Exploratory Analysis Part 2

In part 1, we explored what factors could lead to the changes in price level of HDB prices. For the purposes of this exploration part 2, I have filtered HDBs that are purely residential. This is in order to create the bid rent curve for urban economics analysis purposes.

But before that it is important to ensure all the econometrics assumptions have been met. In this notebook, each assumption is explored and analysed.

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
sb.set()
from collections import Counter
import time
pd.options.mode.chained_assignment = None  # default='warn'
import statsmodels.formula.api as smf
import statsmodels.api as sm
```

In [2]:
```
residential = pd.read_csv('residential.csv', low_memory=False)
```

In [3]:
```
residential.columns
```

Out[3]:
```
Index(['month', 'town', 'flat_type', 'blk_no', 'street', 'storey_range',
       'floor_area_sqm', 'flat_model', 'lease_commence_date',
       'remaining_lease', 'resale_price', 'max_floor_lvl', 'year_completed',
       'residential', 'commercial', 'market_hawker', 'miscellaneous',
       'multistorey_carpark', 'precinct_pavilion', 'bldg_contract_town',
       'total_dwelling_units', '1room_sold', '2room_sold', '3room_sold',
       '4room_sold', '5room_sold', 'exec_sold', 'multigen_sold',
       'studio_apartment_sold', '1room_rental', '2room_rental', '3room_rental',
       'other_room_rental', 'building', 'addr', 'Postal', 'SUBZONE_NO',
       'SUBZONE_N', 'PLN_AREA_N', 'REGION_N', 'MRT_NAME', 'mahattan_distance',
       'mrt_cbd_dist', 'mrt_cbd_time', 'hdb_cbd_distance', 'hdb_cbd_time',
       'hdb_to_mrtdist', 'sgd_persqm', 'No_Bus_Stops', 'real_price',
       'real_price_persqm', 'lease_remaining'],
      dtype='object')
```

Linear Regression on all possible x values (numerical

In [4]:
```
X = residential[['floor_area_sqm','resale_price', 'max_floor_lvl',
       'total_dwelling_units', '1room_sold', '2room_sold', '3room_sold',
       '4room_sold', '5room_sold', 'exec_sold', 'multigen_sold',
       'studio_apartment_sold', '1room_rental', '2room_rental', '3room_rental',
       'other_room_rental', 'mahattan_distance',
       'mrt_cbd_dist', 'mrt_cbd_time', 'hdb_cbd_distance', 'hdb_cbd_time',
       'hdb_to_mrtdist', 'sgd_persqm', 'No_Bus_Stops', 'real_price', 'lease_remain
y = residential[['real_price_persqm']]
X_constant = sm.add_constant(X)
lr = sm.OLS(y, X_constant.astype(float)).fit()
print(lr.summary())
```

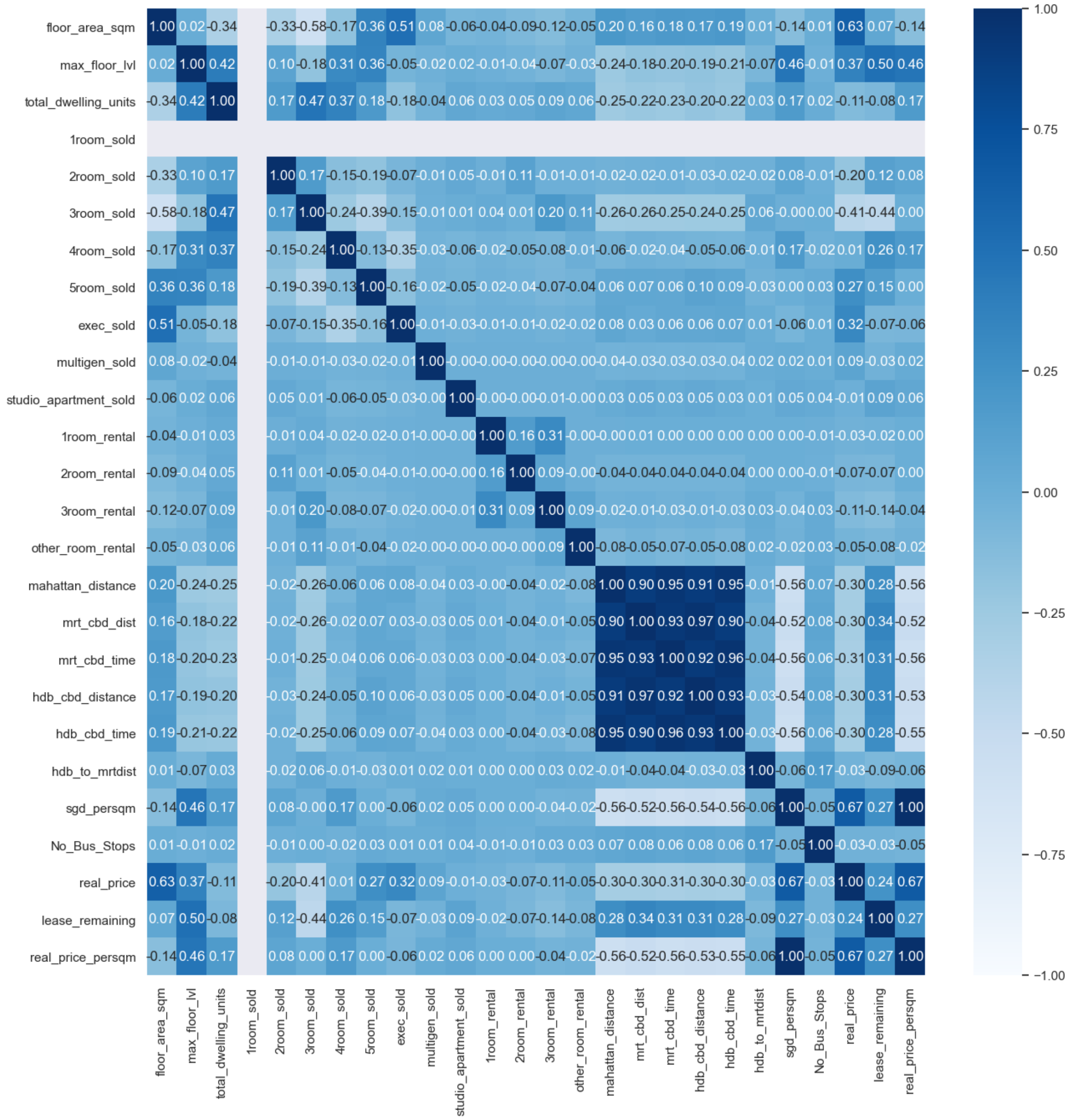
```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:      real_price_persqm   R-squared:                       1.000
Model:                            OLS   Adj. R-squared:                  1.000
Method:                 Least Squares   F-statistic:                 6.391e+07
Date:                Sun, 12 Mar 2023   Prob (F-statistic):               0.00
Time:                        18:05:24   Log-Likelihood:            -6.0363e+05
No. Observations:              158904   AIC:                         1.207e+06
Df Residuals:                  158879   BIC:                         1.208e+06
Df Model:                          24                                         
Covariance Type:            nonrobust                                         
==================================================================================
=======
                            coef    std err          t      P>|t|      [0.025
0.975]
----------------------------------------------------------------------------------
-------
const                    -9.5641      0.774    -12.352      0.000     -11.082
-8.047
floor_area_sqm            0.0235      0.007      3.326      0.001       0.010
0.037
resale_price             -0.0099   5.62e-06  -1766.151      0.000      -0.010
-0.010
max_floor_lvl            -0.0250      0.008     -3.226      0.001      -0.040
-0.010
total_dwelling_units     -0.0202      0.034     -0.595      0.552      -0.087
0.046
1room_sold             8.659e-13   2.31e-12      0.375      0.707   -3.66e-12
5.39e-12
2room_sold                0.0894      0.034      2.631      0.009       0.023
0.156
3room_sold                0.0261      0.034      0.766      0.444      -0.041
0.093
4room_sold                0.0166      0.034      0.487      0.626      -0.050
0.083
5room_sold                0.0261      0.034      0.769      0.442      -0.040
0.093
exec_sold                 0.0424      0.034      1.249      0.212      -0.024
0.109
multigen_sold            -0.0211      0.039     -0.544      0.586      -0.097
0.055
studio_apartment_sold     0.0857      0.034      2.506      0.012       0.019
0.153
1room_rental              0.0117      0.045      0.260      0.795      -0.076
0.100
2room_rental              0.0393      0.035      1.133      0.257      -0.029
0.107
3room_rental             -0.1775      0.058     -3.045      0.002      -0.292
-0.063
other_room_rental        -0.1588      0.373     -0.426      0.670      -0.889
0.572
mahattan_distance      1.299e-05   2.18e-05      0.595      0.552   -2.98e-05
5.58e-05
mrt_cbd_dist             -0.0004   3.52e-05    -12.349      0.000      -0.001
-0.000
mrt_cbd_time              0.2111      0.036      5.824      0.000       0.140
0.282
hdb_cbd_distance          0.0003   3.63e-05      7.371      0.000       0.000
0.000
hdb_cbd_time             -0.0022      0.036     -0.062      0.951      -0.072
0.067
hdb_to_mrtdist         1.956e-05   7.83e-05      0.250      0.803      -0.000
0.000
sgd_persqm                1.0151      0.000   6888.838      0.000       1.015
```

```
1.015
No_Bus_Stops                0.0127      0.008      1.502      0.133      -0.004
0.029
real_price                  0.0098   5.37e-06   1819.522      0.000       0.010
0.010
lease_remaining             0.0625      0.004     16.656      0.000       0.055
0.070
==============================================================================
Omnibus:                    13513.994   Durbin-Watson:                   0.458
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            89808.585
Skew:                           0.012   Prob(JB):                         0.00
Kurtosis:                       6.683   Cond. No.                     1.04e+16
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
[2] The smallest eigenvalue is 6.55e-16. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.
```

Removal of perfect collinearity

In [5]:
```
temp = residential[['floor_area_sqm', 'max_floor_lvl',
       'total_dwelling_units', '1room_sold', '2room_sold', '3room_sold',
       '4room_sold', '5room_sold', 'exec_sold', 'multigen_sold',
       'studio_apartment_sold', '1room_rental', '2room_rental', '3room_rental',
       'other_room_rental', 'mahattan_distance',
       'mrt_cbd_dist', 'mrt_cbd_time', 'hdb_cbd_distance', 'hdb_cbd_time',
       'hdb_to_mrtdist', 'sgd_persqm', 'No_Bus_Stops', 'real_price', 'lease_remain
f = plt.figure(figsize=(15,15))
sb.heatmap(temp.corr(),  vmin = -1, vmax = 1, annot = True, fmt=".2f", cmap ="Blue
```

Out[5]:
```
<AxesSubplot:>
```



High Variance inflation factors

The VIFs measures the extent to which multicollinearity has increased the variance of an estimated coefficient. It looks at the extent to which an explanatory variable can be explained by all other explanatory variables in the equations.
Reference: https://www.sfu.ca/~dsignori/buec333/lecture%2016.pdf

If VIF>5, R squared is more than 0.8. So we shall remove the variables

From the above heatmap, the following variables is easily explained by another:

1. manhattan distance
2. mrt_cbd_dist
3. mrt_cbd_time
4. hdb_cbd_dist
5. hdb_cbd_time
6. sgd_persqm

Among all of them, we shall keep the distance from hdb to the cbd (hdb_cbd_dist).

Factors that affect real_price_persqm

Using the heatmap above, we can roughly gauge which x variables have a `linear` relationship with the y variable of `real_price_persqm`. Anything above 0 is considered. Later, t-test and p-values will be used to test if they are significant.

1. floor_area_sqm
2. max_floor_lvl
3. total_dwelling_unit
4. 2room_sold
5. 4room_sold
6. exec_sold
7. multigen_sold
8. studio_apartment_sold
9. 3room_rental
10. other_room_rental
11. hdb_cbd_dist
12. hdb_to_mrtdist
13. No_Bus_Stops
14. Real price (surprisingly real price and real price per sqm is not highly correlated?)
15. lease_remaining

Finding x variables that are affecting other x variables in a non linear manner

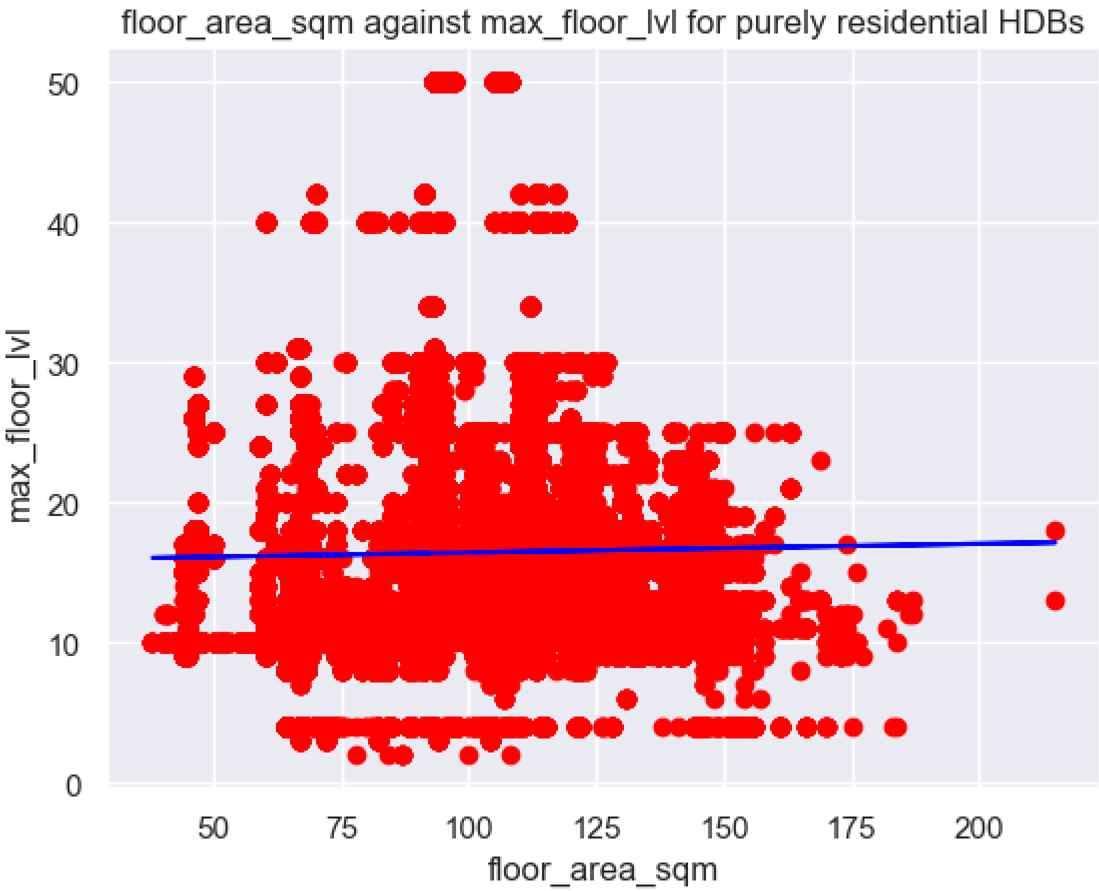
From the above heatmap, we could narrow down the x variables that affects y. We could also remove x variables affecting other x variables strongly in a linear manner.

With the remaining x variables:

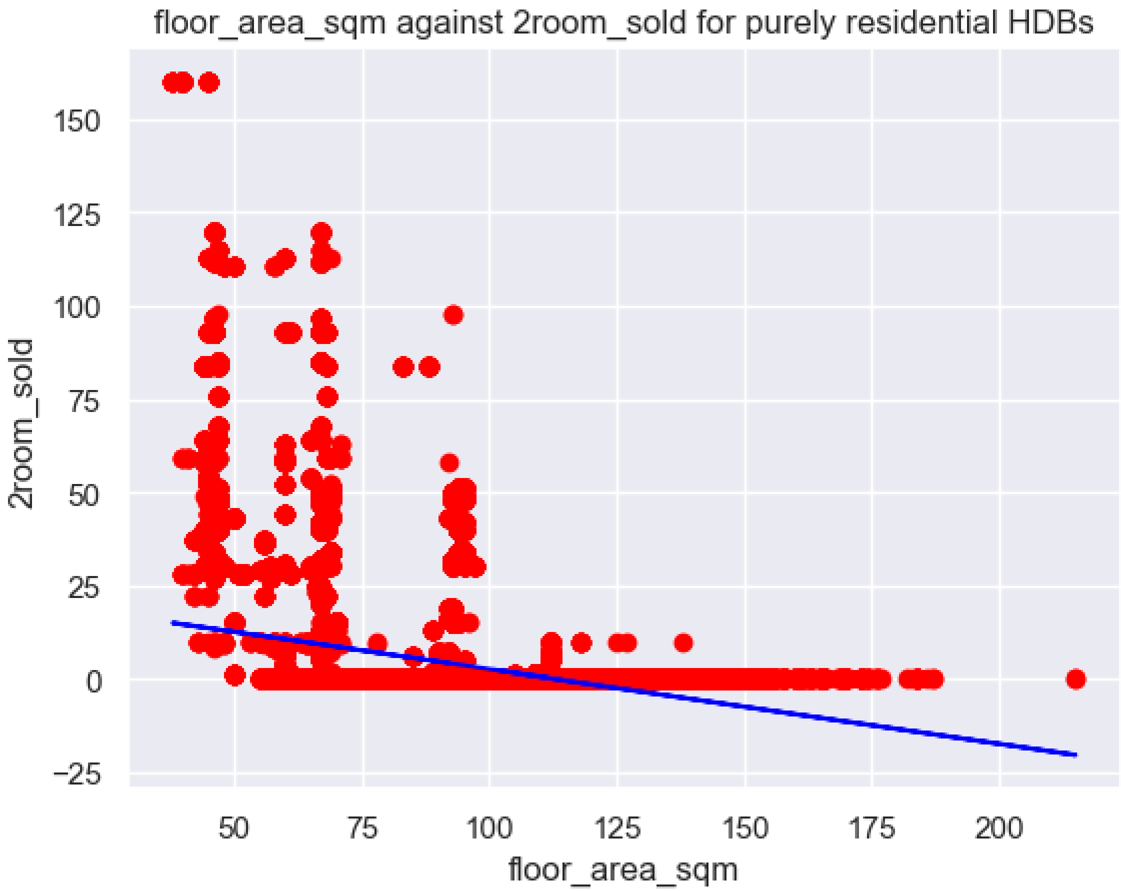
1. floor_area_sqm
2. max_floor_lvl
3. total_dwelling_unit
4. 2room_sold
5. 4room_sold
6. exec_sold
7. multigen_sold
8. studio_apartment_sold
9. 3room_rental
10. other_room_rental
11. hdb_cbd_dist
12. hdb_to_mrtdist
13. No_Bus_Stops
14. Real price (surprisingly real price and real price per sqm is not highly correlated?)
15. lease_remaining

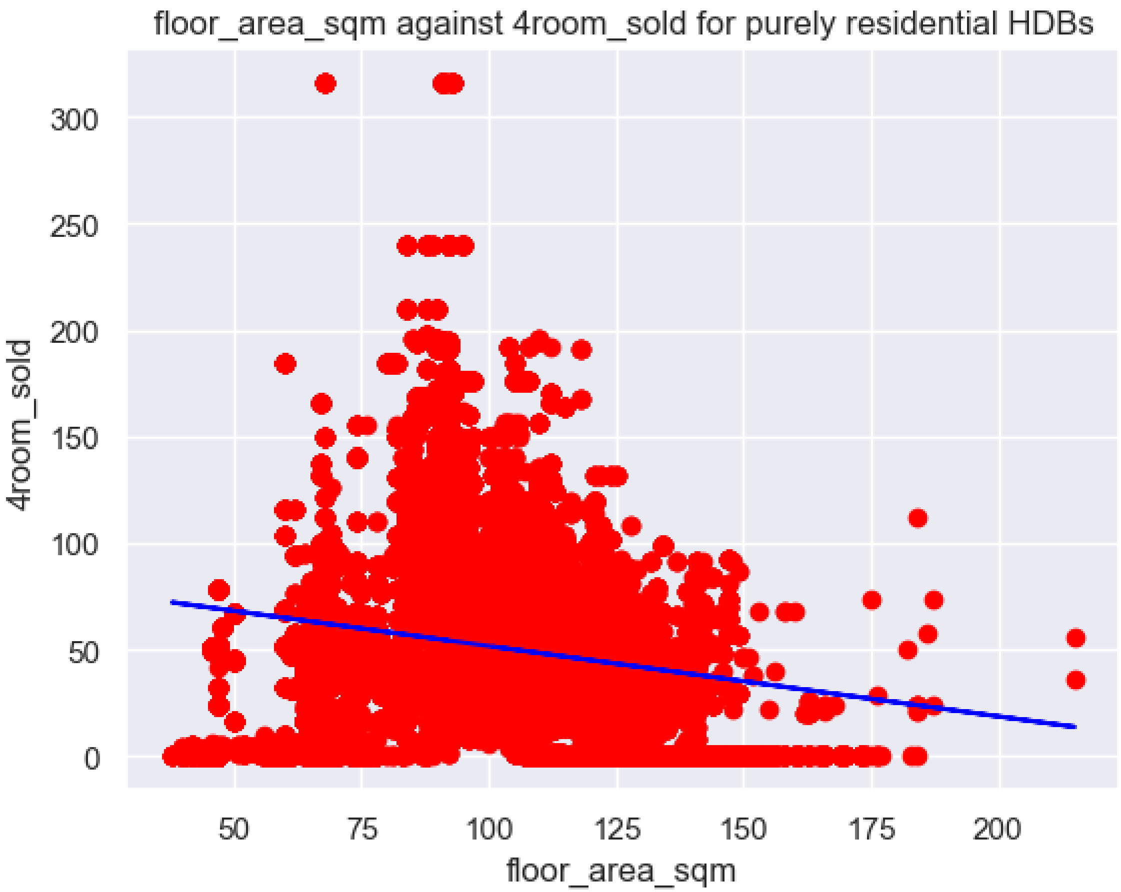
We need to ensure each x variable is not affected by multiple other x variables in a linear or non linear manner

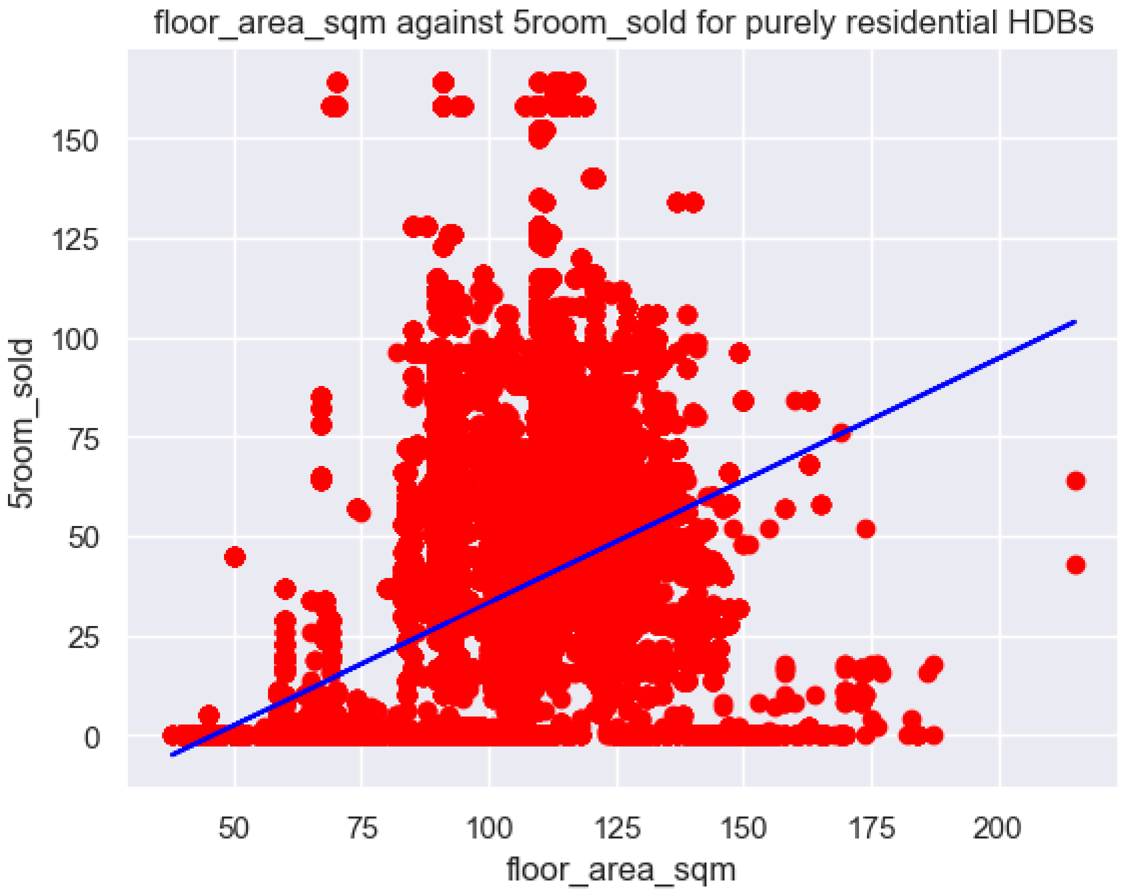
In [6]:
```
from sklearn.linear_model import LinearRegression

def printgraph(x:str,y:str):
    X=residential[[x]]
    Y=residential[[y]]

    titlestr = x + ' against ' + y + ' for purely residential HDBs'

    regressor = LinearRegression()
    regressor.fit(X,Y)
    y_pred = regressor.predict(X)
    plt.scatter(X, Y, color = 'red', )
    plt.plot(X, regressor.predict(X), color = 'blue')
    plt.title(titlestr)
    plt.xlabel(x)
    plt.ylabel(y)
    plt.show()

llist = ['floor_area_sqm','max_floor_lvl',
       'total_dwelling_units', '2room_sold',
       '4room_sold', '5room_sold', 'exec_sold', 'multigen_sold',
       'studio_apartment_sold', '3room_rental',
       'other_room_rental', 'hdb_cbd_distance',
       'hdb_to_mrtdist', 'No_Bus_Stops', 'real_price', 'lease_remaining']

for i in range(0, len(llist)-1):
    for j in range(i+1, len(llist)-1):
        printgraph(llist[i], llist[j])


# for i in range(0, len(llist)-1):
#     printgraph('4room_sold', llist[i])
```

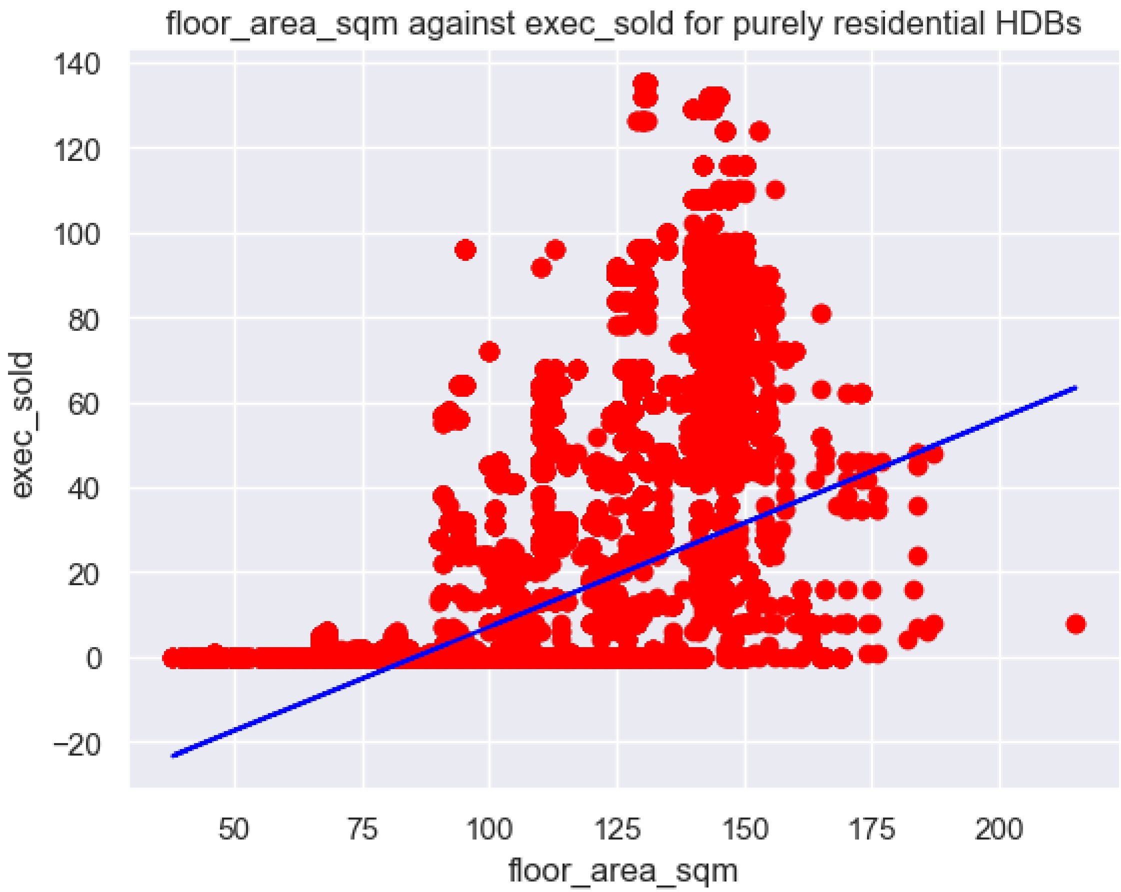

floor_area_sqm against max_floor_lvl for purely residential HDBs
max_floor_lvl
floor_area_sqm
0
10
20
30
40
50
50
75
100
125
150
175
200

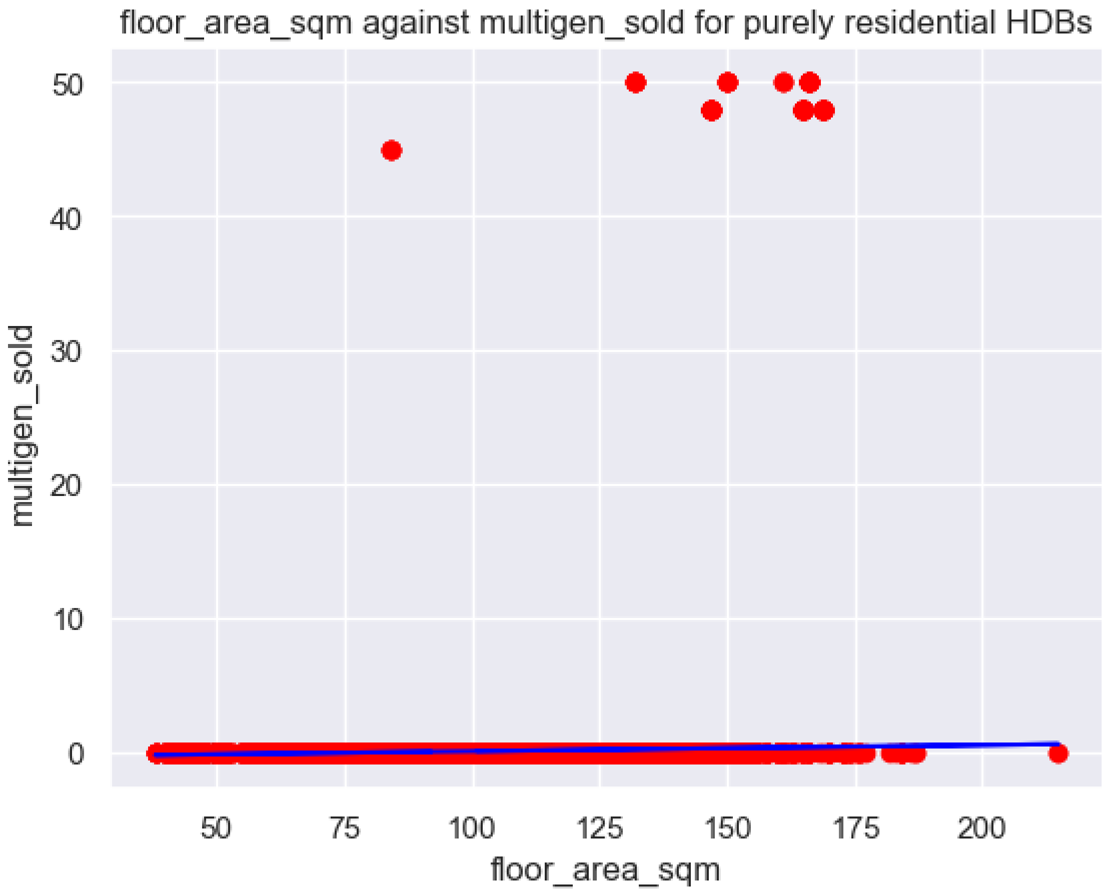

floor_area_sqm against total_dwelling_units for purely residential HDBs
total_dwelling_units
floor_area_sqm
0
50
100
150
200
250
300
350
50
75
100
125
150
175
200


floor_area_sqm against 2room_sold for purely residential HDBs
2room_sold
floor_area_sqm
150
125
100
75
50
25
0
−25
50
75
100
125
150
175
200

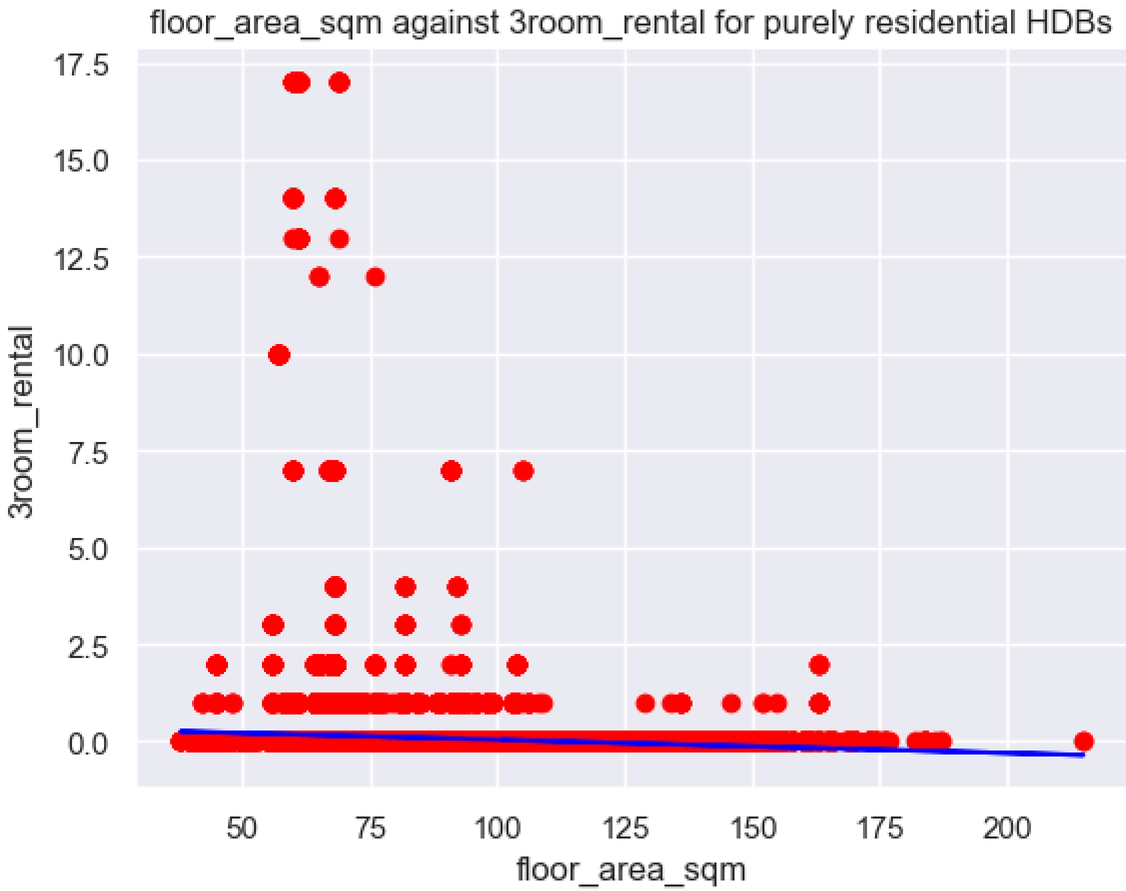

floor_area_sqm against 4room_sold for purely residential HDBs
4room_sold
floor_area_sqm
300
250
200
150
100
50
0
50
75
100
125
150
175
200


floor_area_sqm against 5room_sold for purely residential HDBs
5room_sold
0
25
50
75
100
125
150
50
75
100
125
150
175
200
floor_area_sqm

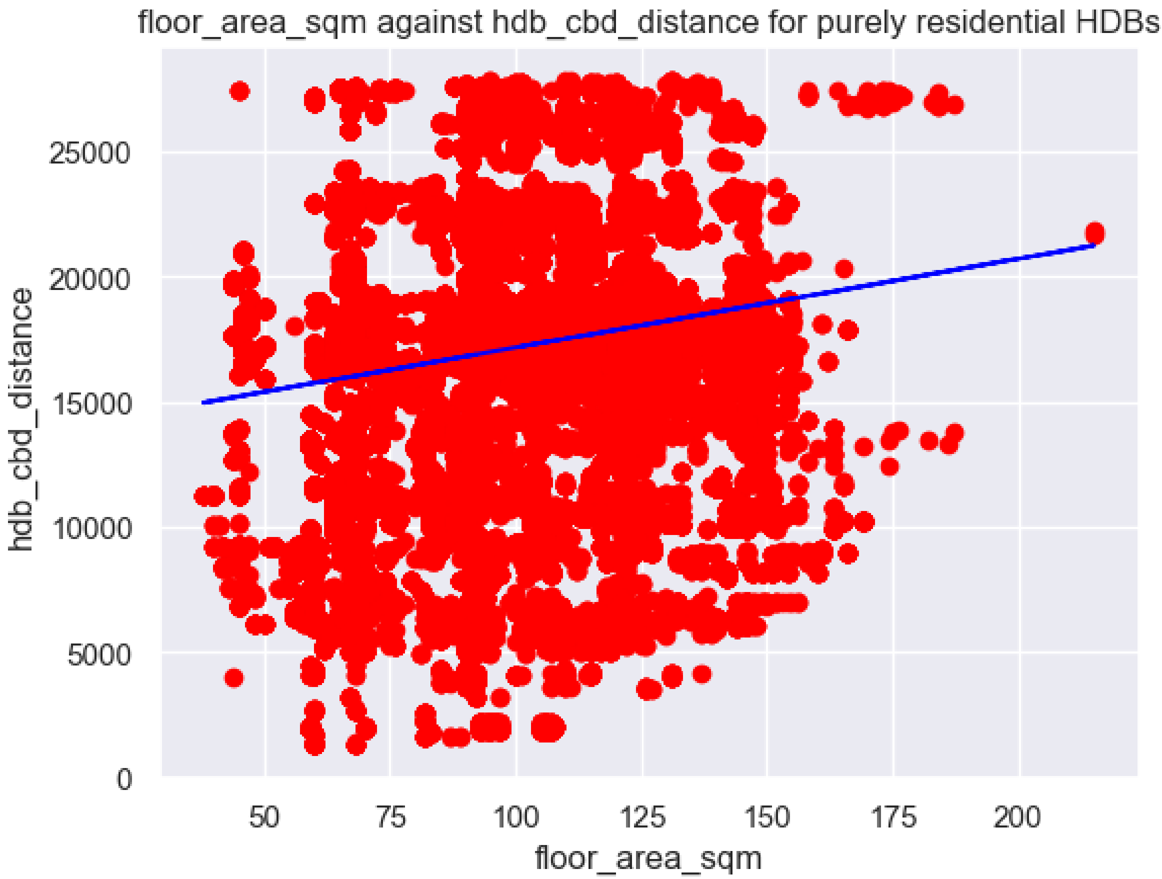

floor_area_sqm against exec_sold for purely residential HDBs
exec_sold
−20
0
20
40
60
80
100
120
140
50
75
100
125
150
175
200
floor_area_sqm

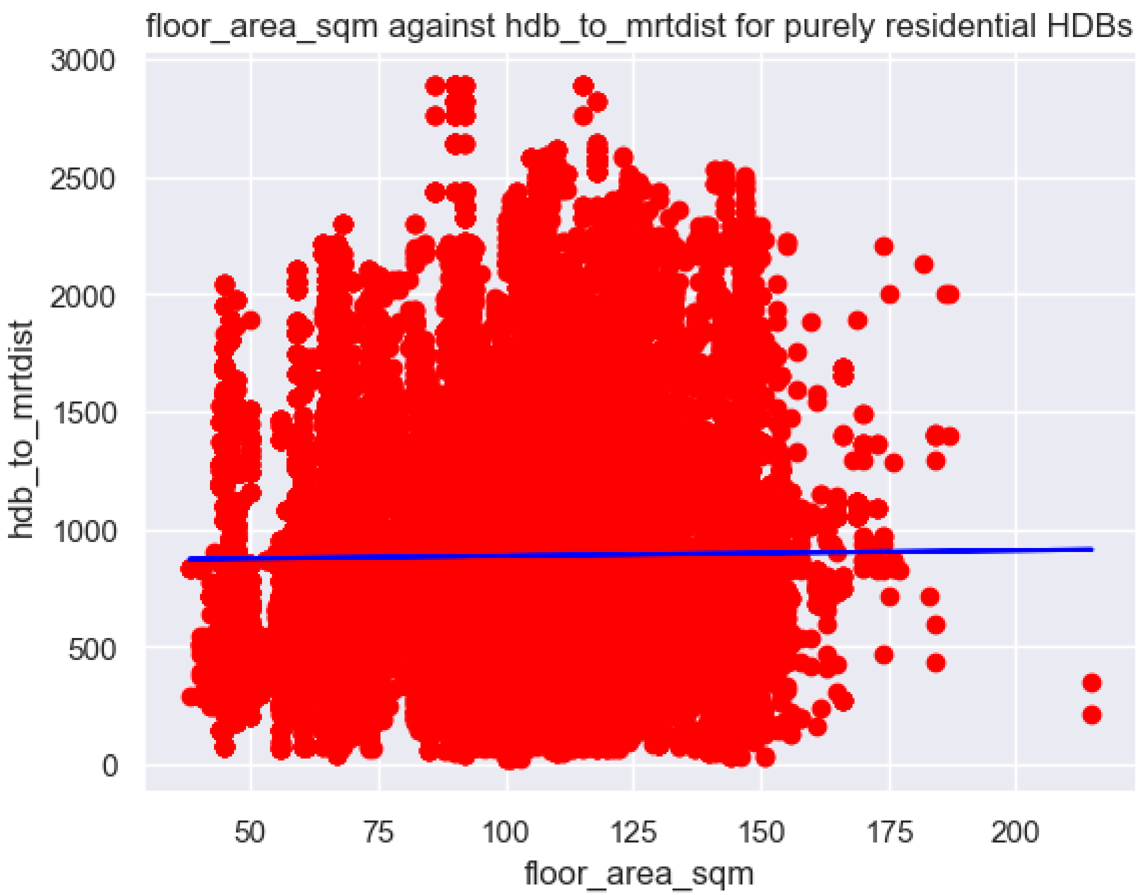

floor_area_sqm against multigen_sold for purely residential HDBs
multigen_sold
0
10
20
30
40
50
50
75
100
125
150
175
200
floor_area_sqm

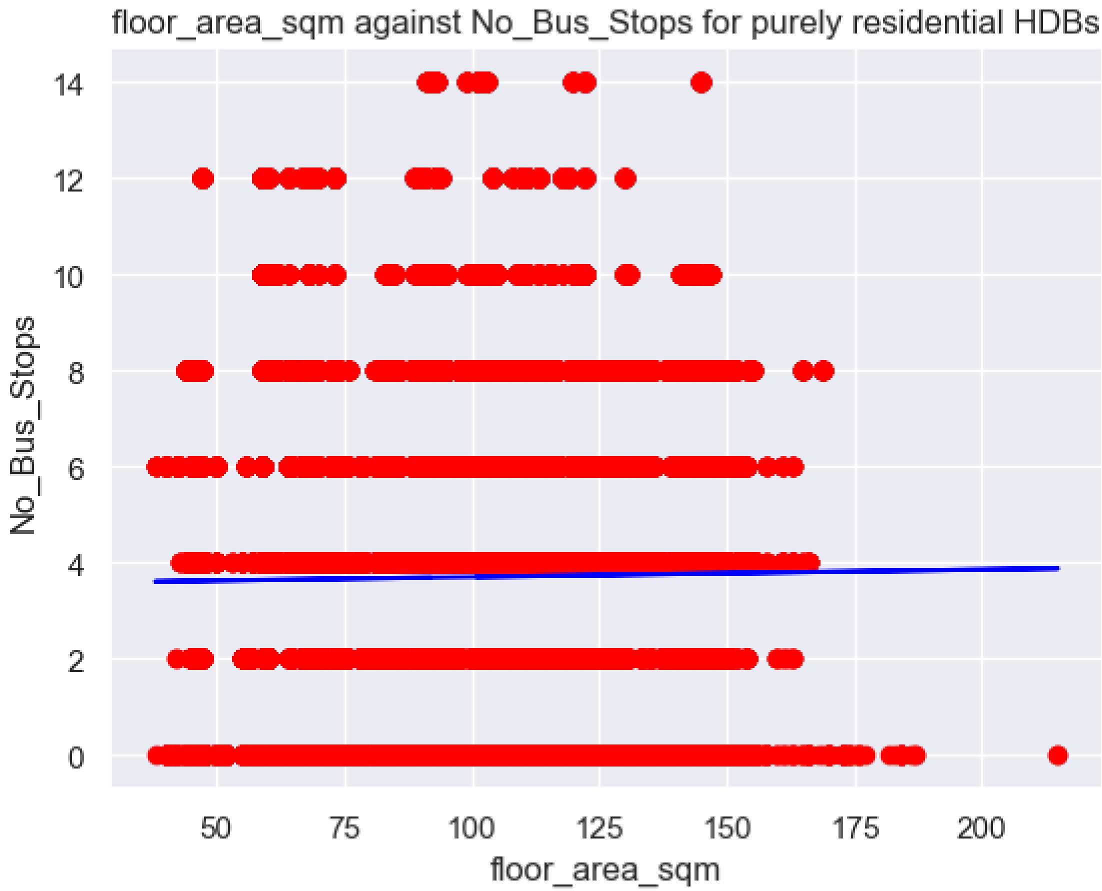

floor_area_sqm against studio_apartment_sold for purely residential HDBs
studio_apartment_sold
0
20
40
60
80
100
120
50
75
100
125
150
175
200
floor_area_sqm


floor_area_sqm against 3room_rental for purely residential HDBs
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
3room_rental
50
75
100
125
150
175
200
floor_area_sqm

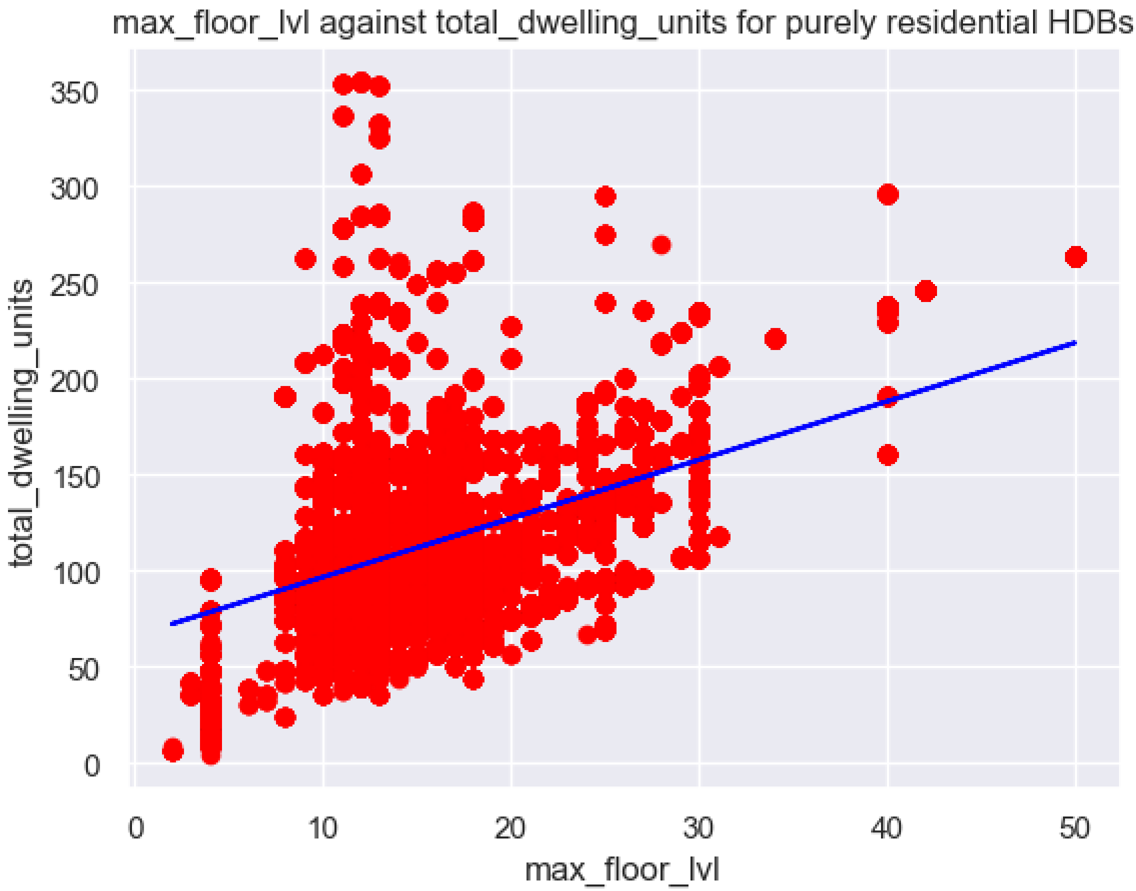

floor_area_sqm against other_room_rental for purely residential HDBs
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
other_room_rental
50
75
100
125
150
175
200
floor_area_sqm

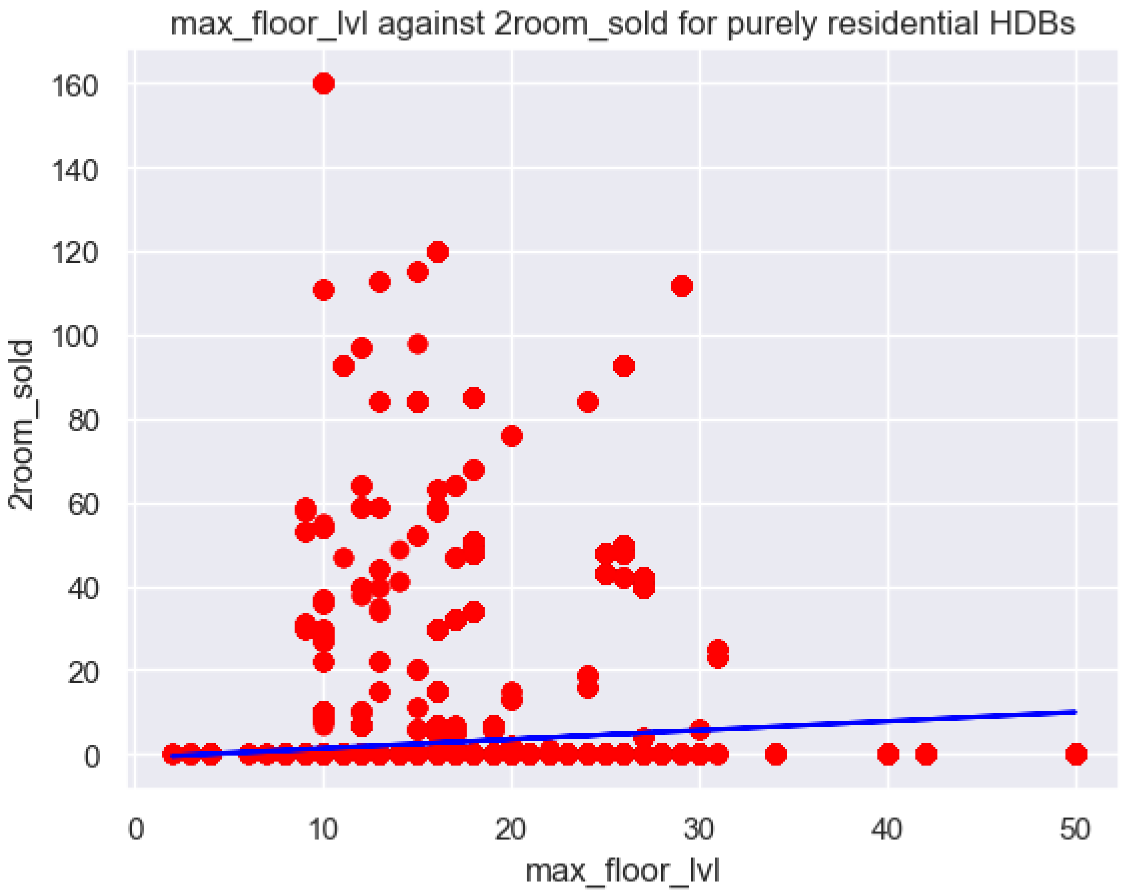

floor_area_sqm against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
0
5000
10000
15000
20000
25000
50
75
100
125
150
175
200
floor_area_sqm

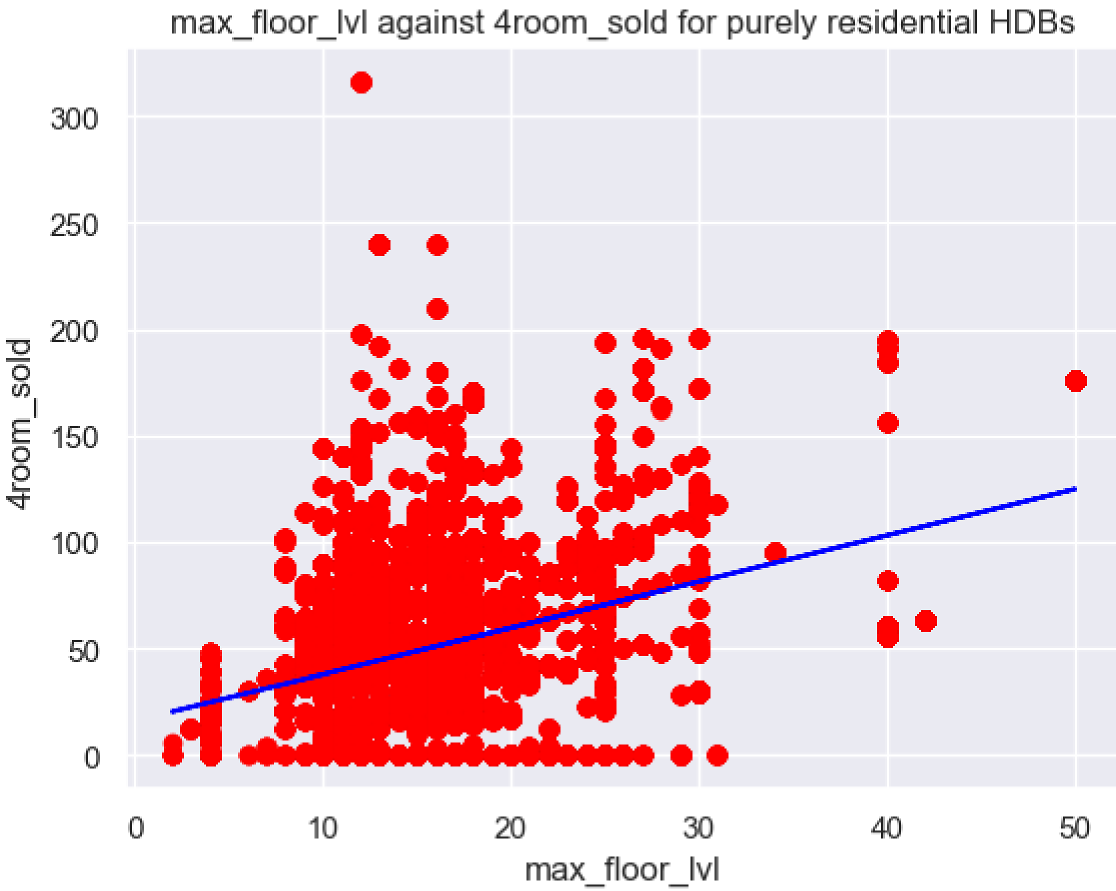

floor_area_sqm against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
0
500
1000
1500
2000
2500
3000
50
75
100
125
150
175
200
floor_area_sqm

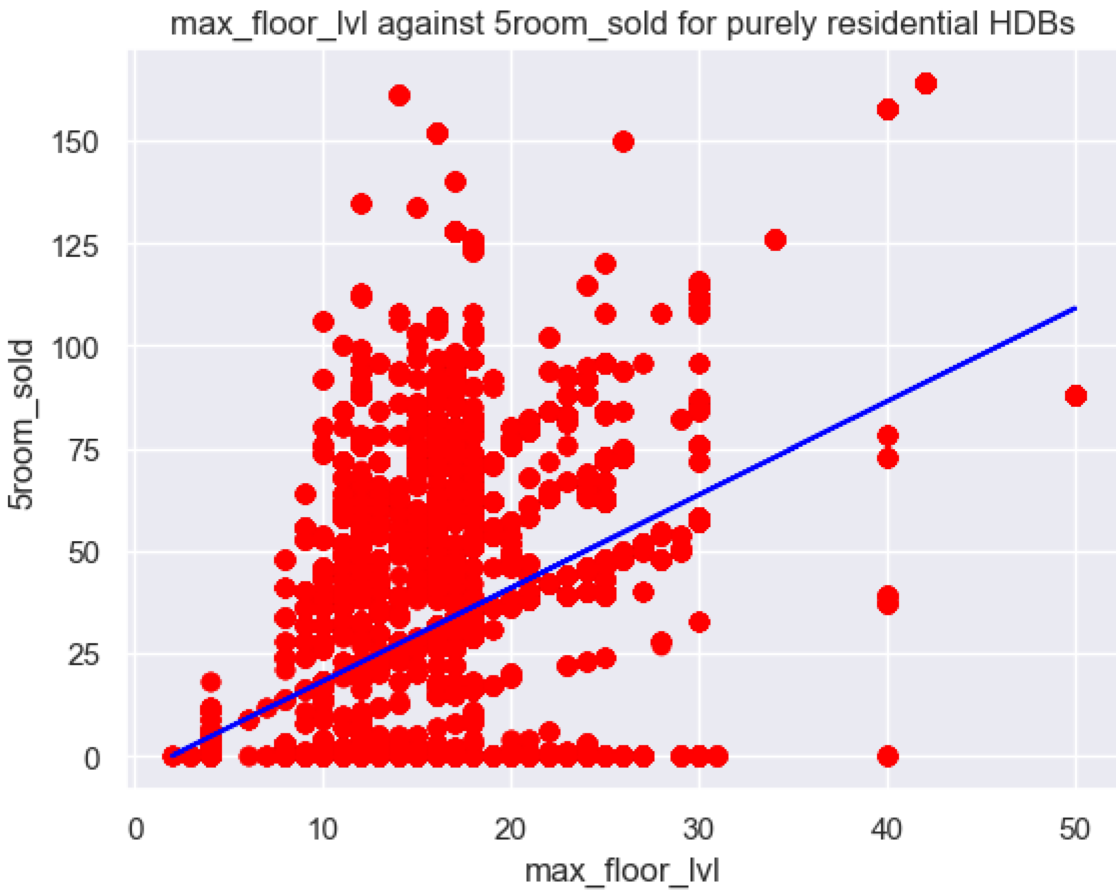

floor_area_sqm against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
0
2
4
6
8
10
12
14
50
75
100
125
150
175
200
floor_area_sqm

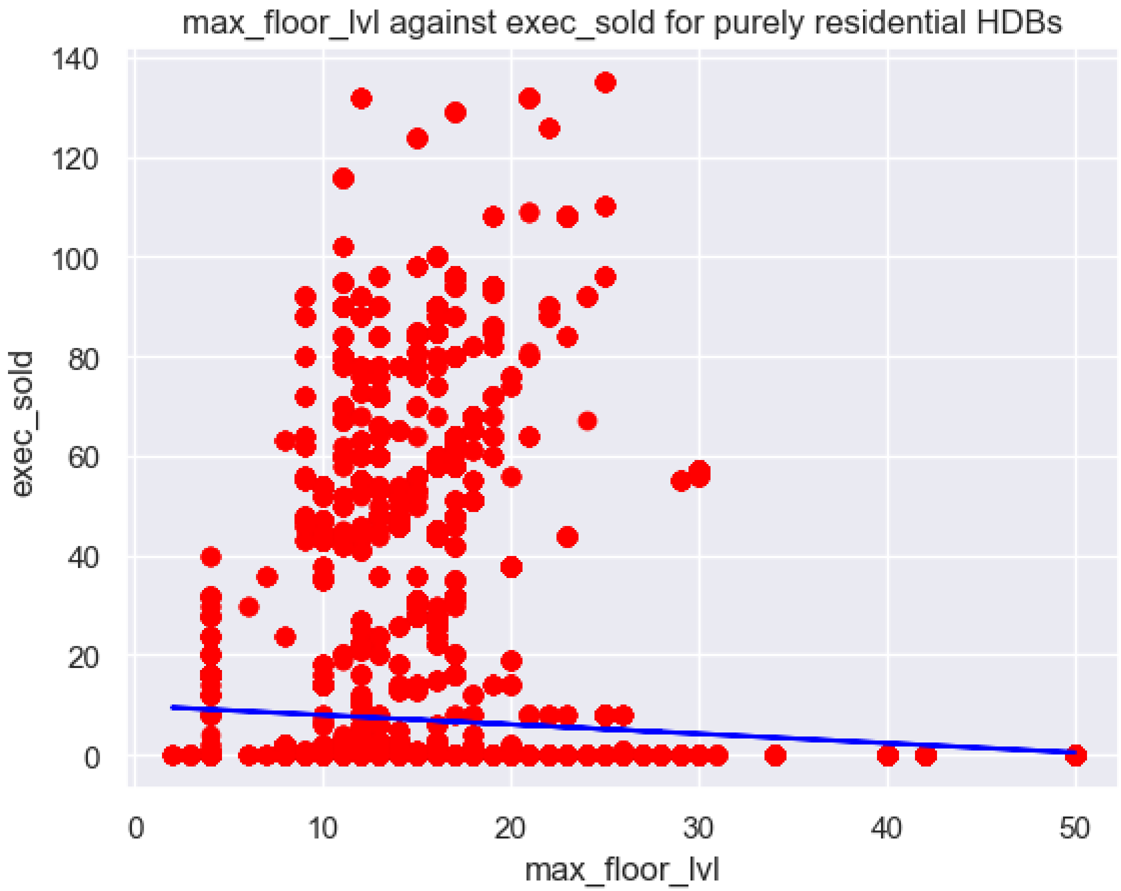

floor_area_sqm against real_price for purely residential HDBs
1e6
real_price
0.2
0.4
0.6
0.8
1.0
1.2
50
75
100
125
150
175
200
floor_area_sqm

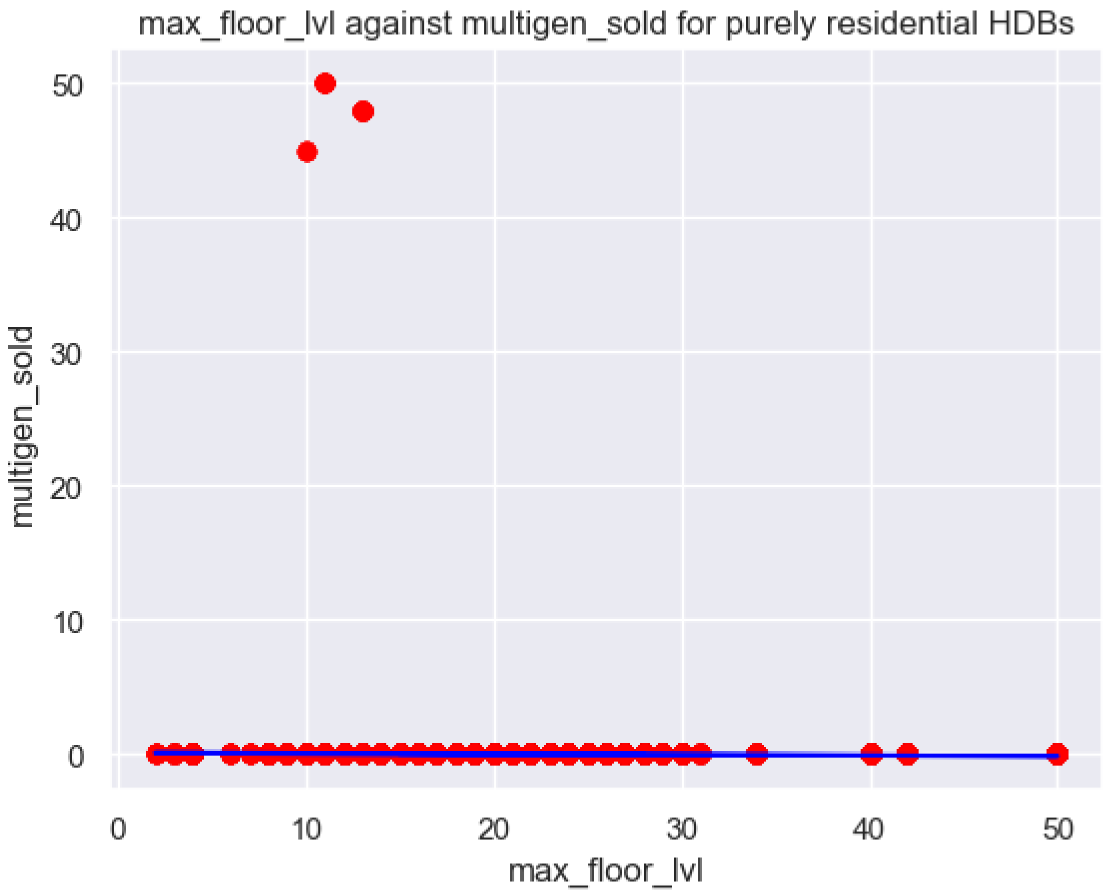

max_floor_lvl against total_dwelling_units for purely residential HDBs
total_dwelling_units
350
300
250
200
150
100
50
0
0
10
20
30
40
50
max_floor_lvl

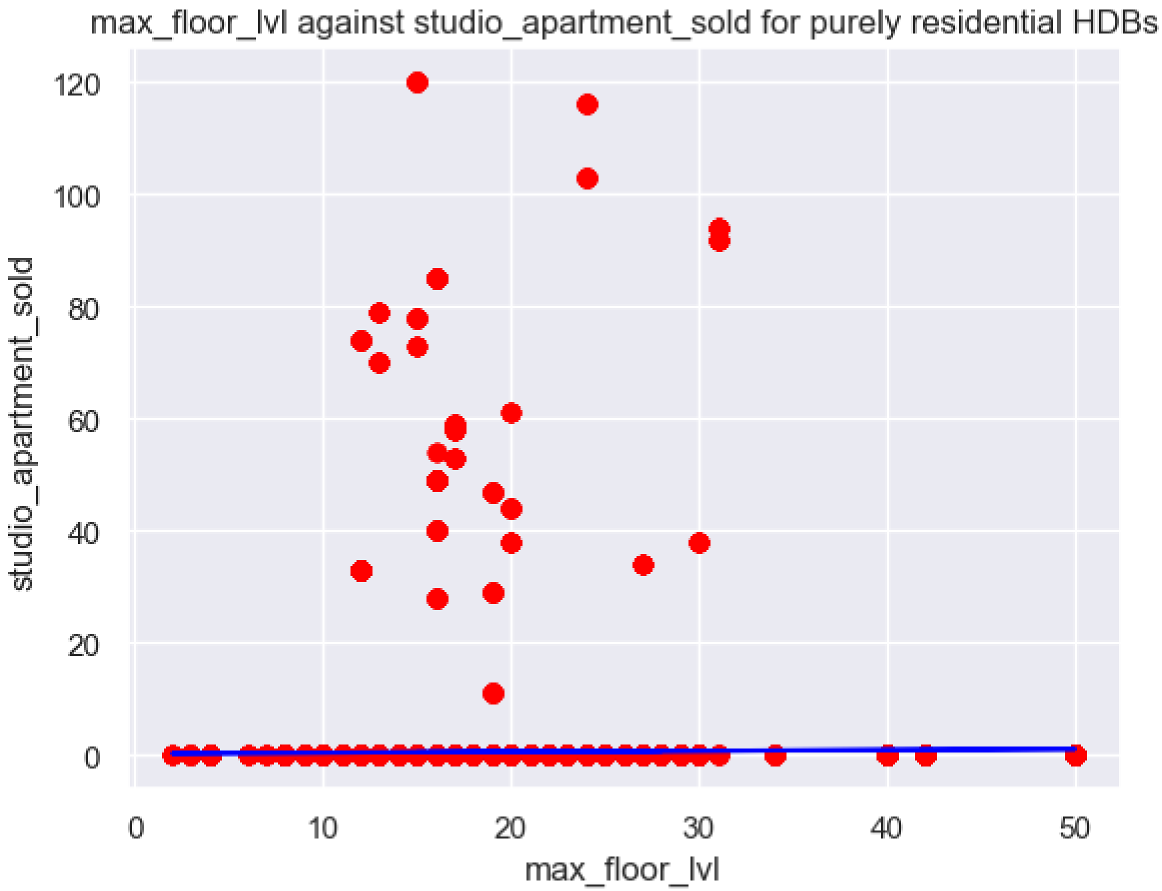

max_floor_lvl against 2room_sold for purely residential HDBs
2room_sold
160
140
120
100
80
60
40
20
0
0
10
20
30
40
50
max_floor_lvl

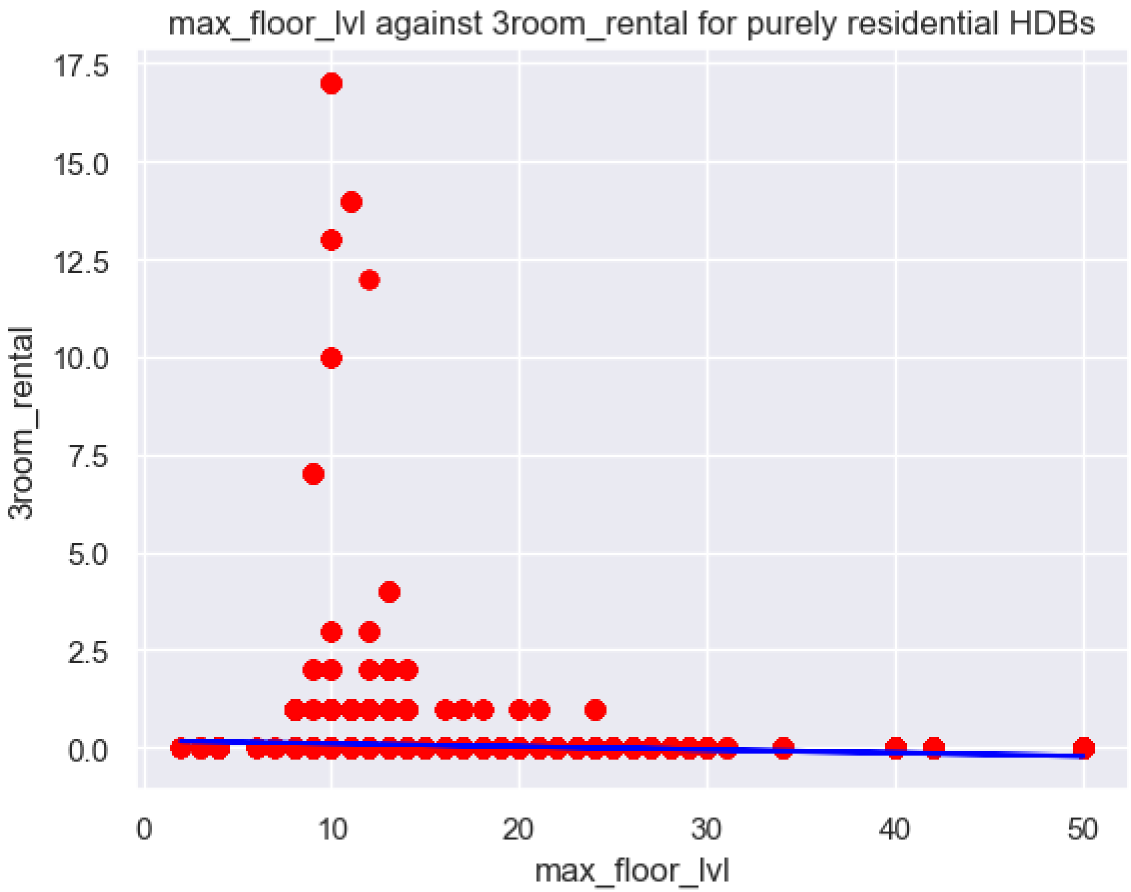

max_floor_lvl against 4room_sold for purely residential HDBs
300
250
200
150
100
50
0
4room_sold
0
10
20
30
40
50
max_floor_lvl

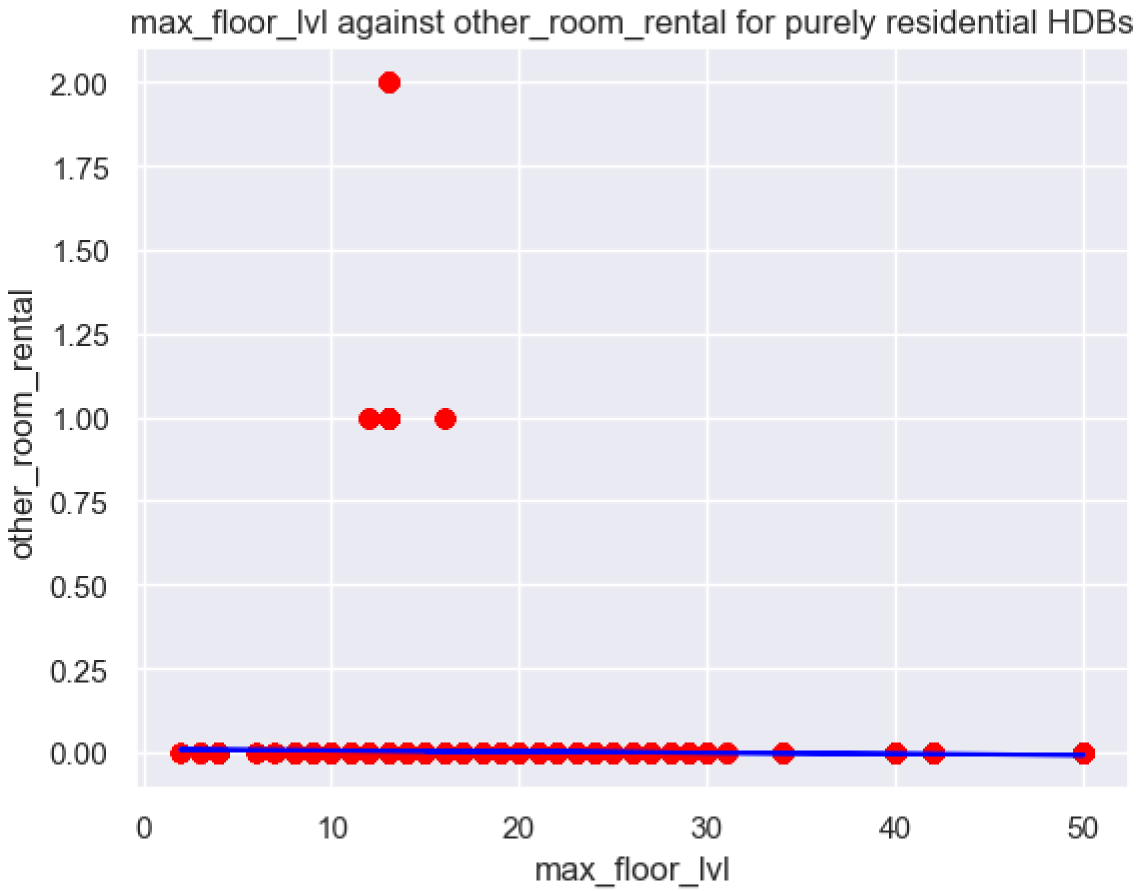

max_floor_lvl against 5room_sold for purely residential HDBs
150
125
100
75
50
25
0
5room_sold
0
10
20
30
40
50
max_floor_lvl

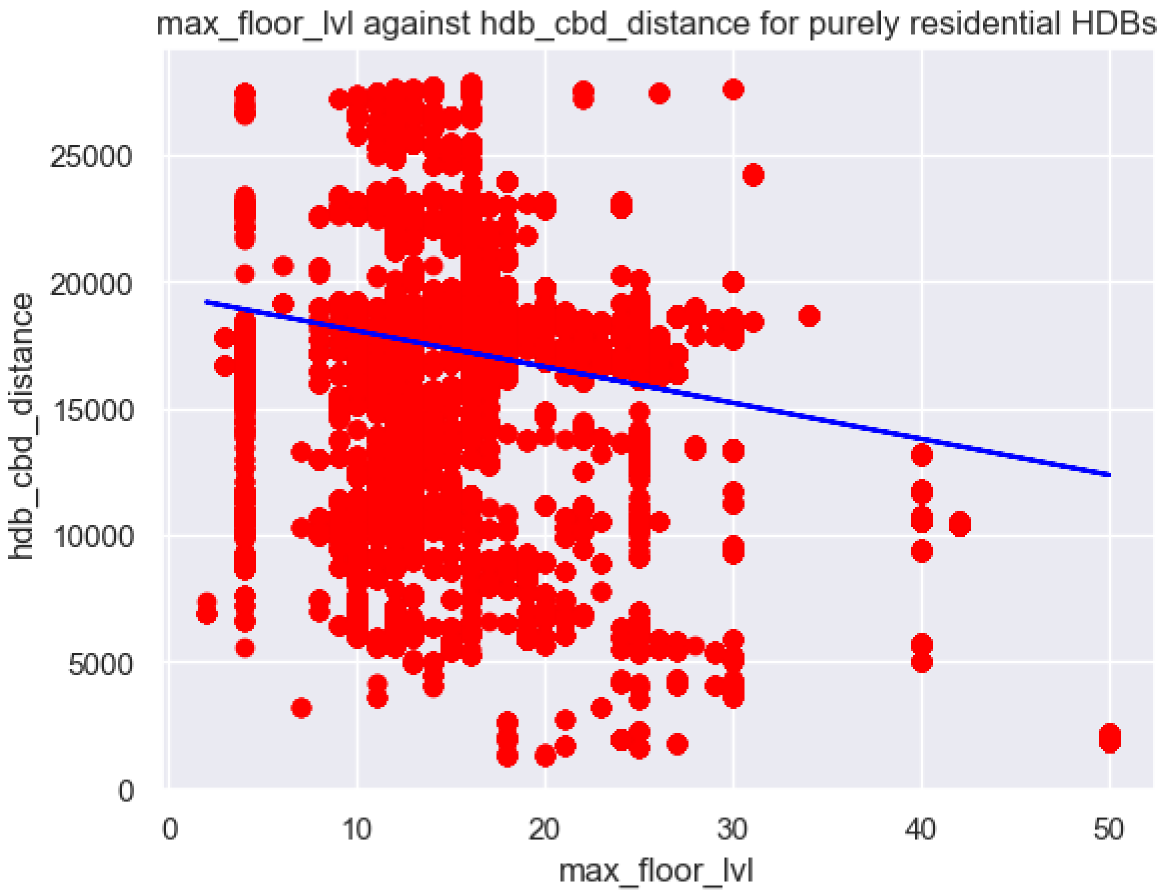

max_floor_lvl against exec_sold for purely residential HDBs
140
120
100
80
60
40
20
0
exec_sold
0
10
20
30
40
50
max_floor_lvl

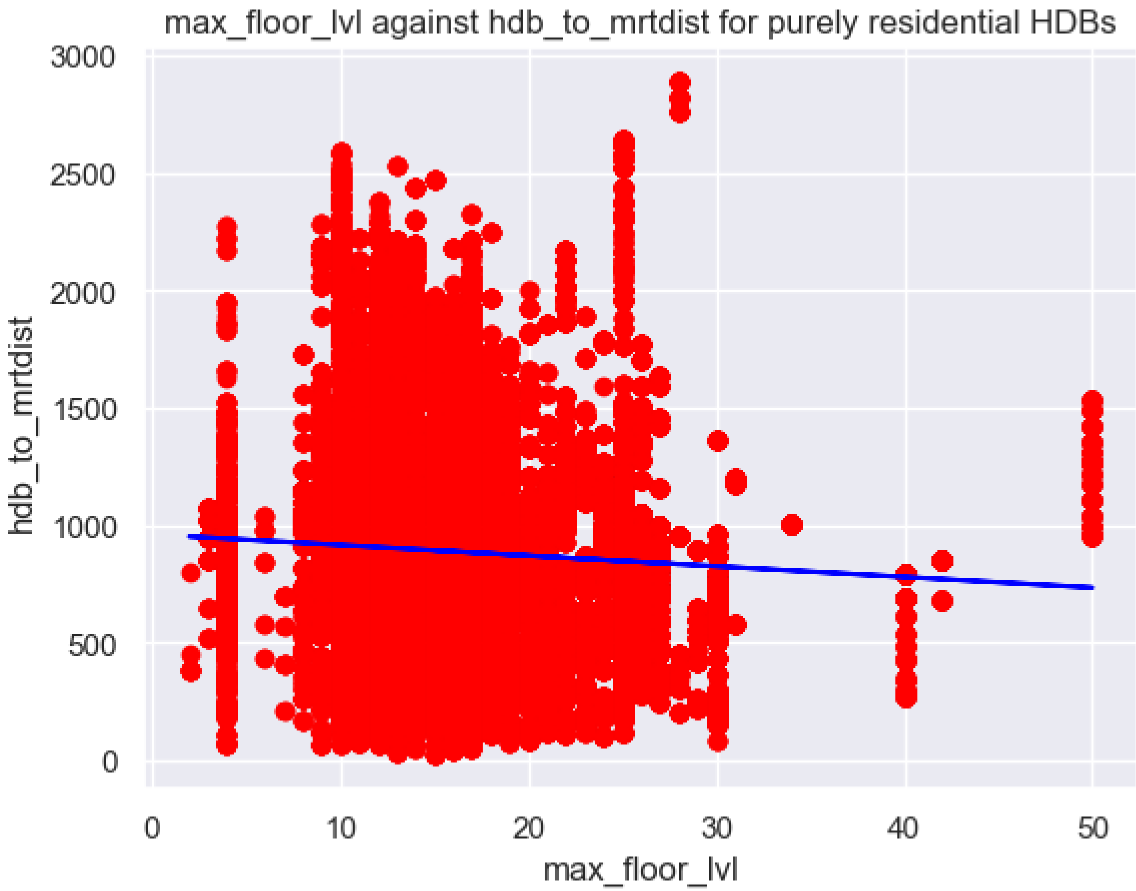

max_floor_lvl against multigen_sold for purely residential HDBs
50
40
30
20
10
0
multigen_sold
0
10
20
30
40
50
max_floor_lvl

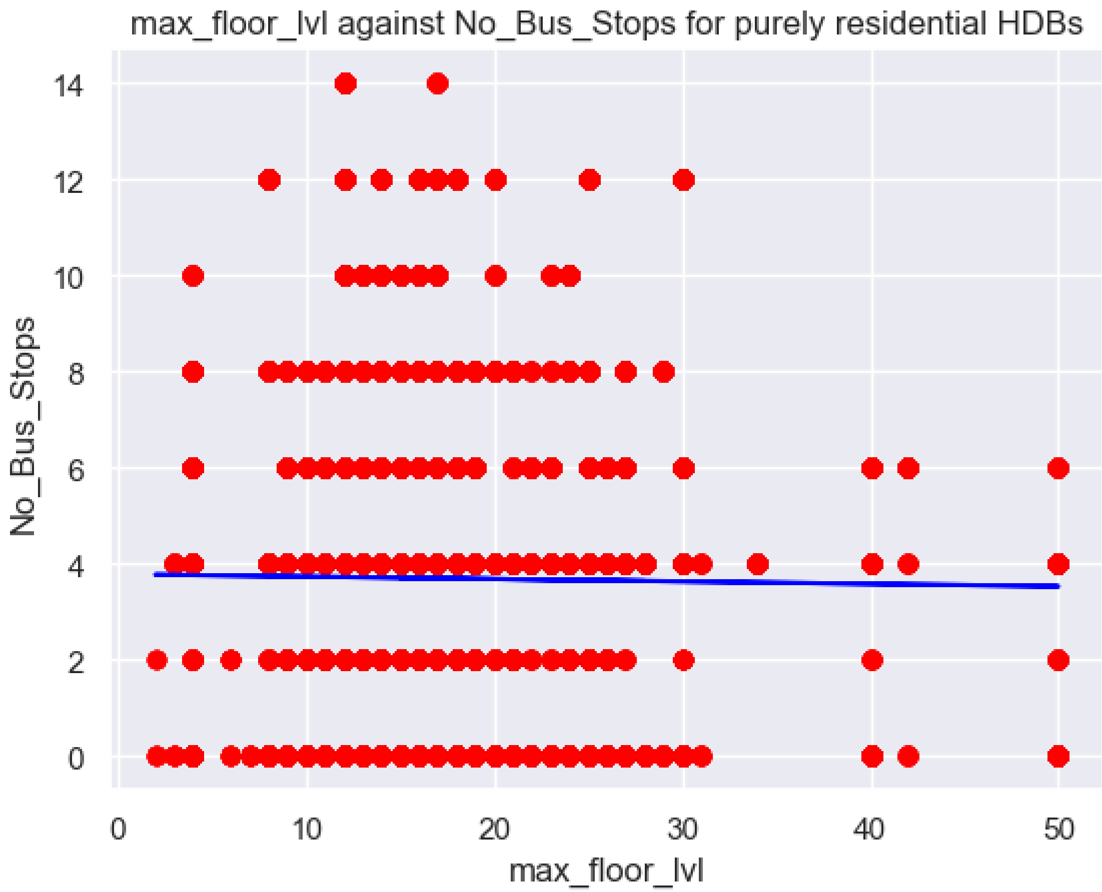

max_floor_lvl against studio_apartment_sold for purely residential HDBs
120
100
80
60
40
20
0
studio_apartment_sold
0
10
20
30
40
50
max_floor_lvl

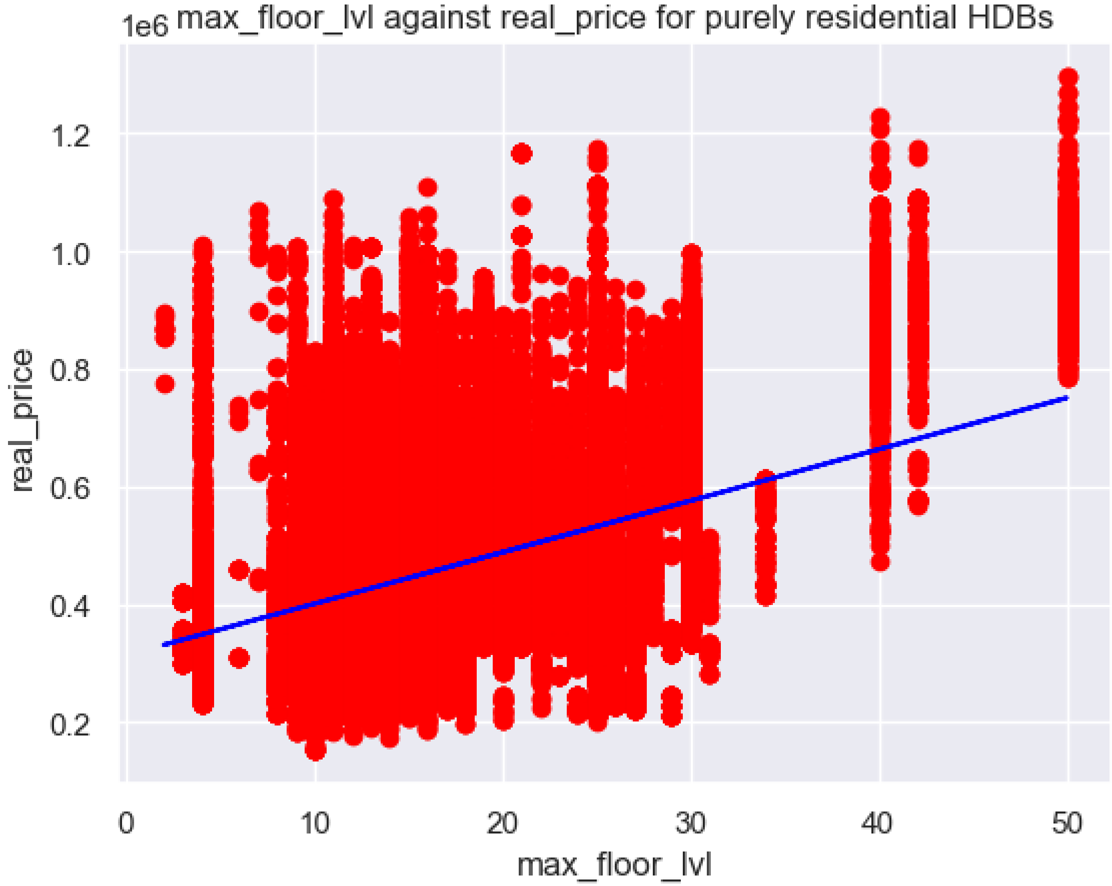

max_floor_lvl against 3room_rental for purely residential HDBs
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
3room_rental
0
10
20
30
40
50
max_floor_lvl

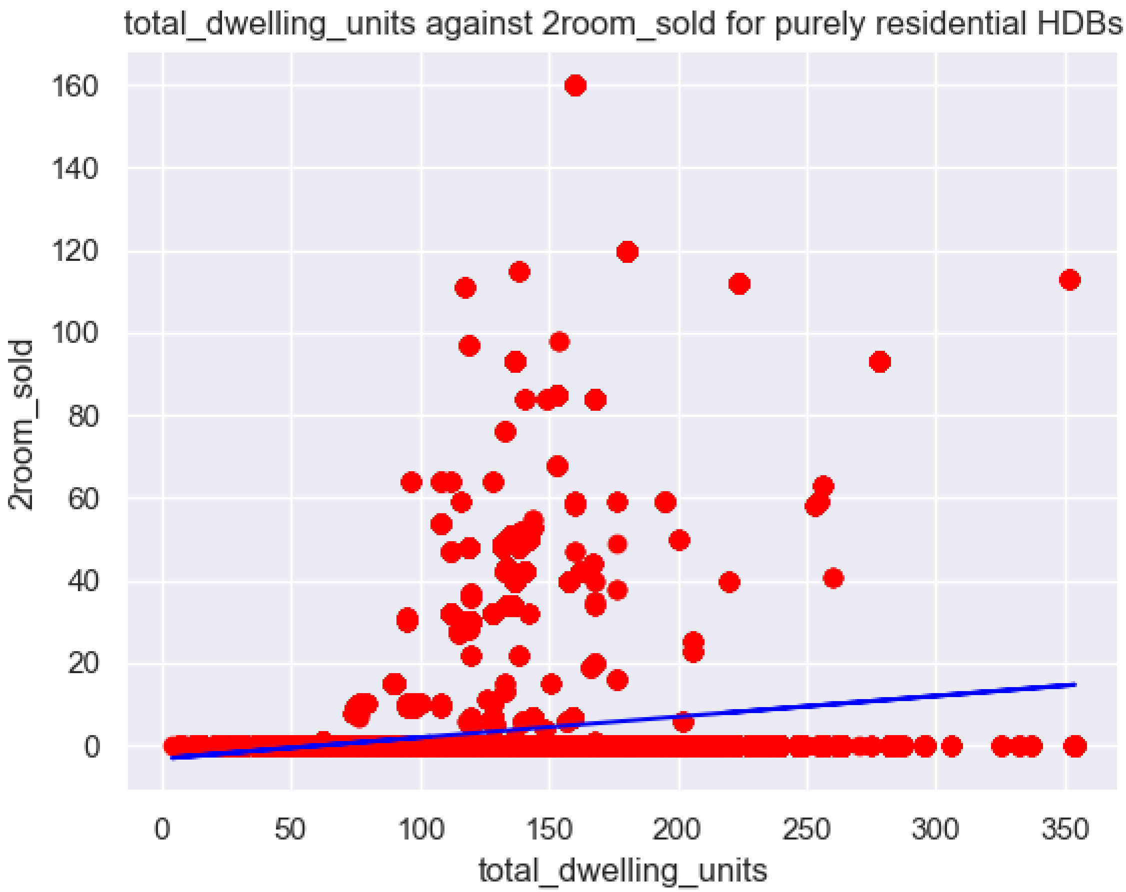

max_floor_lvl against other_room_rental for purely residential HDBs
other_room_rental
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
10
20
30
40
50
max_floor_lvl


max_floor_lvl against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
25000
20000
15000
10000
5000
0
0
10
20
30
40
50
max_floor_lvl

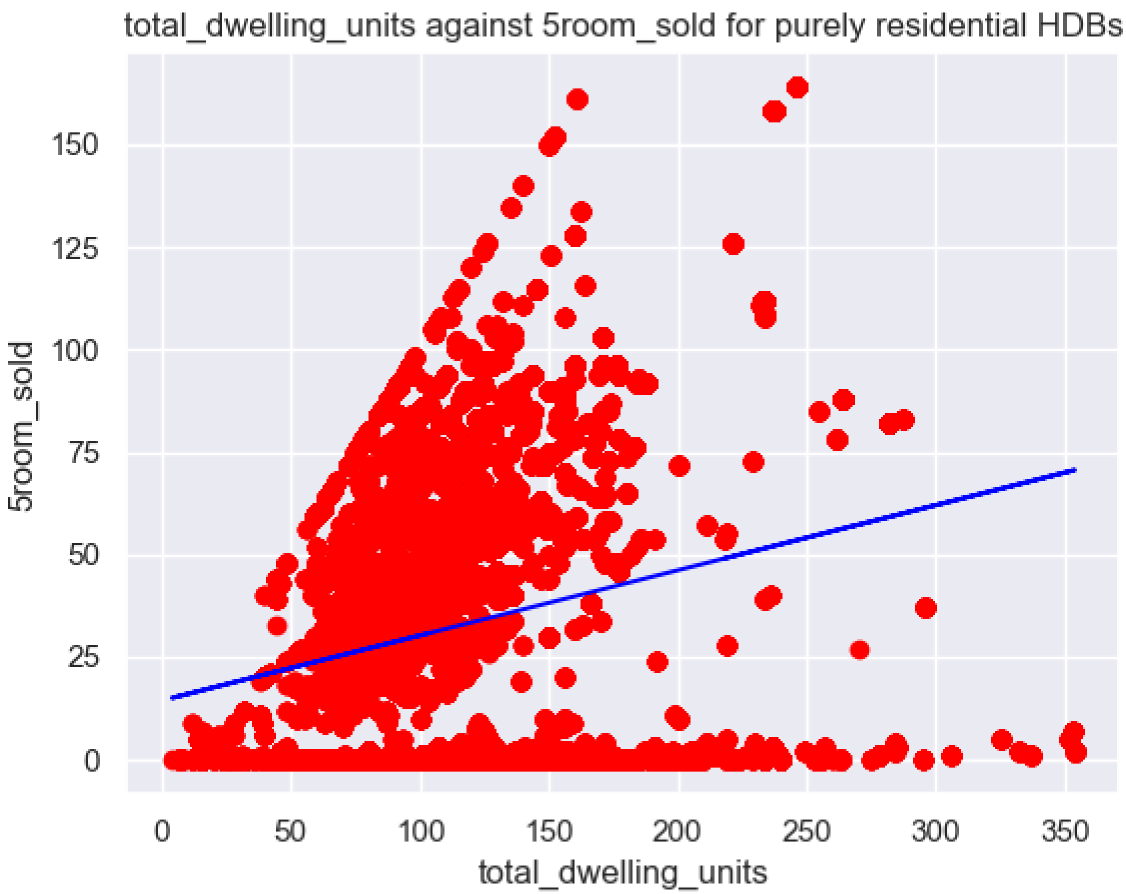

max_floor_lvl against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
3000
2500
2000
1500
1000
500
0
0
10
20
30
40
50
max_floor_lvl

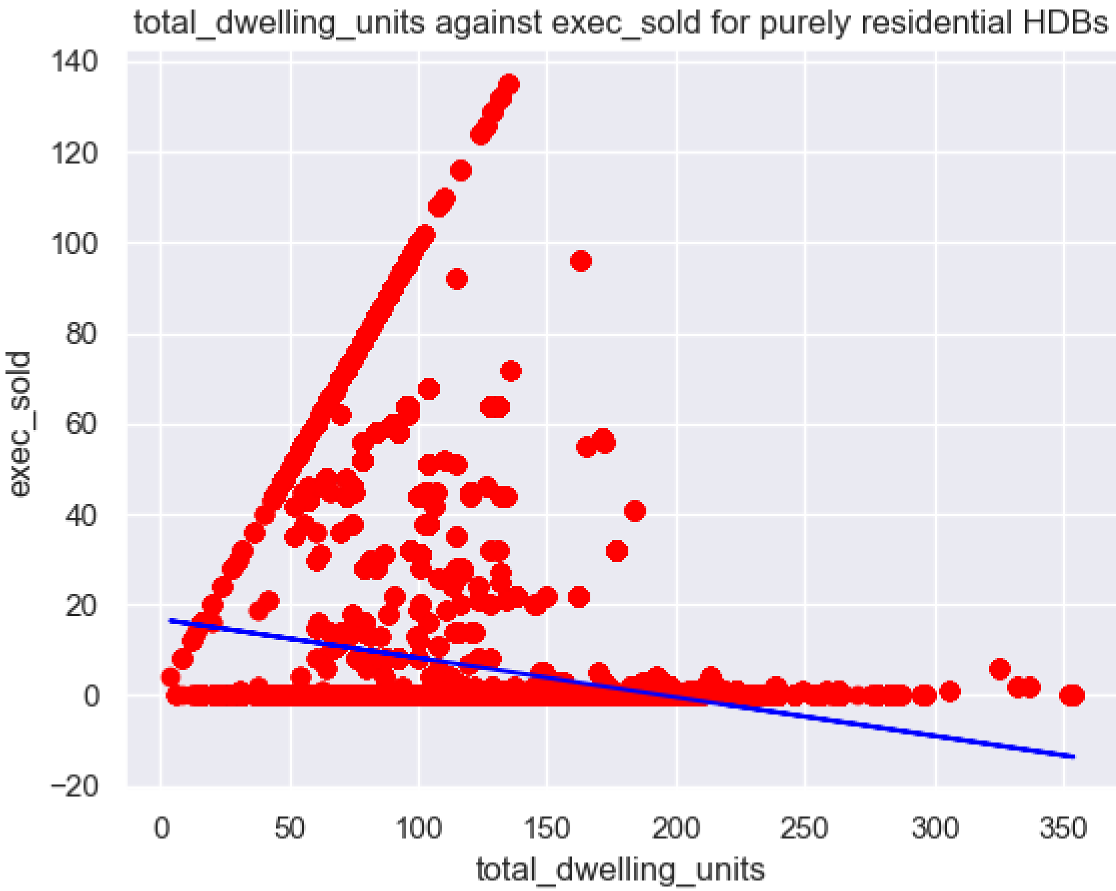

max_floor_lvl against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
14
12
10
8
6
4
2
0
0
10
20
30
40
50
max_floor_lvl


1e6 max_floor_lvl against real_price for purely residential HDBs
1.2
1.0
0.8
0.6
0.4
0.2
real_price
0
10
20
30
40
50
max_floor_lvl

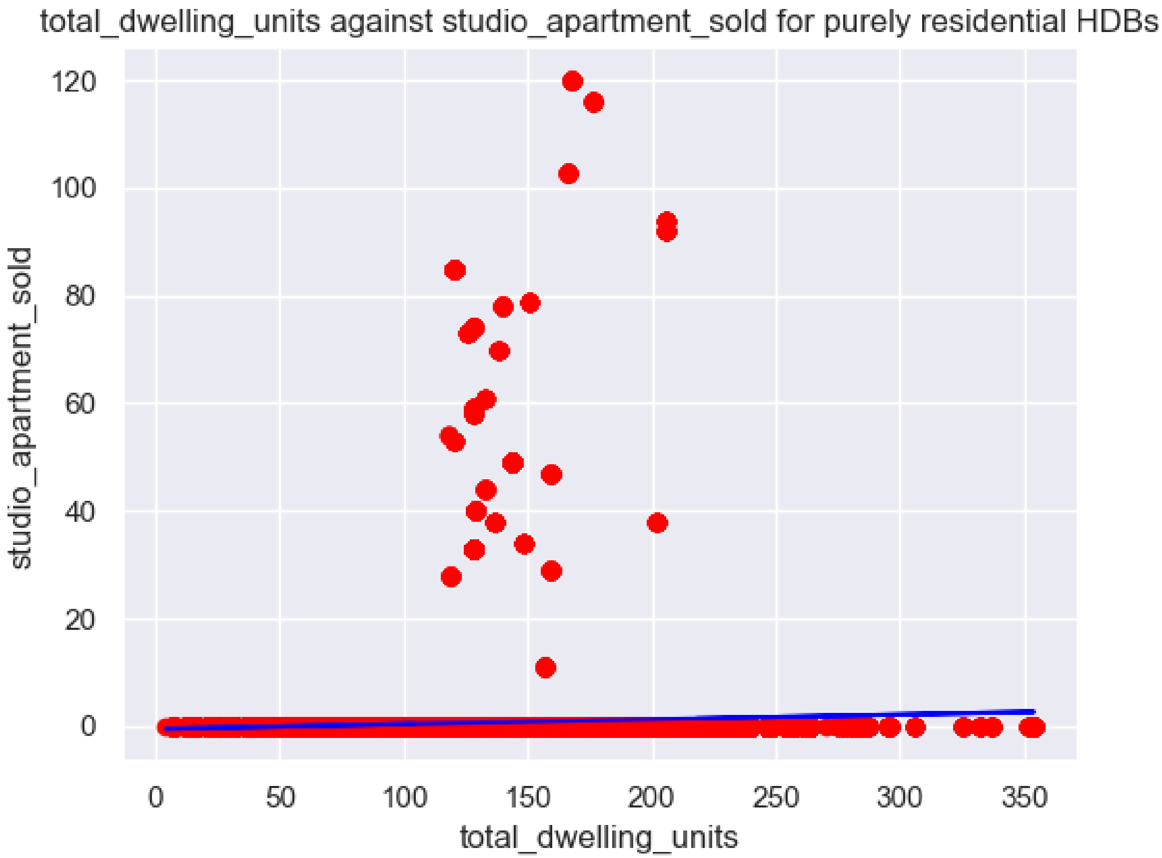

total_dwelling_units against 2room_sold for purely residential HDBs
160
140
120
100
80
60
40
20
0
2room_sold
0
50
100
150
200
250
300
350
total_dwelling_units

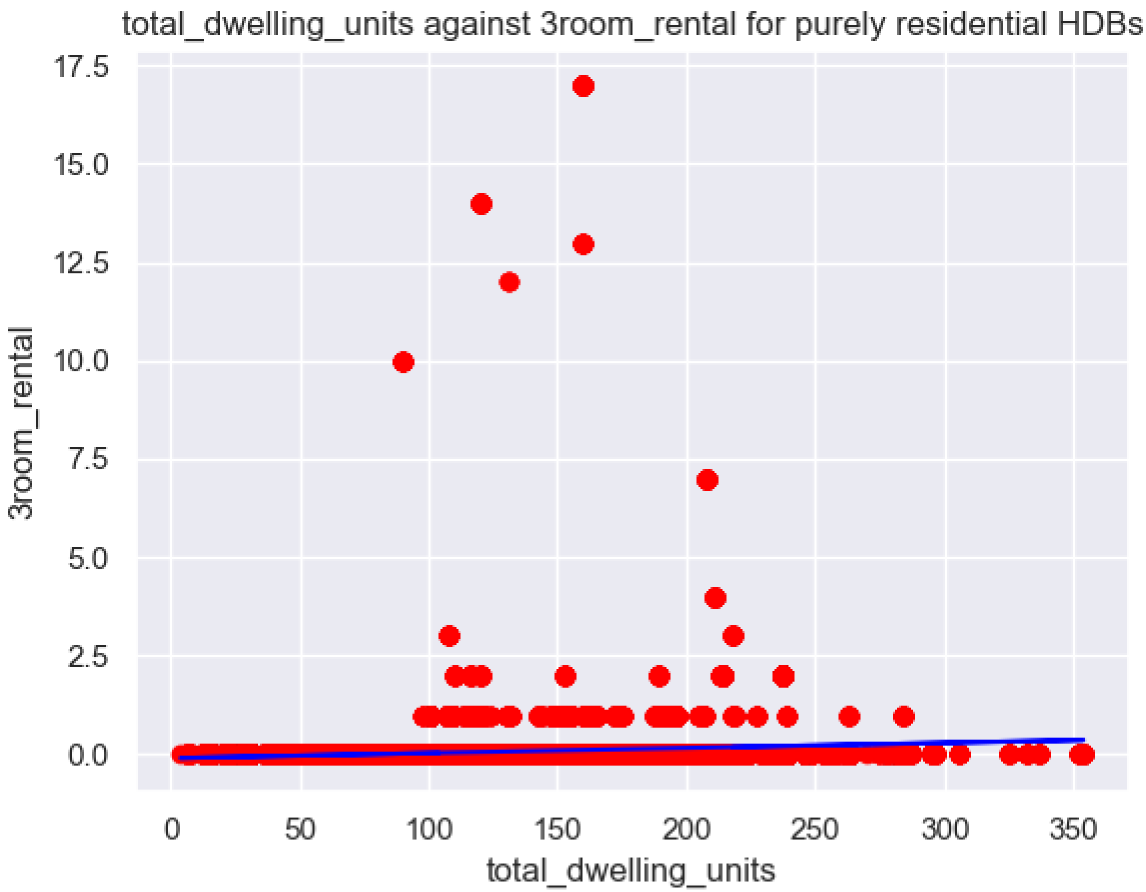

total_dwelling_units against 4room_sold for purely residential HDBs
4room_sold
0
50
100
150
200
250
300
total_dwelling_units
