

**HOUSING: PRICE PREDICTION**

**DOCUMENTATION**

Submitted by:

Pooja Mishra

**ACKNOWLEDGMENT**

I would also like to thank Flip Robo Technologies and Datatrained team who has given me such wonderful opportunity to learn more about Machine Learning and their mentors Shubham Yadav, Sajid Chaudhary for their constant encouragement, valuable suggestions, and cooperation.

This acknowledgement will remain incomplete if I fail to express my deep sense of obligation to my family members and God for their consistent blessings and believe in me.

Thank You.

**INTRODUCTION**

* Business Problem Framing

In this century, having house is one of the necessities for every person. House prices are increasing every year, so there is a need of system to predict the house prices. In this project, we are going to analyse the features and create the model to predict house prices.

* Conceptual Background of the Domain Problem

In this project, the understanding of features is important to analyse the house prices. Since housing price is strongly correlated to other factors such as location, area, population, building year, facilitates, it requires many other information to predict individual housing price house related information to analyse the house pricing.

* Review of Literature

In this dataset, various features have been taken to describe the relationship between houses within a given neighbourhood, and the macro-level equation which specifies the relationships between neighbourhoods. The hedonic price model is a model that estimates house prices using some attributes such as the number of bedrooms in the house, the size of the house, etc.

The data used in this study take the house prices from a US-based housing company named Surprise Housing has decided to enter the Australian market. Each record in the dataset contains data about a house in the area: it contains the address, the unit postcode, property type, the duration (freehold or leasehold), the sale price, the date of the sale, and whether the house was newly-built when it was sold. In total, the dataset contains around 1460 entries each having 81 variables. To enable model training and testing, the dataset was divided into a training set that contains data about house sales, and a test set that contains features data except target variable. For the same purpose, the company has collected a data set from the sale of houses in Australia.

* Motivation for the Problem Undertaken

The main objective of this project is to analyse the train and test dataset to predict the house pricing. This model will then be used by the management to understand how exactly the prices vary with the variables. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns.

In this project we will build a model using Machine Learning to predict the actual value of the prospective properties and decide whether people need to invest or not.

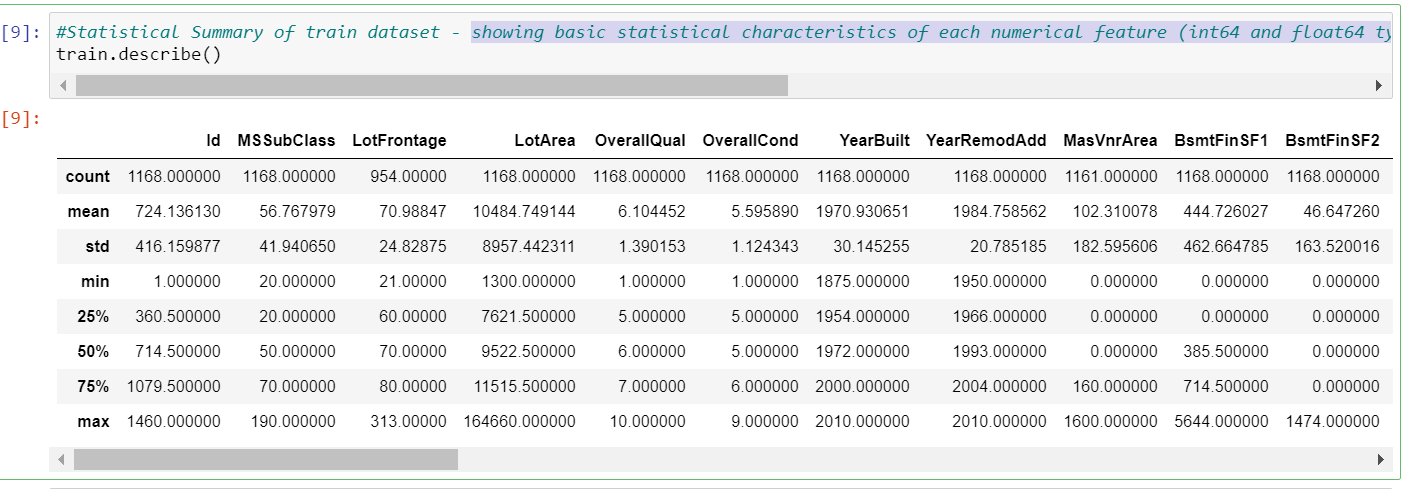
This project demonstrates the house pricing by taking all possible features thus this study is very much motivational to predict the house pricing.

