Predicting Singapore's HDB resale prices (2015 - 2020)

Using HDB resale data from 2015-2020 to create a machine learning model to predict price per square foot for resale HDBs

Part 1: Understanding factors affecting HDB resale prices:

- Data Preparation
- Data Cleaning
- Data Visualization

Part 2: Using machine learning to predict future HDB resale prices:

- Choosing the right machine learning model
- Hyperparameter Tuning (model is acheived good results - therefore did not tune parameters)

Used comments to elaborate on the code

In [126…

```
from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity = "all"
```

In [127…

```
#import the packages for Part 1

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [128…

```
# load

resale2015 = pd.read_csv('resale_flat_prices_from_jan_2015_to_dec_2016.csv')
resale2017 = pd.read_csv('resale_flat_prices_from_jan_2017_onwards.csv')
```

In [129…

```
# Understanding the data
# shape

print (resale2015.shape)
print (resale2017.shape)

# Understanding the data
# headers

resale2015.head()
resale2017.head()
```

```
(37153, 11)
(77895, 11)
```

Out[129…

	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease_commence_date	rer
0	2015-01	ANG MO KIO	3 ROOM	174	ANG MO KIO AVE 4	07 TO 09	60.0	Improved	1986	

	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease_commence_date	rer
1	2015-01	ANG MO KIO	3 ROOM	541	ANG MO KIO AVE 10	01 TO 03	68.0	New Generation	1981	
2	2015-01	ANG MO KIO	3 ROOM	163	ANG MO KIO AVE 4	01 TO 03	69.0	New Generation	1980	
3	2015-01	ANG MO KIO	3 ROOM	446	ANG MO KIO AVE 10	01 TO 03	68.0	New Generation	1979	
4	2015-01	ANG MO KIO	3 ROOM	557	ANG MO KIO AVE 10	07 TO 09	68.0	New Generation	1980	

Out[129…

	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease_commence_date	rer
0	2017-01	ANG MO KIO	2 ROOM	406	ANG MO KIO AVE 10	10 TO 12	44.0	Improved	1979	
1	2017-01	ANG MO KIO	3 ROOM	108	ANG MO KIO AVE 4	01 TO 03	67.0	New Generation	1978	
2	2017-01	ANG MO KIO	3 ROOM	602	ANG MO KIO AVE 5	01 TO 03	67.0	New Generation	1980	
3	2017-01	ANG MO KIO	3 ROOM	465	ANG MO KIO AVE 10	04 TO 06	68.0	New Generation	1980	
4	2017-01	ANG MO KIO	3 ROOM	601	ANG MO KIO AVE 5	01 TO 03	67.0	New Generation	1980	

In [130…

```
# Concat with new index

resale = pd.concat ([resale2015, resale2017], ignore_index= True)

resale
```

Out[130…

	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease_commen
0	2015-01	ANG MO KIO	3 ROOM	174	ANG MO KIO AVE 4	07 TO 09	60.0	Improved	
1	2015-01	ANG MO KIO	3 ROOM	541	ANG MO KIO AVE 10	01 TO 03	68.0	New Generation	
2	2015-01	ANG MO KIO	3 ROOM	163	ANG MO KIO AVE 4	01 TO 03	69.0	New Generation	
3	2015-01	ANG MO KIO	3 ROOM	446	ANG MO KIO AVE 10	01 TO 03	68.0	New Generation	

	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease_commen
4	2015-01	ANG MO KIO	3 ROOM	557	ANG MO KIO AVE 10	07 TO 09	68.0	New Generation	
...	...	...	...	...	...	...	...	...	
115043	2020-08	YISHUN	EXECUTIVE	611	YISHUN ST 61	04 TO 06	146.0	Maisonette	
115044	2020-08	YISHUN	EXECUTIVE	643	YISHUN ST 61	01 TO 03	142.0	Apartment	
115045	2020-08	YISHUN	EXECUTIVE	724	YISHUN ST 71	07 TO 09	142.0	Apartment	
115046	2020-08	YISHUN	EXECUTIVE	877	YISHUN ST 81	04 TO 06	145.0	Maisonette	
115047	2020-08	YISHUN	MULTI-GENERATION	666	YISHUN AVE 4	01 TO 03	164.0	Multi Generation	

115048 rows × 11 columns

In [131…

```
# data types

resale.dtypes
```

Out[131…

```
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
remaining_lease         object
resale_price           float64
dtype: object
```

In [132…

```
# non-nulls

resale.info()

