Holmusk Technical Challenge

ANKIT GUPTA
M. Tech (Computer Science)
Indian Statistical Institute, Kolkata, India

Objective

- With the following dataset, the task is to understand the drivers of prices of the flats.

  Click Here to access the dataset.

- For the flat price prediction by considering the number of rooms

  according to the rules of Singapore Housing and Development Board

  [HDB].

- Used Python Libraries :   Numpy, Pandas, Matplotlib, Seaborn,

  Geopy, Geopandas, and Shapely.

Installation of Python Libraries

- ❖ First, Install all the python libraries in the Jupyter Notebook by using “!pip install Library_Name”. Installation can also be done with conda instead of pip and instead of jupyter notebook, it can also be installed through terminal in Linux.
- ❖ Please check the documentation page of respective python library.

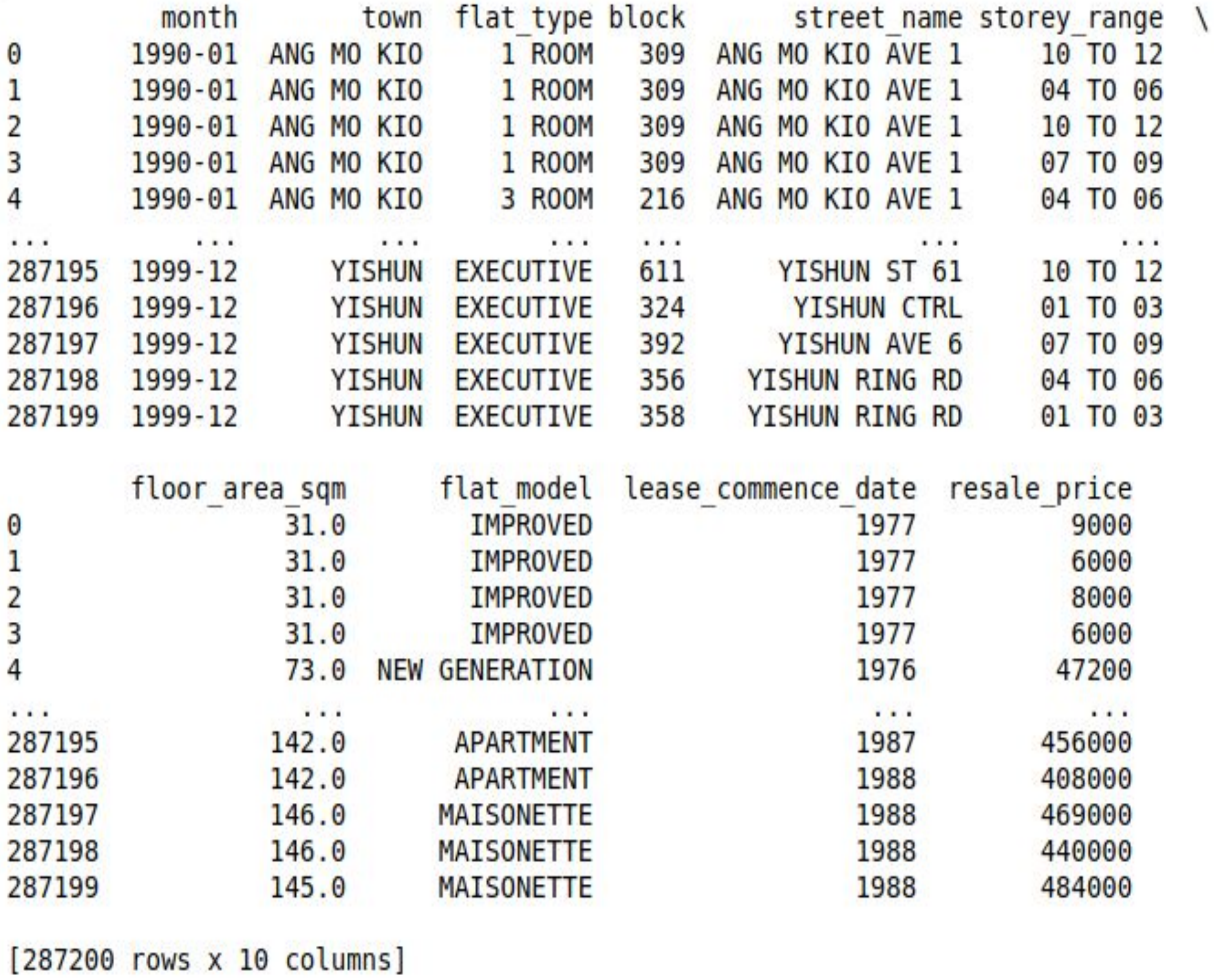
Reading “.csv” file and storing it in a dataframe

❖ By using Pandas’ readcsv() function, read the “.csv” file from the appropriate file path and store it in a dataframe which is “dataset” here.

❖ It has 287200 rows(records) and 10 columns(features).

# print(dataset.shape).

❖ Features are : month , town , flat_type , block , street_name , storey_range , floor_area_sqm , flat_model , lease_commence_date, resale_price.



```
         month        town  flat_type block       street_name storey_range  \
0      1990-01  ANG MO KIO     1 ROOM   309  ANG MO KIO AVE 1     10 TO 12
1      1990-01  ANG MO KIO     1 ROOM   309  ANG MO KIO AVE 1     04 TO 06
2      1990-01  ANG MO KIO     1 ROOM   309  ANG MO KIO AVE 1     10 TO 12
3      1990-01  ANG MO KIO     1 ROOM   309  ANG MO KIO AVE 1     07 TO 09
4      1990-01  ANG MO KIO     3 ROOM   216  ANG MO KIO AVE 1     04 TO 06
...        ...         ...        ...   ...               ...          ...
287195 1999-12      YISHUN  EXECUTIVE   611      YISHUN ST 61     10 TO 12
287196 1999-12      YISHUN  EXECUTIVE   324       YISHUN CTRL     01 TO 03
287197 1999-12      YISHUN  EXECUTIVE   392      YISHUN AVE 6     07 TO 09
287198 1999-12      YISHUN  EXECUTIVE   356    YISHUN RING RD     04 TO 06
287199 1999-12      YISHUN  EXECUTIVE   358    YISHUN RING RD     01 TO 03

        floor_area_sqm      flat_model  lease_commence_date  resale_price
0                 31.0        IMPROVED                 1977          9000
1                 31.0        IMPROVED                 1977          6000
2                 31.0        IMPROVED                 1977          8000
3                 31.0        IMPROVED                 1977          6000
4                 73.0  NEW GENERATION                 1976         47200
...                ...             ...                  ...           ...
287195           142.0       APARTMENT                 1987        456000
287196           142.0       APARTMENT                 1988        408000
287197           146.0      MAISONETTE                 1988        469000
287198           146.0      MAISONETTE                 1988        440000
287199           145.0      MAISONETTE                 1988        484000

[287200 rows x 10 columns]
```

Details about Dataset

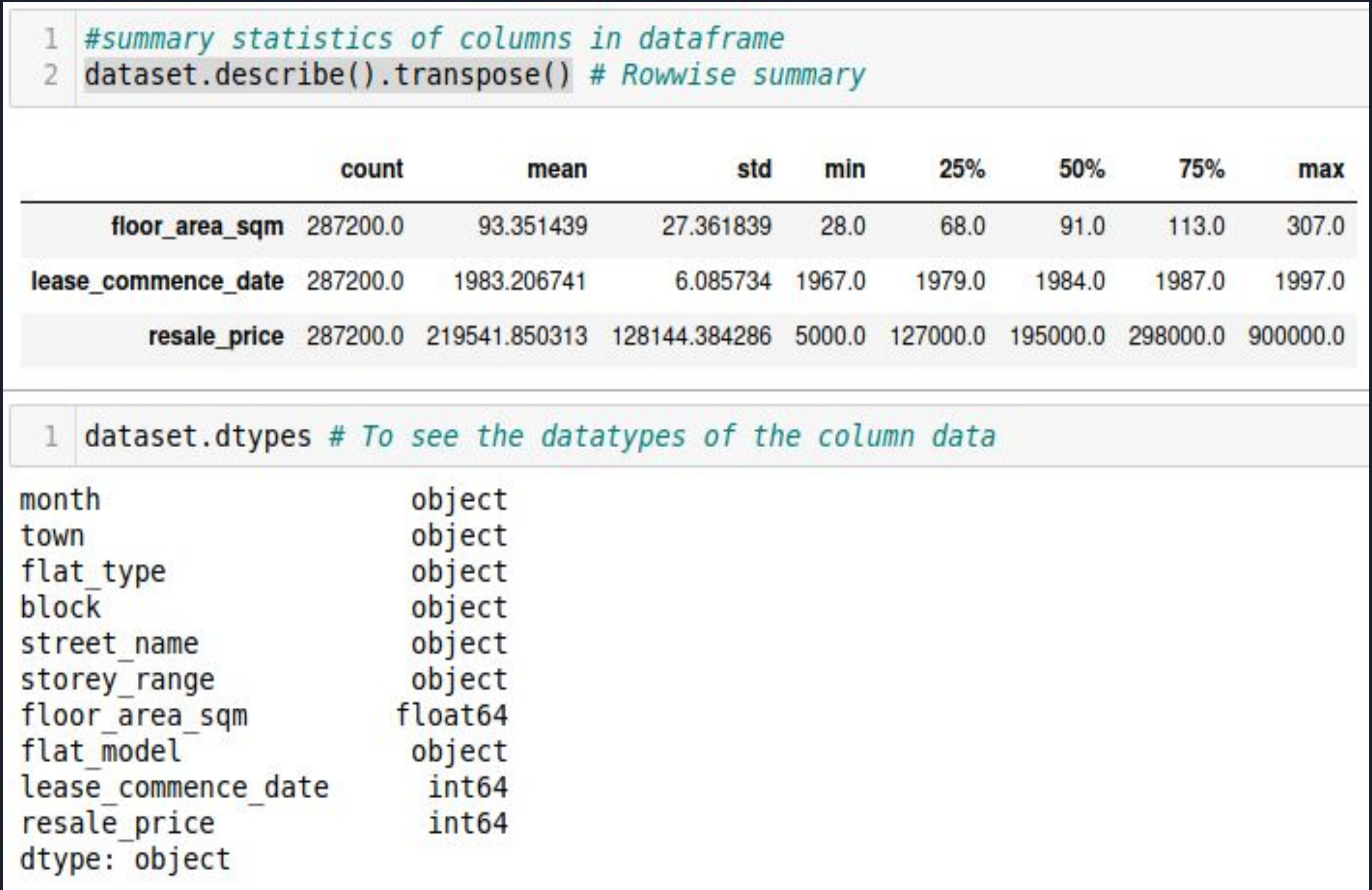
- By using dataframe.column_name.unique(), we can get the unique values of a particular column. I have used it for “flat_type” and “storey_range” column names.

  #dataset.flat_type.unique()

  #dataset.flat_type.unique()

- To get the statistic about the dataset, we can use dataframe.describe() or dataset.describe().transpose().

  #dataset.describe().transpose()



```python
#summary statistics of columns in dataframe
dataset.describe().transpose() # Rowwise summary
```

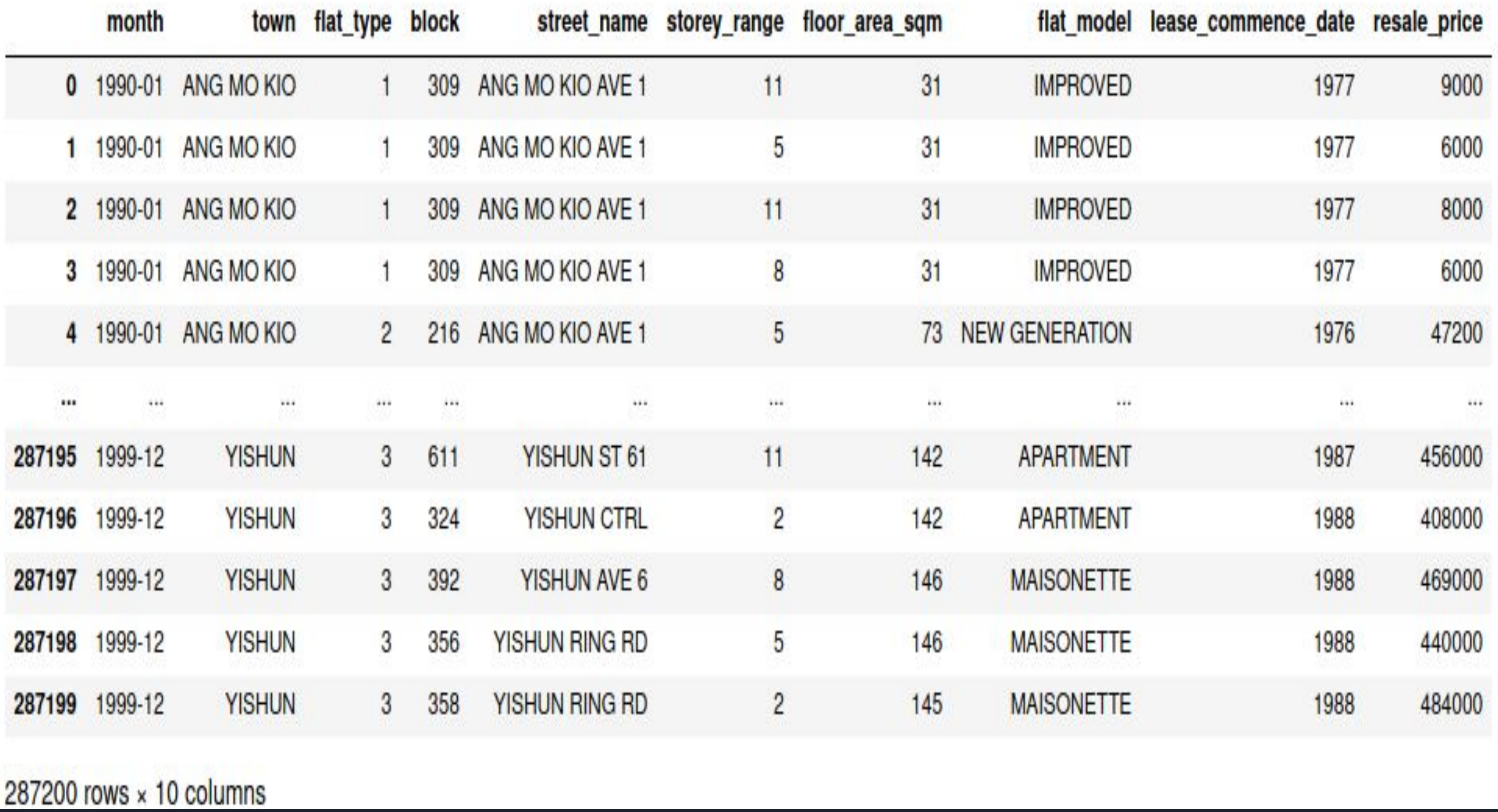
	count	mean	std	min	25%	50%	75%	max
floor_area_sqm	287200.0	93.351439	27.361839	28.0	68.0	91.0	113.0	307.0
lease_commence_date	287200.0	1983.206741	6.085734	1967.0	1979.0	1984.0	1987.0	1997.0
resale_price	287200.0	219541.850313	128144.384286	5000.0	127000.0	195000.0	298000.0	900000.0

```python
dataset.dtypes # To see the datatypes of the column data
```

```
month                   object
town                    object
flat_type               object
block                   object
street_name             object
storey_range            object
floor_area_sqm         float64
flat_model              object
lease_commence_date      int64
resale_price             int64
dtype: object
```

Data Preprocessing

- Any machine learning algorithm (whether it is classification or regression) work on numbers. In the given dataset, “flat_type” and “storey_range” columns have both string and numerical data.
- So convert it into numerical data.
- Based on the rules given by HDB, I have considered 3 bedrooms for “EXECUTIVE” flat_type and 4 bedrooms for “MULTI GENERATION” flat type.



