Practical Assignment: Greater Sydney Analysis

Raquel Kampel, Fawaz Al Khreisha

• Dataset Description

Data sets	
stops.txt	This data was obtained from Transport from NSW. It contains static public transport information about individual locations where vehicles pick up or drop off passengers. The data is in General Transit Feed Specification Format (GTFS). Some significant columns include: - Stop_id which is a primary key. It is a unique ID which identifies a location. Multiple routes can use the same stop_id - Stop code is a VARCHAR. It is either a short text or number that identifies the location. - Stop_lat and stop_long are converted to a geom using geopandas and geoAlchemy of type point.
business.csv	The data was obtained from the Australian Bureau of statistics titled "Counts of Australian Businesses, including Entries and Exits". Columns include: - "industry_code" and "industry_name": classified to a single ANZSIC 2006 industry class. - "Sa2_code" and "sa2_name": medium-sized general purpose area which represent a community - "0-50k_businesses"... : these columns show the turnover size ranges - "Total_businesses": total number of specific industry businesses in this sa2 location.
pollingPlaces2019.csv	This data was obtained from Australian Electoral Commision in 2019. It is in a CSV file and contains spatial data of type polygon. Some significant Columns are: - Polling_place_id which is a primary key, - Latitude and Longitude which contains a geometry field Coordinate references system: EPSG: 4283(GDA_1994)
SchoolCatchments.zip	This data was obtained from the NSW government Department of Education. NSW public schools have defined local areas of enrolment. Every child is entitled to enrol in a particular school based on his/her residential address. This zip file contain catchf (future schools), catchp(primary schools) and catchs(high schools) Some significant columns include

population.csv	This data was obtained from the Australian Bureau of statistics. It contains the column sa2_code/name which represents a neighbourhood. Then it contains the number of people for each age group as columns and a final column with the total number of people in this community.
Income.csv	This data describes total earnings statistics by SA2. Some significant Columns are: Sa2_code contains the unique code for the sa2 area the income is representing Sa2_name contains the the unique name for the sa2 area the income is representing Median income contains the median income for each sa2 area
SA2_2021 shape files	This data set contains information about each area, their respective sa2 code and name as well as their geometry. Some significant Columns are: Sa2_code21 contains the unique code for each sa2 area. Sa2_name21 contains the name of each sa2 area Geom: contains a multipolygon geometry of each sa2 area
Extra imported dataset	
Robbery_JanToDec2021.shp	This data was extracted from the NSW bureau of crime statistics and research. This data contains information about the density of data in each area. Some significant columns are: objectid and geometry.
Petrol 147635_00_0.geojson	This data was extracted from Australian government data (data.gov.au) in geojson file. The dataset presents all spatial locations of petrol stations in Australia in point form. Some significant columns are: ObjectID - which is the primary key for this table Geom - contains point geometry for each petrol station

For pre-processing the data we ensure the spatial data types from GeoPandas are the same as those expected by PostGIS. This is done by converting the geom datas' into the **Well-Known Text (WKT)** format. The SRID/ESPG code we decided to use for all the transformations is the ESPG code of 4283 as it was stated in polling places dataset as well as the catchment dataset.

EPSG:4283

GDA94 (Google it)

- **WGS84 Bounds**: 108.0000, -45.0000, 155.0000, -10.0000
- **Projected Bounds**: 108.0000, -45.0000, 155.0000, -10.0000
- **Scope**: Horizontal component of 3D system.
- **Last Revised**: Aug. 27, 2007
- **Area**: Australia - all states

