greatlearning
Power Ahead

House Prize Prediction

CAPSTONE PRESENTATION

PRABHJOT KAUR CHAHAL (PGP-DSBA -Online Mar_C 2021)

WORK FLOW

1. Problem Understanding
2. Data Report
3. Exploratory Data Analysis
4. Business insights from EDA
5. Model building and interpretation
6. Model Tuning
7. Interpretation of the most optimum model and its implication on the business

Business Problem Understanding

Business problem we are trying to solve

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 may expect — it can't be too low or too high. To find house price you usually try to find similar properties in your neighborhood and ***based on gathered data we will try to assess your house price***.

1. cid: a notation for a house
2. dayhours: Date house was sold
3. price: Price is prediction target
4. room_bed: Number of Bedrooms/House
5. room_bath: Number of bathrooms/bedrooms
6. living_measure: square footage of the home
7. lot_measure: quare footage of the lot
8. ceil: Total floors (levels) in house
9. coast: House which has a view to a waterfront
10. sight: Has been viewed
11. condition: How good the condition is (Overall)
12. quality: grade given to the housing unit, based on grading system
13. ceil_measure: square footage of house apart from basement
14. basement_measure: square footage of the basement
15. yr_built: Built Year
16. yr_renovated: Year when house was renovated
17. zipcode: zip
18. lat: Latitude coordinate
19. long: Longitude coordinate
20. living_measure15: Living room area in 2015(implies-- some renovations) This might or might not have affected the lotsize area
21. lot_measure15: lotSize area in 2015(implies-- some renovations)
22. furnished: Based on the quality of room
23. total_area: Measure of both living and lot

SCOPE:

•The prediction outcomes can **help various real estate stakeholders to make more informed decisions**.

How can we get the profitable pricing for the houses and buildings so that neither the seller nor the buyer is at a loss?

•That is where the factors affecting the price of the house come into picture.

•If a **fair evaluation of all the factors**, how they contribute, why they contribute is made, then a profitable figure can be derived which leads to a **win-win situation for both the parties**.

OBJECTIVES:

•***Take advantage of all of the feature variables*** available, use it to analyze and predict house prices.

•Among the available attributes, we ***try to identify the most valuable attributes highly affecting the price fluctuations***

•What is the ***extent of impact of these factors on the price*** of the house?

•How to ***derive a best deal*** for a house based on these factors?

DATA REPORT AND BUSINESS INSIGHT OF EDA

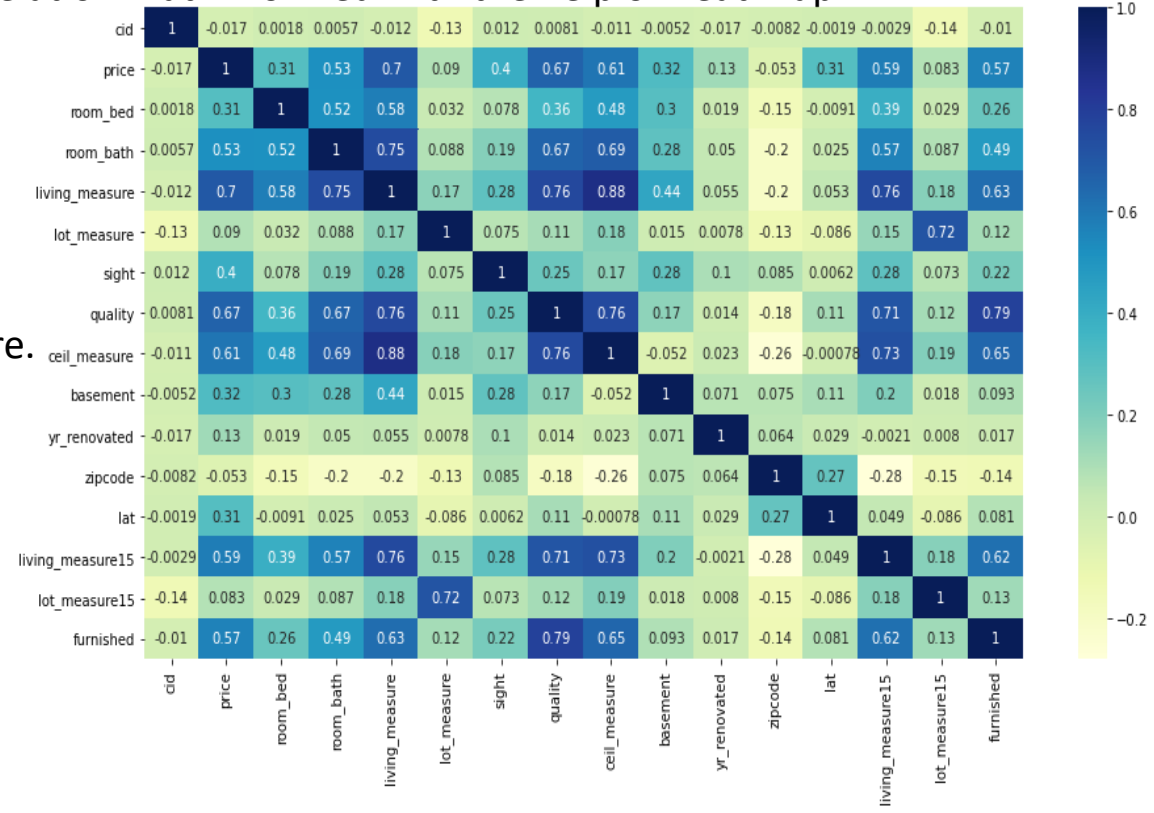
- There are **21613 rows and 23 columns** ,data type is either **object, or float64 or int64**
- **Price is the target** variable, ranging from 75,000 to 77,00,000 and distribution is right-skewed. The mean price of the houses tend to be **high during March, April, May** as compared to that of September, October, November,December period.
- **Most columns distribution is Right-Skewed** and only few features are Left-Skewed (like room_bath, yr_built, lat).
- There was **no error or duplication** the recording of the data.
- There are **max missing null values of 166 count in the living_measure15** followed by bedrooms and bathrooms.
- We have **176 properties that were sold more than once** in the given data
- So the **time line** of the sale data of the properties is from **May-2014 to May-2015 and April** month have the highest mean price
- Most of the houses/properties have **3 or 4 bedrooms**
- Majority of the properties have **bathroom in the range of 1.0 to 2.5**
- Most houses have **1 floor** , coast - most houses **don't have waterfront view**, very few are waterfront ,sight - most **sights have not been viewed** ,condition - Overall most houses are rated as **3 and above for its condition** overall
- Quality - most properties have **quality rating between 6 to 10**
- We have almost **60% of the properties without basement** .Houses have zero measure of basement i.e. they don't have basement
- Only **914 houses were renovated** out of 21613 houses
- Most properties are **not furnished**. Furnish column need to be converted into categorical column

DATA REPORT AND BUSINESS INSIGHT OF EDA

- Properties with higher price have more **no.of sights** compared to that of houses with lower price
- The price of the house increases with **condition** rating of the house
- Smaller houses are in better condition and better condition houses are having higher prices
- There is clear increase in price of the house with higher rating on **quality**
- There is upward trend in price with **ceil_measure**
- In terms of Number of floors, most of the houses are of **single (1) & double (2) floor**
- In terms of Sight Viewed, almost 90% of the time sight is not viewed. And maximum number of sight viewed is 4 times

We have linear relationships in below features, as we got to know from correlation matrix formed with the help of heat map:

1. **price:** room_bath, living_measure, quality, living_measure15, furnished
2. **living_measure:** price, room_bath. So we can consider dropping 'room_bath' variable.
3. **quality:** price, room_bath, living_measure
4. **ceil_measure:** price, room_bath, living_measure, quality
5. **living_measure15:** price, living_measure, quality. So we can consider dropping living_measure15 as well. As it's giving same info as living_measure.
6. **lot_measure15:** lot_measure. Therefore, we can consider dropping lot_measure15, as it's giving same info.
7. **furnished:** quality
8. **total_area:** lot_measure, lot_measure15. Therefore, we can consider dropping total_area feature as well. As it's giving same info as lot_measure.



DATA REPORT AND BUSINESS INSIGHT OF EDA

•**Living measure is the most essential variable** that impacts the price of the house. Hence, while buying or selling or even while valuating a house, it's essential to see that how much square foot of area is covered under living measure of the house.
•**Lot measure is not a useful variable** to determine the price of the house. It can be minutely noted during the valuation of any property.
•**Focus more on houses which are centrally located in the area**. That's a lucrative place to sell the house. Not much of profit can be expected from houses on the coastlines.
•**A house with 3 bedrooms and 3 bathrooms** will be the most lucrative offer to any prospective buyer.

DATA CLEANING AND PREPROCESSING OR FEATURE ENGINEERING:

•**Missing values** were treated with the help of SimpleImputer and median was taken as strategy.
•We have seen **outliers for columns room_bath(33 bed), living_measure, lot_measure, ceil_measure and Basement and going to treat outliers and drop it for these columns only.**
In summary, after treating outliers, we have lost about 15% of the data. We will analyse the impact of this data loss during the model evaluation.

•As we already have this information in other features. We **will drop the unwanted columns** from new copied dataframe instance : cid,dayhours,yr_renovated,zipcode,lat,long

•**Creating dummies for categorical variables**: 'room_bed', 'room_bath', 'ceil', 'coast', 'sight', 'condition', 'quality', 'furnished', 'has_basement', 'has_renovated‘ as a part of feature engineering.

Modelling Approach Used & Why

Before building the model we need to perform and understand few aspects:

1) **Build a new dataframe** for building models with **engineered features**.
2) We **will split our dataset into a training set and testing set** using sklearn train_test_split() in the ratio of 80:20.
3) Price prediction is **a supervised - regression problem** which is going to use different machine learning algorithms.
4) We will train numerous regression models on the train data (e.g., simple linear regression, lasso, ridge, KNN, Support vector regressor-SVR, Decision tree regressor-DT) and **evaluate their performance using R-square,Root Mean Squared Error (RMSE), MSE, MAE on the test data**

Different regression models used here and why are as follow:

1) Linear Regression – is one of the most common model for regression problems.
2) Ridge ,lasso regression-There are two popular regularization techniques, each of them aiming at **decreasing the size of the coefficients by penalizing coefficients**
3) K-Neural Network – can be **used for both classifier and regression** problem.The k-NN algorithm is **used for estimating continuous variables.**
4) Support vector regressor- it is use to predict discrete values and is use to **find the best fit line with maximum points**.
5) Decision Tree Regressor-Is one of the most commonly used, **practical approaches for supervised learning**. It can be used to solve both Regression and Classification

For model tuning we have used ensemble models like:

- Boosting and Bagging: It's another type machine learning model which works on intuition that best possible next model, **when combined with previous models, minimizes the overall prediction error**.
- Random forest: supervised learning technique used for regression problem, it's another form of regression

Modelling Approach Used & Why

	Method	Val Score	RMSE_vl	MSE_vl	MAE_vl	train Score	RMSE_tr	MSE_tr	MAE_tr
0	Linear Reg Model1	0.622318	159467.070404	2.542975e+10	113309.839017	0.623767	155370.510001	2.414000e+10	111870.452000
0	Linear-Reg Lasso1	0.624855	158930.629167	2.525894e+10	113247.375142	0.623681	155388.292306	2.414552e+10	111894.555654
0	Linear-Reg Ridge1	0.625127	158873.013444	2.524063e+10	113297.137425	0.623263	155474.675112	2.417237e+10	111992.775261
0	knn1	0.393344	202106.119760	4.084688e+10	138782.844505	0.998802	8768.399424	7.688483e+07	782.211363
0	SVR1	-0.041009	264749.597492	7.009235e+10	178245.041225	-0.050705	259645.350463	6.741571e+10	179598.539492
0	SVR2	0.451004	192261.623225	3.696453e+10	131552.868372	0.447034	188360.333364	3.547962e+10	131617.700073
0	DT1	0.400568	200899.070073	4.036044e+10	136862.640929	0.998802	8768.399424	7.688483e+07	782.211363
0	DT2	0.559155	172286.379989	2.968260e+10	116846.303028	0.749671	126734.766320	1.606170e+10	93987.746486
0	GB1	0.686828	145210.926188	2.108621e+10	105167.620490	0.732043	131121.047469	1.719273e+10	98049.360527
0	BGG1	0.666885	149763.125718	2.242899e+10	102657.068038	0.951959	55519.380196	3.082402e+09	38604.116644
0	RF1	0.671261	148776.257062	2.213437e+10	101633.946497	0.953149	54827.919341	3.006101e+09	38336.130007

After this will build function or pipeline for the models

	Method	val score	RMSE_val	MSE_val	MSE_vl	train Score	RMSE_tr	MSE_tr	MAE_tr
0	LR	0.622318	159467.070404	2.542975e+10	2.542975e+10	0.623767	155370.510001	2.414000e+10	111870.452000
0	KNNR	0.393344	202106.119760	4.084688e+10	4.084688e+10	0.998802	8768.399424	7.688483e+07	782.211363
0	DTR	0.379308	204430.687410	4.179191e+10	4.179191e+10	0.998802	8768.399424	7.688483e+07	782.211363
0	GBR	0.686828	145210.926188	2.108621e+10	2.108621e+10	0.732043	131121.047469	1.719273e+10	98049.360527
0	BGR	0.666885	149763.125718	2.242899e+10	2.242899e+10	0.951959	55519.380196	3.082402e+09	38604.116644
0	RFR	0.672435	148510.410306	2.205534e+10	2.205534e+10	0.953626	54547.787830	2.975461e+09	38241.189832

Later will perform Feature selection (PCA)

• 52 dimensions covering 97% variance in the data.

• Will recall the ensemble models from our initial run to check the feature selection using featureimp from individual models:

Gradient boosting, Random forest

•From **two models we have 38 importance features**.

We will freeze on the 38 list and make another dataframe (along with 'price')

```
38
['coast_1.0', 'condition_4.0', 'condition_3.0', 'ceil_3.0', 'condition_5.0', 'room_bath_2.5', 'yr_built', 'room_bath_2.25', 'Ho
useLandRatio', 'room_bath_1.75', 'room_bed_2.0', 'quality_10.0', 'living_measure15', 'quality_12.0', 'room_bed_4.0', 'basemen
t', 'sight_4.0', 'ceil_measure', 'living_measure', 'ceil_2.0', 'quality_9.0', 'room_bed_5.0', 'room_bath_3.25', 'quality_7.0',
'room_bed_3.0', 'room_bath_3.0', 'total_area', 'room_bath_3.75', 'quality_8.0', 'lot_measure15', 'furnished_1.0', 'has_renovate
d_Yes', 'quality_11.0', 'lot_measure', 'room_bath_3.5', 'quality_6.0', 'quality_5.0', 'sight_2.0']
```

HYPERTUNING with Gridsearch CV:

Since we have better performance in gradient boosting model, we will hypertune the model for improving the score

- The perforemed iteration gives best result of 0.6759.

Final parameters that are giving best result on training set are:

'learning_rate': 0.1,
'min_samples_leaf': 20,
'min_samples_split': 5,
'n_estimators': 500},
0.6759087726904568)

Insights from Analysis

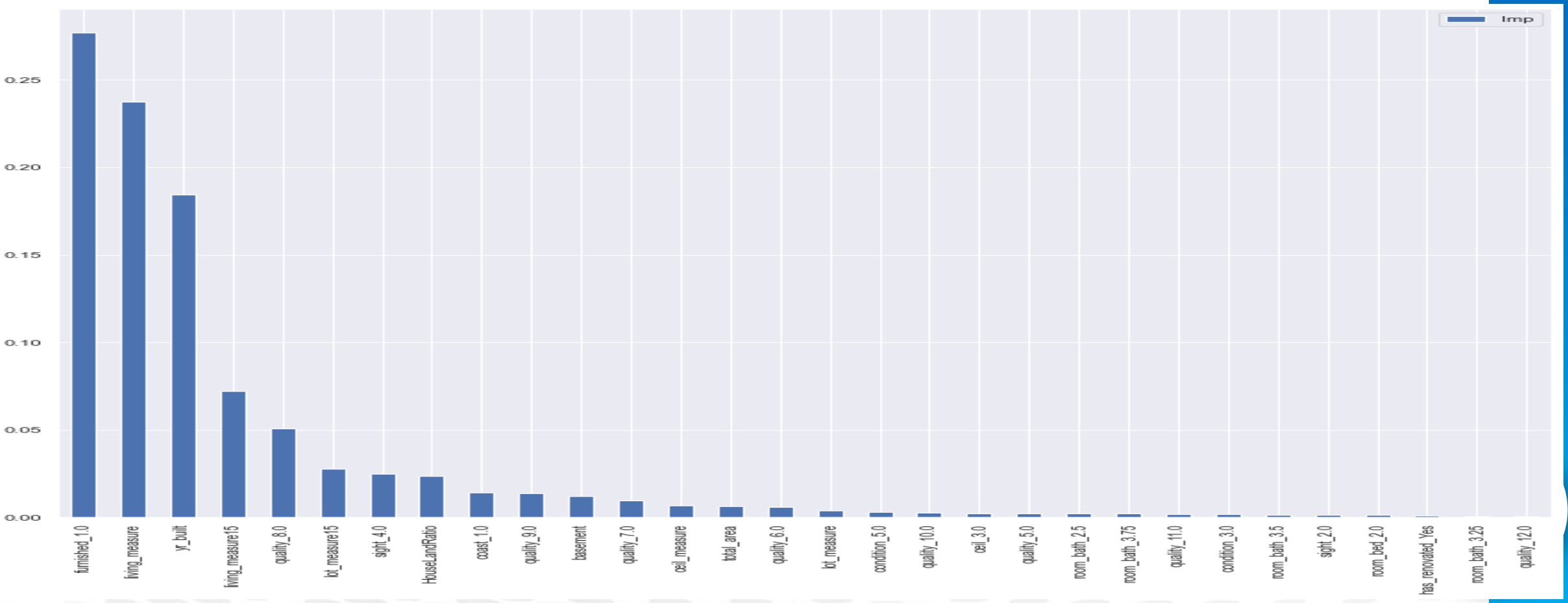
- ***Linear models*** have performed almost with similar results in both regularized model and non-regularized models
- ***KNN regressor*** performed well in training set, the performance score in validation set is very less. This shows that the model is *overfitted in training set*.
- The above negative scores in ***SVR1*** model is due to *non-learning of the model* in the training set which results in non-performance in validation set.
- The ***SVR2*** model with modified parameters *has not performed* well with just ~0.45 in both training and validation data sets
- Performance of initial ***Decision tree- DT1*** model shows *over fit in training set with 0.99 score and low performance in validation set*
- ***Decision tree model*** with modified parameter ***DT2*** has *better performed on the training set and validation set compared to initial decision tree model*. But overall decision tree has not performed well than linear regression models.
- Shows heteroscedasticity (“different scatter”).
- ***Gradient boosting model*** has provided *good scores in both training and validation sets*
- ***Bagging model*** also *performed well in training and validation sets*.There seems to be *overfitting in training set. We need to analyse further by hypertuning*
- ***Random forest model*** has performed *well in training and validation set*. There is scope of further analysis on this model

Ensemble methods are performing better than linear models.

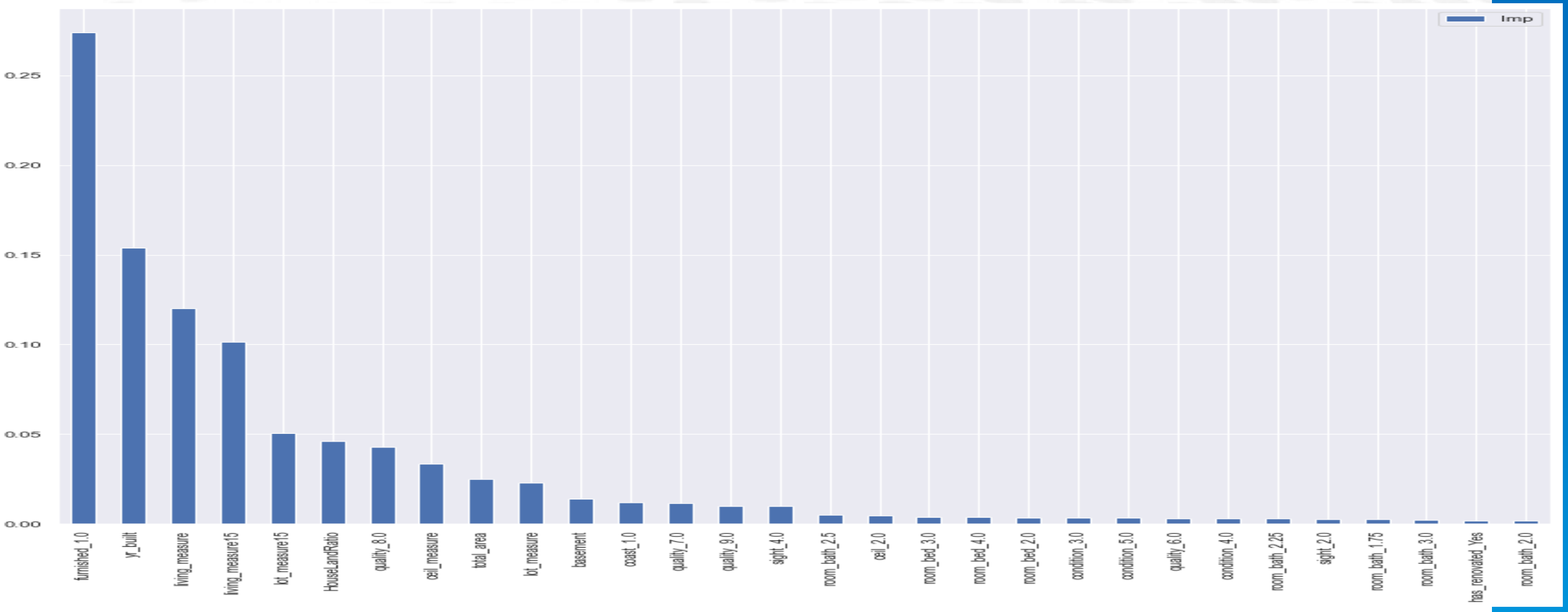
Of all the ensemble models, Gradient boosting regressor is giving better R2 score of **0.73.**

We identified top 30 features that are explaining the 95% variation in model(Random Forest). Will further hypertune the model to improve the model performance.

The top 30 features are covering about 99% in gradient boosting model via PCA.



The top 30 features are covering about 99% in gradient boosting model via PCA.



From two models we have 38 importance features.

```
38
['coast_1.0', 'condition_4.0', 'condition_3.0', 'ceil_3.0', 'condition_5.0', 'room_bath_2.5', 'yr_built', 'room_bath_2.25', 'Ho
useLandRatio', 'room_bath_1.75', 'room_bed_2.0', 'quality_10.0', 'living_measure15', 'quality_12.0', 'room_bed_4.0', 'basemen
t', 'sight_4.0', 'ceil_measure', 'living_measure', 'ceil_2.0', 'quality_9.0', 'room_bed_5.0', 'room_bath_3.25', 'quality_7.0',
'room_bed_3.0', 'room_bath_3.0', 'total_area', 'room_bath_3.75', 'quality_8.0', 'lot_measure15', 'furnished_1.0', 'has_renovate
d_Yes', 'quality_11.0', 'lot_measure', 'room_bath_3.5', 'quality_6.0', 'quality_5.0', 'sight_2.0']
```

After hypertunning with GridSearchCV and finding the best parameters and making model out of it:
The best performance is given by Gradient boosting model with training (score-0.82,RMSE-105884), Validation (score-0.68.1,RSME-146570), Testing(score-0.677,RMSE-148559).

	Method	Val Score	RMSE_vl	MSE_vl	MAE_vl	train Score	RMSE_tr	MSE_tr	MAE_tr	test Score	RMSE_ts	MSE_ts	MAE_ts
0	GBRF	0.680939	146570.010085	2.148277e+10	103007.241841	0.825264	105884.156635	1.121145e+10	74530.908765	0.67721	148559.912908	2.207005e+10	103202.189988

- **Lowest RMSE**
- **Highest R-square**
- **Best computation speed**
- **Realistic Feature Importance**

The top key features that drive the price of the property are: 'furnished_1', 'yr_built', 'living_measure','quality_8', 'HouseLandRatio', 'lot_measure15', 'quality_9', 'ceil_measure', 'total_area

Recommendations

The company can create a mobile application for the salesmen who are on site. The salesman can enter input for 38 variables and the output for the same will be the price of the house.

We recommend that the company deploys **Gradient boosting model with training** (score-0.82,RMSE-105884), Validation (score-0.68.1,RSME-146570), Testing(score-0.677,RMSE-148559

This model can also be used by housing finance companies to evaluate the house prices.

It is strongly insisted that the user enters all the input most accurately to his/her knowledge to get the most accurate results. Any input entered incorrectly or left blank would result in incorrect prediction by the model.

Factors like GDP, average income and the population can have impact on house prize and lead to better findings.

It is suggested that the salesmen quote a price 10% higher than the model prediction so that there is room for negotiation

Thank you