	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease_commence_date	resale_price
0	1990-01	ANG MO KIO	1	309	ANG MO KIO AVE 1	11	31	IMPROVED	1977	9000
1	1990-01	ANG MO KIO	1	309	ANG MO KIO AVE 1	5	31	IMPROVED	1977	6000
2	1990-01	ANG MO KIO	1	309	ANG MO KIO AVE 1	11	31	IMPROVED	1977	8000
3	1990-01	ANG MO KIO	1	309	ANG MO KIO AVE 1	8	31	IMPROVED	1977	6000
4	1990-01	ANG MO KIO	2	216	ANG MO KIO AVE 1	5	73	NEW GENERATION	1976	47200
...	...	...	...	...	...	...	...	...	...	...
287195	1999-12	YISHUN	3	611	YISHUN ST 61	11	142	APARTMENT	1987	456000
287196	1999-12	YISHUN	3	324	YISHUN CTRL	2	142	APARTMENT	1988	408000
287197	1999-12	YISHUN	3	392	YISHUN AVE 6	8	146	MAISONETTE	1988	469000
287198	1999-12	YISHUN	3	356	YISHUN RING RD	5	146	MAISONETTE	1988	440000
287199	1999-12	YISHUN	3	358	YISHUN RING RD	2	145	MAISONETTE	1988	484000

287200 rows × 10 columns

Exploratory Data Analysis

- ❖ Based on the histograms of “flat_type” and “storey_range” and “flat_models” columns, we can say:

  1) 3 and 4 bedroom houses are most commonly sold. So, for a builder having this data , it can make a new flat with more 3 and 4 bedrooms to attract more buyers.

  2) 4 to 6, 7 to 9 , 1 to 3 and 10 to 12 storey_range flats have more count. So, to predict resale flat prices of the flat, we should have to consider these storey range flats.

3) "NEW GENERATION", "IMPROVED" and "MODEL A" flat models have more count compared to other flat models. So, while predicting flat prices, we should have to concentrate these flat models.

4) flats which have resale prices are between 100000$ and 200000$ have highest count which is 80000 and then comes those flats which have resale prices 200000$ and 300000$ with 70000 count and then flats with resale prices between 0 and 100000$ comes with count ~50000.

- ❖ Based on the pairplots between various features, we observe the following key points:

  1) As we can see, there is a spike in the scatter plot between floor_area_sqm and lease commence_date. For lease_commence year between 1960 and 1980, for 2-ROOM flat model, floor area is very high and this 2-ROOM flat model is used between these years only.

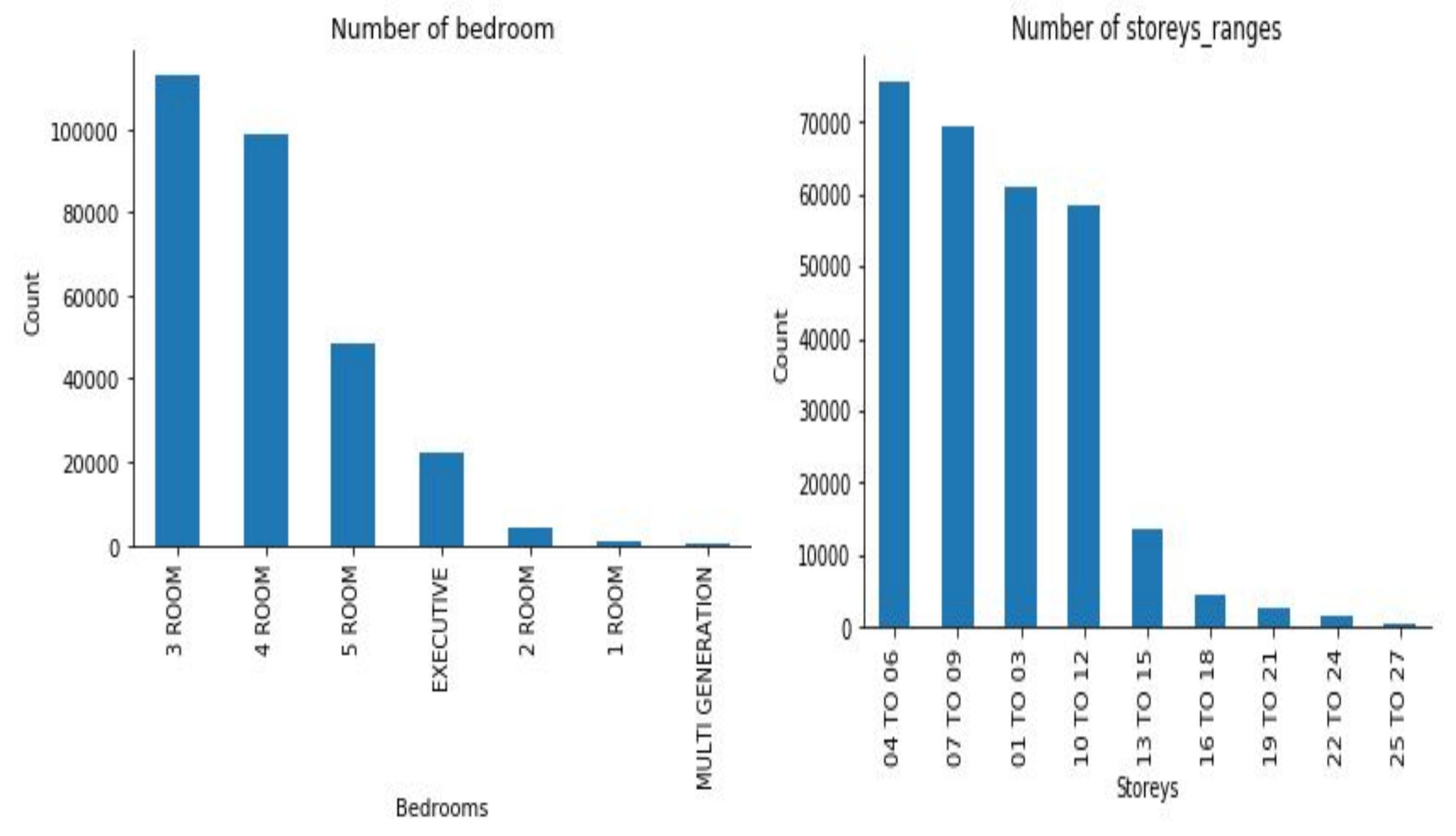
2) Similarly, APARTMENT,MODEL A-MAISONETTE and MAISONETTE flat models were used between 1980 and 2000 lease_commence year. For "MODEL A" AND "STANDARD" flat models, flat area is under 150 square meters.

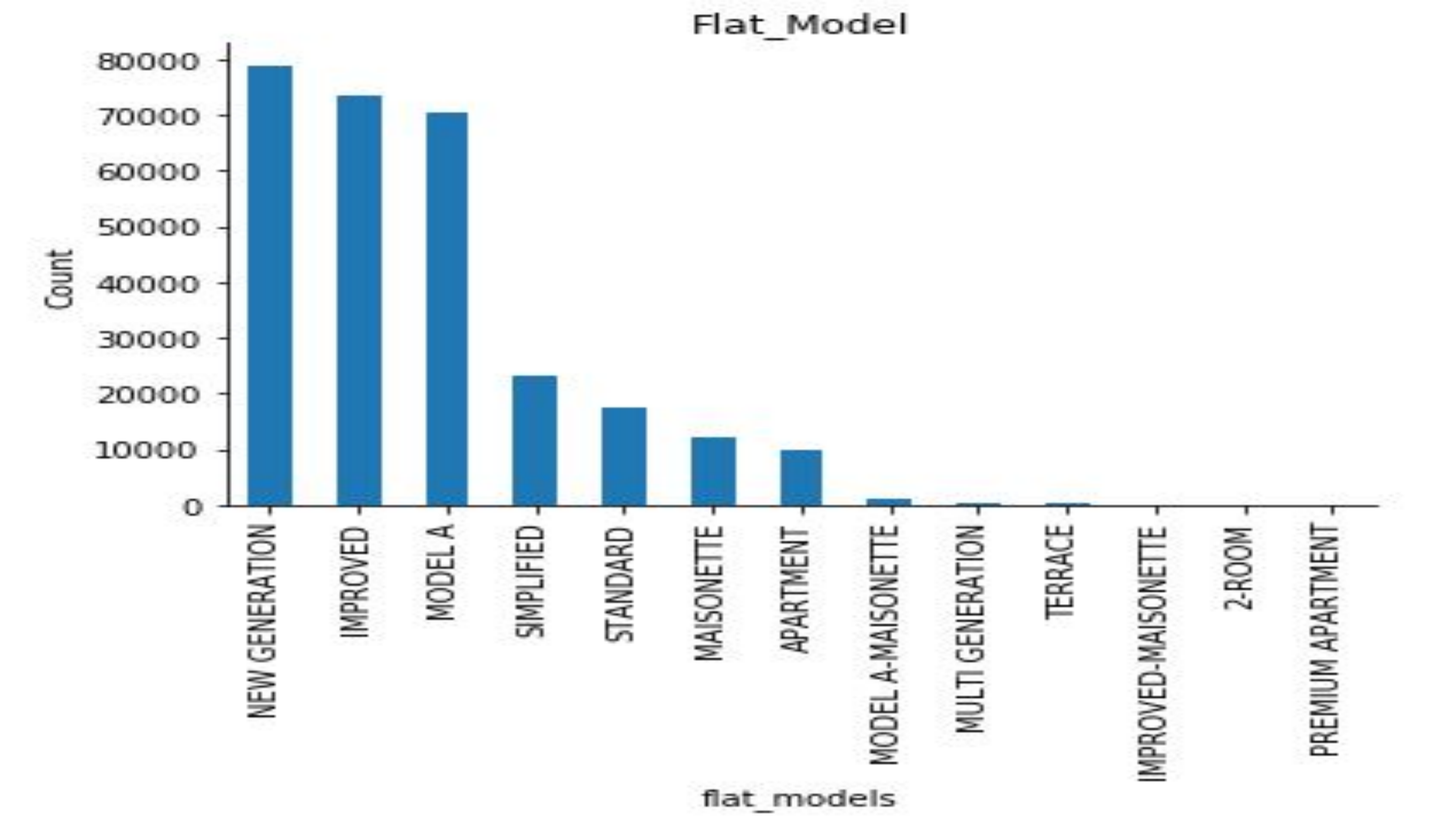
3) There is approximate a linear relationship between resale_price and floor_area_sqm MODEL A and STANDARD flat models have floor_area between 100 and 200 sqm for which resale price is between 200000 and 700000.

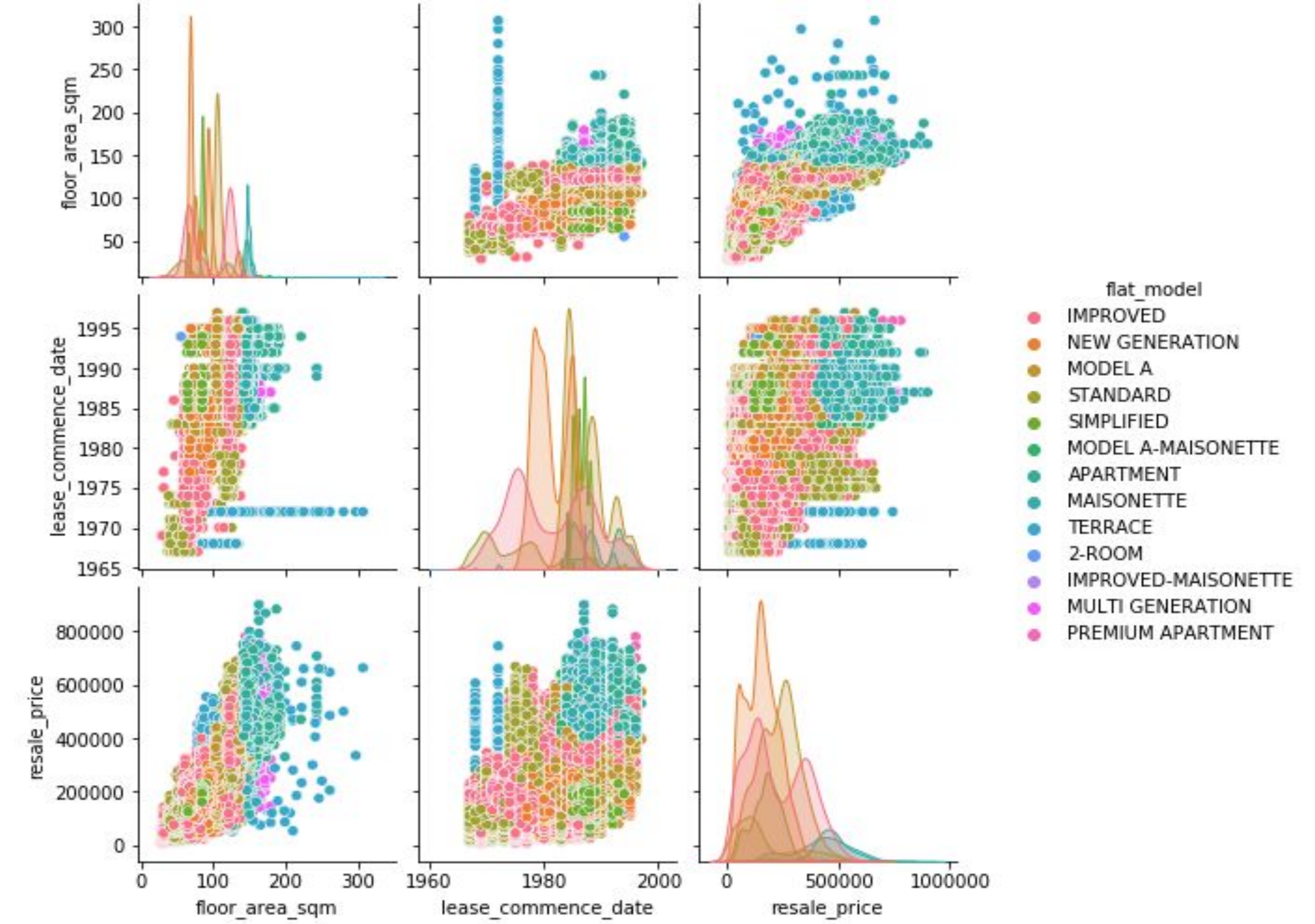
4) PREMIUM APARTMENT have floor_area between 0 and 150 sqm for which resale_prices are under 600000$. For "most" 2-ROOM and TERRACE flat models, lease_commence year is between 1980 and 1998 and resale_prices are above 400000$.

