Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/4/2023362

House price prediction based on machine learning

Hanwen Li

Crossroads Christian High School, CA, 92881, United States

jli@crossroadshs.com

Abstract. Machine learning is commonly used in the real estate market. It is vital to apply the idea of machine learning in this field to predict house prices based on various features. The paper will focus on how to use the most appropriate machine learning models for house price prediction. It will use LightGBM(Light Gradient Boosting Machine), Gradient Boosting, and XGBoost(Extreme Gradient Boosting) to train models to predict house prices using the existing data from the Kaggle website. After three models make predictions, they will get an RMSE (root mean square error), which is 0.02975, 0.02537, and 0.01364. Based on the result, the XGBoost model is the best one among these three models used for house price prediction.

Keywords: House Price Prediction, Machine Learning, Gradient Boosting Regression, LightGBM Regression.

1. Introduction

The demand for houses continues to increase as the population always grows. House prices have gradually become one of the most concerning topics. There has been a lot of research on house price prediction through machine learning. Some papers do some research on different models. The paper "House Price Prediction via Improved Techniques"[1] uses multiple models that also include XGBoost and LightGBM, and the comparison between these two models also indicates that XGBoost has better accuracy. Two other papers, "House Price Prediction using a Machine Learning Model: A Survey of Literature"[2] and "House Price Prediction with Gradient Boosted Trees Under Functions"[3] also introduce the XGBoost model as a type of Gradient Boosted Trees. Meanwhile, the Decision Tree is also used for prediction, which is simpler[4]. The main goal of this paper is to train a model that can predict house prices based on the features of houses. This paper will do some further research on the differences between different models for housing prediction, and this paper will specifically focus on the comparison of different implementations of the Gradient Boosted Trees algorithm. The paper will choose the Gradient Boosting regression model, LightGBM regression model, and XGBoost regression model for prediction using the data from the Kaggle website. The study of house price prediction through machine learning has many crucial benefits. For bankers and economists, house price prediction can help them estimate changes in the mortgage rate. For real estate investors, house prediction can help them foresee the potential values of properties and decide which location is viable for building houses.

DOI: 10.54254/2755-2721/4/2023362

2. Data

2.1. Data introduction

This paper uses a data set from kaggle.com[5], House Prices-Advanced Regression Techniques. The data set has a size of about one hundred variables on different attributes such as lot size in square feet, location, quality, number of floors, number of bedrooms, etc.

2.2. Data preprocessing

Firstly, the missing data should be investigated and dealt with. Any attribute with missing data should be filled through prediction or removed. For instance, there is lots of missing data in LotFrontage, and the paper can use linear regression to figure out the relationship between LotFrontage and LotArea. As shown in Figure 1, a linear regression can be used to predict the missing data for LotFrontage, and the data can be filled in for LotFrontage. For those attributes that have only few missing data, they can be filled in with median and mean.