**Analytical Problem Framing**

* Mathematical/ Analytical Modelling of the Problem

This project consists of train and train datasets each having 80 features and 81 features respectively. The mathematical approach behind this project is to get the strategical overview of the dataset. In this dataset by performing strategical summary we can conclude that train & test dataset shows the basic statistical characteristics of each numerical feature (int64 and float64 type).

We can observe the count, mean, Standard deviation, min and max range, mean, median, first and third quantile values of each features present in the dataset which will help to perform required data engineering functionalities on the dataset.



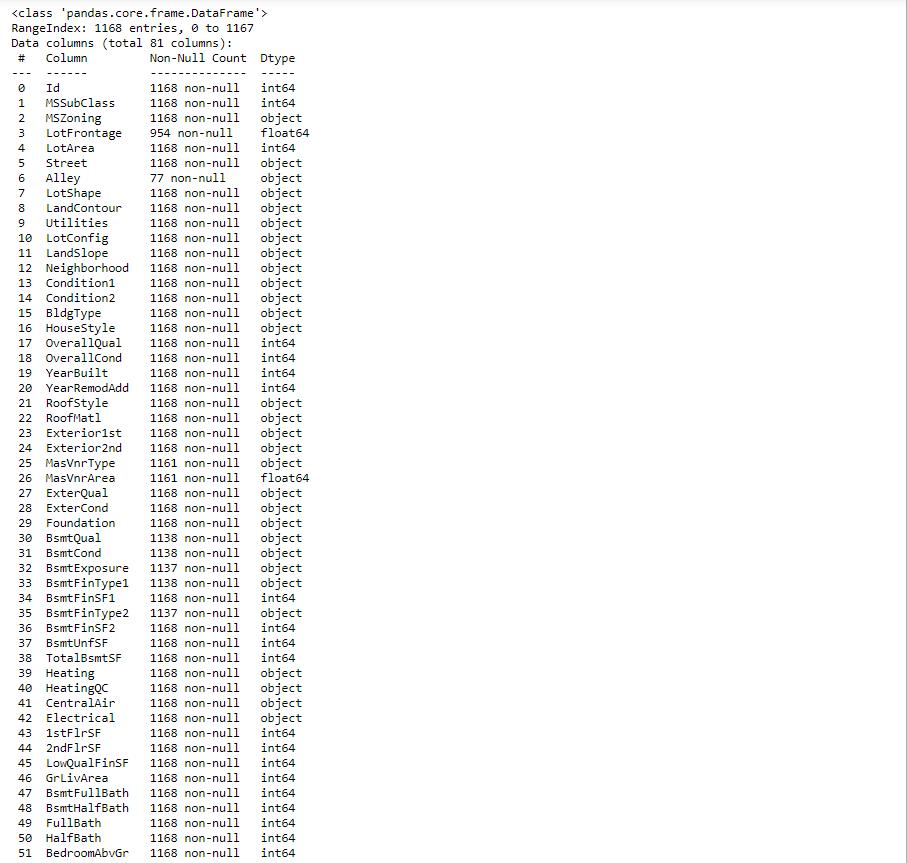
* Data Sources and their formats

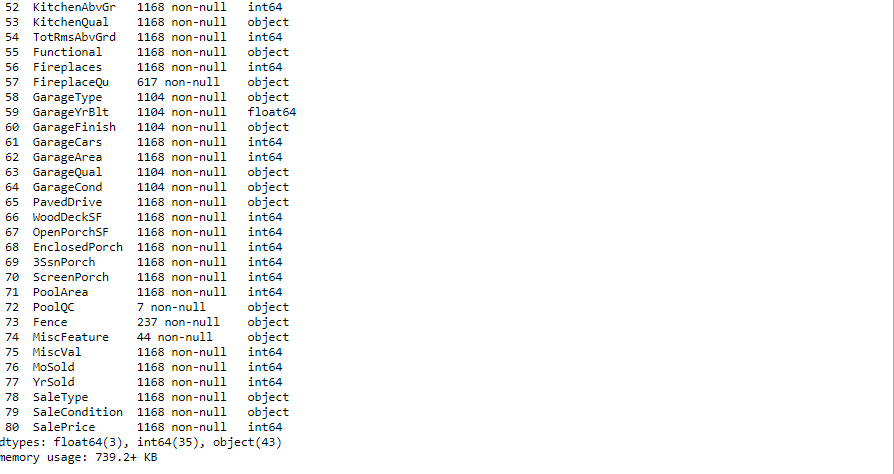
This dataset describes the sales of residential units in Australia. The dataset contains many variables that are involved in determining a house price. The dataset contains 1460 records (rows) and 81 features (columns).

Here, we will provide a brief description of dataset features. Since there are 81 features thus describing each could be tough so mentioning each datatype here to understand the dataset.

Train Dataset Description: This training dataset consists of 1,168 rows with 81 features describing every aspect of the house. We are given sale prices (SalePrice) for each house. The training data is what we will use to “teach” or train our models. This dataset having null values which we will handle later. It consists of 3 float type, 35 integer type and 43 object type.