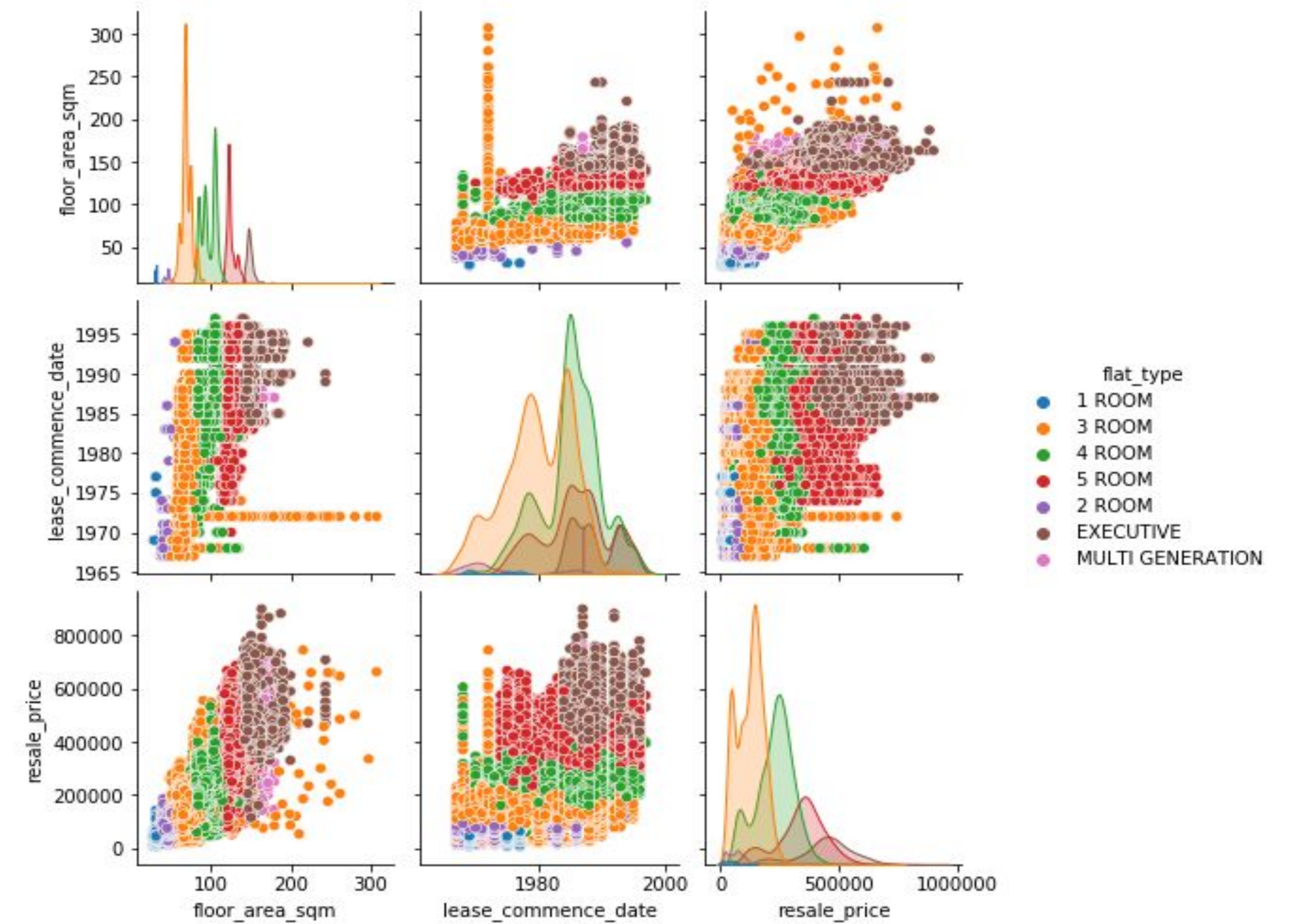
5) There is a spike for "3-ROOM" flat_type for earlier lease_commence_year of 1980 which shows total flat area in sqm is maximum in that year for "3-ROOM" flat type floor area for "4 ROOM" and "5 ROOM" flat_type is less than 150 sqm floor area for "1 ROOM" and "2 ROOM" flat_type is less than 50 sqm.

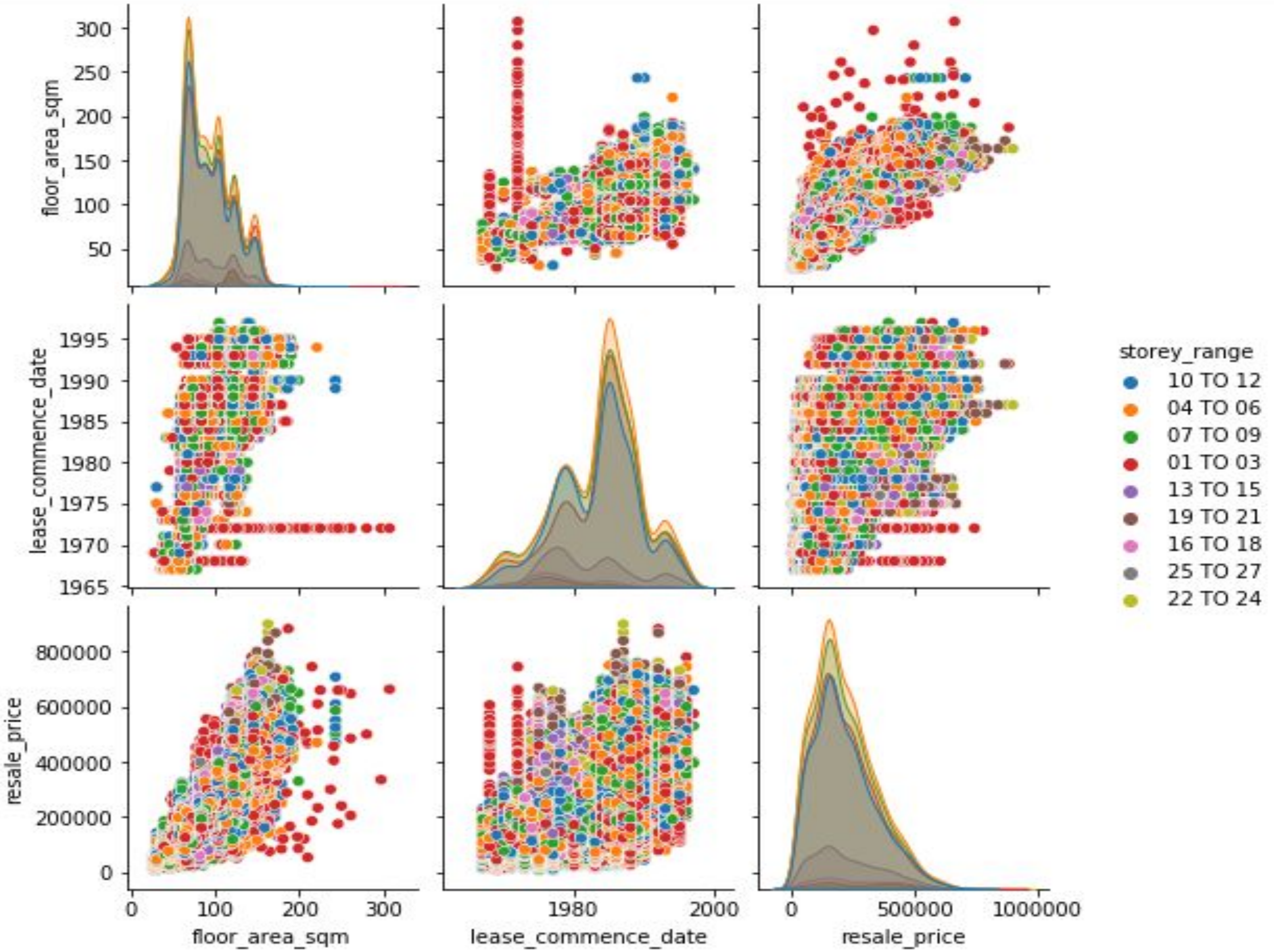
- For the given dataset, resale_price and floor_area_sqm are highly correlated.
- Correlation is a statistical measure to explain the relationship between two or more than two variables which are used to predict the values of target variable.
- If two variables or features are positively correlated with each other, it means when the value of one variable increases then the value of the other variable(s) also increases.
- Box-plots (another way of visualizing and analysing data with min,max,25,50 and 75 percentile values)

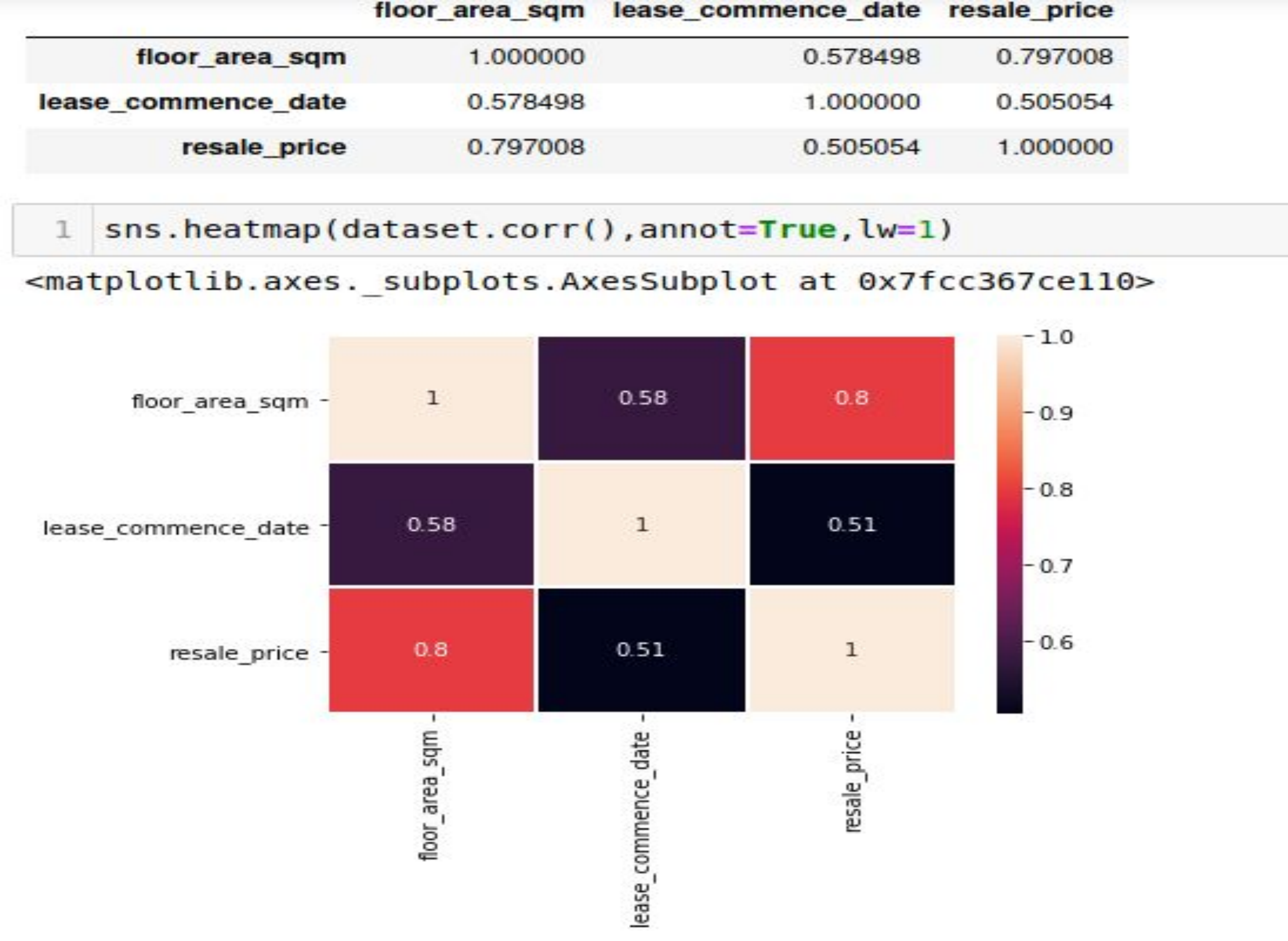

Number of bedroom
Count
100000
80000
60000
40000
20000
0
3 ROOM
4 ROOM
5 ROOM
EXECUTIVE
2 ROOM
1 ROOM
MULTI GENERATION
Bedrooms
Number of storeys_ranges
Count
70000
60000
50000
40000
30000
20000
10000
0
04 TO 06
07 TO 09
01 TO 03
10 TO 12
13 TO 15
16 TO 18
19 TO 21
22 TO 24
25 TO 27
Storeys


Flat_Model
Count
80000
70000
60000
50000
40000
30000
20000
10000
0
NEW GENERATION
IMPROVED
MODEL A
SIMPLIFIED
STANDARD
MAISONETTE
APARTMENT
MODEL A-MAISONETTE
MULTI GENERATION
TERRACE
IMPROVED-MAISONETTE
2-ROOM
PREMIUM APARTMENT
flat_models


floor_area_sqm
lease_commence_date
resale_price
300
250
200
150
100
50
1995
1990
1985
1980
1975
1970
1965
800000
600000
400000
200000
0
0
100
200
300
1960
1980
2000
0
500000
1000000
flat_model
IMPROVED
NEW GENERATION
MODEL A
STANDARD
SIMPLIFIED
MODEL A-MAISONETTE
APARTMENT
MAISONETTE
TERRACE
2-ROOM
IMPROVED-MAISONETTE
MULTI GENERATION
PREMIUM APARTMENT


floor_area_sqm
lease_commence_date
resale_price
300
250
200
150
100
50
1995
1990
1985
1980
1975
1970
1965
800000
600000
400000
200000
0
100
200
300
1980
2000
0
500000
1000000
flat_type
1 ROOM
3 ROOM
4 ROOM
5 ROOM
2 ROOM
EXECUTIVE
MULTI GENERATION

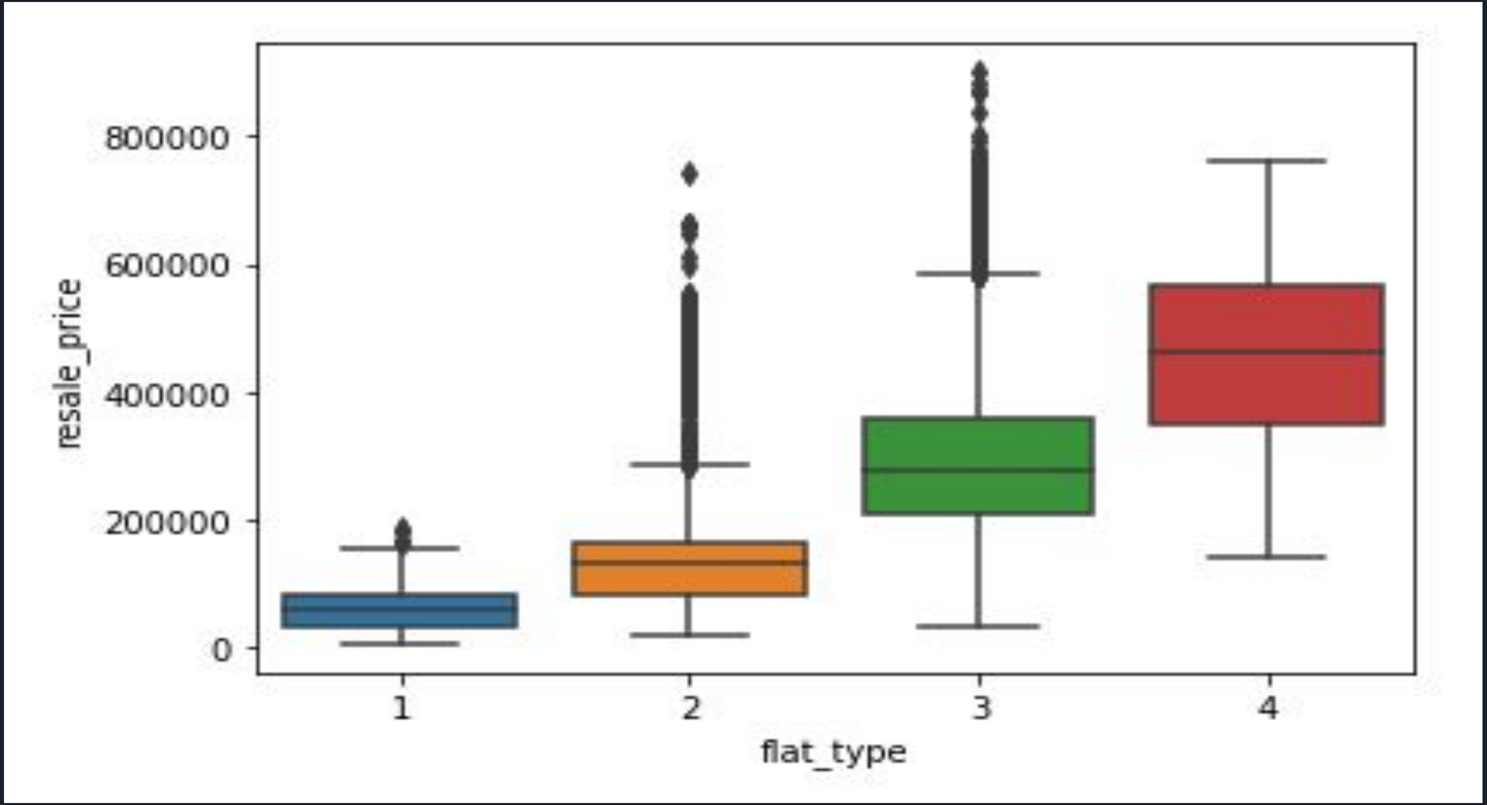

floor_area_sqm
300
250
200
150
100
50
lease_commence_date
1995
1990
1985
1980
1975
1970
1965
resale_price
800000
600000
400000
200000
0
0
100
200
300
floor_area_sqm
1980
2000
lease_commence_date
0
500000
1000000
resale_price
storey_range
10 TO 12
04 TO 06
07 TO 09
01 TO 03
13 TO 15
19 TO 21
16 TO 18
25 TO 27
22 TO 24

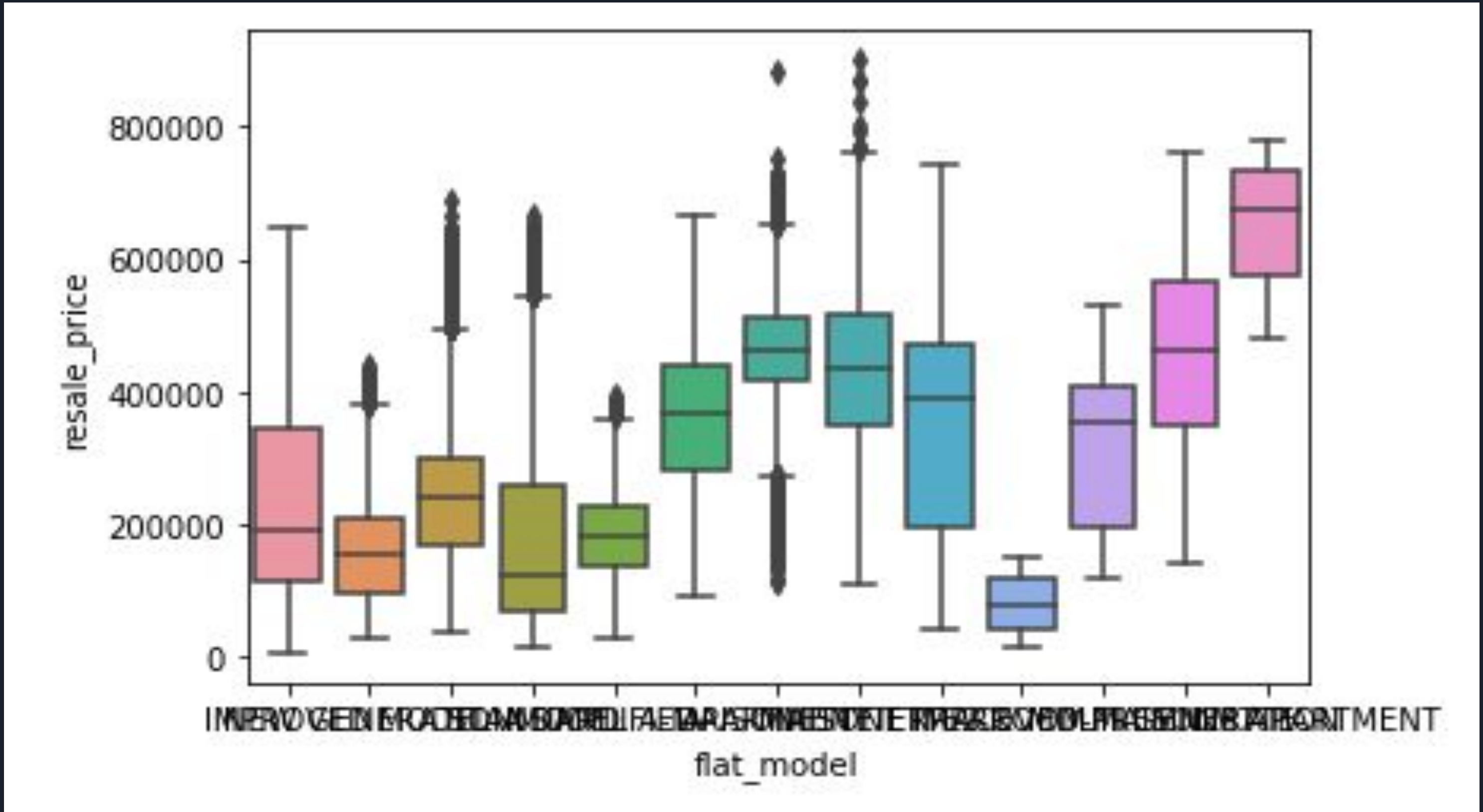


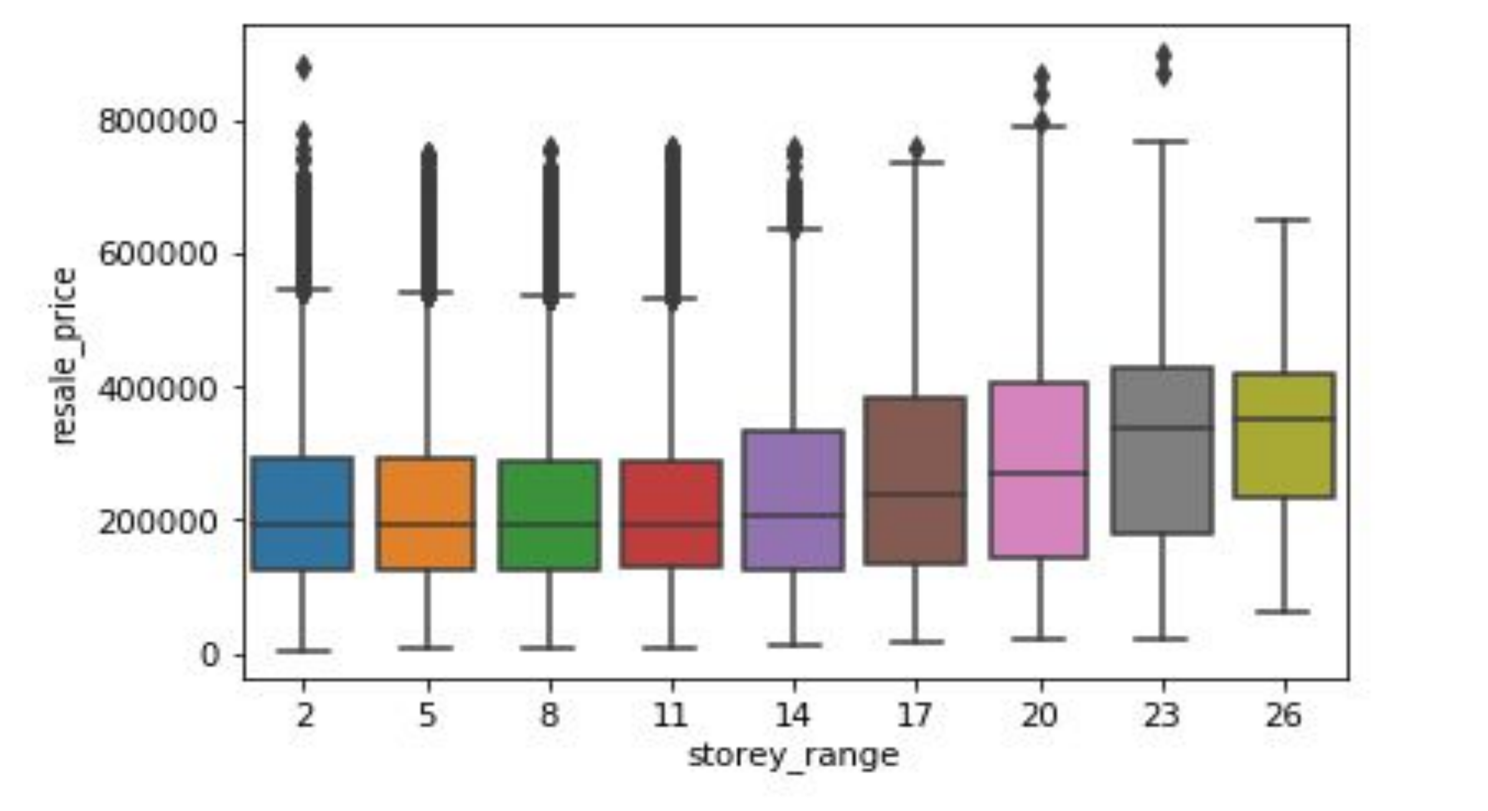
	floor_area_sqm	lease_commence_date	resale_price
floor_area_sqm	1.000000	0.578498	0.797008
lease_commence_date	0.578498	1.000000	0.505054
resale_price	0.797008	0.505054	1.000000

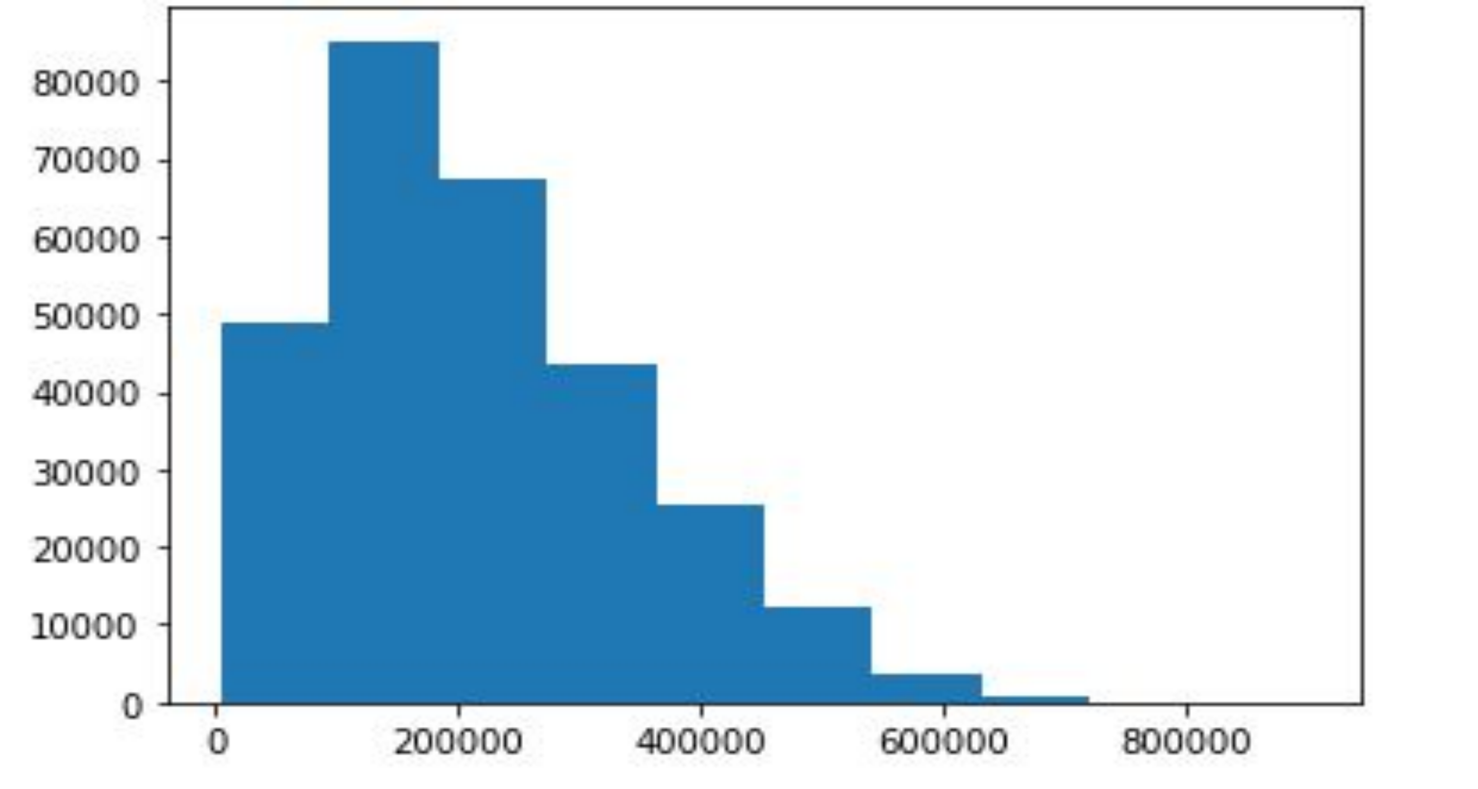
```
sns.heatmap(dataset.corr(),annot=True,lw=1)
```

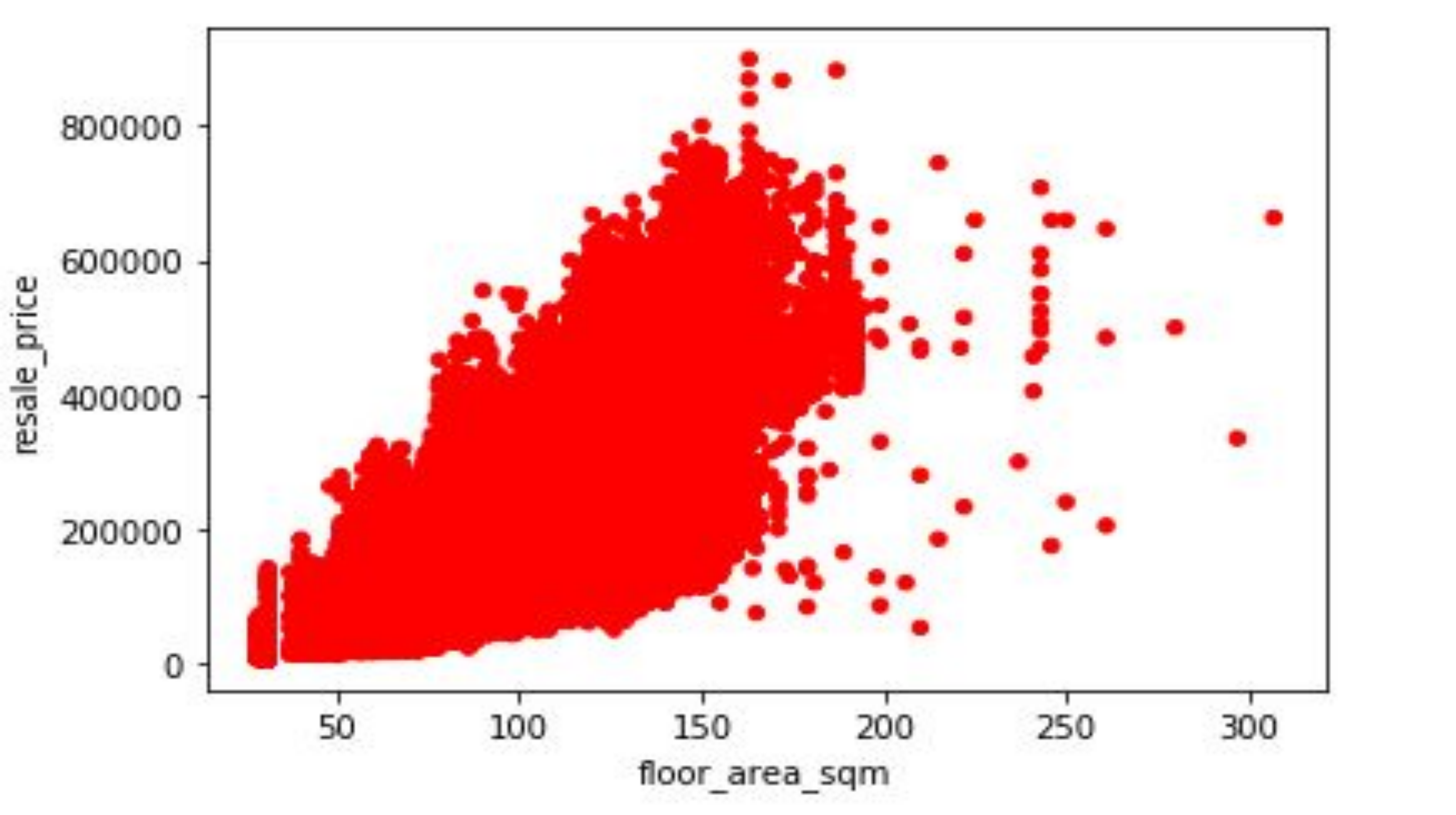
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fcc367ce110>
```

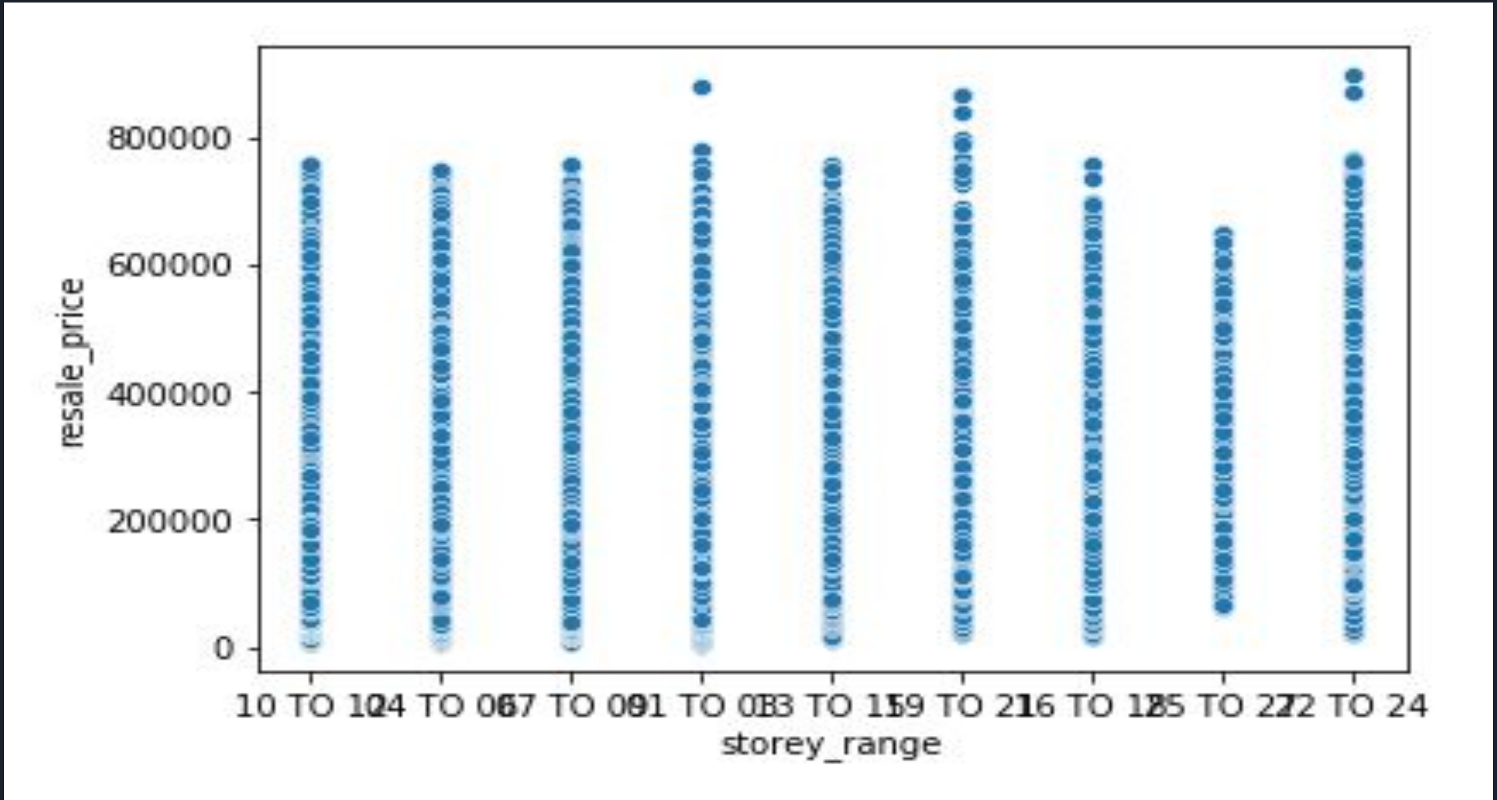

resale_price
800000
600000
400000
200000
0
1
2
3
4
flat_type

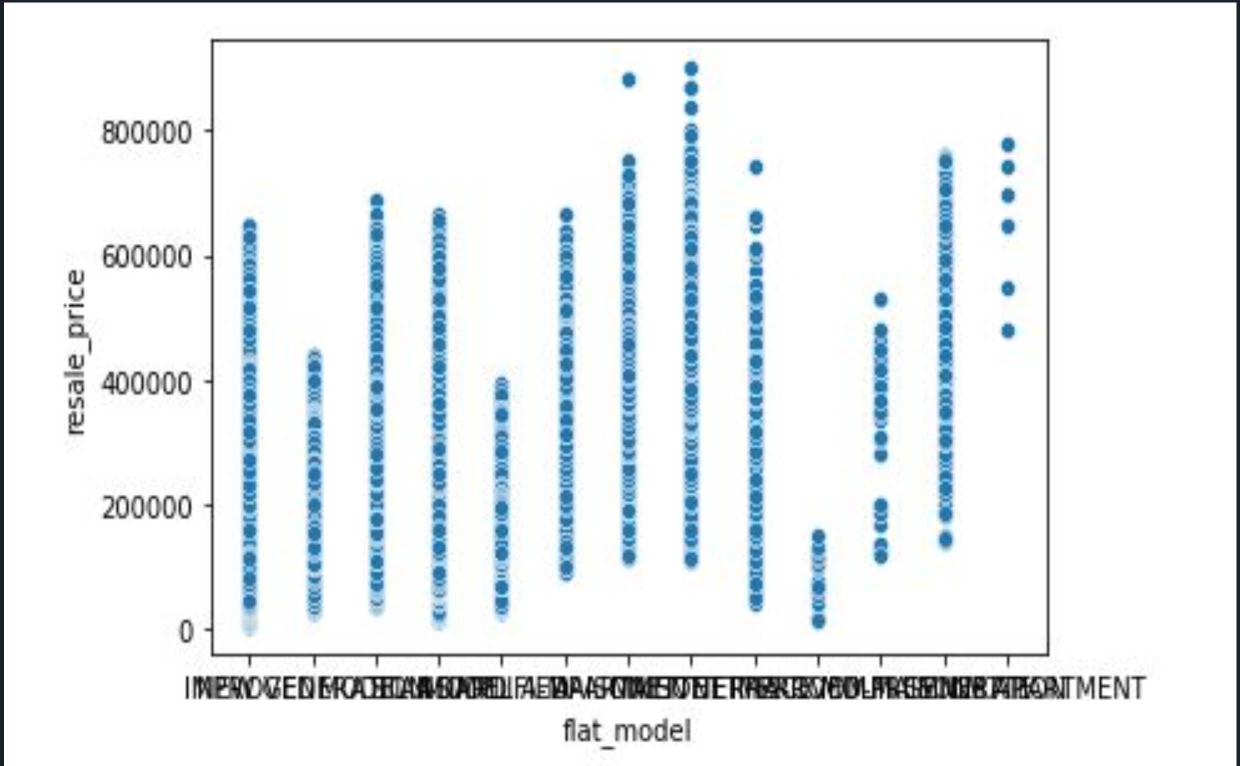

800000
600000
400000
200000
0
resale_price
flat_model

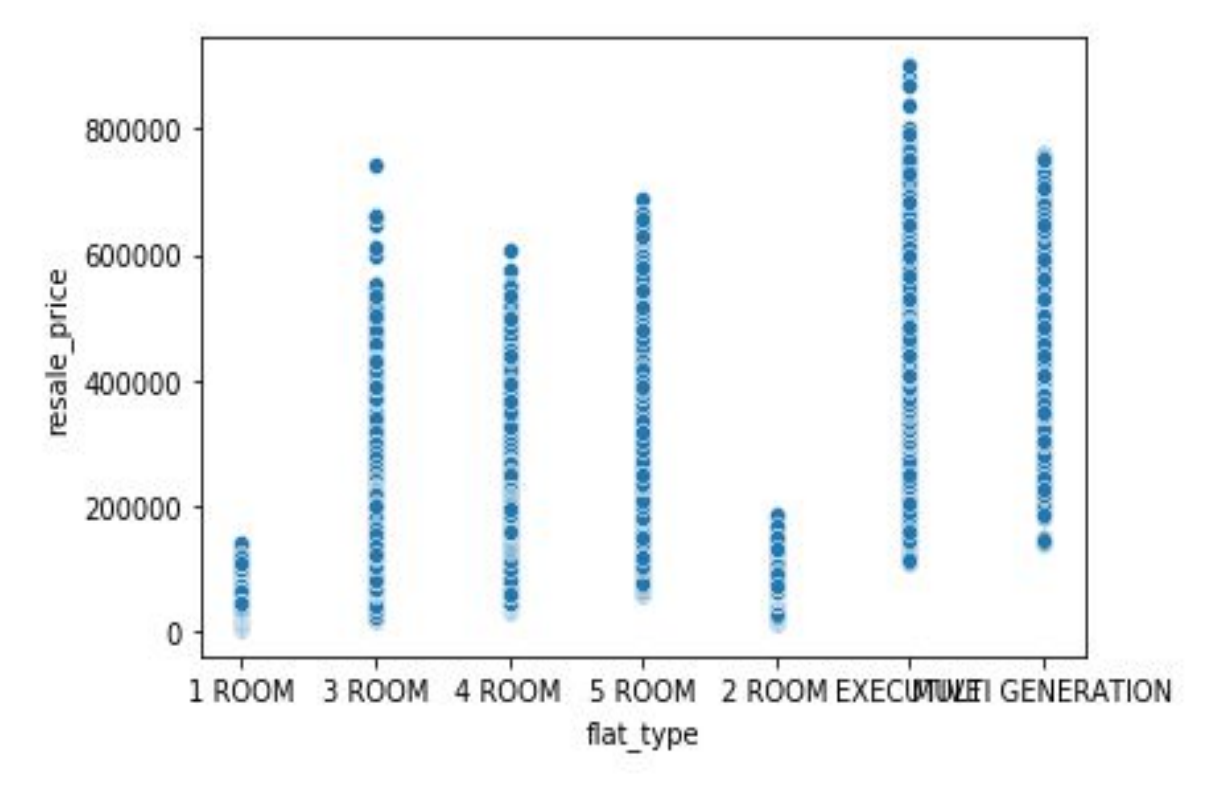

resale_price
800000
600000
400000
200000
0
2
5
8
11
14
17
20
23
26
storey_range


80000
70000
60000
50000
40000
30000
20000
10000
0
0
200000
400000
600000
800000


800000
600000
400000
200000
0
resale_price
50
100
150
200
250
300
floor_area_sqm


resale_price
800000
600000
400000
200000
0
10 TO 12
04 TO 06
07 TO 09
01 TO 03
13 TO 15
19 TO 21
16 TO 18
25 TO 27
22 TO 24
storey_range


800000
600000
400000
200000
0
resale_price
flat_model

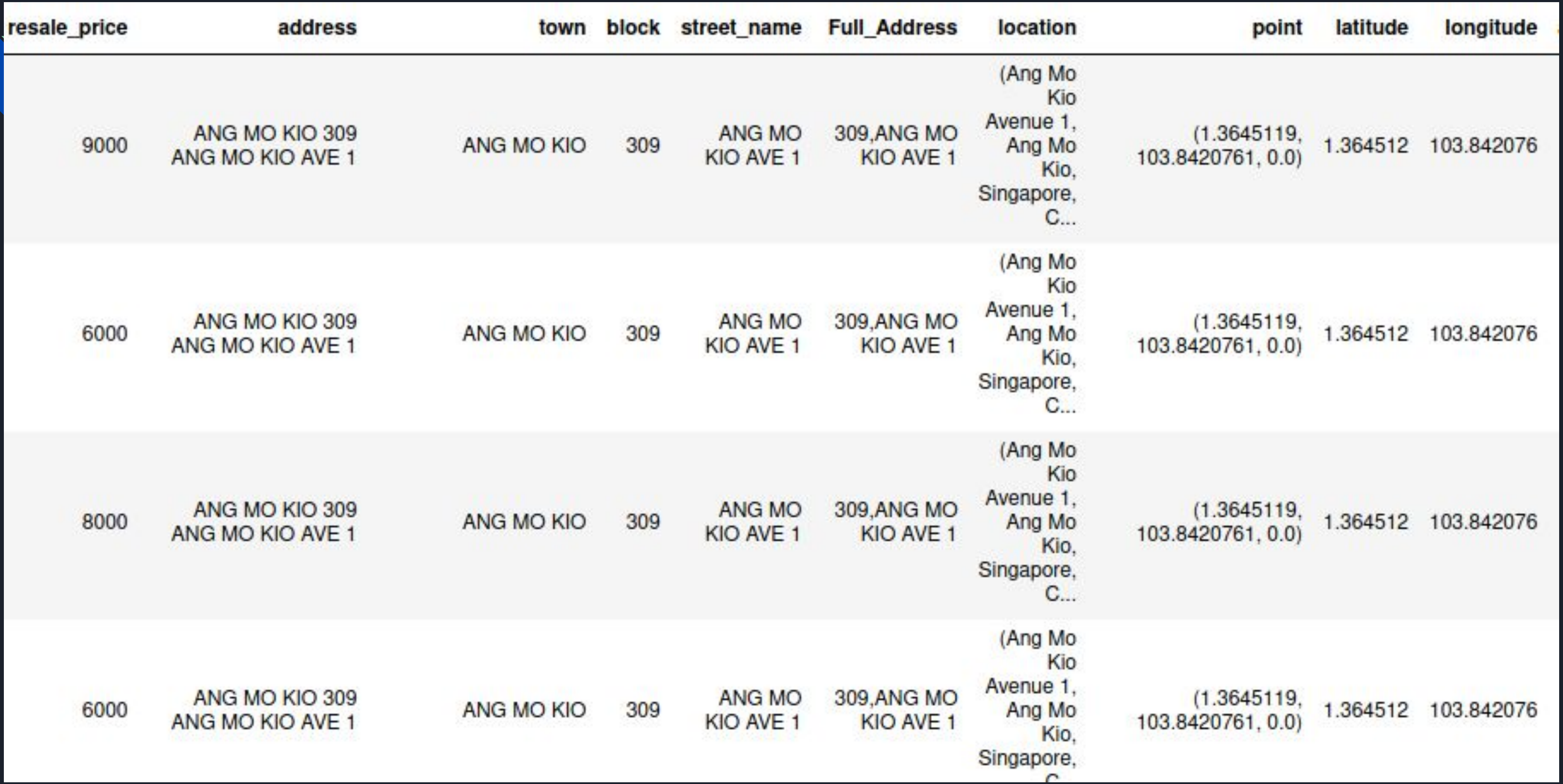

resale_price
800000
600000
400000
200000
0
1 ROOM
3 ROOM
4 ROOM
5 ROOM
2 ROOM
EXECUTIVE
MULTI GENERATION
flat_type

On Adding latitude and longitude columns based on the address given in dataset

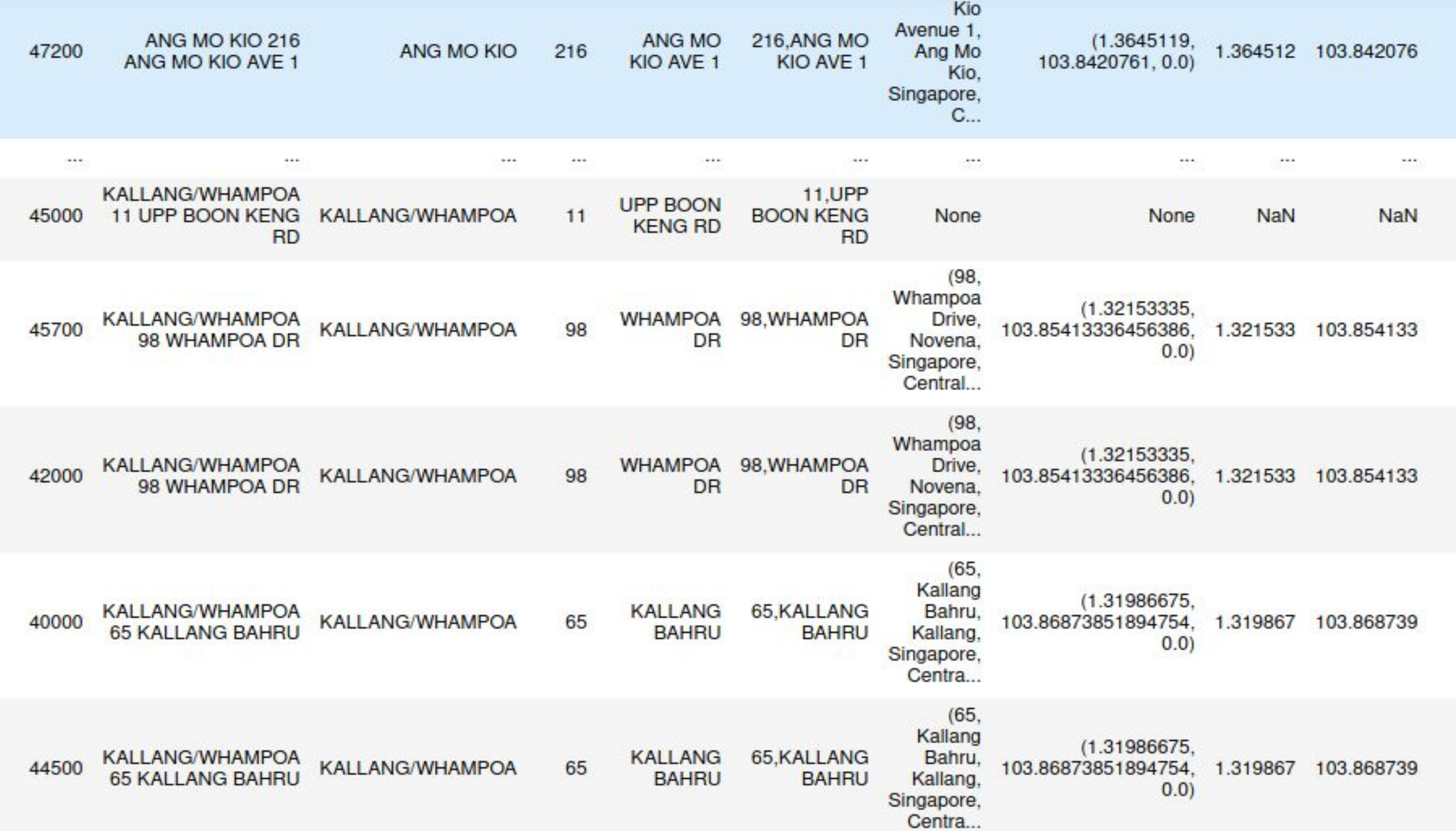
❖ As the record size is 287200 in the given dataset and there is a problem of time out if we use geopy library to convert given address into latitude and longitude. So, just to analyze the data, I have considered less record size with 1000 rows.

❖ I have made a single address column by concatenating 3 separate addresses as :

