**Internship Report**

Name: Abhigyan Sinha

VT Number: VT20222029

Degree: B. Tech, CSE

College: Kalinga Institute of Industrial Technology (KIIT)

Mentor: Mr. Uma Sankar Sahoo

Company: Tata Steel Limited

**House Price Prediction**

****

**Abstract:** The real estate market is exposed to many ﬂuctuations in prices because of existing correlations with many variables, some of which cannot be controlled or might even be unknown. Housing prices can increase rapidly, yet the numerous listings available online where houses are sold or rented are not likely to be updated that often. In some cases, individuals interested in selling a house might include it in some online listing, and forget about updating the price. In other cases, some individuals might be interested in deliberately setting a price below the market price in order to sell the home faster, for various reasons.

In this report, we aim at developing a machine learning application that identiﬁes price of houses in the real estate market. This program can be useful for investors interested in the housing market. We have focused on a case considering housing assets located in Bangalore, India. The application is formally implemented as a regression problem that tries to estimate the market price of a house by using the given features retrieved from Kaggle.

Keywords – Machine learning, Regression Technique, Classification Technique, Cross validation Technique, K-means

**1. Introduction**

House/Home are a basic necessity for a person and their prices vary from location to location based on the facilities available like No. of rooms, Parking Space, Type etc. The house pricing is a point that worries a ton of residents whether it is the rich or white-collar class as one can never judge or gauge the valuing of a house based on area or offices accessible. Buying of a house is one of the significant decisions of a family as it expends the entirety of their investment funds. It is the difficult task to predict the accurate values of house pricing. Our proposed model would make it possible to predict the exact prices of houses.

**1.1 Objective**

This project is proposed to predict house prices and to get better and accurate results. The stacking algorithm is applied on various regression algorithms to see which algorithm has the most accurate and precise results. This would be of great help to the people because the house pricing ids a topic that concerns a lot of citizens whether rich or middle class as one can never judge or estimate the pricing of a house on the basis of locality or facilities available.

**1.2 Proposed Work and Technology**

**a) Python**

Python is an elevated level programming language for broadly useful programming. Python bolsters various programming standards including object arranged, useful and procedural. Python is an easily readable language. Python supports various libraries such as Pandas, NumPy, SciKit, Matplotlib etc. It tends to be utilized to make web applications. It very well may be utilized to peruse and alter documents. It very well may be used to perform complex science. It is an exceptionally helpful language for web improvement and programming advancement.