# no missing data
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 115048 entries, 0 to 115047
Data columns (total 11 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   month                115048 non-null  object
 1   town                 115048 non-null  object
 2   flat_type            115048 non-null  object
 3   block                115048 non-null  object
 4   street_name          115048 non-null  object
 5   storey_range         115048 non-null  object
 6   floor_area_sqm       115048 non-null  float64
 7   flat_model           115048 non-null  object
 8   lease_commence_date  115048 non-null  int64
 9   remaining_lease      115048 non-null  object
 10  resale_price         115048 non-null  float64
```

```
dtypes: float64(2), int64(1), object(8)
memory usage: 9.7+ MB
```

In [133…

```python
# value counts for each column

string_columns = resale.select_dtypes(['object', 'float64']).columns
for col in string_columns:
    resale[col].value_counts()
```

Out[133…

```
2018-07    2539
2020-07    2462
2020-06    2441
2020-08    2438
2019-10    2206
           ...
2015-02    1151
2017-02    1085
2018-01    1072
2020-04     424
2020-05     363
Name: month, Length: 68, dtype: int64
```

Out[133…

```
SENGKANG           8787
JURONG WEST        8717
WOODLANDS          8500
TAMPINES           7705
YISHUN             7621
BEDOK              6645
PUNGGOL            6460
HOUGANG            5614
ANG MO KIO         5360
CHOA CHU KANG      5025
BUKIT PANJANG      4483
BUKIT MERAH        4390
BUKIT BATOK        4375
TOA PAYOH          3658
PASIR RIS          3506
KALLANG/WHAMPOA    3364
SEMBAWANG          3072
QUEENSTOWN         3038
GEYLANG            2904
CLEMENTI           2667
JURONG EAST        2575
SERANGOON          2383
BISHAN             2174
CENTRAL AREA       1046
MARINE PARADE       677
BUKIT TIMAH         302
Name: town, dtype: int64
```

Out[133…

```
4 ROOM              47423
3 ROOM              28888
5 ROOM              28041
EXECUTIVE            9054
2 ROOM               1551
1 ROOM                 51
MULTI-GENERATION       40
Name: flat_type, dtype: int64
```

Out[133…

```
2       430
8       387
1       373
101     337
114     330
       ...
635C      1
860A      1
```

```
640A      1
496F      1
868B      1
Name: block, Length: 2446, dtype: int64
```

Out[133…

```
YISHUN RING RD        1823
BEDOK RESERVOIR RD    1413
ANG MO KIO AVE 10     1294
ANG MO KIO AVE 3      1212
PUNGGOL FIELD         1175
                      ...
KRETA AYER RD            3
SEMBAWANG WAY            2
SUMANG WALK              2
TAMPINES ST 86           1
HENDERSON RD             1
Name: street_name, Length: 548, dtype: int64
```

Out[133…

```
04 TO 06    27063
07 TO 09    24674
10 TO 12    21807
01 TO 03    21039
13 TO 15    10461
16 TO 18     4599
19 TO 21     1953
22 TO 24     1429
25 TO 27      771
28 TO 30      489
34 TO 36      211
31 TO 33      204
37 TO 39      199
40 TO 42      105
46 TO 48       21
43 TO 45       14
49 TO 51        9
Name: storey_range, dtype: int64
```

Out[133…

```
67.0     7395
110.0    6220
92.0     5681
104.0    4699
93.0     4592
         ...
241.0       1
259.0       1
280.0       1
199.0       1
188.0       1
Name: floor_area_sqm, Length: 166, dtype: int64
```

Out[133…

```
Model A                   36596
Improved                  28954
New Generation            17091
Premium Apartment         11934
Simplified                 5156
Apartment                  4575
Standard                   3370
Maisonette                 3342
Model A2                   1603
DBSS                       1464
Type S1                     268
Model A-Maisonette          201
Adjoined flat               199
Type S2                     126
Terrace                      67
Multi Generation             40
Premium Apartment Loft       29
Improved-Maisonette          18
Premium Maisonette           12
```

```
2-room                          3
Name: flat_model, dtype: int64
```

Out[133…

```
68                     2328
67                     1857
69                     1635
71                     1547
84                     1423
                       ...
66 years 0 months         1
48                        1
97 years 01 month         1
97 years 09 months        1
45 years 04 months        1
Name: remaining_lease, Length: 672, dtype: int64
```

Out[133…

```
380000.0    1730
400000.0    1705
420000.0    1584
350000.0    1583
360000.0    1464
            ...
950888.0       1
632088.0       1
971888.0       1
585500.0       1
515515.0       1
Name: resale_price, Length: 2849, dtype: int64
```

In [134…

```
# check for duplicates

resale.duplicated()
resale.duplicated().sum()