Test Dataset Description: This Test dataset consists of 292 rows with 80 features describing every aspect of the house based on test data history. Target feature is not available here as train dataset. The Test data is what we will use to “teach” or train our models. This dataset having null values which we will handle later on. It consists of 4 float type, 34 integer type and 42 object type.

­



* Data Pre-processing

Data pre-processing is very important to get the dataset into the best format before performing algorithm. This is very important step in Machine Learning that should not be skipped. It involves three stages: Data Cleaning, Data Transformation, and Feature Engineering which converts complicated dataset into quality data.

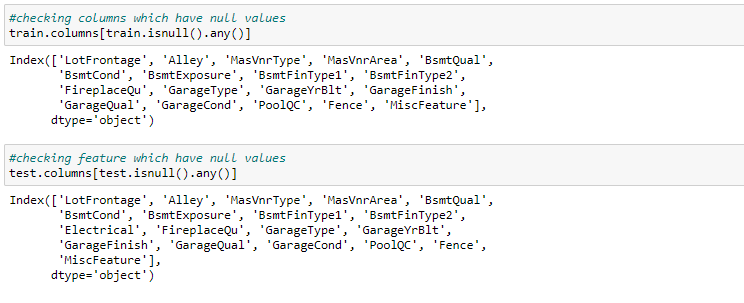
**Data Cleaning:**

In Data cleaning, we understand the data, we perform several activities to clean the train & test data. This stage comprises of following activities:

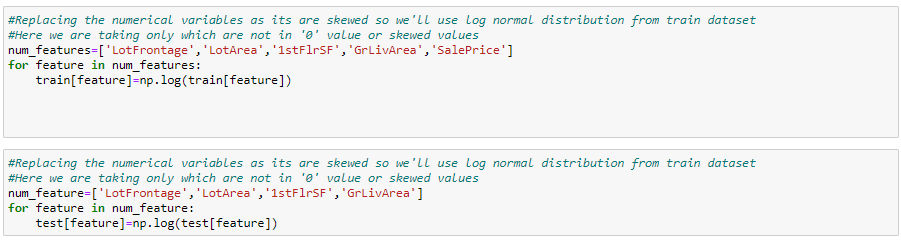
Dealing with Null Values – In this project, train dataset has total 18 columns which consist of null values as shown in below figure.

In test dataset there are 19 columns having missing values.

All missing values in both datasets will be filled with mean and mode as per the category of values as presented in above workflow step.



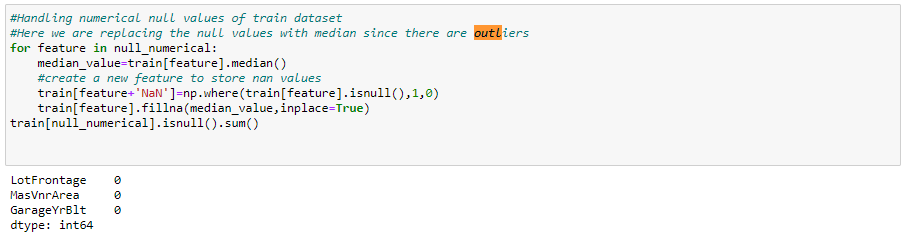
Dealing with Skewness – In these datasets, few of the columns do not follow the normal distribution that need to treat to convert it into normal distribution. In this step, we checked the skewness and used the NumPy log method to make the skew values into normally distributed.