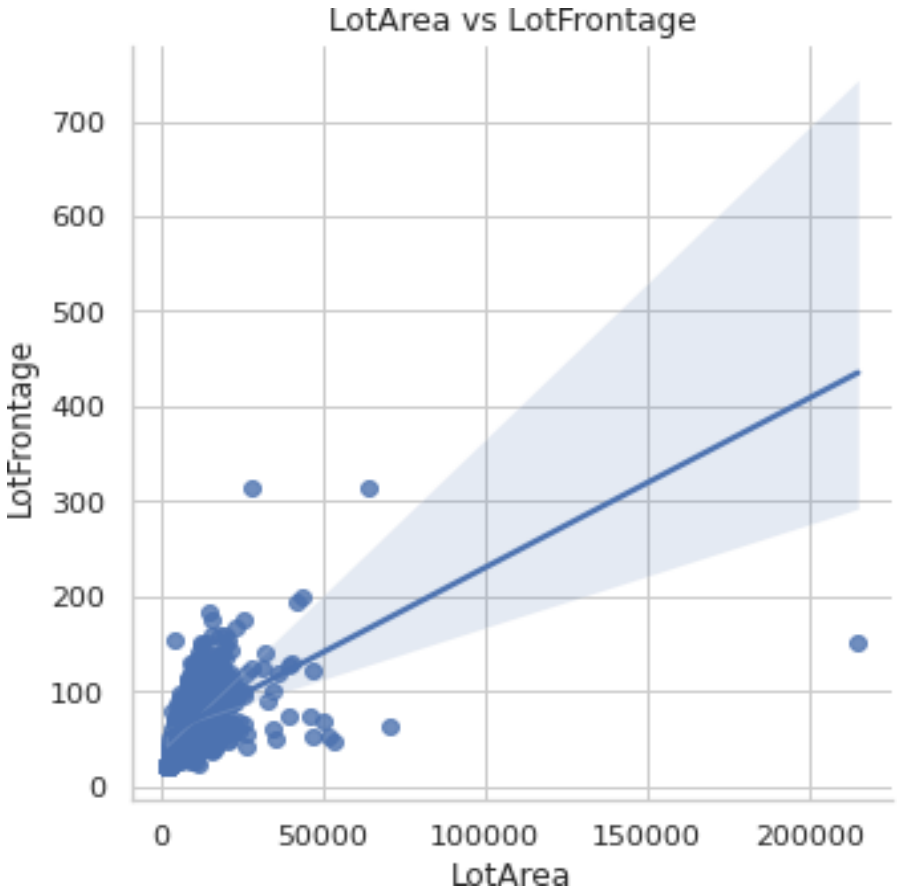

Figure 1. Linear Regression for LotArea vs LosFrontage.

2.3. Categorical variable encoding

For those categorical variables that are not numerical, the author has to convert them to numerical for the algorithm to identify them. This paper uses One Hot Encoding to implement the process of conversion. The basic idea of One Hot Encoding is to change categorical variables into a new form that could be used by machine learning algorithms. This method uses 0 to indicate non-existent while it uses 1 to indicate existent. Based on the method, if the house has a certain attribute, the position of the vector corresponding to this attribute should be 1.

DOI: 10.54254/2755-2721/4/2023362

3. Model architecture

3.1. Model selection

The next step is to select appropriate models so they can be trained and learn values more efficiently. The RMSE(root-mean-square error) can be used to deter- mine how accurately the model predicts the measured values. The smaller this error, the better prediction is done by the model. The formula is as follows:

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\left(\frac{d_i - f_i}{\sigma_i}\right)^2}$$

This paper can use the Cross-validation method to find the best predicting models. Cross-validation is one of the most popular data re-sampling methods that can estimate the true prediction error of models[6]. This paper uses the Cross-validation evaluating estimator from the Scikit-learn library[7] to compute the RMSE. As a result, Gradient Boosting, LightGBM, and XGB only have an error of 0.06, which is the lowest value among several models.

Table 1. Cross-validation result

#	Model Name	RMSE
4	Grandient Boosting	0.06
3	LightGBM	0.06
2	XGBoost	0.06
5	Bayesian Ridge	0.07
1	Ridge	0.07
0	Lasso	0.17

3.2. Light Gradient Boosting Machine(LightGBM)

LightGBM is a type of Gradient Boosting Decision Tree with high efficiency and low storage[1][8]. Meanwhile, it can also handle large-scale data[1]. The LightGBM is different from XGBoost, even though their algorithms are the same. Compared with the XGBoost model, the LightGBM grows tree leaf-wise while XGBoost grows tree level-wise. This paper uses the LightGBM Regressor from the Scikit-learn library to train the model for prediction[9]. The ideal hyperparameter is set as follows:

- Set learning rate=0.1
- Set loss=squared error
- Set n estimator=300

DOI: 10.54254/2755-2721/4/2023362

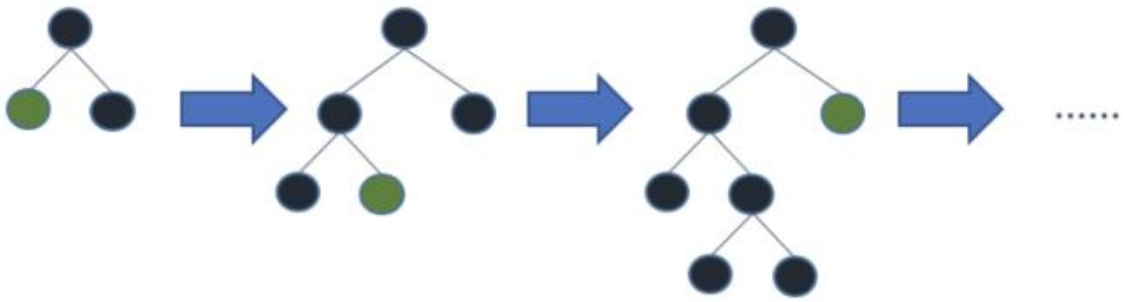

Figure 2. The basic logic of LightGBM which performs a leaf-wise tree growth.