0
50
100
150
200
250
300
350


total_dwelling_units against 5room_sold for purely residential HDBs
5room_sold
0
25
50
75
100
125
150
total_dwelling_units
0
50
100
150
200
250
300
350

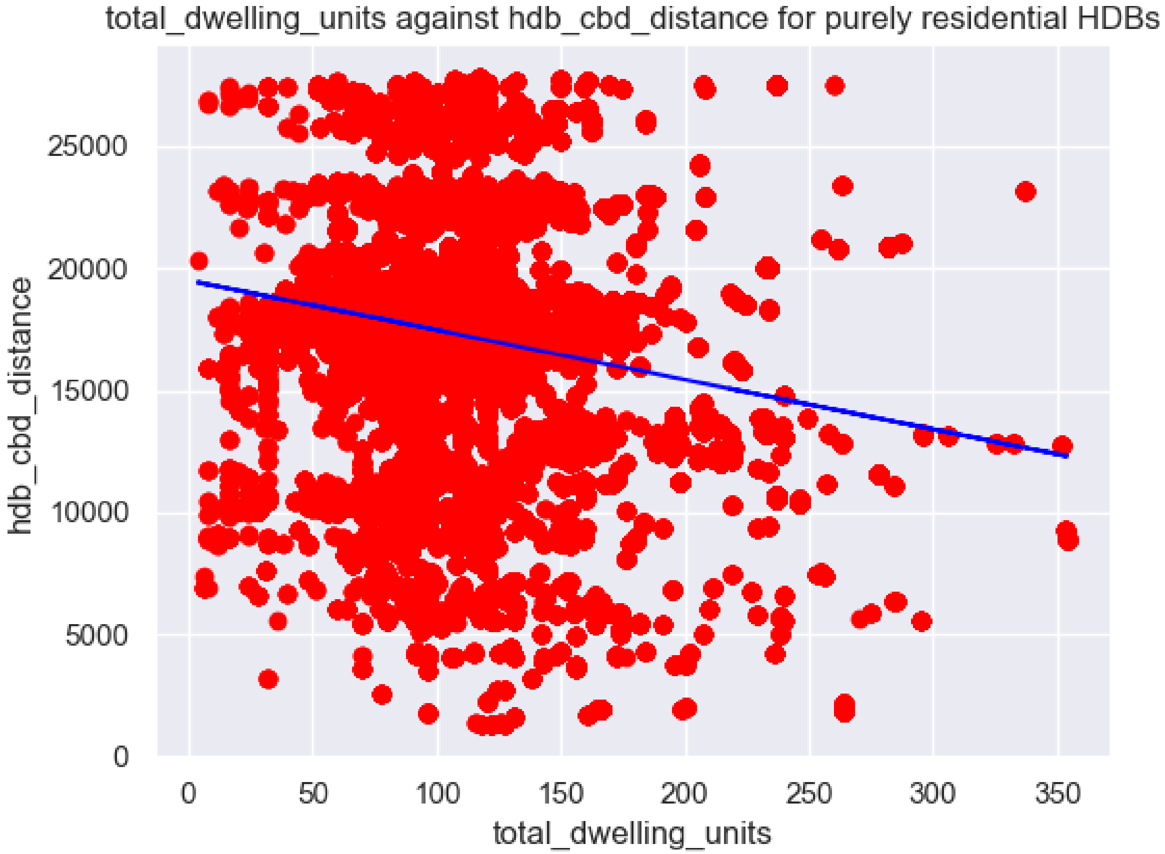

total_dwelling_units against exec_sold for purely residential HDBs
exec_sold
140
120
100
80
60
40
20
0
−20
0
50
100
150
200
250
300
350
total_dwelling_units

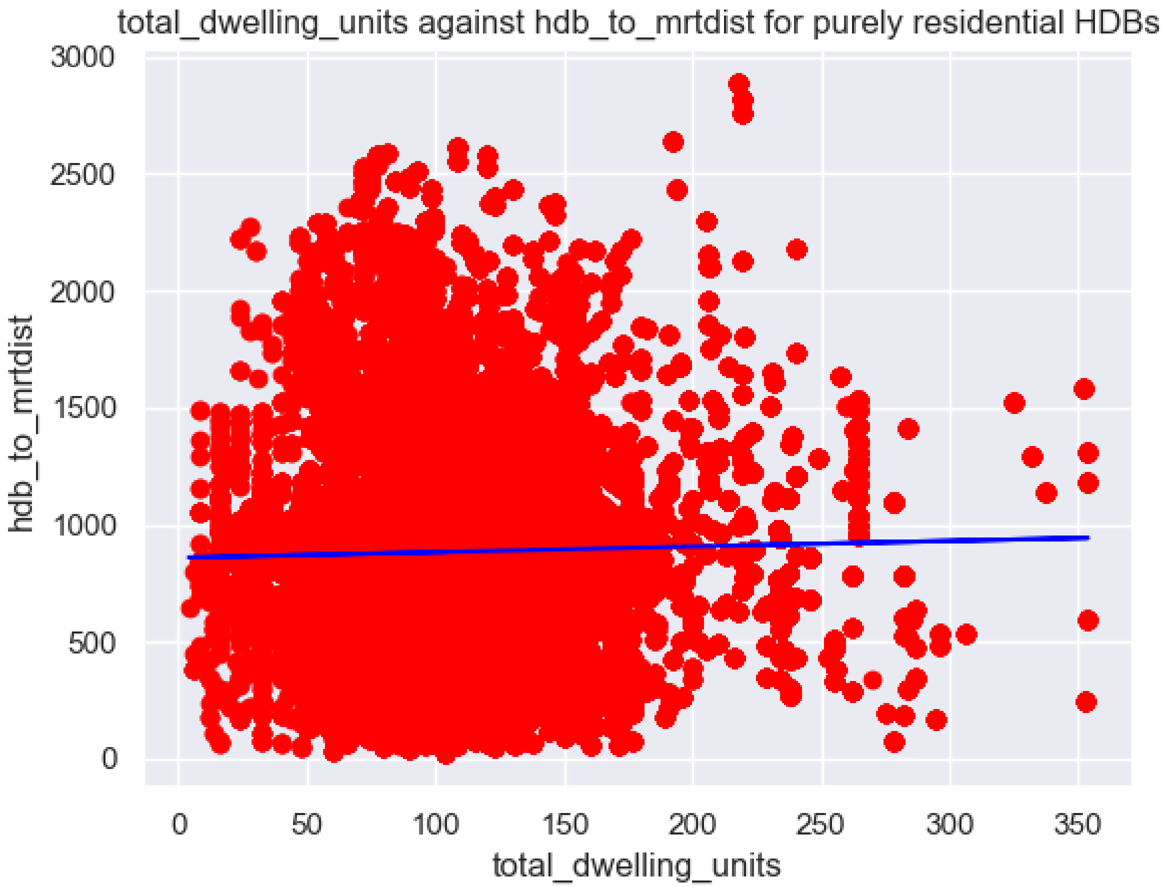

total_dwelling_units against multigen_sold for purely residential HDBs
multigen_sold
50
40
30
20
10
0
0
50
100
150
200
250
300
350
total_dwelling_units

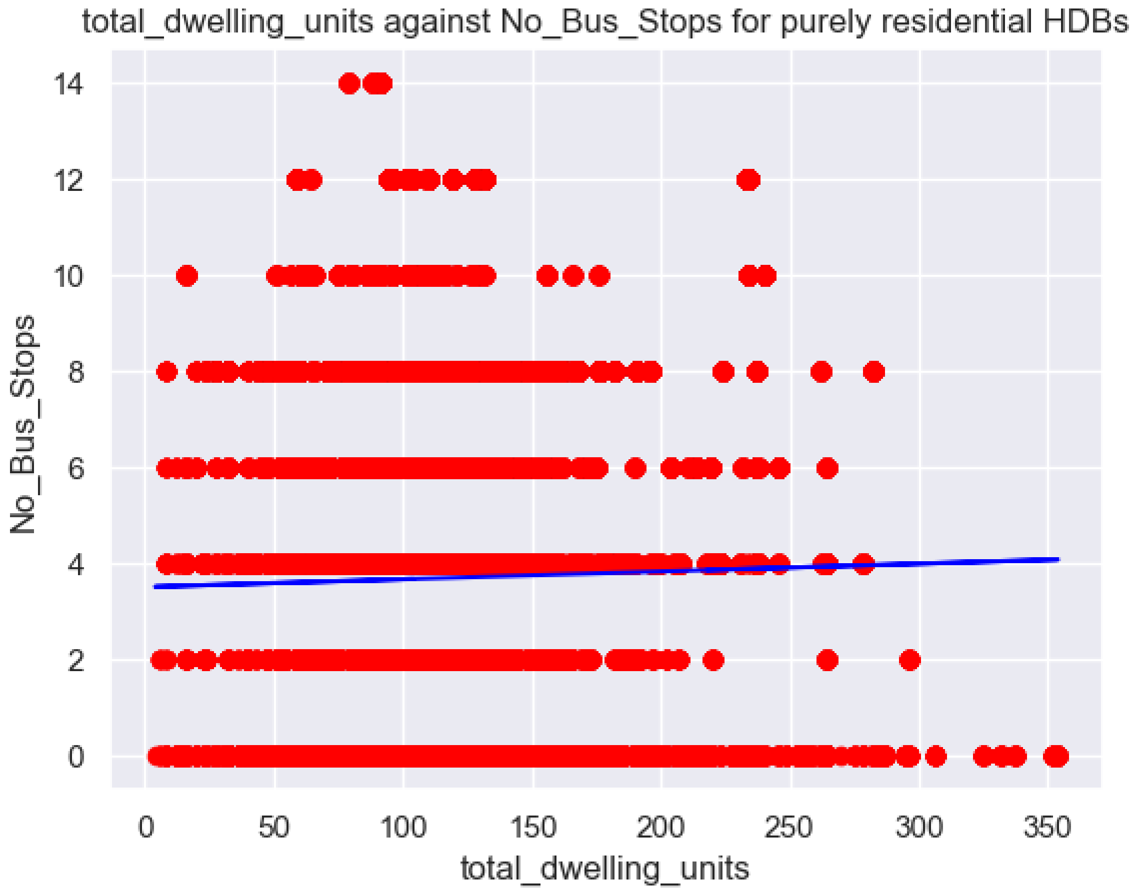

total_dwelling_units against studio_apartment_sold for purely residential HDBs
studio_apartment_sold
total_dwelling_units
0
20
40
60
80
100
120
0
50
100
150
200
250
300
350

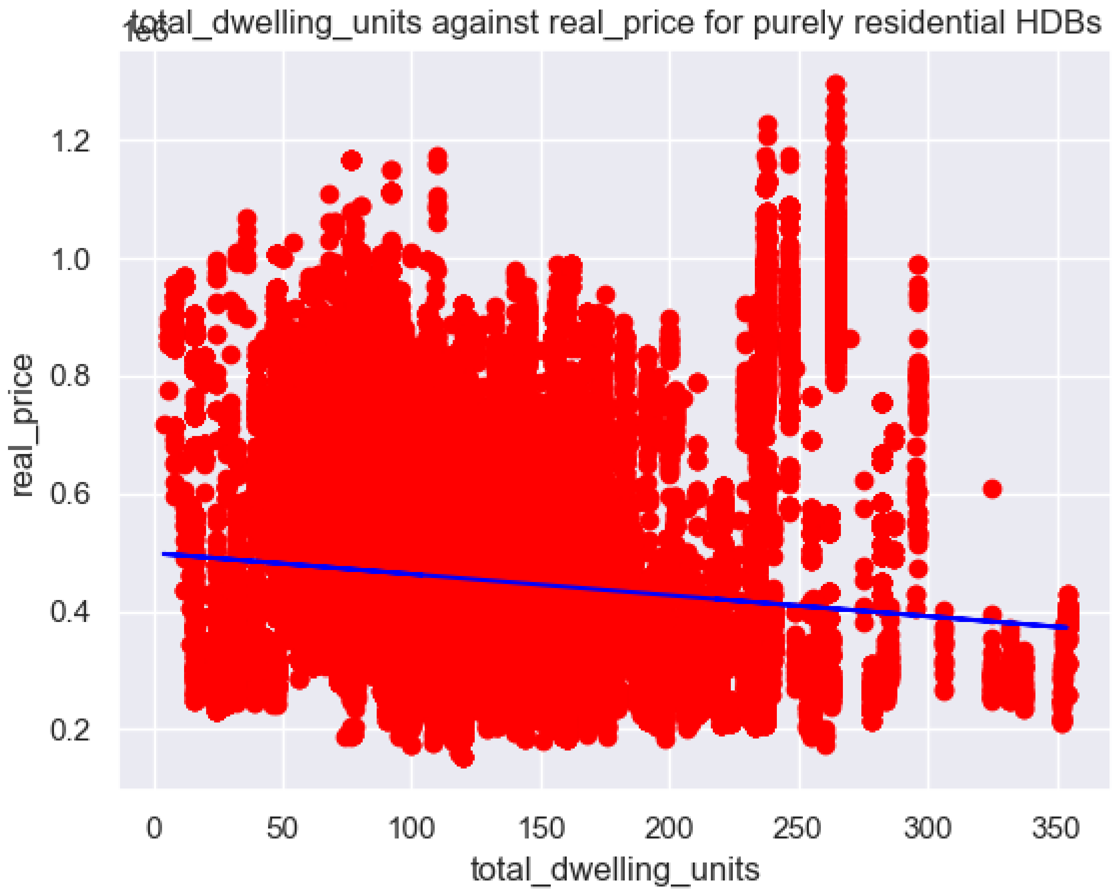

total_dwelling_units against 3room_rental for purely residential HDBs
3room_rental
total_dwelling_units
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
0
50
100
150
200
250
300
350

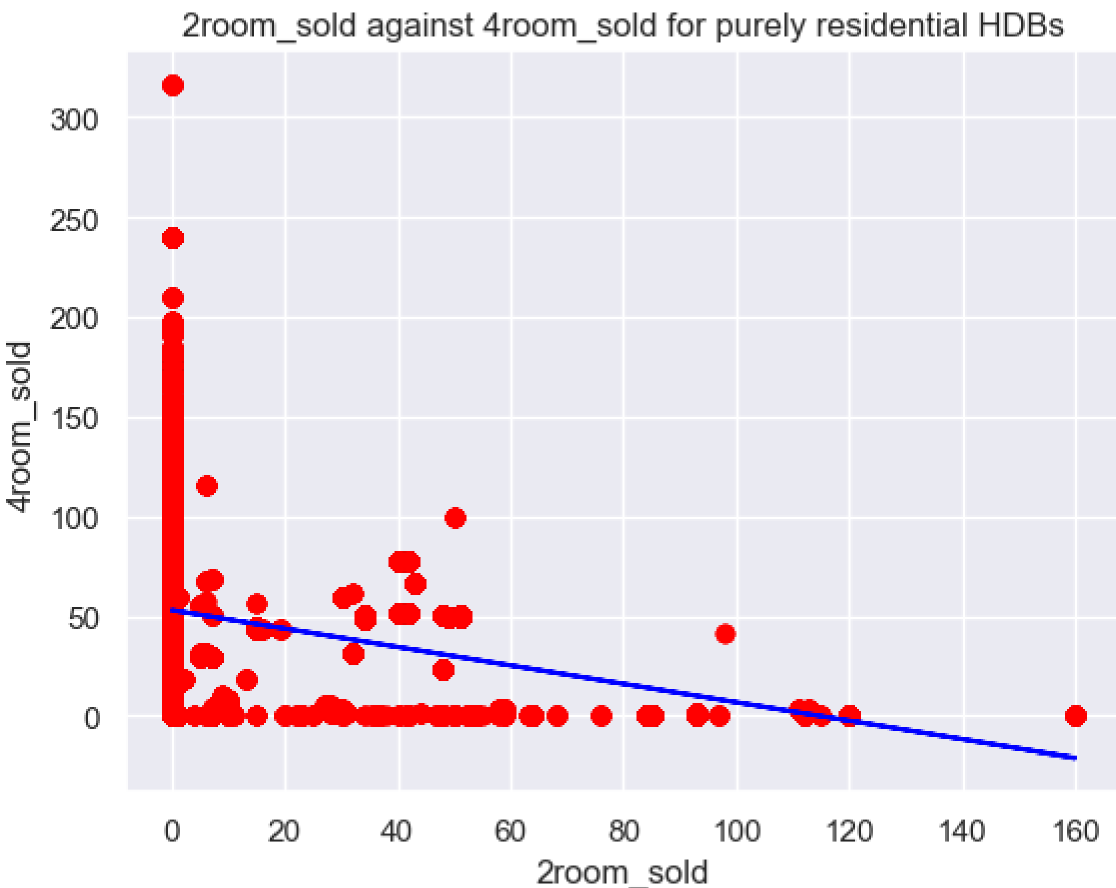

total_dwelling_units against other_room_rental for purely residential HDBs
other_room_rental
total_dwelling_units
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
0
50
100
150
200
250
300
350

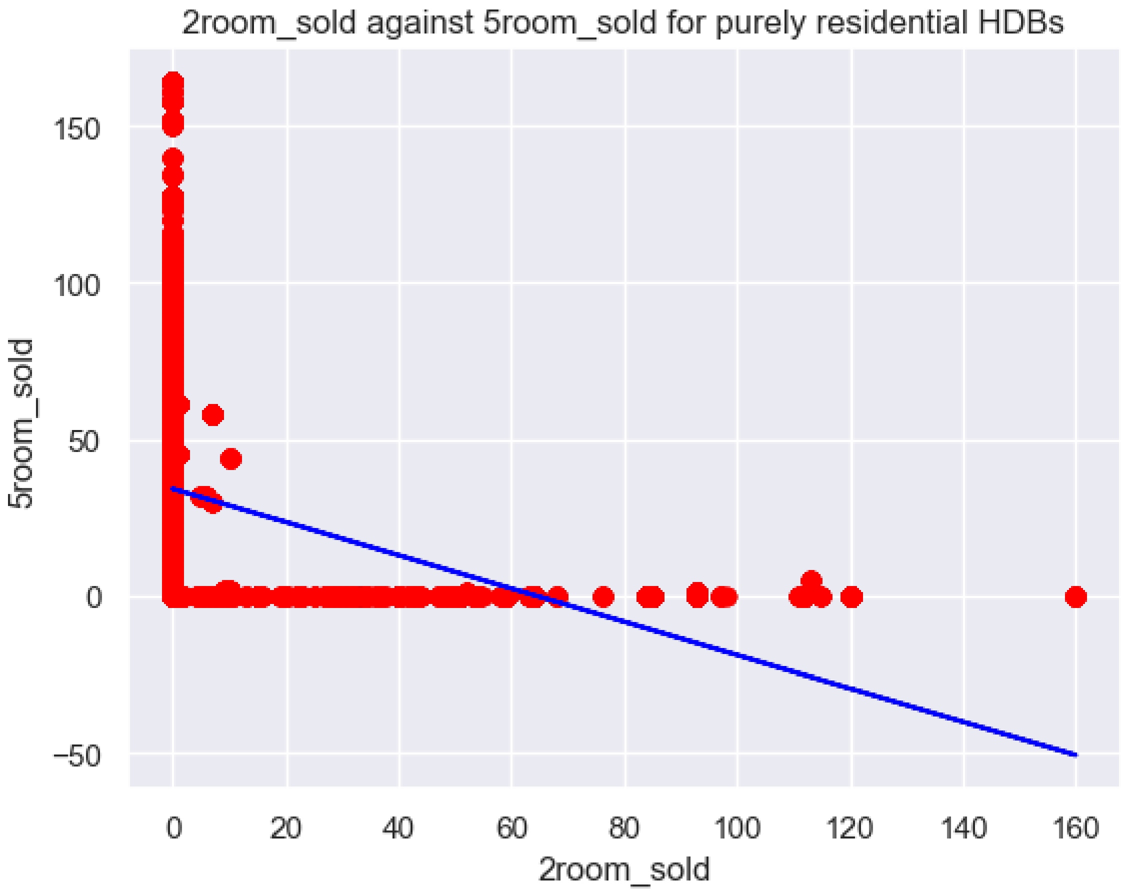

total_dwelling_units against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
total_dwelling_units
0
5000
10000
15000
20000
25000
0
50
100
150
200
250
300
350

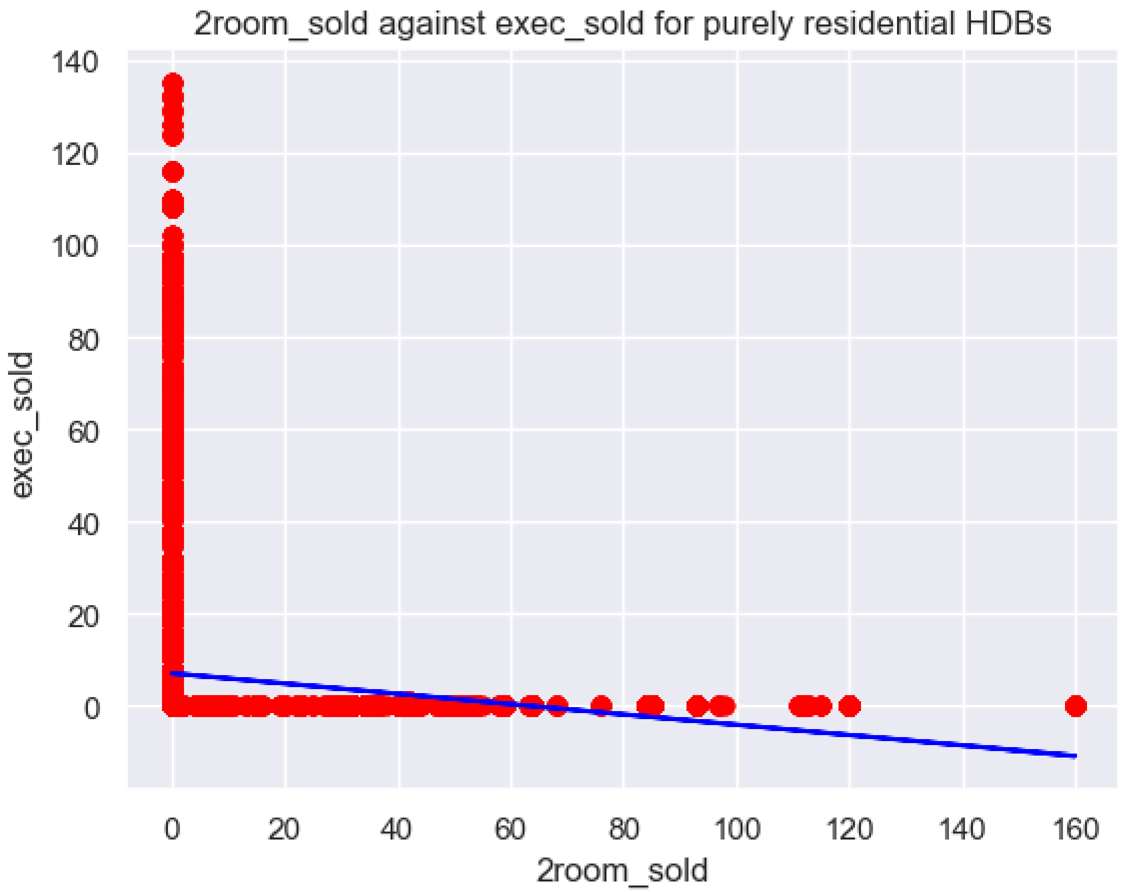

total_dwelling_units against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
3000
2500
2000
1500
1000
500
0
0
50
100
150
200
250
300
350
total_dwelling_units


total_dwelling_units against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
14
12
10
8
6
4
2
0
0
50
100
150
200
250
300
350
total_dwelling_units

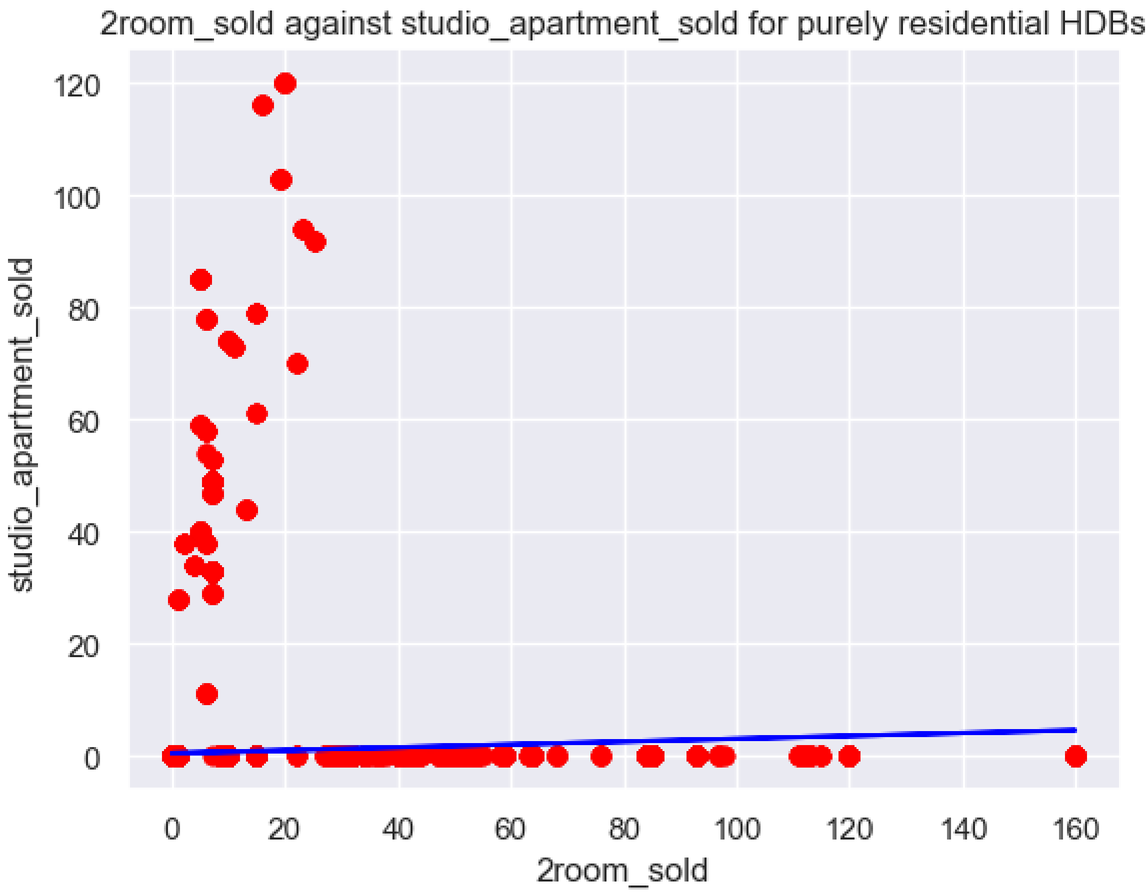

1e6
total_dwelling_units against real_price for purely residential HDBs
real_price
1.2
1.0
0.8
0.6
0.4
0.2
0
50
100
150
200
250
300
350
total_dwelling_units

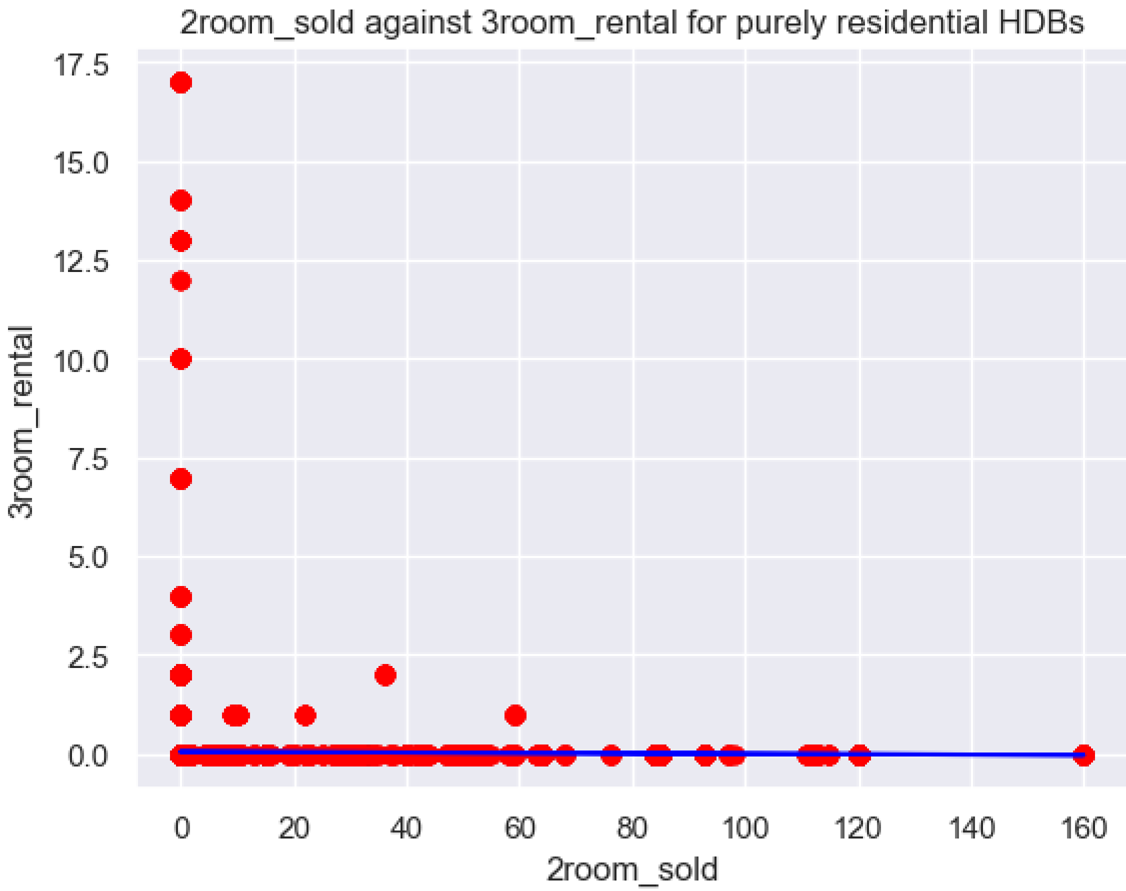

2room_sold against 4room_sold for purely residential HDBs
4room_sold
300
250
200
150
100
50
0
0
20
40
60
80
100
120
140
160
2room_sold


2room_sold against 5room_sold for purely residential HDBs
5room_sold
150
100
50
0
−50
0 20 40 60 80 100 120 140 160
2room_sold


2room_sold against exec_sold for purely residential HDBs
exec_sold
140
120
100
80
60
40
20
0
0 20 40 60 80 100 120 140 160
2room_sold


2room_sold against multigen_sold for purely residential HDBs
multigen_sold
50
40
30
20
10
0
0
20
40
60
80
100
120
140
160
2room_sold

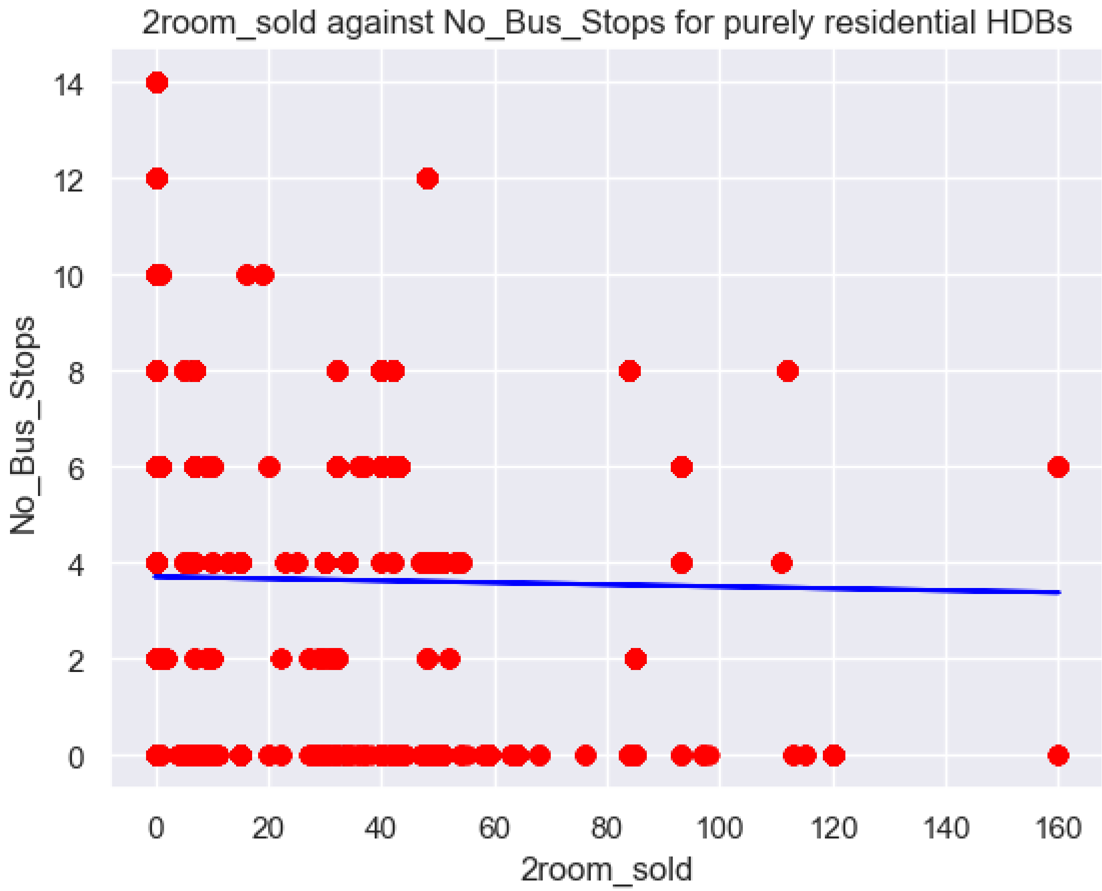

2room_sold against studio_apartment_sold for purely residential HDBs
studio_apartment_sold
120
100
80
60
40
20
0
0
20
40
60
80
100
120
140
160
2room_sold

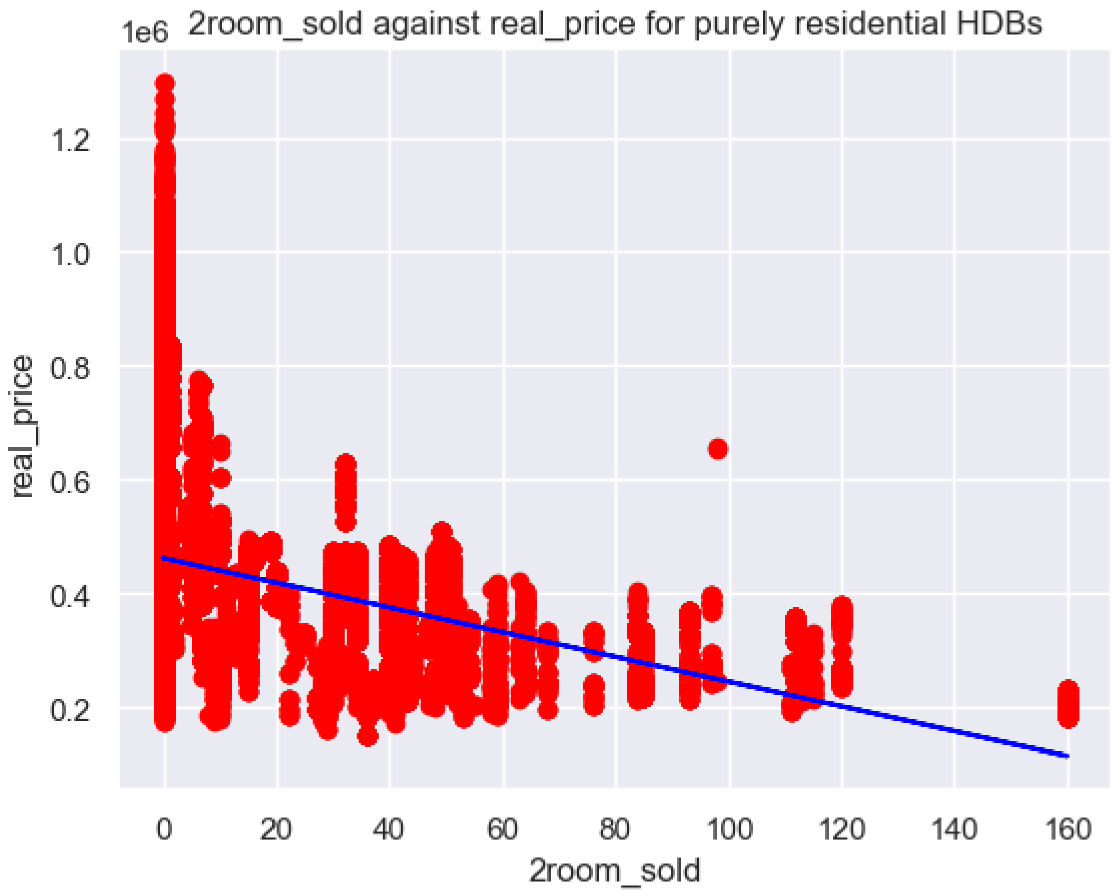

2room_sold against 3room_rental for purely residential HDBs
3room_rental
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
20
40
60
80
100
120
140
160
2room_sold

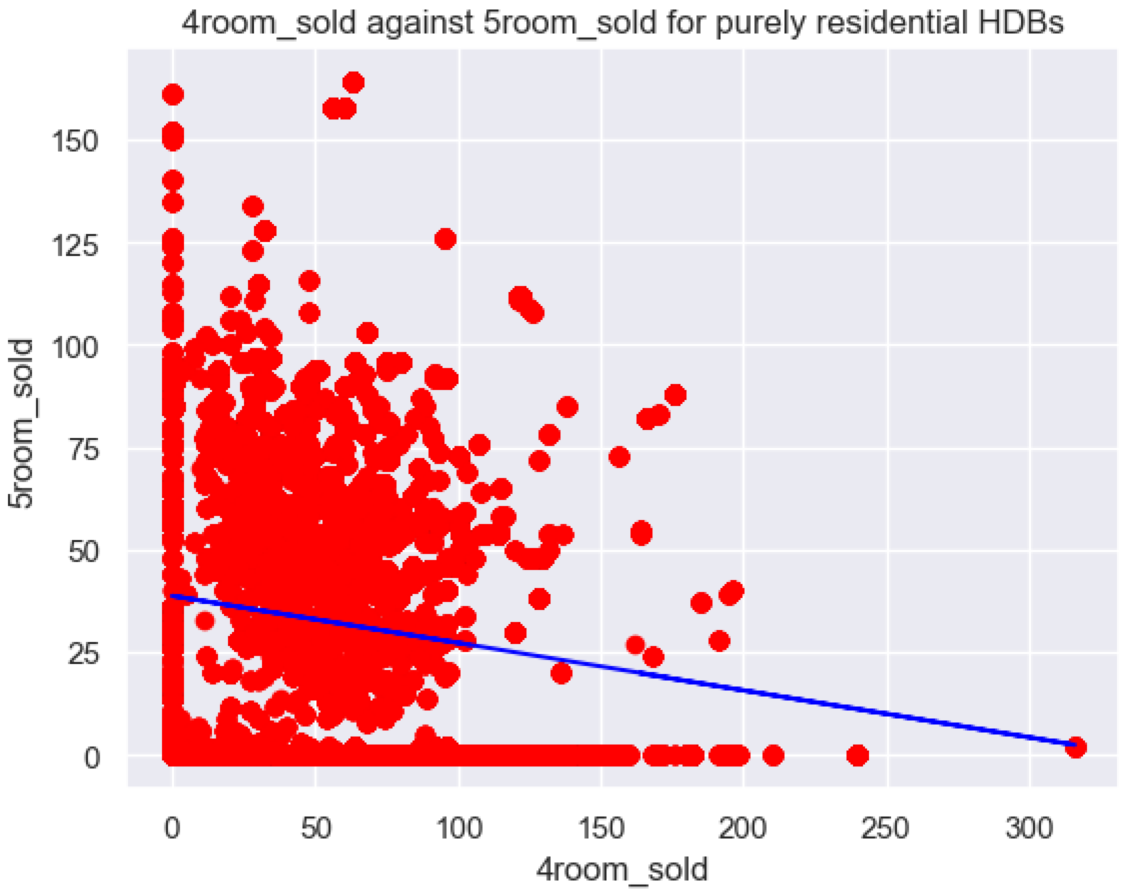

2room_sold against other_room_rental for purely residential HDBs
other_room_rental
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
20
40
60
80
100
120
140
160
2room_sold

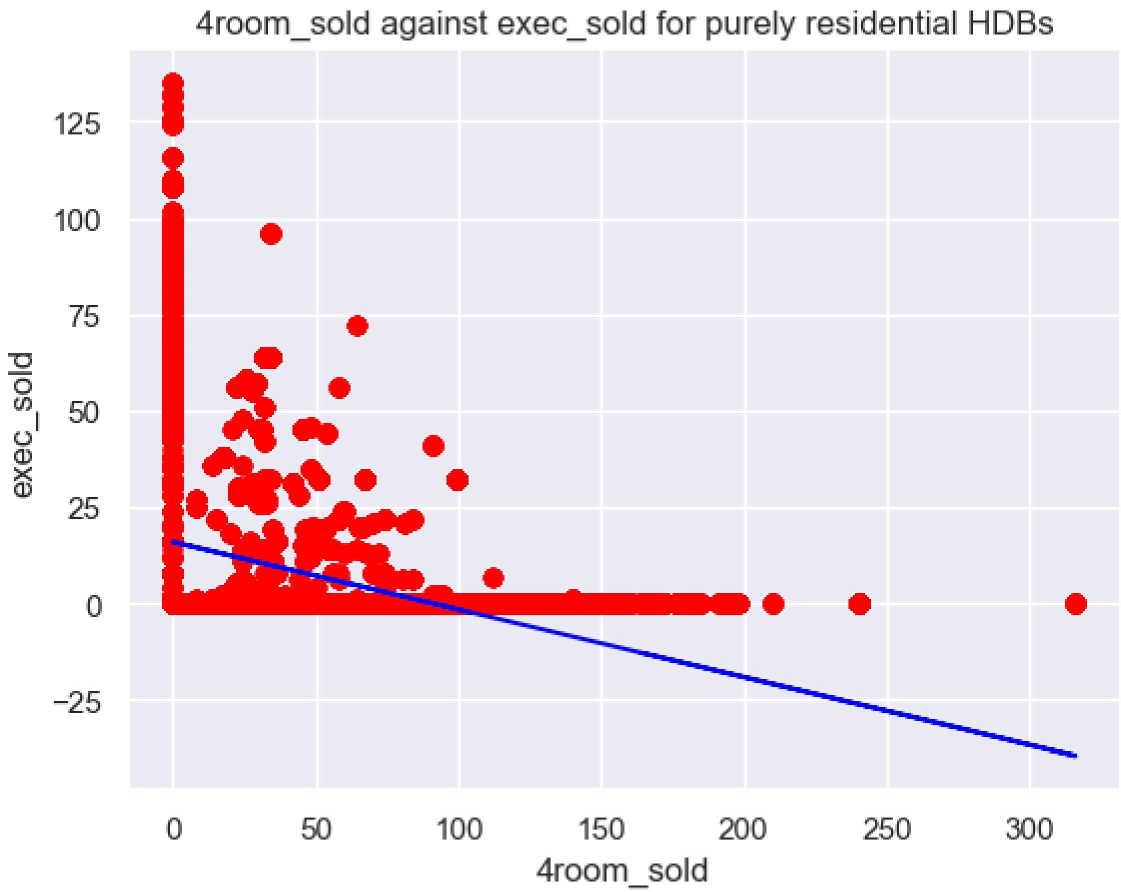

2room_sold against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
0
5000
10000
15000
20000
25000
2room_sold
0
20
40
60
80
100
120
140
160


2room_sold against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
0
500
1000
1500
2000
2500
3000
2room_sold
0
20
40
60
80
100
120
140
160


2room_sold against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
0
2
4
6
8
10
12
14
2room_sold
0
20
40
60
80
100
120
140
160

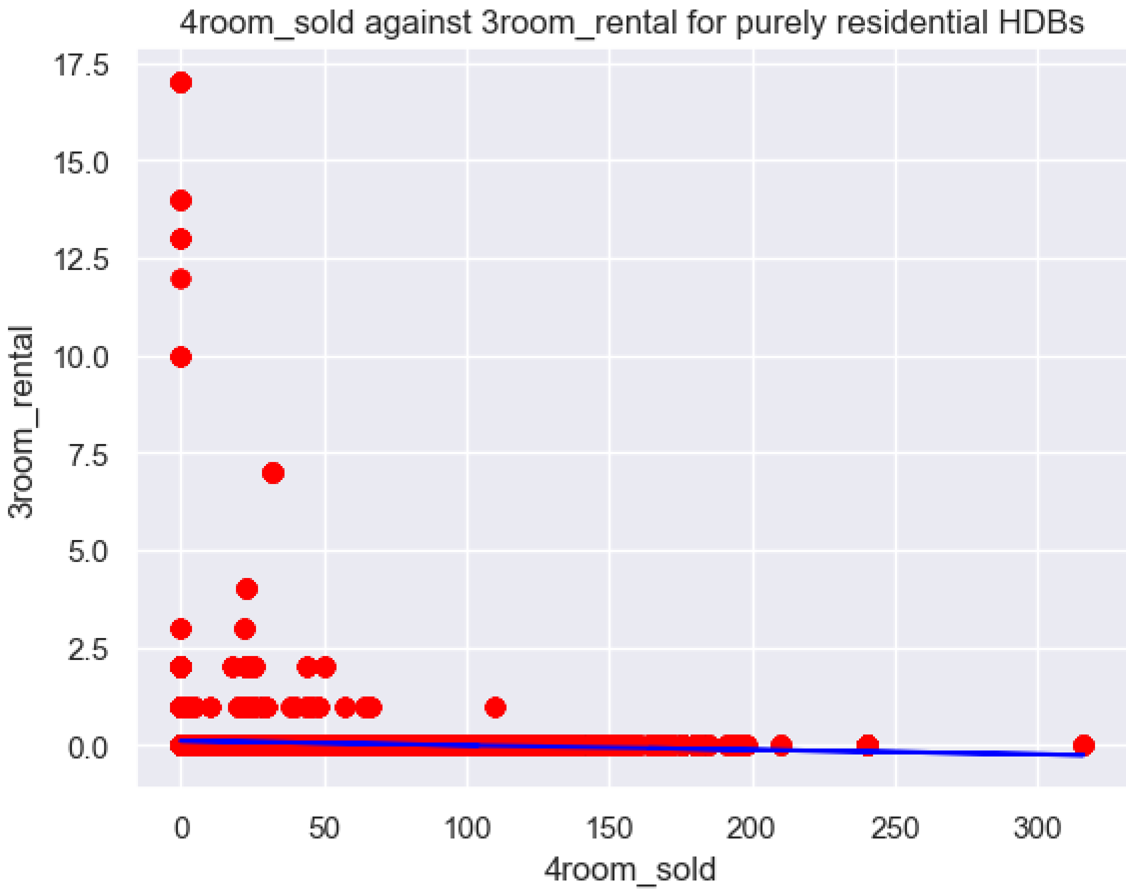

2room_sold against real_price for purely residential HDBs
1e6
real_price
0.2
0.4
0.6
0.8
1.0
1.2
2room_sold
0
20
40
60
80
100
120
140
160


4room_sold against 5room_sold for purely residential HDBs
5room_sold
4room_sold
0
25
50
75
100
125
150
0
50
100
150
200
250
300


4room_sold against exec_sold for purely residential HDBs
exec_sold
4room_sold
−25
0
25
50
75
100
125
0
50
100
150
200
250
300

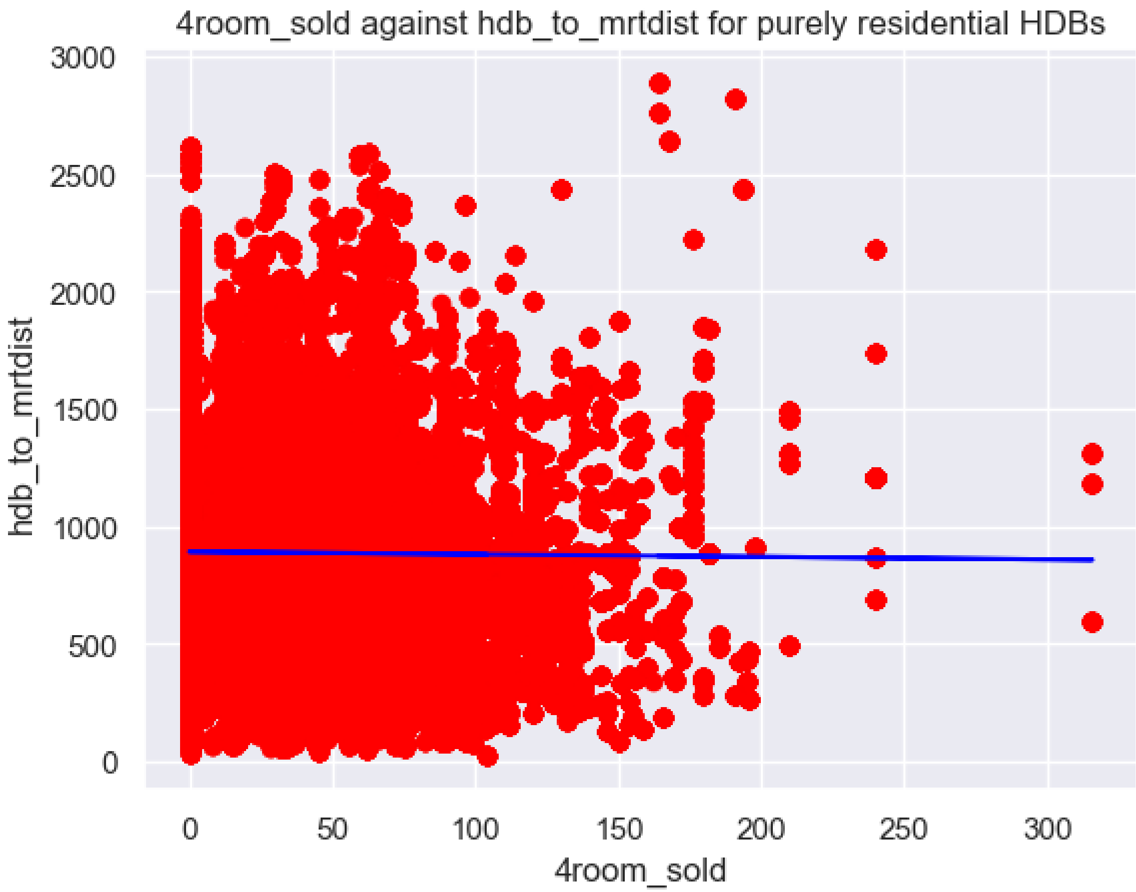

4room_sold against multigen_sold for purely residential HDBs
multigen_sold
0
10
20
30
40
50
4room_sold
0
50
100
150
200
250
300

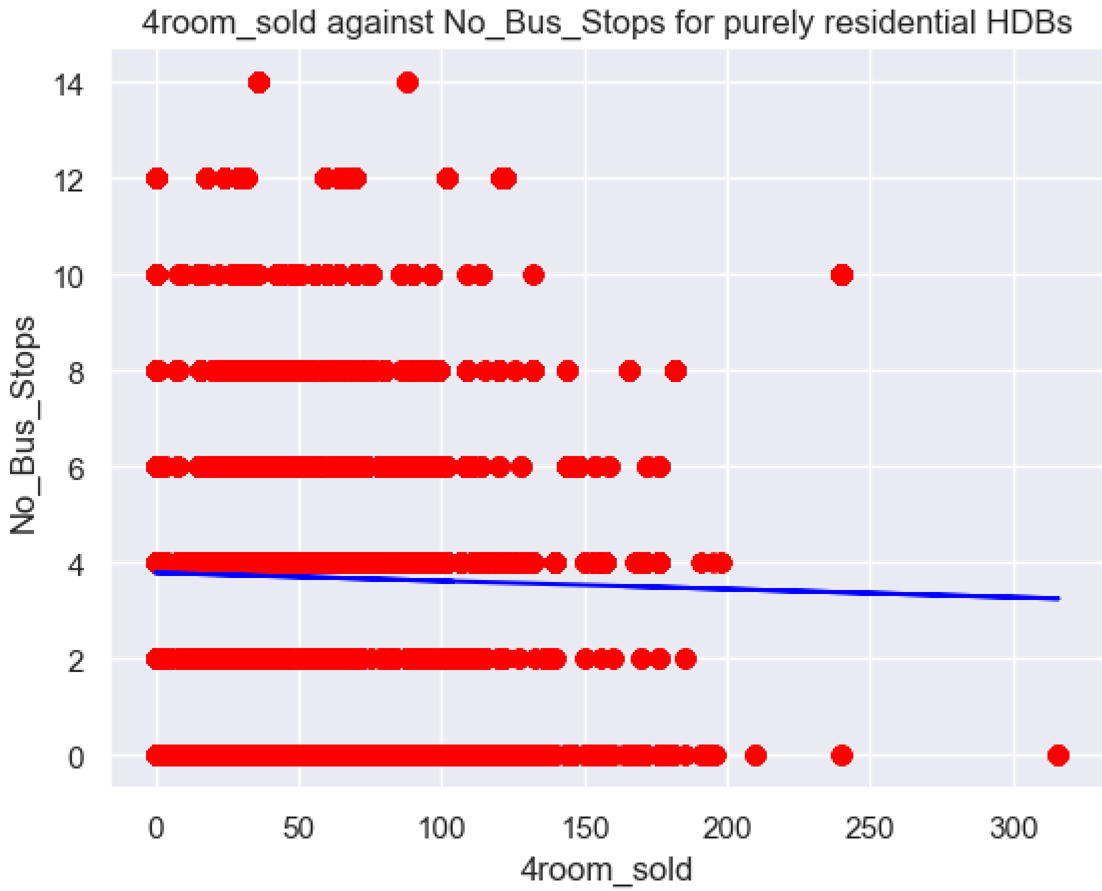

4room_sold against studio_apartment_sold for purely residential HDBs
studio_apartment_sold
0
20
40
60
80
100
120
4room_sold
0
50
100
150
200
250
300


4room_sold against 3room_rental for purely residential HDBs
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
3room_rental
0
50
100
150
200
250
300
4room_sold

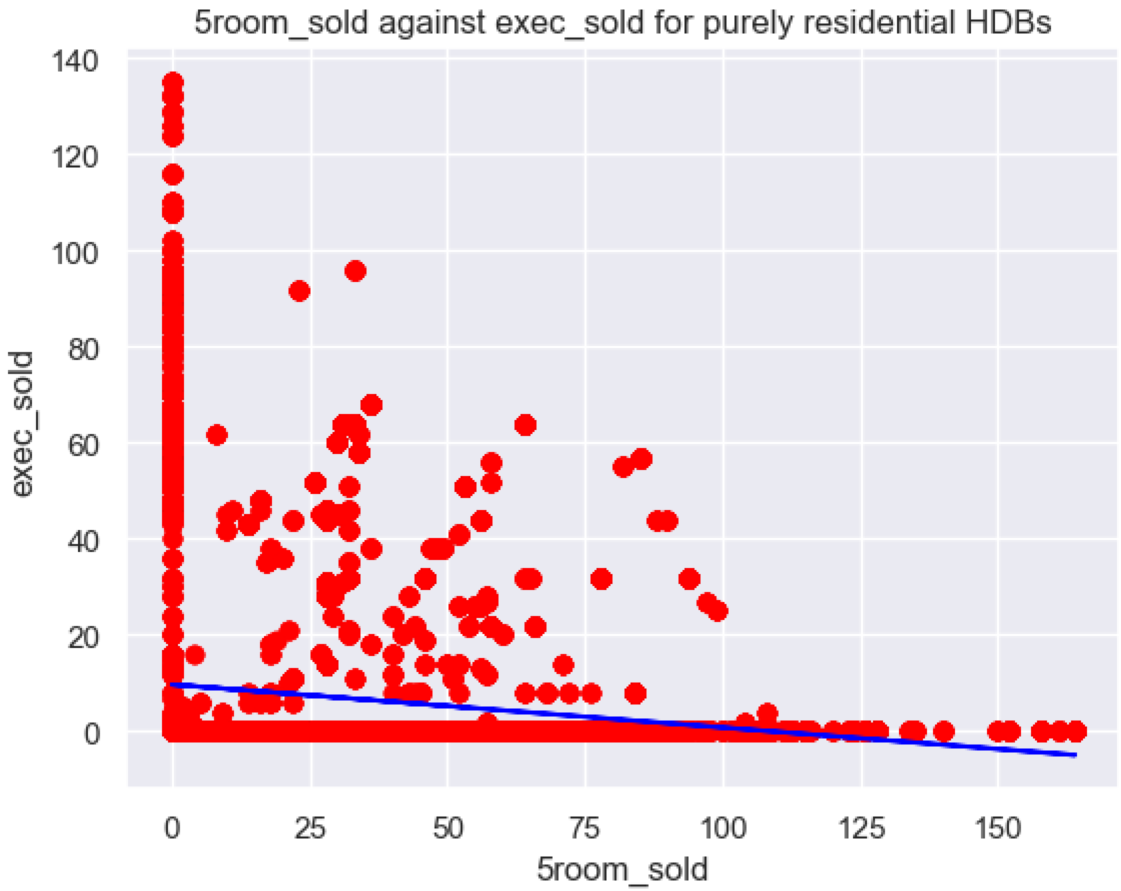

4room_sold against other_room_rental for purely residential HDBs
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
other_room_rental
0
50
100
150
200
250
300
4room_sold


4room_sold against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
25000
20000
15000
10000
5000
0
0
50
100
150
200
250
300
4room_sold

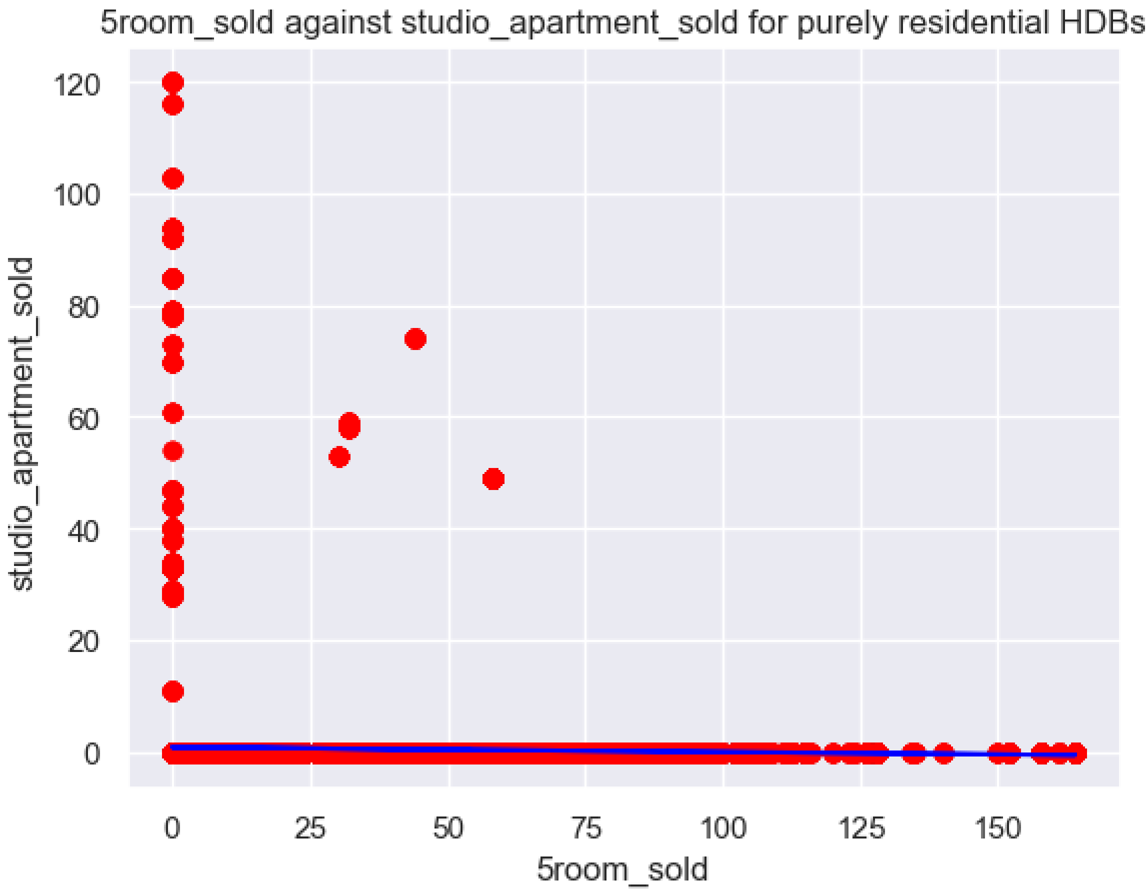

4room_sold against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
3000
2500
2000
1500
1000
500
0
0
50
100
150
200
250
300
4room_sold

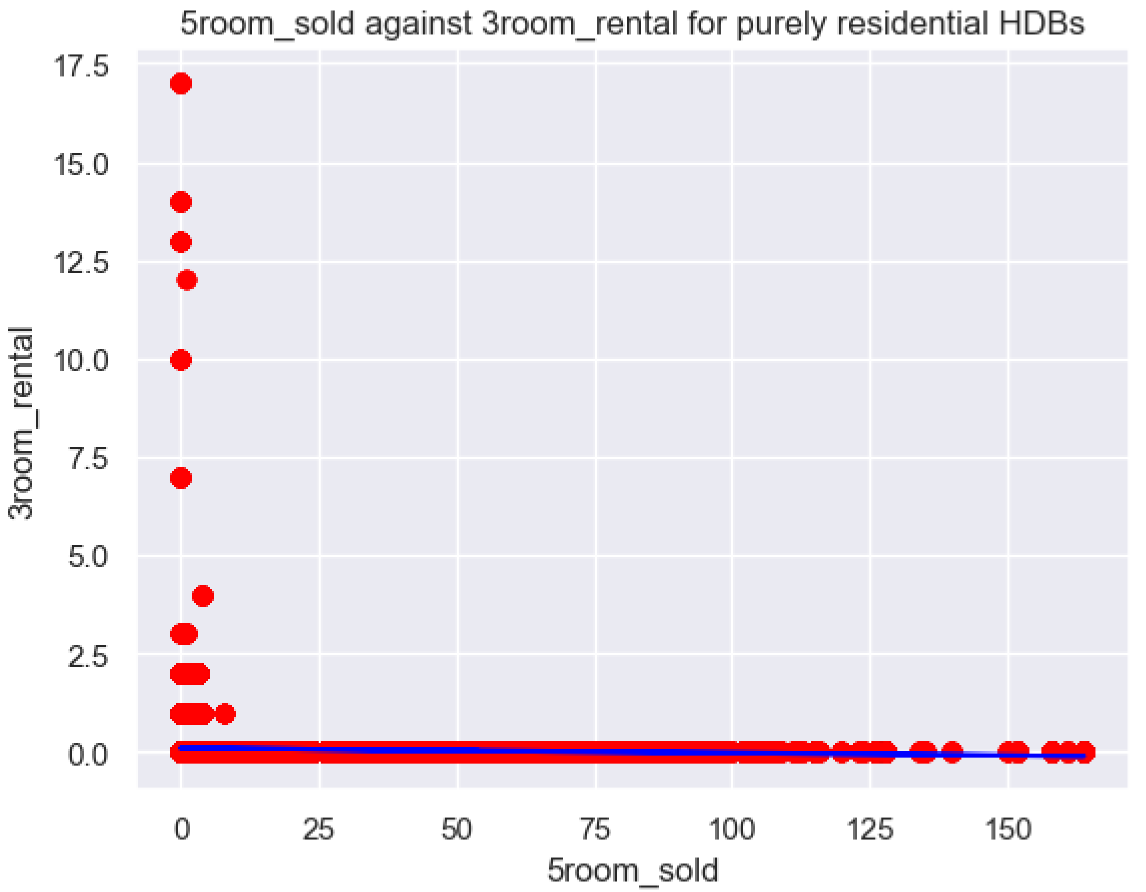