In above image, you can see the columns which have skewed values, i.e. not normally distributed thus making the values near to 0 using NumPy log.

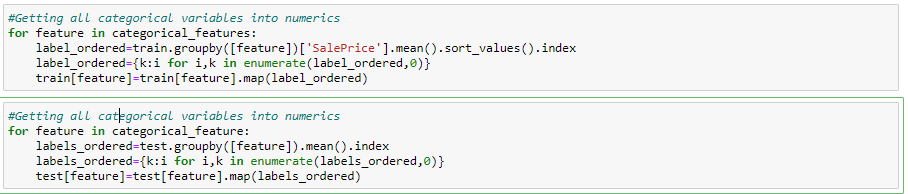
**Outliers:**

An outlier is a data point in a dataset that is distant from all other observations i.e. a data point that lies outside the overall distribution of the data set. Depending on the dataset, we’ll be performing activities to remove it.



**Data Transformation:**

In Data transformation we recheck that all present columns are numerical or not? if not then we perform encoding type depending on the type of values present in columns.



Most important thing we need to keep in mind that all data pre-processing steps will be performed according the dataset present.

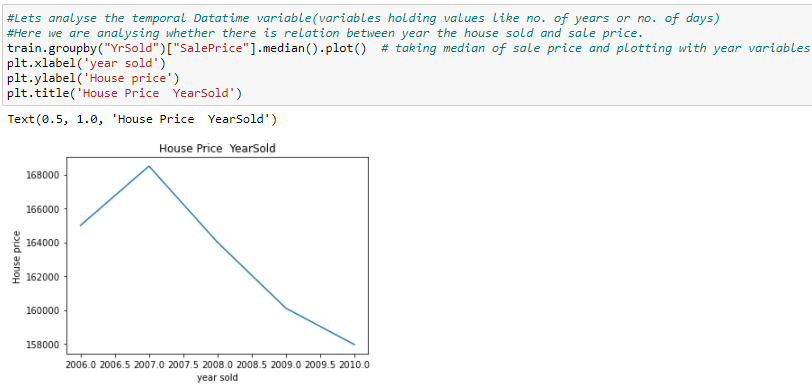
**Correlation Matrix:**

Correlation Matrix is a covariance matrix. A summary measure called the correlation describes the strength of the linear association.

In this dataset, we have described the correlation and relationship present amongst each feature with target variable. Here we observed that there are many correlated variables in our dataset. We notice that Garage Cars and Garage Area have high positive correlation which is reasonable because when the garage area increases, its car capacity increases too. We see also that Gr Liv Area and TotRms AbvGrd are highly positively correlated which also makes sense because when living area above ground increases, it is expected for the rooms above ground to increase too. For negative correlation, we can see that Bsmt Unf SF is negatively correlated with BsmtFin SF 1, and that makes sense because when we have more unfinished area, this means that we have less finished area. We note also that Bsmt Unf SF is negatively correlated with Bsmt Full Bath which is reasonable too.

* Data Inputs- Logic- Output Relationships

As we can see that there are large number of columns thus analyse with the target variable is very important. By performing data visualization, we can analyse columns relationship with the target variable. Few observations with train dataset are mentioned below:



With above graph we observed that the house price is decreasing with increase in year of house sold.

In this project as there are many columns thus, we’ll categorise the features into numerical, discreate, continuous categorical to analyse the input and output variables relationships.

Discreate Features:  There is relationship between discrete variables and sale price like Overall quality showing exponential relationship to saleprice. Monotonic relationship, uniform and non-uniform relationship relationships with sale price. Thus, this visualization giving lots of information.

Temporal features: From these visualization graphs we observed that how Sale price is dependent on year variables. The sale price of houses is higher with recently build house than old constructed. Other information relation to Sale Price gives more information.

