Major Project: Analysis of Indian demographics in Melbourne and the access to religious institutions

Shalika D, Geospatial Programming (GEOM 2157)
Masters in Geospatial Science, RMIT University
Mentor: Dr Amy Griffin
Student ID: s3757053

Abstract

Melbourne is known for its cultural diversity. It is said that around 140 different cultures exist in Melbourne .56% of people in Melbourne are born overseas (Melbourne, 2019).There is a large community of Indians in Melbourne, consisting of 4% of the population as per census 2016 and 35% of migrants to Australia choose Melbourne (ABS, 2016; liveinmelbourne). By understanding the distribution of these communities it is possible to allocate better resources at the right locations. Understanding the accessibility to resources is very important for Urban Planners and builders to decide the right locations for new establishments. Writing a script on python will allow us to make this generic to any community and will allow wider study on this subject.

This project considers the Indian community as subject for study, but the study can be formulated to analyse any community or groups. The study is mainly based on two main elements i.e. people and religion. India is a land with plethora of religions, but the major ones being Christianity, Islam and Hinduism. These three religions are taken into consideration for this projects and act as points of interest. Large amount of data is available in the census data portal. But the task is to collect and analyse this data to produce useful information.

Contents

Major Project: Analysis of Indian demographics in Melbourne
and the access to religious institutions

Aim

The aim of this project is to write a python script, which can run on python console, and that automates the process of analyzing the number of Indians in each electoral division and locates the accessibility to religious institutions in each of these. This analysis will help create map to show how accessible these amenities are, and can be used to understand where new religious institutions can be located.

Objectives

- Create a script that opens layers on QGIS
- Automatically reclassifies locality and urban growth boundary polygons based on population of Indians
- Creates multi ring buffers around religious institutions vector point data to show comfortable walking and driving distances
- Intersect buffer and network data to show accessibility.

Intended End Users

- **Urban Planners** – Such analysis is widely used by planners to assist with planning layouts and estimations of requirements. Knowing the population of a particular community in a particular location even allows them to provide better public amenities based on the kind of people living there. It is also important to provide walkable neighborhoods and to promote community activities, this analysis will allow to easy analyze walkability to any amenity.
- **Government and policy makers** – This analysis can be used to assist with land use allocation processes and also to understand where to place new community based amenities. It also allows better policies that satisfy residents and promote better accessibility.
- **Investors** – Can be used by private investors to know the best locality to place their venture.

Methodology

- Acquire census data regarding population in each locality. Vector data on point locations of religious spaces. For this project we consider the three major religions in India i.e. Christianity, Hinduism and Islam. Hence we are looking at point locations for churches, mosque and temples.
- Get data on urban growth boundary. This will allow us to see how exactly the city is growing. The scale of this project is restricted to the metropolitan regions of Melbourne and not the entire state.
- Intersect all this data to produce a base map
- Create a choropleth map and classify data and project on map based on population of people in each locality.

- Project vector locations on map. Vector data wasn't readily available and has been sourced from open street maps(OpenStreetMap, 2019) and Google maps(Maps, 2019)
- Overlay and Intersect the maps
- Create Multi ring buffers of walking distances (1km considered as comfortable walking distance), and driving distances (up to 5km considered as comfortable driving distance) around these religious centers.
- Intersect buffer with road network map and classify the road network based on walkability and driving distances.
- Apply graduated colour scheme to the layer.
- The acquired layers can be used to create maps of entire area and of a particular locality