3.3. Gradient Boosting

Gradient Boosting was introduced by J. H. Friedman[10] and is a useful machine learning algorithm because of its good conduct. The basic idea of Gradient Boosting algorithm is to randomly use a subsample to replace the full sample every iteration, and this random subsample will be used to compute the model update[9]. The Gradient Boosting Regressor from the Scikit-learn library[11] can help train a model. The best hyperparameter is set as follows:

- Set learning rate=0.1
- Set loss=squared error
- Set n estimator=300

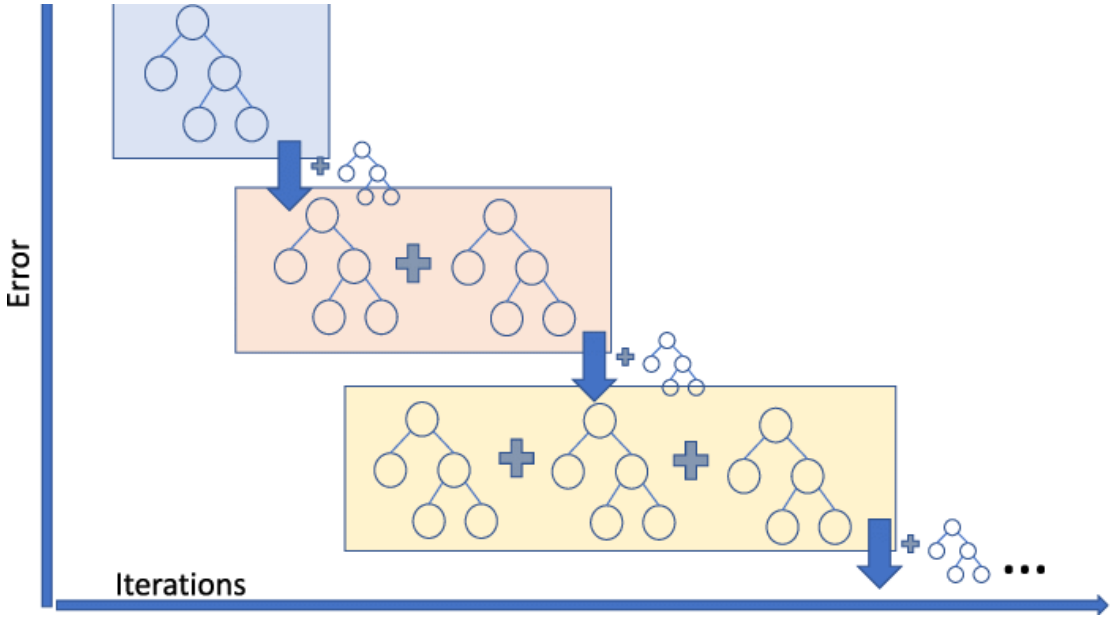

Figure 3. The schematic diagram for Grandient Boosting.

3.4. Extreme Gradient Boosting(XGBoost)

The Extreme Gradient Boosting is also a type of Gradient Boosting Decision Tree, and it has the most distinctive character, which is large scale[12]. XGBoost becomes very popular due to its tremendous scalability[1]. It is more than ten times faster than other existing systems and can scale to billions of examples much more efficiently[1] [12]. This paper uses the XGB Regressor from the XGBoost

DOI: 10.54254/2755-2721/4/2023362

Python package[13] to train the model for prediction. The most appropriate hyperparameter is set as follows:

- Set learning rate=0.1
- Set n estimator=1000

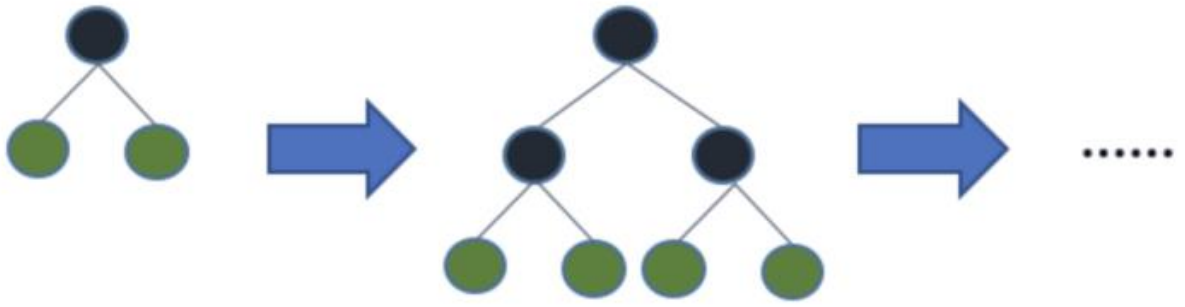

Figure 4. Level-wise Tree Growth.

