Data Exploration Report

The housing market in Melbourne

Student Name: Tang-wei, Hung

Student ID: 29375932

Activity Number: 09

Tutor Name: Jie (Lewis) Liu

Table of Contents

INTRODUCTION

Melbourne, Australia's second-biggest city and world's most liveable city, attracts tourists, students, immigrant, and investors such as real estate investors etc. from all over the world. Therefore, a lot of people want to live and settle down in Melbourne and it will cause the booming of market for real estate in Melbourne.

In this report, I will try to explore the trend of the housing market in Melbourne and to find the relationship between features and the home prices in the Melbourne housing market from 2016 to 2018. Here are three main parts we will focus on. The first part is the suburb vs. price. The second is the price trend in Melbourne housing market from 2016 to 2018. The final part is whether the distance to CBD, distance to nearest train station, and average travel time to CBD (southern cross station) will affect the price of real estate.

DATA WRANGLING

In the following blocks, I will describe the data sources with links and describe the steps in data wrangling with data cleaning and data transformation. Moreover, the tools I used to perform the data wrangling is Python.

Data Sources:

In this report, I use four data sources to complete this project. Here are these four data sources.

1. Tabular data: Melbourne Housing Data from 2016 to 2018 (34858 rows x 21columns)
   (URL: https://www.kaggle.com/anthonypino/melbourne-housing-market)

2. Tabular data: Victoria crime incident data from 2009 to 2018(284098 rows x 7 columns)
   (URL: https://www.crimestatistics.vic.gov.au/crime-statistics/historical-crime-data/year-ending-31-december-2018/download-data)

3. Tabular data: PTV timetable and Geographic Information
   This dataset provides static timetable data and geographic information in the GTFS (General Transit Feed Specification) format.
   (URL: https://discover.data.vic.gov.au/dataset/ptv-timetable-and-geographic-information-2015-gtfs)

4. Spatial data: Victoria State Boundary (shapefile)
   This dataset provides boundary of Victoria
   (URL: https://data.gov.au/data/dataset/vic-suburb-locality-boundaries-psma-administrative-boundaries/resource/4d6ec8bb-1039-4fef-aa58-6a14438f29b1)

Steps of data wrangling:

1. Check Null values in Melbourne Housing data, drop the columns of BuildingArea, YearBuilt, droom2, Bathroom, Car, and Landsize, and then drop the null data

```
Melbourne_housing_df.head(5)
```

	Suburb	Address	Rooms	Type	Price	Method	SellerG	Date	Distance	Postcode	...	Bathroom	Car	Landsize	BuildingArea	YearBuilt	Cc
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	

5 rows × 21 columns

```
Melbourne_housing_df.isna().sum()
```

```
Suburb              0
Address             0
Rooms               0
Type                0
Price            7610
Method              0
SellerG             0
Date                0
Distance            1
Postcode            1
Bedroom2         8217
Bathroom         8226
Car              8728
Landsize        11810
BuildingArea    21115
YearBuilt       19306
CouncilArea         3
Lattitude        7976
Longtitude       7976
Regionname          3
Propertycount       3
dtype: int64
```

```
df= Melbourne_housing_df.drop(columns=['BuildingArea','YearBuilt','Bedroom2','Bathroom','Car','Landsize'])
```

```
df=df.dropna()
```

```
df.head(10)
```

	Suburb	Address	Rooms	Type	Price	Method	SellerG	Date	Distance	Postcode	Landsize	CouncilArea	Lattitude	Longtitude	Regionnan
1	Abbotsford	85 Turner St	2	h	1480000.0	S	Biggin	3/12/2016	2.5	3067.0	202.0	Yarra City Council	-37.7996	144.9984	Northe Metropolit
2	Abbotsford	25 Bloomburg St	2	h	1035000.0	S	Biggin	4/02/2016	2.5	3067.0	156.0	Yarra City Council	-37.8079	144.9934	Northe Metropolit
4	Abbotsford	5 Charles St	3	h	1465000.0	SP	Biggin	4/03/2017	2.5	3067.0	134.0	Yarra City Council	-37.8093	144.9944	Northe Metropolit
5	Abbotsford	40 Federation La	3	h	850000.0	PI	Biggin	4/03/2017	2.5	3067.0	94.0	Yarra City Council	-37.7969	144.9969	Northe Metropolit
6	Abbotsford	55a Park St	4	h	1600000.0	VB	Nelson	4/06/2016	2.5	3067.0	120.0	Yarra City Council	-37.8072	144.9941	Northe Metropolit

2. Open Table 07 sheet in Victoria crime incident data, and check the Null value

```
crime_df= pd.read_excel('Data_tables_Criminal_Incidents_Visualisation_year_ending_December_2018.xlsx'
                        ,sheet_name='Table 07')
```

```
crime_df.head(5)
```