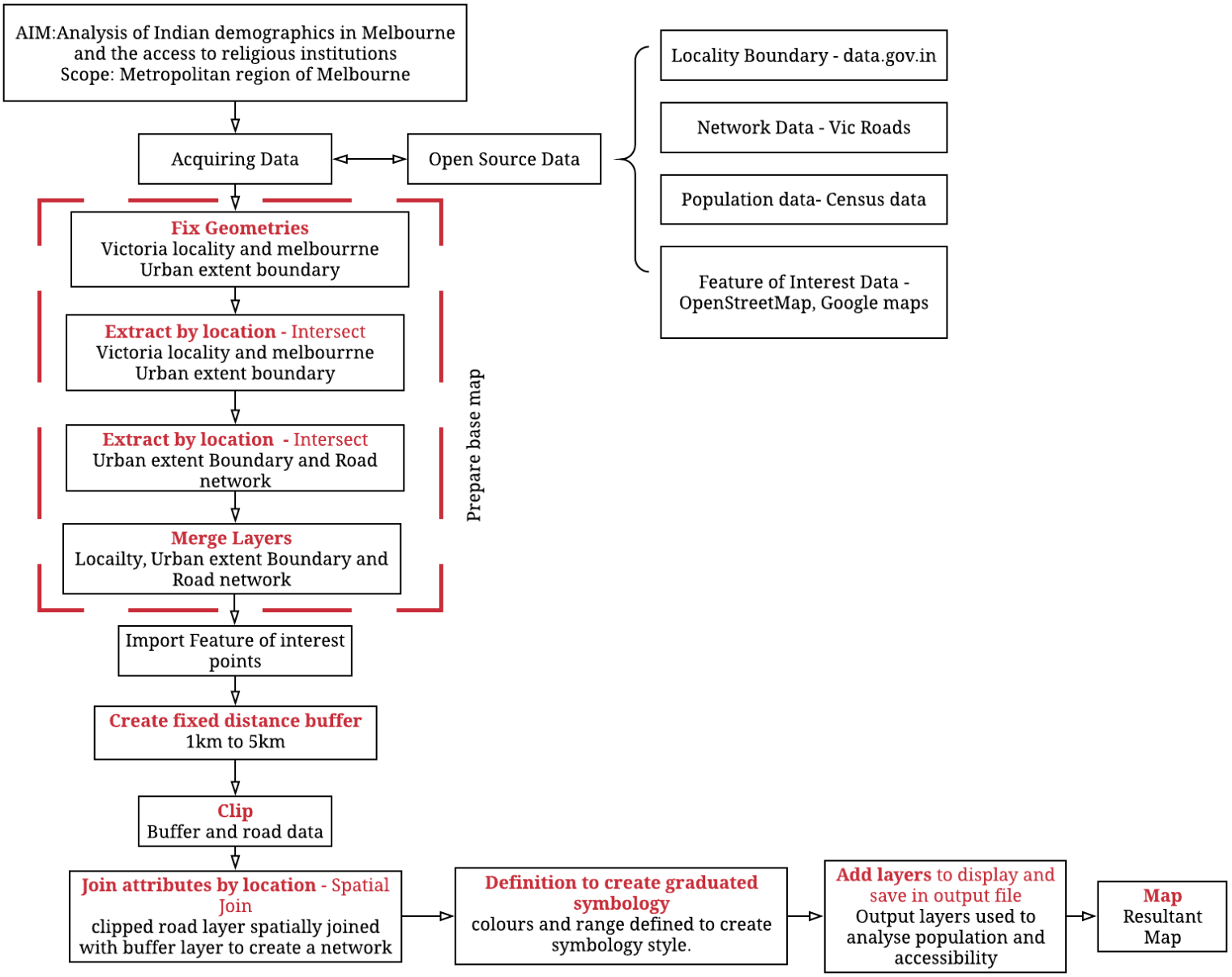

Figure 1: Table of Methodology

Roadmap/Assemble code

Tools described below are as per the use in the program. Few tools might be used more than once and haven't been described again in the roadmap.

1: Import Vector files and add to view
Tool Used: "**iface.addVectorLayer (data_source , layer_name , provider_name)**"

Parameters:
• DATA_SOURCE: Filepath, str
• LAYER_NAME: str
• PROVIDER_NAME: str

2: Fix geometries tool used to fix null values in attribute table, which allows us to use intersection tools.
Tool Used: "**native:fixgeometries**"

Parameters:
• INPUT: Input layer (Layer ID, Layer source, Layer name,
QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• OUTPUT: destination vector file

Output: Fixed geometries

3: Extract by location used to intersect locality layer with urban growth boundary layer, to extract project area.
Tool Used: "**native:extractbylocation**"