Unlike the LightGBM, the XGBoost uses level-wise tree growth as shown in Figure 4.

4. Result and discussion

This paper uses different models for housing price prediction and investigates the difference between each model's accuracy for prediction. This paper uses the cross- validation method to determine the three most appropriate Machine learning models, including LightGBM, Gradient Boosting, and XGBoost. All three models can successfully predict house prices and get a result(RMSE score) back. As shown in Table 2, even though all three methods finally achieve satisfactory accuracy and results, their prediction results are still different from each other due to their different advantages and disadvantages. The LightGBM model achieves accuracy with an RMSE of 0.02975. Although the accuracy of the LightGBM model is not as high as other models, its high efficiency and low storage are outstanding compared to other models. The Gradient Boosting model performs well, with an RMSE of about 0.02537. The error is smaller than the error of the LightGBM model, which means the Gradient Boosting model makes a better prediction than the LightGBM model. The XGBoost model achieves the highest accuracy among these three models that the RMSE is only about 0.01364. Compared with the LightGBM model and the Gradient Boosting model. The error of the XG Boost is about only half of the first two models, and therefore this model performs the best prediction.

Table 2. Results for three different models

Type of Models	RMES (Root Mean Squre Error)
LightGBM	0.02975
Gradient Boosting	0.02537
XGBoost	0.01364

5. Conclusion

In this paper, LightGBM, Gradient Boosting, and XGBoost are used to train models to predict possible selling prices for houses. Each model has its result RMSE score, which indicates how close the

DOI: 10.54254/2755-2721/4/2023362

predicted value is to the actual value. XGBoost has the smallest error, which also represents the best model; Gradient Boosting has the second largest error, which is slightly lower than the error of LightGBM; LightGBM has the largest error of 0.02975, which indicates the least accurate model. These three descent methods are useful for making predictions, but the accuracy of prediction is limited since their algorithms are all based on the Gradient Boosted Trees and are all very similar. More types of methods can be used to train models and may yield a more accurate result. In the future, research should be continued for further investigation of more models, such as Linear and Polynomial Regressions, Hybrid Regression, Random Forest.

Acknowledgment

Firstly, I would like to show my deepest gratitude to my teachers and professors in the program, who have provided me with valuable guidance in every stage of the research and writing. Further, I would like to thank all of those who support and encourage me. Without all their enlightening instruction and impressive kindness, I could not have completed my thesis.

References

[1] Truong, Q., Nguyen, M., Dang, H., & Mei, B. (2020). Housing price prediction via improved machine learning techniques. Procedia Computer Science, 174, 433-442.

[2] Zulkifley N, Rahman S, Hasbiah U. House Price Prediction using a Machine Learning Model: A Survey of Literature. December 2020 International Journal of Modern Education and Computer Science 12(6):46-54.

[3] Hjoirt A, Pensar J, Scheel I, Sommervoll D (2022). House Price Prediction with Gradient Boosted Trees Under Functions. JOURNAL OF PROPERTY RESEARCH2022, AHEAD OF PRINT, 1-27.

[4] Kuvalekar, A., Manchewar, S., Mahadik, S., & Jawale, S. (2020, April). House Price Forecasting Using Machine Learning. In Proceedings of the 3rd International Conference on Advances in Science & Technology(ICAST).

[5] Kaggle. House Prices–Advanced Regression Techniques. Data. https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data (accessed August 9, 2022)

[6] Berrar, D. (2018, January). Cross-Validation. Reference Module in Life Sciences. doi:10.1016/B978-0-12-809633-8.20349-X

[7] Scikit-Learn. Cross-validation: evaluating estimator performance. https://scikit-learn.org/stable/modules/cross_validation.html (accessed August 9, 2022)

[8] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems, 30.

[9] Microsoft. https://github.com/microsoft/LightGBM (accessed August 9, 2022).

[10] Friedman, J. H. (2002). Stochastic gradient boosting. Computational statistics & data analysis, 38(4), 367-378.

[11] Scikit-Learn. Gradient Boosting Regressor. https://scikit-learn.org/ sta- ble/modules/generated/sklearn.ensemble. GradientBoostingRegressor.html (accessed August 9, 2022).

[12] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD 16 2016. doi:10.1145/2939672.2939785.

[13] DMLC. xgboost. GitHub. https://github.com/dmlc/xgboost (accessed Au- gust 9, 2022).