Categorical Features: The input and output features are uniformly and non-uniformly distributed.

Like this we can analyse the input features relationships with target feature which will help a lot in feature Engineering.

* State the set of assumptions (if any) related to the problem under consideration

In this Project to get the right prediction about house pricing we need to do lots of assumptions. The assumptions that I have taken are mostly while handling the missing values as we need to assume that missing values in particular column indicates that it can be filled with ‘0’ or mean or median values depending on the values.

* Hardware and Software Requirements and Tools Used

Machine learning comes with an extensive collection of ML tools, platforms, and software.

Well, in this problem solution we have used following tools that helps to make this project successful as per my possibility.

Jupyter notebook is one of the most widely used machine learning tools among all. It is a very fast processing as well as an efficient platform. Moreover, for this problem solution I have used python programming.

Scikit-Learn is built on top of the three main Python libraries viz. NumPy, Matplotlib, and SciPy. Along with this, it will also help you with testing as well as training your models.

The Libraries are as listed:

1. For Data loading and Visualisation:

Numpy - NumPy is very useful for handling linear algebra, Fourier transforms, and random numbers.

Pandas - Pandas are turning up to be the most popular Python library that is used for data analysis with support for fast, flexible, and expressive data structures designed to work on both “relational” or “labeled” data.

Matplotlib - The library helps to generate histograms, plots, error charts, scatter plots, bar charts with just a few lines of code.

Seaborn – Used for visualization.

For Normalization:

Min-max scaler is the standard approach for scaling. We use this library to balance the dataset and make it normal for further algorithm performance.

train\_test\_split: We use it to perform test train spliting.

Algorithm Libraries:

In this problem solution we have used algorithm libraries:

RandomForestRegressor – for regression algorithm

*from sklearn.ensemble import RandomForestRegressor*

Linear Regression– For Linear Regressor algorithm

*from sklearn import linear\_model*

*model = linear\_model.LinearRegression()*

GradientBoostRegressor – To perform gradient Boost Regressor algorithm

*lr = ensemble.GradientBoostingRegressor()*

Root Mean Square Error-

*from sklearn.metrics import mean\_squared\_error*

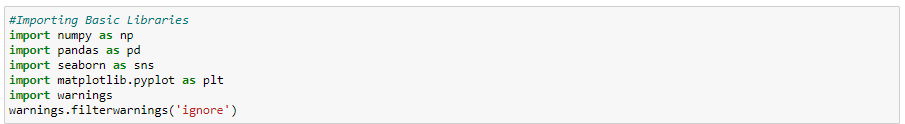
RandomizedSearchCV – For hyper parameter tuning performance.

*from sklearn.model\_selection import RandomizedSearchCV*

Lasso Regression for feature selection

*from sklearn.linear\_model import Lasso*

*from sklearn.feature\_selection import SelectFromModel*



**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

In this project, we will build our prediction model: we will choose algorithms for each of the techniques we mentioned in the previous section. After we build the model, we will evaluate its performance and results. Here, we want to predict the *price* of a house given information about it. The price we want to predict is a continuous value; it can be any real number. This can be seen by looking at the target variable in our dataset SalePrice: That means that the prediction type that is appropriate to our problem is regression. Now we move to choose the modelling techniques we want to use.

* Testing of Identified Approaches (Algorithms)

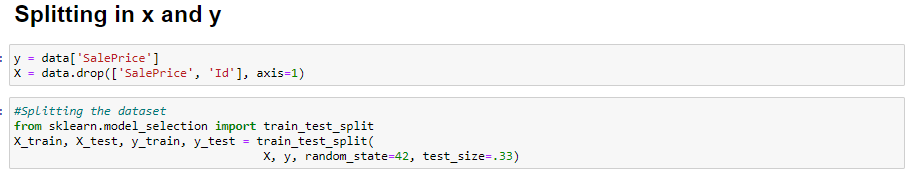
There are a lot of techniques available for regression problems like Linear Regression, Ridge Regression, Decision Trees, Random Forest, etc. In this project, we will test many modelling techniques, and then choose the technique(s) that yield the best results. The techniques that we will try are: Linear Regression, Gradient Boosting Regressor and Random Forest Regressor.

* Run and Evaluate selected models

After getting the simplified dataset, we’ll perform test train split to use it on algorithms.

## Splitting the Dataset

As usual for supervised machine learning problems, we need a training dataset to train our model and a test dataset to evaluate the model. So we will split our dataset randomly into two parts, one for training and the other for testing. For that, we will use another function from Scikit-Learn called train\_test\_split():

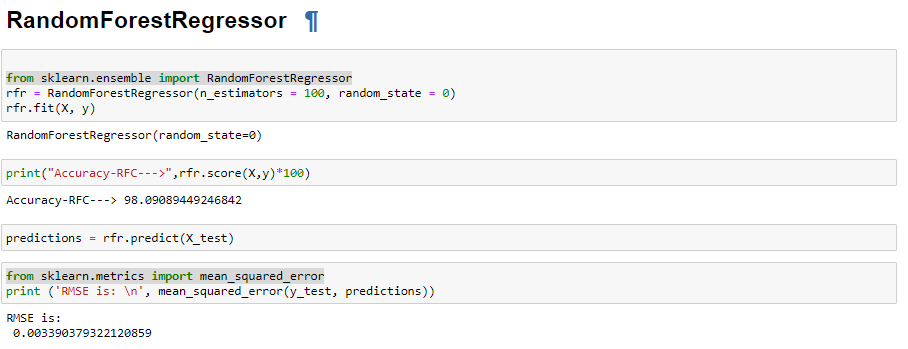


1. Linear Regression[¶](https://www.kaggle.com/ammar111/house-price-prediction-an-end-to-end-ml-project#1.-Linear-Regression)

This technique models the relationship between the target variable and the independent variables (predictors).

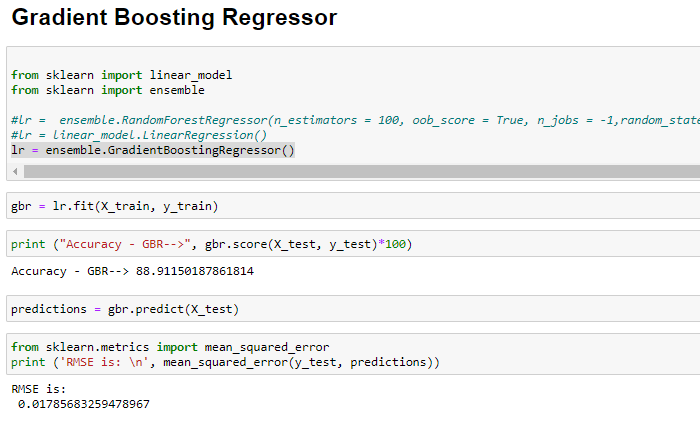


1. Random Forest Regressor: Bagging is an ensemble method where many base models are used with a randomized subset of data to reduce the variance of a the base model.



### Gradient Boosting

Boosting is also an ensemble method where weak base models are used to create a strong model that reduces bias and variance of the base model.



* Key Metrics for success in solving problem under consideration

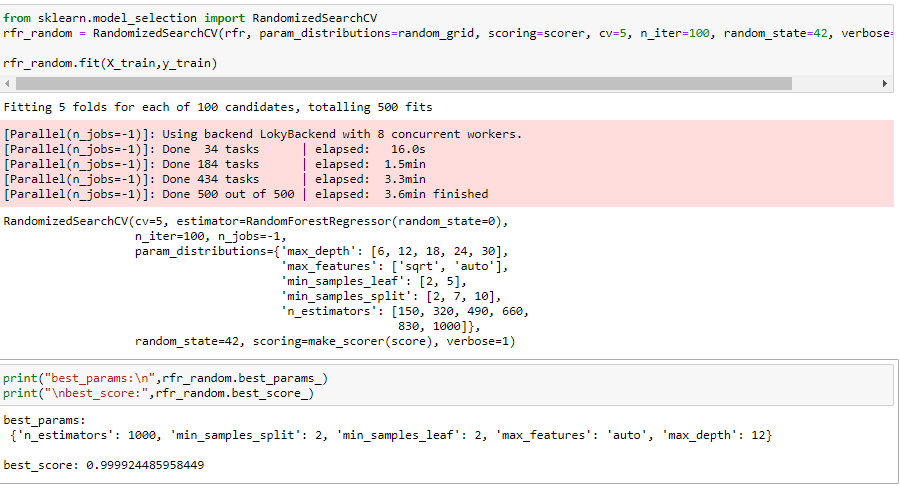
For each one of the techniques mentioned in the previous section (Linear Regression, Random Forest Regressor, Gradient Boost Regressor, etc.), we will follow these steps to build a model:

* Choose an algorithm that implements the corresponding technique
* Search for an effective parameter combination for the chosen algorithm
* Create a model using the found parameters
* Train (fit) the model on the training dataset
* Test the model on the test dataset and get the results

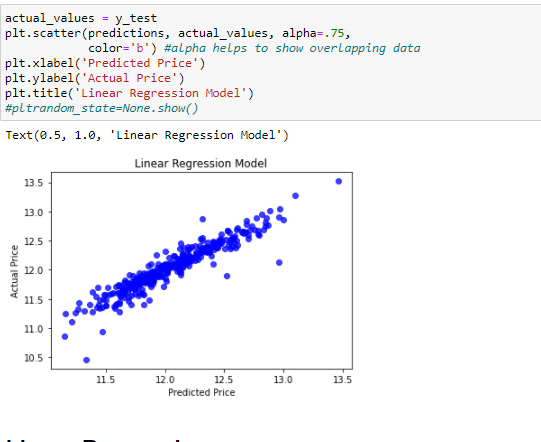
.

* Visualizations

In this project, we observed best model fir by Random Forest Regression as it showing highest value for model fitting thus we will use Randomized Search CV() to search for the best model parameters in a parameter space provided by us. The parameter n\_estimators specifies the number of trees in the forest, bootstrap determines whether bootstrap samples are used are 'n\_estimators': 1000, 'min\_samples\_split': 2, 'min\_samples\_leaf': 2, 'max\_features': 'auto', 'max\_depth': 12}



Visualization of actual and Predicted Model:



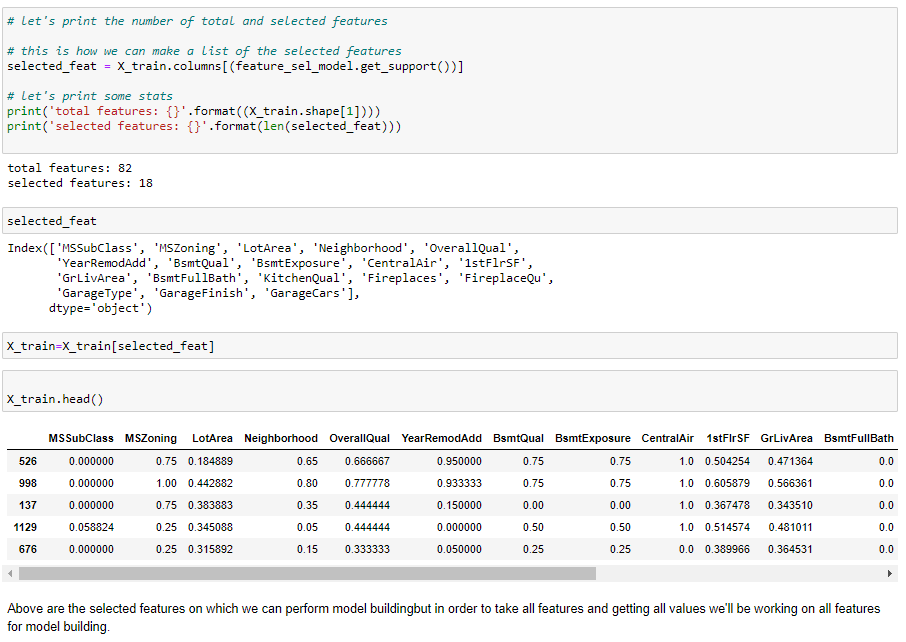
* Interpretation of the Results

### Common Important Features

Now, let us see which features are among the most important features for models, and

Here I specify the Lasso Regression model, and I select a suitable alpha (equivalent of penalty). The bigger the alpha the less features that will be selected. Then I use the selectFromModel object from sklearn, which will select the features which coefficients are non-zero.