Parameters:
• INPUT: Input layer (Layer ID, Layer source, Layer name,
QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• PREDICATE: Where the features (0: intersect, 1: contain, 2: disjoint, 3: equal, 4: touch, 5: overlap, 6: are within, 7: cross) (int, str, QgsProperty)
• INTERSECT: Segments (Layer ID, Layer source, Layer name,
QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• OUTPUT: destination vector file

Output: Extracted (location)

4: Merge church, temple and mosque layers into a single layer to simplify creating buffers

Tool Used: " **native:mergevectorlayers**"

Parameters:
• LAYERS: Input layer (Layer ID, Layer source, Layer name, QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• CRS: Destination CRS (str, QgsCoordinateReferenceSystem, QgsMapLayer, QgsProcessingFeatureSourceDefinition, QgsProperty)
• OUTPUT: destination vector file

Output: Merged

5. Create multiple buffers at 1km distances to denote walking and driving distances
Tool Used: " **saga:fixeddistancebuffer**"

Parameters:
• SHAPES: Input layer (Layer ID, Layer source, Layer name, QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• DIST_FIELD_DEFAULT: Buffer distance (int, float, QgsProperty)
• NZONES: Number of Buffer Zones (int, float, QgsProperty)
• DISSOLVE: Dissolve Buffers(bool, int, str, QgsProperty)
• BUFFER: Buffer (destination vector file)

Output: Buffer

6. Clip road layer with buffer layer (to clip roads as per buffer distances)
Tool Used: " **native:clip** "

Parameters:
• INPUT: Input layer (Layer ID, Layer source, Layer name, QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• OVERLAY: Overlay layer (Layer ID, Layer source, Layer name, QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• OUTPUT: Clipped (destination vector file)

Output: Clipped

7. Intersect Road network and buffer
Tool Used: " **qgis:joinattributesbylocation**"

Parameters:

• INPUT: Input layer (Layer ID, Layer source, Layer name,
QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• JOIN: Join layer (Layer ID, Layer source, Layer name,
QgsProcessingFeatureSourceDefinition, QgsProperty, QgsVectorLayer)
• PREDICATE: Geometric predicate (0: intersect, 1: contain, 2: equals, 3: touch, 4: overlap, 5: are within, 7: cross) (int, str, QgsProperty)
• JOIN_FIELDS: Fields to add (str, QgsProperty)
• METHOD: Join type (0: Create separate feature for each located feature (one-to-many), 1: Take attributes of the first located feature only (one-to-one))(int, str, QgsProperty)
• OUTPUT: destination vector file

Output: Joined layer

8: Add Layers to view
Tool Used: "**QgsVectorLayer.addLayer()**"
Parameters:
• Data source (filepath)
• Layer name (str)
• Provider name

9: Update Features
Tool Used: "**updateFeature()**"

Parameters:
• List of features

Timeline

1. Gathering Information: 5 – 6hrs
2. Extracting and keeping only useful information: 3 -4 hrs.
3. Updating roadmap to suit derived information: 2hrs
4. Writing draft code/Test code: 6 – 10hrs.
5. Revising and Testing: 10hrs.
6. Preparing and submitting final report: 3 – 4hrs.

Existing Studies and Resources

Studies have been conducted regarding the Indian communities and international students. However very little research has been done regarding accessibility to religious spaces.

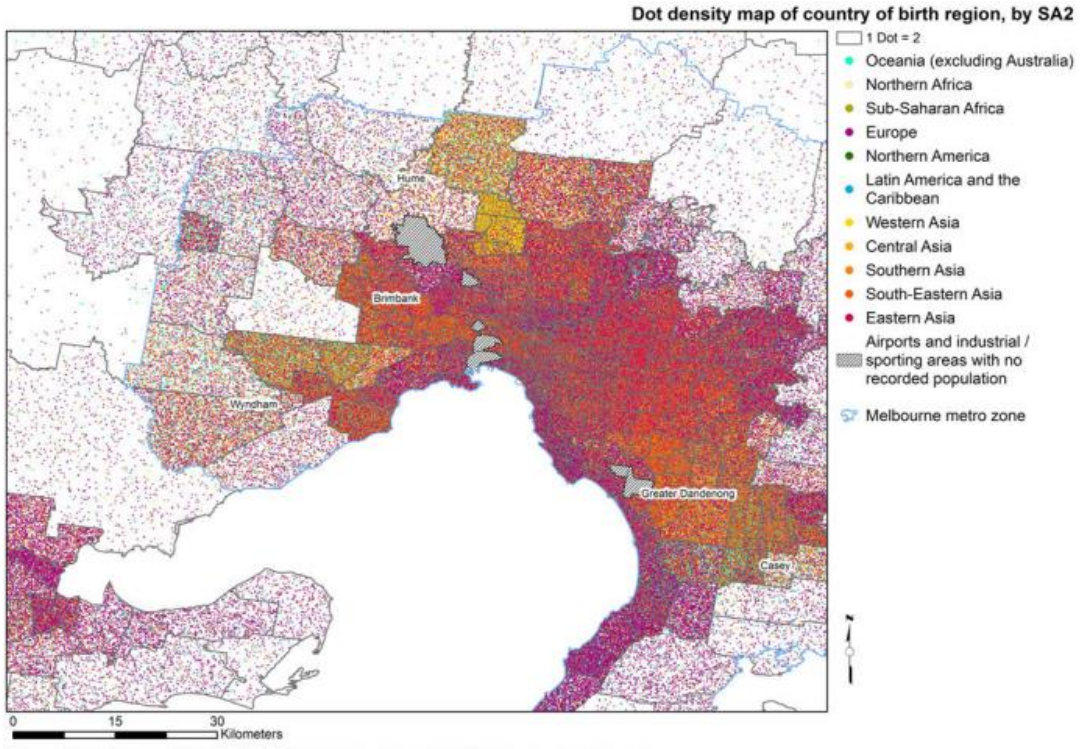

Figure 2: Dot density Map by country of birth (Melanie Davern, 2015)

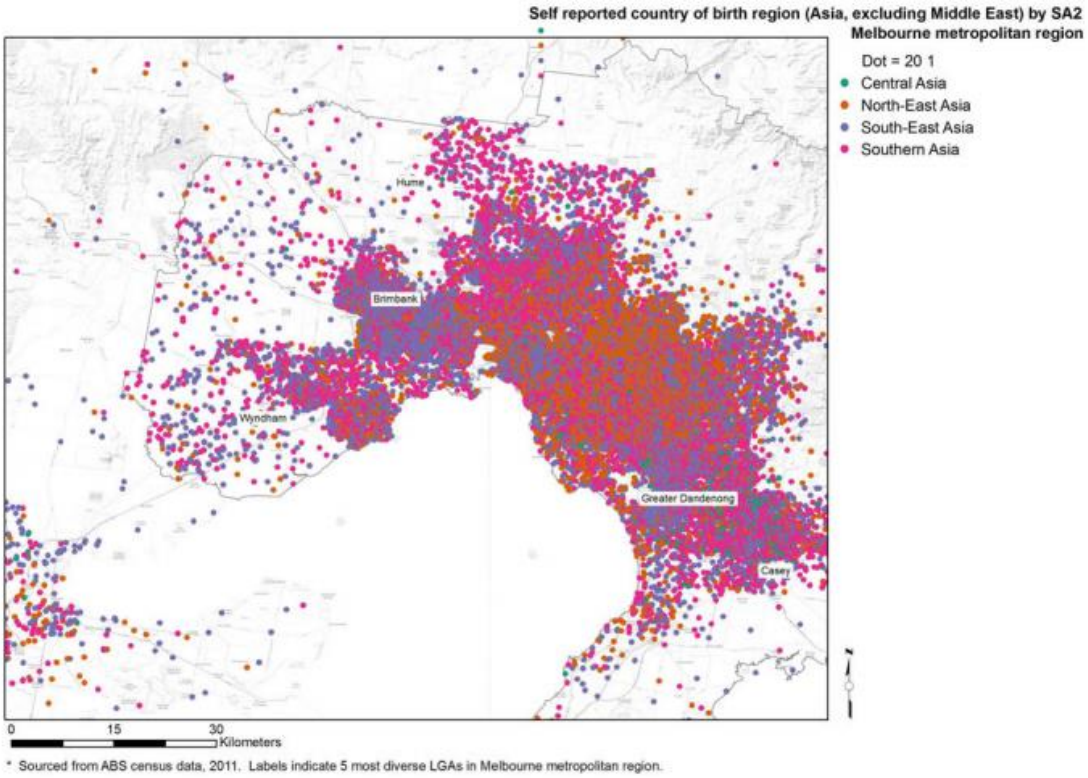

Figure 3: Dot density of country of birth by region (Melanie Davern, 2015)

Figure 2: This map shows the distribution of people based on the country of birth, even though it is possible to recognize areas with dense population and areas with low population, it is really hard to distinguish areas based on prominent country of birth. A categorized map would be a better option for this purpose. The large number of classes/countries also make it really confusing to read.
Figure 3: Even though this map shows dot density only for few regions, it is still difficult to clearly understand the distribution pattern, however this could be a good way to represent population density.

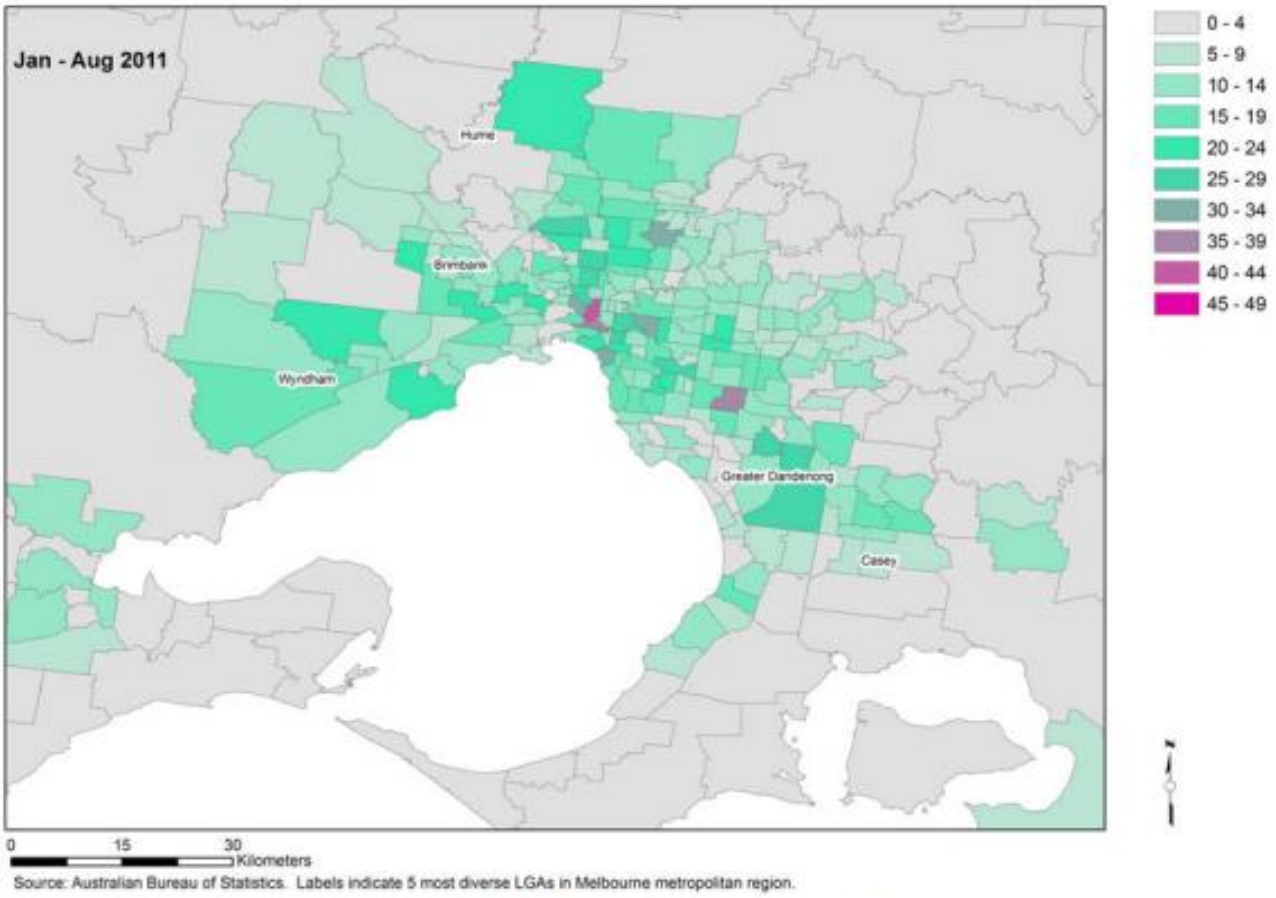

Figure 4: Number of Countries of Birth in Melbourne(Melanie Davern, 2015)

Choropleth representation for this purpose is much more clear and easier to understand for a user. It is easy to understand which region is most multicultural. Hence this kind of map has been used for the purpose of this project.

Results

The program overlays elements which can be used to create useful maps. The code automates the process of clipping the study boundary, intersecting and merging multiple layers and classifying layers based on attributes. The process of composing the maps based on what information is to be show is up to the user. The code is written to be used on python console. Census data for any community and any feature of interest can be used. The map can be used to show a single region or the whole study area.

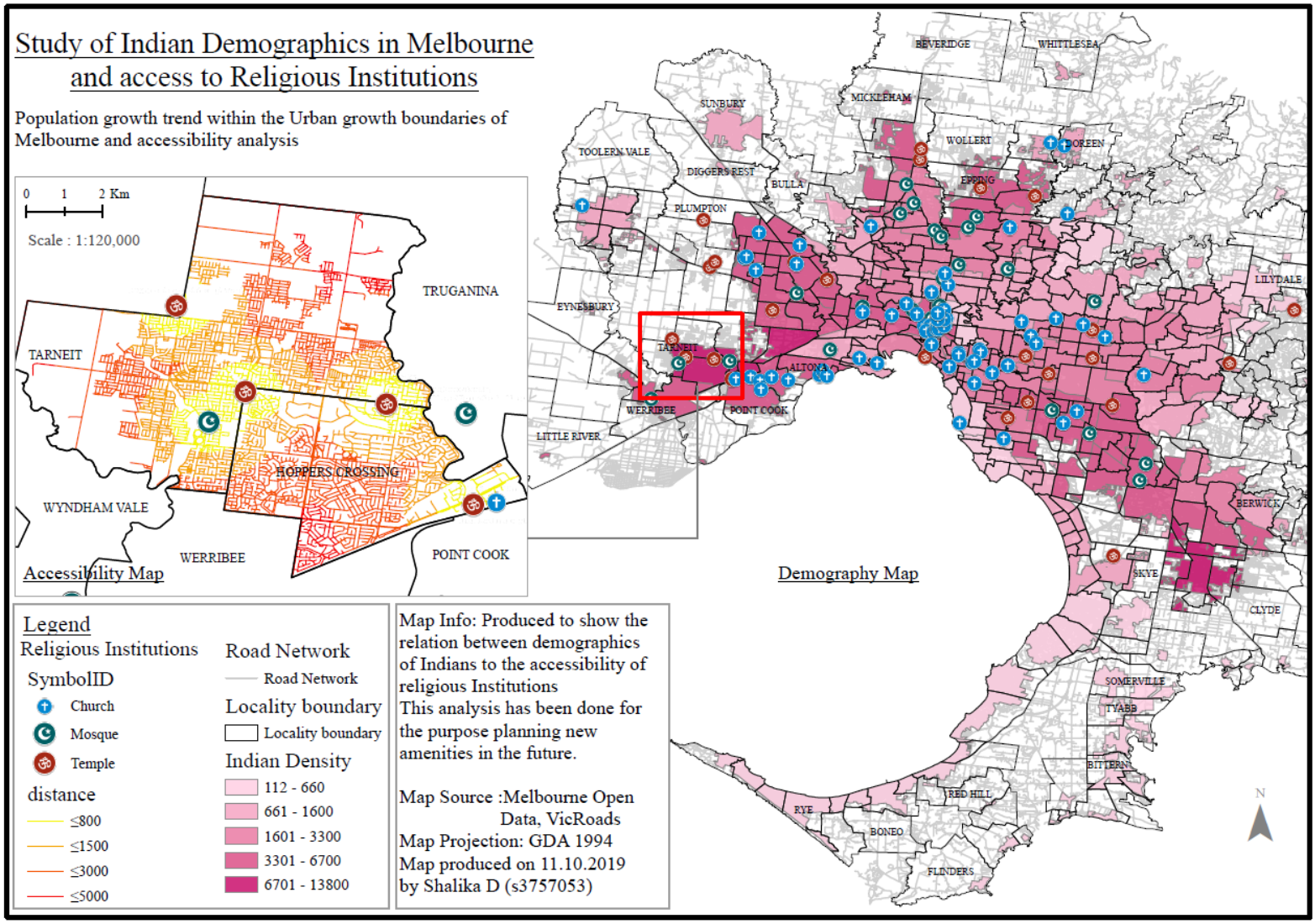

Figure 5: Example for a composed map using derived data

In the example above, we can see how in the demography map, layers are used effectively to represent population in each locality overlaid with the urban growth boundaries. Here the network layer is set to grey to reduce confusion. This gives clear idea about the growth of the city. One of the locality is then zoomed in to show a clear map for the distances from the points of interest. Here, Hoppers crossing having the highest number of Indians is used for analysis.

Inferences from the map

• Suburbs with Higher population of Indians
- Tarneit, Oakleigh, Cranbourne, Altona.

• Suburbs with Low population of Indians
- Mornington, Nepean, Macedon, Eildon.

• General Trend
- Most churches are found around the CBD.
- Temples and mosques are spread out mostly around outer suburbs

• Accessibility
- Churches are located at prominent locations and are easily Accessible by walk

This data can also be represented in the form of a map

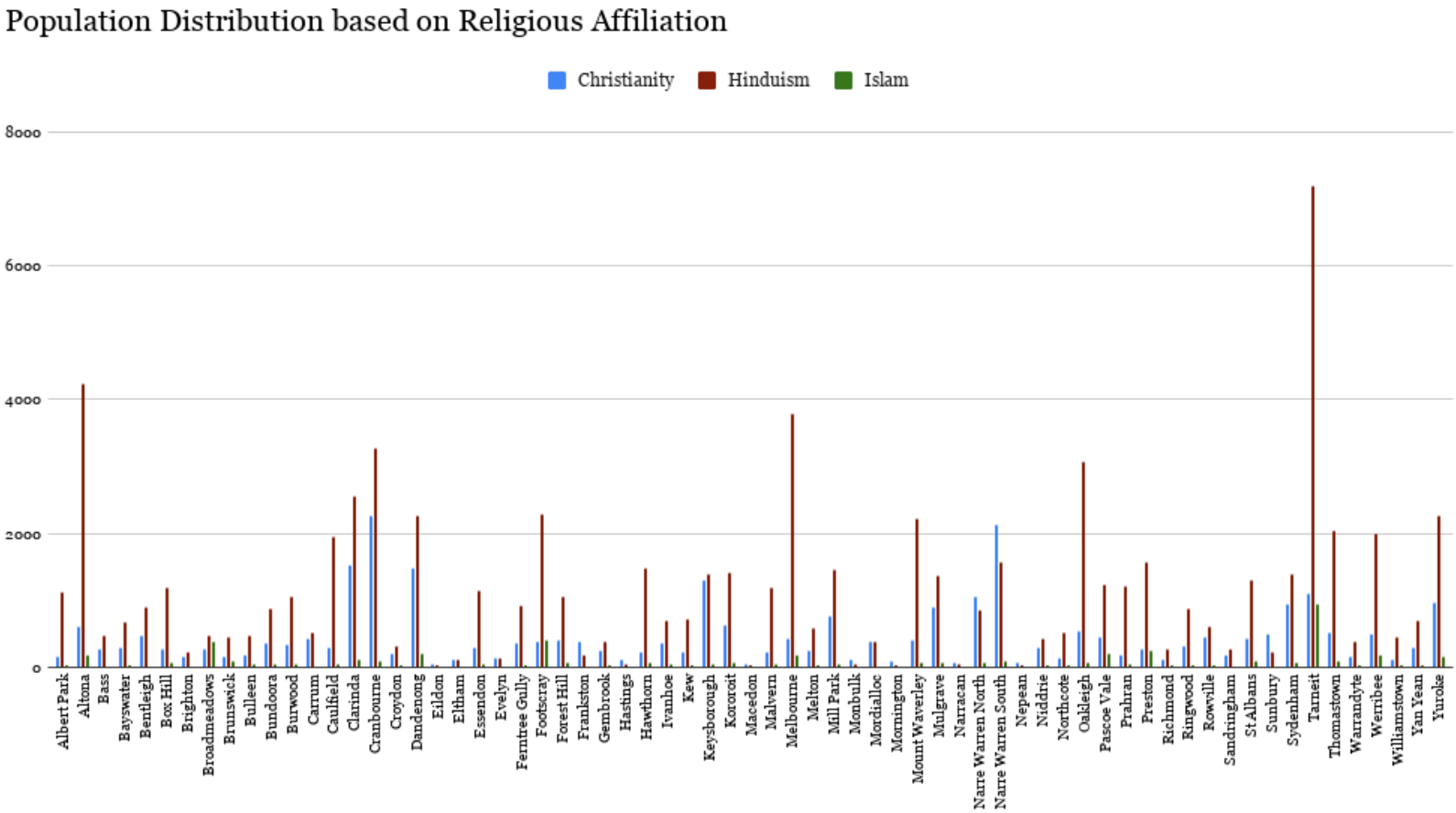

Figure 6: Table of population distribution

Where y- axis shows population count, x- axis shows the locality and columns represent different religions.

Recommendations

- The project can be replicated to analyse the population of any country or any religion.
- Proof check has to be carried out before using any dataset, as population and point of interest data keeping changing.
- Derived results need to be updated regularly to track the change in population growth and should be update with added amenities

Areas for future study

- Network analysis can be done based on cost and distance or road directions to get more detailed analysis
- Analysis can also be done based on rent, employment etc to understand why certain suburbs are popular among particular communities.

Conclusion

Vast amount of demographic data is available on open source webpages, but it is important to harness this data and use it to produce meaningful results. Comparing a particular community and available amenities will allow us analyze areas based on proportion of supply to demand. Planners and builders can use this data to decide best locations for new establishments. Creating a python code which can easily prepare this data, will drastically reduce the time and energy required to execute each step manually.

References

ABS. (2016). Population and Housing. Retrieved from https://www.melbourne.vic.gov.au/about-melbourne/research-and-statistics/city-population/Pages/community-profiles.aspx

liveinmelbourne. Melbourne's Indian community. Retrieved from https://liveinmelbourne.vic.gov.au/discover/multicultural-communities/indian

Maps, G. (2019). Retrieved from https://www.google.com.au/maps/place/Melbourne+VIC/@-37.9722342,144.7729553,10z/data=!3m1!4b1!4m5!3m4!1s0x6ad646b5d2ba4df7:0x4045675218ccd90!8m2!3d-37.8136276!4d144.9630576

Melanie Davern, D. W., Carl Higgs, Helen Dickinson, Jenny Phillimore. (2015). Superdiversity in Melbourne.

Melbourne, C. o. (2019). Multicultural communities. Retrieved from https://www.melbourne.vic.gov.au/about-melbourne/melbourne-profile/multicultural-communities/Pages/multicultural-communities.aspx

OpenStreetMap. (2019). Retrieved from https://www.openstreetmap.org/search?query=melbourne#map=13/-37.8131/144.9442