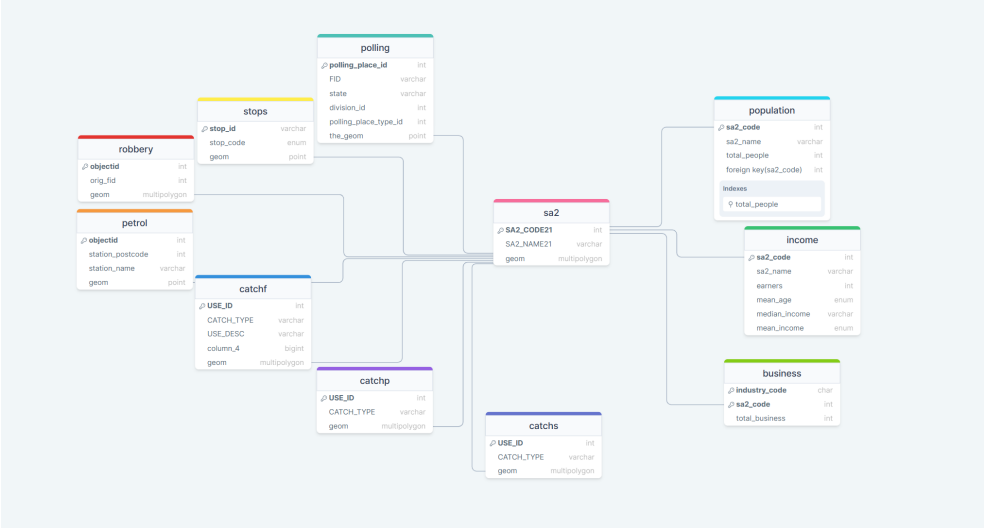
The .copy() function is used to create a copy of all the datasets so as not to change to original data.
In the **Sa2 table**, all polygons were converted to multipolygons. All NA values were dropped and the geometry was converted using WKT.
Seeing as the assignment is analysing Greater Sydney, the .query() function is used to filter to greater Sydney i.e dropping rows where the column GCC_NAME21 is not 'Greater Sydney'. This can be

seen in Figure 1 in the appendix where a map of the Greater Sydney area is shown according to that data in the SA2 file. The columns “AUS_NAME21" and "AUS_CODE21” containing Aus were dropped using the .drop() function. These columns including “LOC_URI21” contained redundant/ unuseful data as all data comes from Australia and the links provided will not be affecting the scores.

In the **catchp, catchs and catchf table,** a WKT element is created and the old geometry column is dropped as it is now redundant.
In the **population** table, all columns with names that start with digits were changed to valid names starting with words.
In the **income table**, all rows with invalid age of ‘np’ were dropped.
In the **business table**, all columns with names that start with digits were changed to valid names starting with words.
In the **stops table**, a geometry column is created as well as a WKT element. The individual latitude and longitude columns were dropped.
In the **polling table**, a new WKT element replaces the old ‘the_geom’ column. Premise address columns were dropped as many values were missing so this information was not useful to us, as well as latitude and longitude columns.

• Database Description

The diagram below shows the schema including all important columns. Columns where no information was used are left out of the schema diagram below. The schema contains eleven tables each with their own primary key.



The population table contains the foreign key (Sa2_Code) which is referenced to the sa2 table. The income and business table do not contain the foreign key (sa2_code) as some of the codes common in both the sa2 table and the income table/ business table, have been dropped. This is because these sa2 codes were part of the ‘rest of NSW’ and not ‘Greater sydney’ GCC. The polling, stops, catchs, catchp and catchf tables are joined with the sa2 table using their geometry which is either in point or multipolygon format.
The catchf, catchs and catchp table were individually joined to the sa2 table using ST.Intersects() on their geometries because young people can still go to nearby schools even if it is not explicitly in their specific geometry. It was individually joined so we can visualise each join that is occurring and minimise any error or conflict that would arise from joining them all together.

The data is then cleaned and indexes are created to optimise query performance on the population and sa2 tables. Specifically as all scores are calculated to a population with at least 100 people, it is useful to create an index on “total_population” in the population table. This was also done on the spatial index for the “geom” column in the sa2 table as it is used throughout most queries for the join operations.

Additional datasets:

The robbery and petrol stations dataset contain primary key objectid and are joined to the sa2 table using ST_contains on their geometry in point multipolygon and point form respectively.

• Score Analysis

The Z-score was calculated by subtracting the mean of each variable from the value of that variable in each neighbourhood and then dividing the result by the standard deviation of that variable in all the neighbourhoods. This standardisation process will help us compare the variables on the same scale and remove the impact of outliers.

1. Retail and health z-scores:

   The Business table was joined with the sa2 table using the sa2_code21. The population table was also joined to the sa2 table to filter out the population of at least 100 people. For retail a WHERE condition was put in place to filter our industry code G and for health the industry code was Q.

   A view was created as the business per 1000 people was needed. This was calculated by dividing the total number of businesses by the total number of people in a specific area and multiplied by 1000.

   A second view was created to store variables such as the mean and standard deviation of the business/1000 column.

   The z-score was then calculated by taking the total business minus the mean and then divided by the standard deviation. This process was done filtering for health and retail to create two separate z scores.

   $$z\_retail = \frac{((retail\ per\ 1000\ people) - mean(retail))}{std(retail)}$$

   $$z\_health = \frac{((health\ per\ 1000\ people) - mean(health))}{std(health)}$$

2. Stops z-score:

   A view was created by joining sa2 to population using the sa2_code and then sa2 was joined to stops using geometry. This was done by using the ST.Contains function which returns true when geometry B is fully inside geometry A. We decided to use this function to count the number of stops we can find inside a specific sa2 region. As every stop has a geometry but not every geometry has a stop, sa2 table was entered first.

   Another view was created to store the mean and standard deviation of the count. The z score was then calculated by taking the count of the stop ids per area minusing the mean and then dividing by the standard deviation.

   $$z\_stops = \frac{((stops\ count) - mean(stops))}{std(stops)}$$

3. Polling z-score:
   The polling z score is calculated in the same was as stops z-score

   $$z\_polls = \frac{((polls) - mean(polls))}{std(polls)}$$

4. Catchment z-score:
   The school z-score is calculated in the same way, however the score is found per 1000 young people. Thus the number of young people is filtered out, the score is divided by the addition of the people and times by 1000.

   $$z\_schools = \frac{((schools\ count) - mean(schools))}{std(schools)}$$

5. Total z score is then calculated by adding the z-scores of all the variables together. Finally, we can pass this total z-score through the sigmoid function to obtain the score for each neighbourhood:

   $$Total\ z\ score = z\_retail + z\_health + z\_stops + z\_polls + z\_schools$$

6. The sigmoid function maps any value to a range between 0 and 1, with values close to 0 indicating low scores and values close to 1 indicating high scores.

   $$score = S(Total\ z\ scores) \qquad S(x) = \frac{1}{1+e^{-x}}$$

   ADDITIONAL DATA

7. robbery z-score:
   A view was created by joining sa2 to population using the sa2_code and then sa2 was left joined to robbery using geometry. This was done by using the ST.Contains function which returns true when geometry A is fully inside geometry S. The sum of the number of robberies is then added up and placed in a new column 'c'. COALESCE function is used to substitute zero for null when needed. The results are filtered to areas where the population is over 100 people.
   Another view called robbery_stats is created to store the mean and standard deviation of the count 'c' per sa2 region. The z score is then calculated by taking the count of the robberies per sa2 area minusing the mean and then dividing by the standard deviation.

   $$z\_robbery = \frac{((robberies\ count) - mean(robbery))}{std(robbery)}$$

8. Petrol stations z-score:
   A view called petrol_count is created in the same way as a robbery view.
   Another view called petrol_stats is created to store the mean and standard deviation of the count 'c' per sa2 region. The z score is then calculated by taking the count of the petrol stations per sa2 area minusing the mean and then dividing by the standard deviation.

   $$z\_petrol = \frac{((petrol\ count) - mean(petrol))}{std(petrol)}$$

9. <u>The final z-score and sigma score:</u>
   The z-scores from the schools, polling, stops, health, and retails tables are added up. The robbery z score is then subtracted as robberies have a negative effect on the resources of a sa2 area. The petrol stations z -score is added as petrol stations have a positive effect on the resources of an area.
   The sigma function is then calculated on the new z-score using the following formula and listed in a table:

$$Sigmoid = \frac{1}{1+e^{-(z\_retail + z\_health + z\_stops + z\_polls + z\_schools - z\_robbery + z\_petrol}}$$

• Correlation Analysis

An analysis is computed to find the correlation between the sigma score of how well each region is resourced and the median income of each region.
When excluding extra datasets, sigma score is calculated on average z score of the number of retail and health businesses, number of stops, number of polling places and number of public schools in each region. The median income was given in the income table.
This resulted in a correlation coefficient r of -0.0069. Which indicates no correlation as the score is almost at 0. This can be seen and supported by Figure 3 in the Appendix.
A correlation coefficient of 0 indicated no linear relationship between the variables. The absence of a correlation does not imply an absence of relationship, as there may be a nonlinear complex relationship between the variables.
Factors/limitations that led to no correlation include:

- The report defines ”well-resourced” as addition of public transport stops, however some sa2 areas which are well resourced have fewer stops as the area defined is smaller.
- Another addition is school catchment areas. Only public schools are listed in this data. Some well resourced areas do not have as many public schools as there is more funding towards private schools
- Another constraint noticed was that the median income was used as a variable. A better representation of the relationship between the variables would have been seen if the mean income was used. This is because the mean is more sensitive to all data points in the dataset and is a measure of the central tendency. When using the mean income value, more information can be retained about the magnitude and dispersion of the variable allowing for a more comprehensive understanding of the relationship

From the factors listed above, it can be seen that the scores do not in fact mean the area is well resourced, thus making the correlation sensible and intuitive.
As can be seen in the appendix figure 2. Woronora Heights has a relatively low score just above 0.4. This is probably because it is a suburb 29 km away from sydney with a population of only 3551 people. Annandale, which is only 5 km west of Sydney's central business district, has a perfect score of one and a higher population of 9516 people.

As a whole, the implications of extending the formula with the additional dataset do not result in a significant change of the scores. However, some noticeable changes can be seen in Woronora heights where the score changes from 0.428 to 0.332059. This is due to the area having no petrol stations which impacts how well resourced it is. Summerland Point however, increases from 0.5768 to 0.7 due to the 2 petrol stations and no robberies. There are no noticeable trends in the scores, as we have stated above that the z scores does not necessarily imply the areas are well resourced

Appendix

Figure 1. Map of Greater Sydney

```
greater_sydney = sa2og[(sa2og['GCC_NAME21']=='Greater Sydney')]
greater_sydney.plot(cmap='Set2', figsize=(10, 10))
```

```
<Axes: >
```

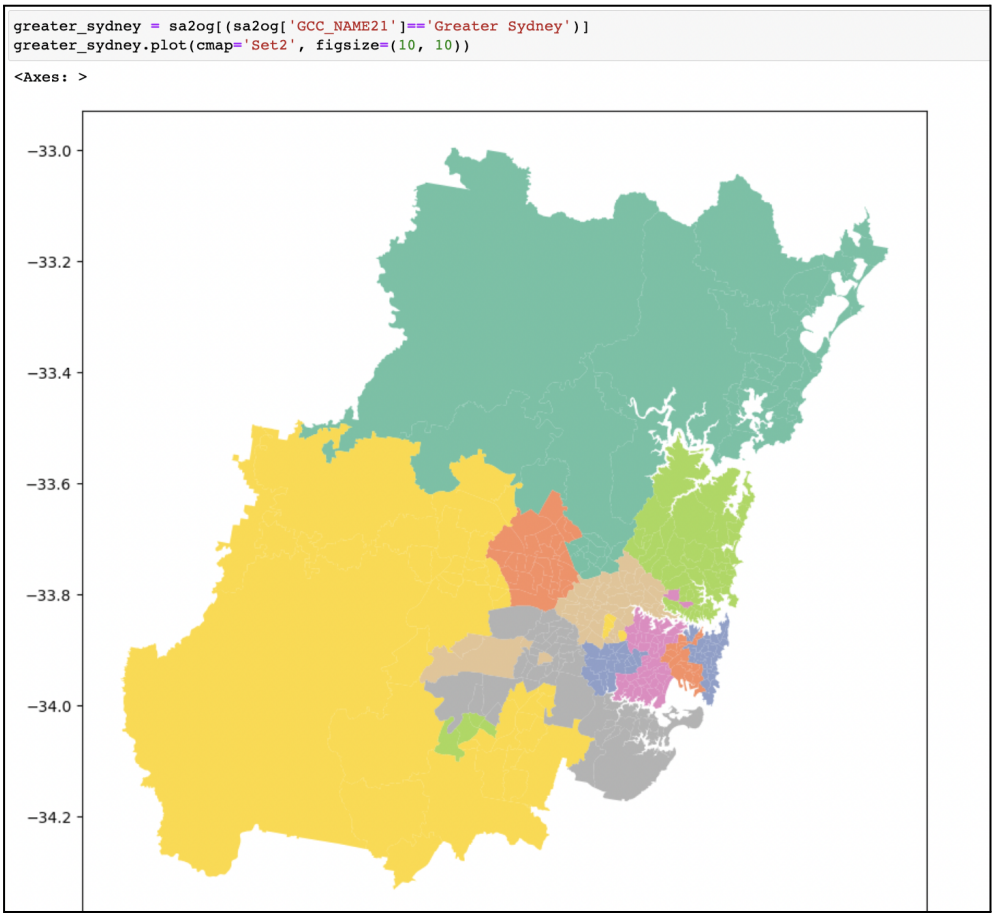


Figure 2.

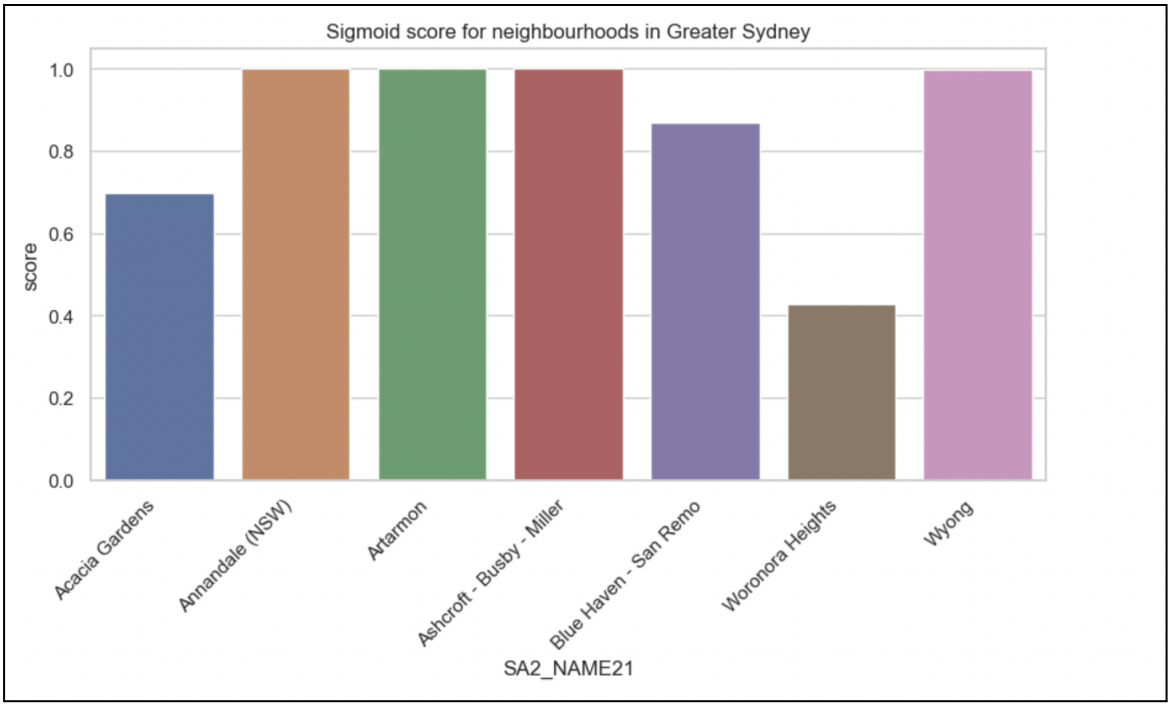


Figure 3. Correlation coefficient regression line