```
#dataset["address"] = dataset["town"] + "  " + dataset["block"] + " " + dataset["street_name"]
```



resale_price	address	town	block	street_name	Full_Address	location	point	latitude	longitude
9000	ANG MO KIO 309 ANG MO KIO AVE 1	ANG MO KIO	309	ANG MO KIO AVE 1	309,ANG MO KIO AVE 1	(Ang Mo Kio Avenue 1, Ang Mo Kio, Singapore, C...	(1.3645119, 103.8420761, 0.0)	1.364512	103.842076
6000	ANG MO KIO 309 ANG MO KIO AVE 1	ANG MO KIO	309	ANG MO KIO AVE 1	309,ANG MO KIO AVE 1	(Ang Mo Kio Avenue 1, Ang Mo Kio, Singapore, C...	(1.3645119, 103.8420761, 0.0)	1.364512	103.842076
8000	ANG MO KIO 309 ANG MO KIO AVE 1	ANG MO KIO	309	ANG MO KIO AVE 1	309,ANG MO KIO AVE 1	(Ang Mo Kio Avenue 1, Ang Mo Kio, Singapore, C...	(1.3645119, 103.8420761, 0.0)	1.364512	103.842076
6000	ANG MO KIO 309 ANG MO KIO AVE 1	ANG MO KIO	309	ANG MO KIO AVE 1	309,ANG MO KIO AVE 1	(Ang Mo Kio Avenue 1, Ang Mo Kio, Singapore,	(1.3645119, 103.8420761, 0.0)	1.364512	103.842076

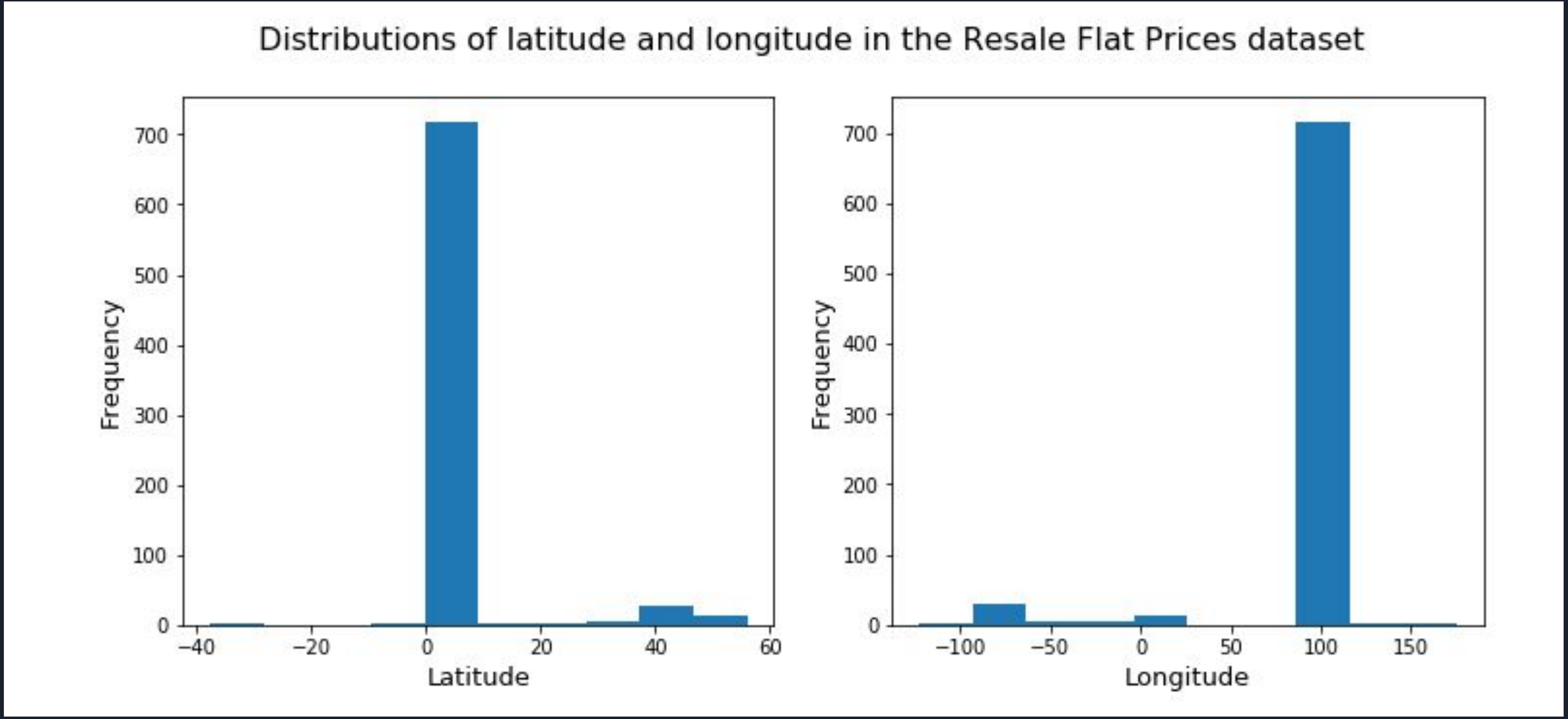


47200	ANG MO KIO 216 ANG MO KIO AVE 1	ANG MO KIO	216	ANG MO KIO AVE 1	216,ANG MO KIO AVE 1	Kio Avenue 1, Ang Mo Kio, Singapore, C...	(1.3645119, 103.8420761, 0.0)	1.364512	103.842076
...	...	...	...	...	...	...	...	...	...
45000	KALLANG/WHAMPOA 11 UPP BOON KENG RD	KALLANG/WHAMPOA	11	UPP BOON KENG RD	11,UPP BOON KENG RD	None	None	NaN	NaN
45700	KALLANG/WHAMPOA 98 WHAMPOA DR	KALLANG/WHAMPOA	98	WHAMPOA DR	98,WHAMPOA DR	(98, Whampoa Drive, Novena, Singapore, Central...	(1.32153335, 103.85413336456386, 0.0)	1.321533	103.854133
42000	KALLANG/WHAMPOA 98 WHAMPOA DR	KALLANG/WHAMPOA	98	WHAMPOA DR	98,WHAMPOA DR	(98, Whampoa Drive, Novena, Singapore, Central...	(1.32153335, 103.85413336456386, 0.0)	1.321533	103.854133
40000	KALLANG/WHAMPOA 65 KALLANG BAHRU	KALLANG/WHAMPOA	65	KALLANG BAHRU	65,KALLANG BAHRU	(65, Kallang Bahru, Kallang, Singapore, Centra...	(1.31986675, 103.86873851894754, 0.0)	1.319867	103.868739
44500	KALLANG/WHAMPOA 65 KALLANG BAHRU	KALLANG/WHAMPOA	65	KALLANG BAHRU	65,KALLANG BAHRU	(65, Kallang Bahru, Kallang, Singapore, Centra...	(1.31986675, 103.86873851894754, 0.0)	1.319867	103.868739

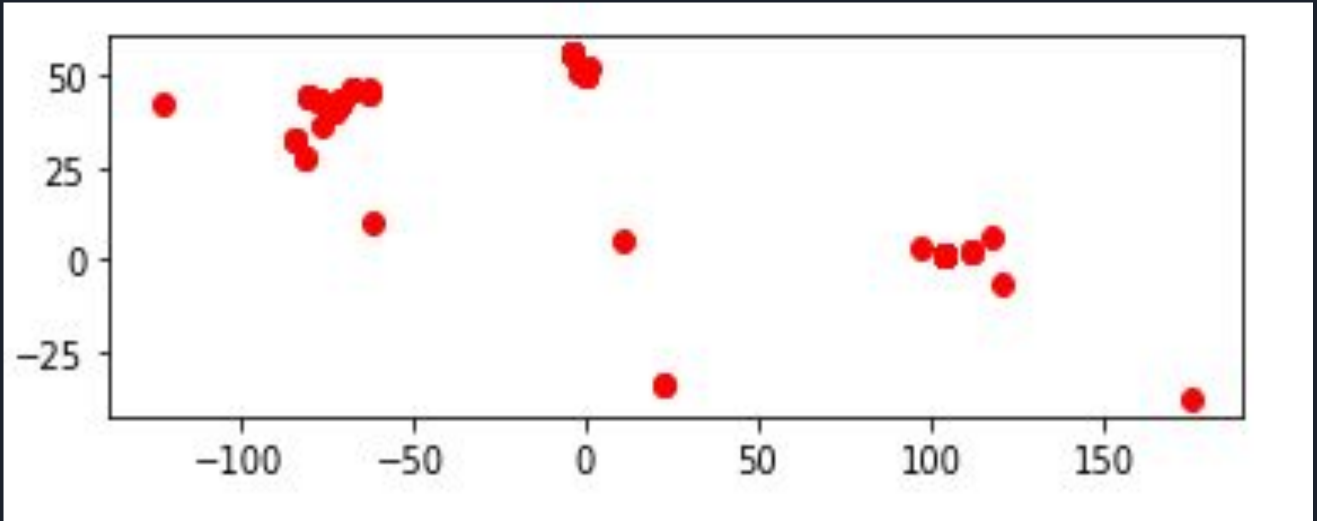
- I have removed the Removing the undefined values in the form of 'NaN' :

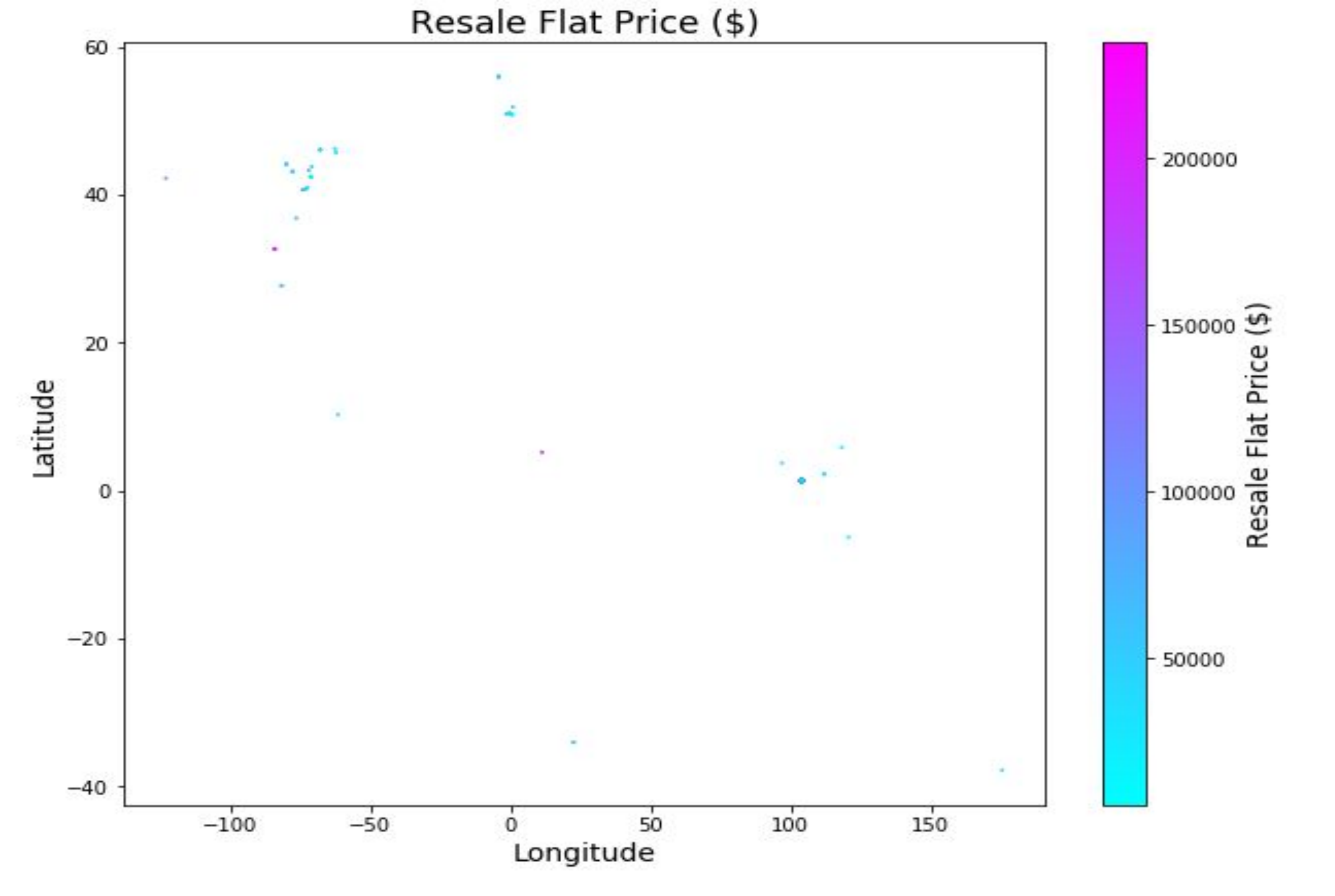
```
# new_dataset=modified_dataset.dropna()
```

Now, We are going to see the common locations where the flats are placed.



- With latitude range 0 to 10, maximum flats were sold and same for longitude around 100. So, these locations might be ideal location for flat sale in future also.



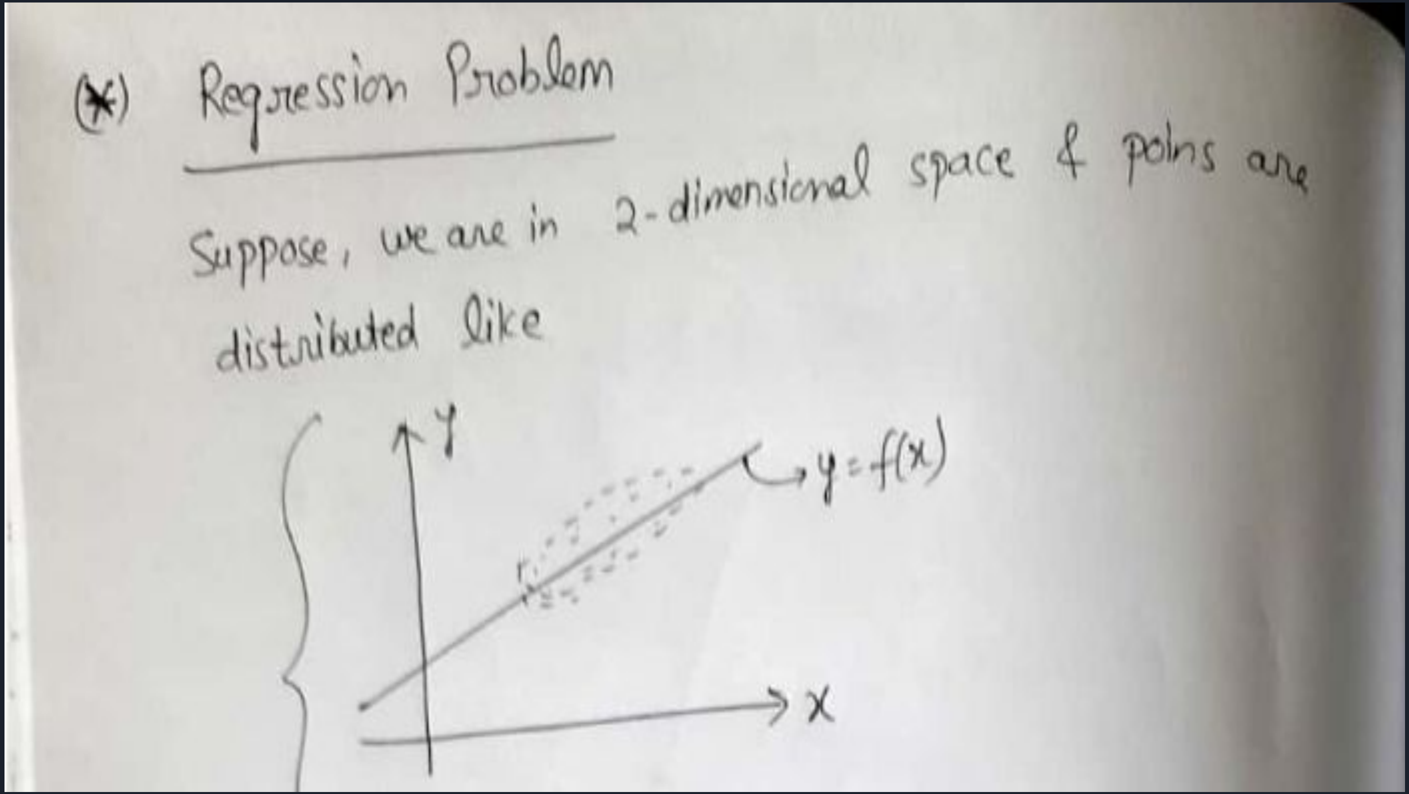

Resale Flat Price ($)
60
40
20
0
−20
−40
Latitude
−100
−50
0
50
100
150
Longitude
200000
150000
100000
50000
Resale Flat Price ($)

- As we can see Locations for which longitude is between -100 to 50 and latitude between 30 to 50, Flat prices are below 5000$.

Building the Model: Predicting the resale flat price

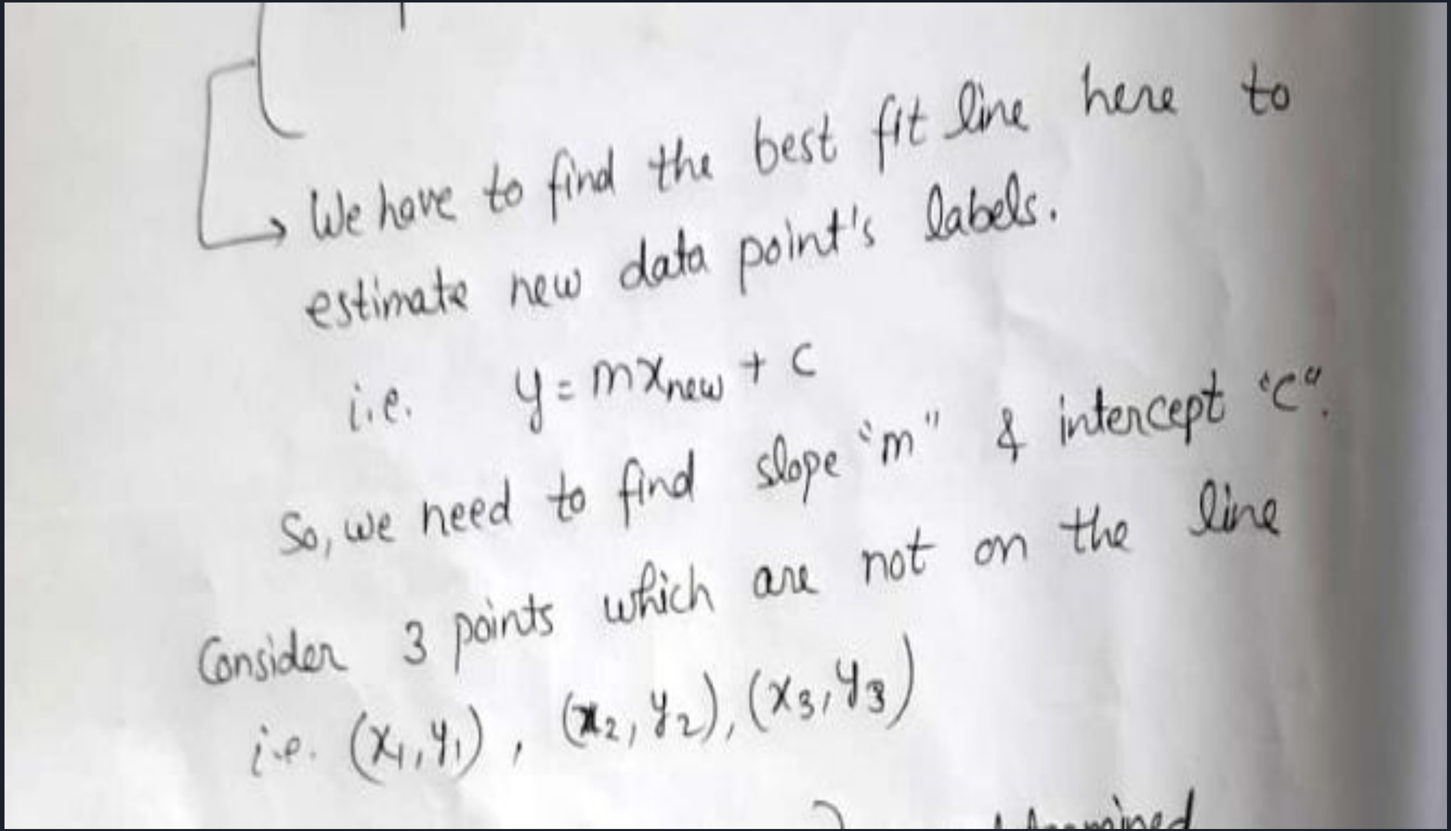
- Once we get a good fit, we will use this model to predict the sale price of the flat.
- I have used the Linear Regression model here to get the best fit hyperplane for the given datapoints.
- Equation of Best Fit Hyperplane: $z^{*} = (A^{T}A)^{-1} A^{T}b$ for system of equations $Az=b$.
- where , vector b is not in plane of column vectors of matrix A and to get the approximate solution, we have to project the vector ‘b’ in the plane which is ƀ and so solving $Az^{*} =$ ƀ, we get, $z^{*} = (A^{T}A)^{-1}A^{T}b$.

Maths behind Linear Regression



(*) Regression Problem

Suppose, we are in 2-dimensional space & poins are distributed like



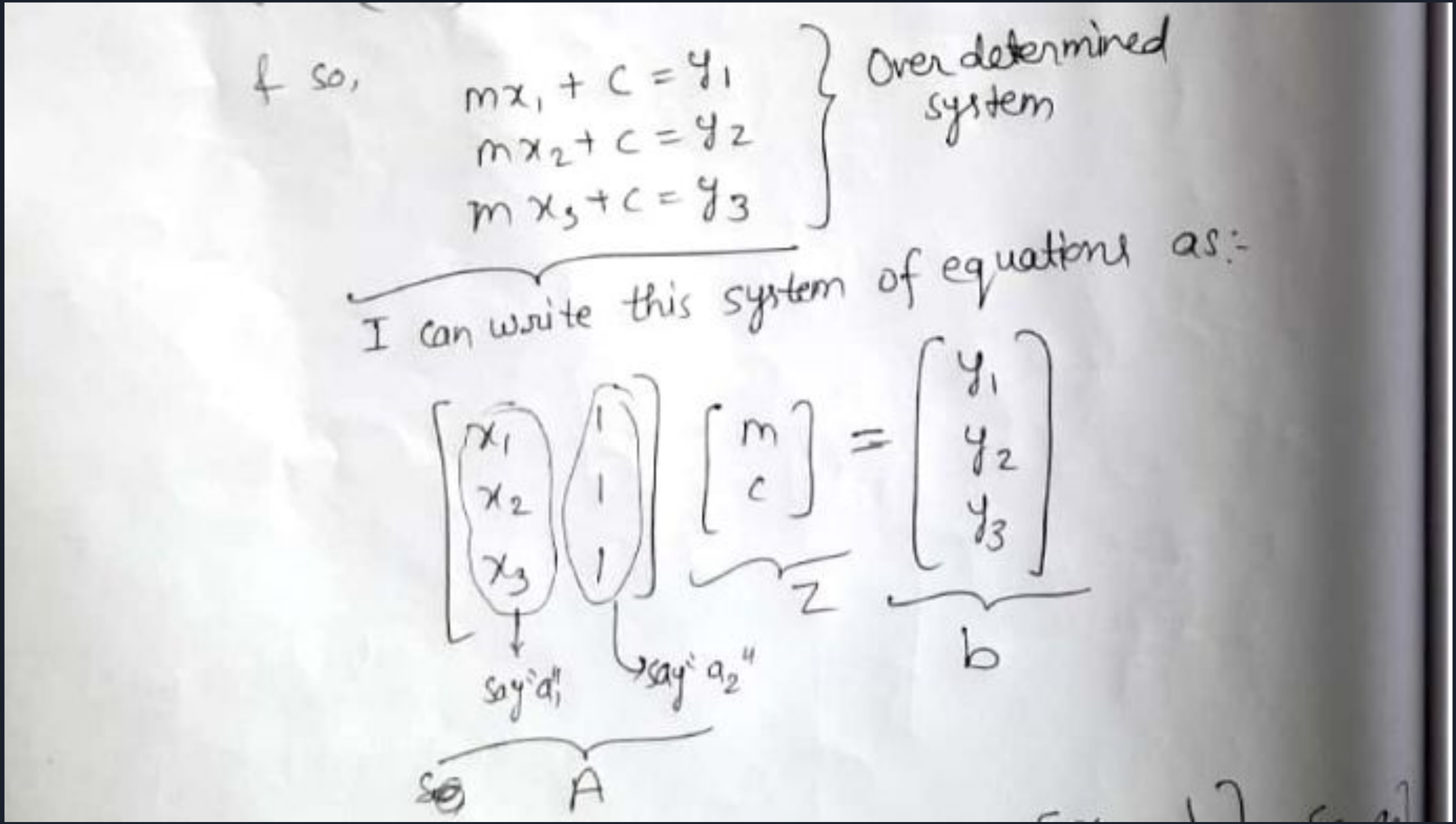
We have to find the best fit line here to estimate new data point's labels.

i.e. $y = mx_{new} + c$

So, we need to find slope "m" & intercept "c".

Consider 3 points which are not on the line

i.e. (x_1, y_1), (x_2, y_2), (x_3, y_3)



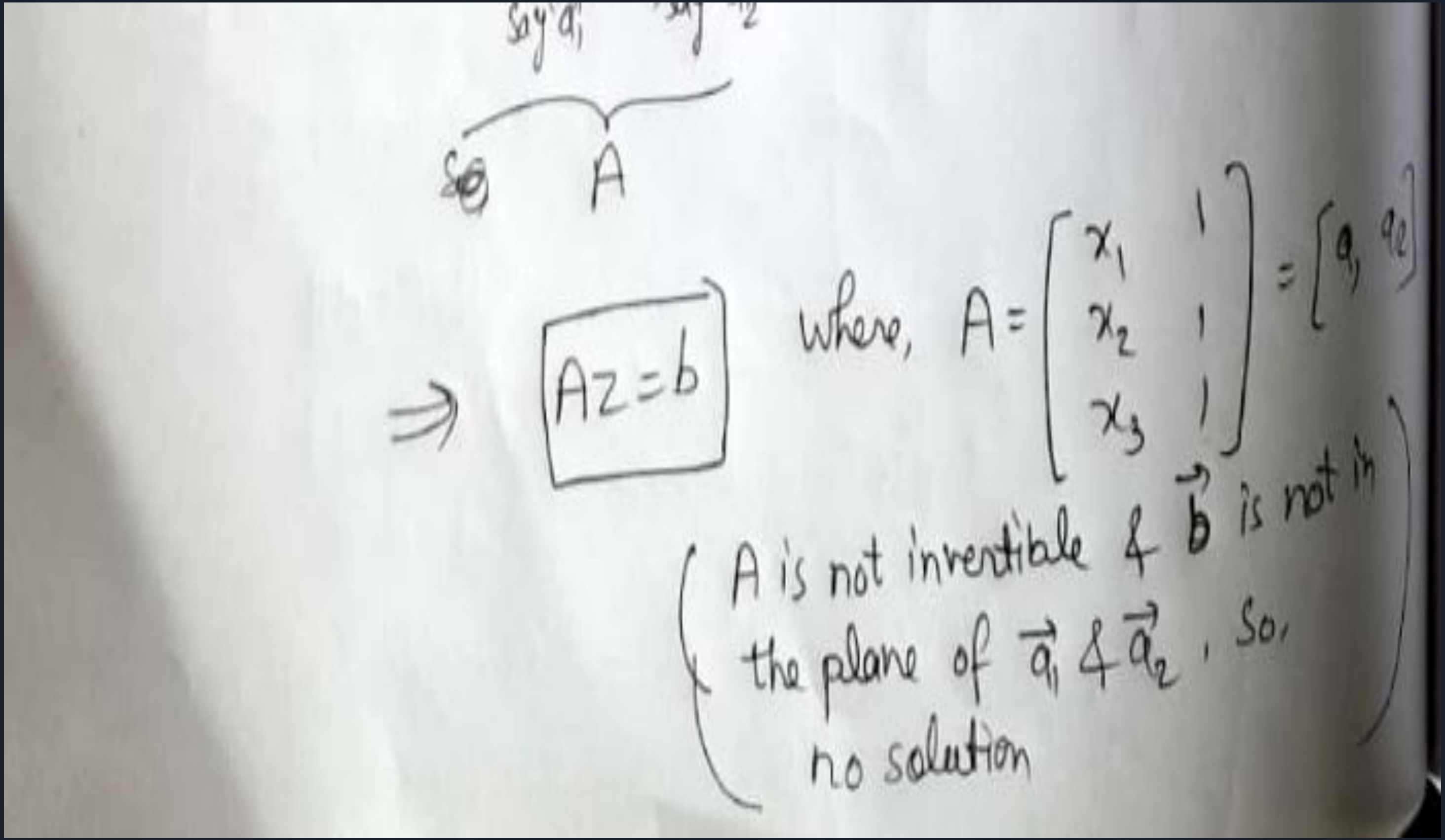