**Compiler/Editor used: Jupyter Notebook embedded in VSCode**

**Language used: Python 3.10.4**

**Packages used: Pandas 1.4.2, NumPy 1.22.3**

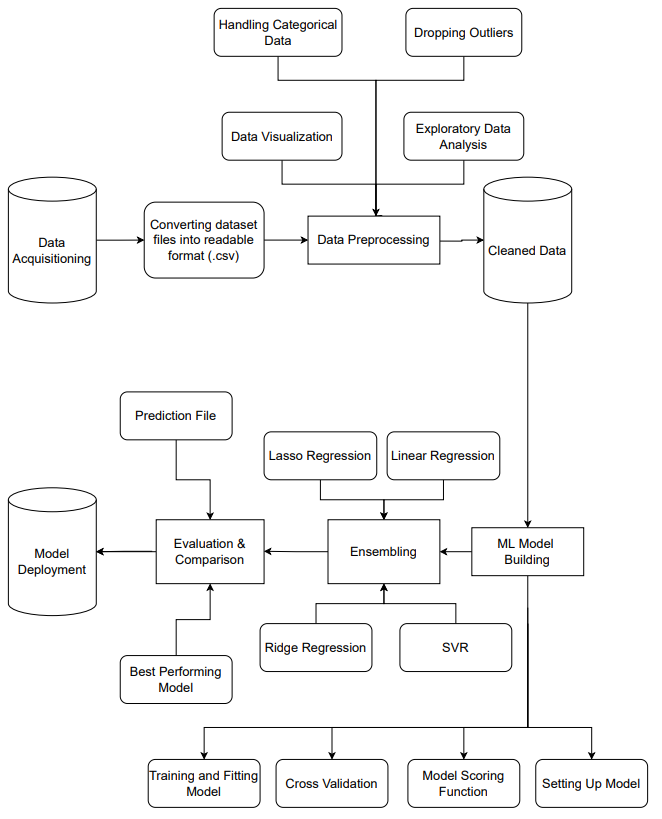
**b) Machine Learning**

Machine Learning is a field of Artificial Intelligence which enables PC frameworks to learn and improve in execution with the assistance of information. It is used to study the construction of algorithms that make predictions on data. Machine learning is used to perform a lot of computing tasks. It is also used to make predictions with the use of computers. The process of machine learning involves providing data and then training the computers by building machine learning models with the help of various algorithms. Machine learning can be used to make various applications such as face detection application, etc.

**2. Literature Review**

There are a couple of components that impact house costs. In this exploration, there are 8 components, there are location, total square feet, bath, area-type, availability, society, balcony and BHK. Location is a critical factor in shaping the expense of a house. The location chooses the normal land cost. It also opens the basic passage to open workplaces, for instance, schools, grounds, crisis facilities and prosperity centers, similarly as family preoccupation workplaces, for instance, strip malls, culinary visits, or much offer awesome landscape. Availability is also a factor that tells you whether the house still in building phase or ready to move in. BHK define the total number of bedrooms in the house. Total Square Feet is a major factor in price prediction. It defines the area of the land which has been used for building the house. Bath and Balcony shows the number of bathrooms and balcony respectively and Area Type gives you the information of the type of area of the house i.e., super built-up, Plot, Built-up etc.

**3. Methodology**

****

**3.1 Data Collection**

Data collection is the process of systematically collecting information about variables. This helps you find hypothesis which in turn allows you to answer the relevant query at that point and evaluate the results. I've tried different Kaggle datasets that fit my project goals. After looking through many records, I found this particular record of Bangalore City Home Price Dataset. This dataset is a very popular machine learning dataset with few errors and variations.

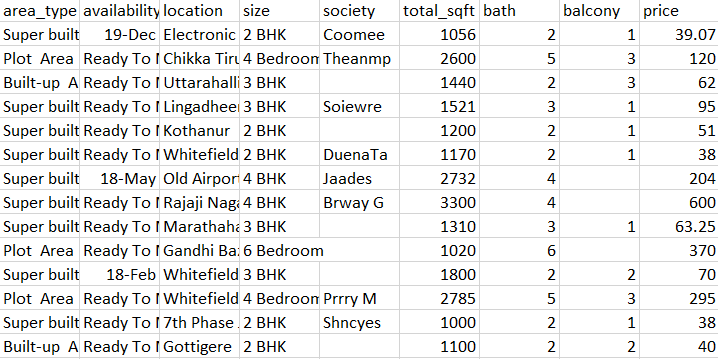
**Link of the dataset:** <https://www.kaggle.com/datasets/amitabhajoy/bengaluru-house-price-data>

**3.2 Data Pre-Processing**

It is the process of transforming data before feeding it into the algorithm. It is utilized to change over crude information into a clean dataset. It is an information mining strategy that includes moving crude information into a justifiable organization. The result of data preprocessing is the last dataset utilized for preparing and testing reason.

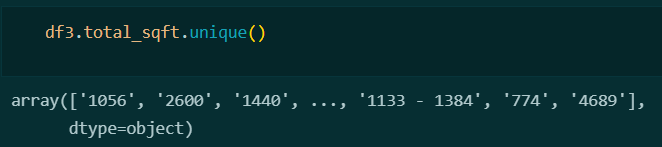
Data preprocessing is required errands for cleaning the information and making it appropriate for a Machine Learning model which likewise expands the precision and proficiency of a Machine Learning model.

For this project, we’ve used the dataset of house prices of Bengaluru. The dataset is shown below:



As per the domain knowledge, features like society, balcony, availability and area\_type do not have much importance for the price prediction. So, we remove those features from the dataset.

In the size column, I see that there are values like 4BHK as well as 4 Bedroom which are essentially the same. So to overcome this situation, I made a new column for size named “BHK” and took out the numbers from the size column and put it into the new “BHK” column to make my job easier.

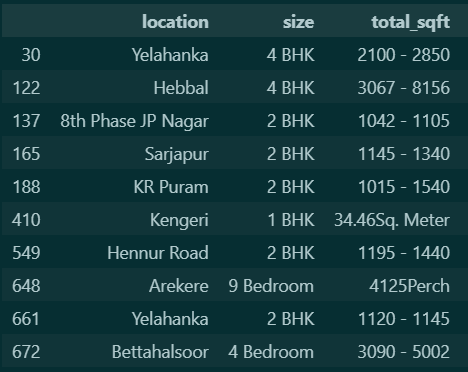


In total\_sqft, there are some instances that show the total sqft in range

for ex: 1133-1384

So, we have taken the average of the two numbers in the range i.e. (1133+1384)/2 and substituted it in the place of the corresponding range.

Another thing we see is that in size, there are instances where the size has units, we need to remove that as well for doing the job.

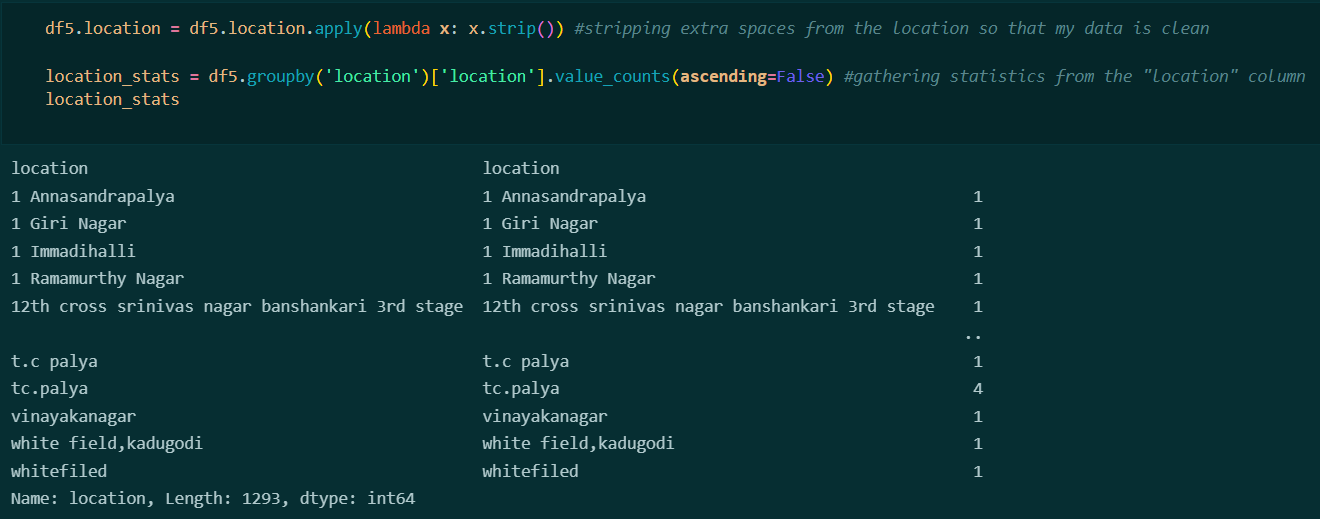


One of the most important observation was that when predicting a house price, we need to have price per sqft so we created a new column called “price\_per\_sqft”

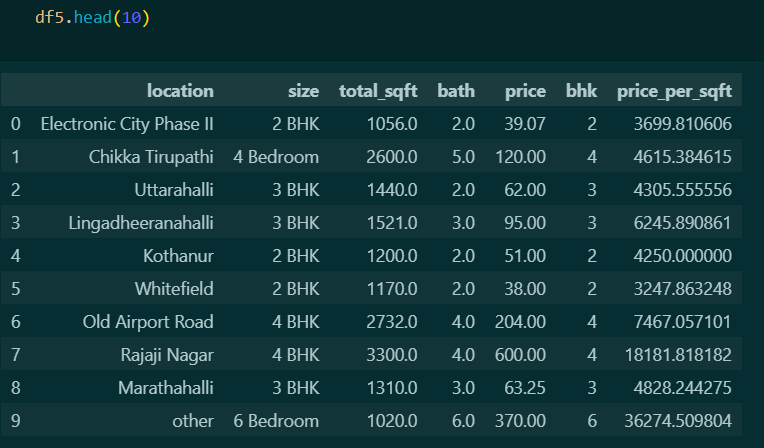


One Hot Encoding is a process by which categorical variables are converted into a form that could provide to ML algorithms to do a better job in prediction.

In our dataset, we have a categorical column called “locations”. In that column, we can apply the technique of one hot encoding but since the column has around 1300 locations, it becomes very difficult during One Hot Encoding as we will have 1300 features then. In order to overcome this situation, I have come up with another column named “other”. This column will have only those locations that have only one or two data points.

Given below are such locations:

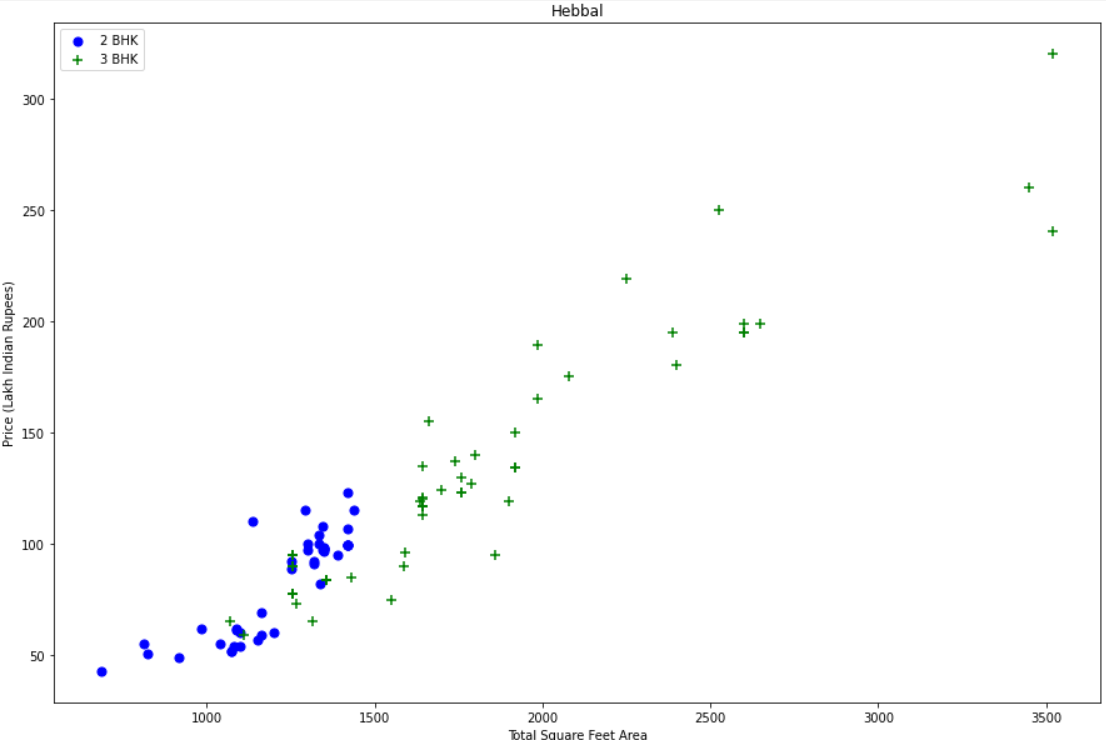
For our convenience, we take the locations that have 10 or less than 10 points in the “other” column. As a result, we are now left with 242 locations from 1300 locations having more than 10 data points.

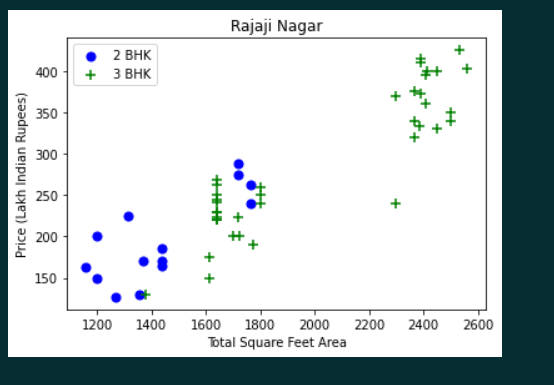


**3.2 Exploratory Data Analysis**

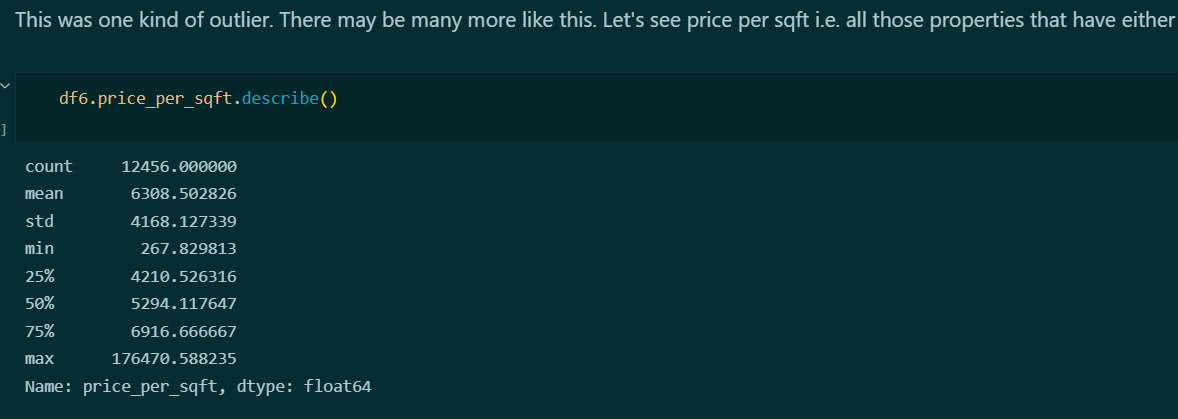
Exploratory Data Analysis or Data Visualization is the pictorial or graphical representation of information. It enables to grasp difficult concepts or identify new patterns and includes the creation and investigation of the visual portrayal of information. Effective visualization assists customers with separating and reason about data and verification. It makes complex data progressively accessible, reasonable and usable. Customers may have explicit logical endeavors, for instance, making assessments or getting causality, likewise, the structure standard of the reasonable follows the undertaking.

For the following project, we have done some extensive data visualization of which some examples are listed below.

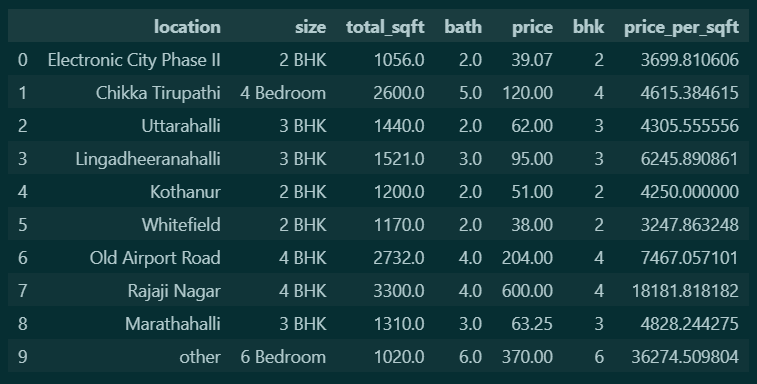
****

****

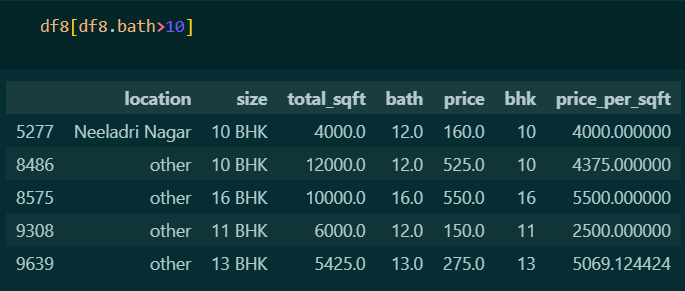
The first example of my data analysis was that in some locations (for ex: Rajaji Nagar & Hebbal) it was to be seen through the graphs that the prices of 2BHK houses were more than the prices of 3BHK which is anomalous. So to overcome that we removed such outliers for every location if they had these kind of anomalies.

****

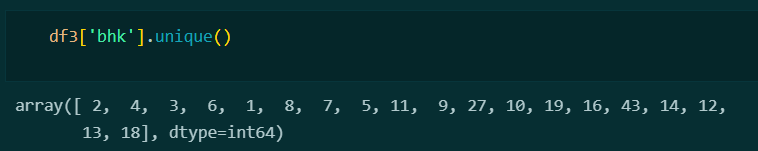
Another outlier that I detected and removed in my visualization was that price per square feet of some houses were either too high or too low. As you can see above that the max price per square feet is around 1.7 lakh rupees which is too much price per square feet and the minimum price per square feet is 267 rupees which was very low.



Another outlier that was detected was that for a 2BHK house, the size of 2BHK is 1056. If we divide 1056 by 2, it’s approximately 500 sqft which is impossible. So, we have removed this kind of outlier as well.



The bathrooms in some house are very unusual. Some houses have 10-16 bathrooms which is very unusual. We have to remove this kind of outlier as well by having the condition of total bedrooms = total bathrooms + 1 as per the domain knowledge.

****

For BHK also, I had the same kind of outlier as for bath i.e., we had numbers of BHK like 43, 27 etc. which was unusual to have in a house. So, I had to remove such kind of outlier as well by taking a minimum threshold of 6 bedrooms.

**3.3 Final Outcome**

After doing an extensive data analysis, I came up with the following points that will help us to make our dataset more compatible for training purpose.

**1) The NA values were filled with the median of the particular column instead of dropping it in order to bring accuracy to our prediction.**

**2) For making outlier detection easier, we have to make a new column that contains price per square feet.**

**3) Due to around 1300 locations, we have made a new column named “other”. This column will have only those locations that have only one or two data points.**

**4) In exploration of BHK and Bath columns, we saw that there were some unusual data values in BHK like 43, 27 etc. and so was in bath which had values like 16, 13 etc. which needs to be removed.**

**5) In the price per square feet column, prices of some houses were either too high or too low which was anomalous and had to be removed.**

**6) In some locations, prices of 3BHK are greater than 4BHK or prices of 2BHK are greater than 2BHK. That had to be resolved for correct prediction of price.**

**7) The size of some houses for ex: 2BHK was around 1000 sqft which was impossible. So that issue also had to be resolved.**

**4) Experimentation**

In Machine Learning, we use various kinds of algorithms and techniques to allow machines to learn the relationships within the data provided and make predictions based on patterns or rules identified from the dataset. So, regression is a machine learning technique where the model predicts the output as a continuous numerical value.

Regression analysis is often used in finance, investing, and others, and finds out the relationship between a single dependent variable dependent on several independent ones. For example, predicting house price, stock market or salary of an employee, etc. are the most common regression problems.

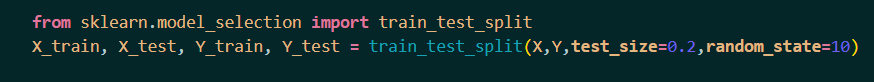
**Train-Test-Splitting**

The train test split technique can be used for classification and regression problems to test machine learning algorithms. The procedure takes the given dataset and splits it into two subsets:

* Training dataset: it is used to train the algorithm and fit the machine learning model.
* Test dataset: Using the input element from the training data, the algorithms make predictions.

The model is first to fit on the available data with known inputs and outputs. It is then run to make predictions on the rest of the data subset to learn from it. This can be used to make predictions on future data sets where the expected input and output values are non-existent.

For this model, I have split my dataset into training and test dataset as well. The test size is 20 percent whereas the training dataset is 80 percent.



Keeping this in mind, I have used some of the most commonly used regression algorithms to train my model for predicting the house prices.

**4.1 XGBoost**

**XGBoost** is an optimized distributed gradient boosting library designed to be highly **efficient, flexible** and **portable.** It implements machine learning algorithms under the [Gradient Boosting](https://en.wikipedia.org/wiki/Gradient_boosting) framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way. The same code runs on major distributed environment and can solve problems beyond billions of examples.

**4.2 Ridge**

Ridge model solves a regression model where the loss function is the linear least square function and regularization is given by the l2-norm. This estimator has built-in support for multi-variate regression (i.e., when y is a 2d-array of shape (n\_samples, n\_targets)).

It minimizes the objective function: ||y - Xw||^2\_2 + alpha \* ||w||^2\_2

**4.3 Linear Regression**

Simple linear regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent(x) and dependent(y) variable. Based on the given data points, we try to plot a line that models the points the best. The line can be modelled based on the linear equation shown below.

y = a\_0 + a\_1 \* x

The motive of the linear regression algorithm is to find the best values for a\_0 and a\_1. Before moving on to the algorithm, let’s have a look at two important concepts you must know to better understand linear regression.

**4.4 Random Forest**

A random forest is a machine learning technique that’s used to solve regression and classification problems. It utilizes ensemble learning, which is a technique that combines many classifiers to provide solutions to complex problems.

A decision tree consists of three components: decision nodes, leaf nodes, and a root node. A decision tree algorithm divides a training dataset into branches, which further segregate into other branches. This sequence continues until a leaf node is attained. The leaf node cannot be segregated further.

**4.5 KNN**

A k-nearest-neighbor is a data classification algorithm that attempts to determine what group a data point is in by looking at the data points around it.

An algorithm, looking at one point on a grid, trying to determine if a point is in group A or B, looks at the states of the points that are near it. The range is arbitrarily determined, but the point is to take a sample of the data. If the majority of the points are in group A, then it is likely that the data point in question will be A rather than B, and vice versa.

**4.6 Decision Tree**

Decision tree builds regression or classification models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches, each representing values for the attribute tested. Leaf node represents a decision on the numerical target. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

**4.7 SVR**

Support Vector Regression (SVR) is a regression function that is generalized by [Support Vector Machines](https://en.wikipedia.org/wiki/Support-vector_machine) - a machine learning model used for data classification on continuous data. Unlike in the Ordinary Least Squares, the SVR model sets a threshold error allowance ϵϵ around the regression line such that all the data points within ϵϵ are not penalized for their error.

**4.8 Lasso**

Lasso regression is a regularization technique. It is used over regression methods for a more accurate prediction. This model uses shrinkage. Shrinkage is where data values are shrunk towards a central point as the mean. The lasso procedure encourages simple, sparse models. This particular type of regression is well-suited for models showing high levels of multicollinearity or when you want to automate certain parts of model selection, like variable selection/parameter elimination. Lasso Regression uses L1 regularization technique (will be discussed later in this article). It is used when we have more features because it automatically performs feature selection.

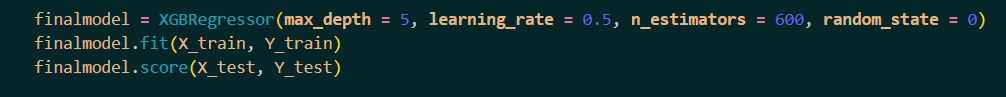
**5) Result**

To achieve the results, various Regression techniques are utilized in python language. Various factors which affect the house pricing are considered and further worked upon them. Machine learning has been considered to complete out the desired task. Firstly, data collection is performed. Then data cleaning is performed to remove all the errors from the data and make it clean. Then data pre-processing is done. Then with the help of data visualization, different plots are created, which intends to depict the distribution of data in different forms. Towards the end, the business costs of the houses were determined with exactness and accuracy. This could be achieved because a simple stacking algorithm called as GridSearchCV is used to improve the accuracies of the various regression algorithms that are applied on our house pricing dataset so that they would provide better results.

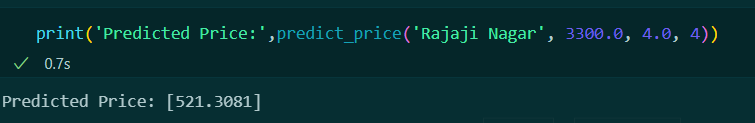
**5.1 GridSearchCV**

GridSearchCV is a function that comes in Scikit-learn’s model\_selection package. It helps in performing hyperparameter tuning in order to determine the optimal parameters for a given model. The performance of a model significantly depends on the value of hyperparameters. Choosing the parameters manually could take a considerable amount of time and resources and thus we use GridSearchCV to automate the tuning of hyperparameters.

****In this model, we have used the GridSearchCV for the following algorithms mentioned above as well which found the best parameters for each and every algorithm to perform at its best.

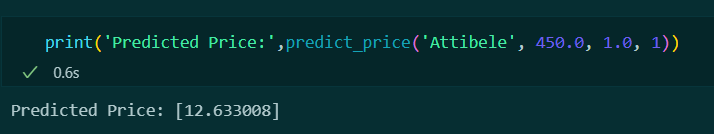
****As we can see, The XGBoost model is performing the best with a total score of 0.85 or 85 percent whereas all the other algorithms are not having a score above 85 percent. The optimal parameters used the winning model were as follows:

After training the model using the dataset, I predicted some prices of the houses that were in the dataset.













As you can see, the two predictions are accurate enough in predicting the correct price of the houses on comparing the code output with the values of price in the dataset file.

**6) Conclusion**

In future, many more algorithms can be applied on this dataset such as decision tree, Naïve Bayes, SVM etc. and find out their respective accuracies and use them to predict a better outcome and hence increase the accuracy. Hence, it would be of great help for the government and the people themselves. Regression algorithms are initially taking up for our project but in the future, this can also be achieved using the classification algorithms. The classification algorithms can be used and it can also be applied to our house pricing dataset and see if they are being applied properly or not. The accuracy and precision of these algorithms can also be improved according to our needs. This would be of great help for the people as they would get to choose from a variety of options open up to them. They can choose the house that best suits their budgets so that they don’t have to take any kind of loan from the banks.

In the future, an application can also be developed for the same. That would make it even easier for the people to select the houses that best suits their budgets. More factors that can affect the house pricing of a particular area will also be considered. Although many algorithms are used in our project still, many more regression and Page classification algorithms are used to top make our project and make it more helpful for the people. Various methodologies from the field of machine learning are used to make our project more relevant. In the future the model deployment of more algorithms can be performed to achieve accurate results.

**7. REFERENCES**

[1] <https://www.kaggle.com/datasets/ahmettezcantekin/beginner-datasets?select=beginner_datasets>

[2] <https://github.com/SanikaVT/Real-Estate/blob/master/Real_Estate_CaseStudy.ipynb>

[3] <https://github.com/shrutijain0/Real-Estate-Price-Prediction>

[4] <https://python.plainenglish.io/data-science-project-real-estate-house-price-prediction-website-df71ac98a132>

[5] <https://towardsdatascience.com/create-a-model-to-predict-house-prices-using-python-d34fe8fad88f>

[6] <https://github.com/satishgunjal/House-Price-Prediction-Project/blob/master/House_Price_Prediction_Project.ipynb>

[7] <https://www.reechcorp.com/insights/machine-learning-real-estate/>

[8]<https://www.researchgate.net/publication/329092297_Identifying_Real_Estate_Opportunities_Using_Machine_Learning>

[9] <https://www.ijert.org/real-estate-price-prediction>

[10] <https://hackernoon.com/what-is-one-hot-encoding-why-and-when-do-you-have-to-use-it-e3c6186d008f>

[11] <https://www.analyticsvidhya.com/blog/2021/05/5-regression-algorithms-you-should-know-introductory-guide/>

[12] <https://xgboost.readthedocs.io/en/stable/>

[13] <https://scikit-learn.org/stable/supervised_learning.html#supervised-learning>

[14] <https://github.com/qlanners/scikit-learn_disect/blob/master/all_model_tester.ipynb>

[15] <https://www.jigsawacademy.com/blogs/ai-ml/train-test-split>