# show duplicates

resale[resale.duplicated(keep = False)]
```

Out[134…

```
0         False
1         False
2         False
3         False
4         False
          ...
115043    False
115044    False
115045    False
115046    False
115047    False
Length: 115048, dtype: bool
```

Out[134…

```
233
```

Out[134…

	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease
660	2015-01	KALLANG/WHAMPOA	3 ROOM	57	GEYLANG BAHRU	16 TO 18	65.0	Improved	
661	2015-01	KALLANG/WHAMPOA	3 ROOM	57	GEYLANG BAHRU	16 TO 18	65.0	Improved	
2165	2015-02	TAMPINES	3 ROOM	403	TAMPINES ST 41	07 TO 09	69.0	Improved	
2166	2015-02	TAMPINES	3 ROOM	403	TAMPINES ST 41	07 TO 09	69.0	Improved	

	month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease
3895	2015-04	BEDOK	4 ROOM	701	BEDOK RESERVOIR RD	10 TO 12	93.0	New Generation	
...	...	...	...	...	...	...	...	...	
112958	2020-08	BUKIT MERAH	2 ROOM	45	TELOK BLANGAH DR	04 TO 06	45.0	Improved	
113392	2020-08	HOUGANG	3 ROOM	115	HOUGANG AVE 1	01 TO 03	67.0	New Generation	
113395	2020-08	HOUGANG	3 ROOM	115	HOUGANG AVE 1	01 TO 03	67.0	New Generation	
113652	2020-08	JURONG WEST	5 ROOM	183D	BOON LAY AVE	04 TO 06	112.0	Improved	
113654	2020-08	JURONG WEST	5 ROOM	183D	BOON LAY AVE	04 TO 06	112.0	Improved	

466 rows × 11 columns

In [135…

```
# drop duplicates

resale = resale.drop_duplicates()
```

In [136…

```
# add column price per sqf

resale['price_psf'] = resale.resale_price / (resale.floor_area_sqm * 10.7639 )
```

```
C:\Users\baoqu\AppData\Local\Temp/ipykernel_8912/2632934240.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_gu
ide/indexing.html#returning-a-view-versus-a-copy
  resale['price_psf'] = resale.resale_price / (resale.floor_area_sqm * 10.7639 )
```

In [137…

```
# remaining lease ( lease left at the year 2020)

resale['lease_left'] = 99 - (2020 - (resale.lease_commence_date))
```

```
C:\Users\baoqu\AppData\Local\Temp/ipykernel_8912/1014363545.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_gu
ide/indexing.html#returning-a-view-versus-a-copy
  resale['lease_left'] = 99 - (2020 - (resale.lease_commence_date))
```

In [138…

```
# drop block

resale = resale.drop('block', axis=1)

# drop street name

resale = resale.drop('street_name', axis=1)
```

```
# drop flat model

resale = resale.drop('flat_model', axis=1)

# drop lease commence date

resale = resale.drop(['lease_commence_date', 'remaining_lease'] , axis=1)

resale
```

Out[138…

	month	town	flat_type	storey_range	floor_area_sqm	resale_price	price_psf	lease_left
0	2015-01	ANG MO KIO	3 ROOM	07 TO 09	60.0	255000.0	394.838302	65
1	2015-01	ANG MO KIO	3 ROOM	01 TO 03	68.0	275000.0	375.711187	60
2	2015-01	ANG MO KIO	3 ROOM	01 TO 03	69.0	285000.0	383.730319	59
3	2015-01	ANG MO KIO	3 ROOM	01 TO 03	68.0	290000.0	396.204525	58
4	2015-01	ANG MO KIO	3 ROOM	07 TO 09	68.0	290000.0	396.204525	59
...	...	...	...	...	...	...	...	...
115043	2020-08	YISHUN	EXECUTIVE	04 TO 06	146.0	598000.0	380.521039	66
115044	2020-08	YISHUN	EXECUTIVE	01 TO 03	142.0	510000.0	333.666171	66
115045	2020-08	YISHUN	EXECUTIVE	07 TO 09	142.0	555000.0	363.107304	65
115046	2020-08	YISHUN	EXECUTIVE	04 TO 06	145.0	639888.0	409.983434	66
115047	2020-08	YISHUN	MULTI-GENERATION	01 TO 03	164.0	725000.0	410.699812	66

114815 rows × 8 columns

In [139…