4room_sold against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
0
2
4
6
8
10
12
14
0
50
100
150
200
250
300
4room_sold


1e6
4room_sold against real_price for purely residential HDBs
real_price
0.2
0.4
0.6
0.8
1.0
1.2
0
50
100
150
200
250
300
4room_sold

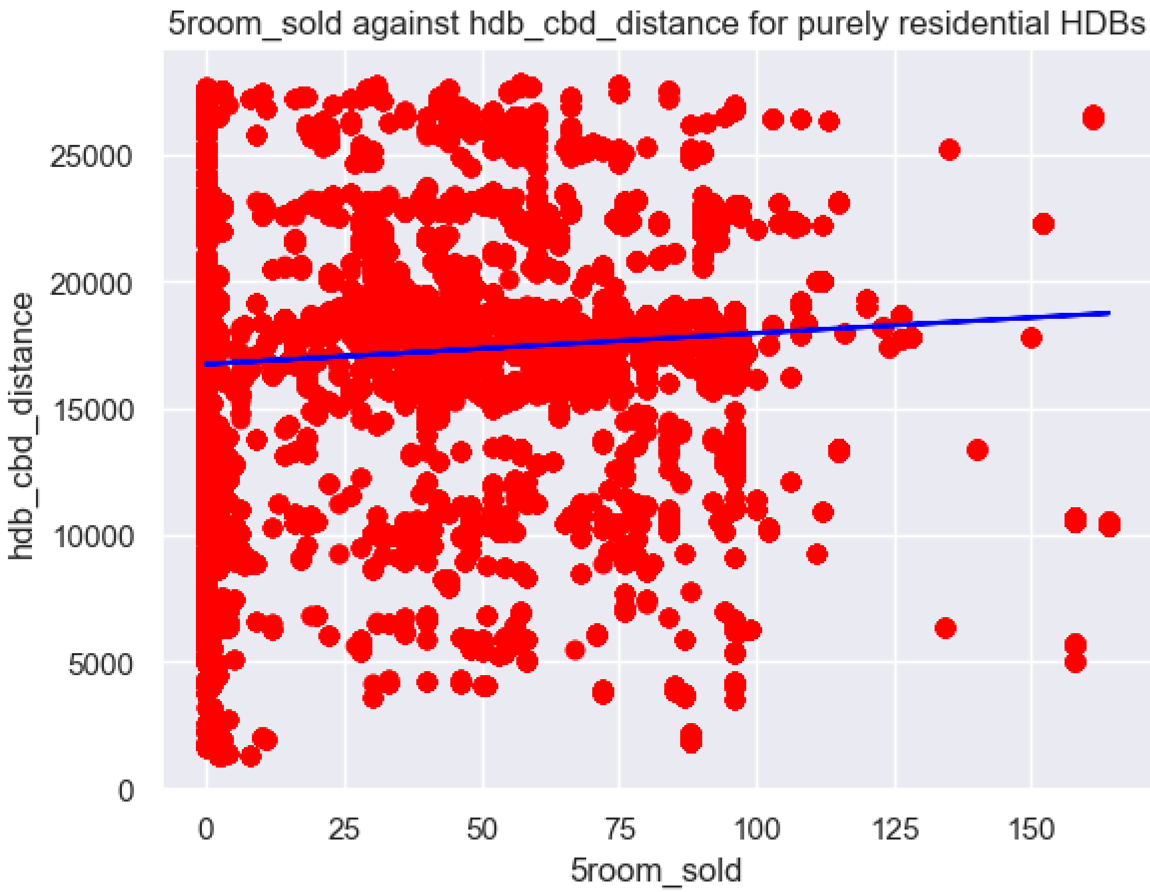

5room_sold against exec_sold for purely residential HDBs
exec_sold
0
20
40
60
80
100
120
140
5room_sold
0
25
50
75
100
125
150

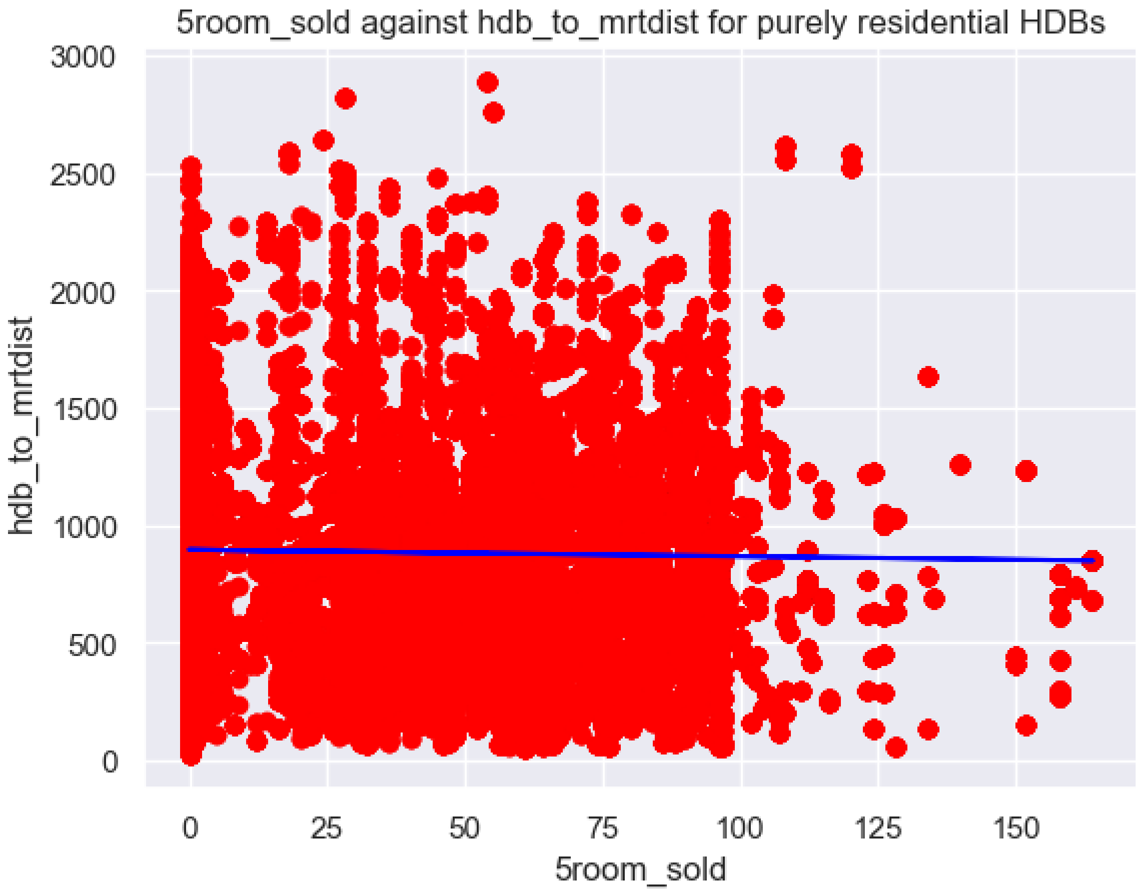

5room_sold against multigen_sold for purely residential HDBs
multigen_sold
0
10
20
30
40
50
5room_sold
0
25
50
75
100
125
150

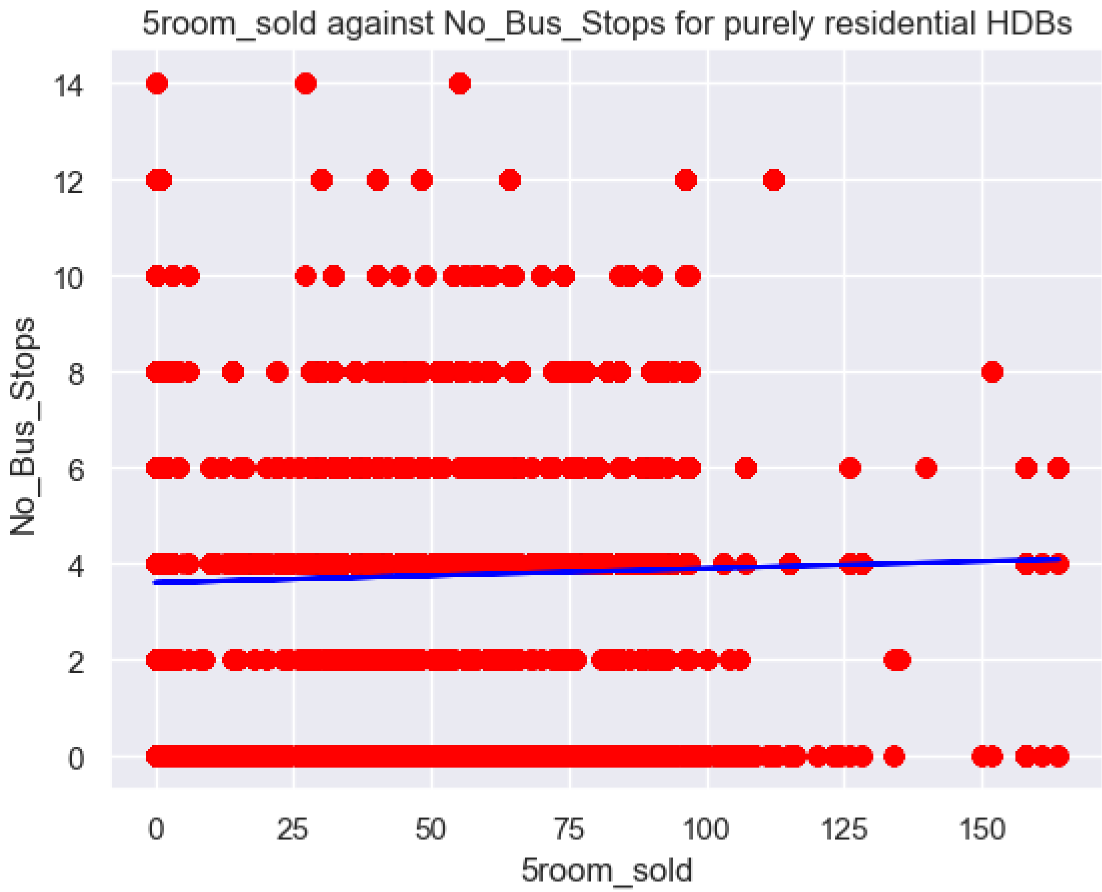

5room_sold against studio_apartment_sold for purely residential HDBs
studio_apartment_sold
120
100
80
60
40
20
0
0
25
50
75
100
125
150
5room_sold


5room_sold against 3room_rental for purely residential HDBs
3room_rental
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
25
50
75
100
125
150
5room_sold


5room_sold against other_room_rental for purely residential HDBs
other_room_rental
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
25
50
75
100
125
150
5room_sold

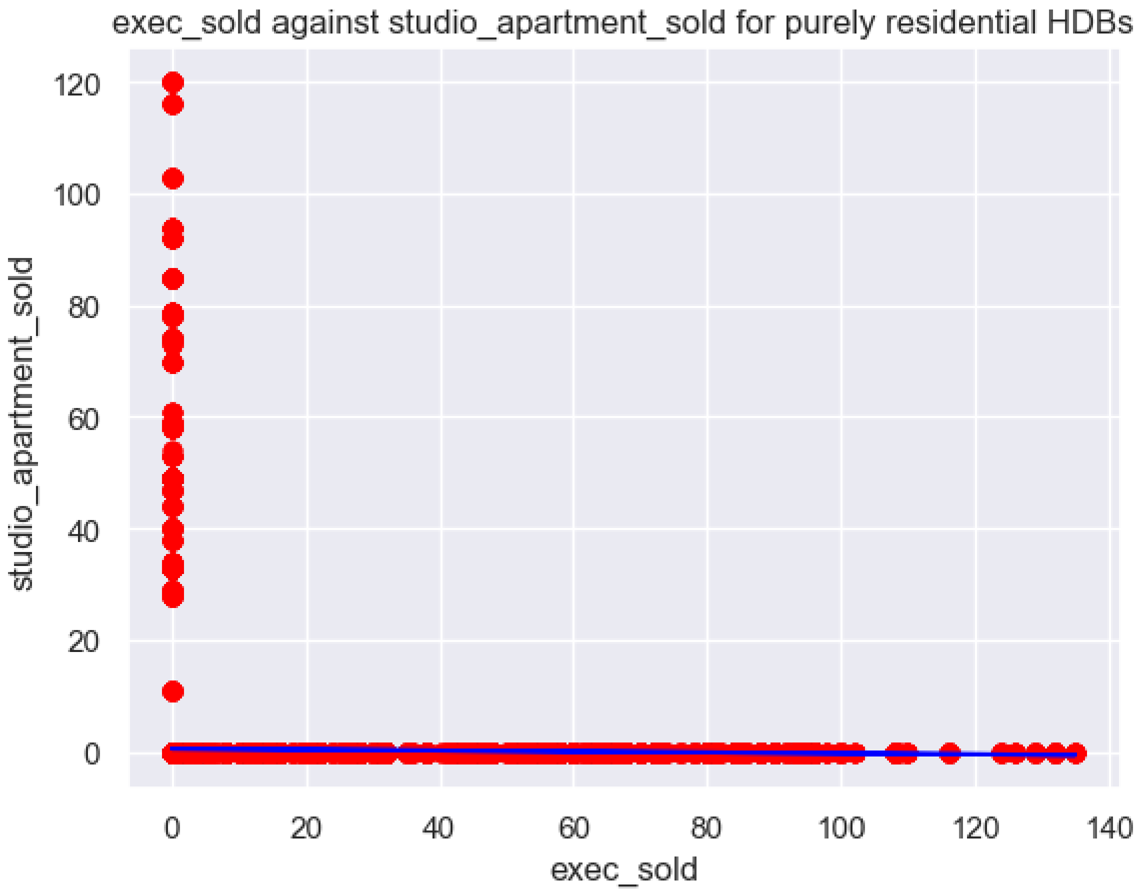

5room_sold against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
25000
20000
15000
10000
5000
0
0
25
50
75
100
125
150
5room_sold

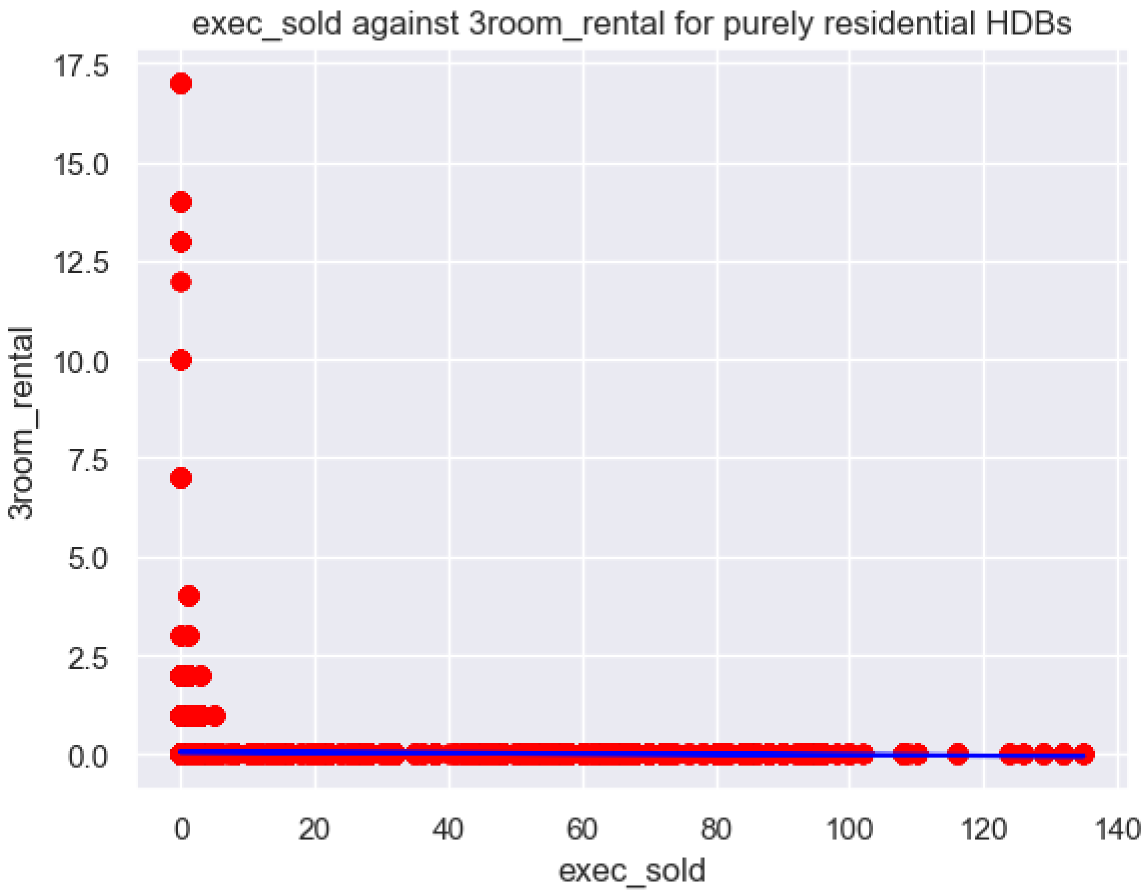

5room_sold against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
3000
2500
2000
1500
1000
500
0
0
25
50
75
100
125
150
5room_sold

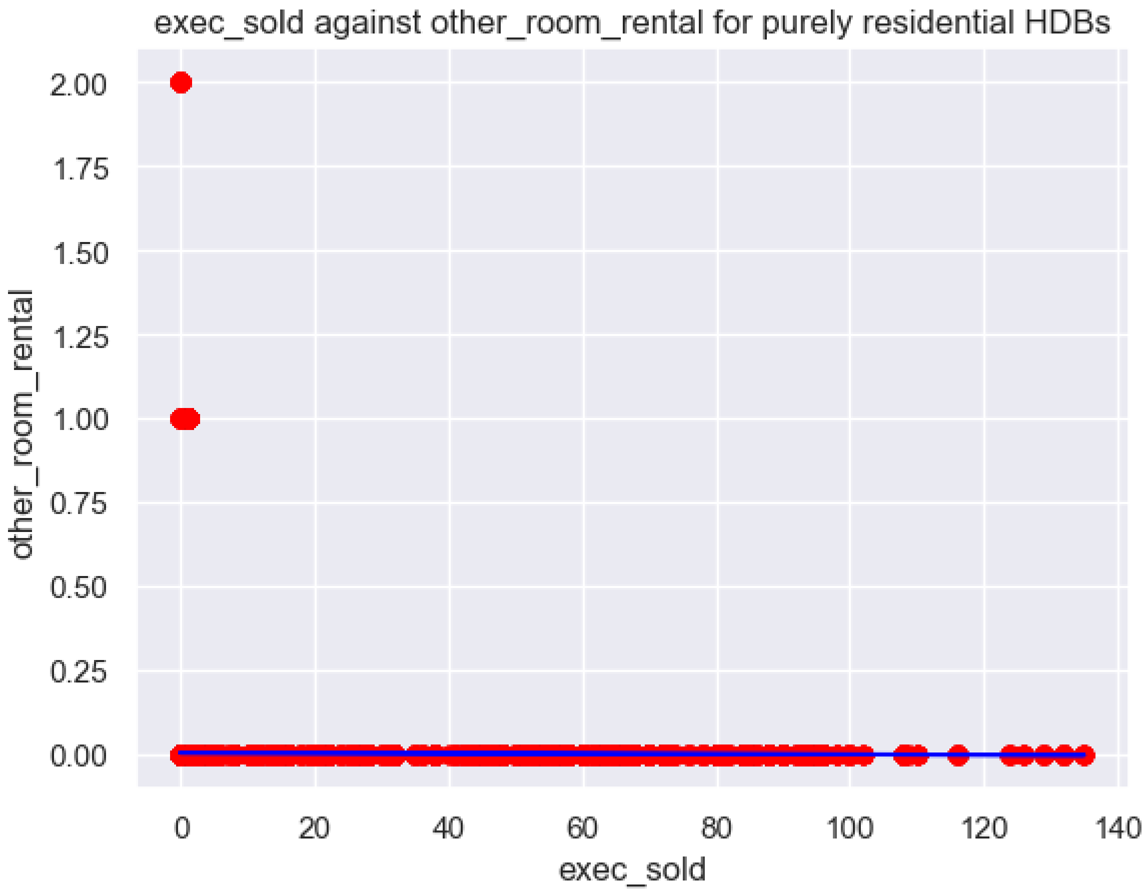

5room_sold against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
14
12
10
8
6
4
2
0
0
25
50
75
100
125
150
5room_sold


5room_sold against real_price for purely residential HDBs
1e6
real_price
1.2
1.0
0.8
0.6
0.4
0.2
0
25
50
75
100
125
150
5room_sold


exec_sold against multigen_sold for purely residential HDBs
multigen_sold
50
40
30
20
10
0
0
20
40
60
80
100
120
140
exec_sold


exec_sold against studio_apartment_sold for purely residential HDBs
studio_apartment_sold
exec_sold
0
20
40
60
80
100
120
140


exec_sold against 3room_rental for purely residential HDBs
3room_rental
exec_sold
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

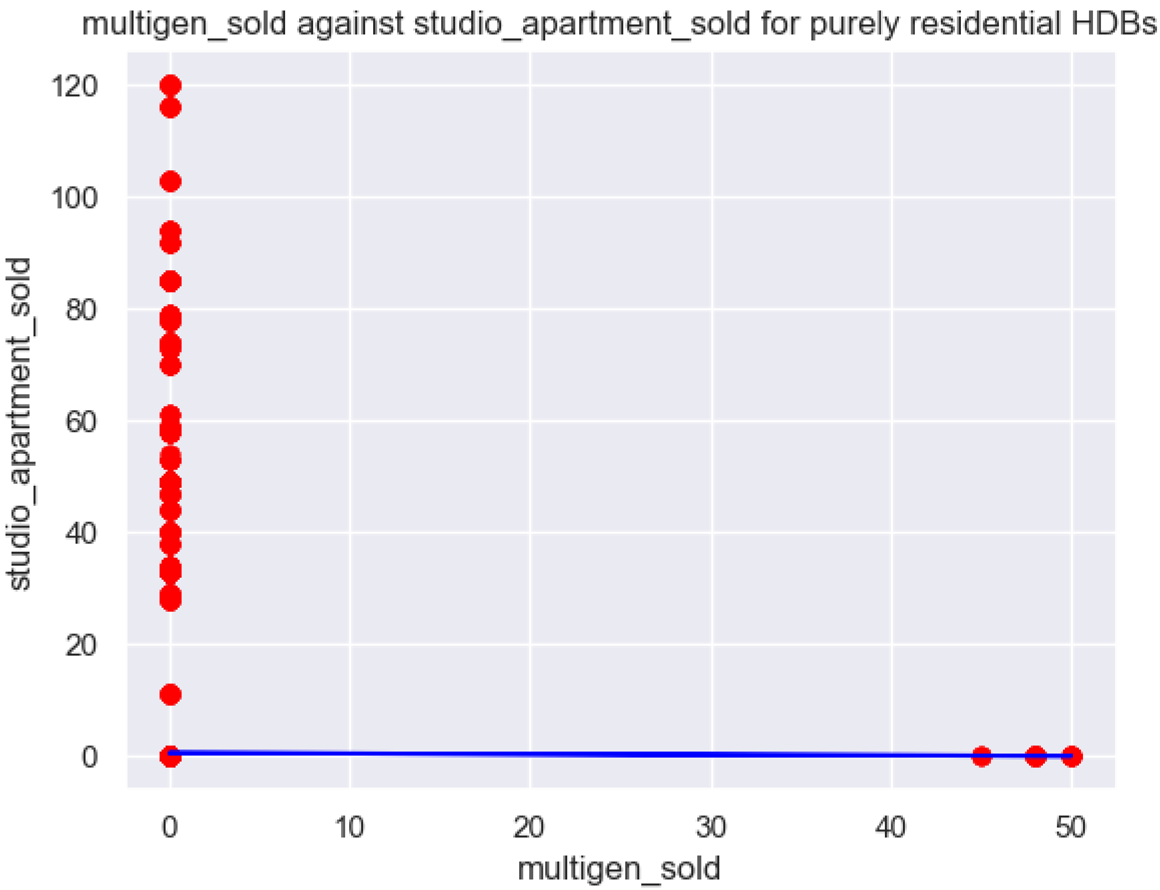

exec_sold against other_room_rental for purely residential HDBs
other_room_rental
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
20
40
60
80
100
120
140
exec_sold

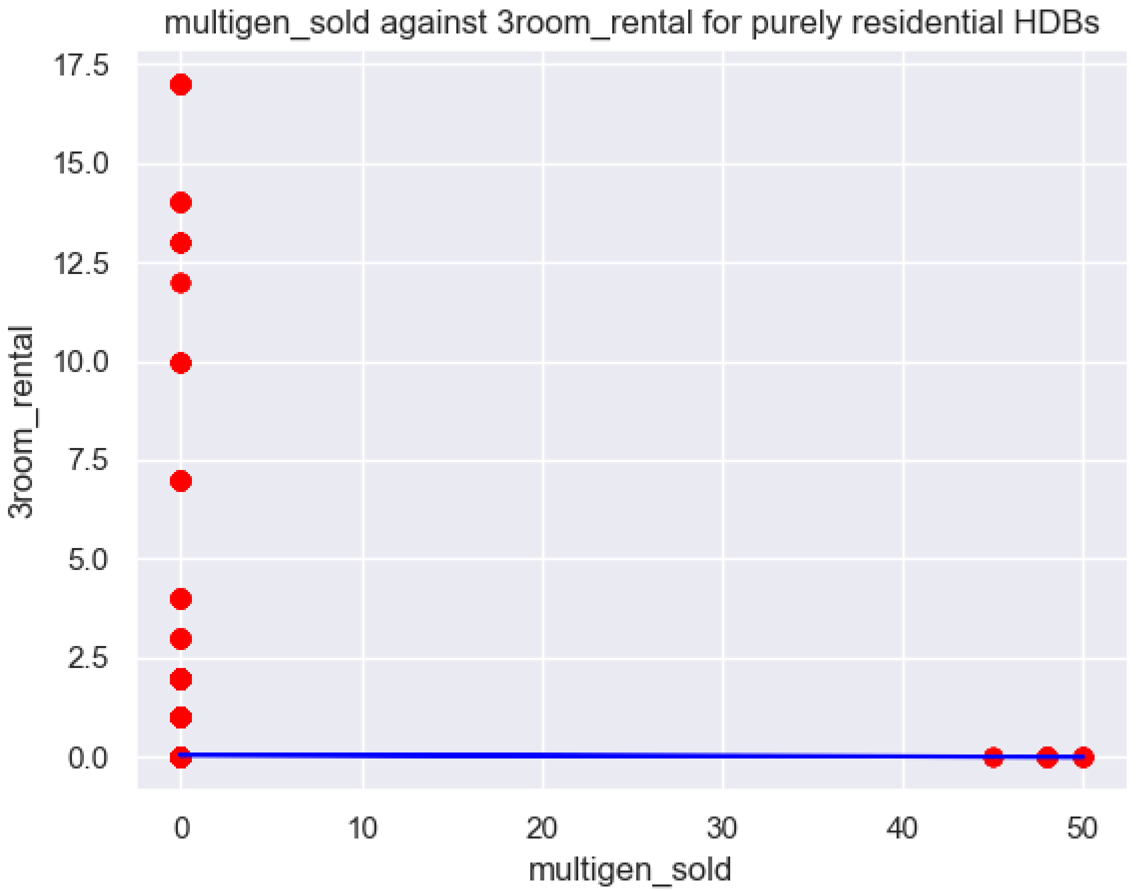

exec_sold against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
25000
20000
15000
10000
5000
0
0
20
40
60
80
100
120
140
exec_sold


exec_sold against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
exec_sold
0
500
1000
1500
2000
2500
3000
0
20
40
60
80
100
120
140

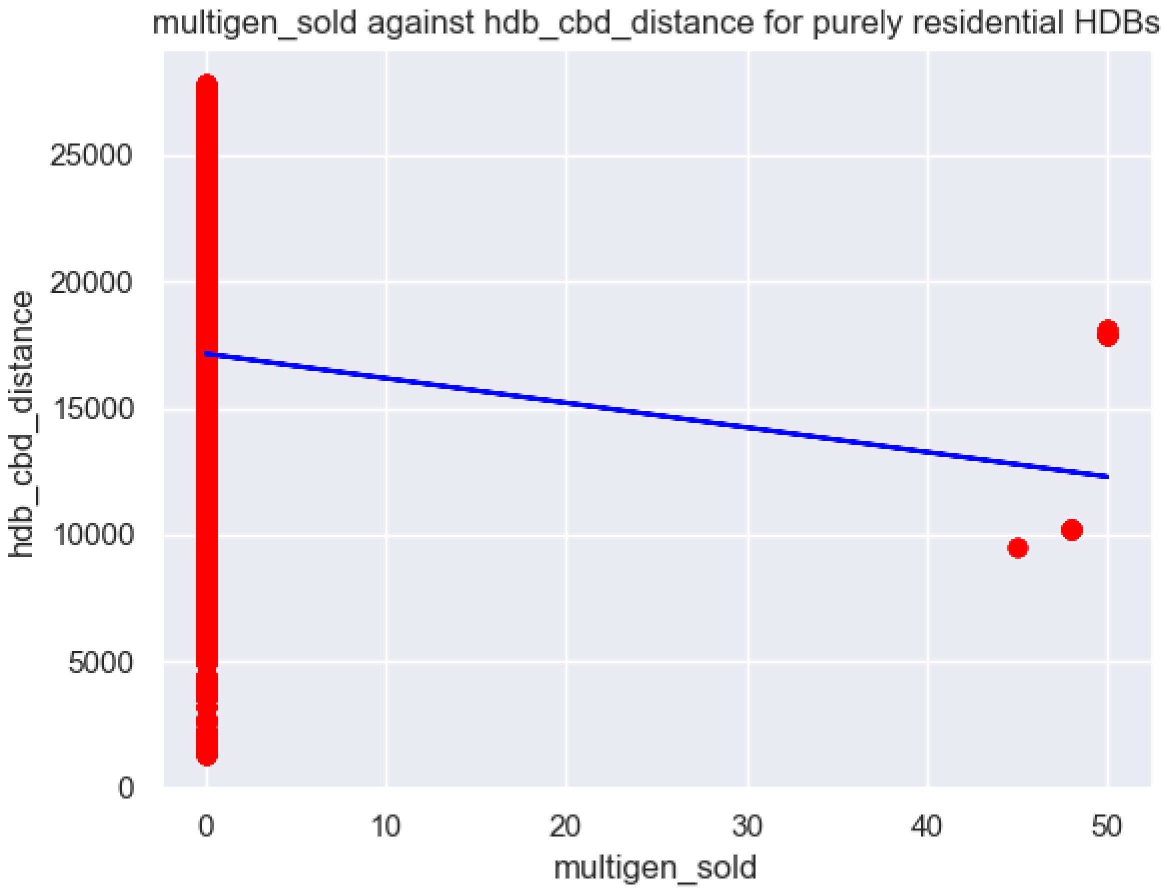

exec_sold against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
exec_sold
0
2
4
6
8
10
12
14
0
20
40
60
80
100
120
140


exec_sold against real_price for purely residential HDBs
1e6
1.2
1.0
0.8
0.6
0.4
0.2
real_price
0
20
40
60
80
100
120
140
exec_sold

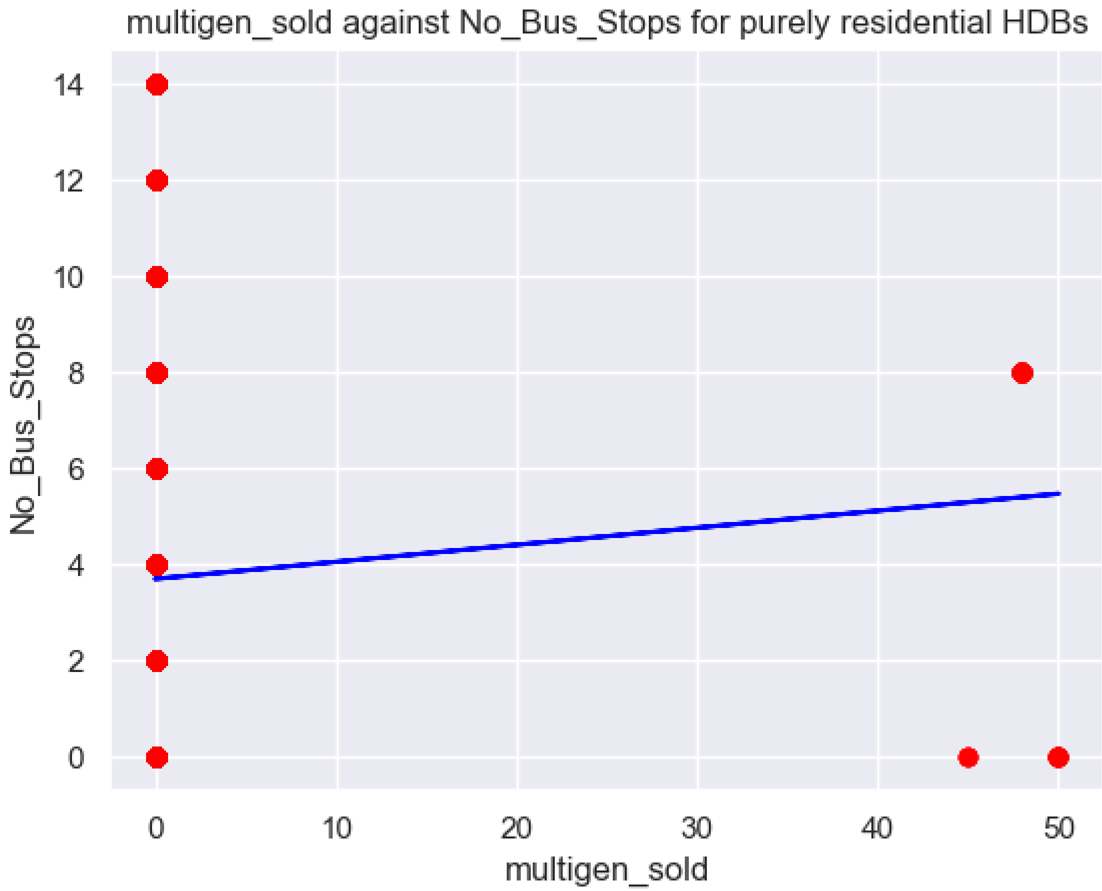

multigen_sold against studio_apartment_sold for purely residential HDBs
120
100
80
60
40
20
0
studio_apartment_sold
0
10
20
30
40
50
multigen_sold


multigen_sold against 3room_rental for purely residential HDBs
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
3room_rental
0
10
20
30
40
50
multigen_sold

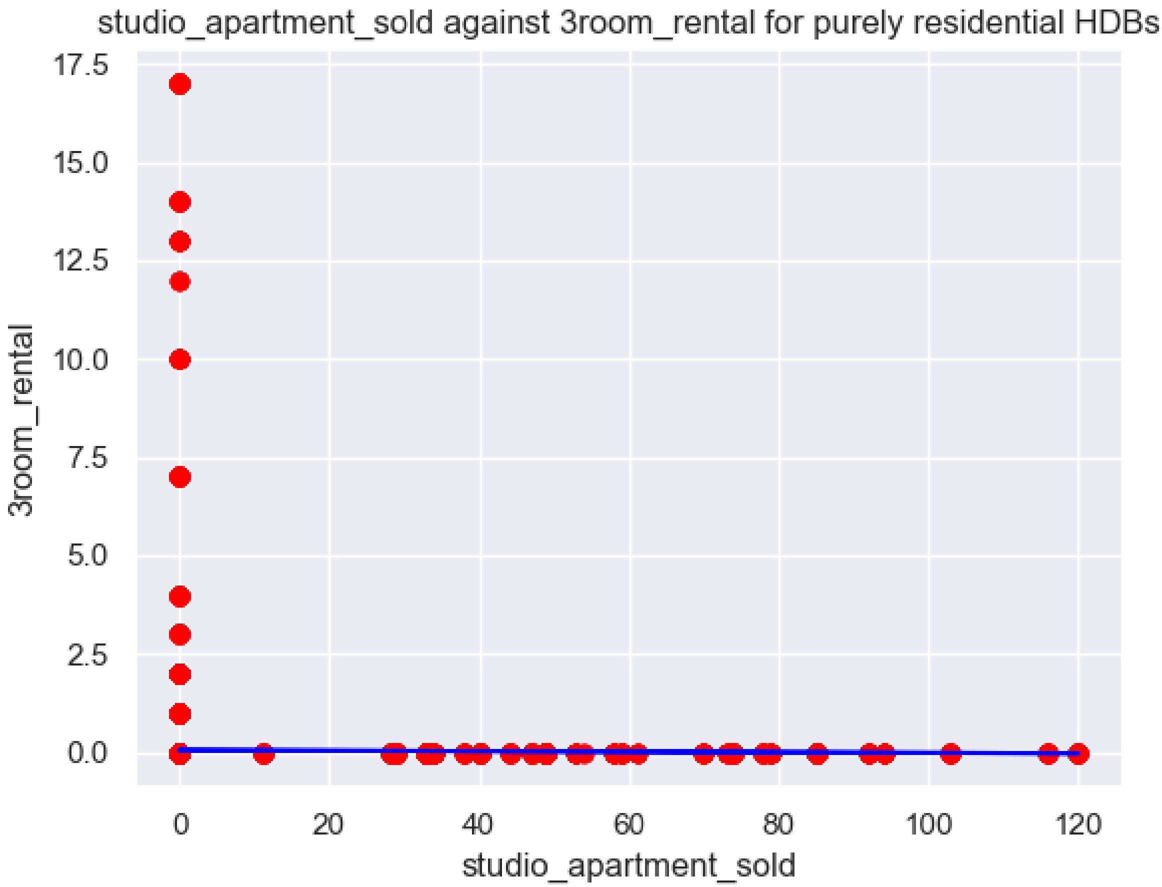

multigen_sold against other_room_rental for purely residential HDBs
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
other_room_rental
0
10
20
30
40
50
multigen_sold


multigen_sold against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
0
5000
10000
15000
20000
25000
0
10
20
30
40
50
multigen_sold


multigen_sold against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
0
500
1000
1500
2000
2500
3000
0
10
20
30
40
50
multigen_sold


multigen_sold against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
14
12
10
8
6
4
2
0
0
10
20
30
40
50
multigen_sold

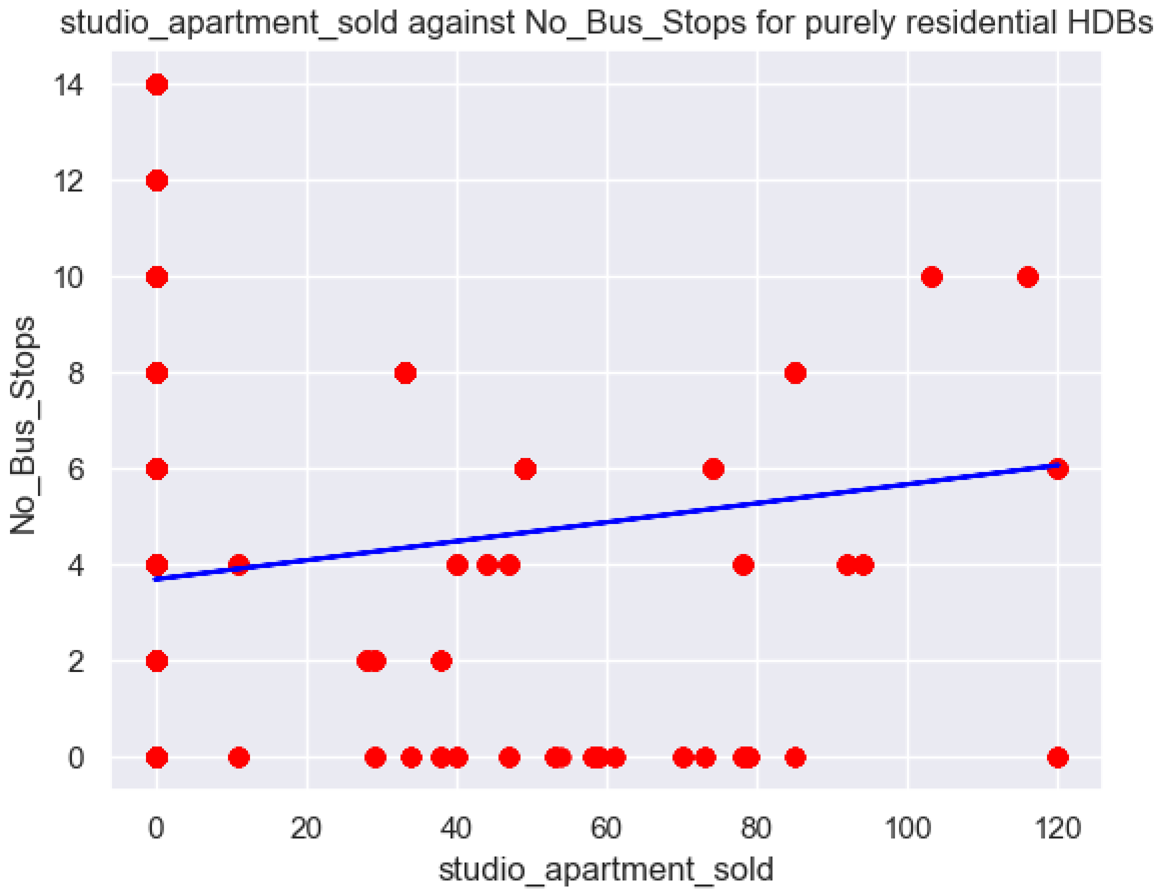

1e6
multigen_sold against real_price for purely residential HDBs
real_price
1.2
1.0
0.8
0.6
0.4
0.2
0
10
20
30
40
50
multigen_sold


studio_apartment_sold against 3room_rental for purely residential HDBs
3room_rental
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
20
40
60
80
100
120
studio_apartment_sold

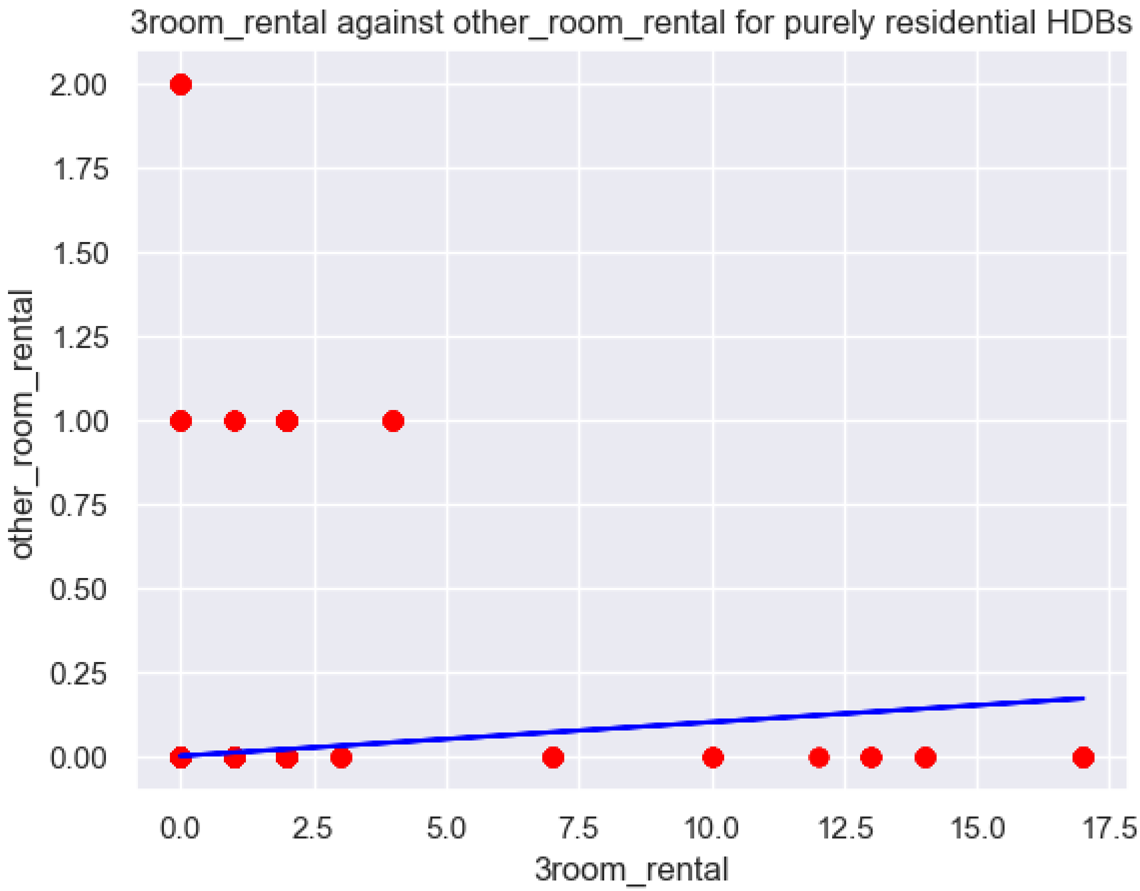

studio_apartment_sold against other_room_rental for purely residential HDBs
other_room_rental
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
20
40
60
80
100
120
studio_apartment_sold

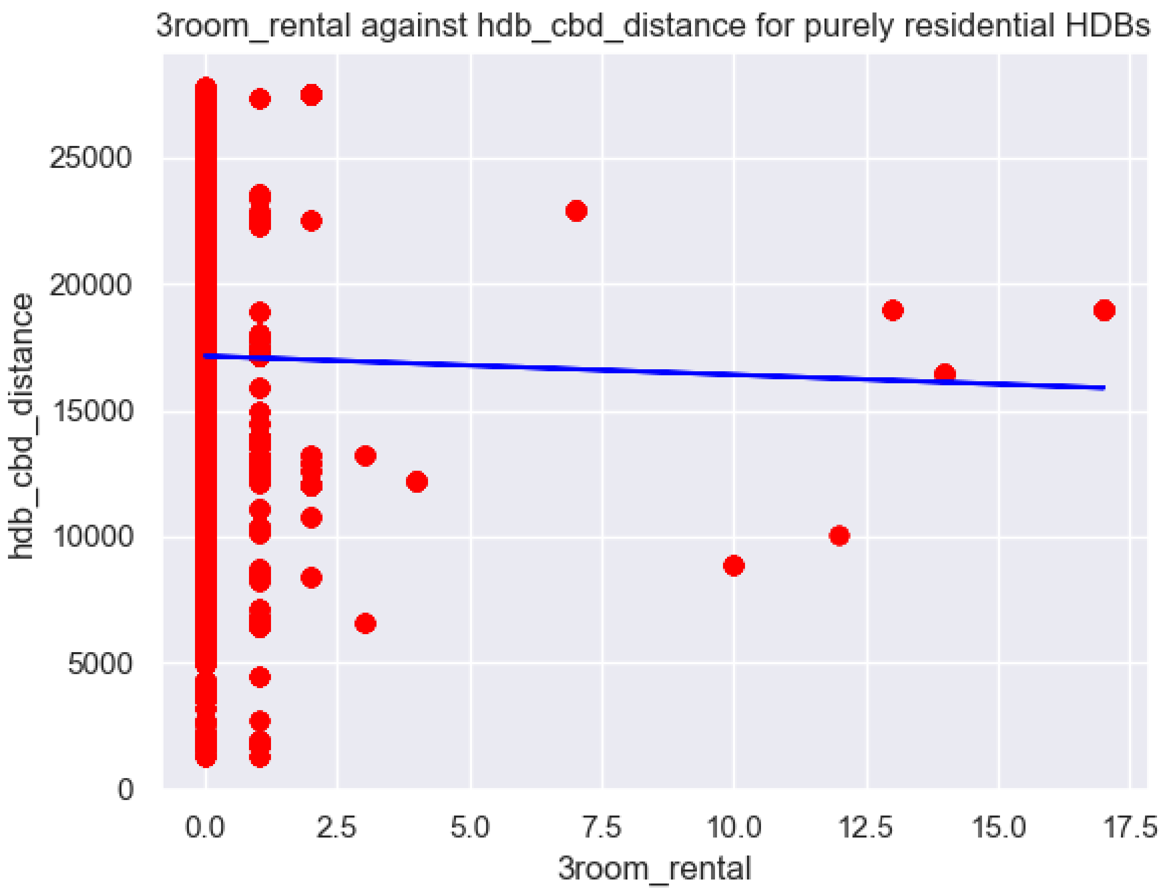

studio_apartment_sold against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
25000
20000
15000
10000
5000
0
0
20
40
60
80
100
120
studio_apartment_sold

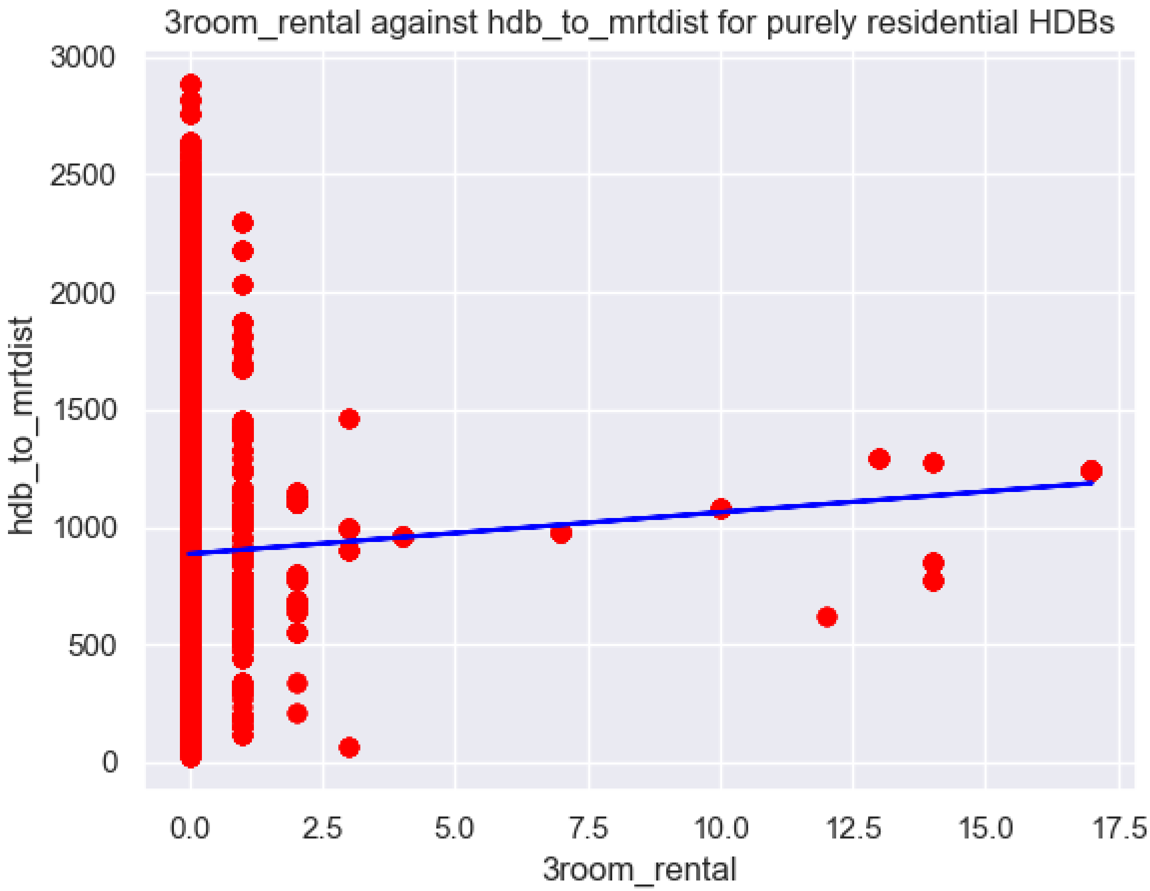

studio_apartment_sold against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
3000
2500
2000
1500
1000
500
0
0
20
40
60
80
100
120
studio_apartment_sold

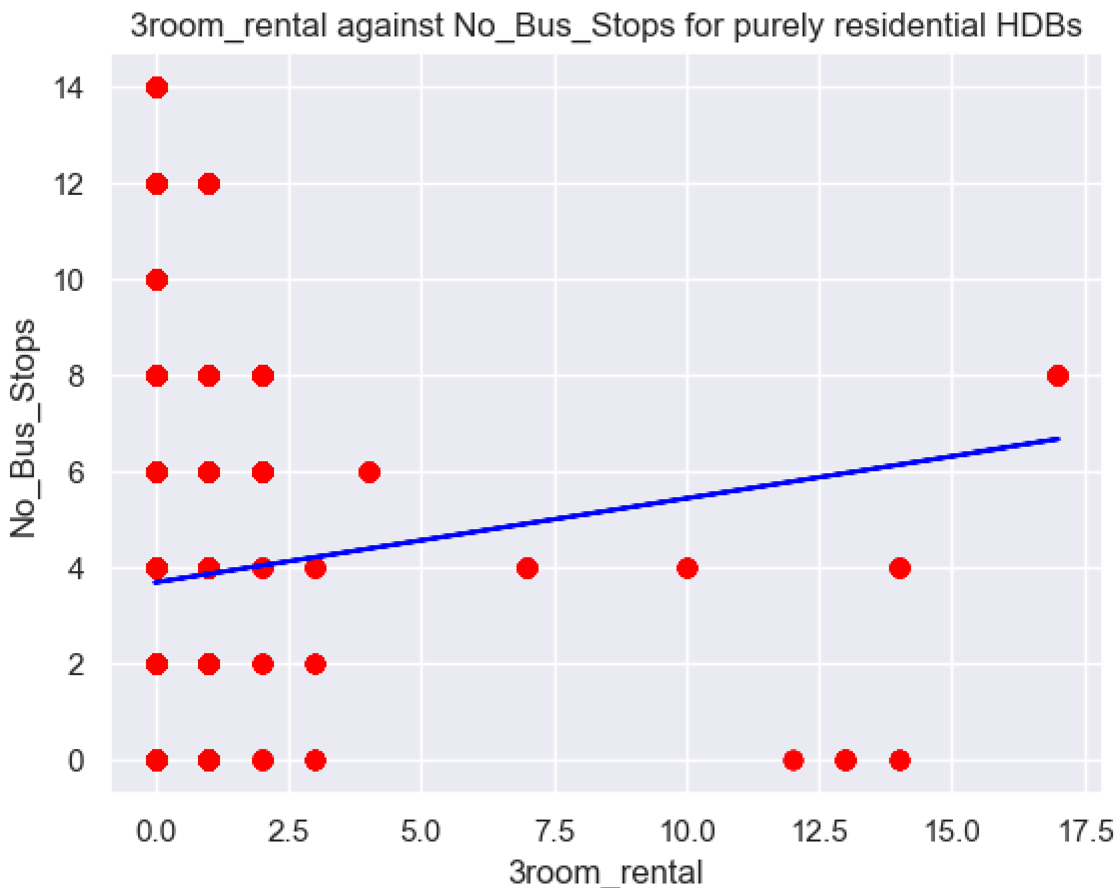

studio_apartment_sold against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
studio_apartment_sold
0
2
4
6
8
10
12
14
0
20
40
60
80
100
120

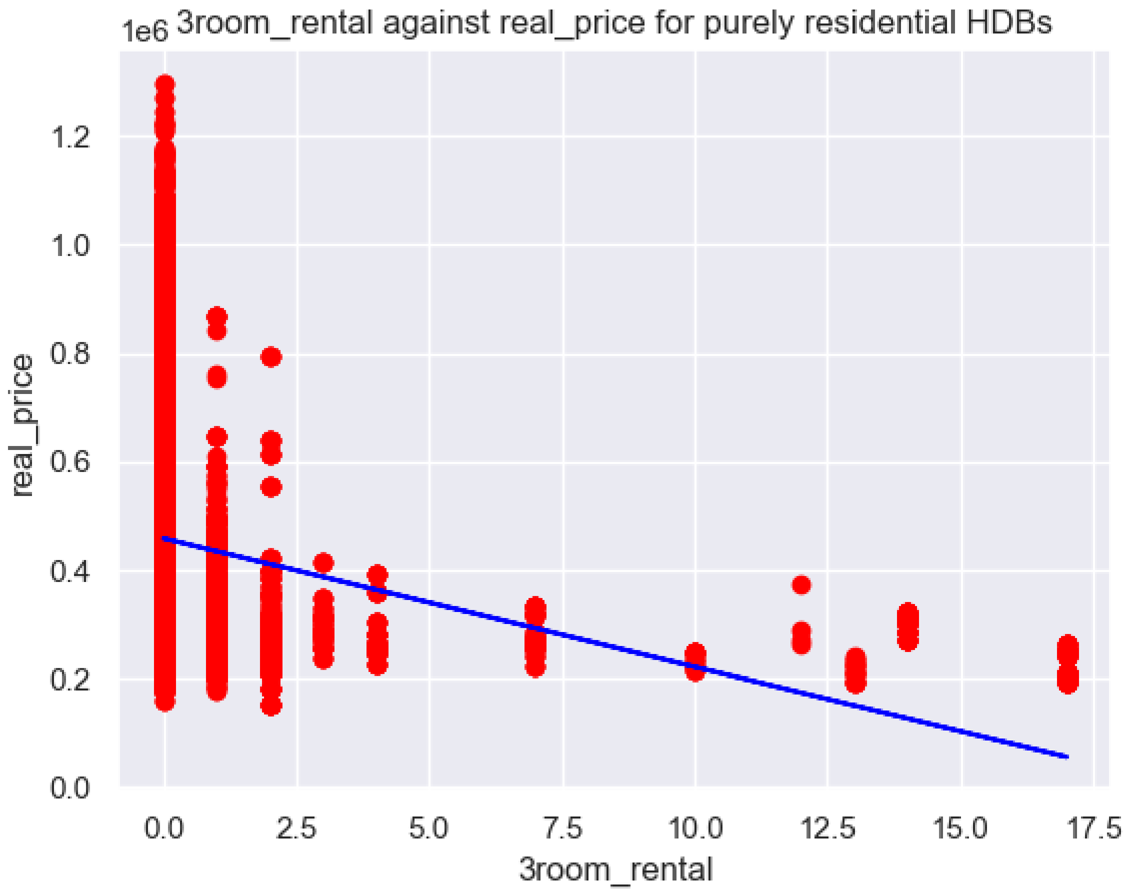

studio_apartment_sold against real_price for purely residential HDBs
1e6
real_price