	Year ending December	Postcode	Suburb/Town Name	Offence Division	Offence Subdivision	Offence Subgroup	Incidents Recorded
0	2009	3000	MELBOURNE	A Crimes against the person	A20 Assault and related offences	A232 Non-FV Common assault	407
1	2009	3000	MELBOURNE	A Crimes against the person	A20 Assault and related offences	A231 FV Common assault	26
2	2009	3000	MELBOURNE	A Crimes against the person	A20 Assault and related offences	A212 Non-FV Serious assault	618
3	2009	3000	MELBOURNE	A Crimes against the person	A20 Assault and related offences	A211 FV Serious assault	25
4	2009	3000	MELBOURNE	A Crimes against the person	A20 Assault and related offences	A22 Assault police, emergency services or othe...	182

```
crime_df.isna().sum()
```

```
Year ending December    0
Postcode                0
Suburb/Town Name        0
Offence Division        0
Offence Subdivision     0
Offence Subgroup        0
Incidents Recorded      0
dtype: int64
```

3. Aggregate the data by using columns of Year ending December and Postcode in Victoria crime incident data.

```
crime_df=crime_df.rename(columns={'Year ending December':'Year'})

crime_groupby=crime_df.groupby(['Year','Postcode'], as_index=False)["Incidents Recorded"].sum()
```

```
crime_groupby.head(5)
```

	Year	Postcode	Incidents Recorded
0	2009	3000	17615
1	2009	3002	871
2	2009	3003	429
3	2009	3006	1369
4	2009	3008	507

4. Join the Victoria crime incident and Melbourne housing data together by merging on columns of Year and Postcode.

```
final_df = pd.merge(df,crime_groupby, on=['Year','Postcode'])
```

```
final_df.head(5)
```

Type	Price	Method	SellerG	Date	Distance	Postcode	CouncilArea	Lattitude	Longtitude	Regionname	Propertycount	Year	Month	Weekday	Incidents Recorded
h	1480000	S	Biggin	2016-12-03	2.5	3067	Yarra City Council	-37.7996	144.9984	Northern Metropolitan	4019	2016	12	5	994
h	1035000	S	Biggin	2016-02-04	2.5	3067	Yarra City Council	-37.8079	144.9934	Northern Metropolitan	4019	2016	2	3	994
h	1600000	VB	Nelson	2016-06-04	2.5	3067	Yarra City Council	-37.8072	144.9941	Northern Metropolitan	4019	2016	6	5	994
h	941000	S	Jellis	2016-05-07	2.5	3067	Yarra City Council	-37.8041	144.9953	Northern Metropolitan	4019	2016	5	5	994
h	1876000	S	Nelson	2016-05-07	2.5	3067	Yarra City Council	-37.8024	144.9993	Northern Metropolitan	4019	2016	5	5	994

5. Read the PTV timetable and Geographic Information in GTFS form in Python and get the station information and public transport schedules.

```
#Extract the gtfs file
zip_gtfs = ZipFile('gtfs.zip')
zip_gtfs.extractall()
```

```
sched = pygtfs.Schedule(":memory:")
# append data to schedule object
pygtfs.append_feed(sched, "./1/google_transit.zip")
pygtfs.append_feed(sched, "./2/google_transit.zip")
```

```
Loading GTFS data for <class 'pygtfs.gtfs_entities.Agency'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Stop'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Route'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Trip'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.StopTime'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Service'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.ServiceException'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Fare'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.FareRule'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.ShapePoint'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Frequency'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Transfer'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.FeedInfo'>:
Loading GTFS data for <class 'pygtfs.gtfs_entities.Translation'>:
1 record read for <class 'pygtfs.gtfs_entities.Agency'>.
109 records read for <class 'pygtfs.gtfs_entities.Stop'>.
188 records read for <class 'pygtfs.gtfs_entities.Route'>.
.5869 records read for <class 'pygtfs.gtfs_entities.Trip'>.
.............72489 records read for <class 'pygtfs.gtfs_entities.StopTime'>.
40 records read for <class 'pygtfs.gtfs_entities.Service'>.
5 records read for <class 'pygtfs.gtfs_entities.ServiceException'>.
.......................................................................................................................
.............................735691 records read for <class 'pygtfs.gtfs_entities.ShapePoint'>.
Complete.
```

6. Calculate the direct distance from the house to the closest train station and the average travel time from the house's closest train station to Southern Cross Railway Station.

```
final_df['train_station_id']=final_df.apply(lambda x: find_station_id(x.Lattitude, x.Longtitude),axis=1)
final_df['distance_to_train_station']=final_df.apply(lambda x: find_station_distance(x.Lattitude, x.Longtitude),axi
final_df['travel_min_to_CBD']=final_df.apply(lambda x: average_min(x.train_station_id),axis=1)
final_df['train_station_name']=final_df.apply(lambda x: find_station_name(x.Lattitude, x.Longtitude),axis=1)
```

```
final_df.head(5)
```

Postcode	...	Regionname	Propertycount	Year	Month	Weekday	Incidents Recorded	train_station_id	distance_to_train_station	travel_min_to_CBD	train_station_name
3067	...	Northern Metropolitan	4019	2016	12	5	994	19975	350.781	13.235294	Victoria Park Railway Station (Abbotsford)
3067	...	Northern Metropolitan	4019	2016	2	3	994	19977	289.148	10.125000	North Richmond Railway Station (Richmond)
3067	...	Northern Metropolitan	4019	2016	6	5	994	19976	299.261	12.235294	Collingwood Railway Station (Abbotsford)
3067	...	Northern Metropolitan	4019	2016	5	5	994	19976	144.364	12.235294	Collingwood Railway Station (Abbotsford)
3067	...	Northern Metropolitan	4019	2016	5	5	994	19976	542.509	12.235294	Collingwood Railway Station (Abbotsford)

DATA CHECKING

In this section, I will try to find whether errors and outliers in this data set and then correct them.