f so,

$$\left.\begin{aligned} mx_1 + c &= y_1 \\ mx_2 + c &= y_2 \\ mx_3 + c &= y_3 \end{aligned}\right\}\ \text{Over determined system}$$

I can write this system of equations as:-

$$\underbrace{\begin{array}{c} \left[\begin{array}{cc} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{array}\right] \\ \text{say "}a_1\text{"} \quad \text{say "}a_2\text{"} \end{array}}_{A} \underbrace{\begin{bmatrix} m \\ c \end{bmatrix}}_{z} = \underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}}_{b}$$

So

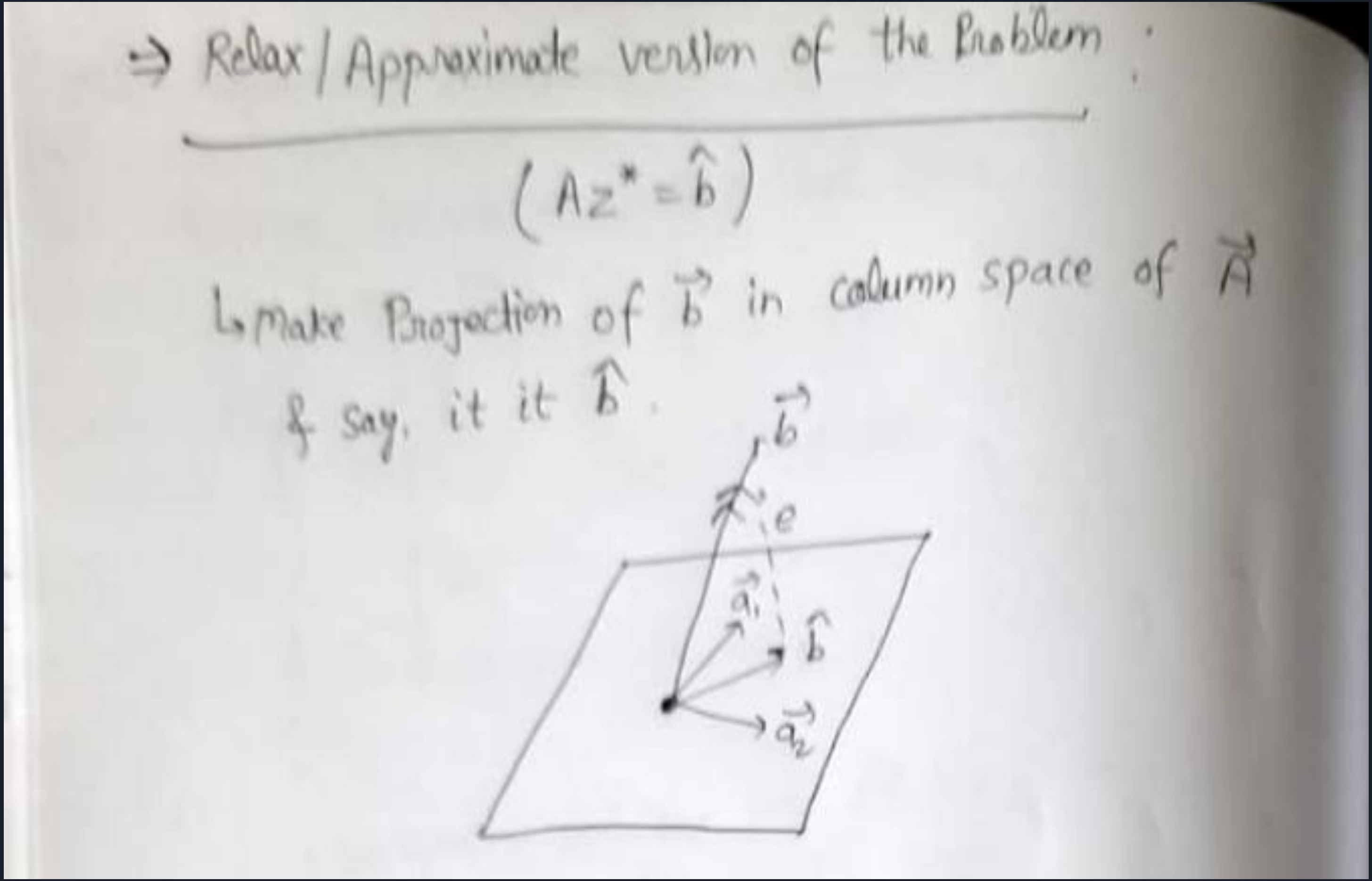


say a_1 say a_2

So $\underbrace{}_{A}$

$\Rightarrow \boxed{Az = b}$ where, $A = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 \end{bmatrix}$

(A is not invertible & $\vec{b}$ is not in the plane of $\vec{a}_1$ & $\vec{a}_2$. So, no solution)

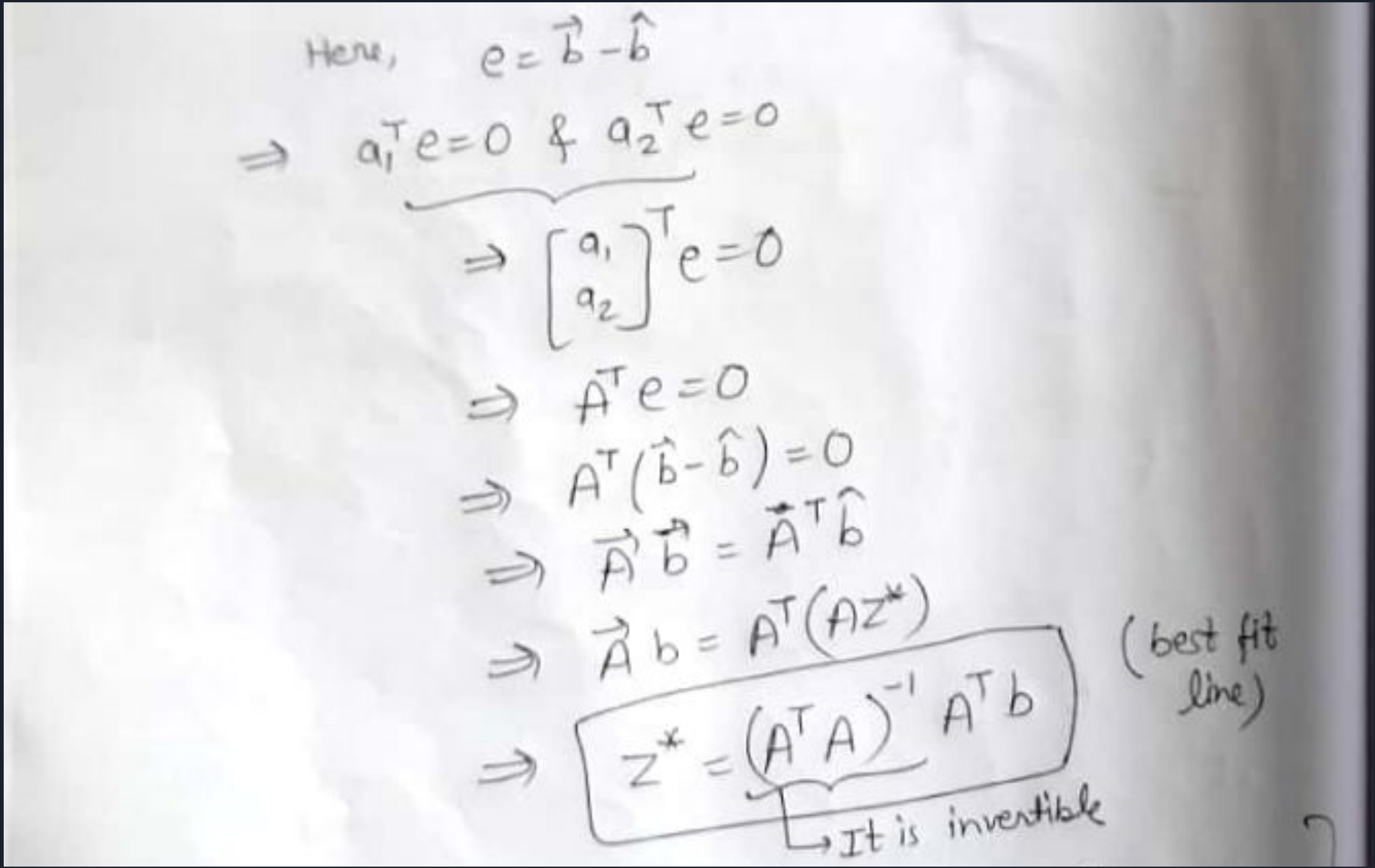


⇒ Relax / Approximate version of the Problem :

$(Az^* = \hat{b})$

↳ make Projection of $\vec{b}$ in column space of $\vec{A}$ & say, it it $\hat{b}$.

$\vec{b}$
$\vec{e}$
$\vec{a_1}$
$\hat{b}$
$\vec{a_2}$



Here, $e = \vec{b} - \hat{b}$

$$\Rightarrow \underbrace{a_1^T e = 0 \ \& \ a_2^T e = 0}$$

$$\Rightarrow \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}^T e = 0$$

$$\Rightarrow A^T e = 0$$

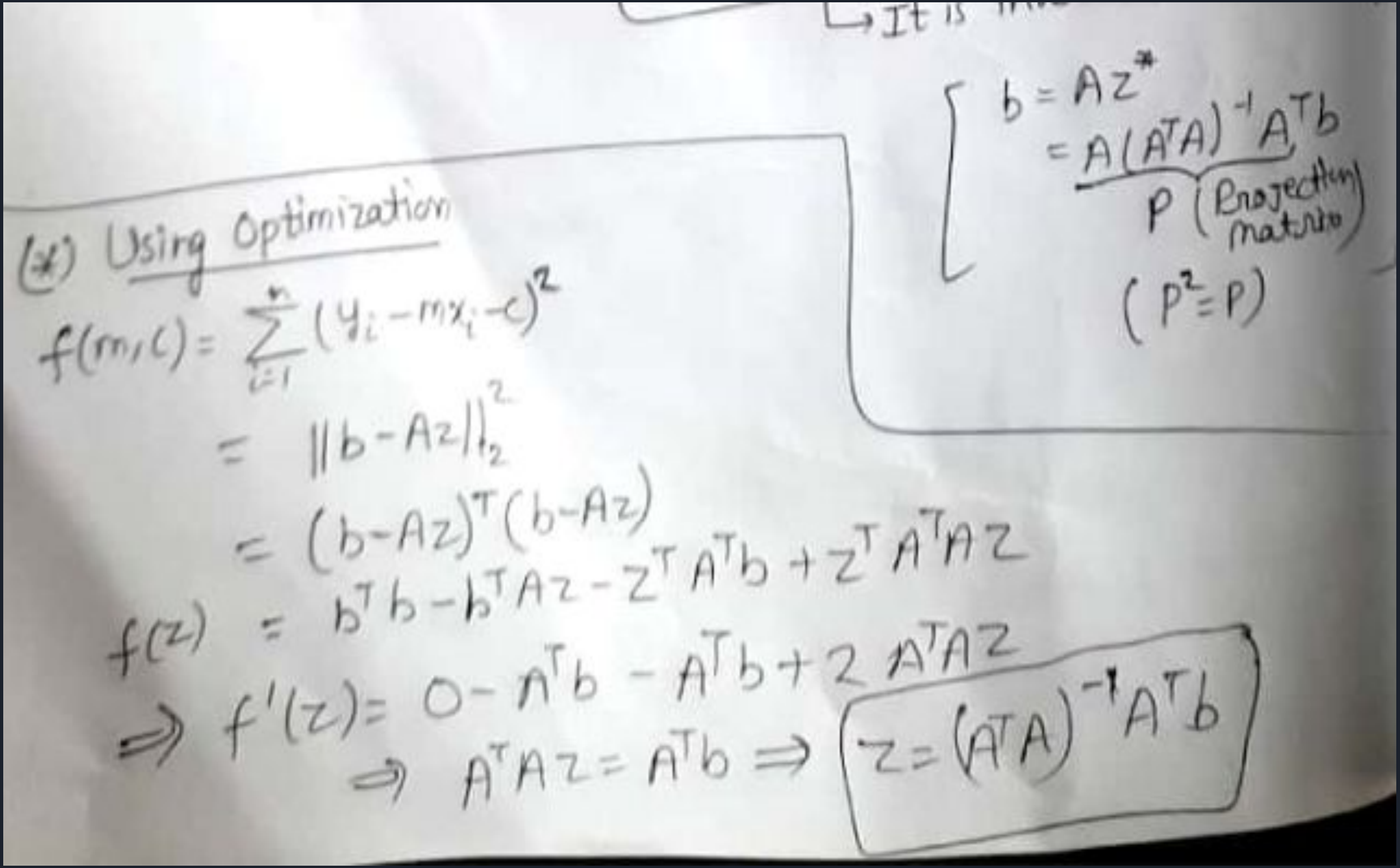
$$\Rightarrow A^T(\vec{b} - \hat{b}) = 0$$

$$\Rightarrow \vec{A}\vec{b} = \vec{A}^T \hat{b}$$

$$\Rightarrow \vec{A} b = A^T(Az^*)$$

$$\Rightarrow \boxed{z^* = \underbrace{(A^T A)^{-1}}_{\text{It is invertible}} A^T b}$$

(best fit line)



↳ It is

$$b = Az^*$$

$$= \underbrace{A(A^TA)^{-1}A^T}_{P}b$$

P (Projection matrix)

$(P^2 = P)$

(*) Using Optimization

$$f(m,c) = \sum_{i=1}^{n}(y_i - mx_i - c)^2$$

$$= \|b - Az\|_2^2$$

$$= (b-Az)^T(b-Az)$$

$$f(z) = b^Tb - b^TAz - z^TA^Tb + z^TA^TAz$$

$$\Rightarrow f'(z) = 0 - A^Tb - A^Tb + 2A^TAz$$

$$\Rightarrow A^TAz = A^Tb \Rightarrow \boxed{z = (A^TA)^{-1}A^Tb}$$

Thank You !