

HOUSE PRICE PREDICTION

Submitted by:

BALPREET KAUR

**ACKNOWLEDGMENT**

The references I used are:

* Pandas documentation.
* Libraries documentation like:
* [Scikit-learn](https://scikit-learn.org/)
* Seaborn
* Used Stack Overflow website for some technical issues.
* Other tutorial websites for some visibility such as:
  + <https://wellsr.com/python/how-to-make-seaborn-boxplots-in-python/>
  + <https://wellsr.com/python/how-to-make-seaborn-boxplots-in-python/>
  + <https://www.py4u.net/discuss/187401>
  + <https://www.kaggle.com/ammar111/house-price-prediction-an-end-to-end-ml-project>
  + <https://www.researchgate.net/publication/347584803_House_Price_Prediction_using_a_Machine_Learning_Model_A_Survey_of_Literature>
  + <https://www.section.io/engineering-education/house-price-prediction/>
  + <http://www.iraj.in/journal/journal_file/journal_pdf/12-477-153396274234-40.pdf>
  + <https://www.dataquest.io/blog/kaggle-getting-started/>
  + <https://towardsdatascience.com/machine-learning-with-python-regression-complete-tutorial-47268e546cea>
  + <https://www.codegrepper.com/code-examples/python/boxplot+for+all+columns+in+python>
  + <https://ieeexplore.ieee.org/abstract/document/8614000>
* Books consulted:
  + <https://www.researchgate.net/publication/340896273_Machine_Learning_based_Predicting_House_Prices_using_Regression_Techniques>
  + <https://www.researchgate.net/publication/343241145_Housing_Price_Prediction_via_Improved_Machine_Learning_Techniques>
  + <https://www.diva-portal.org/smash/get/diva2:1456610/FULLTEXT01.pdf>

Mu J, Wu F, Zhang A

**Housing Value Forecasting Based on Machine Learning Methods**

Abstract and Applied Analysis, 2014 (2014), pp. 1-7

doi:10.1155/2014/648047.

Lu S, Li Z, Qin Z, Yang X, Goh RSM. A hybrid regression technique for house prices prediction. 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM) 2017. doi:10.1109/ieem.2017.8289904.

**INTRODUCTION**

* Business Problem Framing

The business problem is the following one:

Decide whether to invest or not on the prospective properties depending on the comparison between expected and actual prices.

How this can be related to the real world:

This study is useful whenever there is a study of house prices and can guide the features and parameters that are important in order to get the best property by studying its best features.

* Conceptual Background of the Domain Problem

Domain concepts that will be useful for this project is to fully understand the meaning of the features and understand the correlation with the target and study and get full knowledge of the important features concluded by the algorithm. This last acknowledgement should be required after having analysed and conclude the algorithm and get to know the best of the best features concluded by the algorithm in order to give the best conclusions and recommendations.

* Review of Literature

Most of the literature review is based on online full-text articles, open access articles and Research Gate publications and the Towards Data Science collection and some machine learning books to have solid foundation in regression techniques, regularization and modeling the ML learning about how it can be applied accurately to house price prediction

I checked the attached documentation and identified the names of the features. And researched different websites in order to understand fully the meaning of each of one.

So, basically in the research part, I did try to understand the changes in properties prices over time and how we can describe those changes through its respective properties we have in our data frame. And also understand how the market of house pricing works as of point of view of Sale and Offers and understand the clients demands regarding the property’s characteristics.

And my main goal for the research was to find, understand and have the better knowledge I can have of the properties features and market in order to better understand the relationship between all features and apply the intensive understanding and knowledge of these features on my price prediction project.

It was helpful to have a quick effective research on the concept and how the market works and how the features impact on the price in order to understand better the context.

It was also much more helpful the intensive research I did on the characteristics as it helped me understand for example how a client does the research on the house characteristics and demand the finalized house for their personal use. And on the other hand, how an investor of houses does their research on the characteristics of the house to find the best deal between supply and demand in order to get the best plus value.

* Motivation for the Problem Undertaken

Being honest, my real motivation was to get some intensive good knowledge of the house market and understand all the variables that comes in and are important to evaluate before we decide and select any property.

I am saying it as honest motivation as maybe in a near future I would be investing in this market. So, that’s why I did work much more work on the research part like 6 days and half, only on the research investigation and understanding all the features I can understand before I start working on the machine learning process.

**Analytical Problem Framing**

* Mathematical/ Analytical Modeling of the Problem:

I did statistical description and studied correlation of the features of the dataset I had for training purposes.

* + In statistical description I checked a first view of skewness. I checked if we had any type of skewness (at first sight) and if so, what type of skewness we had by checking the median, mean and mode.
  + Also, as important study in the statistical descriptive analysis, we checked and got some first insights on the outliers comparing the maximum of each feature with the 75%.

Then in the correlation matrices and heatmaps we did understand the correlation and the quantification of the impact of each feature on our target price value.

And after that, we did an intensive study on the features understanding their important relations and the impacts each feature has on the other one.

I also did print some VIFs values in order to get ride of the features that were mostly correlated to other features and were impacting more due to high multicollinearity.

And after having studied the important features relationships and fully understand the data we have, we started with the process of modelling. As it was a case of price prediction, we followed the basics steps and instructions for linear relationship studies as price has more or less linear type of relation with the other features.

In summary, we did pre-process the data and evaluate the prediction accuracy of the models. It has multiple stages that are required to get the prediction results.

These steps are:

-processing: both datasets will be checked and pre-processed using the methods which have various ways of handling data such as mode, median or mean for each feature.

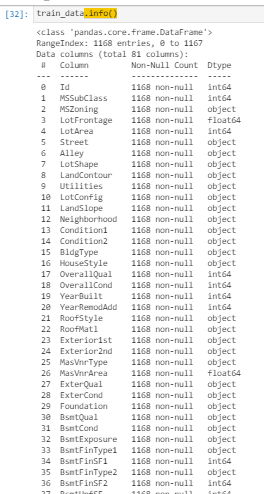
- Data splitting: separating the dataset into two parts n order to train the model with one and use the other in the evaluation. The dataset will be split 75% for training and 25% for testing.

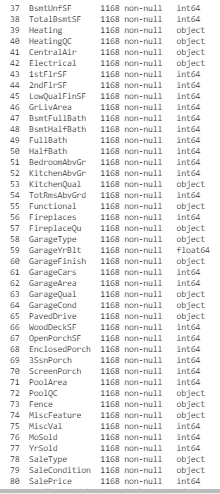
- Evaluation: the accuracy of both datasets will be evaluated by measuring the R2 and RMSE rate when training the model alongside an evaluation of the actual prices on the test dataset with the prices that are being predicted by the model. 10

- Performance: alongside the evaluation metrics, the required time to train the model will be measured to show the algorithm vary in terms of time.

- Correlation: correlation between the available features and house price will be evaluated using the Pearson Coefficient Correlation (default correlation method) to identify whether the features have a negative, positive or zero correlation with the house price.

* Data Sources and their formats





The data was given in the ziP format file and attached to the study case. In the dataset we have a total of 1168 rows and 81 columns. Checking the data types, we have a total of 3 floats and 35 integer as numerical features and 43 features as object type.

So, for our analysis, throughout the process, we reconverted the type of data we have in order to analyse it more in detail. For example, we had to convert the features and encode them in order to add them in the machine learning algorithms.

Several features had object data type when that should have been numerical float or integer data type. So, I needed to convert them to numerical for proper analysis and also for some statistical checks such as outliers.

* Data Preprocessing Done

The steps we followed for data processing and cleaning was to impute the missing values of the training set.

So, basically, summarizing we imputed 0 for numerical missing values when the NA meant to be “No data - source” which mostly was because there was not such feature info or details. For example: In case of Feature PoolQC, NA means there is No Pool. So, missing values for features like this one was replaced by a zero as there was no such data in the actual world.

And regarding the categorical object type data, we did impute the missing values by the most frequent value of the corresponding feature.

* Data Inputs- Logic- Output Relationships

So, when we study the relation ship between our features with regards our output target, we have several considerations to have in mind:

* + First of all, the data format of the input and output:
    - As we said, we had several different formats of our input data such as integer, float and object.
    - Regarding the integer and float data type, there was no such issue as these were already numerical type data.
    - The problem was basically with other object type data when considering them for the relationship purposes. And for these, we had to label encode them in order to convert them in the right format for the correlation studies.
  + Relation and affection between the input and output:
    - When checking the relationships between the features and the target, we see some feature like GarageCars, GarageArea, TotalBsmtSF, 1stFlrSF , FullBath, TotRmsAbvGrd, YearBuilt, YearRemodAdd that are more correlated and have more possibility to impact more on our target variable SalePrice compared to others like BsmtFinSF2, BsmtHalfBath, MiscVal, Id, LowQualFinSF and YrSold which have lowest correlation with regards our target variable SalePrice.
* Hardware and Software Requirements and Tools Used

I used Jupyter-notebook and Python 3 language.

Imported required libraries such as:

- pandas

- numpy

- matplotlib.pyplot

- seaborn

- sklearn

For modelling, we imported different algorithms, pre-processing, splitting tools, data transformation and metrics from different libraries:

* From sklearn.linear\_model imported LinearRegression
* From sklearn.linear\_model import Lasso (Lasso is a type of regression that uses shrinkage)
* From sklearn.tree imported DecisionTreeRegressor
* From sklearn.neighbors imported KNeighborsRegressor
* From sklearn.svm imported SVR
* From sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor, AdaBoostRegressor

All these previous imports were executed in order to fit and train the data.

* From sklearn.model\_selection imported train\_test\_split

The train\_test\_split was used to split the data in the training and testing set.

* From sklearn.preprocessing import LabelEncoder

LabelEncoder helped me to convert the object type data to numerical for statistic analysis including correlations, plotting purposes and machine learning algorithm training.

* From scipy.stats import skew 🡪 for skewness calculation.
* From statsmodels.stats.outliers\_influence import variance\_inflation\_factor 🡪 to calculate VIF and check multicollinearity.
* From scipy.stats import zscore 🡪 to calculate zscore and check the outliers.
* From sklearn.preprocessing import power\_transform🡪 to transform and reduce the skewness.
* From sklearn.preprocessing import StandardScaler🡪 to normalize the data and put it in the same scale for better comparison.
* From sklearn.metrics import r2\_score, mean\_absolute\_error,mean\_squared\_error🡪 metrics error to calculate for algorithm comparison and will help us to select the best algorithm for our case.
* From sklearn.model\_selection import GridSearchCV
* From sklearn.model\_selection import RandomizedSearchCV

The gridsearch CV and randomizedsearch CV is used to get the best CV we can get in order to get the best scores through our best algorithm.

* From sklearn.model\_selection import cross\_val\_score

Cross\_val\_score helps us to calculate the score of the cross validation.

* Executed %matplotlib inline to call the predefined functions that will plot and default style the plot for you.

Also did import warnings in order to ignore them:

import warnings

warnings.filterwarnings('ignore')

And also imported pickle for the model saving.

**Model/s Development and Evaluation**

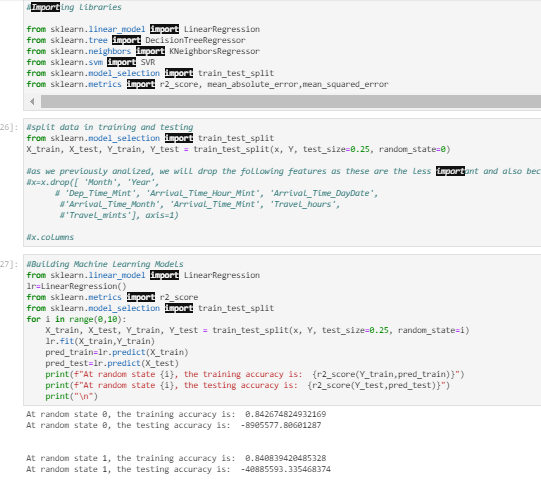
* Identification of possible problem-solving approaches (methods)

The main approach I decided to follow is to use linear algorithm

and a few other regressions algorithms as the model should do a prediction of a continuous target variable.

* Run and Evaluate selected models

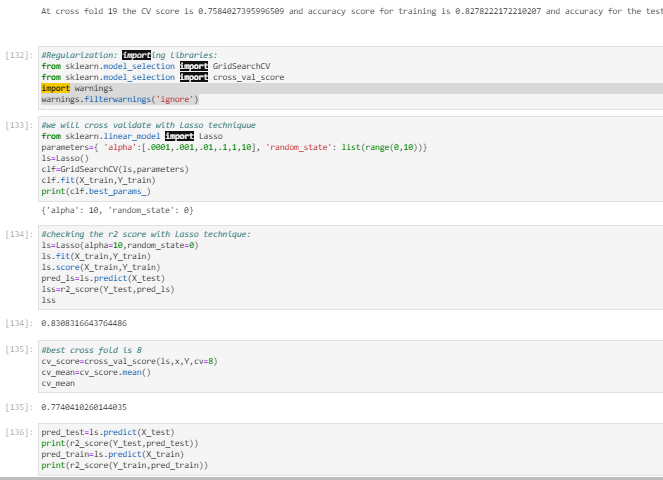
Then, after checking and doing some statistical descriptive analysis, we decided starting with simple Linear regression modelling.





As you see above, we trained the algorithm, we did some study on the best CVs and printed the results score and checked for MAE, MSE and RMSE so that we can decide which algorithm is working best for our study case.

After having printed the results score and errors metrics for linear regression, we tried other algorithms in order to check if we can get better results than the linear regression. We did some Lasso Regression too in order to improve the current results and scores.



I also tried some ensemble techniques such as Random Forest Regressor and compared the results:





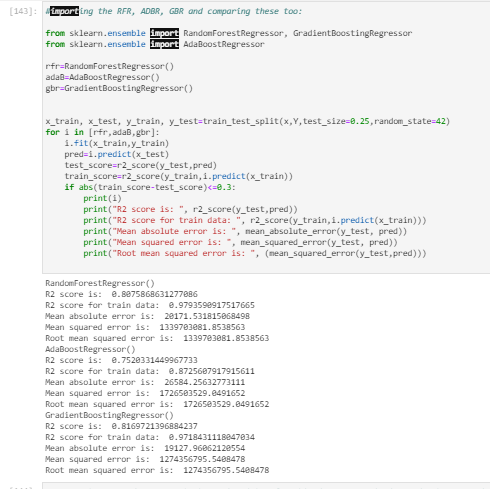
I also decided to check and compare the results of:

* + DecisionTreeRegressor
  + SVR
  + KNeighborsRegressor
  + LinearRegression
  + RandomForestRegressor
  + AdaBoostRegressor
  + GradientBoostingRegressor

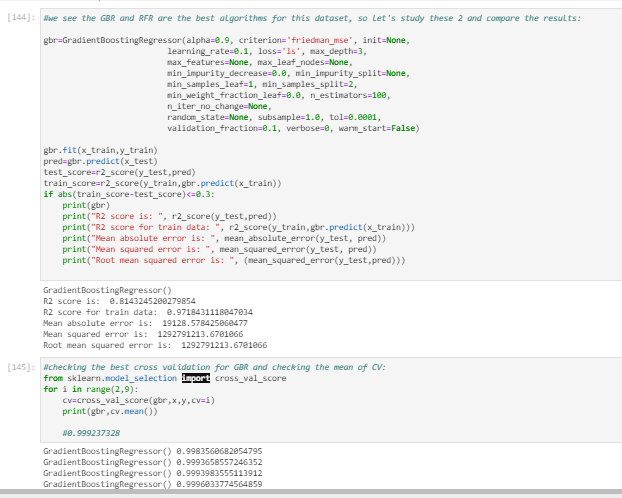
Checked these algorithms metrics and compared the results and scores:



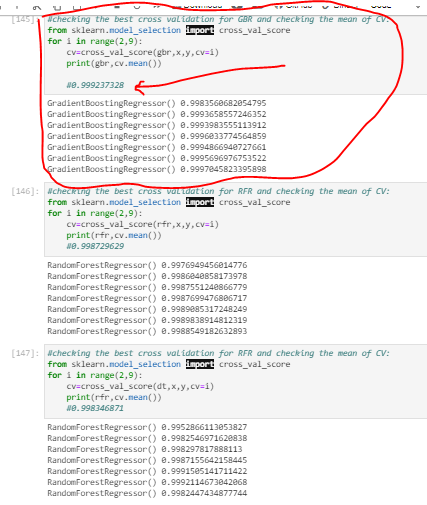
Also compared the results of RFR, ABR and GBR:



As we see GBR and RFR are the best algorithms that we can get from the previous steps, I decide to check the r2 score and errors metrics MAE, MSE and RMSE on these two and compare the results of both:

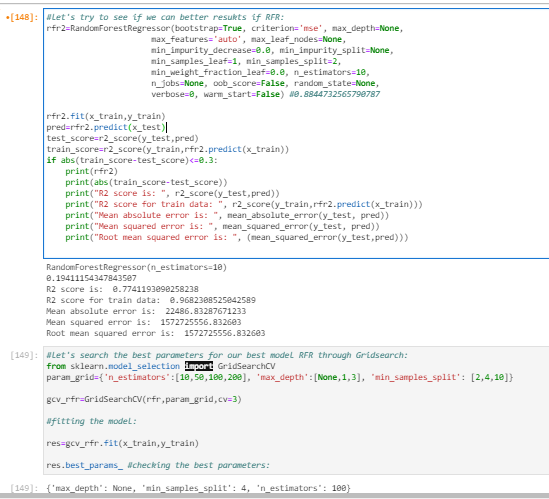


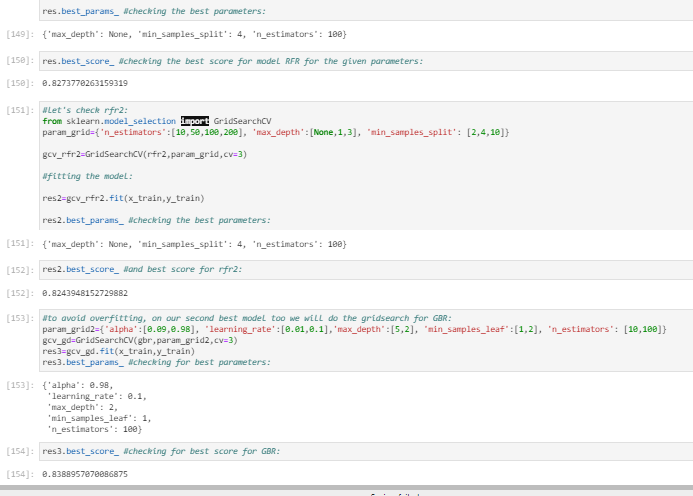
When checking and comparing the CV mean of each algorithm, we see that GBR is the algorithm that wins the position as it has 0.999237328 as result of mean of the CV.mean:



And this first position mean 0.999237328 of the cv.means is the highest I got after comparing the mean of the cv.means of all 3 best algorithms.

Since we know now that GBR is taking the number one place as the best algorithm. Let’s check if either we can improve the results with RFR or if it is possible for RFR to overcome the GBR results doing some GridSearchCV:





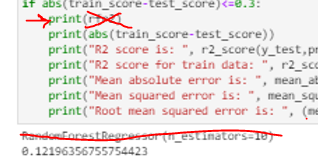
Here above screenshot we clearly see that our best algorithm GBR has better score results (83.88%) to offer than the algorithm RFR which has score of 82.43%.

Let’s now create the best instance of the model GBR and check the score results and the metrics of error: MAE, MSE and RMSE alongside with R2 Score:

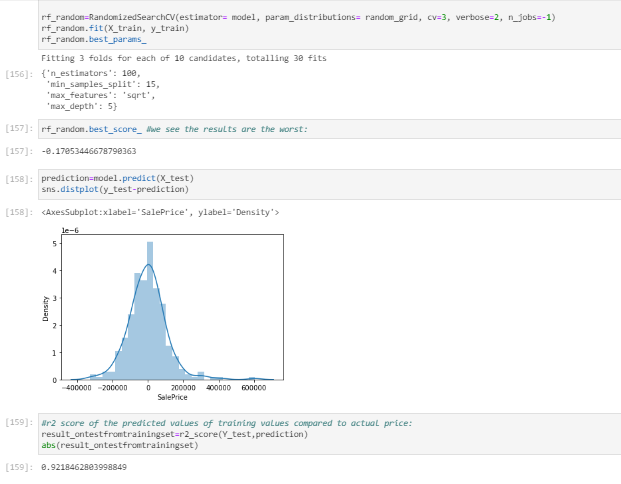


Trying to do a Randomized Search CV on the same best model till now GBR.

I just realized the error in the last moment when writing this report and the error I did is to print the name variable rfr2 when I should have put gbr there:

(I should have printed the name of GBR instead of printing the RFR, [😅](https://emojipedia.org/grinning-face-with-sweat/))

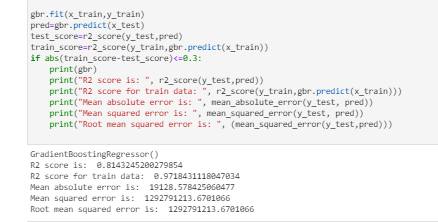
Ok, so as we know our best algorithm is GBR and we did the Randomized Search CV on the same best model which is till now GBR.



Even after checking both algorithm scores, we clearly see that GBR scores are way better. And the test score is 92,18% from the split test data.

* Key Metrics for success in solving problem under consideration

Basically, I used the 3 errors metrics and R2 score alongside the test score in order to compare and choose the best algorithms as shown below:

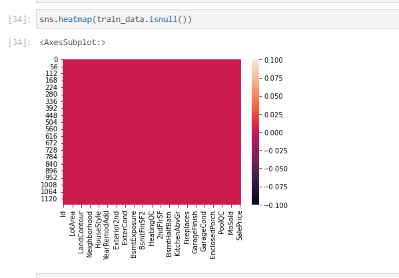


I considered all the results and scores one by one and compared this way all algorithms and chose the best one which is GBR.

You can easily understand the all the process I did when you read my python script as it has complete similar comments on most of the main steps and everything written here will make more sense and you will understand all the steps and considerations commented here. Please go through all the comments in the given code script and will also make it easy to understand this report.

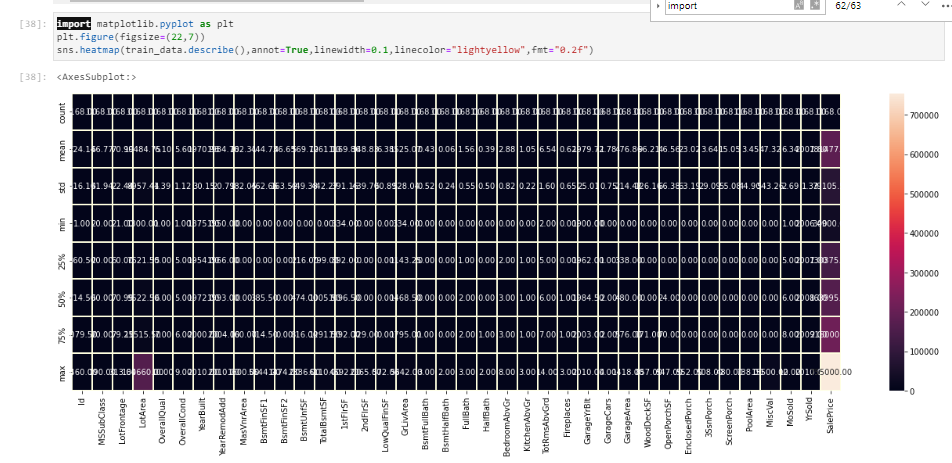
* Visualizations

First of all, we plotted the missing values to check if we were missing any at this stage.



At this stage we checked and confirmed there is no missing values left.

We also plotted descriptive details of our data:

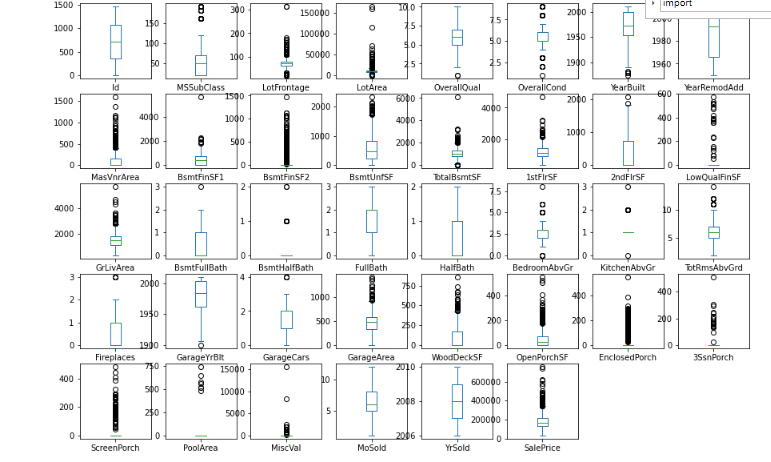


From the above plot we could check at first sight if we had any skewness or outliers. Also checking the total number of instances, we could check and confirm that we do not have any missing values. With the above plot we also can check if we have same or different scaling of the features. If so, apply any kind of normalization technique such as standard scaler or tmin-tmax scaler.

On the one hand, for all the features where the mean is smaller than the median, we can say it has left skewed data which also means negative skewness. On the other hand, if a feature has the mean higher than the median, than it is right skewed which is also called as positive skewness.

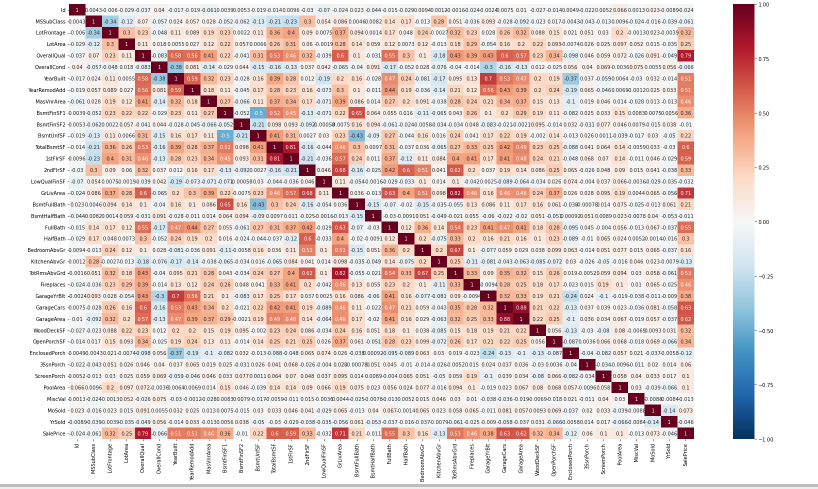
For outlier check at first sight, we can notice it when we have larger values for the maximum compared to 75% percentile values.

We also plotted box plots for all our features including the target in order to check outliers in all the features:

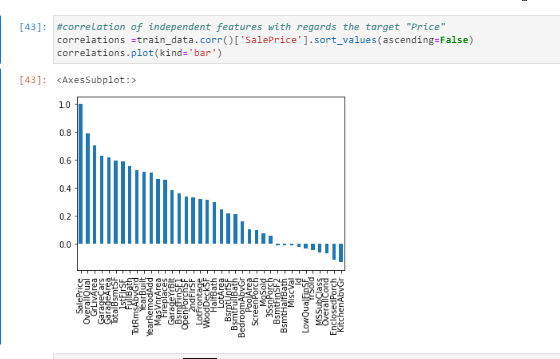


The circles are the outliers for each feature. Almost all the features have outliers including our target variable which is one of the variables that has high number of outliers.

Then, we also plotted the correlation between features through a heatmap in order to see easily the best correlations between all our features including our target “SalePrice” :



If we concentrate on the target and check the best correlations against it, then we will see that the following features are considered the best correlated features within our dataset:



So here are the answers to the following questions:

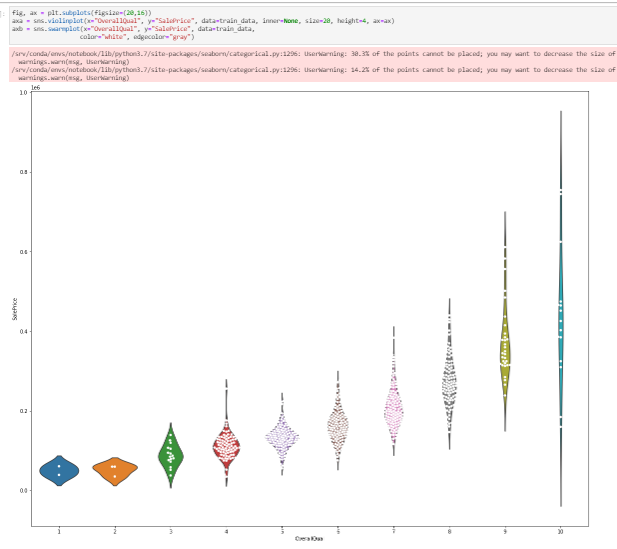
• Which variables are important to predict the price of variable?

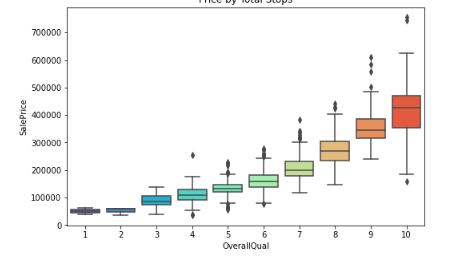
We see several features such as OverrallQual, GrLivArea, GarageCars, GarageArea, TotalBsmtSF, 1stFlrSF, Fullbath, TotRmsAbvGrd, YearBuilt, YearRemodAdd among the features shown in the above screenshot are the features that have higher impact comparing all the available features.

• How do these variables describe the price of the house?

These variables have higher impact on the target then the others when comparing the percentage of correlation. So, the higher correlation, more probability to have higher impact on the Sale Price of the house.

Let’s plot these features against our target variable SalePrice and check the relation and outliers:

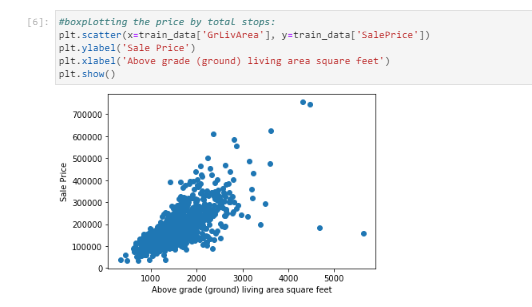




As we can see in the above boxplot the SalesPrice gets higher when the higher gets the rates of the overall material and finish of the house.

We see some outliers too in the above boxplot (black dots).

Then, we boxplot the Above grade (ground) living area square feet with regards our target SalePrice:

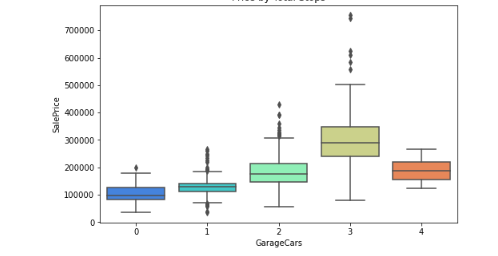


We see the relation is kind of linear and positive relation.

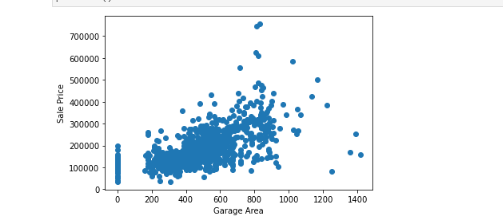
The more Above grade (ground) living area square feet we have, higher the Sale Price gets.

Regarding the outlier, we also have some outliers.

Next, we boxplot the SalePrice y diferents categories of GarageCars:

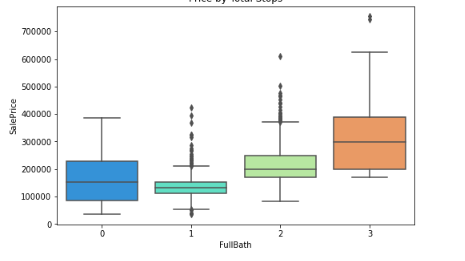


In the above boxplot we see that Garage of 3 cars is the highest rated. The surprising thing is that the 3cars Garage is more expensive than 4car garage which has a similar average Sale Price as the 2cars garage.



By Garage Area, we see the price is higher when the more garage area is acquired. We also see outliers outside the main segment/cluster and some outliers are far away from the main cluster.

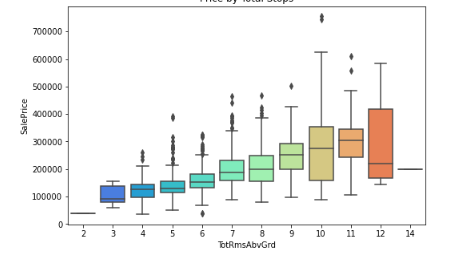
Let’s check now the sale price by full bath:



As we see in the boxplot above, we have higher dispersion of data when talking about the grade 0 and 3 of full bathrooms compared to others.

As we see the black dots in the boxplot, we have some outliers too.

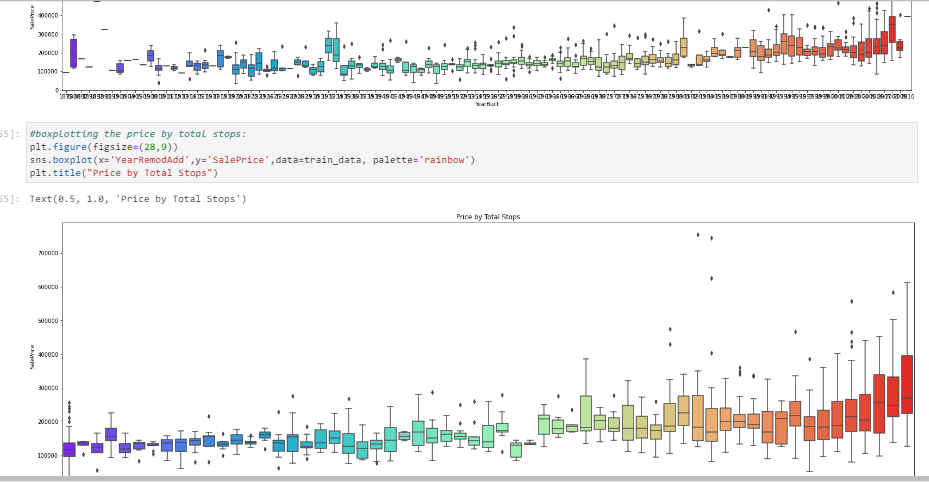
Now let’s check the SalePrice by Total Rooms above grade:



We see the price dispersion gets higher as the total rooms above grade has higher number of rooms.

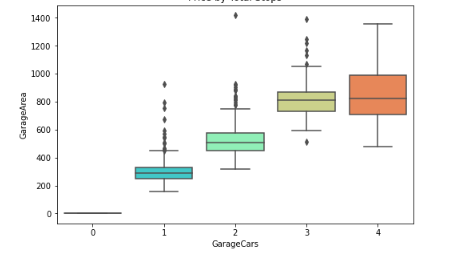
Here we also have outliers for rooms 4,5,6,7,8,9,10,11.

Now let’s check the Year Built and YearRemodAdd variable with respect our target SalesPrice:



Similar case happens here too, the higher the Year gets, the higher is the dispersion of the price in most of the cases.

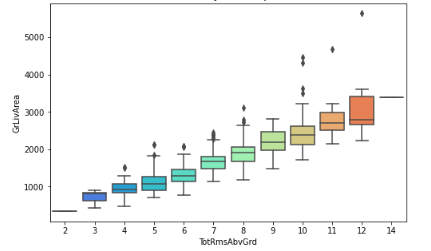
Let’s know analyse the boxplot of the Garage Area with respect Garage Cars:



Makes fully sense, the higher the GarageCars number is, the higher have to be the GarageArea. On the other way around, means that as more garage area we have, more cars can be parked in that space. We can also notice that 4 cars garage has more dispersion then the other ones.

Here we also have some outliers in black dots.

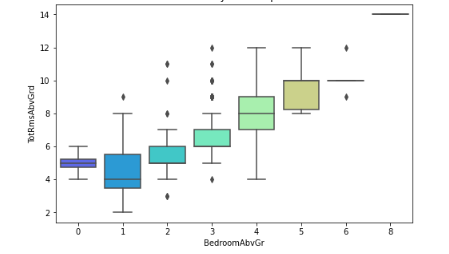
Let’s now boxplot the GrLivArea by TotRmsAbvGrd:



We easily see that the higher the total rooms above grade are, the higher it has to be the Ground Live Area.

In this case, we also have outliers in this feature.

Let’s check the boxplot for TotRmsAbvGrd by BedroomAbvGr:

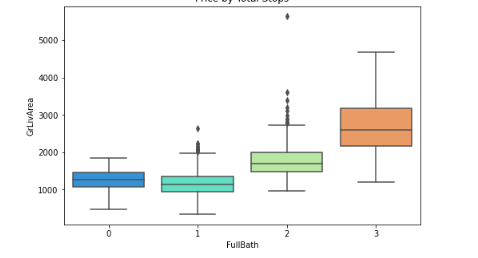


We can see the Total rooms above grade (does not include bathrooms) has high dispersion among the Bedrooms above grade atleast for 1 and 4 bedrooms.

And regarding ouliers, we have them one of them for 1 bedroom above grade and several for 2 or 3 bedrooms above the grade and a couple of outliers for 6 bedrooms above grade.

If we check 6 or 8 bedrooms above the grade, we can see no boxplot is plotted on those 2 categories which mean no data is there for these 2 missing boxplots, or at least either the data is 0 or there is no data.

Now let’s check the Grade Live Area and compare it by Full Baths:

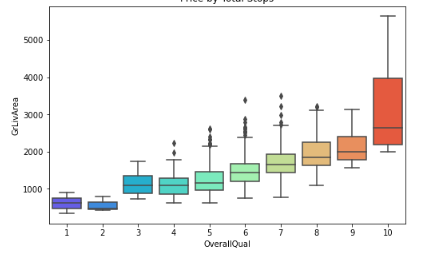


Make sense, as the higher counts of baths above the grade would use amount of Grade Live Area.

What we can see highlighted here is that the 3 full bathrooms above grade have more dispersion of data than the other ones.

Regarding outliers, we see many outliers for 1 full bathroom and 2 bathrooms above grade as I suppose these are the most constructed ones in the houses.

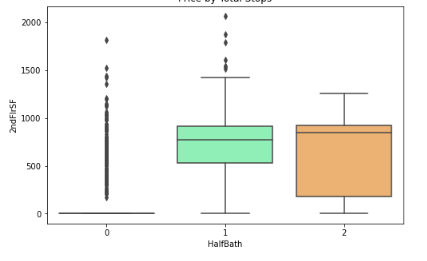
Now let’s check the GrLivArea with regards Overallqual



It shows what normally happens, the Ground Live Area has more definition and importance when we normally have good qualification rates for the overall material and finish of the house. The grade Live Area is even higher when the overall material rate and finish gets a rate of 10, an excellent rate for a house as of material and finish of the house.

Looking at the outliers, we have several for overall qualifications between 4 to 8.

Now let’s check the boxplot of 2ndFlrSF with regards Half Bath:

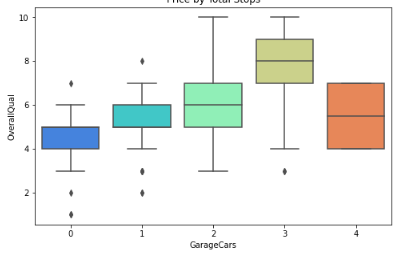


So here we see the Half Bath 1 above grade has less dispersion with regards the the 2nd Floor Square Feet than the Half Bath 2.

Interesting, I have never seen such details before. But, what I am guessing is that this data or conclusion of this plot is not as useful as other ones. I think I should have not included in my analysis but let’s keep it.

Maybe it is useful for our business user, never say no. But, from my personal perspective I would have excluded it from my analysis.

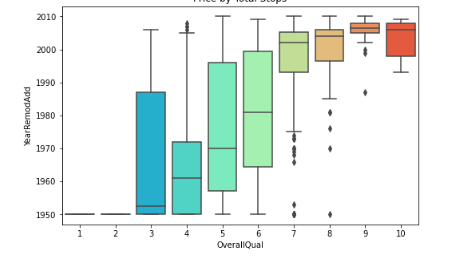
Now, let’s check the overall qualification/rate of the overall material with regards the number of cars in the garage:



We see, again, the cars garage seems to have higher overall rate for material and house finish when we have garage going from 0 to 2 cars. But as we previously commented, it is interesting to see that garage of 3 cars has higher rate for overall material and house finish.

We have outliers for 0, 1 and 3 cars garage.

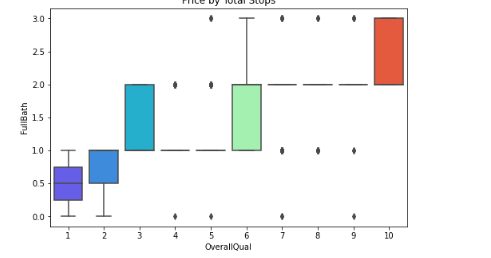
Let’s check now the YearRemodAdd by the OverallQual:



Here we can highlight that we have more dispersion over the years when we have overallqualification between 3 and 6. This is because over the years we have more houses being qualified between 3 and 6 rates as rate of overall material and house finish.

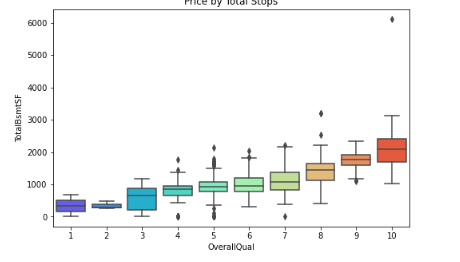
And with regards the outliers, we have several outliers for overall rate of 7, 8 or 9 when these rates are for years different than the normally included in the boxplot’box.

Let’s see the FullBath with regards OverallQual:



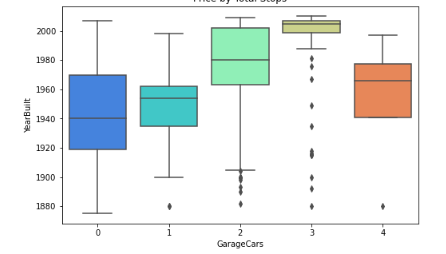
We see no such relation, but we can mention that the higher the qualification is, the higher is the dispersion of the data of full bath.

This is another case of relation that doesn’t get us any useful information for our analysis as there is no relation that can be studied and investigated. So, in my personal opinion, I would have deleted from my personal analysis.



Here what we see in the above boxplot is that the higher the Total Basement square Feet is, the higher is the overall material and house finish is. For me, it made sense when I thought about it in depth. The bigger the basement square is, the higher amount of good material and house finish it would have.

Now let’s check the Year Built by the Garage Cars variable:



As overview, we see almost all the garageCars are built after 1920.

Another conclusion we can make here is that 0 cars Garages have been built year between 1920 and 1970. And 1 car garages have been built between years 1940 and 1960.

And we can highlight that the 2 and 3 cars garage are the garage that were constructed recently. Being specific, 3 cars garage is the most recent category of garage. People have more money, hehe!

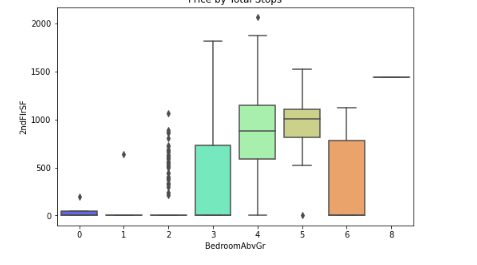
And finally, 4 cars garage were built from year 1940 and 1990, with one outlier/exception that one of these garage was built around year 1880.

Continuing commenting on the outliers, we see the 3 cars garage has more outliers than the other categories as all the data for it is displayed mostly in the few recent years basically around year 2000. And it has outliers in the previous years as the 3 cars garage would also have been built in some other previous years to year 2000.

And we see we have several outliers for 2 cars garage and 1 outlier for 1 car garage.

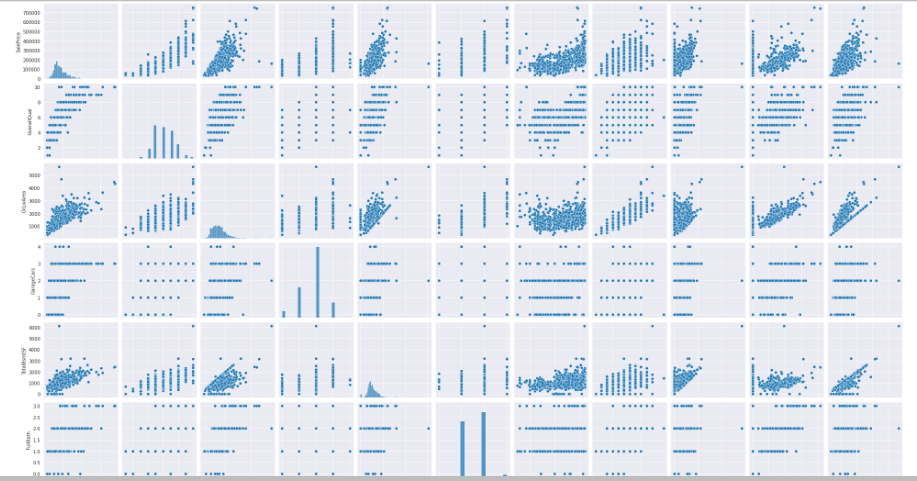
Let’s explain the last one so that we have an idea of what we have been commenting till now regarding outliers. So, if we take the boxplot of garage for 1 car we see that almost all the data is dispersed between years 1930 and 1960. But we have one outlier which means that we have one 1car garage that was built before 1930 and that was in year 1880.

Now let’s try to relate 2ndFlrSF with BedroomAbvGr and see if we find any considerable relation between them:



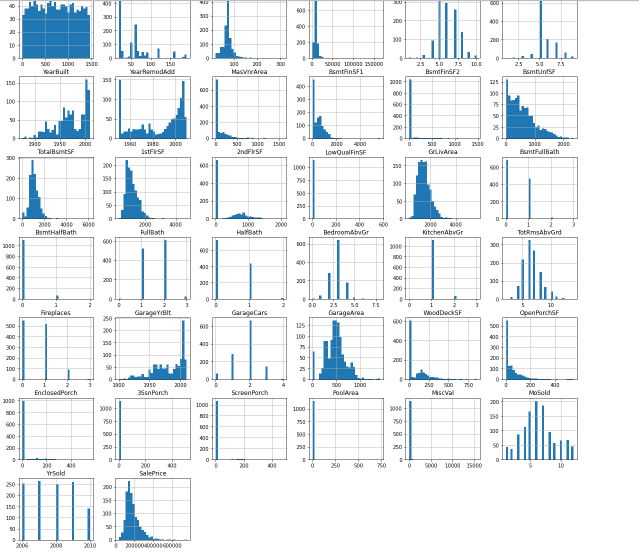
At first sight, we see that bedrooms above grade between 3 and 6 has higher probability to have some kind of relation with the second-floor square feet. Let’s see. So, checking the boxplot, we see that 3 and 6 bedrooms above grade have higher dispersion of data with regards the 2nd Floor square Feet than the 4 or 5 bedrooms above grade.

And we can notice the 5 bedrooms above grade has the smallest dispersion compared with the other categories when we are relating it to the 2nd Floor square feet.



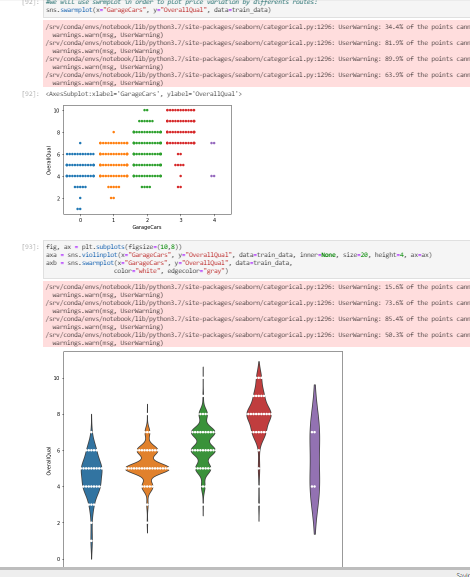
Looking at the above pairplot, we see some features have linear relationship between them and others not that linear.

And finally we did the following histograms in order to check the skewness of he data:

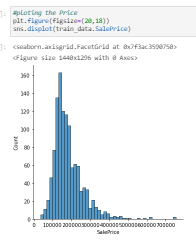


Here we see that almost all the features have some kind of skewness as these are not statically balanced on the medium. In some of the features we have skewness either on the right side or on the left side.

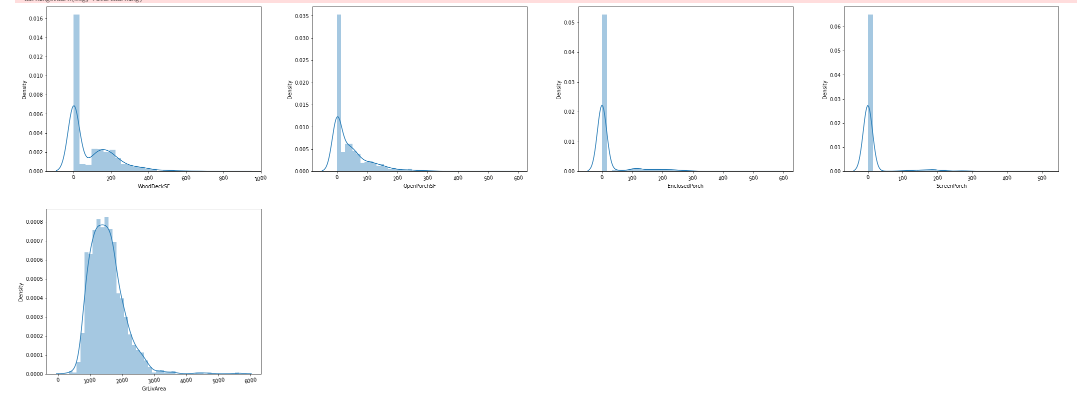
In the below swarmplots we see the same, the higher the size of garage is, the higher it normally has the rate of overall material and finish house:



And as we said, the garage for 3 cars has the highest rate of overall material and house finish. The interesting insight here is that even the 4 cars garage has less rate of overall materials and house finish than garage with 3 cars.



Checking the Sale Price from the above distplot, we see some kind of right skewness in it as well as in the below distplots:



* Interpretation of the Results

First of all, we managed with the missing values by imputing them either with mode or mean of the same feature.

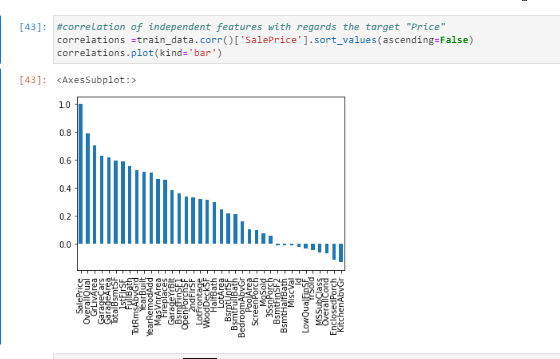
Then, we had skewness, multicollinearity and outliers that we had in all the columns. We looked and managed these 3 statistical measures through some power transformation yeo-johnson method, dropping irrelevant columns through VIF calculation and calculating ZScore for the 3 statistical measures respectively.

Then, after analysing several different visualizations, pre-processing and modelling, we got the following results:

In machine learning, it is a general practice to rely on a correlation matrix to decide what features to be incorporated into our models. We noticed we had several features such as OverrallQual, GrLivArea, GarageCars, GarageArea, TotalBsmtSF, 1stFlrSF, Fullbath, TotRmsAbvGrd, YearBuilt, YearRemodAdd among the features which are the features that have higher impact comparing all the available features.

We also saw SalesPrice gets higher when we have higher rate or amount of other features such as the rates of the overall material and finish of the house or garage Area among others.

And then, we had some features that impact the most on the calculation of the prediction of the SalePrice and that are the following ones:



These variables have higher impact on the target then the others when comparing the percentage of correlation. So, the higher correlation, more probability to have higher impact.

And in the modelling section, we studied and saw that RF and GBR where the best models when training or data set.

When comparing estimations of R2 score for the training set under LR, Lasso, SVM, RF, ABR and GBM algorithms, together with the performance metrics: MSE, RMSE and MAPE. Our discussion first starts with LR. The best CVscore score for a linear LR model with CV 8 is 0. .772932742 as the training marks 0.827822217 and the test got 0.83004367.

Then we compared it with Lasso which obtained R2 score 0.8308316643. We saw that mean of CV score is 0.774041026 which got my attention as it improved and got better than the scoring of LR which was 0.77293274. Checking the cross validation, we have checked that the CV score 0.7740410260144035 was obtained also at the CV 8.

Then, we checked and compared the obtained results with algorithm Random Forest Regressor which got us R2 Score of 88,3132% and a CV mean of 83,37%

Then we checked different algorithms like DT, KNN, SVR, Lasso, RFR,LR, AB.

Before deciding the best algorithm, we have gone through the several different steps such as: split data, train data through diferents algorithms and their corresponding different parameters.

The list of each algorithm used is as following:

* + LinearRegression
  + DecisionTreeRegressor
  + KNeighborsRegressor
  + SVR
  + Lasso
  + RFR
  + AdaBoostRegressor
  + GradientBoostingRegressor

From which we selected GradientBoostingRegressor which gave me the best results in comparison with the other algorithms after applying GridSearchCV.

Following this is the estimation of Gradient Boosting Regressor. Employing alpha=0.098, learning\_rate=0.1, max\_depth=2, min\_samples\_leaf=1, n\_estimators=100, the results are then evaluated by the MAE, MSE and RMSE criteria. These values are estimated to be:

R2 score is: 0.8248015242251906

R2 score for train data: 0.9467650917827348

Mean absolute error is: 19605.17013144875

Mean squared error is: 1219843622.670082

Root mean squared error is: 1219843622.670082

demonstrating that GBR is implying a reasonably good fit. In an iterative algorithm, the ‘optimal’ weight for each feature is indeed obtained by fine tuning an arbitrary value on a function, and going down its slope step by step until it reaches the lowest point of the function. Hence, our focus should concentrate on the explanatory power R2R2 and the error minimisation of the model.

Using random forest, our base model provides a R2R2 that is as high as 0.774119309

for the test set while the MSE, RMSE and MAPE are estimated to be:

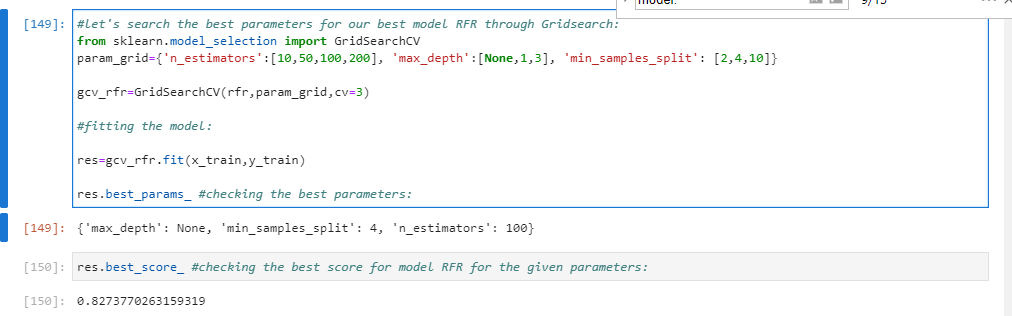
Mean absolute error is: 22486.83287671233

Mean squared error is: 1572725556.832603

Root mean squared error is: 1572725556.832603

In random forest, the estimation of SGD is not feasible because some of the hyperparameters are not continuous, such as the number of estimators. Rather, we can employ the grid search to minimise the model errors by tuning the hyperparameters.

For hyperparameter tuning, we perform many iterations of the entire fivefold cross-validation process, each of which sets different model settings, such as the number of estimators, min samples split, min sample leaf, max features, and max depth. Then, we compare all these models, pick up the best one, train it on the training set, and then evaluate on the test set as follows:



Hence, we take the result based on grid search as the best result associated with random forest.

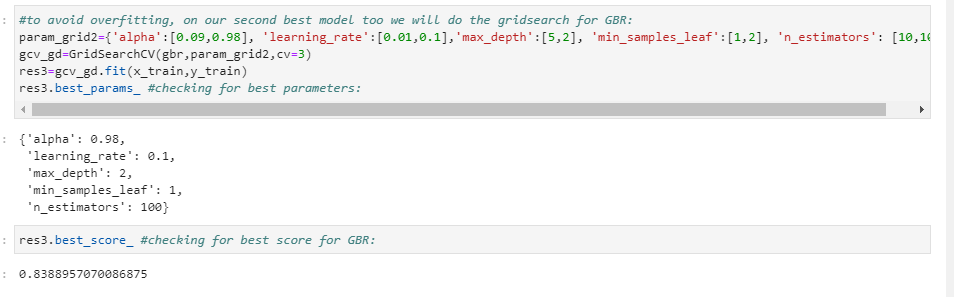
Lastly, we discuss the results based on the gradient boosting regression. In the base model, GBM provides a R2R2 of 0.8248015242251906 for the test set while the MSE, RMSE and MAPE are estimated to be:

Mean absolute error is: 19605.17013144875

Mean squared error is: 1219843622.670082

Root mean squared error is: 1219843622.670082

It is very obvious that GBM fits our data reasonably well. In order to obtain even better results, we then employ the grid search with 5 CV to minimise the model errors by tuning the hyperparameters, including subsample, learning rates, the number of estimators, min samples split, min sample leaf, max features, and max depth:



We then obtain a R2R2 as high as 0.9218462803998849 for the test set .The results based on grid search are the best results associated with gradient boosting regression.

For most of the time, GBM also fits the data very well for example in this case. However, the model does not perform well with high bias for some extreme values that are lying far away from the cluster. Again, the model has achieved very good model fitting since most of the predicted values are lying close to the predicted values and its performance is slightly better than those of RF in relation to R2\_Score.

Finally, we compare the results of these estimation techniques. In terms of explanatory power, this study shows that RF and GBM have equally good performances. In terms of error minimisation, GBM is slightly better than RF with respect to the three performance error metrics and R2 score while it outperforms SVM. Hence, we demonstrate that GBM are very powerful tools for making accurate prediction of property prices, since the performances of this algorithm is higher to the other. In comparison, the performance of SVM is found to be below that of RF and GBM in all aspects. However, this does not imply that SVM is necessarily inferior.

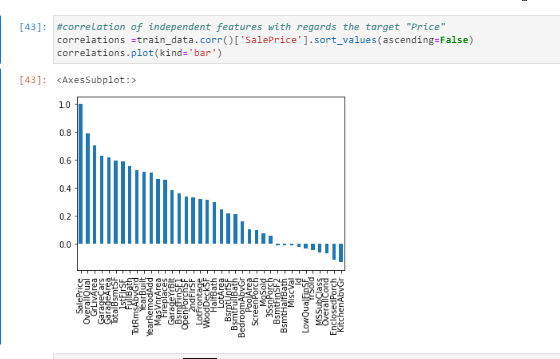
**CONCLUSION**

* Key Findings and Conclusions of the Study

If we concentrate on the main 3 questions, then the results answer the research questions as follows:

* Question 1 –Which variables are important to predict the price of variable?

If we first concentrate on the target and check the best correlations against it, then we will see that the following features are considered the best correlated features within our dataset:



Answer: We see several features such as OverrallQual, GrLivArea, GarageCars, GarageArea, TotalBsmtSF, 1stFlrSF, Fullbath, TotRmsAbvGrd, YearBuilt, YearRemodAdd among the features shown in the above screenshot are the features that have higher impact comparing all the available features.

* Question 2 – Which machine learning algorithm performs better and has the most accurate result in house price prediction? And why?

In general, the selection of a ML trained and tested model for a case like house price prediction-estimation is very much influenced by the quality and availability of data. When we have wide range of data, the model can be trained easily and will be effective as sufficiently long time series of house prices and relevant fundamental variables allow error-correction or vector error-correction models to be estimated.

So basically, here the study shows a comparison the regression algorithms work when predicting house prices. The results were promising for the data due to it being rich with features and having strong correlation.

However, RF gave the improved the results score compared to the results previously obtained, but GBR got the best R2 score and results overall. The final results of this study showed that GBR makes better prediction compared to other used algorithms.

Summarizing, the results answer the research questions as follows:

- Question 1 – Which machine learning algorithm performs better and has the most accurate result in house price prediction? And why? GBR made the best performance overall when both R2 and RMSE scores are taking into consideration.

- Question 2 –Which variables are important to predict the price of variable? We see several features such as OverrallQual, GrLivArea, GarageCars, GarageArea, TotalBsmtSF, 1stFlrSF, Fullbath, TotRmsAbvGrd, YearBuilt, YearRemodAdd among the features shown in the above screenshot are the features that have higher impact comparing all the available features.

- Question 3 - How do these variables describe the price of the house?

These variables have higher impact on the target then the others when comparing the percentage of correlation. So, the higher correlation, more probability to have higher impact on the Sale Price of the house.

* Learning Outcomes of the Study in respect of Data Science

As we may know improvement in computer technology has made it possible data that cannot previously be captured, processed and analysed such as property research. This study is an attempt to use machine learning algorithms in estimating housing prices, and then compare their results.

In this study, our models are trained with housing property data using random forest (RF) and gradient boosting machine (GBM) among others. We have demonstrated that advanced machine learning algorithms can achieve very accurate prediction of property prices, as evaluated by the performance metrics. Given our dataset used for this study, our main conclusion is that RF and GBM are able to generate comparably accurate price estimations with lower prediction errors, compared with the SVM results.

The existing literature has demonstrated mixed results about which algorithm is superior in making predictions. Previous studies suggest that RF has a better performance than GBR but other studies demonstrate that both models have comparable predictive power and almost equally applicable in making predictions (Ahmad et al., [2017](https://www.tandfonline.com/doi/full/10.1080/09599916.2020.1832558)). In another study about the forecast of daily lake water levels, RF is found to exhibit the best results when compared to linear regression, SVM and ANN with respect to R2R2 and RMSE (Li et al., [2016](https://www.tandfonline.com/doi/full/10.1080/09599916.2020.1832558)). Utilising RF, stochastic gradient boosting and SVM to predict genomic breeding values, Ogutu et al. ([2011](https://www.tandfonline.com/doi/full/10.1080/09599916.2020.1832558)) conclude that boosting has had the best performance, followed by SVM and RF. In this study, SVM performs better than RF (see also Caruana & Niculescu-Mizil, [2006](https://www.tandfonline.com/doi/full/10.1080/09599916.2020.1832558)).

I think we should suggest that the choice of algorithm depends on several factors including, the size of data set, computing power, time constraint and researcher’s knowledge about machine learning. If the accuracy of prediction is the priority, we recommend utilising RF or GBM, although their computations often take much longer times than SVM. Needless to say, it is a good practice to use more than one algorithm to compare the predictions.

* Limitations of this work and Scope for Future Work

To conclude, the application of machine learning in property research is still at an early stage. We hope this study can provide some methodological improvements and contributions to property calculation and presenting an alternative to the valuation of housing prices. Future direction of research may consider incorporating additional property transaction data including more features.

Even though, we are very happy with our final model’s performance, but here we can improve the results by working more intensively on the model and performance results. The inclusion of GBR for continuous variables improved our model’s performance. That being said, there are still additional features and methods we would like to try that we think could further improve the model’s performance, as we already mentioned.

Future work on this study could be divided into main areas to improve the result even further. Which can be done by:

- The used pre-processing methods do help in the prediction accuracy. So, experimenting with different combinations of pre-processing methods to achieve better prediction accuracy.

- Make use of the available features and if they could be combined as binning features that can improve the data and hence the performance of the model and their results.

- Training the datasets with different regression methods such as Elastic net regression that combines both L1 and L2 norms. In order to expand the comparison and check the performance.

- Some of the factors that have been studied and analised in this study has a weak correlation with the sale price. Hence, by adding more factors to the dataset that affect the house price, such as GDP, average income, and the population in order to increase the number of factors that have an impact on house prices. This could also lead to a better finding for question 1 and 2, hence also 3.

. - The results of this study have shown that GBR is prone to overfitting. However, GBR still a strong algorithm that has a lot of options that could, with the right methods, provide a better prediction accuracy.

For future work, we recommend that working on large dataset would yield a better and real picture about the model training and testing results. We have undertaken only few Machine Learning algorithms that are regressors, but we need to train many other regressors and understand their predicting behaviour for continuous values. By improving the error values this work can be useful for development of applications for various respective house Sale pricing.