Melbourne Suburbs Housing Data Analysis using Foursquare API

Introduction

Melbourne is a popular destination in Australia for immigrants to settle down in and also for locals to move to due to it's vast amount of facilities and ever growing opportunities for career and education. Hence it provides an ideal location for an individual to settle down with a family.

The intention of this project will be to provide a versatile selection scheme with respect to which suburb an individual wants to settle down in. It allows potential buyers to:

a) Select suburbs within the same cluster and have similar facilities in the vicinity, while not experiencing the hustle and bustle of the city.
b) Select suburbs in the same cluster, with a lower housing price with similar facilities in the vicinity
c) Protect themselves against real estate agents scamming them with higher prices by knowing the average price of a given suburb.

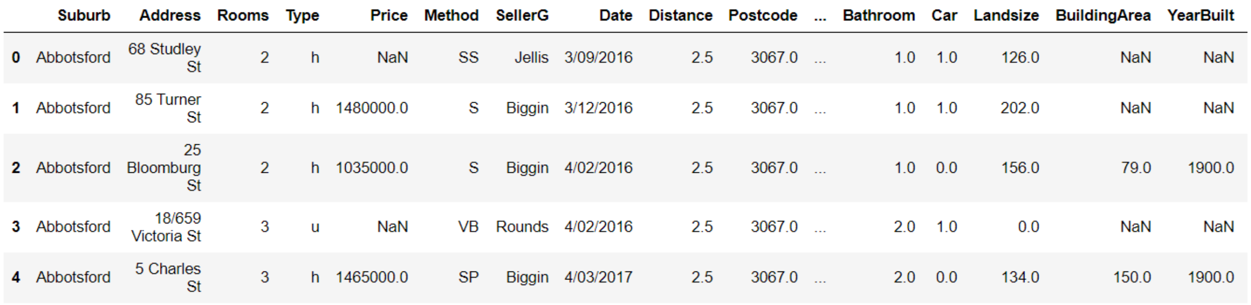
Data Being Used in this Project

In this project we aim to use the FourSquare API along with the dataset for Melbourne housing prices taken from this github site (https://raw.githubusercontent.com/nagoya-foundation/r-functions-performance/master/data/Melbourne_housing_FULL.csv). The Foursquare API will be used to find the most popular venue categories in each suburb in the dataset. The dataset of Melbourne housing prices will be used to find the average housing price for each suburb.

To plot a clear folium plot showing the variation of prices and the presence of different clusters, GeoJson data from this github site (https://github.com/tonywr71/GeoJson-Data) was used.

Methodology

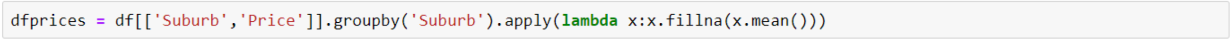
The dataset containing the Melbourne Housing Prices consisted of missing values as shown in the following screenshot:



	Suburb	Address	Rooms	Type	Price	Method	SellerG	Date	Distance	Postcode	...	Bathroom	Car	Landsize	BuildingArea	YearBuilt
0	Abbotsford	68 Studley St	2	h	NaN	SS	Jellis	3/09/2016	2.5	3067.0	...	1.0	1.0	126.0	NaN	NaN
1	Abbotsford	85 Turner St	2	h	1480000.0	S	Biggin	3/12/2016	2.5	3067.0	...	1.0	1.0	202.0	NaN	NaN
2	Abbotsford	25 Bloomburg St	2	h	1035000.0	S	Biggin	4/02/2016	2.5	3067.0	...	1.0	0.0	156.0	79.0	1900.0
3	Abbotsford	18/659 Victoria St	3	u	NaN	VB	Rounds	4/02/2016	2.5	3067.0	...	2.0	1.0	0.0	NaN	NaN
4	Abbotsford	5 Charles St	3	h	1465000.0	SP	Biggin	4/03/2017	2.5	3067.0	...	2.0	0.0	134.0	150.0	1900.0

Given that our interest is in the Price of a house, we apply the Groupby method along with a lambda expression to calculate the average price of a suburb and use that value to fill in the NaN values in the Price column.

