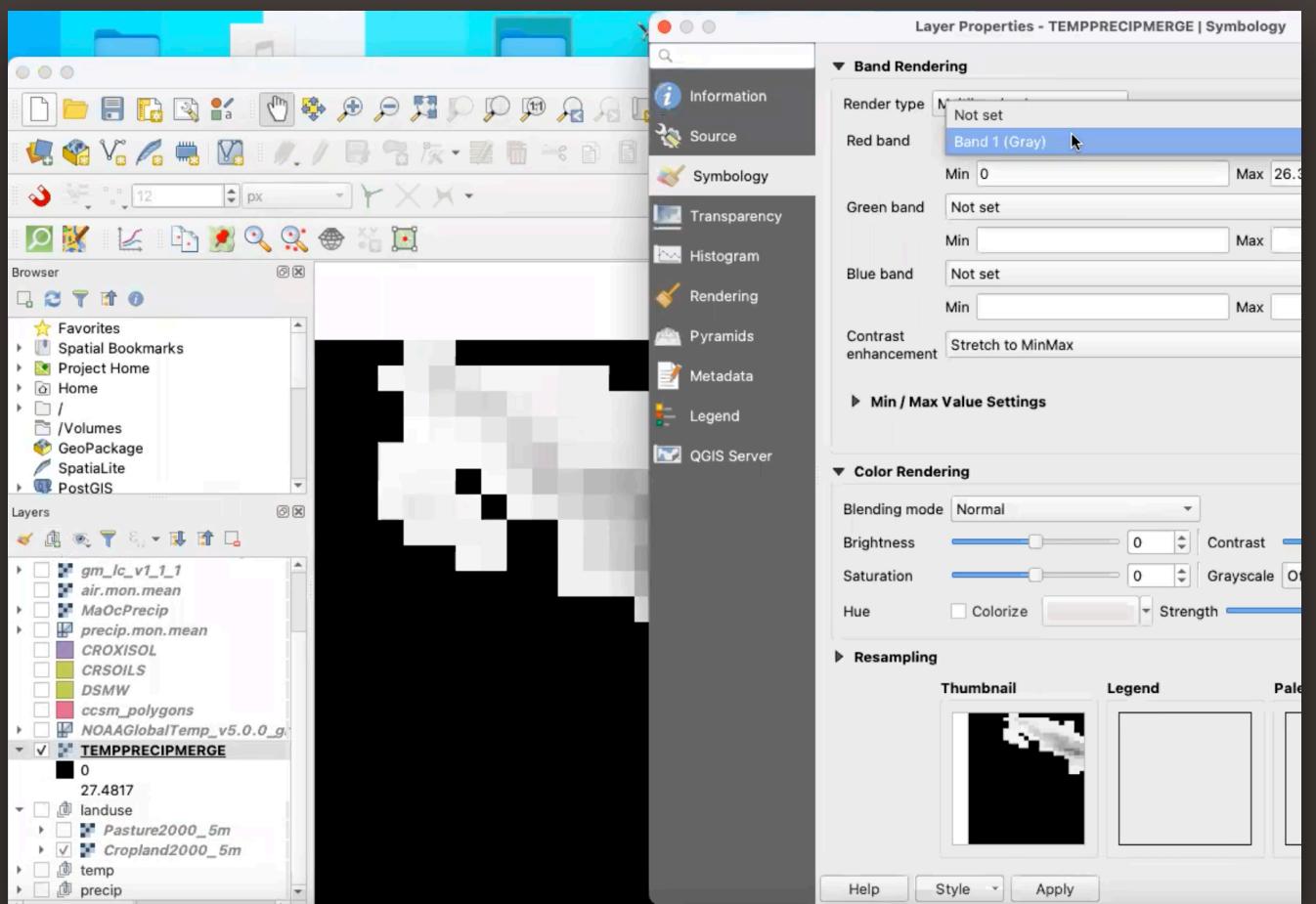
We continued for **another 12 minutes** before arriving at the desired output.

```
so_2021_gdf[so_2021_gdf.geometry.within(Somalia_boundary.loc[0, 'geometry'])]
```

# Reasoning about geospatial operator behavior

Participants understood an operator's behavior by running it and inspecting the output.

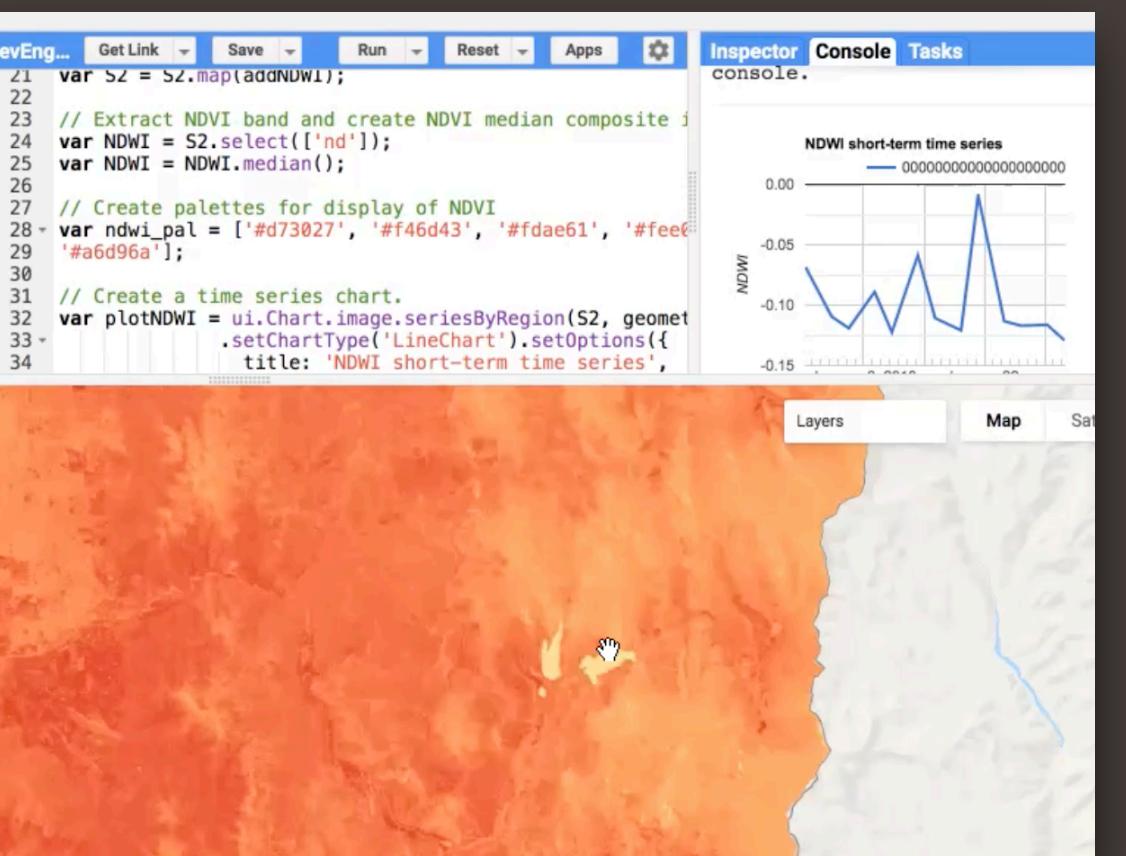
Participant 11 tried using `geopandas`' `.contains` API to see if it would work on point-within-closed-line operations.



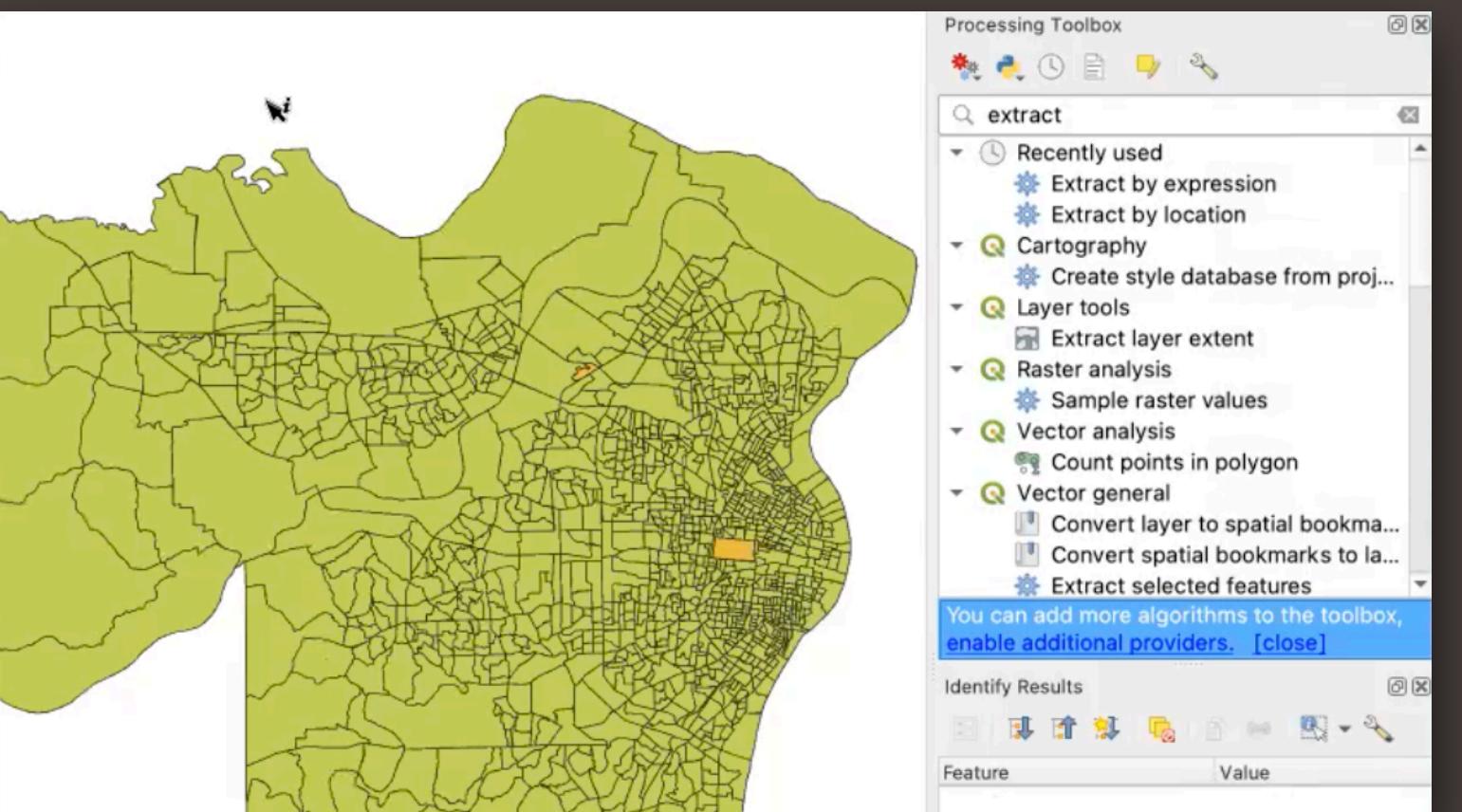
Participant 3 executed a **Merge** on their temperature and precipitation rasters and learned that the operator, by default, combines all input rasters into a single band rather than preserving multiple bands.

