Process Workflow

Date: February 11, 2025

Working document to outline the various data processing and integration steps used to compute the disproportion index and other metrics.

Main Content

Database Details

Host: pgsql.dataconn.net Database: cosea_db User: cosea_user Password: CoSeaIndex

Workflow

Filtering List of Schools

Datasets Needed:

- fte2024-1-enroll-demog sch.xlsx: provides list of all K-12 public schools with enrollments, demographics, and grade range. Also contains district id and school id.
- GA_SCHOOL_CONTACT_LIST.xlsx: provides names and addresses and grade range. Add geocodes to the this table.

Process:

1. Create UNIQUESCHOOLID by concatenating DistrictID and School ID (4 digits each). Use zfill(4) + zfill(4). Files: GA_SCHOOL_CONTACT_LIST.xlsx and fte2024-1-enroll-demog sch.xlsxFilter records that >=09 and <=12 for grade range. Note that you may need to split the grade range column on -. Reasoning: (1) school enrollment is not broken down by each grade level. (2) CS teacher counts could include lower grade level CS teachers. Files: fte2024-1-enroll-demog sch.xlsx
2. Filter out alternative schools: anything containing: Title Like "Academy", Title Like "STEM", Title Like "Charter", District Like "State Schools", Title Like "Virtual", Title Like "Institute", Title Like "Foundry", Title Like "Transition", Title Like "Center", Title Like "Online", Title Like "Intervention", Title Like "S.T.E.M.", Title Like "Treatment", Title Like "Youth", Title Like "Home", Title Like "Ministries", Title Like "Chance", Title Like "Comprehensive", Title Like "Career", Title Like "Arts", Title Like "E-Learning", Title Like "Humanities". Create table of schools that meet these criteria. Manually check and maintain this table. Reason: remove schools that may not offer traditional curricula. Files: fte2024-1-enroll-demog sch.xlsx
3. Join the address and geocode from the GA_SCHOOL_CONTACT_LIST.xlsx to fte2024-1-enroll-demog sch.xlsx using UNIQUESCHOOLID.
4. **Resulting data to be stored in a database table: tbl_approvedschools**

5. Create a new geometry column that contains the point geom derived from the x,y coordinates. Use ST_GEOMFROMTEXT(). Also, use SRID 4269 when using this function. This creates point geometries using lat/lon data. Call this column schoolgeom.

Join district information (locale type) to school enrollment file (fte2024-1-enroll-demog sch.xlsx)

Datasets Needed:

- Approved schools: tbl_approvedschools
- NCES District File: DistrictSearch2024_10_14_17.18.49.xlsx

Process:

1. Split State District ID column by '-'. Retain the right hand portion - this is the district ID that will be used to perform the join.
2. Left join from DistrictSearch2024_10_14_17.18.49.xlsx to table tbl_approvedschools using the field created above and District ID.
3. **Resulting data to be stored in a database table: tbl_approvedschools**

Define Buffer Distances for Attendance Zones

Datasets Needed:

- Approved Schools: tbl_approvedschools
- Buffer Table: tbl_bufferlookup

Process:

1. Create the buffer lookup table to store the catchment region buffer definitions by locale type (tbl_bufferlookup). Create the following fields: Locale_Type (matching Locale records in tbl_approvedschools) and buffer_distance.
2. Distances used: City = 1 mile; Suburb = 3 miles; Town = 7 miles; Rural = 18 miles.
3. Join buffer_distance from tbl_bufferlookup to the tbl_approvedschools
4. **Resulting data to be stored in a database table: tbl_approvedschools**

Creating the Attendance Zones

Datasets Needed:

1. TIGER Census Block Groups (CBG) Shapefile: tl_2023_13_bg.shp
2. Approved schools: tbl_approvedschools

Process:

- Load the CBG file tl_2023_13_bg.shp to a postgresql table with postgis extensions enabled. Call this table: tbl_cbg.
  - Use QGIS data import tool to import this data into postgresql.
  - Use the approriate SRID when importing.
  - Rename the geometry column from the default "geom" to cbgpolygeom.
- In the tbl_cbg create a new column of type geom called cbgcentroidgeom
- Update cbgcentroidgeom column using postgis function st_centroid(cbgpolygeom)
- CREATE TABLE AS select tbl_approvedschools.*, tbl_cbg.* from tbl_approvedschools, tbl_cbg where st_intersect(st_buffer(st_transform(tbl_approvedschools.schoolgeom,102005), buffer_distance), tbl_cbg.cbgcentroidgeom). Table: tbl_cbgassignment1
- create a new column in tbl_cbgassignment1 called distance. Compute distance from cbgcentroidgeom to schoolgeom using update tbl_cbgassignment1 set distance = st_distance(st_transform(cbgcentroidgeom,102005),st_transform(schoolgeom, 102005)).
- CREATE TABLE AS select * from tbl_cbgassignment1 group by GEOID, order by distance, limit 1. Call this Table: tbl_cbgassignment
- Identify block groups that were not assigned. Query: CREATE TABLE AS select * from tbl_cbg where geoid not in (select geoid from tbl_cbgassignment). Table name: tbl_cbg_notassigned
- Assign unassigned census block groups to closest school. CREATE TABLE AS select tbl_cbg_notassigned.*, tbl_approvedschools.*, st_distance(tbl_cbg_notassigned. geom, tbl_approvedschools.schoolgeom from tbl_cbg_notassigned, tbl_approvedschools group by GEOID order by st_distance(st_transform(tbl_cbg_notassigned.cbgcentroidgeom, 102005), st_transform(tbl_approvedschools.schoolgeom,102005)) limit 1. Tablename: tbl_cbg_notassigned.
- Union tbl_cbgassignment and tbl_cbg_notassigned. Call this table: tbl_cbg_finalassignment
- CHECK number of rows in tbl_cbg_finalassignment = number of rows in tbl_cbg. No duplicates. *This is to ensure all block groups are assigned to a school.*
- CHECK number of rows in tbl_approvedschools and count the number of unique school IDs in tbl_cbg_finalassignment. *This is to ensure that every school has at least one block group in its attendance zone.*

NOTE: All block groups have been assigned to exactly one school and we have an approved list of schools.

Joining Attribute Data

Datasets Needed:

- Final assignment table: tbl_cbg_finalassignment
- Other tables from the census

Process:

- Join attribute data from any census derived table to the tbl_cbg_finalassignment using GEOID/GEOFIPS