The following code is used to accomplish the above requirement:

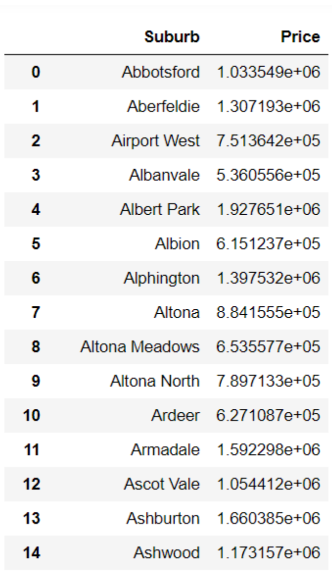


```
dfprices = df[['Suburb','Price']].groupby('Suburb').apply(lambda x:x.fillna(x.mean()))
```

Nevertheless, some NaN values will still occur due to certain suburbs occurring only once in the dataset with a NaN for the price. The mean on 1 NaN value will always be NaN. Hence we drop such instances from the analysis.

The average price of a house in a particular suburb is calculated to obtain the following DataFrame presented in Figure 1:

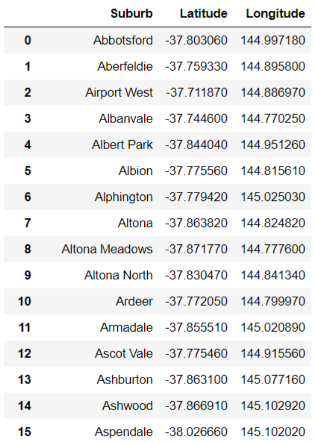
Figure 1: Part of the DataFrame showing the average price of a suburb



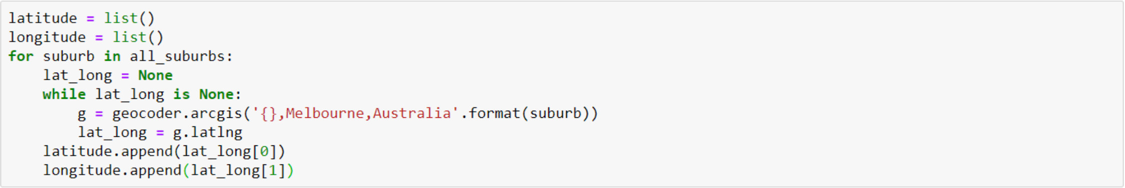
	Suburb	Price
0	Abbotsford	1.033549e+06
1	Aberfeldie	1.307193e+06
2	Airport West	7.513642e+05
3	Albanvale	5.360556e+05
4	Albert Park	1.927651e+06
5	Albion	6.151237e+05
6	Alphington	1.397532e+06
7	Altona	8.841555e+05
8	Altona Meadows	6.535577e+05
9	Altona North	7.897133e+05
10	Ardeer	6.271087e+05
11	Armadale	1.592298e+06
12	Ascot Vale	1.054412e+06
13	Ashburton	1.660385e+06
14	Ashwood	1.173157e+06

To visualize the location of suburbs in Melbourne, the latitude and longitude of a suburb is found using the Geocoder library and is presented below in Figure 2:

Figure 2: Part of the DataFrame showing the latitude and longitude of a suburb

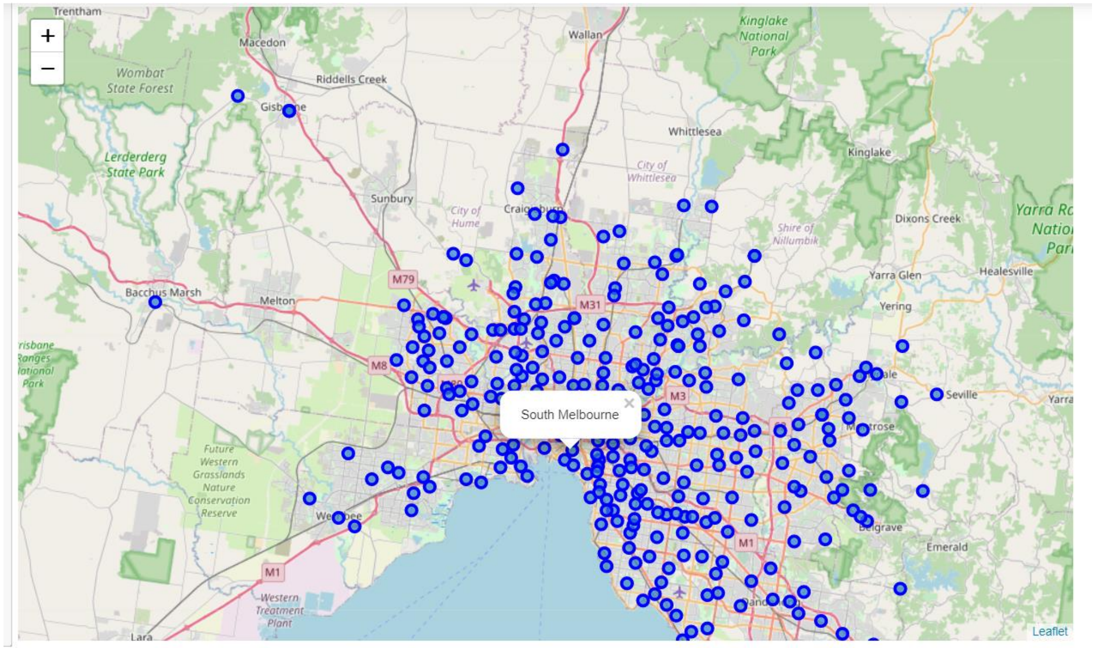


	Suburb	Latitude	Longitude
0	Abbotsford	-37.803060	144.997180
1	Aberfeldie	-37.759330	144.895800
2	Airport West	-37.711870	144.886970
3	Albanvale	-37.744600	144.770250
4	Albert Park	-37.844040	144.951260
5	Albion	-37.775560	144.815610
6	Alphington	-37.779420	145.025030
7	Altona	-37.863820	144.824820
8	Altona Meadows	-37.871770	144.777600
9	Altona North	-37.830470	144.841340
10	Ardeer	-37.772050	144.799970
11	Armadale	-37.855510	145.020890
12	Ascot Vale	-37.775460	144.915560
13	Ashburton	-37.863100	145.077160
14	Ashwood	-37.866910	145.102920
15	Aspendale	-38.026660	145.102020



```
latitude = list()
longitude = list()
for suburb in all_suburbs:
    lat_long = None
    while lat_long is None:
        g = geocoder.arcgis('{},Melbourne,Australia'.format(suburb))
        lat_long = g.latlng
    latitude.append(lat_long[0])
    longitude.append(lat_long[1])
```

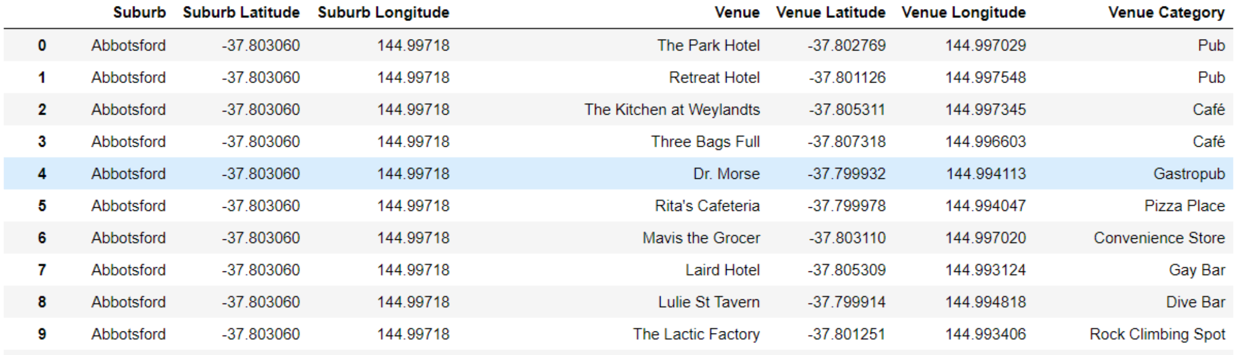
Figure 3: Visualizing the suburbs of in melbourne.



Obtaining the top 50 venues in the vicinity of a suburb

Using the FourSquare API, requests are made to obtain the json data of all the venues in the vicinity of all the suburbs in the dataset. Figure 4 shows the results of scraping data using the Foursquare API for the Abbortsford suburb.

