Name: Rahul Vijayvargiya,
Student ID: 245784

Task:
Real estate valuation data set Link for dataset:

[UCI Machine Learning Repository: Real estate valuation data set Data Set]

The dataset contains historical data, from the Taiwanese property market.
The dataset allows learning a model that will estimate housing prices based on the data describing them. The collection is designed for the regression task.

-Performing Multivariate Regression

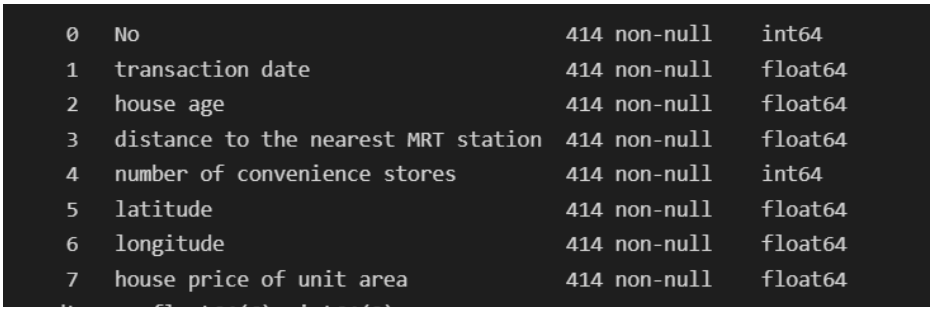
Sol:

Columns names:

['No',
'transaction date', 'house age',
'distance to the nearest MRT station',
'number of convenience stores',
'latitude', 'longitude',
'house price of unit area'] are the columns in our dataset,

Which tells us about index no.
Transaction date, house age,
Distance from the metro-station, no. of convenience stores around, coordinates and in that area
What are the price as per unit area

1.