Errors:

In this part, I use Python to check if there are any errors in this data set.

```
df.describe().T
```

	count	mean	std	min	25%	50%	75%	max
Rooms	20993.0	3.059163e+00	0.949881	1.00000	2.0000	3.00000	4.000000e+00	1.600000e+01
Price	20993.0	1.089746e+06	653028.263712	85000.00000	657000.0000	910000.00000	1.335000e+06	1.120000e+07
Distance	20993.0	1.135902e+01	6.891418	0.00000	6.4000	10.40000	1.420000e+01	4.810000e+01
Postcode	20993.0	3.114631e+03	114.810599	3000.00000	3046.0000	3087.00000	3.152000e+03	3.978000e+03
Lattitude	20993.0	-3.780696e+01	0.091619	-38.19043	-37.8609	-37.80046	-3.774897e+01	-3.739780e+01
Longtitude	20993.0	1.449967e+02	0.120680	144.42379	144.9253	145.00320	1.450688e+02	1.455264e+02
Propertycount	20993.0	7.516751e+03	4411.397778	83.00000	4380.0000	6567.00000	1.033100e+04	2.165000e+04
Year	20993.0	2.016818e+03	0.627999	2016.00000	2016.0000	2017.00000	2.017000e+03	2.018000e+03
Month	20993.0	7.135283e+00	3.066354	1.00000	5.0000	7.00000	1.000000e+01	1.200000e+01
Weekday	20993.0	4.889344e+00	0.933629	0.00000	5.0000	5.00000	5.000000e+00	6.000000e+00
Incidents Recorded	20993.0	1.740383e+03	1460.434184	33.00000	812.0000	1295.00000	2.243000e+03	1.765300e+04
train_station_id	20993.0	2.104833e+04	5143.796725	15351.00000	19917.0000	19962.00000	2.001700e+04	5.216100e+04
distance_to_train_station	20993.0	1.460002e+03	1164.891911	22.91800	640.1250	1105.31900	1.923964e+03	1.984636e+04
travel_min_to_CBD	20993.0	2.849908e+01	11.909687	0.00000	20.0000	28.50000	3.515385e+01	7.560000e+01

According to the above graph, we could notice that it may have input error in the room column. This is because it is very hard to have 16 rooms in one house. Therefore, we could consider this is an input error, and then delete this row.

Outliers:

In this part, I use Python to find the outliers which are over 5 standard deviation in Price column.

```
df['outlier_price'] = 0

price_mean = df['Price'].mean()
price_std = df['Price'].std()

df['outlier_price'] = np.where(abs(df['Price'] - price_mean) > 5 * price_std, 1, 0)
```

```
df[df['outlier_price']==0]
```

Postcode	...	Propertycount	Year	Month	Weekday	Incidents Recorded	train_station_id	distance_to_train_station	travel_min_to_CBD	train_station_name	outlier_price
3067	...	4019	2016	12	5	994	19975	350.781	13.235294	Victoria Park Railway Station (Abbotsford)	0
3067	...	4019	2016	2	3	994	19977	289.148	10.125000	North Richmond Railway Station (Richmond)	0
3067	...	4019	2016	6	5	994	19976	299.261	12.235294	Collingwood Railway Station (Abbotsford)	0
3067	...	4019	2016	5	5	994	19976	144.364	12.235294	Collingwood Railway Station (Abbotsford)	0
3067	...	4019	2016	5	5	994	19976	542.509	12.235294	Collingwood Railway Station (Abbotsford)	0
...	...	...	...	...	...	...	...	...	...	...	...
3197	...	3351	2018	3	5	726	19859	1883.229	58.583333	Bonbeach Railway Station (Bonbeach)	0
3335	...	538	2018	3	5	623	19981	1358.710	34.333333	Rockbank Railway Station (Rockbank)	0

```
final=df[df['outlier_price']==0]
final= final.drop(columns=['outlier_price'])
```

```
final.to_csv("housing_final.csv",index=False)
```