The Lasso Regression helps to get the important features which are useful for the modelling. Here are listed features which are important:



1. In this paper, we built several regression models to predict the price of some house given some of the house features. We evaluated and compared each model to determine the one with highest performance i.e. random Forest Regression. We also looked at how some models rank the features according to their importance. In this document, we followed the data science process starting with getting the data, then cleaning and pre-processing the data, followed by exploring the data and building models, then evaluating the results and communicating them with visualizations.
2. We also suggest that people take into consideration the features that were deemed as most important as seen in the previous section; this might help them estimate the house price better.

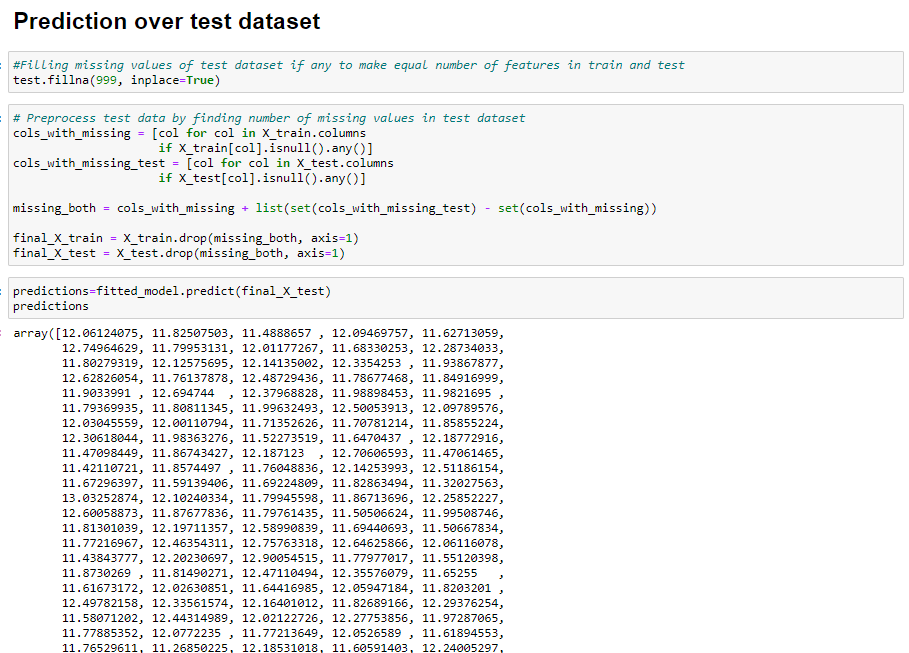
**CONCLUSION**

* Key Findings and Conclusions of the Study

After performing 3 algorithms I can conclude that Random Forest is giving the best results in comparison to other algorithms. Considering all evaluation and performance metrics, Random Forest is getting the highest accuracy more than Gradient Boost Regressor as well.

We came to confirmation that Random Forest is best fit model by performing hyper parameter tuning on it which gives same accuracy more than 95%. We also found the important features that are important in this project.

After training the dataset we performed the prediction over test dataset as shown in below figure:



we created many models: for each model, we searched for good parameters then we constructed the model using those parameters, then trained (fitted) the model to our training data (X\_train and y\_train), then tested the model on our test data (X\_test) and finally, we evaluated the model performance by comparing the model predictions with the true values in y\_test.

* Learning Outcomes of the Study in respect of Data Science

After working on this project, we came to conclude following observations:

To apply data preprocessing and preparation techniques in order to obtain clean data.

To build machine learning models able to predict house price based on house features.

To analyze and compare models performance in order to choose the best model.

As This dataset consist of many features thus was very helpful to gather more information by performing feature engineering.

* By Limitations of this work and Scope for Future Work

With help of this project data science has given solution to one of the necessary problems where thousands of houses are sold every day. There are some questions every buyer asks himself like: What is the actual price that this house deserves? Am I paying a fair price? In this paper, a machine learning model is proposed to predict a house price based on data related to the house (its size, the year it was built in, etc.) Data science able to answer all these queries.

By better understanding and more learning we can perform model building in better way and can give better solutions to world.

Thank You.