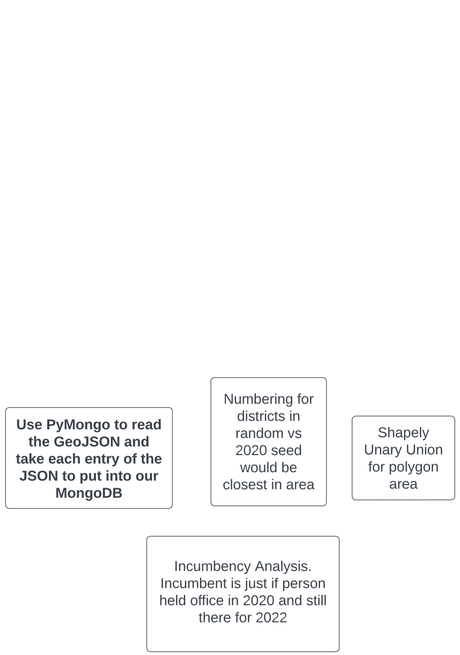
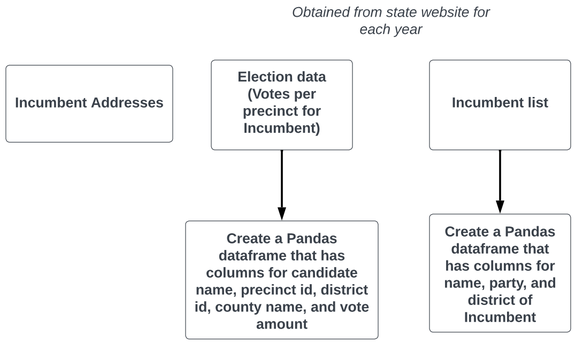
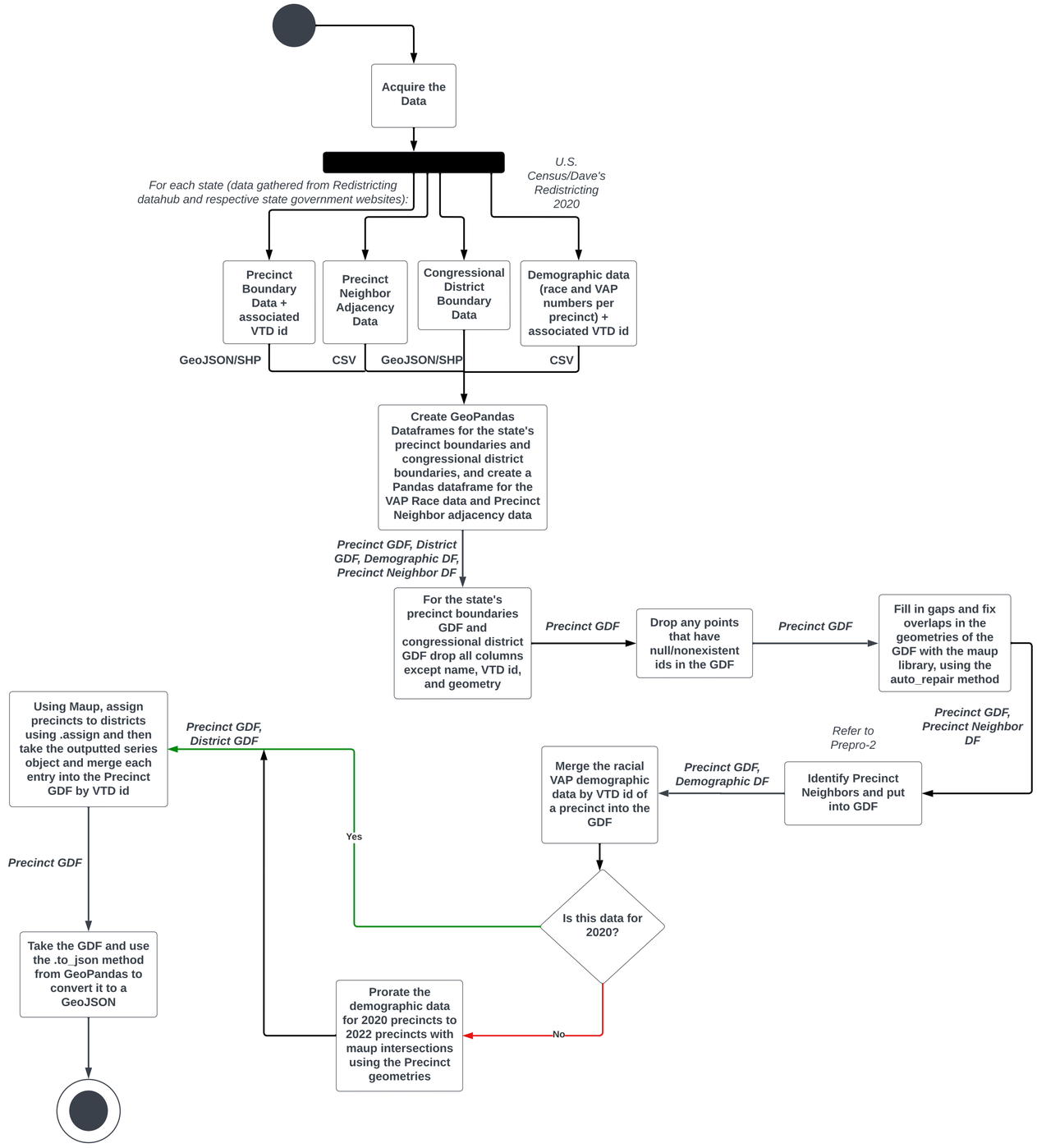
Prepro-1: Integrate multiple data sources







1. Label Data flows. 2. Figure out Incumbent/Election data 3. Maybe just do preprocessing as a whole graph instead of breaking it up as cases

Prepro-2: Identify precinct neighbors
Built by: Pirates

Precinct GDF from Prepro-1

Add a column labeled "NEIGHBORS" to the GDF and initialize the column with an empty string

Do we have a dataframe that contains the precinct neighbor information already?

Yes

No

Convert the GDF's Coordinate Reference System to use meters instead of degrees (EPSG:3857) gdf.to_crs(3857)

Precinct GDF

Keep a copy of the GDF and buffer this copy in its geometries by 200 ft, or in this case 60.96 meters

Precinct GDF, Buffered GDF

Overlay the original GDF over the buffered GDF and create a new GDF that stores the intersections as polygons

Intersections GDF

Create a zip object of tuples to store immediately intersecting precincts as a pair of 2 tuples from the Interesections GDF

Tuple

Create an empty dictionary to map a precinct id to a list of its neighbors

Tuple, Dictionary

Iterate over the the zip object (Tuple)

Dictionary

Tuple

Are the 2 ids in the pair of tuples the same?

Yes

No

Tuple

Is the id already in the dictionary?

Tuple

Yes

Add the 2nd id to the dictionary at the 1st id's position

Tuple

No

Create an entry in the dictionary with the 1st id as the key and the 2nd as the value

Tuple

Iterate over the dictionary to add empty lists to self intersecting precincts

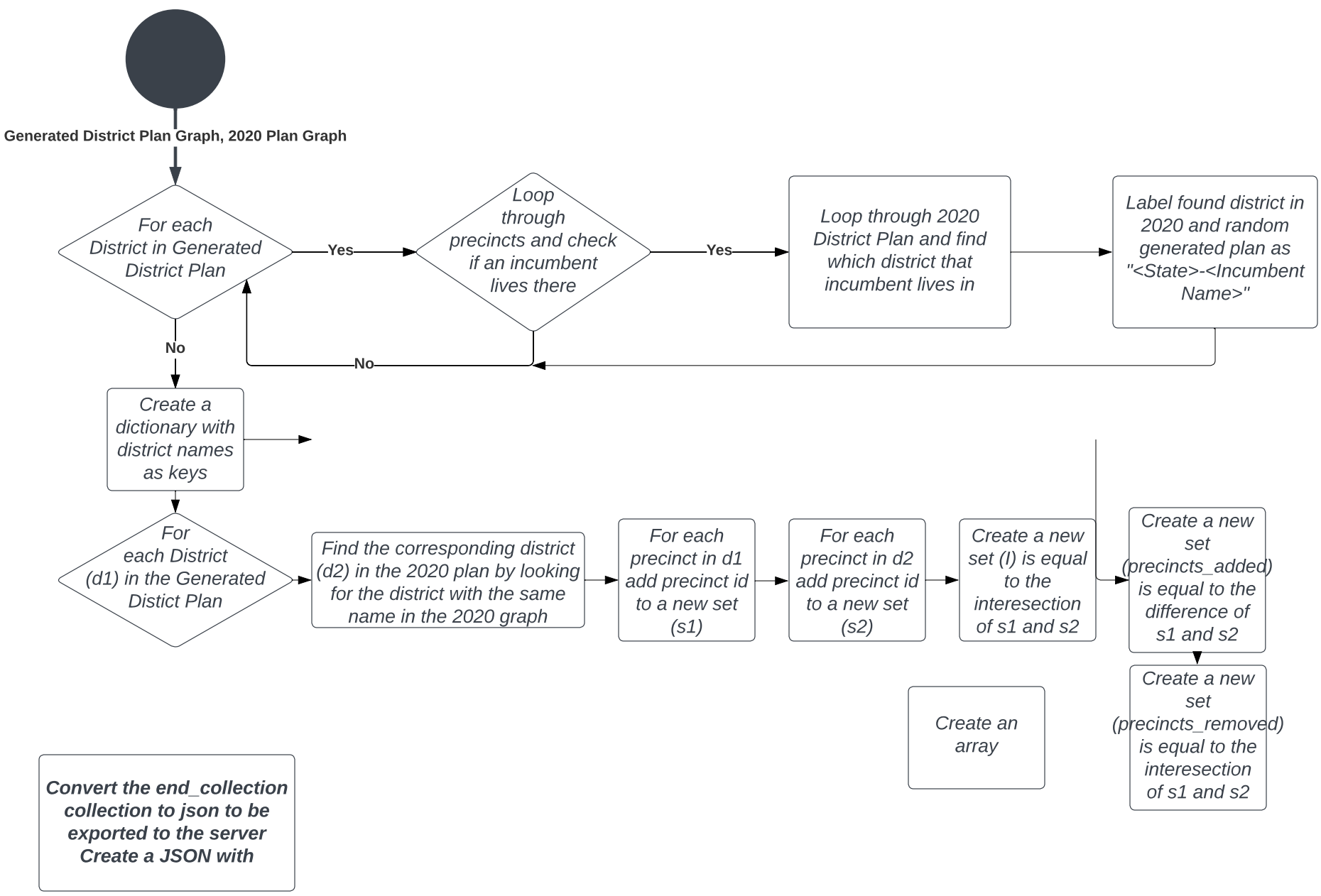
Dictionary, Precinct GDF

Populate the Precinct GDF "NEIGHBORS" column by the precinct id to have a string of list items from the dictionary for that specific precinct

Precinct GDF

Refer back to Prepro-1

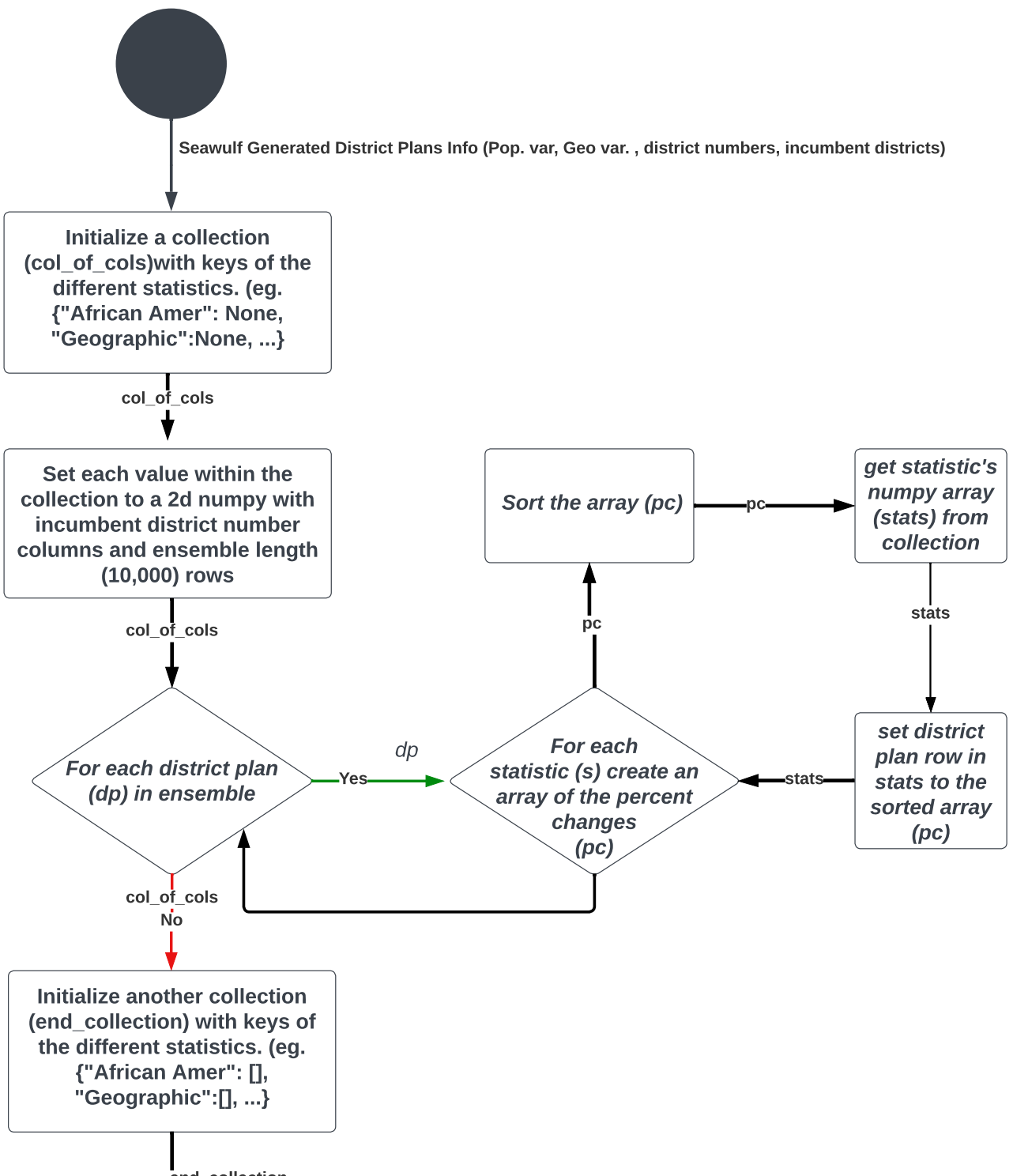
Seawulf-4: Calculate district correspondence between random plan and 2020 plan
Built by: Pirates