# Reasoning about geospatial operator behavior

Participants use specific features or pixels to **validate operator semantics**.



Participant 1 checked the pixel values of a lake in their normalized difference water index (NDWI) raster to validate the algorithm's results.



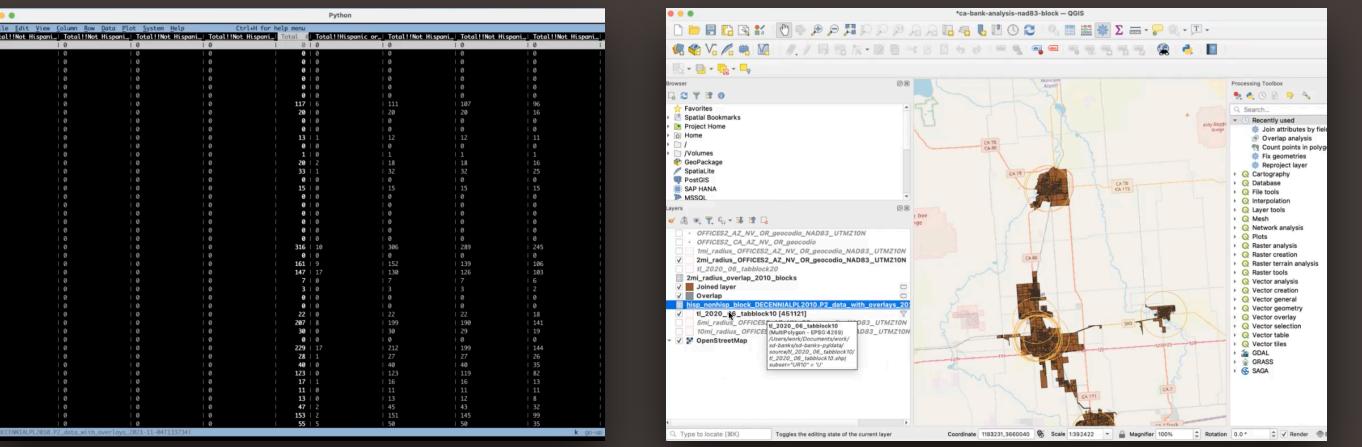
Participant 4 used color to validate that two features that should have been removed by **Extract by Expression** were indeed removed.

# Reasoning about geospatial operator behavior

“I can never remember the vector operations. There’s like Union and Merge. Combine! **I can never remember exactly what they do.** I know exactly what the output should look like in the end, **I’m just trying to figure out the tool that gets me that output.**”

— Participant 15

# Findings



Lack of spatial visibility in code-based tools



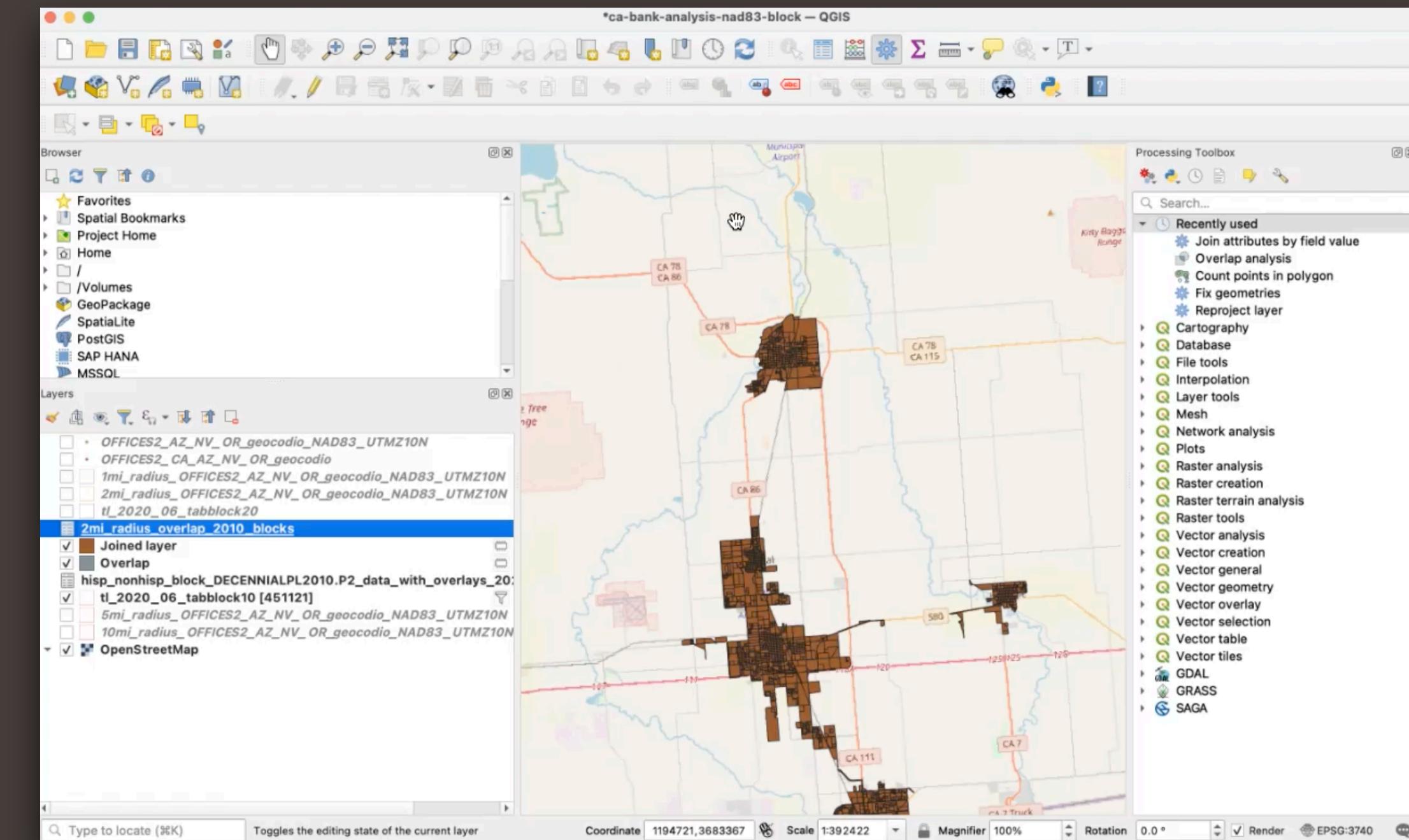
Programming Environments

Code-based tools represent geospatial data as tables, **but users rely heavily on visual spatial reasoning during data exploration.**

# Lack of spatial visibility in code-based tools

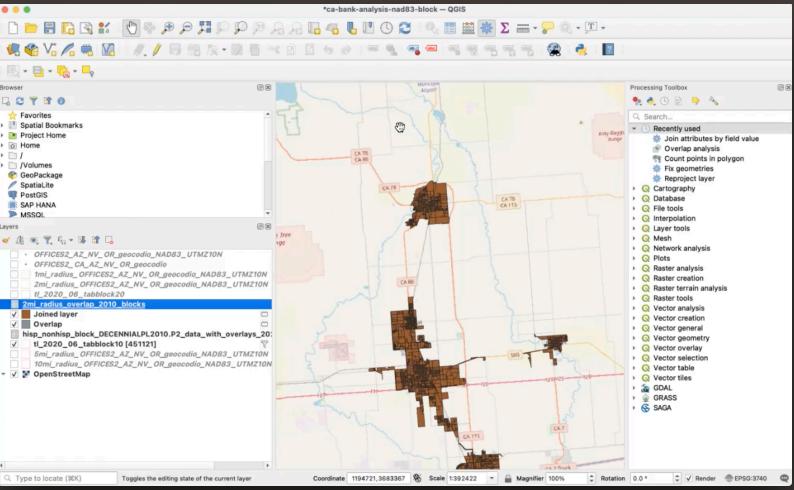
1.

Load TIGER/Line  
shapefile of urban  
Census block groups in  
Imperial County into  
QGIS.



# Lack of spatial visibility in code-based tools

1.

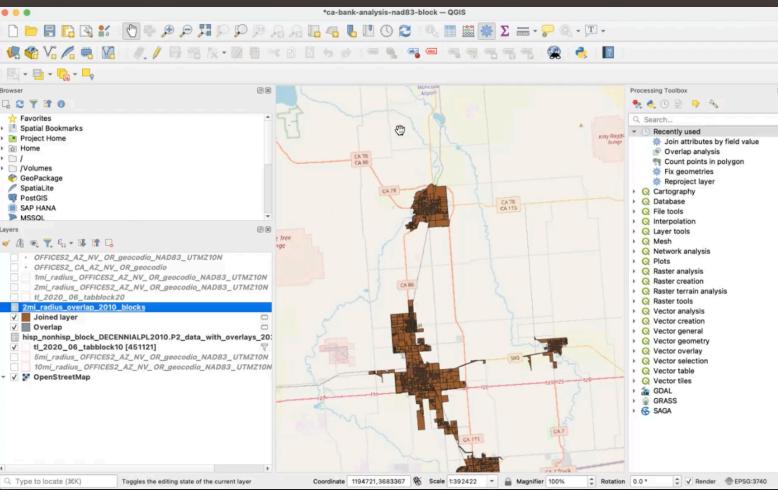


2.

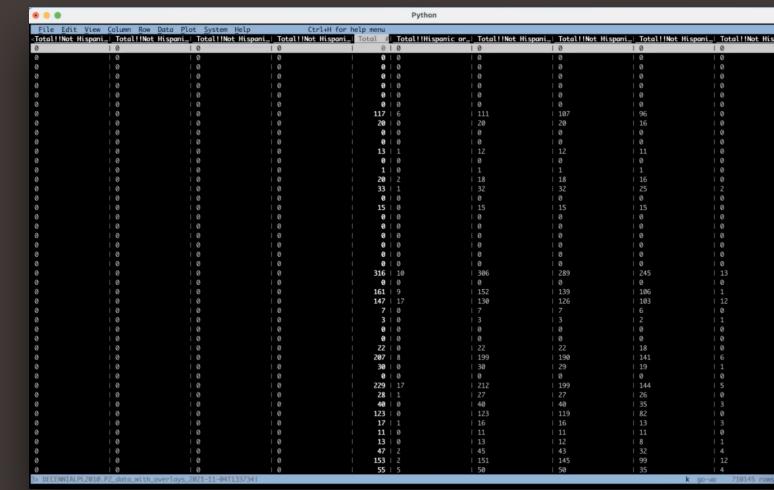
Move to VisiData, a CLI  
tool for data wrangling,  
to Join 2010 Census  
data to block groups.