studio_apartment_sold
0.2
0.4
0.6
0.8
1.0
1.2
0
20
40
60
80
100
120


3room_rental against other_room_rental for purely residential HDBs
other_room_rental
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
3room_rental


3room_rental against hdb_cbd_distance for purely residential HDBs
hdb_cbd_distance
25000
20000
15000
10000
5000
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
3room_rental

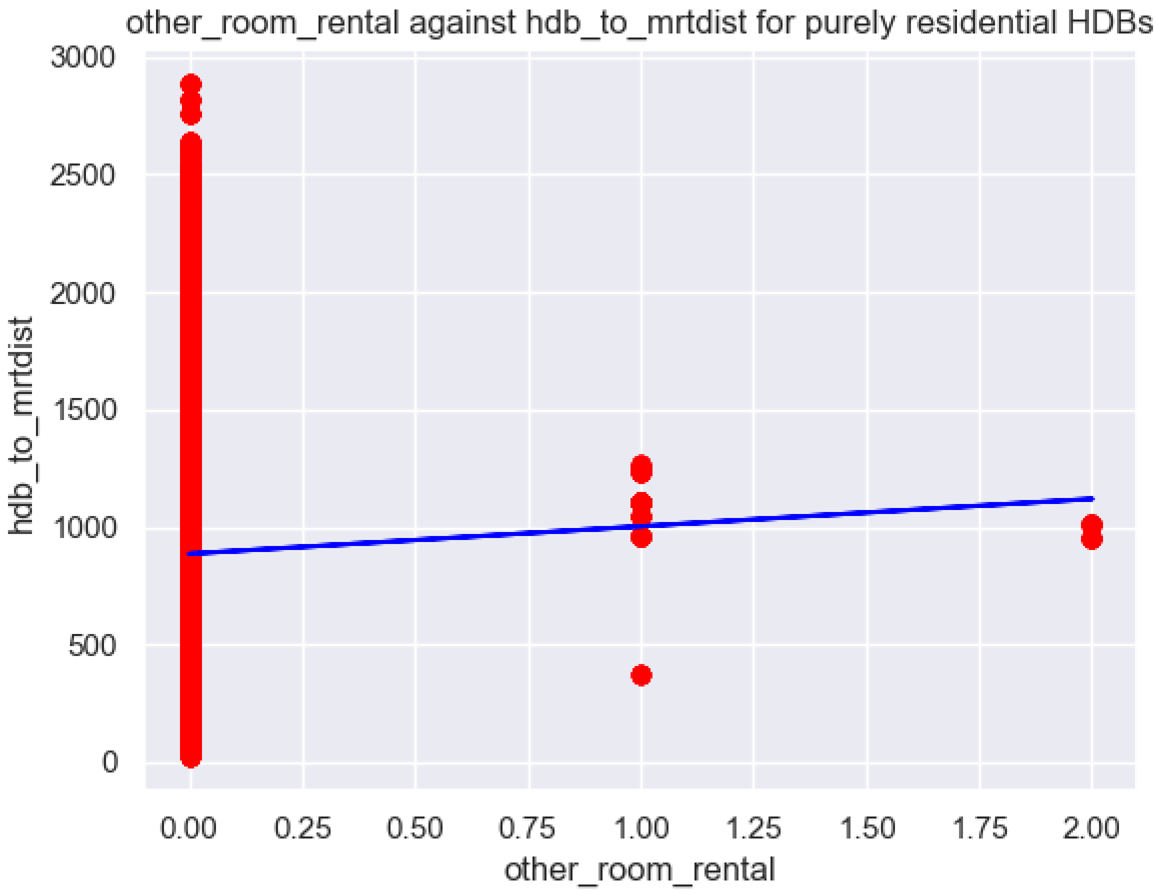

3room_rental against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
3000
2500
2000
1500
1000
500
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
3room_rental


3room_rental against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
14
12
10
8
6
4
2
0
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
3room_rental

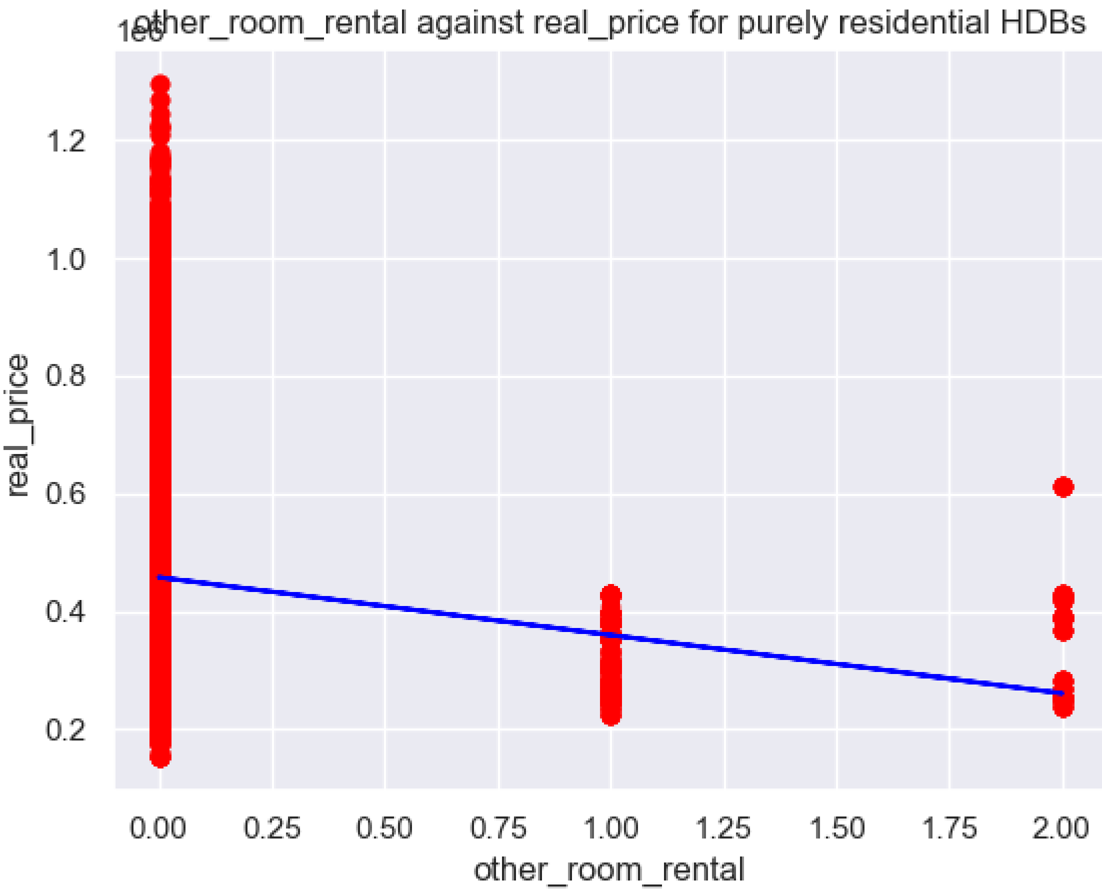


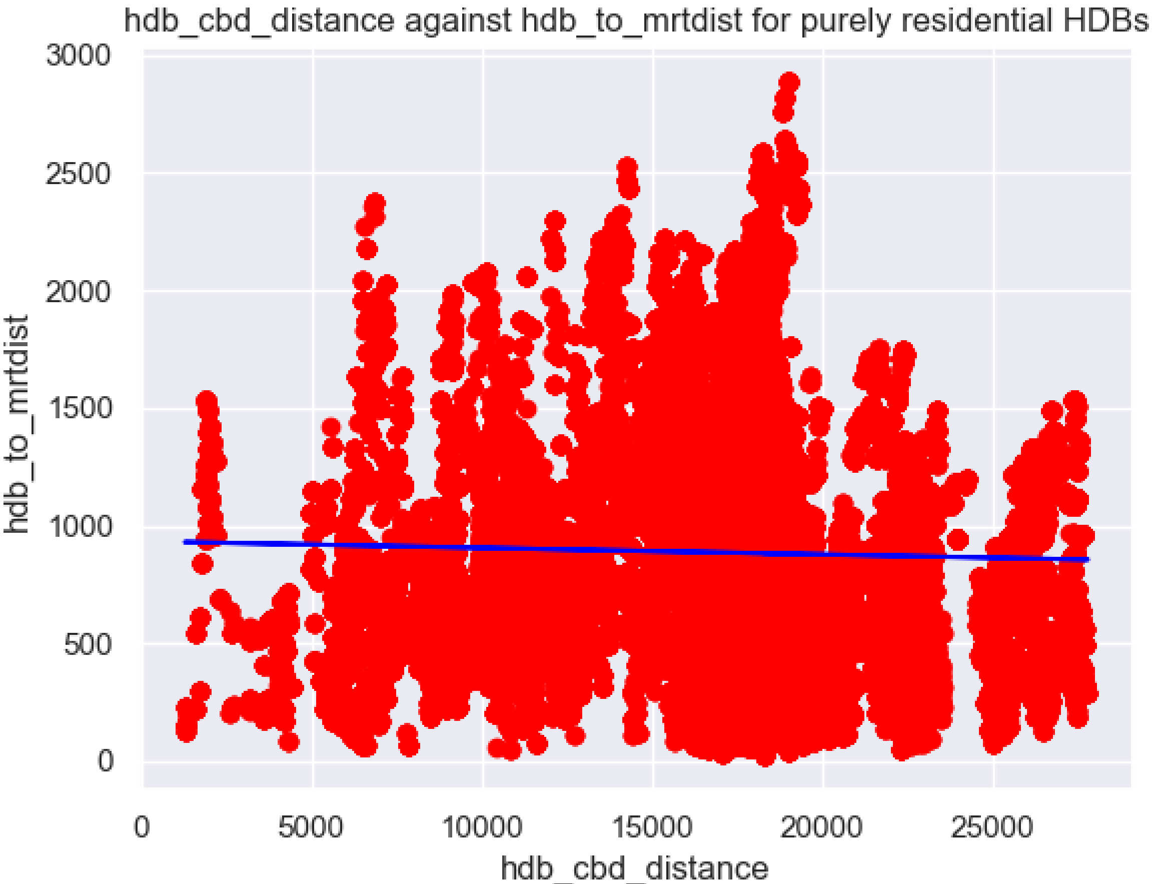
3room_rental against real_price for purely residential HDBs
1e6
1.2
1.0
0.8
0.6
0.4
0.2
0.0
real_price
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
3room_rental

other_room_rental against hdb_cbd_distance for purely residential HDBs
25000
20000
15000
10000
5000
0
hdb_cbd_distance
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
other_room_rental


other_room_rental against hdb_to_mrtdist for purely residential HDBs
hdb_to_mrtdist
3000
2500
2000
1500
1000
500
0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
other_room_rental

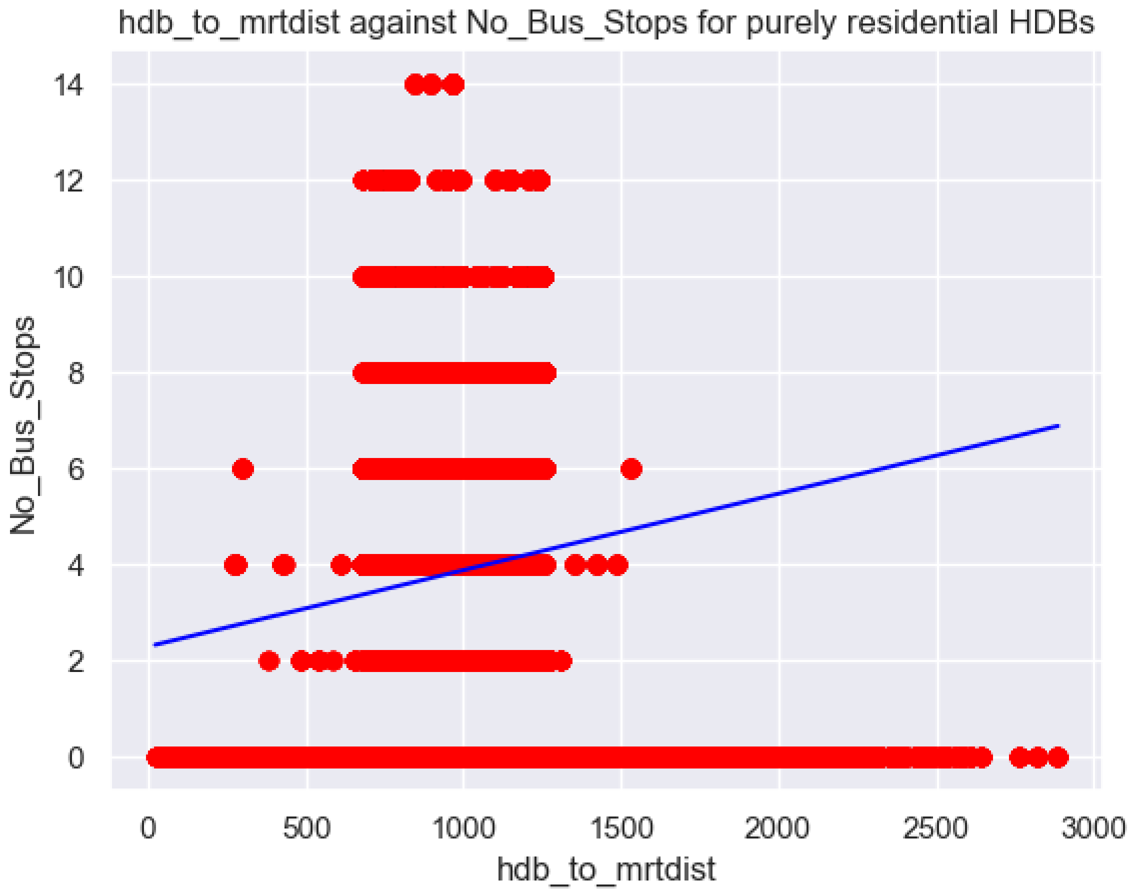

other_room_rental against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
14
12
10
8
6
4
2
0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
other_room_rental

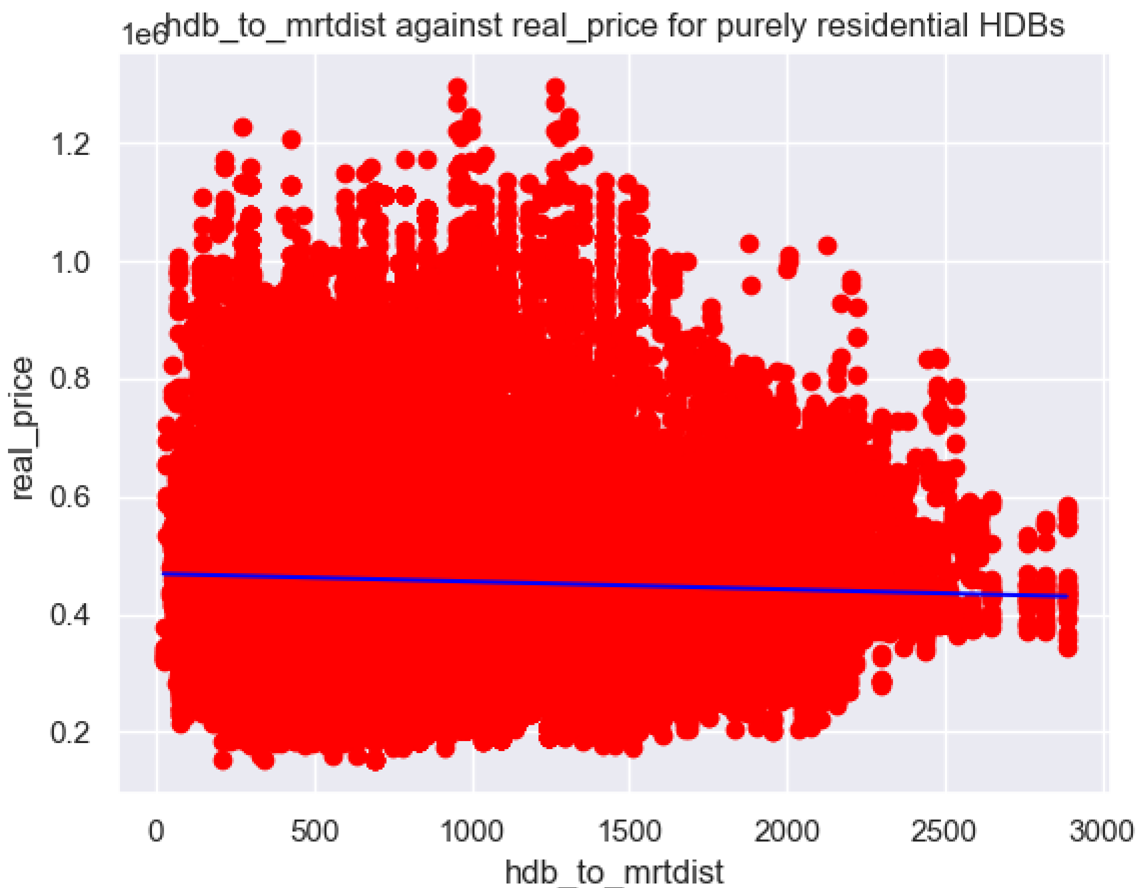

other_room_rental against real_price for purely residential HDBs
1e6
1.2
1.0
0.8
0.6
0.4
0.2
real_price
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
other_room_rental

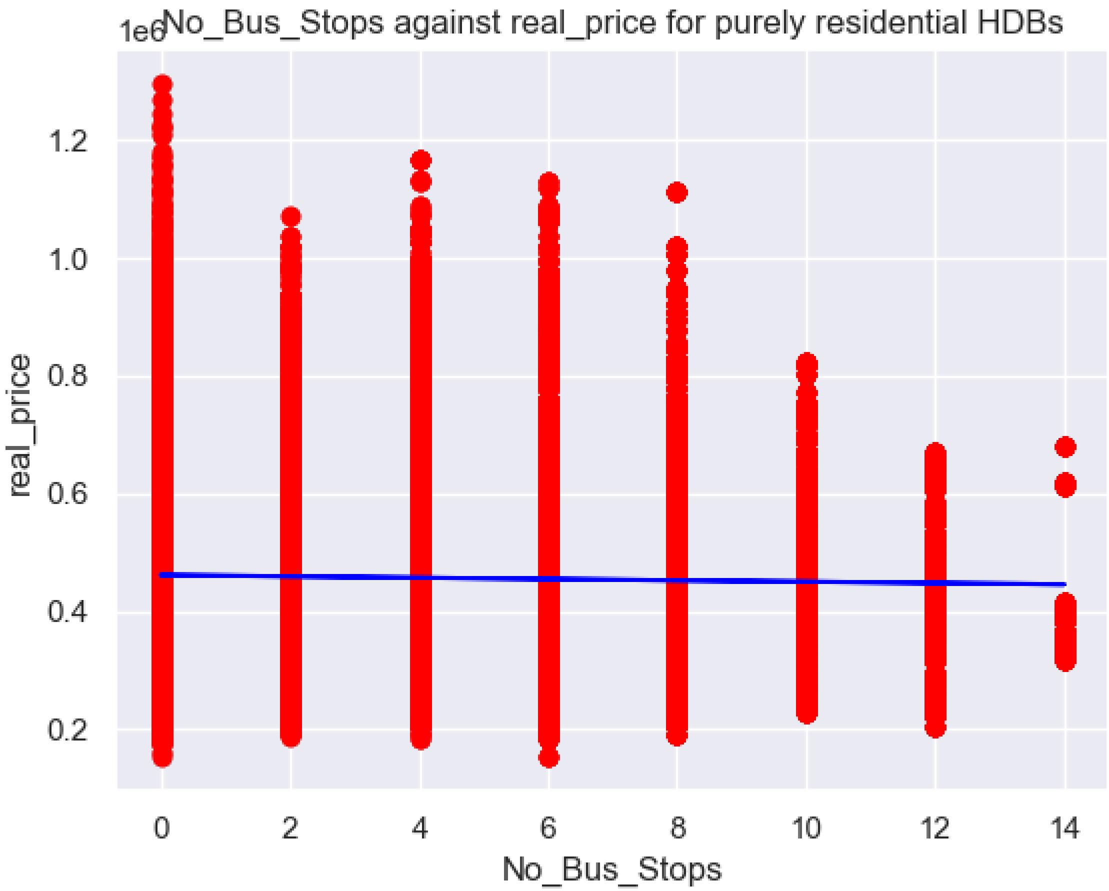

hdb_cbd_distance against hdb_to_mrtdist for purely residential HDBs
3000
2500
2000
1500
1000
500
0
hdb_to_mrtdist
0
5000
10000
15000
20000
25000
hdb_cbd_distance


hdb_cbd_distance against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
hdb_cbd_distance
0
2
4
6
8
10
12
14
0
5000
10000
15000
20000
25000


hdb_cbd_distance against real_price for purely residential HDBs
1e6
real_price
hdb_cbd_distance
0.2
0.4
0.6
0.8
1.0
1.2
0
5000
10000
15000
20000
25000


hdb_to_mrtdist against No_Bus_Stops for purely residential HDBs
No_Bus_Stops
0
2
4
6
8
10
12
14
hdb_to_mrtdist
0
500
1000
1500
2000
2500
3000


hdb_to_mrtdist against real_price for purely residential HDBs
1e6
real_price
0.2
0.4
0.6
0.8
1.0
1.2
hdb_to_mrtdist
0
500
1000
1500
2000
2500
3000



Explanation of the above graphs

1. real price is affected by floor_area_sqm at R squared value of 0.63. But logically it does affect.
2. total dwelling units is affected by max floor at R squared value of 0.42
3. real price is affected by max floor at R squared value of 0.37
4. 4 room and 5 room are only sold on buildings with higher max floor levels.
5. real price is affect by hdb cbd distance with R square value of -0.3

Put it together:

1. Real price could be affect by floor area sqm, max floor, hdb cbd distance
2. max floor could be affected or correlated with the 4 room and 5 room houses and total dwelling units

In [7]:
```
X = residential[['floor_area_sqm', 'max_floor_lvl',
        'hdb_cbd_distance']]
y = residential[['real_price']]
X_constant = sm.add_constant(X)
lr = sm.OLS(y, X_constant.astype(float)).fit()
print(lr.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:             real_price   R-squared:                       0.646
Model:                            OLS   Adj. R-squared:                  0.646
Method:                 Least Squares   F-statistic:                 9.678e+04
Date:                Sun, 12 Mar 2023   Prob (F-statistic):               0.00
Time:                        18:06:33   Log-Likelihood:            -2.0212e+06
No. Observations:              158904   AIC:                         4.042e+06
Df Residuals:                  158900   BIC:                         4.042e+06
Df Model:                           3
Covariance Type:            nonrobust
================================================================================
==
                       coef    std err          t      P>|t|      [0.025      0.97
5]
--------------------------------------------------------------------------------
--
const             1.058e+05   1351.696     78.247      0.000    1.03e+05    1.08e+
05
floor_area_sqm    4363.1962      9.640    452.627      0.000    4344.303    4382.0
90
max_floor_lvl     6771.3365     36.002    188.080      0.000    6700.772    6841.9
01
hdb_cbd_distance   -11.2945      0.048   -234.611      0.000     -11.389     -11.2
00
==============================================================================
Omnibus:                    14515.635   Durbin-Watson:                   0.227
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            19928.742
Skew:                           0.753   Prob(JB):                         0.00
Kurtosis:                       3.860   Cond. No.                     1.18e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
[2] The condition number is large, 1.18e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