```
# Ordinal Encoding town to district number (as integer data for machine learning)
# in general lower district numbers are closer to central - which affects psf

resale ['town'] = resale.town.map( {'SENGKANG' : 19,
                                    'JURONG WEST' : 22,
                                    'WOODLANDS' : 25,
                                    'TAMPINES' : 18,
                                    'YISHUN' : 27,
                                    'BEDOK' : 16,
                                    'PUNGGOL' : 19,
                                    'HOUGANG' : 19,
                                    'ANG MO KIO' : 20,
                                    'CHOA CHU KANG' : 23,
                                    'BUKIT PANJANG' : 23,
                                    'BUKIT MERAH' : 3,
                                    'BUKIT BATOK' : 23,
                                    'TOA PAYOH' : 12,
                                    'PASIR RIS' : 17,
                                    'KALLANG/WHAMPOA' : 14,
                                    'SEMBAWANG' : 27,
```

```
                                        'QUEENSTOWN' : 3,
                                        'GEYLANG' : 14,
                                        'CLEMENTI' : 21,
                                        'JURONG EAST' : 22,
                                        'SERANGOON' : 8,
                                        'BISHAN' : 20,
                                        'CENTRAL AREA' : 1,
                                        'MARINE PARADE' : 15,
                                        'BUKIT TIMAH' : 21 })

resale
```

Out[139…

	month	town	flat_type	storey_range	floor_area_sqm	resale_price	price_psf	lease_left
0	2015-01	20	3 ROOM	07 TO 09	60.0	255000.0	394.838302	65
1	2015-01	20	3 ROOM	01 TO 03	68.0	275000.0	375.711187	60
2	2015-01	20	3 ROOM	01 TO 03	69.0	285000.0	383.730319	59
3	2015-01	20	3 ROOM	01 TO 03	68.0	290000.0	396.204525	58
4	2015-01	20	3 ROOM	07 TO 09	68.0	290000.0	396.204525	59
...	...	...	...	...	...	...	...	...
115043	2020-08	27	EXECUTIVE	04 TO 06	146.0	598000.0	380.521039	66
115044	2020-08	27	EXECUTIVE	01 TO 03	142.0	510000.0	333.666171	66
115045	2020-08	27	EXECUTIVE	07 TO 09	142.0	555000.0	363.107304	65
115046	2020-08	27	EXECUTIVE	04 TO 06	145.0	639888.0	409.983434	66
115047	2020-08	27	MULTI-GENERATION	01 TO 03	164.0	725000.0	410.699812	66

114815 rows × 8 columns

In [140…

```
# Ordinal Encoding flat types

resale ['flat_type'] = resale.flat_type.map( {'1 ROOM' : 1,
                                              '2 ROOM' : 2,
                                              '3 ROOM' : 3,
                                              '4 ROOM' : 4,
                                              '5 ROOM' : 5,
                                              'EXECUTIVE' : 6,
                                              'MULTI-GENERATION' : 7 })

# Ordinal Encoding storey range

resale ['storey_range'] = resale.storey_range.map( {'01 TO 03' : 1,
                                                    '04 TO 06' : 4,
                                                    '07 TO 09' : 7,
                                                    '10 TO 12' : 10,
                                                    '13 TO 15' : 13,
                                                    '16 TO 18' : 16,
                                                    '19 TO 21' : 19,
                                                    '22 TO 24' : 22,
                                                    '25 TO 27' : 25,
                                                    '28 TO 30' : 28,
                                                    '31 TO 33' : 31,
                                                    '34 TO 36' : 34,
                                                    '37 TO 39' : 37,
                                                    '40 TO 42' : 40,
                                                    '43 TO 45' : 43,
                                                    '46 TO 48' : 46,
```

```
                                                    '49 TO 51' : 49 })

resale
resale.dtypes
```

Out[140…

	month	town	flat_type	storey_range	floor_area_sqm	resale_price	price_psf	lease_left
0	2015-01	20	3	7	60.0	255000.0	394.838302	65
1	2015-01	20	3	1	68.0	275000.0	375.711187	60
2	2015-01	20	3	1	69.0	285000.0	383.730319	59
3	2015-01	20	3	1	68.0	290000.0	396.204525	58
4	2015-01	20	3	7	68.0	290000.0	396.204525	59
...	...	...	...	...	...	...	...	...
115043	2020-08	27	6	4	146.0	598000.0	380.521039	66
115044	2020-08	27	6	1	142.0	510000.0	333.666171	66
115045	2020-08	27	6	7	142.0	555000.0	363.107304	65
115046	2020-08	27	6	4	145.0	639888.0	409.983434	66
115047	2020-08	27	7	1	164.0	725000.0	410.699812	66

114815 rows × 8 columns

Out[140…

```
month               object
town                 int64
flat_type            int64
storey_range         int64
floor_area_sqm     float64
resale_price       float64
price_psf          float64
lease_left           int64
dtype: object
```

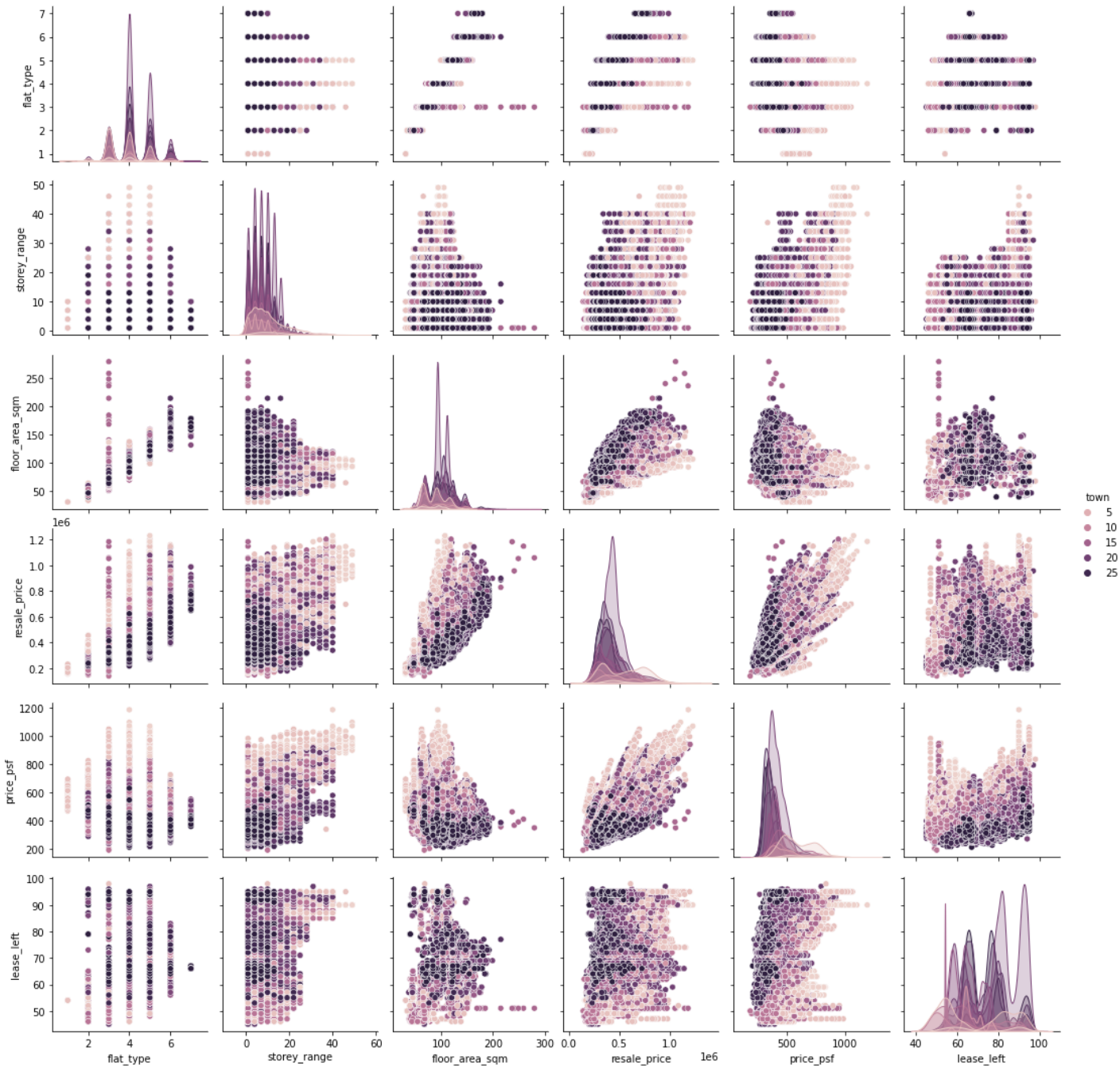
In [141…

```
# looking for relationships (Exploratory Data Analysis)
# pairplots with town

sns.pairplot (resale,hue = 'town')
```

Out[141…

```
<seaborn.axisgrid.PairGrid at 0x2d5ebf2eac0>
```



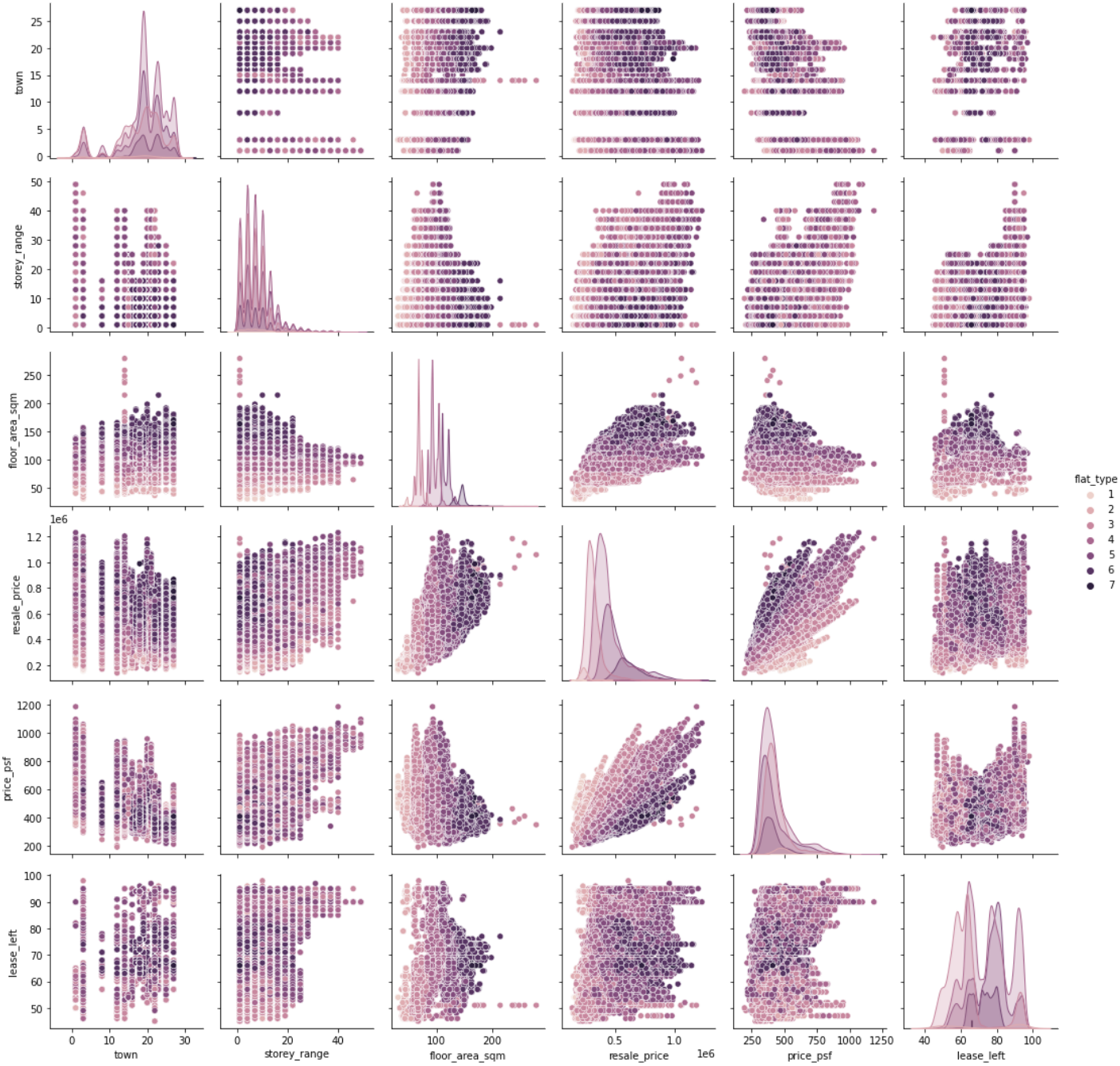
In [142…