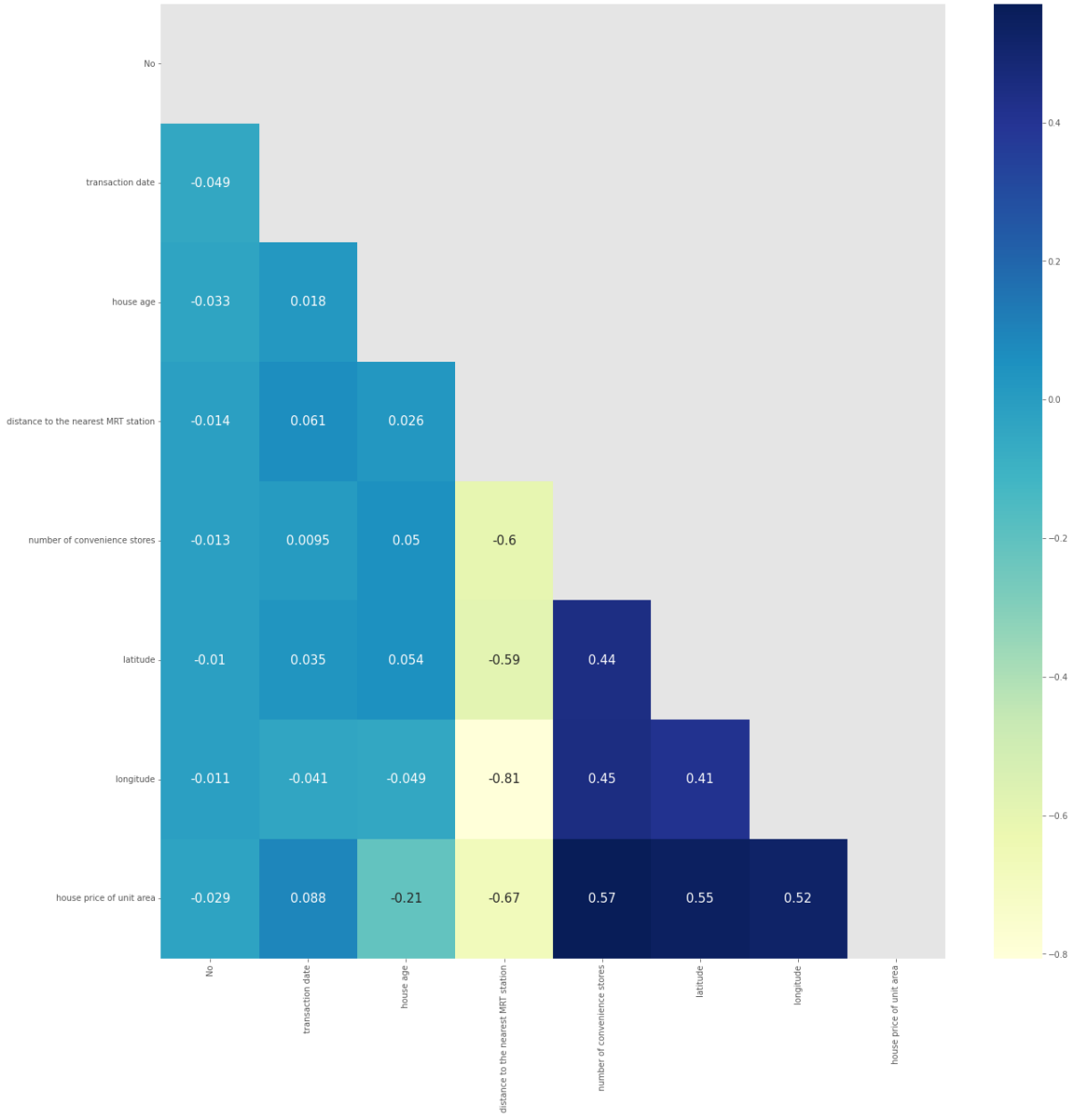


0	No	414 non-null	int64
1	transaction date	414 non-null	float64
2	house age	414 non-null	float64
3	distance to the nearest MRT station	414 non-null	float64
4	number of convenience stores	414 non-null	int64
5	latitude	414 non-null	float64
6	longitude	414 non-null	float64
7	house price of unit area	414 non-null	float64

start with a checking a correlation with columns of dataset,

Correlation is a statistical measure that expresses the extent to which two variables are linearly related or not

A correlation coefficient of +1 indicates a perfect positive correlation. As variable x increases, variable y increases. As variable x decreases, variable y decreases. A correlation coefficient of -1 indicates a perfect negative correlation.



2. Converting our dependent and independent variables into numpy array

3. Splitting Dataset into training and test purposes

Finally here, we are converting our Array into train and test dataset, with 75% Data into Training and 25% data for the test

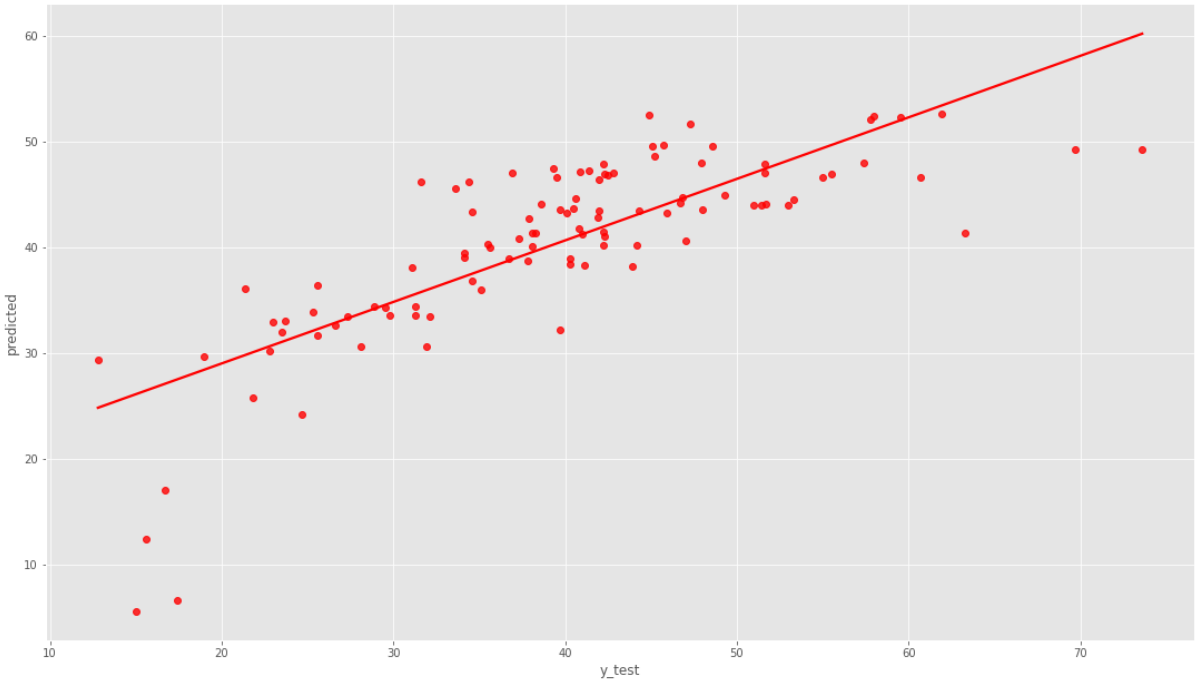
Linear Regression:

a linear approach for modelling the relationship between a scalar response and one or more explanatory variables (also known as dependent and independent variables).

We are using sklearn from python, which has regression model inside,
We are calling and fitting our training data into it

4. We are fitting our data into Model, Linear Regression from Sk-Learn

5.



As you can see a regression line has been plotted using Linear Regression

6. We have model accuracy on test data is 65.49%

Furthermore checking, our actual values, predicted values and the difference b/w

Actual Value	Predicted Value	Difference	
0	**48.0**	**43.568546**	**4.431454**
1	**31.3**	**34.419184**	**-3.119184**
2	**59.5**	**52.324788**	**7.175212**
3	**34.1**	**39.444524**	**-5.344524**
4	**48.6**	**49.525871**	**-0.925871**
99	**17.4**	**6.600372**	**10.799628**
100	**40.8**	**41.817961**	**-1.017961**
101	**37.9**	**42.741396**	**-4.841396**
102	**60.7**	**46.615230**	**14.084770**
103	**40.3**	**38.932491**	**1.367509**

7.

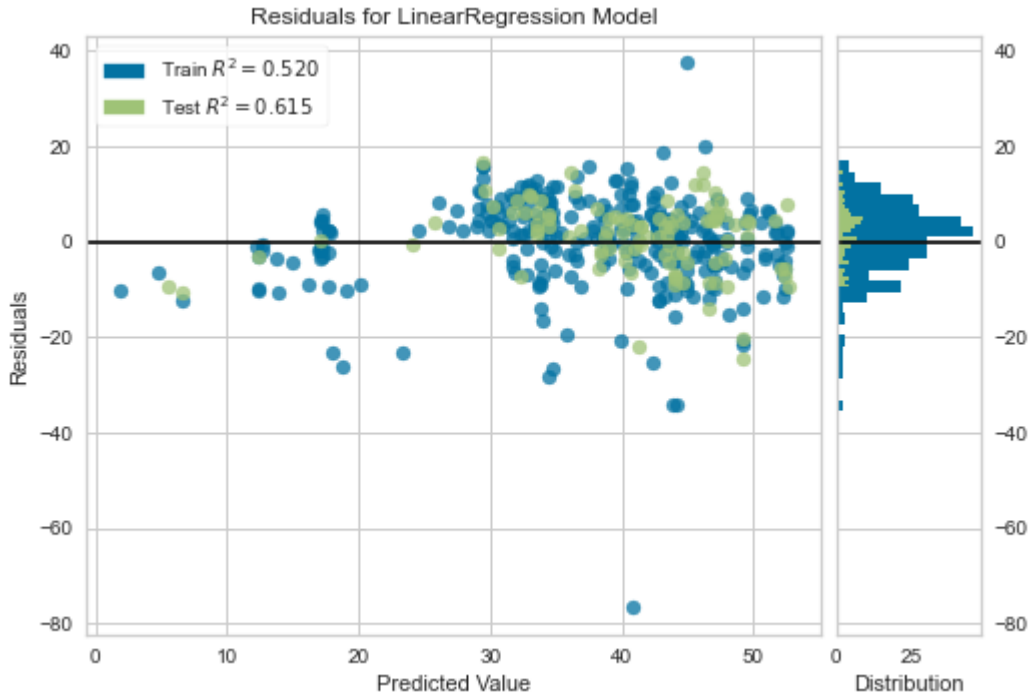
Metrics	
MAE	**5.830975**
MSE	**54.255359**
RMSE	**7.365824**

As you can see above, mean absolute error, mean square error, and root mean square error

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit.

The Mean squared error (MSE) tells you how close a regression line is to a set of points. It does this by taking the distances from the points to the regression line (these distances are the “errors”) and squaring them. The squaring is necessary to remove any negative signs.

The Mean of the absolute error(MAE) tells us about the absolute value of the difference between the forecasted value and the actual value. MAE tells us how big of an error we can expect from the forecast on average.



Cross Validation:

R-Squared (R^2 or the coefficient of determination) is a statistical measure in a regression model that determines the proportion of variance in the dependent variable that can be explained by the independent variable. In other words, r-squared shows how well the data fit the regression model (the goodness of fit).

From Sk-Learn we are using methods like KFOLD, Cross_Val_Score and GridSearchCV

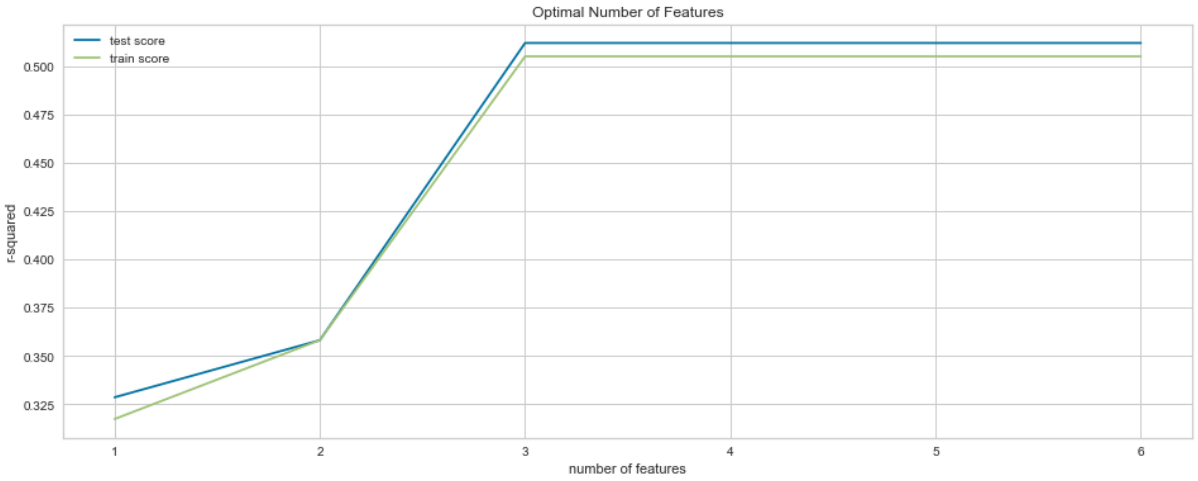
We observe we have different r square for all methods

From Cross_val_score we get a mean r-square of 48%

From KFOLD we get r-square of 51.22%

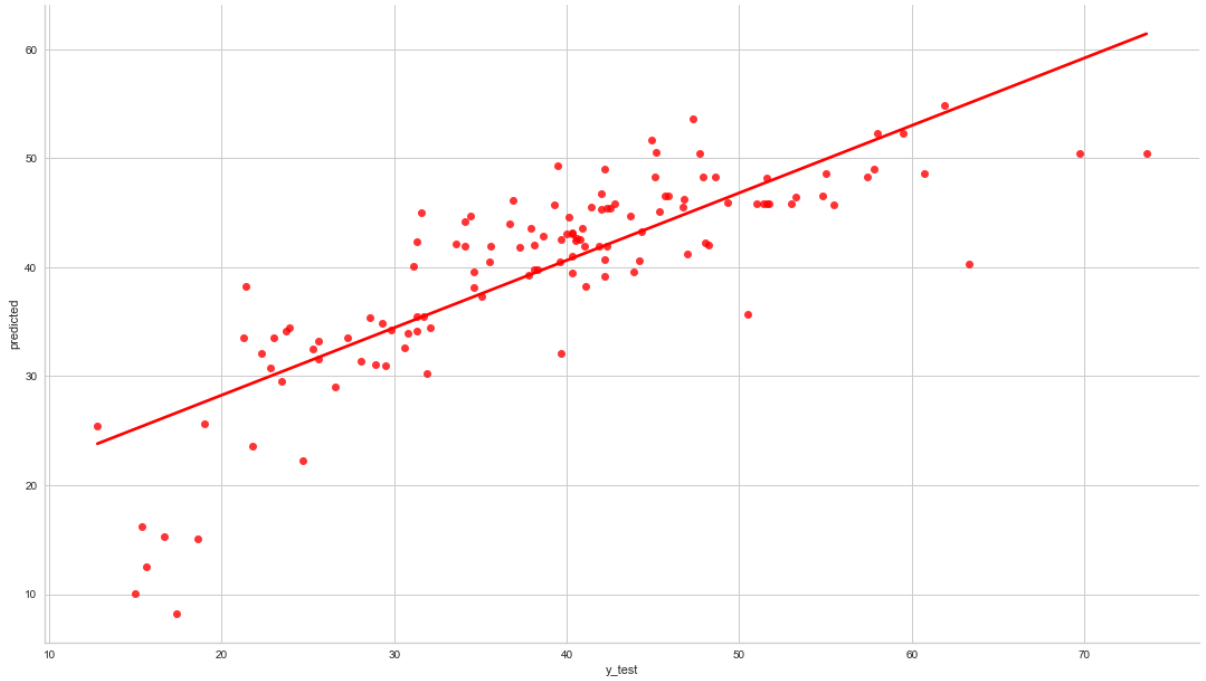
From GridSearchCV we observe the best hyper parameter are which gives the most accuracy is

```
GridSearchCV(cv=KFold(n_splits=5, random_state=100, shuffle=True),
             estimator=RFE(estimator=LinearRegression()),
             param_grid=[{'n_features_to_select': 3}],
             return_train_score=True, scoring='r2', verbose=1)>
```



The Experiments:

Using different test data and adding more features and removing feature, selecting data randomly we observed the our model gave us different accuracy and in some scenarios it's giving good and rest it worse, huge difference occur in the r-square on different parameter, since in grid search method we use RFE, it tell us 3 feature are the optimal no. of dependent variable to use in the model


predicted
y_test
10
20
30
40
50
60
70