In [8]:

```
X = residential[['4room_sold', '5room_sold']]
y = residential[['total_dwelling_units']]
X_constant = sm.add_constant(X)
lr = sm.OLS(y, X_constant.astype(float)).fit()
print(lr.summary())
```

```
                            OLS Regression Results
================================================================================
Dep. Variable:     total_dwelling_units   R-squared:                       0.192
Model:                              OLS   Adj. R-squared:                  0.192
Method:                   Least Squares   F-statistic:                 1.894e+04
Date:                  Sun, 12 Mar 2023   Prob (F-statistic):               0.00
Time:                          18:06:33   Log-Likelihood:           -8.0245e+05
No. Observations:                158904   AIC:                       1.605e+06
Df Residuals:                    158901   BIC:                       1.605e+06
Df Model:                             2
Covariance Type:              nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         85.6576      0.184    465.831      0.000      85.297      86.018
4room_sold     0.4161      0.002    176.896      0.000       0.411       0.421
5room_sold     0.2696      0.003    103.046      0.000       0.264       0.275
==============================================================================
Omnibus:                    55613.054   Durbin-Watson:                   0.081
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           217774.619
Skew:                           1.726   Prob(JB):                         0.00
Kurtosis:                       7.580   Cond. No.                         139.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
```

Confirming the above, we can see that real price is affected by floor area sqm, max floor, hdb cbd distance. So let us remove real price. Other than that, there is nothing to worry about total dwelling units, 4 room, 5 room and max floor as the R-squared is low.

With the remaining x variables:

1. floor_area_sqm
2. max_floor_lvl
3. total_dwelling_unit
4. 2room_sold
5. 4room_sold
6. exec_sold
7. multigen_sold
8. studio_apartment_sold
9. 3room_rental
10. other_room_rental
11. hdb_cbd_dist
12. hdb_to_mrtdist
13. No_Bus_Stops
14. lease_remaining

In [9]:
```
X = residential[['floor_area_sqm','max_floor_lvl',
       'total_dwelling_units', '2room_sold',
       '4room_sold', 'exec_sold', 'multigen_sold',
       'studio_apartment_sold', '3room_rental',
       'other_room_rental', 'hdb_cbd_distance',
       'hdb_to_mrtdist', 'No_Bus_Stops', 'lease_remaining']]
y = residential[['real_price_persqm']]
X_constant = sm.add_constant(X)
```

```
lr = sm.OLS(y, X_constant.astype(float)).fit()
print(lr.summary())
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:      real_price_persqm   R-squared:                       0.525
Model:                            OLS   Adj. R-squared:                  0.525
Method:                 Least Squares   F-statistic:                 1.254e+04
Date:                Sun, 12 Mar 2023   Prob (F-statistic):               0.00
Time:                        18:06:33   Log-Likelihood:            -1.2735e+06
No. Observations:              158904   AIC:                         2.547e+06
Df Residuals:                  158889   BIC:                         2.547e+06
Df Model:                          14                                         
Covariance Type:            nonrobust                                         
==================================================================================
=======
                            coef    std err          t      P>|t|      [0.025      
0.975]
----------------------------------------------------------------------------------
-------
const                  4542.8832     20.037    226.720      0.000    4503.610    4
582.156
floor_area_sqm           -5.3176      0.116    -45.675      0.000      -5.546      
-5.089
max_floor_lvl            28.6945      0.493     58.217      0.000      27.728      
29.661
total_dwelling_units     -0.0036      0.061     -0.059      0.953      -0.124      
0.117
2room_sold               -3.1213      0.165    -18.952      0.000      -3.444      
-2.799
4room_sold               -0.1778      0.057     -3.095      0.002      -0.290      
-0.065
exec_sold                 3.1202      0.111     28.102      0.000       2.903      
3.338
multigen_sold            23.9142      1.406     17.008      0.000      21.158      
26.670
studio_apartment_sold     7.5218      0.308     24.433      0.000       6.918      
8.125
3room_rental             20.5291      3.086      6.653      0.000      14.481      
26.577
other_room_rental      -436.5917     27.324    -15.978      0.000    -490.147     -
383.036
hdb_cbd_distance         -0.1513      0.001   -299.861      0.000      -0.152      
-0.150
hdb_to_mrtdist           -0.1124      0.005    -21.347      0.000      -0.123      
-0.102
No_Bus_Stops              7.1592      0.569     12.591      0.000       6.045      
8.274
lease_remaining          34.9524      0.230    151.886      0.000      34.501      
35.403
==============================================================================
Omnibus:                    10880.860   Durbin-Watson:                   0.244
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            13914.200
Skew:                           0.638   Prob(JB):                         0.00
Kurtosis:                       3.689   Cond. No.                     2.64e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
[2] The condition number is large, 2.64e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Now we can remove 2room_sold as the P value is too high.

In [10]:
```
X = residential[['floor_area_sqm','max_floor_lvl',
       'total_dwelling_units',
       '4room_sold', 'exec_sold', 'multigen_sold',
       'studio_apartment_sold', '3room_rental',
       'other_room_rental', 'hdb_cbd_distance',
       'hdb_to_mrtdist', 'No_Bus_Stops', 'lease_remaining']]
y = residential[['real_price_persqm']]
X_constant = sm.add_constant(X)
lr = sm.OLS(y, X_constant.astype(float)).fit()
print(lr.summary())
```

```
                             OLS Regression Results                            
==============================================================================
Dep. Variable:      real_price_persqm   R-squared:                       0.524
Model:                            OLS   Adj. R-squared:                  0.524
Method:                 Least Squares   F-statistic:                 1.345e+04
Date:                Sun, 12 Mar 2023   Prob (F-statistic):               0.00
Time:                        18:06:33   Log-Likelihood:            -1.2737e+06
No. Observations:              158904   AIC:                         2.547e+06
Df Residuals:                  158890   BIC:                         2.547e+06
Df Model:                          13                                         
Covariance Type:            nonrobust                                         
==================================================================================
=======
                            coef    std err          t      P>|t|      [0.025      
0.975]
----------------------------------------------------------------------------------
-------
const                  4533.6966     20.054    226.073      0.000    4494.391    4
573.002
floor_area_sqm           -4.6108      0.110    -41.760      0.000      -4.827      
-4.394
max_floor_lvl            28.8085      0.493     58.387      0.000      27.841      
29.776
total_dwelling_units     -0.1971      0.061     -3.246      0.001      -0.316      
-0.078
4room_sold                0.1527      0.055      2.786      0.005       0.045      
0.260
exec_sold                 3.0002      0.111     27.035      0.000       2.783      
3.218
multigen_sold            23.1019      1.407     16.419      0.000      20.344      
25.860
studio_apartment_sold     7.6708      0.308     24.897      0.000       7.067      
8.275
3room_rental             24.5639      3.082      7.971      0.000      18.524      
30.604
other_room_rental      -425.3960     27.349    -15.554      0.000    -478.999    -
371.793
hdb_cbd_distance         -0.1510      0.001   -299.081      0.000      -0.152      
-0.150
hdb_to_mrtdist           -0.1126      0.005    -21.355      0.000      -0.123      
-0.102
No_Bus_Stops              7.1987      0.569     12.647      0.000       6.083      
8.314
lease_remaining          34.0740      0.226    150.997      0.000      33.632      
34.516
==============================================================================
Omnibus:                    11478.182   Durbin-Watson:                   0.243
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            14944.378
Skew:                           0.654   Prob(JB):                         0.00
Kurtosis:                       3.740   Cond. No.                     2.64e+05
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly s
pecified.
[2] The condition number is large, 2.64e+05. This might indicate that there are
strong multicollinearity or other numerical problems.
```

In [ ]: