# House Price Prediction with Regression

This project aims to predict house prices using two machine learning techniques: a simple algorithm, Linear Regression, and a more sophisticated one, Random Forest. Additionally, we will apply regularized regression methods like Ridge and Lasso to enhance the accuracy of our predictions.

The Random Forest model proved to be the most effective for predicting house prices, outperforming the regression algorithms with an accuracy of 85% based on the R-squared metric. The most significant predictor was the overall quality of the house, followed by the size of the above-ground living area and the total basement square footage.

This project serves as an initial attempt to quickly develop a reasonably good model prototype.

| | Id | MSSubClass | MSZoning | LotFrontage | LotArea | Street | Alley | LotShape | LandContour | Utilities | ... | PoolArea | PoolQC | Fence | MiscFeature |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 60 | RL | 65.0 | 8450 | Pave | NaN | Reg | Lvl | AllPub | ... | 0 | NaN | NaN | NaN |
| **1** | 2 | 20 | RL | 80.0 | 9600 | Pave | NaN | Reg | Lvl | AllPub | ... | 0 | NaN | NaN | NaN |
| **2** | 3 | 60 | RL | 68.0 | 11250 | Pave | NaN | IR1 | Lvl | AllPub | ... | 0 | NaN | NaN | NaN |
| **3** | 4 | 70 | RL | 60.0 | 9550 | Pave | NaN | IR1 | Lvl | AllPub | ... | 0 | NaN | NaN | NaN |
| **4** | 5 | 60 | RL | 84.0 | 14260 | Pave | NaN | IR1 | Lvl | AllPub | ... | 0 | NaN | NaN | NaN |

5 rows × 81 columns

We can see some features are numeric while others are text. There are also missing values in the dataset.

[6]:

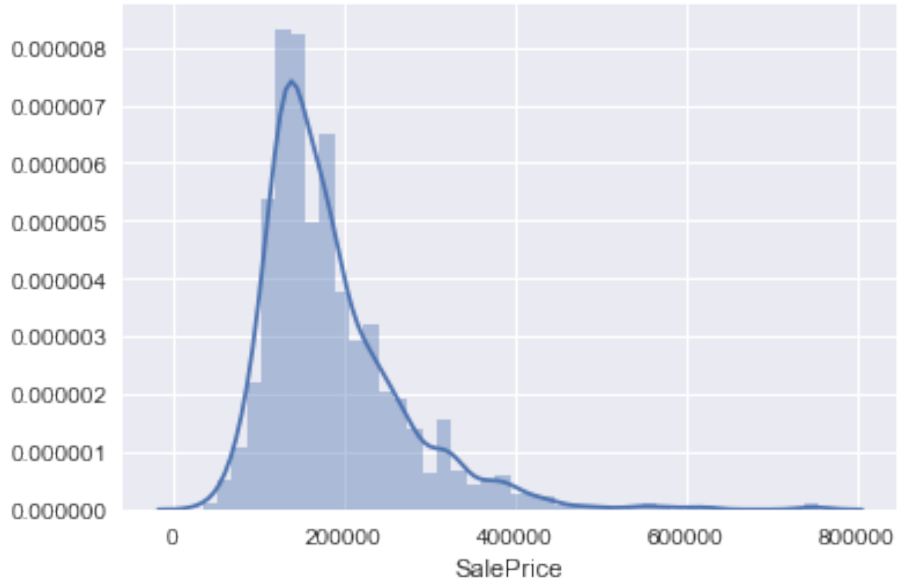
| | MissvalCount | Percent |
|---|---|---|
| **PoolQC** | 1453 | 99.52 |
| **MiscFeature** | 1406 | 96.30 |
| **Alley** | 1369 | 93.77 |
| **Fence** | 1179 | 80.75 |
| **MasVnrType** | 872 | 59.73 |
| **FireplaceQu** | 690 | 47.26 |
| **LotFrontage** | 259 | 17.74 |
| **GarageYrBlt** | 81 | 5.55 |
| **GarageCond** | 81 | 5.55 |
| **GarageType** | 81 | 5.55 |
| **GarageFinish** | 81 | 5.55 |
| **GarageQual** | 81 | 5.55 |
| **BsmtFinType2** | 38 | 2.60 |
| **BsmtExposure** | 38 | 2.60 |
| **BsmtQual** | 37 | 2.53 |
| **BsmtCond** | 37 | 2.53 |
| **BsmtFinType1** | 37 | 2.53 |
| **MasVnrArea** | 8 | 0.55 |
| **Electrical** | 1 | 0.07 |

## Literature review:

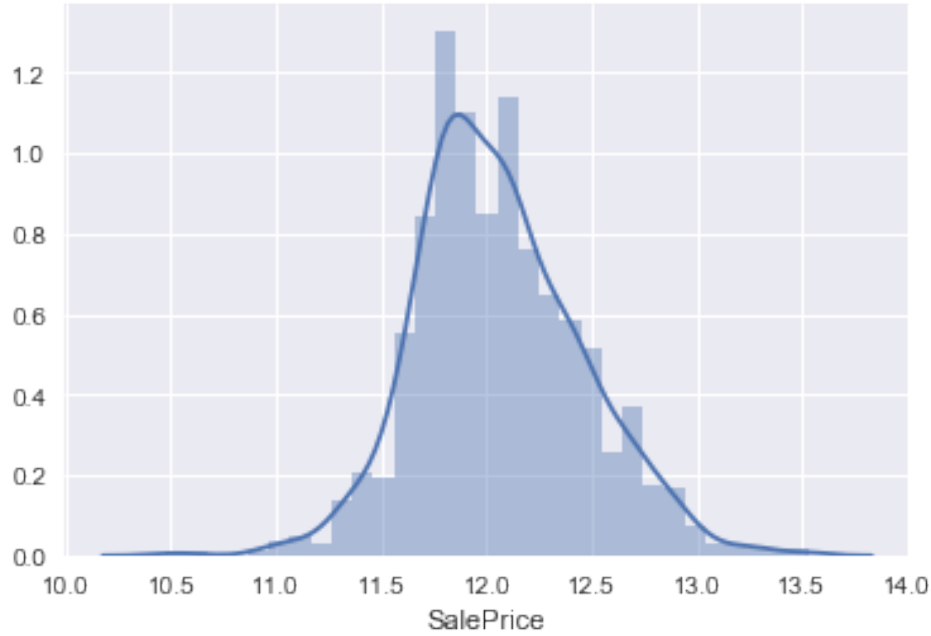
Manasa and Gupta [1] have taken Bengaluru as a city for the case study. The property size in square feet, location, and facilities are all key aspects affecting cost. 9 different attributes are used. Multiple linear regression (Least Squares), Lasso/Ridge regression, SVM, and XG Boost are used for experimental work. In [2], Luo suggests that to explain the factors that determine residential asset prices, most studies have concentrated on macroeconomic aspects. It looks at some micro characteristics, such as lot size and pool size, that can be utilised as features to estimate house prices in this research. Random forest and support vector machine are two machine learning methods which are used to predict asset pricing. R-squared is more than 0.9 in all regression models. Panjali and Vani [3] state that forecasting the resale price of a house in the long term is vital, especially for those who will be residing there for a considerable duration while selling it again later. It also applies to those who want no risks while the dwelling is being constructed. Authors utilize various classification methods such as Logistic regression, Decision tree, Naive Bayes, and Random Forest to work out the house's resale value. It also applies the AdaBoost technique to assist weak learners to be strong ones. The physical characteristics, location, as well as numerous economic aspects persuading at the time, decide the resale price of a house. Accuracy is used to measure performance for different datasets and unleash the optimal way for sellers while expecting the resale price. Sawant and Jangid [4] indicate that over the next decade, India's housing market is expected to increase at a rate of 30-35 42 A Literature Survey on Housing Price Prediction per cent. It is only second to the agriculture industry in terms of job creation. Pune makes it an excellent spot to invest in real estate. The inconsistency in housing valuation is a challenge for a house buyer. The estimated price must be a win-win midpoint for both the seller and the buyer. This will confirm whether the price is underestimated or overestimated. To do this, various features from the set of features are picked as input, while using algorithms such as Decision Tree and bagging techniques such as Random Forest. Wang et al. [5] state that studies that do not take into account all of the factors influencing property values, provide inaccurate forecast results. As a consequence, for house prediction, the authors propose a full circle joint self-attention model. Authors employ satellite imagery to assess the environment around the residential area. Input information about public facilities such as gardens, academic institutions, and BRT stops are used to depict the amenities.

## Data Exploration:

Let's begin by examining the distributions of the features, starting with the target variable, SalePrice. Ensuring that SalePrice follows a normal distribution is crucial because many machine learning algorithms assume that the input data is normally distributed. When data fits a normal distribution, it allows us to make more reliable inferences about the population using analytical techniques.




We can see the SalePrice distribution is skewed to the right. Let's transform it so that it follows a Gaussian normal distribution.




Values closer to zero are less skewed. The results show some features having a positive (right-tailed) or negative (left-tailed) skew. We can see YearBuilt is slightly skewed to the left but pretty much normal distributed while LotArea and PoolArea are highly skewed to the right. Highly skewed distributions in the dataset may benefit from data transforms in some way to improve our prediction accuracy.

## Train-Test Split dataset:

Before we can start modelling the data, we need to split the dataset into training and test sets. We will train the models with the training set and cross-validate with the test set. Recall we have lots of features in the dataset that are text. Most machine learning models require numerical input features. Since the process of converting text features to a numeric representation is an involved task, we will only use the numeric features in our price prediction (for simplicity's sake).

To split the dataset, we will use random sampling with a 75/25 train-test split; that is, we'll use 75% of the dataset for training and set aside 25% for testing.

## Modelling:

We will build four models and evaluate their performances with the R-squared metric. Additionally, we will gain insights into the features that are strong predictors of house prices.

LinearRegression(copy_X=True,fit_intercept=True, n_jobs=1, normalize=False)

Accuracy: 0.886663711473874

We will do cross-validation to see whether the model is overfitting the data.

Cross-validation results: [0.88426462 0.83605032 0.86145344 0.89201551 0.6154792 ]
R2: 0.817852618686709

It doesn't appear that for this train-test dataset, the model is not overfitting the data (the cross-validation performance is very close in value). It may be a slightly over-fitted but we can't really tell by the R-squared metric alone. If it is over-fitted, we can do some data transformations or feature engineering to improve its performance. But our main objective initially is to spot-check a few algorithms and fine-tune the model later on.

To help prevent over-fitting in which may result from simple linear regression, we can use regression models with regularization. Let's look at ridge and lasso next.

## Regularization:

The alpha parameter in ridge and lasso regularizes the regression model. The regression algorithms with regularization differ from linear regression in that they try to penalize those features that are not significant in our prediction. Ridge will try to reduce their effects (i.e., shrink their coeffients) in order to optimize all the input features. Lasso will try to remove the not-significant features by making their coefficients zero. In short, Lasso (L1 regularization) can eliminate the not-significant features, thus performing feature selection while Ridge (L2 regularization) cannot.

## Ridge Regression:

Cross-validation results: [0.88428067 0.83605927 0.86144661 0.89217415 0.61559687]
R2: 0.817911511863605

## Lasso Regression:

Cross-validation results:  [0.88474308 0.83495207 0.8596755  0.8932596  0.61075654]
R2:  0.8166773577482322

Alpha is the regularization parameter. The alpha values choosen for ridge and lasso serve as a starting point and are not likely the best. To determine the best alpha for the model, we can us e GridSearch. We would feed GridSearch a range of alpha values and it will try them all in cro ss-validation to output the best one for the model.

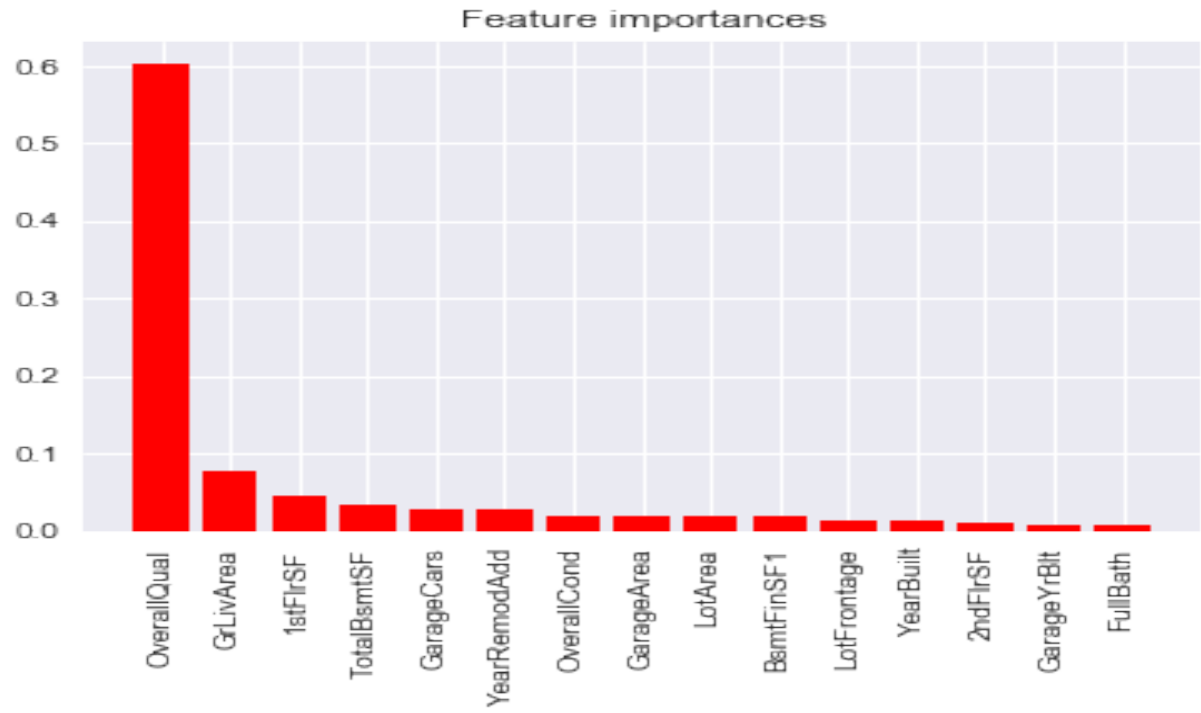
## Random Forest:

R2:  0.8459333542175683
Random forest is an advanced decision tree based machine learning. It has a classification and a regression random forest algorithm. Its performance is slightly better than regression. Like regularization, we can optimize the model parameters for best performance using gridsearch.

## Plotting the Feature Importance:

Let's see the features that are the most promissing predictors:

['OverallQual', 'GrLivArea', '1stFlrSF', 'TotalBsmtSF', 'GarageCars', 'YearRemodAdd', 'Overa llCond', 'GarageArea', 'LotArea', 'BsmtFinSF1', 'LotFrontage', 'YearBuilt', '2ndFlrSF', 'Garage YrBlt', 'FullBath']



## Conclusion:

Random Forest is the most accurate model for predicting the house price. It scored an estimated accuracy of 85%, outperforming the regression models (linear, ridge, and lasso) by about 2%. Random Forest determined the overall quality of a home is by far the most important predictor. Following are the size of above grade (ground) living area and the size of the total basement square footage. Surprisingly, the lot area did not rank as high as I had expected.

Machine learning is an iterative process. This first round of data exploration and model evaluation served as a good start to quickly gain insights to get a first reasonably good model prototype. There is a lot of structure in this dataset and further work is required to build a high performing prediction model.

Future research:

Try different types of data transforms to expose the data structure better, so we may be able to improve model accuracy

- Feature selection and removing the most correlated features (multicollinearity)
- Rescaling or normalizing the training dataset to reduce the effects of differing scales
- Standardizing the training set to reduce the effects of differing distributions
- Feature engineering to expose underlying data structures
- Binning of data (this can help improve accuracy for decision tree algorithms)

## References :

[1] J. Manasa, R. Gupta, and N. Narahari, “Machine learning based predicting house prices using regression techniques,” in 2020 2nd International conference on innovative mechanisms for industry applications (ICIMIA). IEEE, 2020, pp. 624–630.

[2] Y. Luo, “Residential asset pricing prediction using machine learning,” in 2019 International Conference on Economic Management and Model Engineering (ICEMME). IEEE, 2019, pp. 193–198.

[3] P. Durganjali and M. V. Pujitha, “House resale price prediction using classification algorithms,” in 2019 International Conference on Smart Structures and Systems (ICSSS). IEEE, 2019, pp. 1–4.

[4] R. Sawant, Y. Jangid, T. Tiwari, S. Jain, and A. Gupta, “Comprehensive analysis of housing price prediction in pune using multi-featured random forest approach,” in 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). IEEE, 2018, pp. 1–5.

[5] P.-Y. Wang, C.-T. Chen, J.-W. Su, T.-Y. Wang, and S.- H. Huang, “Deep learning model for house price prediction using heterogeneous data analysis along with joint self-attention mechanism,” IEEE Access, vol. 9, pp. 55 244–55 259, 2021.