House Price Prediction

Mahjabin Hassan

ITCS 5156 Spring 2023 – Desai

May 4, 2023

Abstract

The main purpose of this paper is to explore Random Forest machine learning technique to predict house prices. The random forest will be compared to other machine learning algorithms like Ridge, Linear, Lasso. The outputs of the results will then be compared to each other. This paper is based on the research paper "House Price Prediction using Random Forest Machine Learning" [1]. Other research papers [2,3] will also be explored to identify other predictions techniques available, and problems and challenges associated with them.

1 Introduction

House prices depend on a lot of factors. For anyone, the decision to buy property, especially a house, is a very important financial decision. Machine learning predicting techniques can help prospective owners gauge the upcoming housing market and ease the process of buying a house.

Housing price is an important factor that reflects the economy [1]. There are so many ways housing prices can be explored and predicted using machine learning. This paper tries to follow the approach of random forest regression to predict house prices [1]. Regression, inference, neural networks, and deep learning are some of the most popular machine learning methods. Truong et al. compares the conventional techniques with more advanced techniques [2]. Similarly, another paper by Abdul-Rahman et al. compares recent advanced ML algorithms named LightGBM and XGBoost with multiple regression analysis and ridge regression.

This paper tries to follow the approach of random forest regression to predict house prices [1]. In order to find the best model, it is compared to Linear, Ridge and Lasso regression. Performance evaluation is performed using R2 score, Maximum Error (MaxErr), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Mean Squared Error (MSE).

2 Related Work

Prediction models use supervised machine learning. Supervised machine learning involves labelled data and unsupervised machine learning has unlabeled data. Price prediction has two categories. First, category forecast trends in a time series format and second category estimates price depending on the characteristics that influence the price for example house price [1]. The second category is our point of interest for the paper and there are a lot of papers which also deal with this second category.

One such study was conducted in Kuala Lampur housing data set, by Abdul-Rahman et al. [3]. The study found out that advanced regression techniques performed better than traditional techniques. It used two advanced techniques Extreme Gradient Boosting (XGBoost) and Light Gradient Boosting Machine (LightGBM). XGBoost was the better performer out of the two and it also mentioned that it is one of the most powerful algorithms for any regression or classification problem.

Another paper by Truong et al. also predicts housing prices on a Beijing housing dataset [3]. This study compared Random Forest, LightGBM and XGBoost but in contrast to [3] it used Hybrid Regression and Stacked Generalization Regression for optimal solutions [2]. The study found out that all the techniques were able to predict house prices, but each had their respective pros and cons. For example, LightGBM was the best in terms of time and the Hybrid Regression method was the simplest but performed better due to the generalization.

3 Methods

The dataset for the project was obtained from Github. https://github.com/pplonski/datasets-for-start/blob/master/house_prices/data.csv. The dataset was similar to the Boston dataset that was used in the main research paper from which this study is adapted. The paper did not provide a source code. The Boston dataset was an older dataset with few data entries. Therefore, for this study this github dataset was used which had 81 columns (features) and 1460 rows (data entries). The project used this notebook as a reference https://www.kaggle.com/code/subhradeep88/house-price-predict-decision-tree-random-forest. After the dataset was loaded, data exploration was performed in order to identify the various important characteristics. For example, if it requires data cleaning or preprocessing before the different machine algorithms or models can be applied.

The first step was to check the different features. There were 81 features, after having a closer look at the features a lot of the features seemed to be repetitive. So, removing some of them would help us to have a much nicer dataset. Next, the check for missing or null values was performed. A list of the features with a lot of null values was obtained. The figure below shows the total number of missing values for each feature.

```
PoolQC          1453
MiscFeature     1406
Alley           1369
Fence           1179
FireplaceQu      690
LotFrontage      259
GarageYrBlt       81
GarageCond        81
GarageType        81
GarageFinish      81
GarageQual        81
BsmtFinType2      38
BsmtExposure      38
BsmtQual          37
BsmtCond          37
BsmtFinType1      37
MasVnrArea         8
MasVnrType         8
Electrical         1
Id                 0
dtype: int64
```

Fig.1 Total missing value count

Looking at the above results, the features till Lotfrontage were dropped since most of the values are missing also dropping id as this only an id number of the houses. Garage year built, garage condition, garage type, garge finish, garage quality was also dropped as Garage Cars could give the same information. Similarly dropping BsmtFinType2, BsmtExposure, BsmtQual, BsmtCond, BsmtFinType1 since the basement features had no missing values and other basement features can give us the same information. MasVnrArea, MasVnrType are not important features for house price so deleting this would not affect the analysis.

After preprocessing steps, the new dataset had 34 features and 1459 entries. Only numerical values were considered for machine learning since the model will predict sale price which is a regression technique and not classification. Data exploration on the sale price also showed the distribution of the prices. The figure below is a representation of the sale price.

```
# Data Exploration
sale_price_desc = df_house['SalePrice'].describe()
sale_price_desc
```

```
count      1459.000000
mean     180930.394791
std       79468.964025
min       34900.000000
25%      129950.000000
50%      163000.000000
75%      214000.000000
max      755000.000000
Name: SalePrice, dtype: float64
```

Fig. 2 Sale price distribution

Only numerical values were considered for machine learning since the model will predict sale price which is a regression technique and not classification. The models used for the study were Linear Regression, Ridge, Lasso and Random Forest.

Linear regression is a method used to predict relationships between variables. It assumes a linear relationship between dependent and independent variables and finds the best fitting line. The sum of the squared differences between actual and predicted values determines the best fit line.

Next regularized linear regression like ridge and lasso is implemented. Ridge regression uses L2 regularization. It is normally used in data that has multicollinearity. Lasso is another regression method which uses L1 regularization. In order to improve accuracy, it performs both regularization and variable selection.

Random Forest Regression is an ensemble learning method technique. It is a popular supervised learning technique. Ensemble learning performs better by combing the predictions from multiple machine learning models and therefore makes more accurate predictions than a single model.

4 Experiments

Before carrying out the experiments with different machine learning models, the data needs to be split into test and train data using Scikit learn train_test_split function. 80% was train datan 20% was test data.

```
# Splitting data into test and train data
# 80% train data and 20% test data

from sklearn.model_selection import train_test_split

def data_splitting(data, target):
    X_train, X_test, t_train, t_test = train_test_split(data,target, test_size=0.2, random_state=0)
    return X_train, X_test, t_train, t_test

X_train, X_test, t_train, t_test = data_splitting(
    data = df_house_n.iloc[:, :33],
    target = df_house_n['SalePrice']
)

print("Train data shape: {}".format(X_train.shape))
print("Train target shape: {}".format(t_train.shape))
print("Test data shape: {}".format(X_test.shape))
print("Test target shape: {}".format(t_test.shape))
```

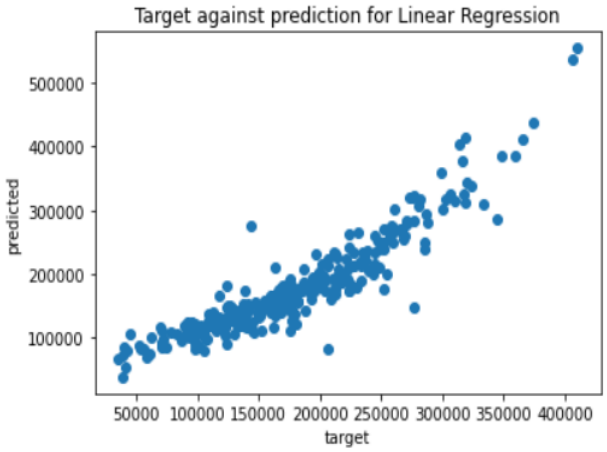
```
Train data shape: (1167, 33)
Train target shape: (1167,)
Test data shape: (292, 33)
Test target shape: (292,)
```

Fig.3 Data splitting

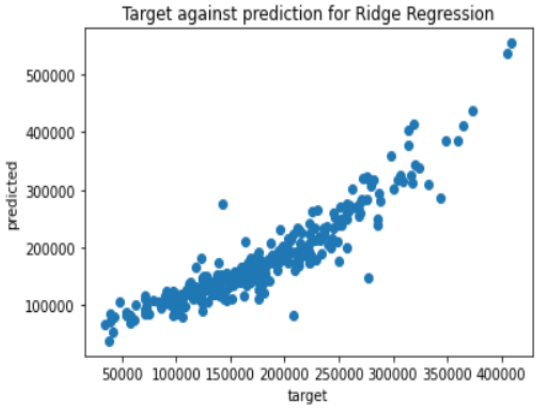
Next the models were implemented using the Scikit learn library. The Github repository for the project https://github.com/mahjhass/ITCS-5156-Machine-Learning.git has the code for the Linear,

Ridge, Lasso and Random Forest regression. The below figure shows the output of the models implemented.

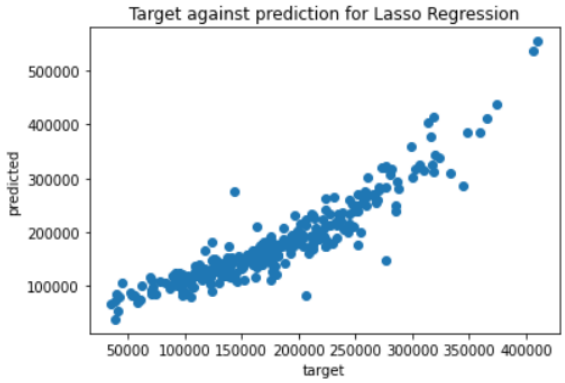
```
Test score/Accuracy: 0.8437564137343283

EVALUATION METRICS:
R2_score:  0.8437564137343283
MaxErr:  144408.4179855144
MAE:  21782.460958898362
MAPE:  0.13676206496804927
MSE:  949653271.4733695

Text(0.5, 1.0, 'Target against prediction for Linear Regression')
```




```
Test score/Accuracy: 0.8445160764819009

EVALUATION METRICS:
R2_score:  0.8445160764819009
MaxErr:  145322.99206565705
MAE:  21653.058038450068
MAPE:  0.13569852816307643
MSE:  945036018.1787478

Text(0.5, 1.0, 'Target against prediction for Ridge Regression')
```



```
Test score/Accuracy: 0.8437576404122049

EVALUATION METRICS:
R2_score:  0.8437576404122049
MaxErr:  144409.60372044996
MAE:  21782.289561851198
MAPE:  0.1367608232780223
MSE:  949653271.4733695

Text(0.5, 1.0, 'Target against prediction for Lasso Regression')
```




```
Test score/Accuracy: 0.88616280426496

EVALUATION METRICS:
R2_score:  0.88616280426496
MaxErr:  142378.21000000002
MAE:  16940.69544520548
MAPE:  0.10547837816652159
MSE:  691905939.4944715

: Text(0.5, 1.0, 'Target against prediction for Random Forest Regression')
```

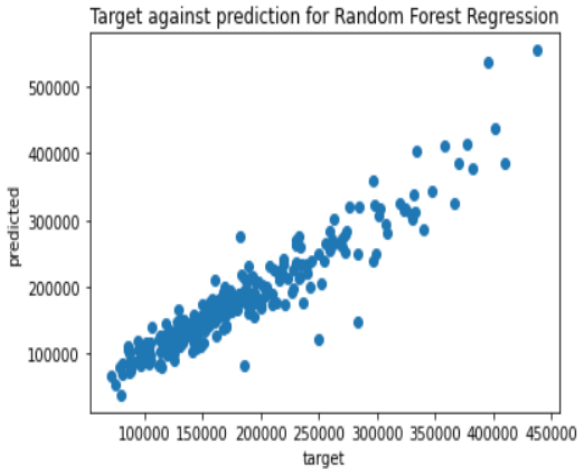


Fig.4 Evaluation Measure output of different models

From the above results it can be seen that random forest performs the best out of all the models. It has an accuracy of 88.6% which is the highest amongst all compared. The performance evaluation metrics also support the analysis. The results are as expected and are similar to the main paper [1]. The only difference is that the proposed paper only implemented the random forest method to make a housing price prediction [1]. They removed the housing prices from the actual dataset and simulated the model to determine the predictive power of the random forest technique. But for this study, random forest was compared to other regression techniques to predict housing price to find the best machine learning model. In addition to that several visualizations were created to observe the target against predicted values.

Finally in order to explore more, a figure was plotted for the predicted and the actual values in random forest regressor model. The output can give an idea of how close the predicted and the actual values were.

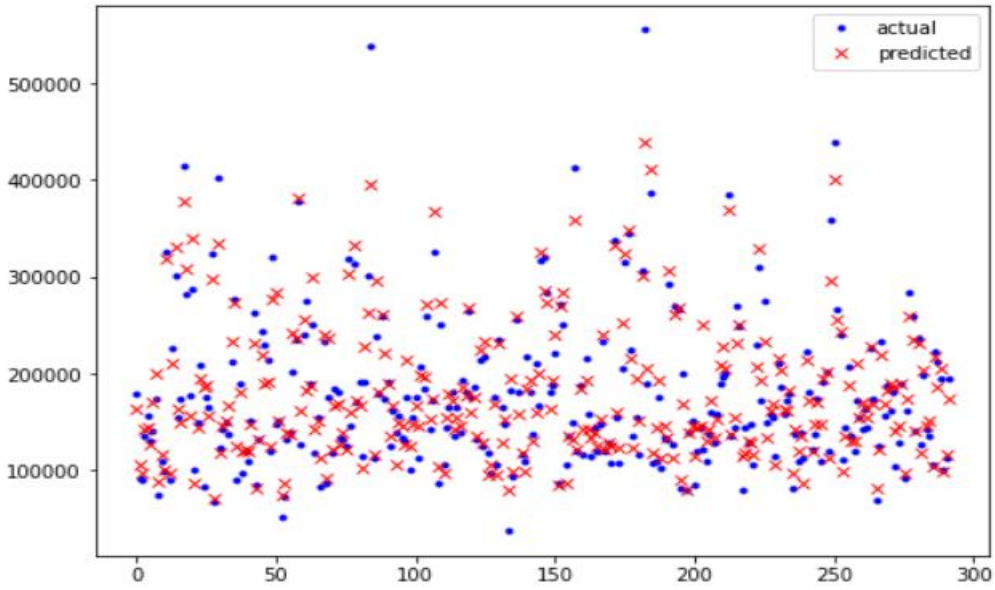


Fig. 5 Plot of Predicted vs Actual for Random Forest Regressor

5 Conclusions

The purpose of this project was fulfilled which was to check if random forest technique can predict house prices better than other regression models. The main challenge was in the data cleaning process as it had a lot of features. A lot more can be done with this dataset. From the other research papers, it was clear that various machine learning models can predict house prices. But depending on the researchers' level of machine learning knowledge, they implemented advanced techniques as well. Since I am new to machine learning, I was able to implement what I learned from this machine learning course. In future, I plan to explore advanced machine learning techniques along with different classification techniques. Feature selection and hyperparameter selection can also help develop a better prediction model in the future.

Sharing agreement

Do you agree to share your work as an example for next semester?

No

References

[1] A. B. Adetunji, O. N. Akande, F. A. Ajala, O. Oyewo, Y. F. Akande, and G. Oluwadara, "House price prediction using random forest machine learning technique," *Procedia Comput. Sci.*, vol. 199, pp. 806–813, 2022.

[2] Q. Truong, M. Nguyen, H. Dang, and B. Mei, "Housing price prediction via improved machine learning techniques," Procedia Comput. Sci., vol. 174, pp. 433–442, 2020.

[3] S. Abdul-Rahman, N. H. Zulkifley, I. Ibrahim, and S. Mutalib, "Advanced machine learning algorithms for house price prediction: Case study in Kuala Lumpur," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 12, 2021.