# Lack of spatial visibility in code-based tools

1.

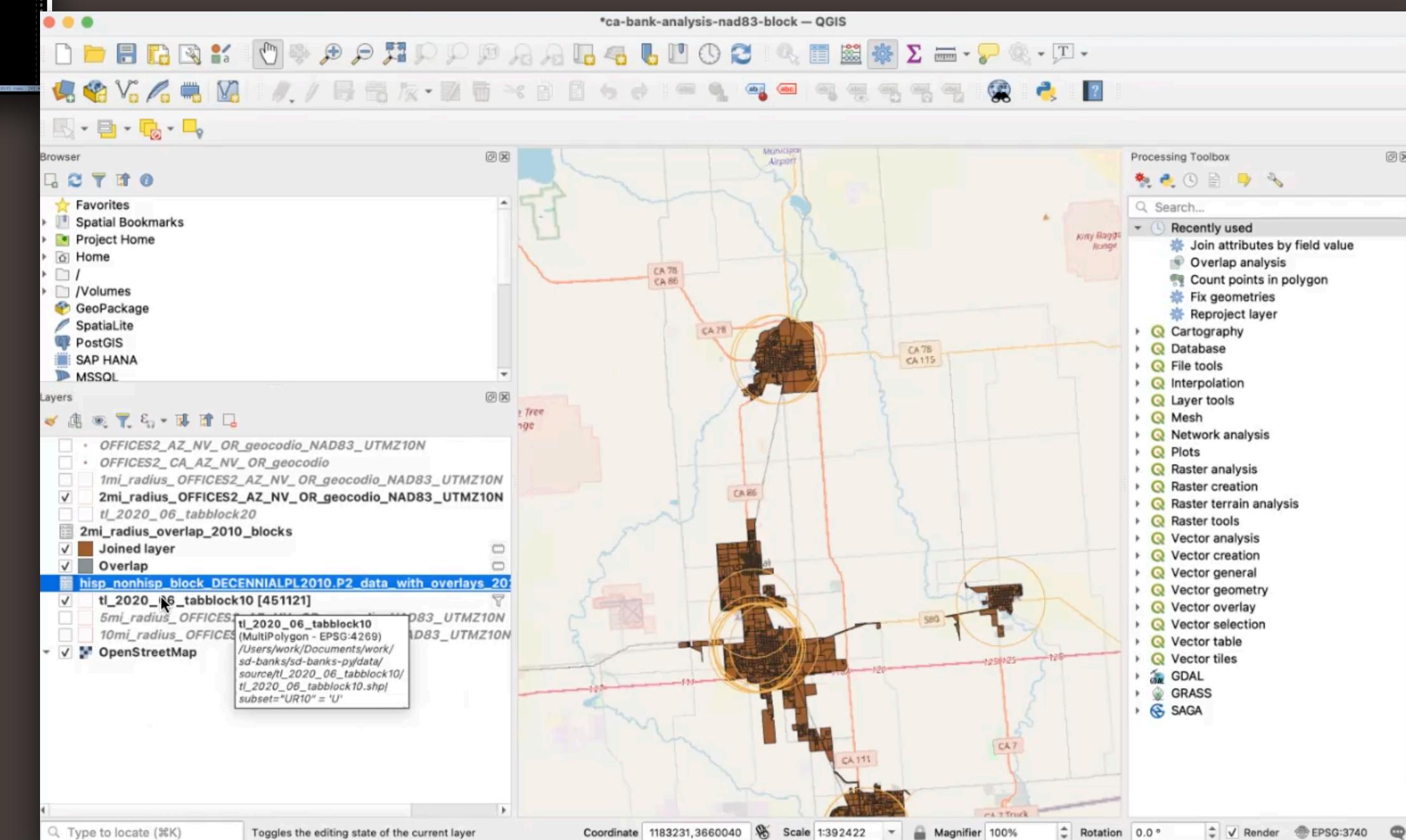


2.



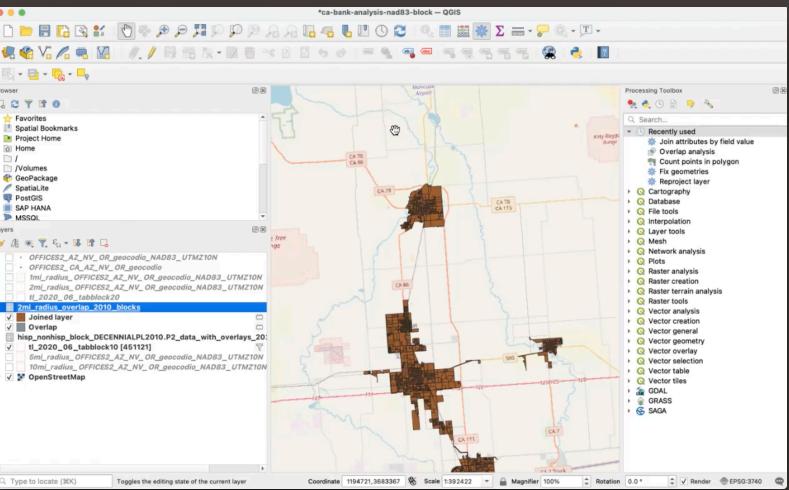
3.

Move back to QGIS and generate 2-mile buffers around the point locations of banks.



# Lack of spatial visibility in code-based tools

1.



2.

A screenshot of a Python console window. It displays a large table of data with many columns and rows. The data appears to be a grid of numerical values.

3.

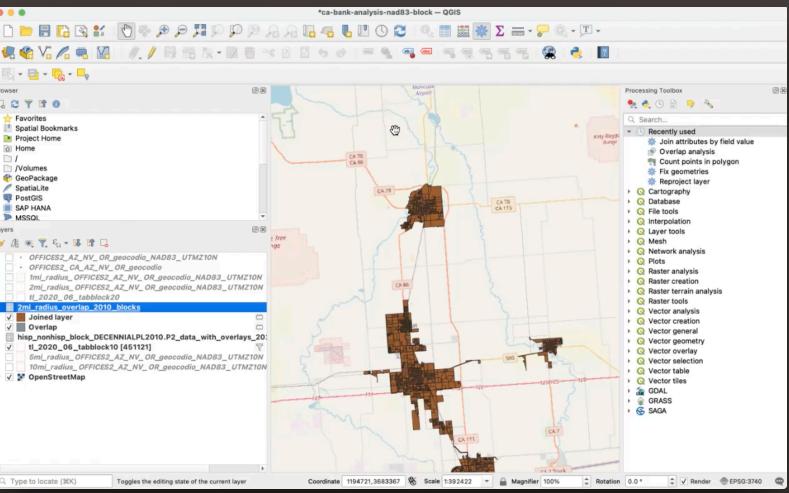
A screenshot of the Processing Toolbox in QGIS. The 'Overlap Analysis' algorithm is selected. The parameters dialog shows the 'Input layer' set to 'tl\_2020\_06\_tabblock10 [EPSG:4269]' and the 'Overlay layers' section set to 'Selected features only'. The 'Log' tab is visible at the top right of the dialog.

4.

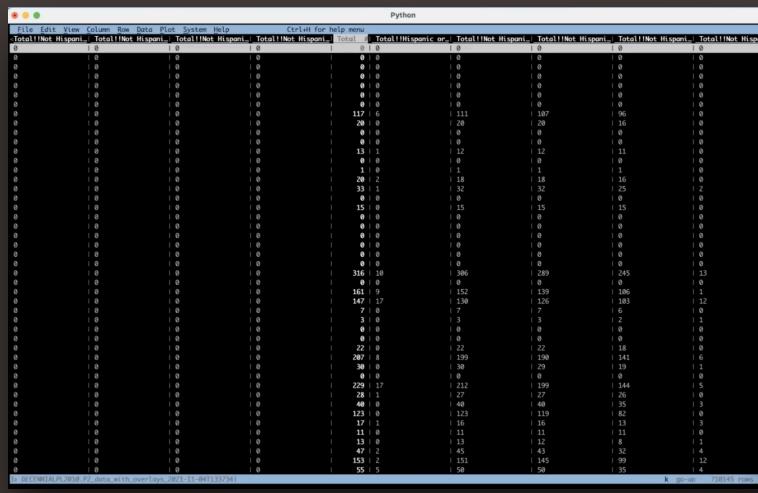
Use the **Overlap analysis** operator to compute the area overlap between each block group and any buffer.

# Lack of spatial visibility in code-based tools

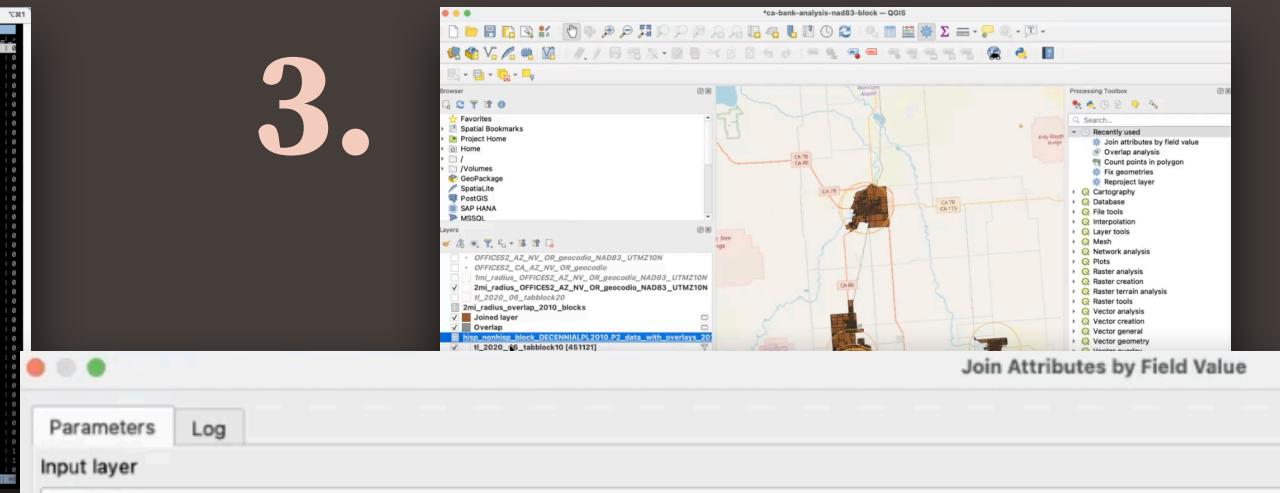
1.



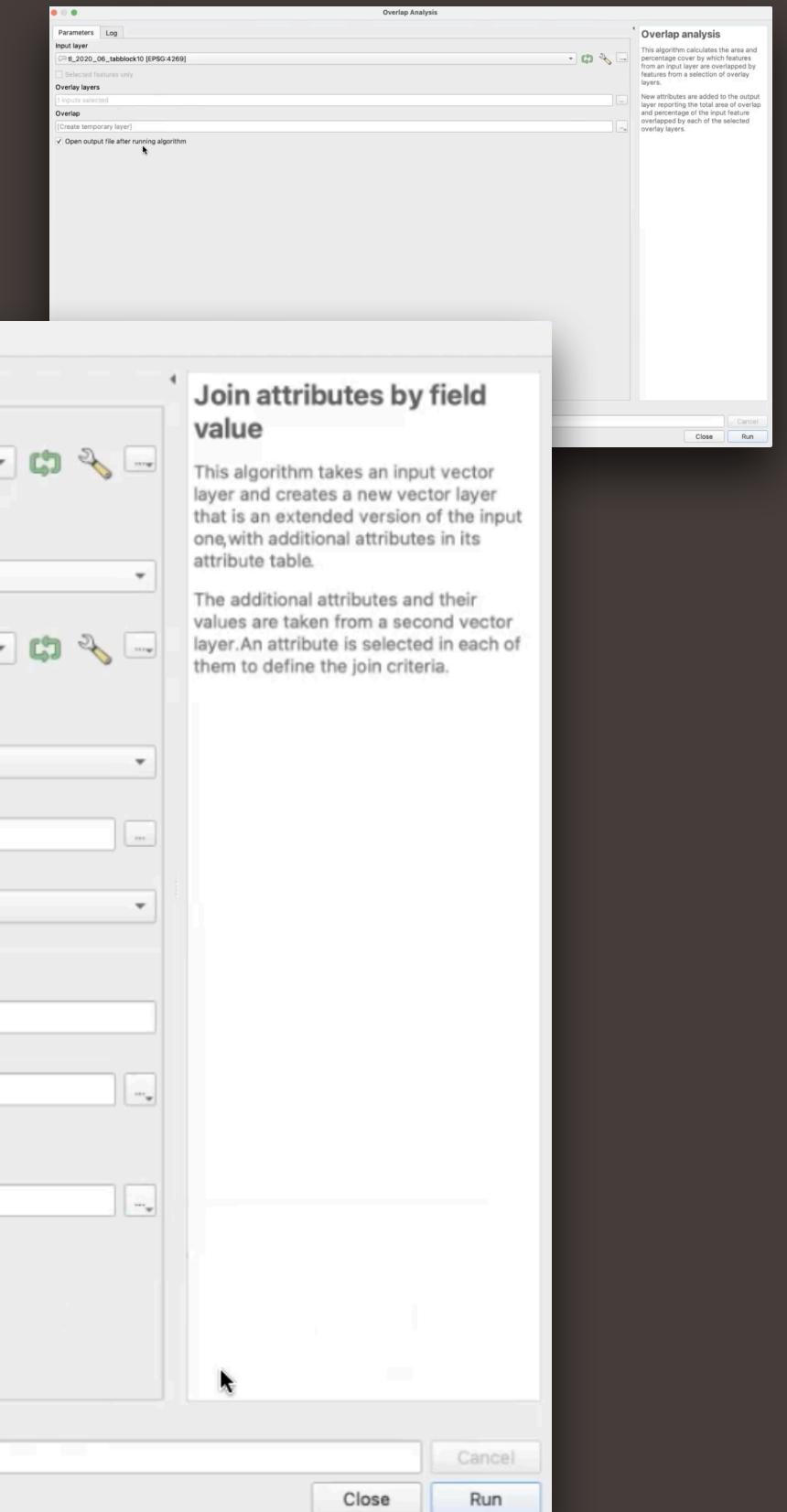
2.



3.



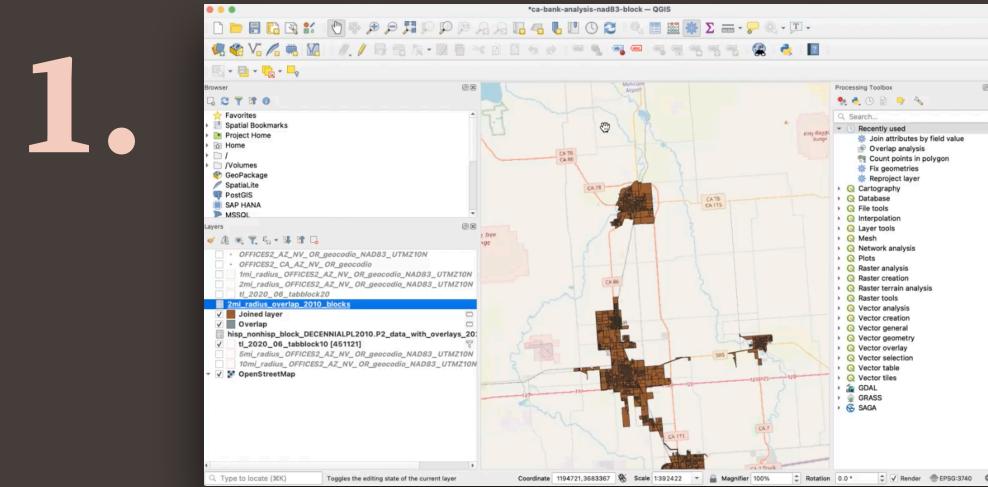
4.



# 5.

Use the Join Attributes By Field Value operator to join the areal overlay back to the 2010 Census data.

# Lack of spatial visibility in code-based tools



2

3

The screenshot shows a QGIS interface with a map of a residential area. The map includes several layers: 'OFFICES2\_AZ\_NV\_OR\_geocodio\_NAD83\_UTMZ10N' (highlighted in red), '2mi\_radius\_OFFICES2\_AZ\_NV\_OR\_geocodio\_NAD83\_UTMZ10N\_2020\_06\_tbblock20' (highlighted in blue), '2mi\_radius\_overlap\_2010\_blocks' (highlighted in green), and 'Overlap'. A legend on the right indicates 'Any address overlap'. The Python console at the bottom shows the command: `<1mi_radius_OFFICES2_AZ_NV_OR_geocodio_NAD83_UTMZ10N_pc> geoid`.

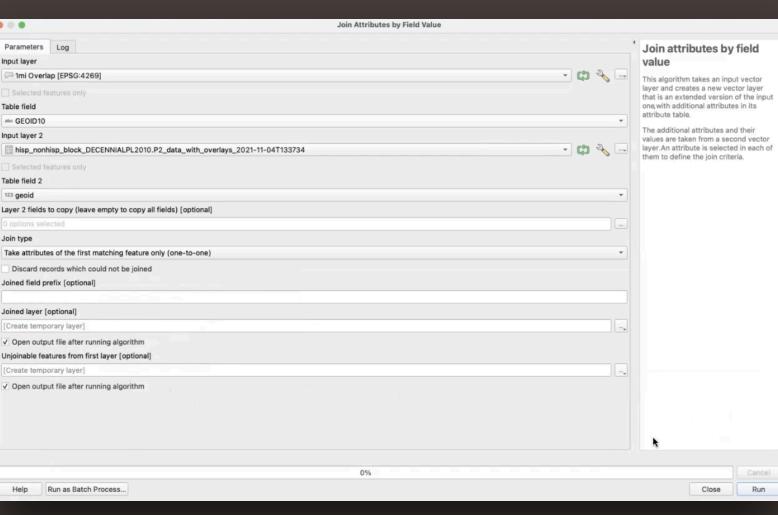
4

This algorithm calculates the area and percentage cover by which features from one or more layers are covered by features from a selection of overlay layers.

New attributes are added to the output layer reporting the total area of overlap and percentage of the input feature overlapped by each of the selected overlay layers.