```
# pairplots with flat types

sns.pairplot (resale,hue = 'flat_type')
```

Out[142…

```
<seaborn.axisgrid.PairGrid at 0x2d5806c5fa0>
```



In [143…

```
# pairplots with storey range

sns.pairplot (resale,hue = 'storey_range')
```

Out[143…

```
<seaborn.axisgrid.PairGrid at 0x2d5806cacd0>
```

In [144…

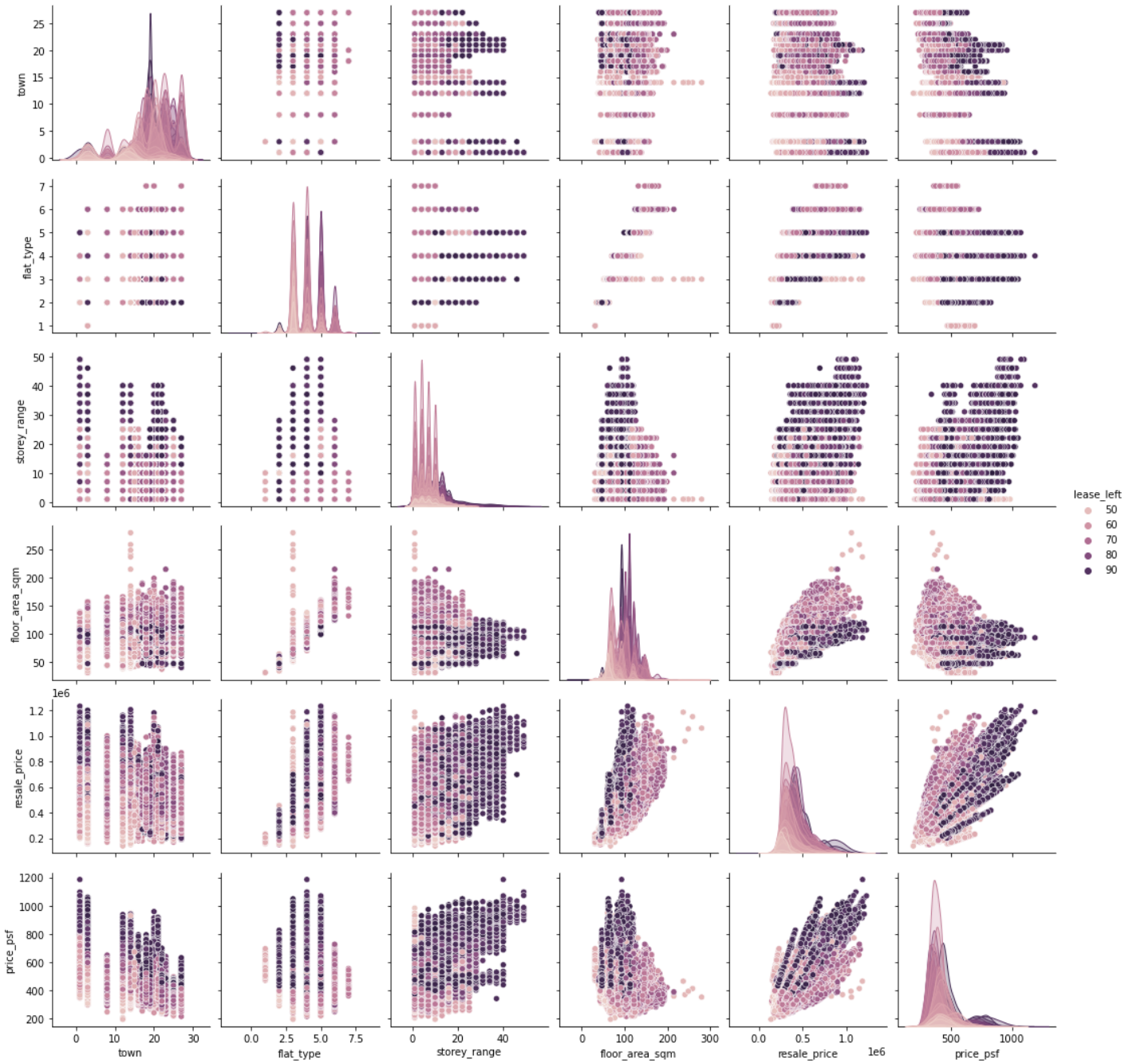
```
# pairplots with lease left

sns.pairplot (resale,hue = 'lease_left')

# no definitative feature that stands out as strong indicator affecting psf
```

Out[144…

```
<seaborn.axisgrid.PairGrid at 0x2d59717fc40>
```



In [145…

```
# Part 2 - Linear Regression & train_test_split

from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score
from sklearn.model_selection import train_test_split

# define adj_r2

def adjusted_r2_score(y, preds, X):
    return 1 - (len(y)-1)/(len(y)-len(X.columns)-1)*(1-r2_score(y, preds))
```

In [146…

```
# Select X and y

y = resale.pop('price_psf')
X = resale.select_dtypes(['int', 'float64'])

print(y.shape)
print(X.shape)
```

```
(114815,)
(114815, 6)
```

In [147…

```python
# split the data to train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=0)

print (X_train.shape)
print (X_test.shape)
print (y_train.shape)
print (y_test.shape)
```

```
(86111, 6)
(28704, 6)
(86111,)
(28704,)
```

In [148…

```python
# train_predict model >> instantiate / fit / predict / evaluate

# instantiate
lm = LinearRegression()

# fit
lm.fit(X_train, y_train)

# predict
predictions = lm.predict (X_test)

print (lm.intercept_)
print (lm.coef_)
```

Out[148…

```
LinearRegression()

418.9273753135438
[-1.32799113e+00 -2.47034950e+00  3.70407831e-01 -3.87807509e+00
  8.64121265e-04  4.74017725e-01]
```

In [149…

```python
# plot predictions vs y_test (answer)

plt.figure(figsize=(10,8))
plt.scatter(predictions, y_test, s=30, c='r', marker='+')
plt.xlabel('Predicted price_psf')
plt.ylabel('Actual price_psf')
```

Out[149…

```
<Figure size 720x576 with 0 Axes>
```

Out[149…

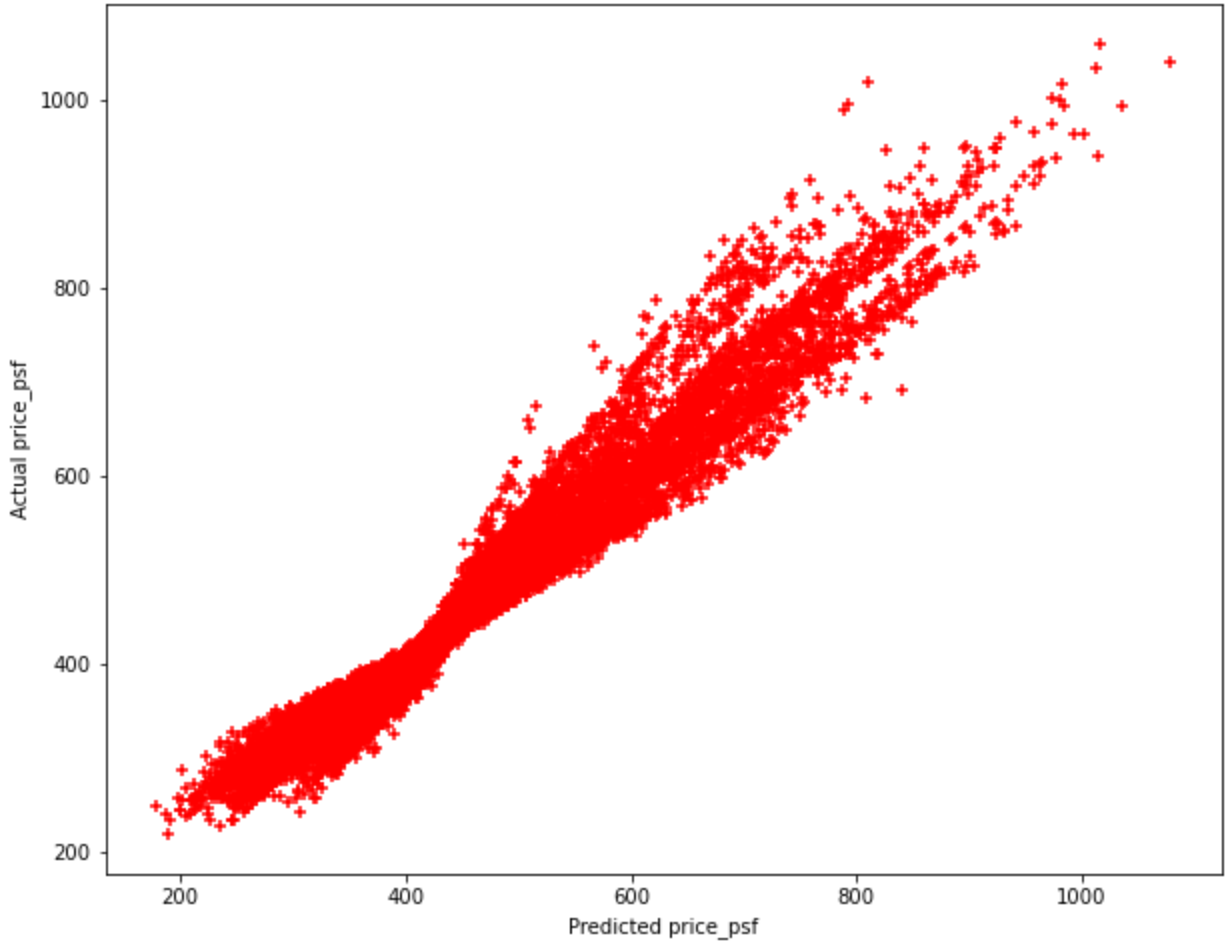
```
<matplotlib.collections.PathCollection at 0x2d5aee45130>
```

Out[149…

```
Text(0.5, 0, 'Predicted price_psf')
```

Out[149…

```
Text(0, 0.5, 'Actual price_psf')
```



In [150…

```
# evaluate model

mse = mean_squared_error(y_test, predictions)
r2 = r2_score(y_test, predictions)
adj_r2 = adjusted_r2_score(y_test, predictions, X_test)

print('MSE:', mse)
print('R-squared:', r2)
print('Adjusted R-squared:', adj_r2)

# test results are good
```

```
MSE: 445.24128369392474
R-squared: 0.9616624575681333
Adjusted R-squared: 0.9616544419130268
```

In [ ]: