**Problem statement:**

A house value is simply more than location and square footage. Like the features that make up a person, an educated party would want to know all aspects that give a house its value. For example, you want to sell a house and you don’t know the price which you can take — it can’t be too low or too high. To find house price you usually try to find similar properties in your neighbourhood and based on gathered data you will try to assess your house price.

1. **Summary of problem statement, data and findings**

>> The proposed project is about determining the ‘price’ of a house based on all the features of the available dataset and not only on the location and square footage. The problem statement also clearly states that the value of the house is not being predicted only from the buyer perspective but also from the seller perspective with the given dataset to decide on the correct ‘price’ to be tagged for the house. To have a clear understanding of the given dataset we started our work by analyzing it in excel workbook using filter which gave some interesting insights on the given dataset,

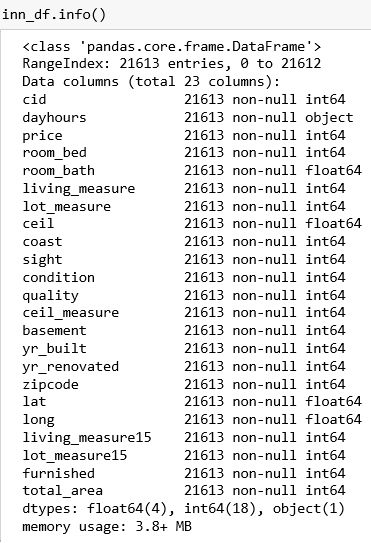
1. The room\_bed attribute had a value 33, which when compared with the total area, indicates this value could be an outlier.
2. The room\_bath(no of bathrooms/no of bedrooms) attribute has some integer non zero value when the room\_bed has ‘0’ value, which indicates wrong values in the data set.
3. The attribute lot\_measure15 is only a part of the total area, but in few cases the value is greater than total\_area which is not possible.
4. The attribute ceil (Total floors (levels) in the house) has few values in decimals. How can the total floors be in decimals
5. The attribute year\_renovated has few values ‘0’ but the lot\_measure15 has greater value than lot\_measure in few cases. This indicates that few values of ‘0’ might be missing values in year\_innovated.
6. The correlation between attributes is not clear from excel workbook, which can be seen using python notebook.

With this inference it made us easy to start our analysis for the given dataset.

**Note:** Since we have one target variable – ‘price’ and all other attributes are independent variables, we will be training our model based on the independent variables. So this problem lies under S**upervised learning method**.

**2. Summary of the Approach to EDA and Pre-processing**

* As a first step we imported the dataset to the python notebook.
* Then we used the info() function to see the type of each attribute which helped us to sort between int, float and object.



From the info() function we can clearly see that the attribute ‘dayhours’ is of the type ‘object’, so using this attribute data analysis is cannot done in python notebook. Either this attribute has to be converted to integer or float type if it is going to provide more value to the dataset or it can be removed if the attribute is not that important. In this dataset ‘dayhours’ attribute is not that important so it can be removed for further analysis.

* Then we used the describe function to find the statistical values for each attribute. The statistical values helped to understand the distribution of the dataset.
* Before starting the univariate analysis we removed the cid and dayhours attributes as they were not support the model in predicting the target column, (i;e predicting the price of the house).

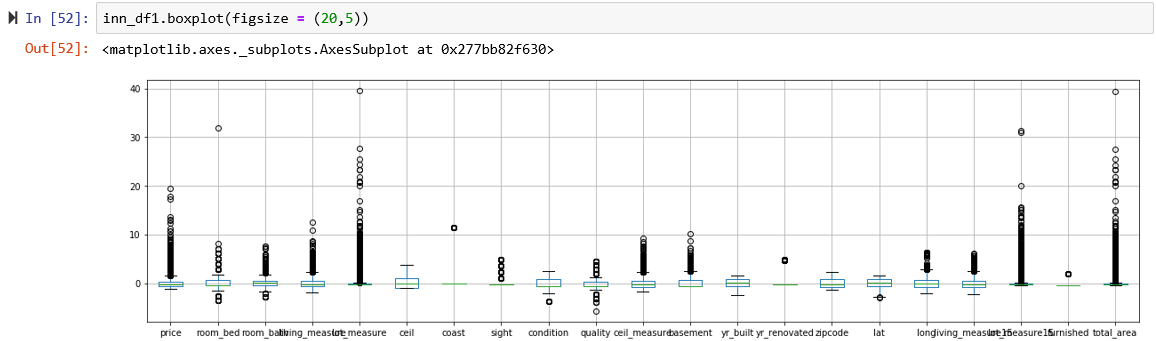
**Univariate Analysis**

* The univariate analysis was performed with the distance plots to find the distribution of each attribute.
* We see that almost all the attributes are right skewed, this might be because of the outliers in the attributes
* To find whether my attribute is normally distributed or not we used skewness and kurtosis to check whether it is within the range of [-2,2]
* To find the range of outliers we used boxplot.
* We also measured the count of outliers, so that it helps us see how much amount of outliers can be removed.

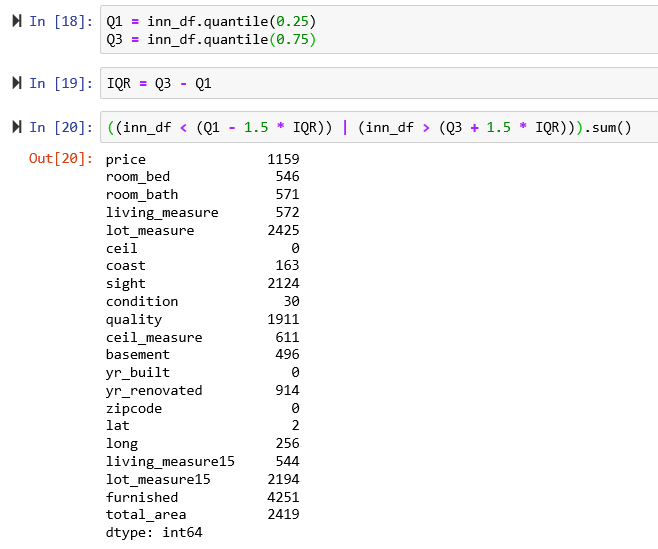
**Univariate Analysis – Distance Plots**



* From the distance plot we can see that more than 10 attributes are categorical. Among those attributes many are having non-gaussian type distribution. The other attributes which are not categorical are mostly right skewed. So as most of the attributes are having non-gaussian type distribution they possess outliers. The outliers can be visually seen using boxplot.



* From the boxplot we can clearly see that except few attributes like ceil, yr\_built & zipcode all the other attributes having outliers.
* For better analysis of dataset the outliers have to be removed, this can done only when the no of outliers for each attribute is found.

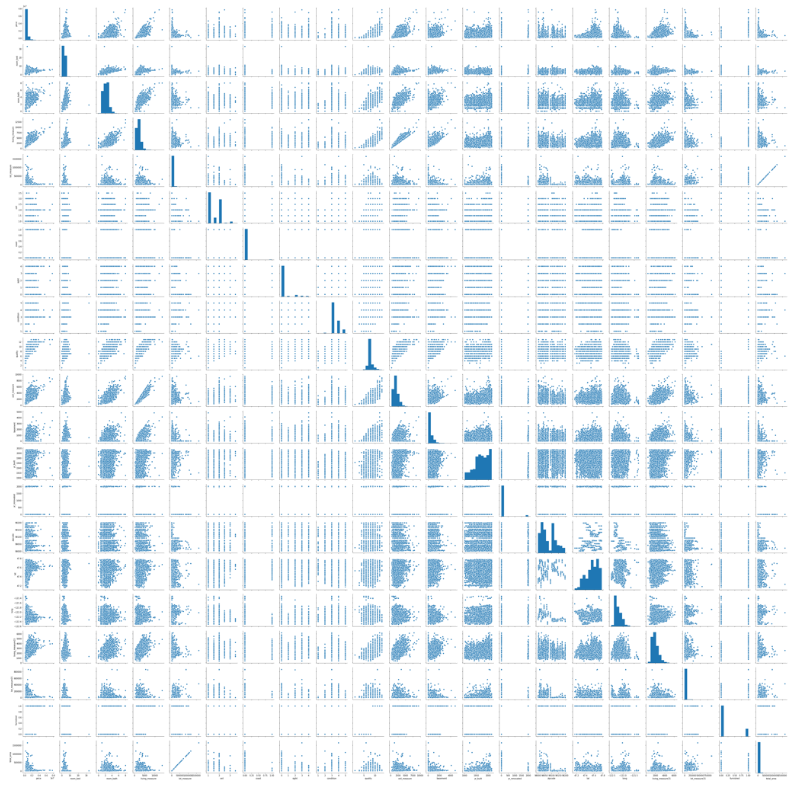


* This clearly gives us information on which attributes have more outliers. With this we started our analysis for reducing the outliers and better utilization of the dataset.

**Bivariate Analysis**

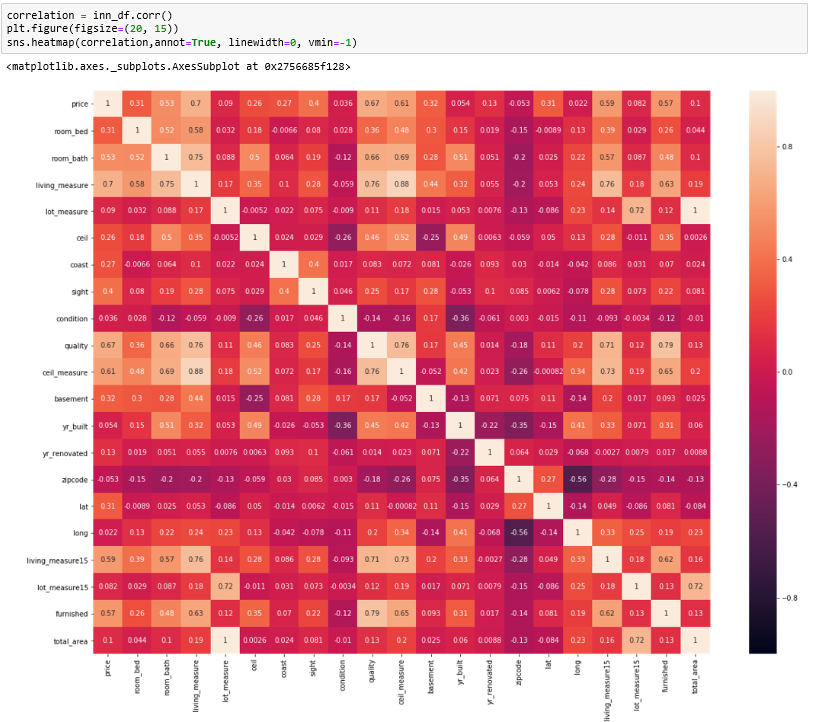
* We then started our bivariate analysis to understand the relation between attributes.
* There are two methods which can clearly show the relationship between attributes, one is pairplot and other is correlation heat map.

**Pairplot**



* From the pair plot we can see linear correlation between few attributes and few has clustered correlation with no proper relation with each other.
* Let us do the correlation analysis between each attribute using heat map to exactly find the relation between two attributes.

**Correlation Heatmap**



* From the heat map it is very clear that total\_area and lot\_measure has a correlation of '1' so any one column can be neglected.
* living\_measure & ceil\_measure has a correlation of '.88' which means all the houses are almost built with a ceiling.
* living\_measure has also got a good relationship with quality & room\_bath, which means with greater living capacity the quality is good and the no of bathrooms/bedrooms increases.
* living\_measure15 is just an extension of living\_measure, so has good correlation
* All the furnished houses have better quality.
* living\_measure, quality, ceil\_measure, furnished & room\_bath have better correlation with the target variable ('price'). So for predicting the price of a property these attributes play a major role in this dataset.

**3. Deciding Models and Model Building**

* Before we started to modify the dataset, we developed a simple linear model to find how it performs on the existing dataset.



* We splitted the dataset into train and test data, then we trained the dataset using simple Linear Regression.
* We then tried to find the score for test data set which gave us around 70%
* The target column of the dataset is said to be continuous. So we cannot use any classification model for testing the dataset.
* We then tried to find the score using Logistic Regression and SVM, which showed us an interesting error that the dataset is said to be continuous.
* Then we understood that even few model developing methods in regression also cannot be used for testing the continuous data directly. So we used a label encoder to convert continuous dataset to numerical form which made the dataset to be suitable for testing using Logistic Regression and SVM. But even after that the score was less than 1%. So these models are not suitable for the given dataset.
* Ridge & Lasso methods gave a model score of 70%
* Decision Tree Regressor gave a model score of 76%
* Gradient Boost Regressor(GB) gave a model score of 87%
* Random Forest Regressor(RF) gave a model score of 87.2%
* Comparing all the model scores Gradient Boost Regressor & Random Forest Regressor seem to perform well on the given dataset.

**4. How to improve your model performance?**

The different iterations through which we try to improve our model score are,

1. Removing the outliers using (mean+3\*SD) on the existing dataset and developing a model using GB and RF method.
2. Replacing ‘Zeros’ from the attributes which are providing wrong information with their respective column medians and developing a model using GB and RF
3. Using PCA to analyse the minimum number of attributes needed for developing a better model using GB or RF which can provide atleast 95% variance.
4. Analysing the model performance using GB and RF by removing attributes.
5. Using Polynomial function on the existing dataset and test the model score using GB & RF
6. Changing the learning rate and the estimators in GB & RF