Figure 4: Venues in Abbotsford

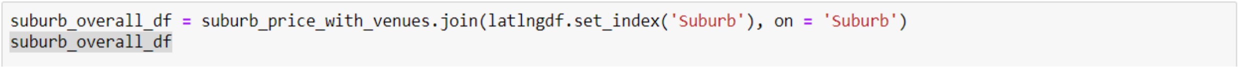


	Suburb	Suburb Latitude	Suburb Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Abbotsford	-37.803060	144.99718	The Park Hotel	-37.802769	144.997029	Pub
1	Abbotsford	-37.803060	144.99718	Retreat Hotel	-37.801126	144.997548	Pub
2	Abbotsford	-37.803060	144.99718	The Kitchen at Weylandts	-37.805311	144.997345	Café
3	Abbotsford	-37.803060	144.99718	Three Bags Full	-37.807318	144.996603	Café
4	Abbotsford	-37.803060	144.99718	Dr. Morse	-37.799932	144.994113	Gastropub
5	Abbotsford	-37.803060	144.99718	Rita's Cafeteria	-37.799978	144.994047	Pizza Place
6	Abbotsford	-37.803060	144.99718	Mavis the Grocer	-37.803110	144.997020	Convenience Store
7	Abbotsford	-37.803060	144.99718	Laird Hotel	-37.805309	144.993124	Gay Bar
8	Abbotsford	-37.803060	144.99718	Lulie St Tavern	-37.799914	144.994818	Dive Bar
9	Abbotsford	-37.803060	144.99718	The Lactic Factory	-37.801251	144.993406	Rock Climbing Spot

One-hot encoding is applied on the Venue categories followed by grouping the data with respect to the Suburb. The Grouped data is averaged across the columns to find the frequency of occurrence of a particular venue category in that suburb.

KMeans clustering with 20 clusters is applied to the resulting dataset to obtain the clusters of suburbs which have similar frequency of venue categories occurring in the suburb.

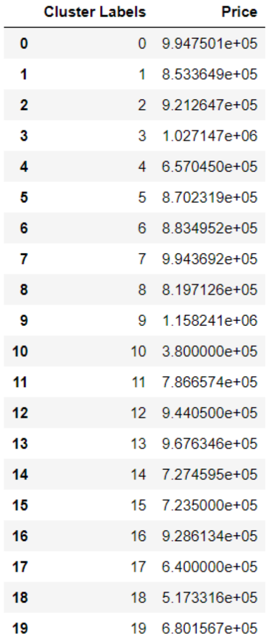
The resulting dataset is merged with the Price of the respective suburb using the following function:



```
suburb_overall_df = suburb_price_with_venues.join(latlngdf.set_index('Suburb'), on = 'Suburb')
suburb_overall_df
```

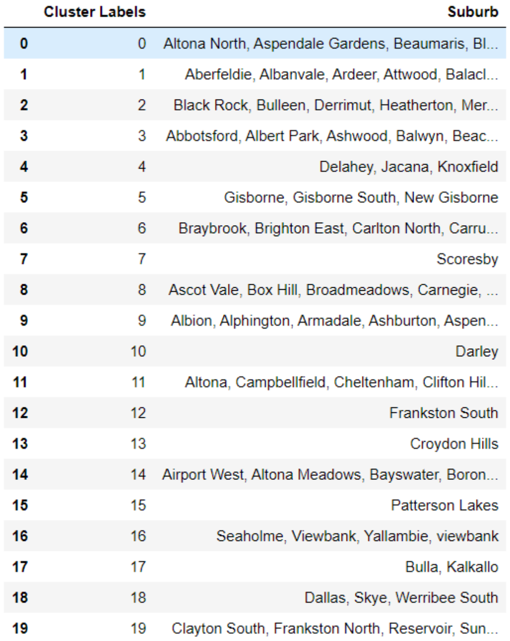
Now we have obtained the required dataset where we have the cluster to which each suburb belongs to and the average price of the suburb. Figures 5,6 and 7 show the average price of each cluster, the suburbs in each cluster and the clusters arranged in descending order showing the cluster with the highest price.

Figure 6: Average price of a cluster



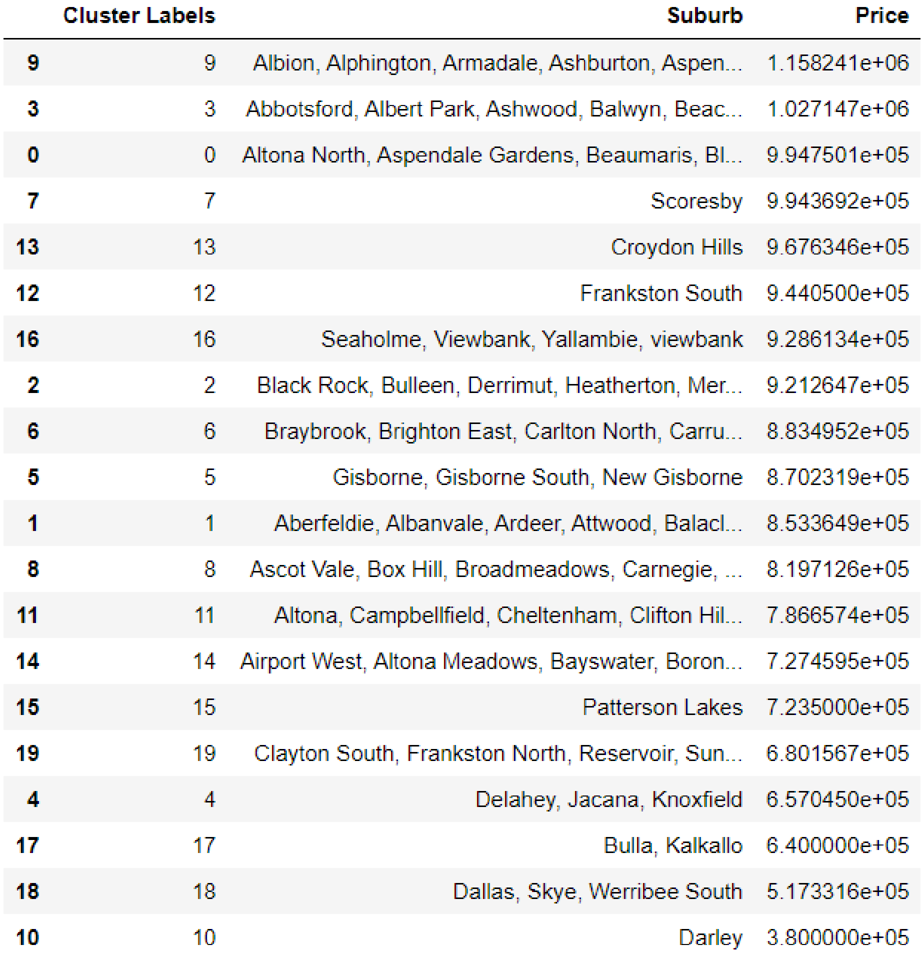
	Cluster Labels	Price
0	0	9.947501e+05
1	1	8.533649e+05
2	2	9.212647e+05
3	3	1.027147e+06
4	4	6.570450e+05
5	5	8.702319e+05
6	6	8.834952e+05
7	7	9.943692e+05
8	8	8.197126e+05
9	9	1.158241e+06
10	10	3.800000e+05
11	11	7.866574e+05
12	12	9.440500e+05
13	13	9.676346e+05
14	14	7.274595e+05
15	15	7.235000e+05
16	16	9.286134e+05
17	17	6.400000e+05
18	18	5.173316e+05
19	19	6.801567e+05

Figure 7: Suburbs in a cluster



	Cluster Labels	Suburb
0	0	Altona North, Aspendale Gardens, Beaumaris, Bl...
1	1	Aberfeldie, Albanvale, Ardeer, Attwood, Balacl...
2	2	Black Rock, Bulleen, Derrimut, Heatherton, Mer...
3	3	Abbotsford, Albert Park, Ashwood, Balwyn, Beac...
4	4	Delahey, Jacana, Knoxfield
5	5	Gisborne, Gisborne South, New Gisborne
6	6	Braybrook, Brighton East, Carlton North, Carru...
7	7	Scoresby
8	8	Ascot Vale, Box Hill, Broadmeadows, Carnegie, ...
9	9	Albion, Alphington, Armadale, Ashburton, Aspen...
10	10	Darley
11	11	Altona, Campbellfield, Cheltenham, Clifton Hil...
12	12	Frankston South
13	13	Croydon Hills
14	14	Airport West, Altona Meadows, Bayswater, Boron...
15	15	Patterson Lakes
16	16	Seaholme, Viewbank, Yallambie, viewbank
17	17	Bulla, Kalkallo
18	18	Dallas, Skye, Werribee South
19	19	Clayton South, Frankston North, Reservoir, Sun...

Figure 8: Clusters arranged in descending order with respect to price



	Cluster Labels	Suburb	Price
9	9	Albion, Alphington, Armadale, Ashburton, Aspen...	1.158241e+06
3	3	Abbotsford, Albert Park, Ashwood, Balwyn, Beac...	1.027147e+06
0	0	Altona North, Aspendale Gardens, Beaumaris, Bl...	9.947501e+05
7	7	Scoresby	9.943692e+05
13	13	Croydon Hills	9.676346e+05
12	12	Frankston South	9.440500e+05
16	16	Seaholme, Viewbank, Yallambie, viewbank	9.286134e+05
2	2	Black Rock, Bulleen, Derrimut, Heatherton, Mer...	9.212647e+05
6	6	Braybrook, Brighton East, Carlton North, Carru...	8.834952e+05
5	5	Gisborne, Gisborne South, New Gisborne	8.702319e+05
1	1	Aberfeldie, Albanvale, Ardeer, Attwood, Balacl...	8.533649e+05
8	8	Ascot Vale, Box Hill, Broadmeadows, Carnegie, ...	8.197126e+05
11	11	Altona, Campbellfield, Cheltenham, Clifton Hil...	7.866574e+05
14	14	Airport West, Altona Meadows, Bayswater, Boron...	7.274595e+05
15	15	Patterson Lakes	7.235000e+05
19	19	Clayton South, Frankston North, Reservoir, Sun...	6.801567e+05
4	4	Delahey, Jacana, Knoxfield	6.570450e+05
17	17	Bulla, Kalkallo	6.400000e+05
18	18	Dallas, Skye, Werribee South	5.173316e+05
10	10	Darley	3.800000e+05

Results and Discussion

Figure 9 shows a scatter plot presenting the variation in price within the cluster. Figure 10 shows a folium plot visualizing the price of different suburbs and showing markers of different suburbs while each marker is color coded depending on which cluster they below to. Note that the price of a suburb has been kept independent from the KMeans Clustering. Price was only introduced after clustering was completed.

Figure 9: Variation in price within a cluster

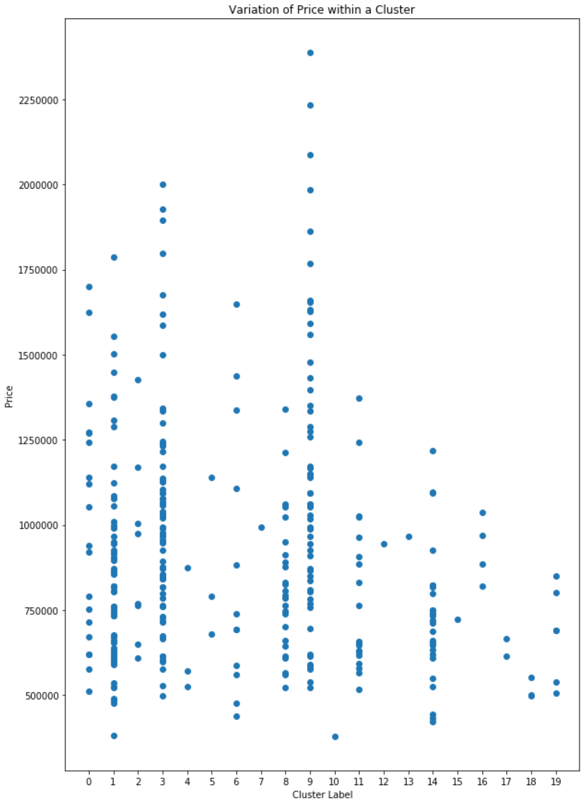


Figure 10: Folium plot visualizing the price of different suburbs. The pop up label shows suburb name, cluster and average suburb of the suburb

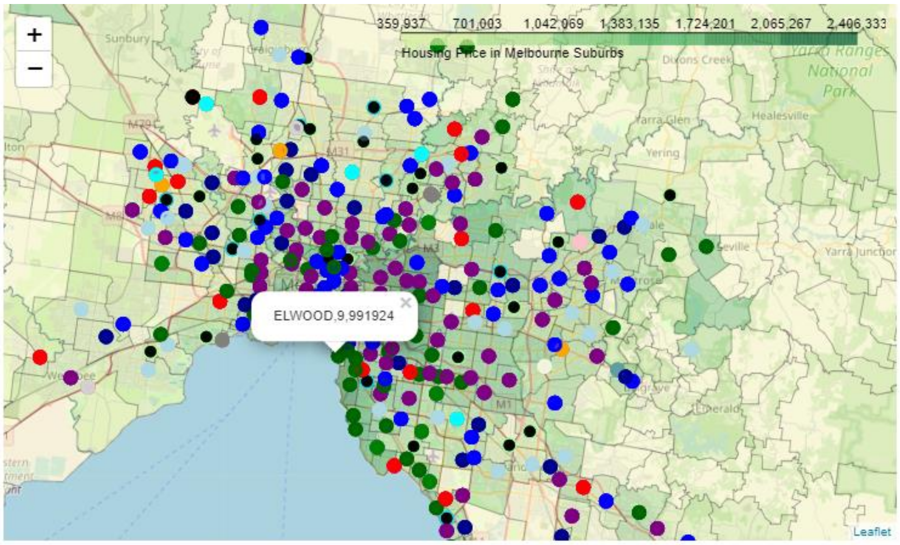
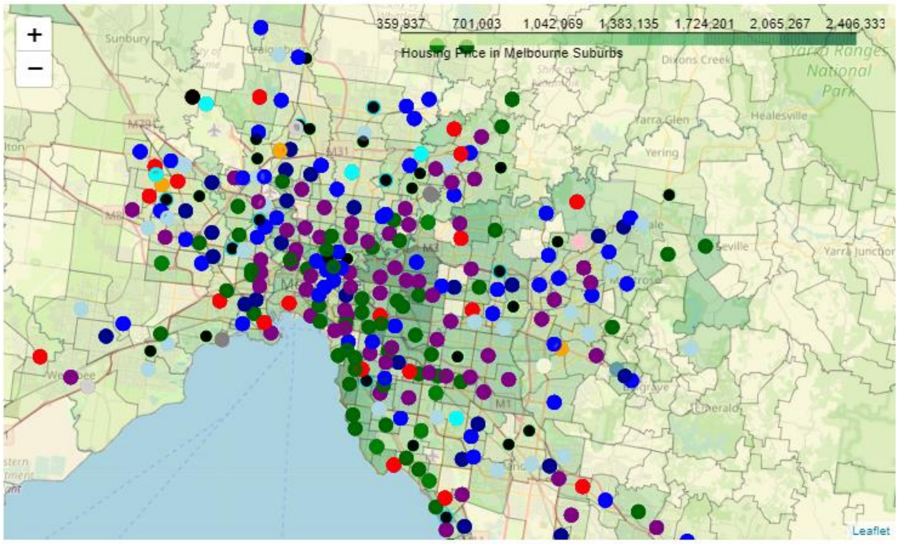


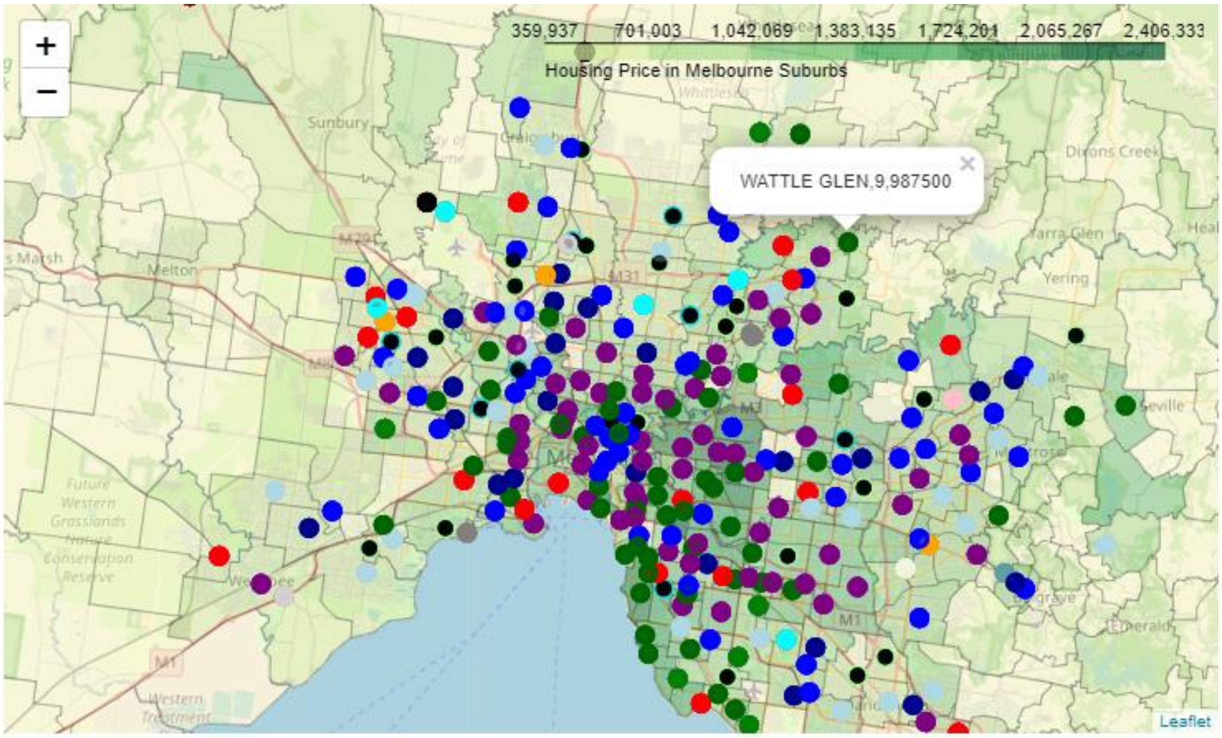
Figure 11: Folium plot presenting the clusters of suburbs



As presented in Figure 8, the clusters with highest average housing prices are Clusters 9 and 3. We see that the suburbs closer to central melbourne are predominantly made of these 3 clusters. Hence as expected, closer to the inner-city regions, the price of these suburbs increase. Given that these clusters were processed independently of the price and used the frequency of venue categories, we can conclude that expensive suburbs tend to have similar facilities in their vicinity.

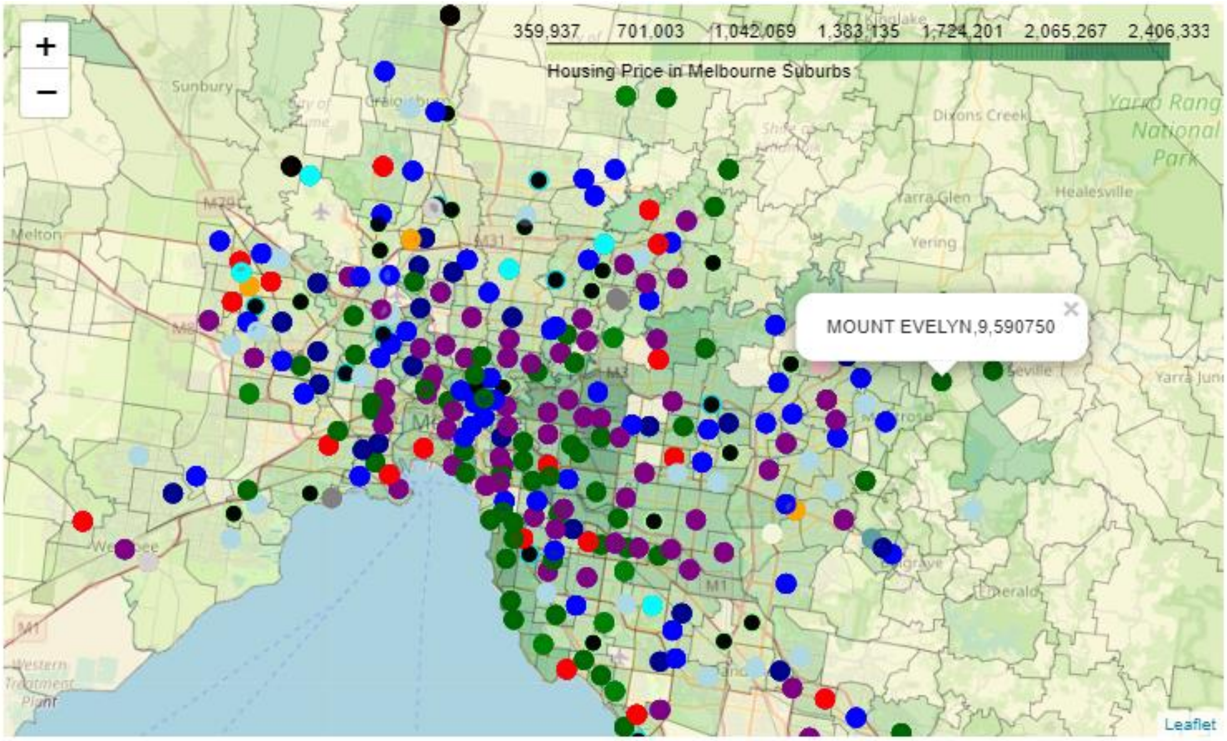
Buyers who want to be away from the inner city, yet have similar facilities may select suburbs within the cluster but away from the inner city as shown in Figure 12.

Figure 12: A suburb with a similar price but away from the inner city



Furthermore, as shown in the scatter plot Variation of Price within a Cluster, the price of a house may vary within these clusters itself. This presents an opportunity for buyers as they may opt to go to suburbs with similar facilities in the vicinity, while paying a lower price for a house as shown in Figure 13.

Figure 13: Lower priced housing within the same cluster



Conclusion

This project has accomplished the goals that were set out at the start of the project which was to:

a) Find clusters of suburbs based on the venue categories in their vicinity
b) Combine them with the average price of those suburbs
c) Present home buyers more flexible options where they may either:
   1) Buy homes away from the inner city, yet have similar facilities in their vicinity
   2) Buy homes in suburbs with similar facilities to that of inner city suburbs while paying a lower price for the house.