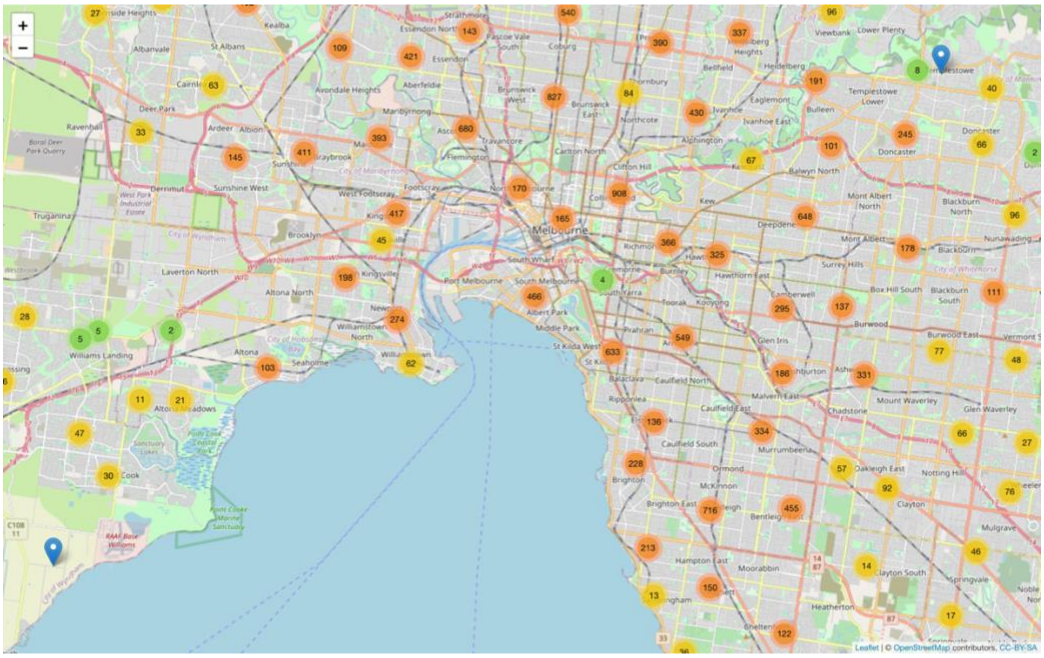
DATA EXPLORATION

In data exploration part, there are three main parts we will focus on. The first part is the suburb vs. price. The second is the price trend in Melbourne housing market from 2016 to 2018. The final part is whether the distance to CBD, distance to nearest train station, and average travel time to CBD (southern cross station) will affect the price of real estate. We will use R and Tableau to complete this task.

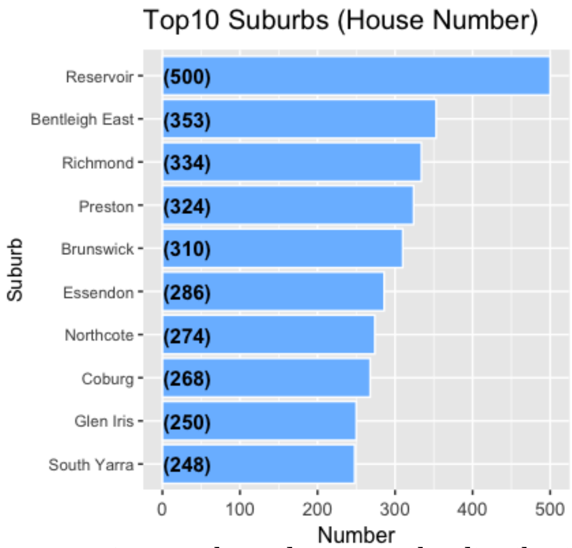
Suburb Vs Price:

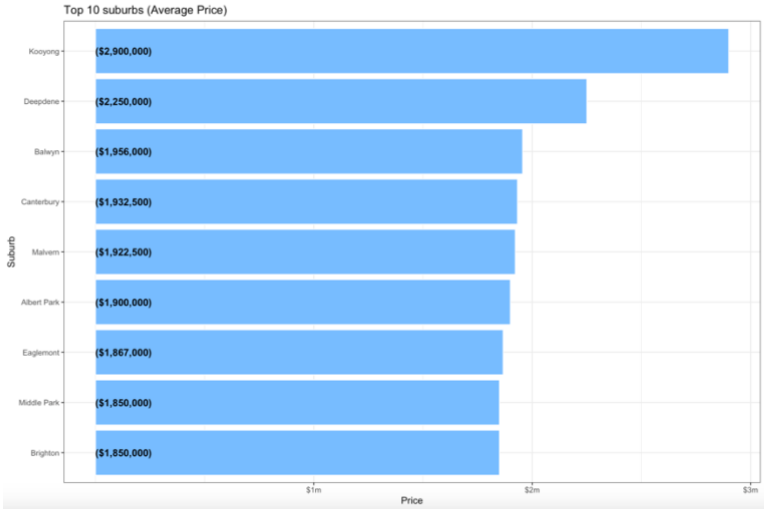
In this part, I want to know which suburb has higher real estate price and what is the distribution of price in the top 10 expensive suburbs. Also, I want to know whether the crime rate will affect the purchase intention of people to buy the house and affect the house price. The tools I use in this part are R and Tableau.

1. To plot the distribution of the house on the map in R, and we can notice that the houses near Collingwood, St Kilda, and Bentleigh have a higher number of sales.

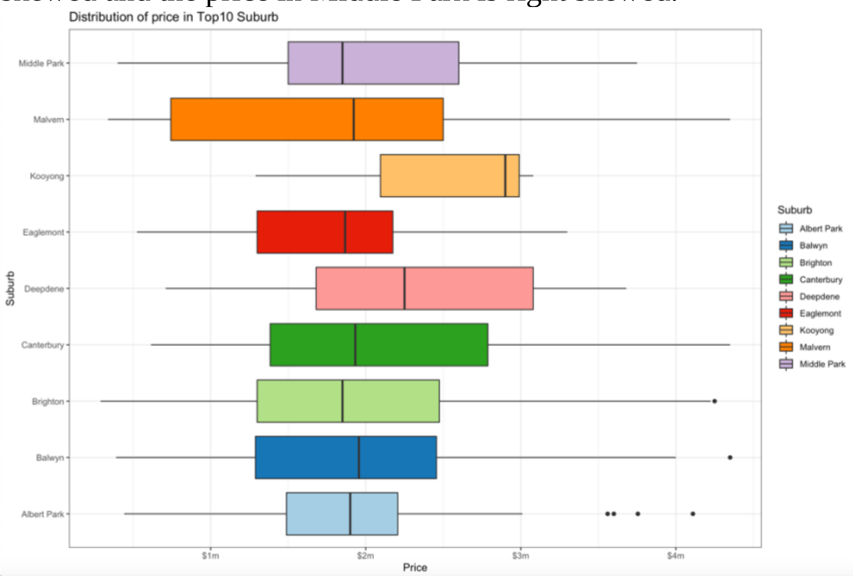


2. Use aggregate function to find the houses sell number and the average price in each suburb and find the Top 10 number of houses sell suburbs and average price suburbs, in R. We can notice the Reservoir has highest number of houses sell and Kooyong has highest average price of $2,900,000.



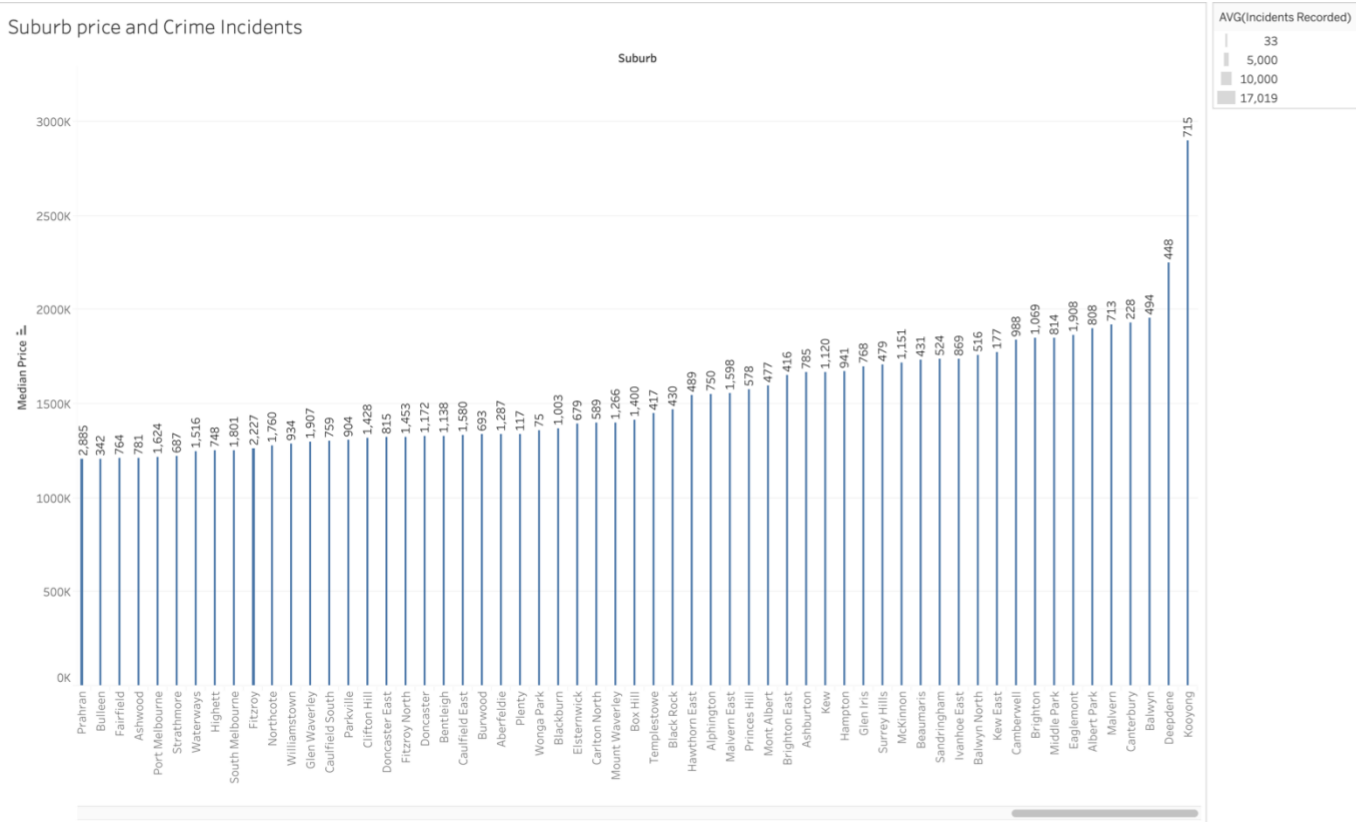


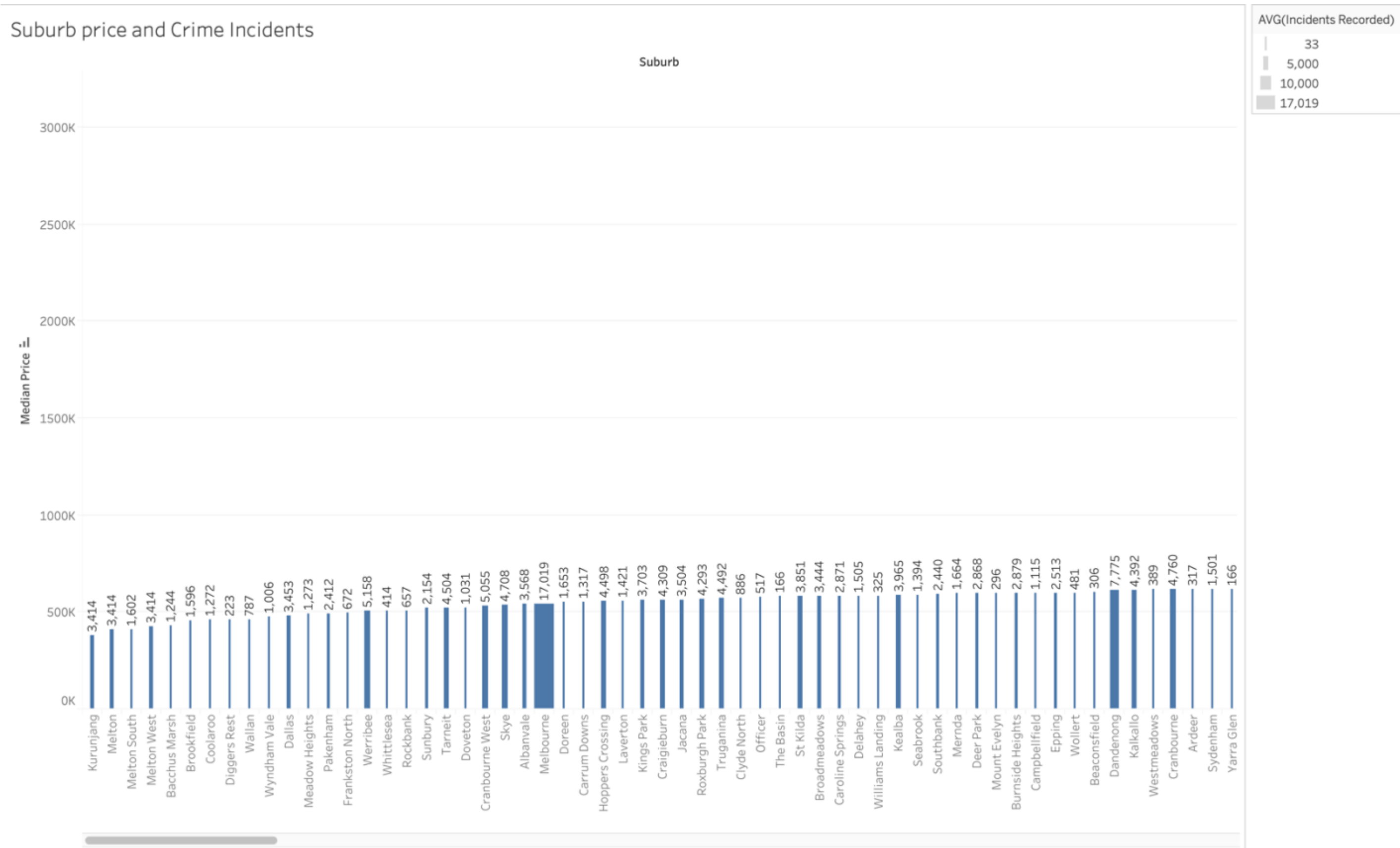
3. Use box plot to see the distribution of the house price in Top 10 suburb (average price). We could notice that the distribution of price in Kooyong and Malvem is left skewed and the price in Middle Park is right skewed.



4. Use choropleth map to plot the result of suburb and average price in Melbourne in R.

5. Use Tableau to see the relationship between suburb, price, and crime incidents number.
   According to these plots, we could notice that the higher average house price suburbs have lower number of crime incidents. For example, the Kooyong which has only average 715 crime incidents from 2016 to 2018. However, the lower average house price suburbs tend to have higher number of crime incidents.