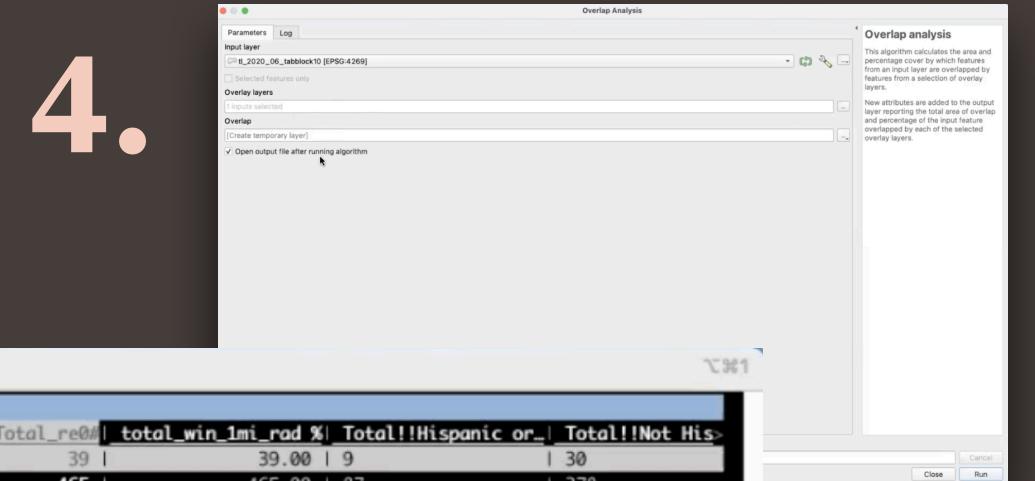
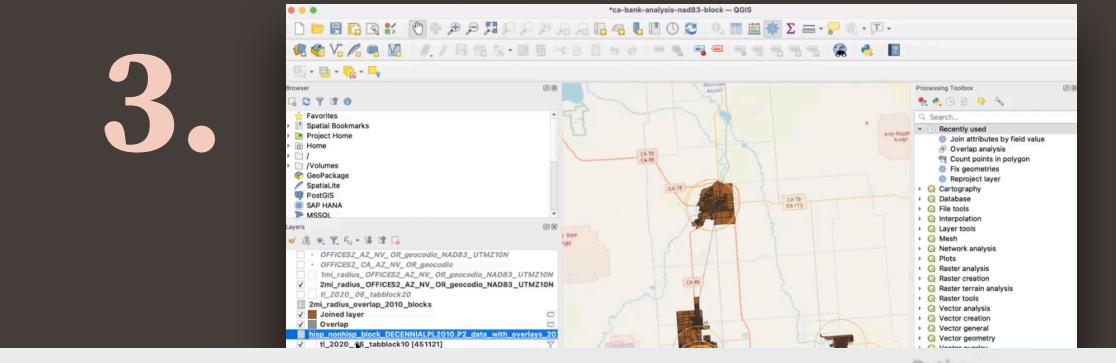
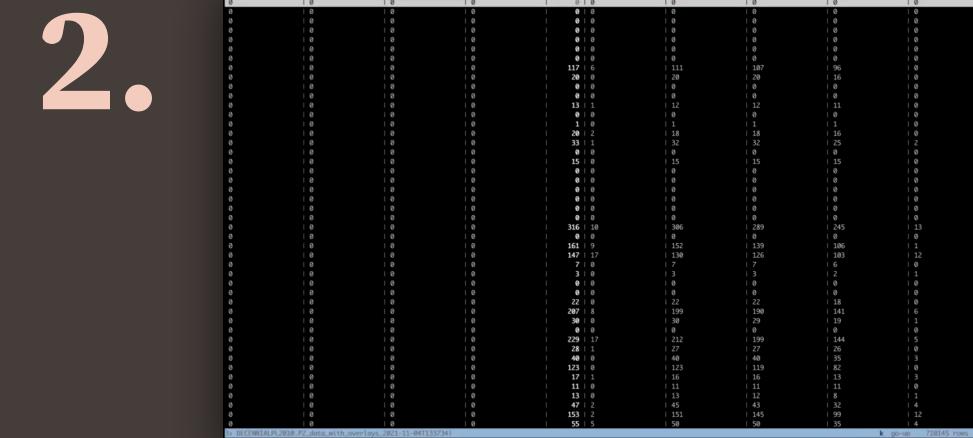
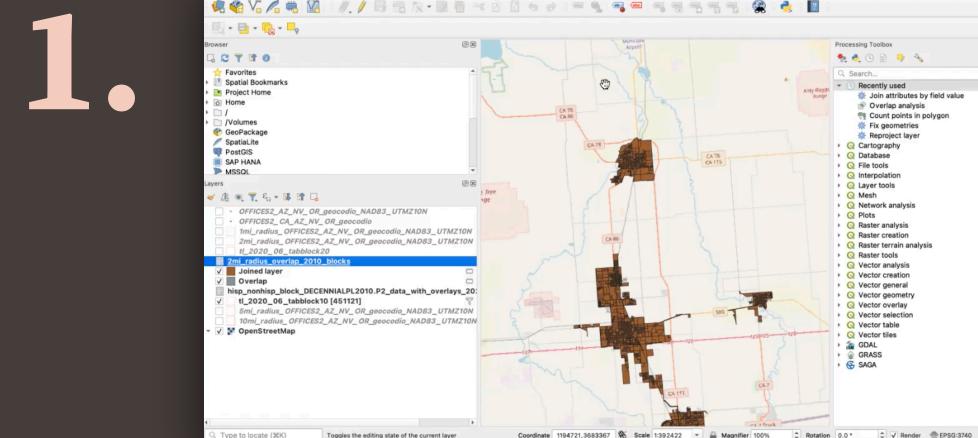
# 6.

Move back to VisiData  
to multiply % overlay  
with bank buffers by  
total population for each  
block group.



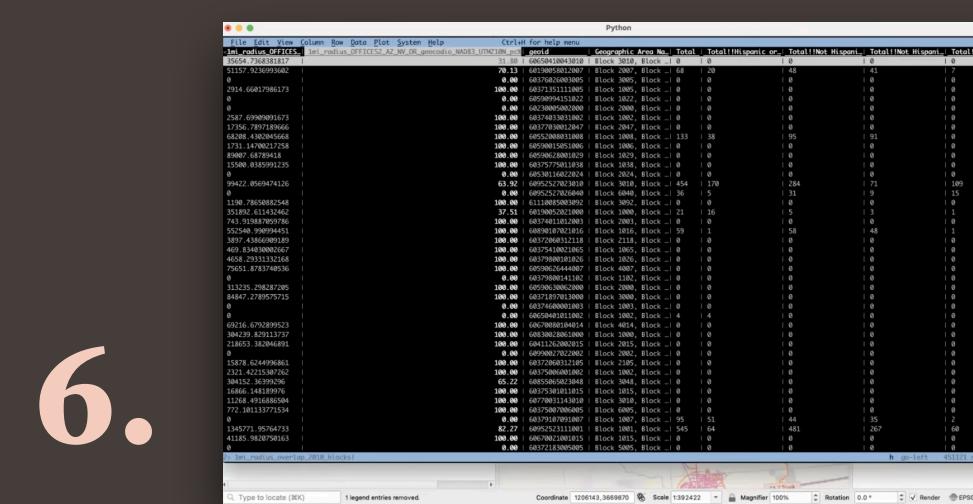
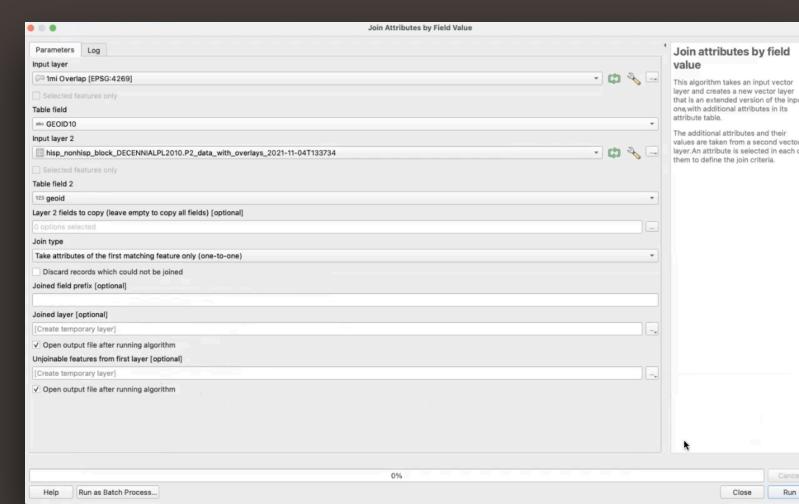
5.

# Lack of spatial visibility in code-based tools



7.

Use the macOS  
Calculator to compute  
the percentage of  
Imperial County  
residents living in a  
banking desert.



5.

1.

6.

2.

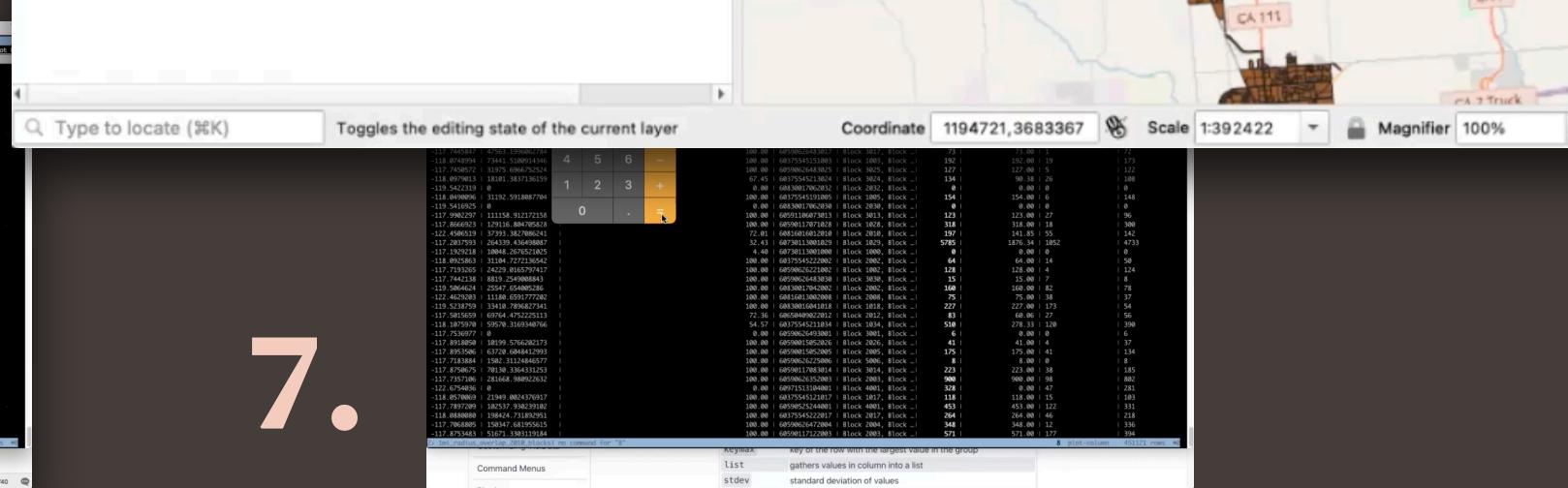
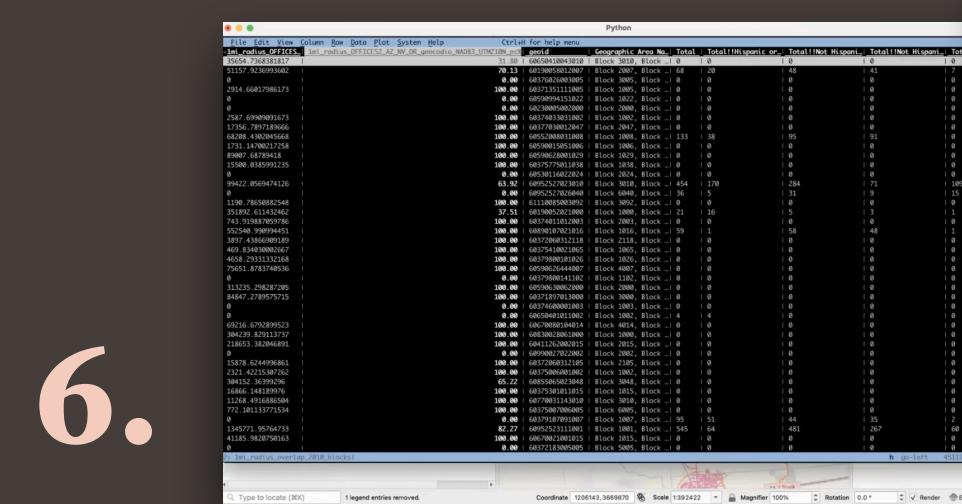
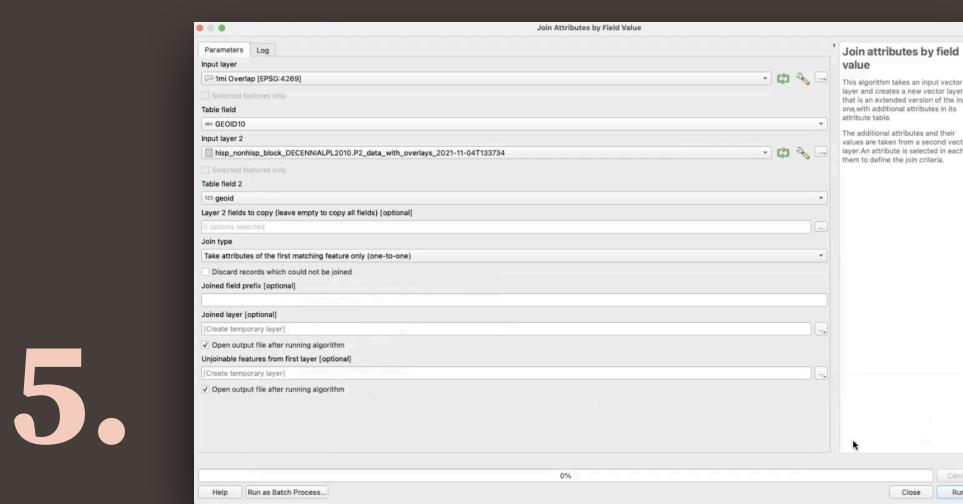
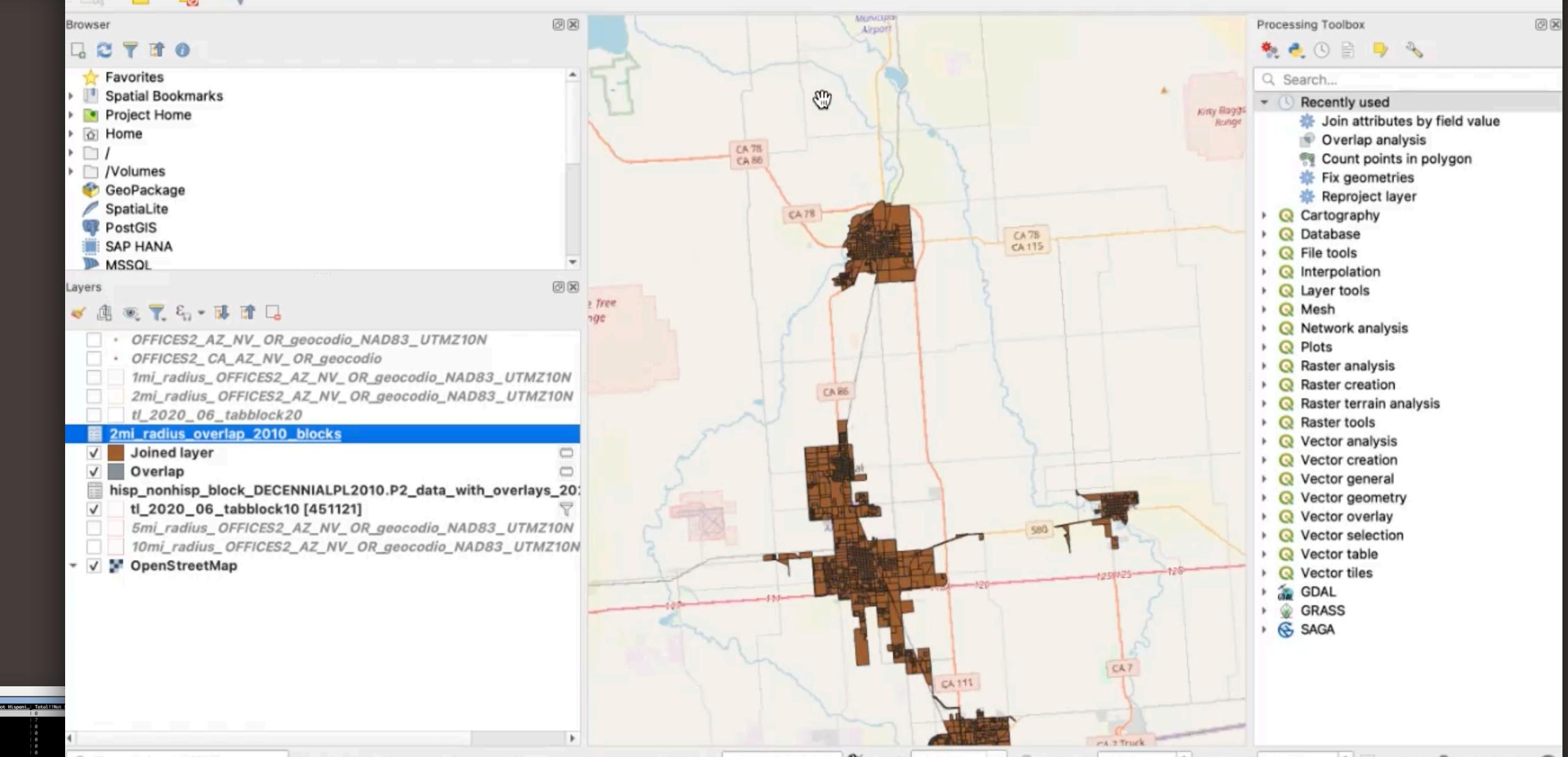
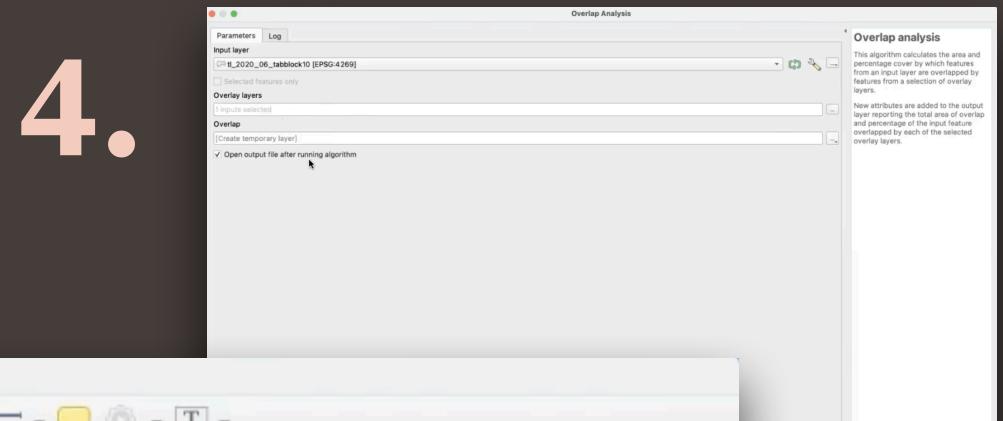
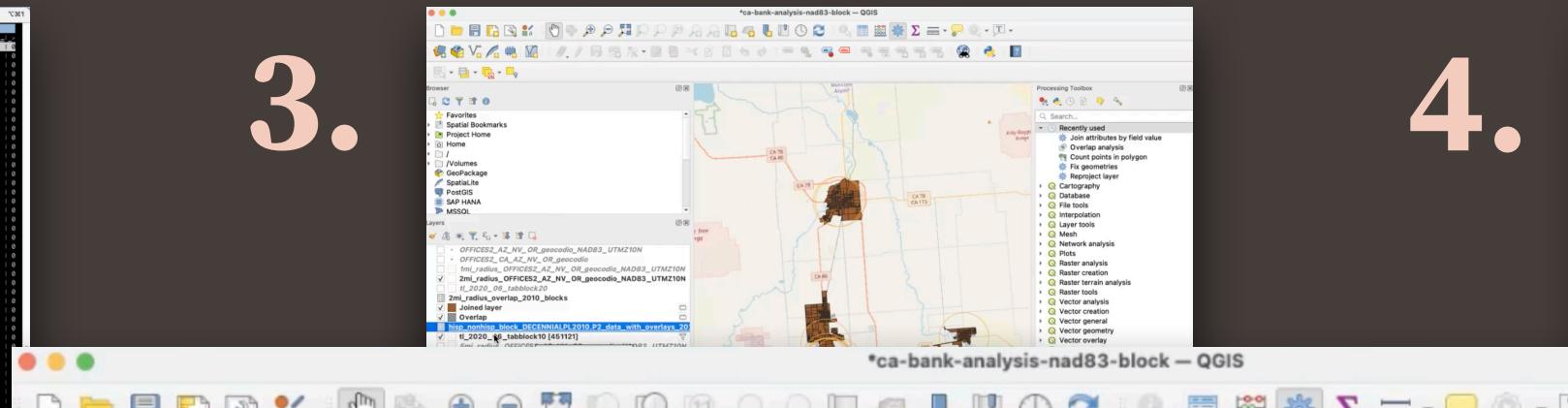
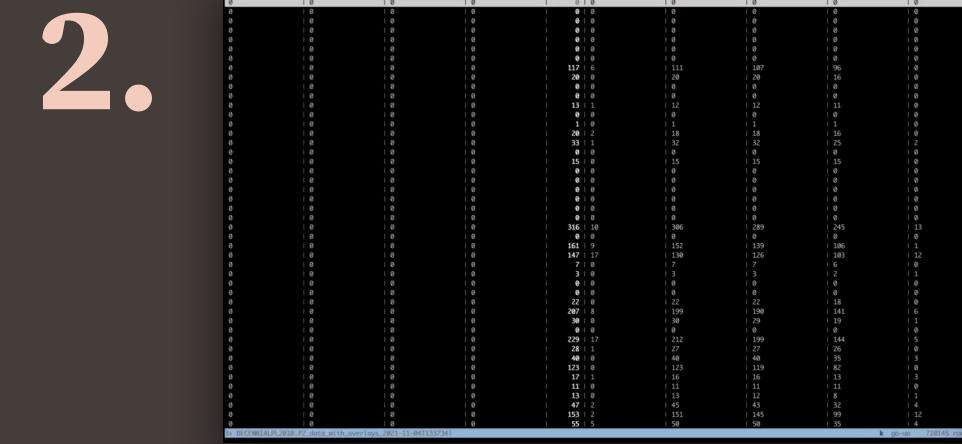
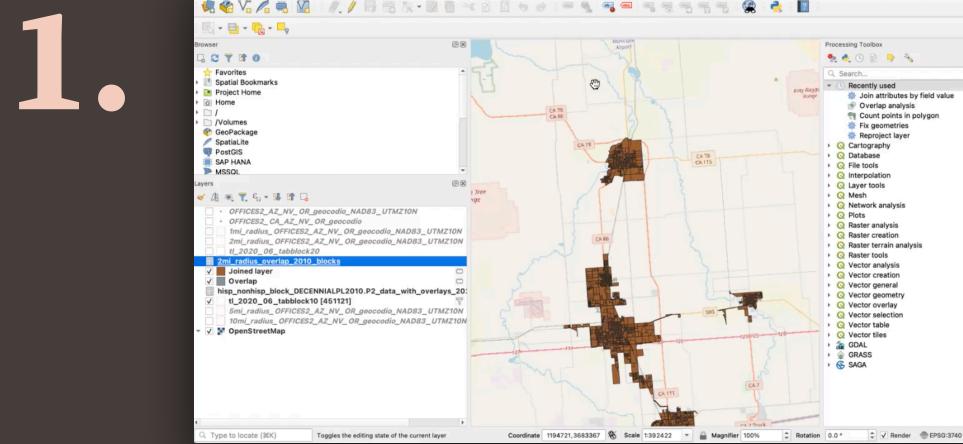
3.

4.

7.

8.

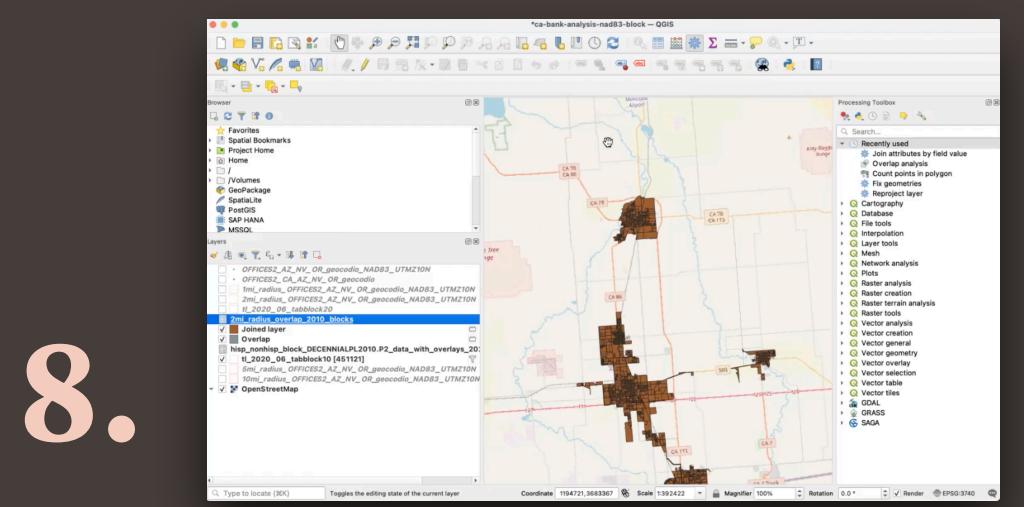
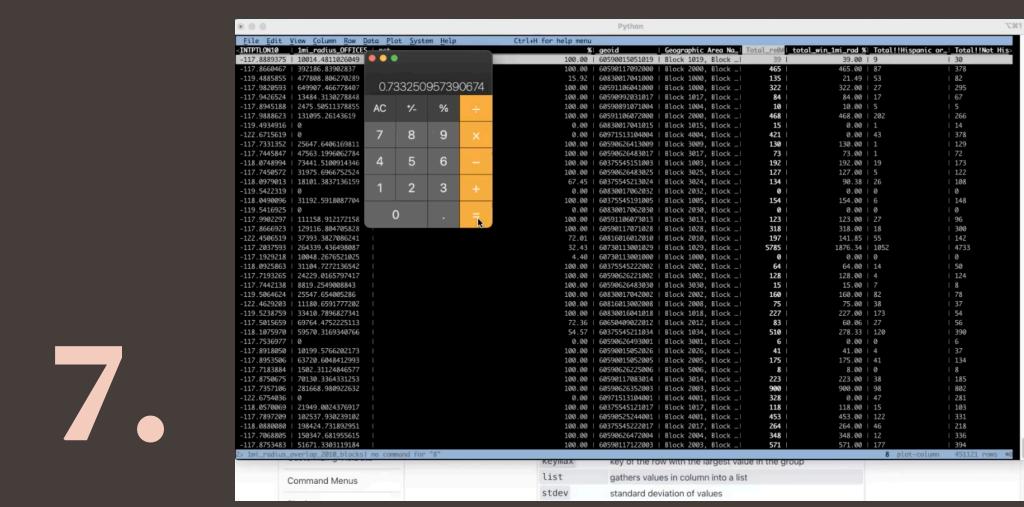
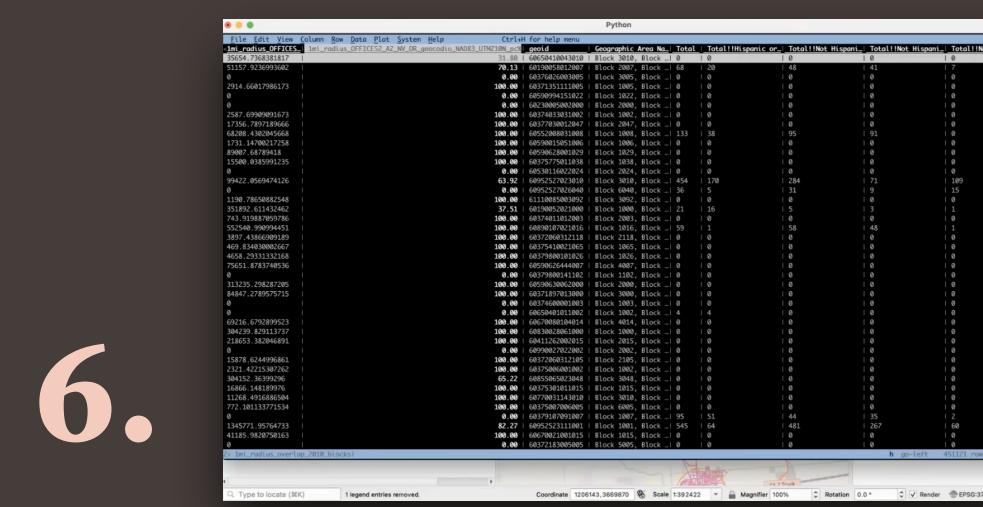
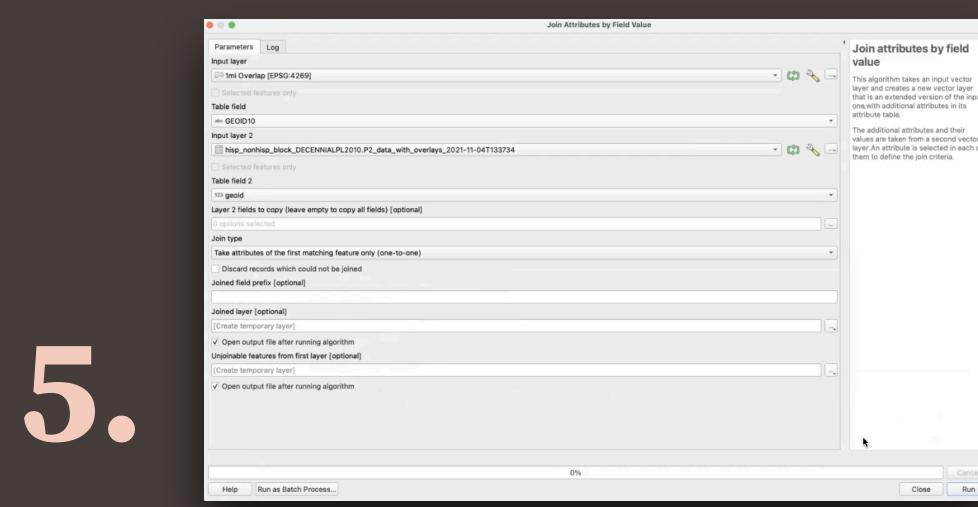
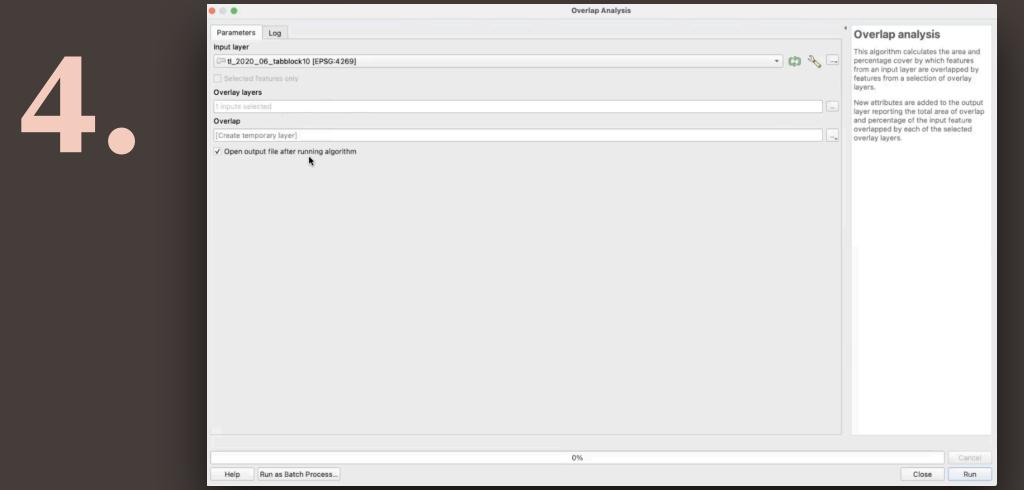
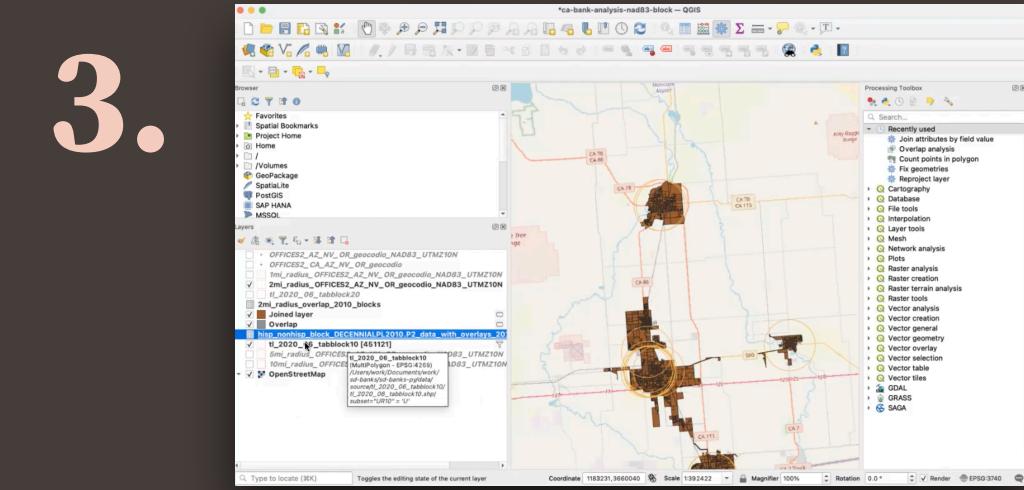
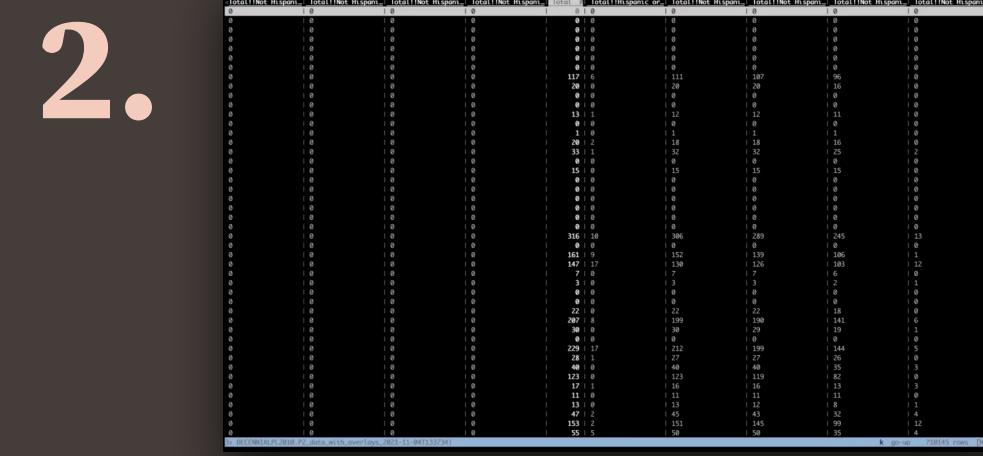
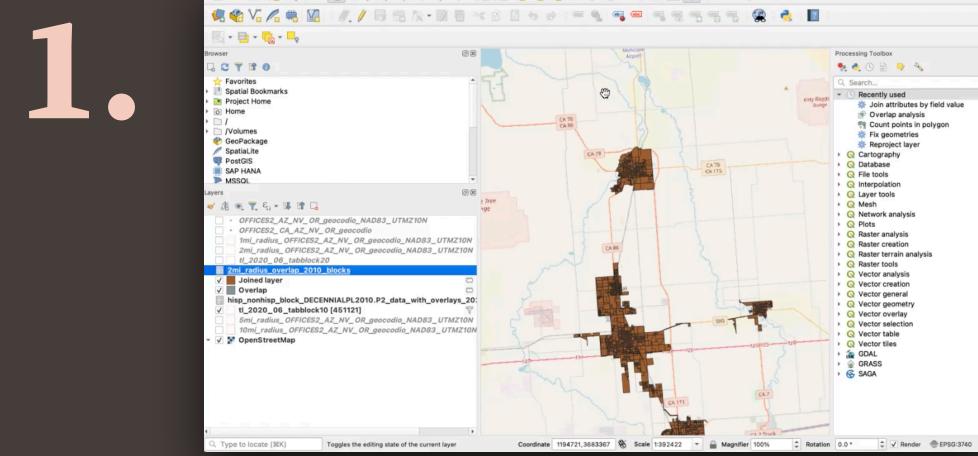
# Lack of spatial visibility in code-based tools



8.

Move back to QGIS to repeat the analysis using 1-mile buffers.

# Lack of spatial visibility in code-based tools



Participant 9 oscillates between a programming environment and GIS software, **moving to QGIS for all geometric operations**.

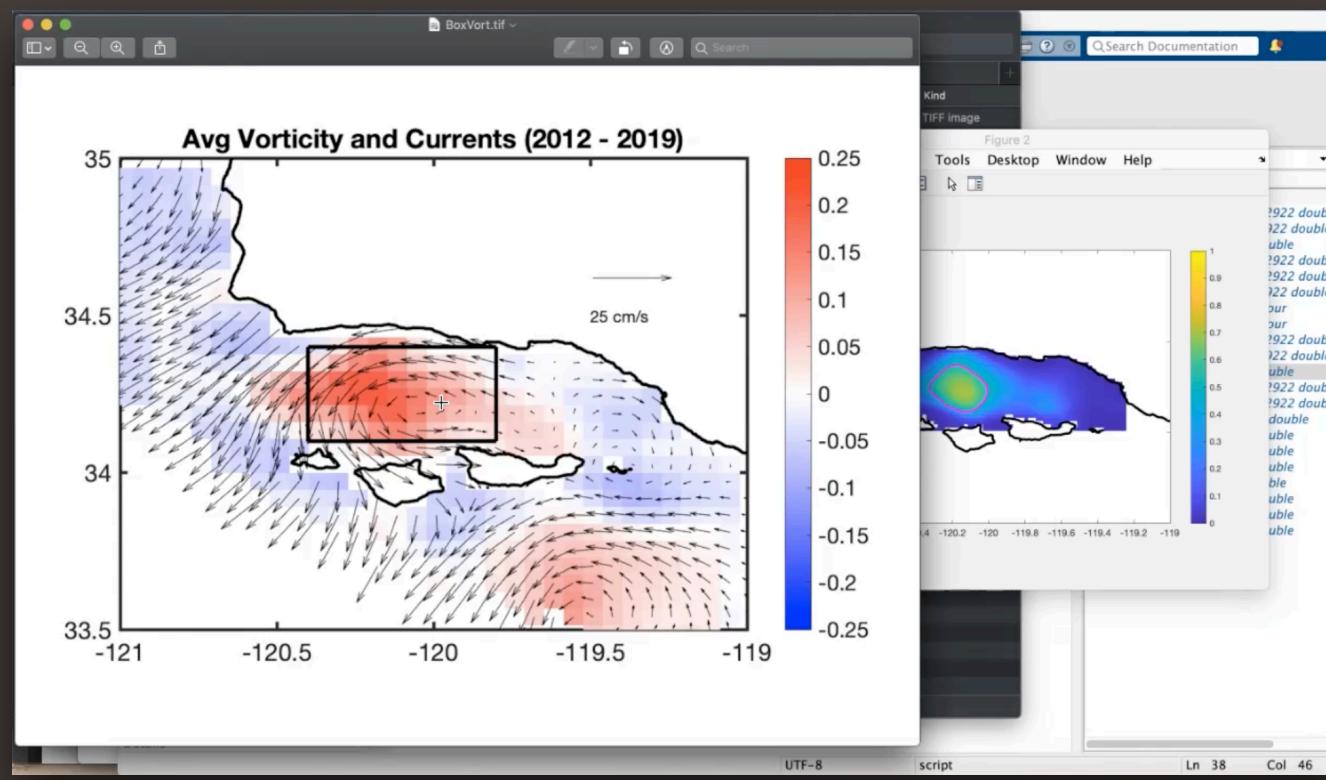
# Lack of spatial visibility in code-based tools

“I'm working in QGIS. I know that it's slower than it would be to do it in PostGIS or maybe even `geopandas` and so I've considered switching to that. But I'm still sort of new enough that **I need to kind of ‘see’** to make sure my projections are right and stuff like that.”

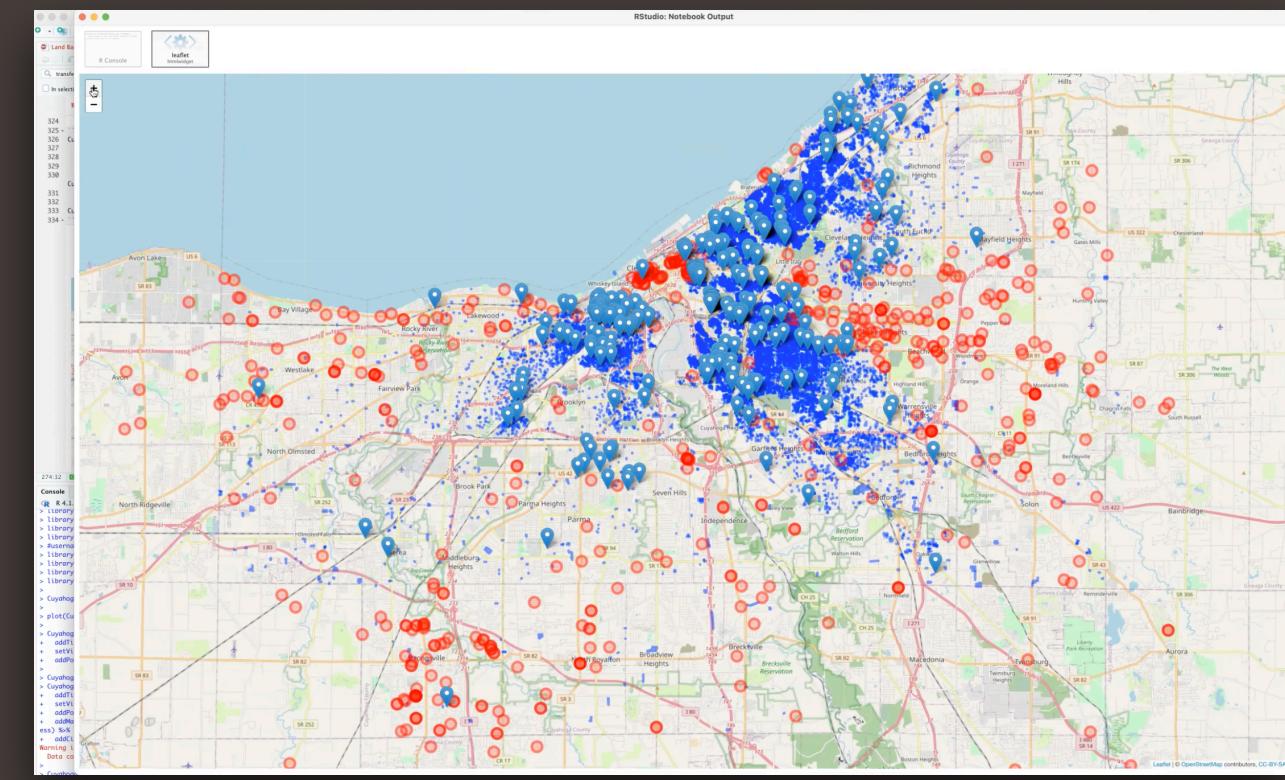
— Participant 9

# Lack of spatial visibility in code-based tools

Participants rely heavily on **visual overlays of layers** to investigate spatial correlations, and recreate this functionality in programming environments.



Participant 16 compared an image of a plot against his Matlab figure to “eyeball” the spatial correlation between two variables of interest.



Participant 6 wrote an 8-line function in R to render three layers on a Leaflet map, allowing her to inspect the spatial overlap between them.

# Lack of spatial visibility in code-based tools

“Now I *could* try to visualize it here with `matplotlib` and `geopandas`, but I know those things are shitty, shitty, shitty and **not interactive** and so I'm like, '**I gotta take this to QGIS.**'”

— Participant 18

# Roadmap

## 5. Findings and Discussion



# Roadmap

## Highlights

1. The importance of **informal program representations** to the working process of GIS software users
2. The challenge of **reasoning about the behavior of geospatial operators**
3. The **lack of spatial visibility in code-based tools**