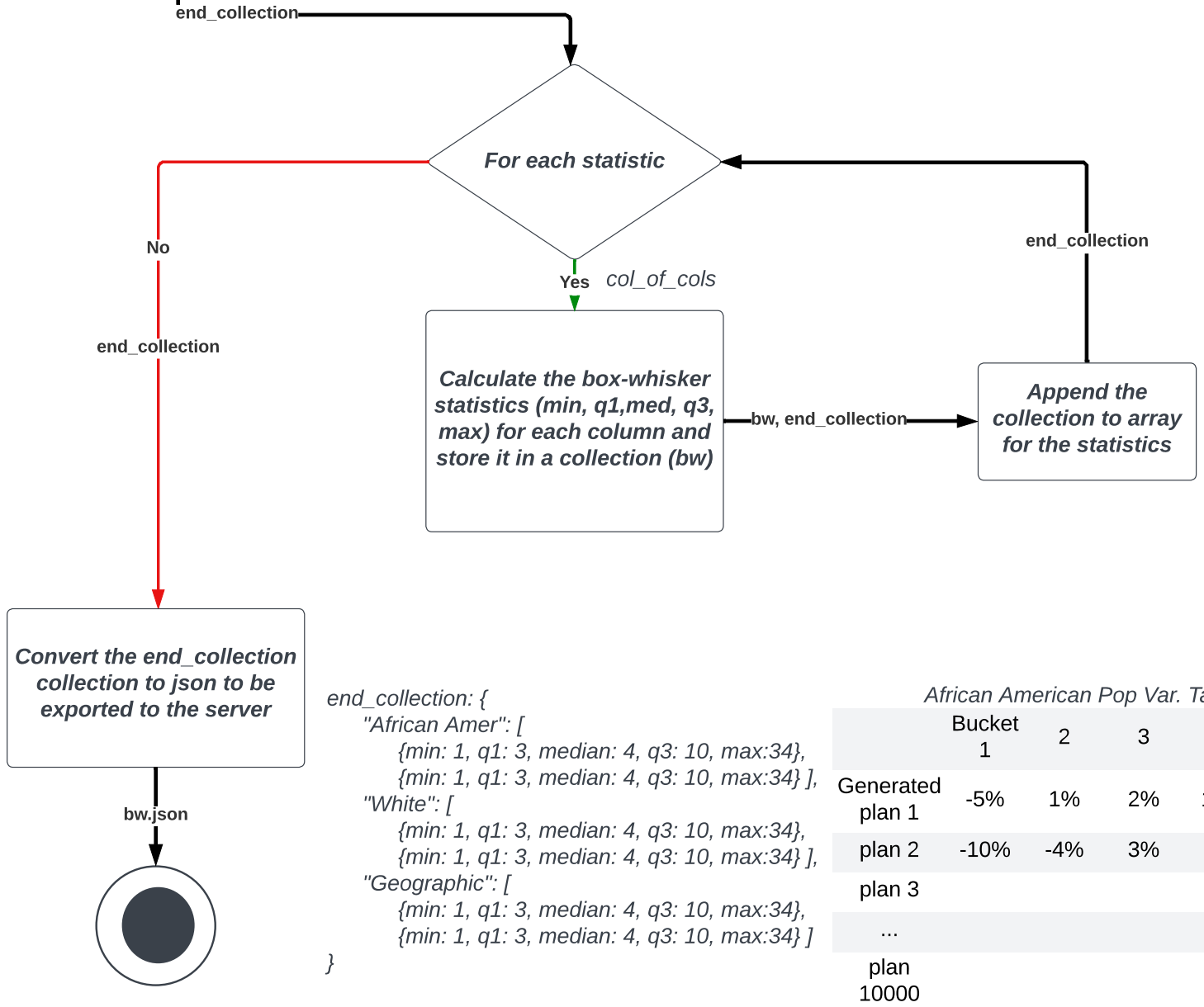


For each Congressional district boundary being considered (e.g., 2022 enacted plan and SeaWulf generated random plans) calculate the geometric area of 1) regions added to the district, 2) regions removed from the district, and 3) region common to both the 2020 district plan and the random or enacted plan being analyzed.

Read in generated plan graph and 2020 plan graph, create a collections with district names as the keys. Go through the generated graph. For each district create two arrays look at each precinct, if precinct is in 2020 plan, put in 1st array, if not in 2nd array. Now look 2020 plan, if precinct in 1st array

Seawulf-16: Calculate box & whisker data
Built by: Pirates






```
{ "Plan 1": {
        1: {
            pop var: 0.35
            geo var: 0.23
            incumbent: true
        },
        2: {
            pop var: 0.35
            geo var: 0.23
            incumbent: false
        }
    },
    Plan 2: {
        1: {
            district_id: 1
            pop var: 0.35
            geo var: 0.23
            incumbent: true
        },
        2: {
            district_id: 2
            pop var: 0.35
            geo var: 0.23
            incumbent: false
        }
    },
```

```
end_collection: {
    "African Amer": [
        {min: 1, q1: 3, median: 4, q3: 10, max:34},
        {min: 1, q1: 3, median: 4, q3: 10, max:34} ],
    "White": [
        {min: 1, q1: 3, median: 4, q3: 10, max:34},
        {min: 1, q1: 3, median: 4, q3: 10, max:34} ],
    "Geographic": [
        {min: 1, q1: 3, median: 4, q3: 10, max:34},
        {min: 1, q1: 3, median: 4, q3: 10, max:34} ]
}
```

African American Pop Var. Table

	Bucket 1	2	3	4	5
Generated plan 1	-5%	1%	2%	15%	45%
plan 2	-10%	-4%	3%	6%	21%
plan 3					
...					
plan 10000					

There are n columns for the n districts in the state
Go through the generated district plans and sort the districts based on a factor: change in race, etc in increasing order
put the statistic into the corresponding bucket, so bucket 1 has the district with lowest statistics
Make a boxes and whisker with the df