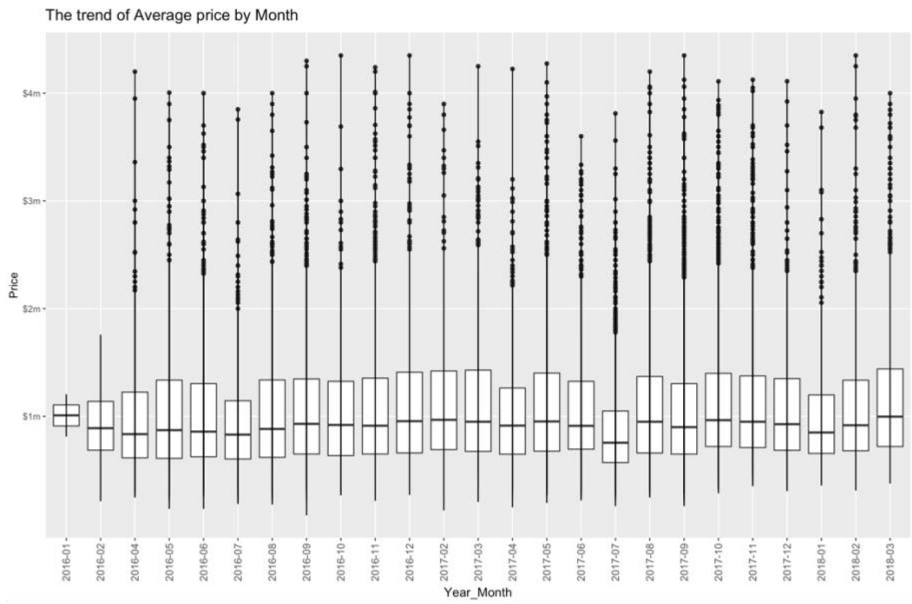




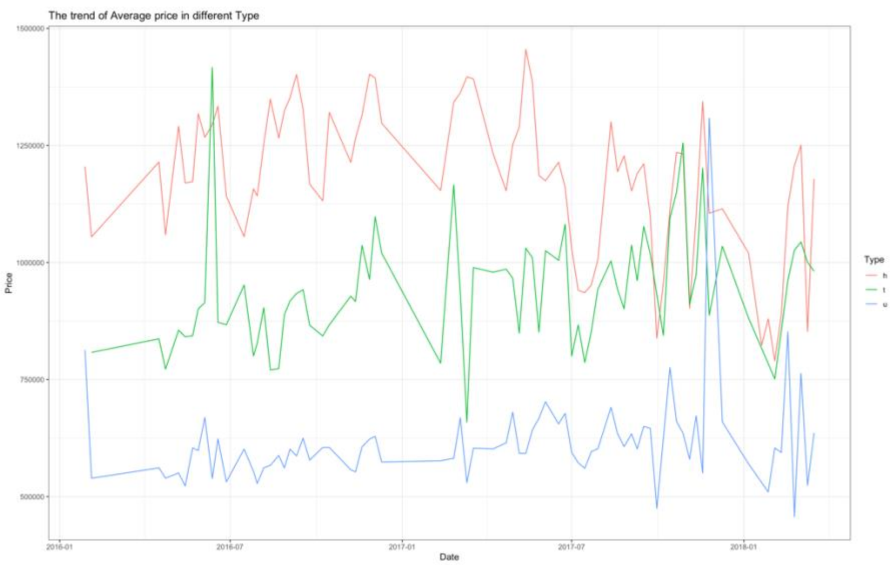
Price Trend from 2016 -2018:

In this section, I want to know the price trend from 2016 to 2018 in Melbourne housing market. Also, I want to know the price trend for different type of house from 2016 to 2018.

1. Use the boxplot to see the price trend from 2016 to 2018 in R. According to the graph, we can notice that the average price from 2016 to 2018 are not increase or decrease sharply. Although the average price has slight drop and slight raise, the overall trend for average price is remain constant around $1,000,000.



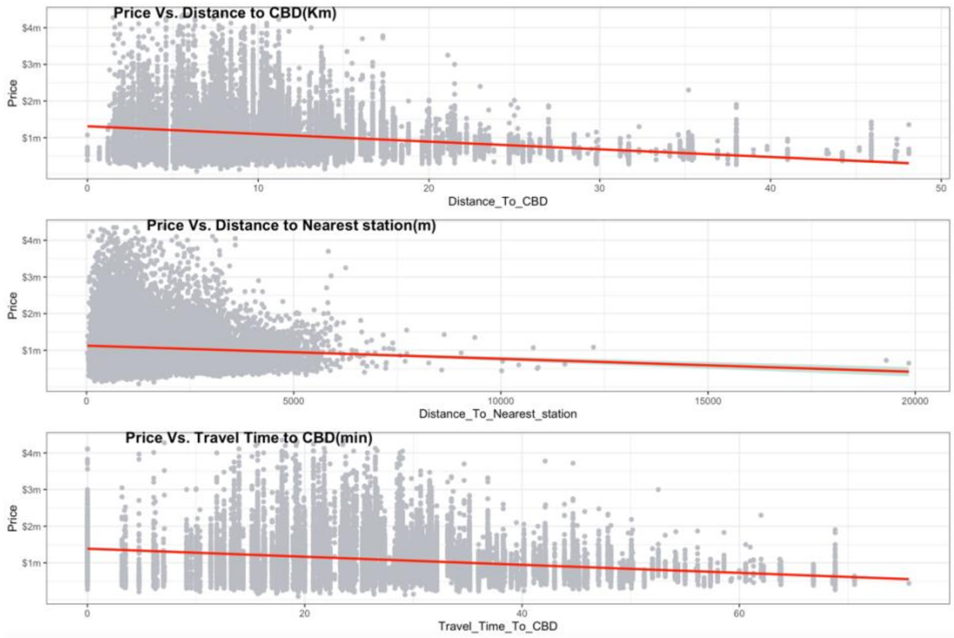
2. Aggregate by the types and see price trend for different type of house from 2016 to 2018 in R.
   The line chart reflects several trends. These three types have similar price trends from 2016 to 2018 in Melbourne housing market. In addition, the average price of type h is higher than type u and type t. Finally, the average price of type t in 2018/12 has a large increase to the price level of type h.
   (h – house, cottage, villa, semi, terrace / u – unit / t – townhouse)



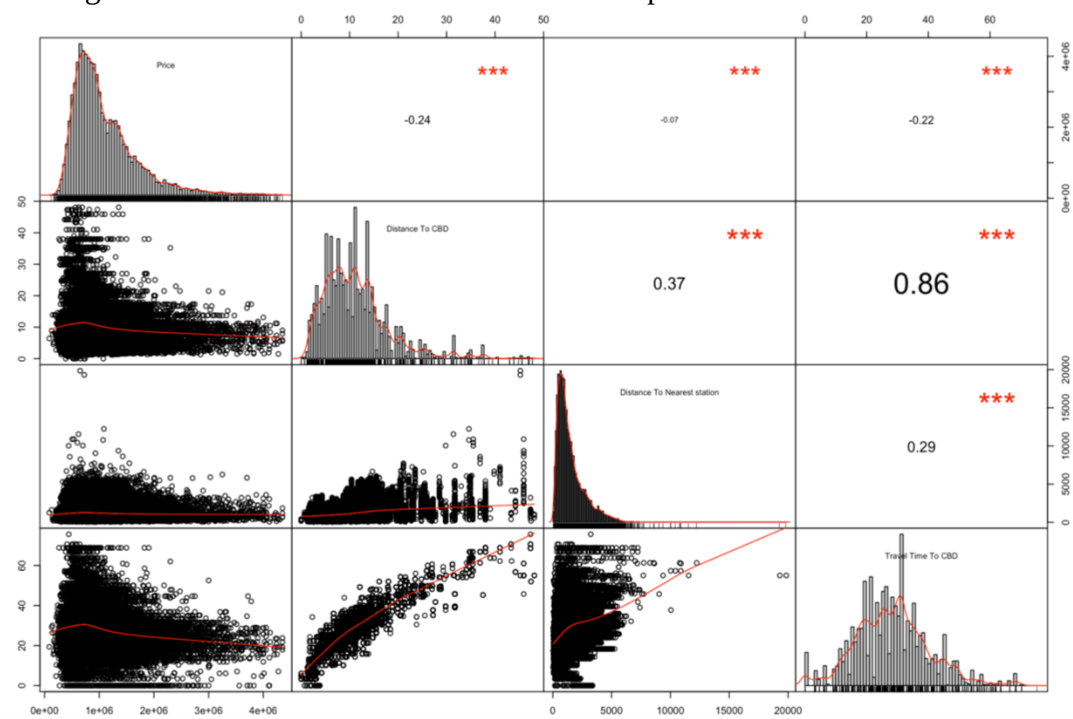
Distance/ Average time to CBD Vs. Price:

In this part, we wonder to know whether the distance to CBD, average time to CBD, and distance to nearest train station will affect the price of real estate.

1. Add a linear trend to a scatterplot of price and distance to CBD, price and average time to CBD, and price and distance to nearest train station in R.
   According to following graph, we could notice that all these three factors (distance to CBD, average time to CBD, and distance to nearest train station) are have small negative relationship with price.



2. Display a chart of a correlation matrix of Price, distance to CBD, average time to CBD, and distance to nearest train station in R.
   According to following matrix, we could find the distributions of price, distance to CBD, and distance to nearest station are right skew, and the distribution of average travel time to CBD is like normal distribution. In addition, all these three factors have negative value of the correlation and has little p-values.



CONCLUSION

In the first part (suburb vs price), we could notice that the Reservoir has the highest number of houses sell and Kooyong has a highest average price of $2,900,000 from 2016 to 2018 in Melbourne. Besides, we could see the distribution of the house price in Kooyong is left skew. Finally, we use Tableau to see the relationship between suburb, price, and crime incidents and find that it seems to have some relationship. For example, the Kooyong which has the only average of 715 crime incidents from 2016 to 2018. However, the lower average house price suburbs tend to have a higher number of crime incidents.

For the second part (Price trend from 2016 to 2018), we could notice that the average price from 2016 to 2018 does not increase or decrease sharply. Although the average price has a slight drop and a slight rise, the overall trend for the average price is remain constant at around $1000000. Also, if we aggregate the data by type, we could find these three types have similar price trends from 2016 to 2018 in the Melbourne housing market.

For the third part (Distance/Average Time to CBD Vs. price), we could find that all these three factors (distance to CBD, the average time to CBD, and distance to nearest train station) have a small negative relationship with price. Therefore, it might have other important factors that influence the price of the house a lot.

REFLECTION

In this report, it helps me learn how to wrangle the data into the suitable format and to check whether there are any errors such as input errors and outliers in this dataset and then to correct them. In addition, the part of data exploration helps me learn how to display the plot in R, know how to choose a suitable statistical graphics to perform the result, and know how to use common analytics for tabular such as aggregation and ranking.

BIBLIOGRAPHY

1. Melbourne Housing Data from 2016 to 2018 (34858 rows x 21columns) (URL: https://www.kaggle.com/anthonypino/melbourne-housing-market)

2. Victoria crime incident data from 2009 to 2018(284098 rows x 7 columns) (URL: https://www.crimestatistics.vic.gov.au/crime-statistics/historical-crime-data/year-ending-31-december-2018/download-data)

3. PTV timetable and Geographic Information (GTFS format) (URL: https://discover.data.vic.gov.au/dataset/ptv-timetable-and-geographic-information-2015-gtfs)

4. Victoria State Boundary (shapefile) (URL: https://data.gov.au/data/dataset/vic-suburb-locality-boundaries-psma-administrative-boundaries/resource/4d6ec8bb-1039-4fef-aa58-6a14438f29b1)