

Guangzhou Housing Price Prediction Based On Machine Learning Regression Models

Ziyan Huang^{1, *, †}, Kexin Li^{2, †}, Chuming Wang^{3, †}, and Jiazhi Wang^{4, †}

¹Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast, BT7 1NN, UK

²Computer Science, Guangdong University of Foreign Studies, Guangzhou, Guangdong, 510006, China

³College of Liberal Arts and Sciences, Iowa State University, Iowa, 50011, United States of America

⁴College of Liberal Arts and Sciences, University of Illinois, Urbana-Champaign, Champaign, IL 61820, United States of America

*Corresponding author email: 20191003175@gdufs.edu.cn

†These authors contributed equally to this article

Abstract. Nowadays, the fluctuation of housing prices is a key concern of homeowners, the real estate market and the government. The researchers found that it was possible to predict house prices accurately by analyzing relevant attributes and using the most effective models. However, due to the impact of the COVID-19 pandemic, there is still a research gap on which model is more suitable for predicting housing prices after the outbreak of COVID-19. Therefore, this study collected the latest real estate data of Guangzhou in China through the web crawler, trained the random forest regression model with collected data, and obtained a model that could predict the housing price by inputting corresponding attributes. The MSE of the trained regression model is 2802, MAE is 534, and the determination coefficient (R^2_score) is 0.89, while the MSE of the trained XGBoost regression model is 3108, MAE is 643, and the determination coefficient (R^2_score) is 0.87. The random forest regression model has been shown to be more accurate at predicting Guangzhou house prices after the COVID-19 outbreak. Our paper has great potential in house price prediction under the pandemic situation.

Keywords: Random Forest Regression Model, XGBoost Regression Model, Housing prices, COVID-19.

1. Introduction

China's housing prices have been one of the main concerns. Around 1990, housing privatization greatly promoted the real estate market [1]. The housing reserves and housing prices got a strong boost. The government gradually implemented housing monetization, accelerated housing construction, filled housing demand, and made housing become a new economic growth point in China. In 2008, the Subprime mortgage crisis dealt a severe blow to the global economy, and China's economy was also affected by the crisis [2]. China's housing prices began to slide. To this end, the Chinese government proposed the Chinese Economic Stimulus Program, ushering the country into the era of large-scale infrastructure [3]. By expanding investment in this way, China has successfully boosted domestic demand and achieved remarkable GDP growth. Rising housing prices and housing inventories have made the supply side gradually outstrip the demand side, and people are beginning to realize that they might not be able to keep up with the increase in home prices [4].

Now, the COVID-19 pandemic has affected housing prices in China. Around 2021, prices start to slide [5]. Fluctuations in housing prices might lead to bias in pre-pandemic forecasting models. This paper focuses on the housing price volatility in Guangzhou after the outbreak of COVID-19. Most housing price prediction models related to Guangzhou included data before the pandemic. When COVID-19 enters people's lives, local policies, per capita income, consumer prices, consumption

levels, etc., may indirectly interplay the local housing price [6], reducing the accuracy of previous forecasting models [7]. Therefore, forecasting housing prices using data from the pandemic will be able to create forecasting models with higher accuracy. A highly accurate forecast model of house prices will have important reference value for the post-COVID-19 economic layout. Web crawlers will be used to obtain the required housing data. The study will use the Random forest regression and the XGBoost regression for forecast model training, and the model's reliability will be calculated.

This study used the Guangzhou housing data (web-crawled in June 2022) from Lianjia (bj.lianjia.com) to get forecast models that can predict housing prices during and after COVID-19. This study used mean absolute error score and mean squared error score to rate the models' accuracy, and used the coefficient of determination to rate the models' rationality. The random forest regression forecast model's MSE score is 2802, the MAE score is 534, and the R^2 _score is 0.89. The XGBoost regression forecast model's MSE score is 3108, the MAE score is 643, and the R^2 _score is 0.87. Based on the scores above, the forecast model training using the random forest regression method has better accuracy than the XGBoost regression.

2. Method

2.1. Data format processing

The data is from Lianjia.com, which is a website providing information about all sorts of houses and apartments including newly-built and second-hand. This paper collected house price data of Guangzhou in the past six months in 2022 which could generally represent the overall house price trend during the period of COVID-19.

The dataset contains nineteen attributes including house names, regions, built-year and etc. This research only applied nine attributes in regression models building phase using 'unit_price' as label attribute. The names and meanings of the remaining eight attributes are as follows.

Table 1. Data attribute introduction

Attributes	Description
room	The number represents the amount of bedrooms.
living room	The number represents the amount of living rooms.
area	The number represents the total area of the house
decoration	The number represents the decoration degree of the house. (1:simply decorated, 2:well decorated, 3:other situation)
elevator	The number represents whether the house has an elevator or not. (1:yes, 0:no)
level	The number represents the height of the house. (1:below 10 floors, 2: between 10 to 20 floors, 3: over 20 floors)
total_level	The number represents the total floors of the building.
building_year	The year number represented when the house was built.

After removing the null and missing values, further information on the dataset can be checked using python commands in the Jupyter notebook(version-6.1.4). Table 2 is the output after apply pandas' describe() function to the collected data.

Table 2. Dataset information

	room	livingroom	area	decoration	elevator	level	total_level	building_year	unit_price
count	31349.000000	31349.000000	31349.000000	31349.000000	31349.000000	31349.000000	31349.000000	31349.000000	31349.000000
mean	2.423203	1.698236	82.055915	2.640339	0.746435	2.053750	20.414973	2005.284698	32730.612045
std	0.609418	0.461595	19.697528	0.484605	0.435058	0.743361	10.045352	7.033434	8824.138890
min	1.000000	0.000000	23.740000	1.000000	0.000000	1.000000	3.000000	1970.000000	7728.000000
25%	2.000000	1.000000	65.700000	2.000000	0.000000	1.000000	9.000000	1999.000000	27065.000000
50%	2.000000	2.000000	85.410000	3.000000	1.000000	2.000000	18.000000	2003.000000	28117.000000
75%	3.000000	2.000000	95.730000	3.000000	1.000000	3.000000	29.000000	2012.000000	41404.000000
max	7.000000	4.000000	513.620000	3.000000	1.000000	3.000000	56.000000	2018.000000	121433.000000

In order to improve the performance and efficiency of the regression models, data standardization is needed to control every attribute within a certain standard range, so that their values can be on the same scale. The ‘StandardScaler’ method in Scikit-Learn package was therefore used in this phase. The final results are as follows.

Table 3. Standardized data set

	room	livingroom	area	decoration	elevator	level	total_level	building_year
0	-2.321631	-1.506599	-1.805836	0.739834	0.58375	-0.07631	0.057589	0.955103
1	0.945994	0.656318	0.598517	0.739834	0.58375	-1.42025	1.152800	1.097165
2	-0.687818	0.656318	-0.481765	0.739834	0.58375	-0.07631	-0.241104	0.670979
3	0.945994	0.656318	0.943927	0.739834	0.58375	1.26763	0.754541	-0.749638
4	0.945994	0.656318	0.598517	0.739834	0.58375	-1.42025	1.152800	1.097165
...	...	...	...	...	...	...	...	...
25074	0.945994	-1.506599	0.303166	-1.326346	0.58375	1.26763	-0.539798	-1.033761
25075	-0.687818	0.656318	-0.853206	0.739834	0.58375	-1.42025	1.351929	1.381288
25076	0.945994	0.656318	0.959946	0.739834	0.58375	1.26763	0.953671	-0.039329
25077	-0.687818	0.656318	-0.332588	-1.326346	0.58375	1.26763	-0.241104	0.670979
25078	-0.687818	0.656318	-0.853206	0.739834	0.58375	-1.42025	1.351929	1.381288

2.2. Data analysis

Through the use of Python's 2D graphics library Matplotlib, a heatmap examining the correlation between various attributes may be created using the second-hand house pricing data gathered by Lianjia.com for the previous six months in Guangzhou in 2022. As the result of data visualization, heatmap is the most intuitive form.

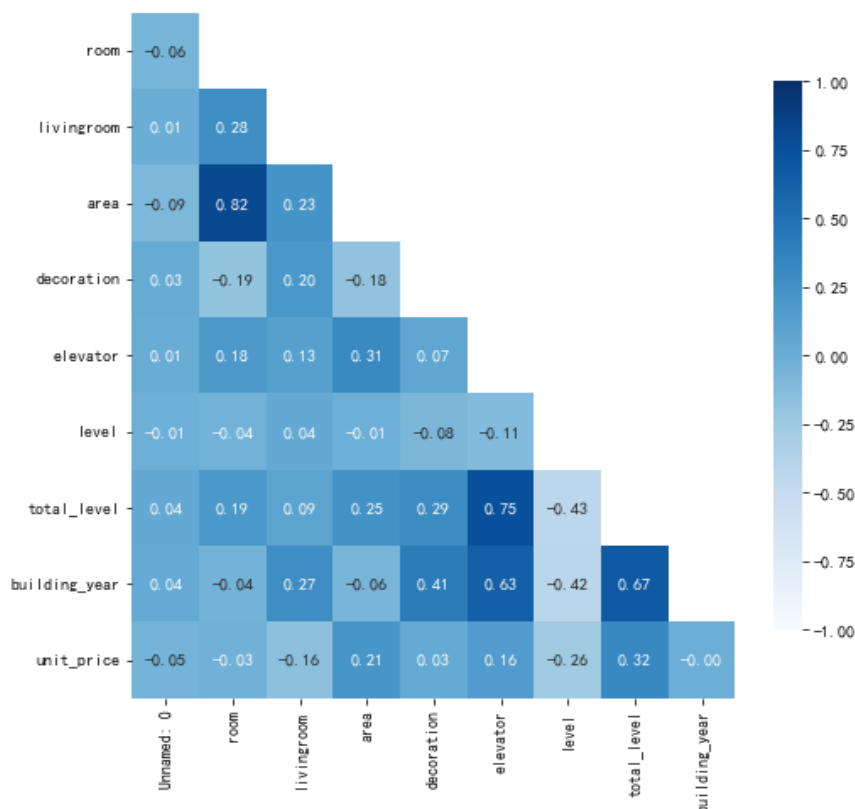


Figure 1. Heatmap of attribute relationship

A preliminary understanding of the relationship can pave the way for further exploration of the importance of features under different models after model building. Thus, the factors that may affect the unit_price are analyzed.

According to the heatmap, the attributes with positive correlation can be selected for further analysis. For example, data samples were extracted to observe the relationship between a single building_year and unit_price, and the result is as follows. It is difficult to see an obvious correlation between the two.

the impact of construction time on housing prices

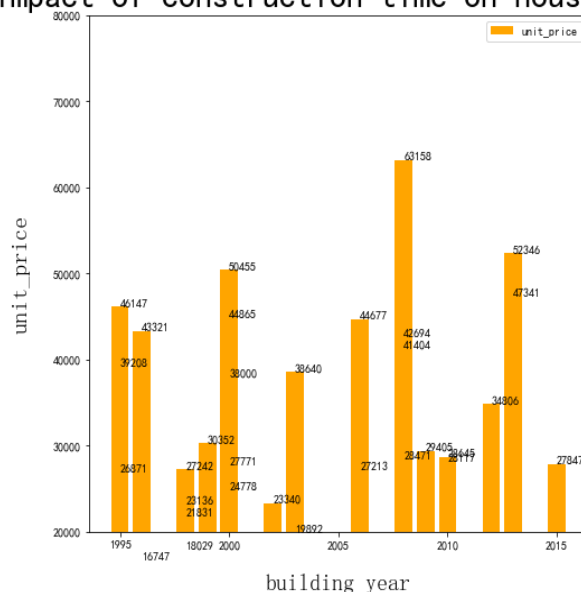


Figure 2. The impact of building_year on unit_price.

Analysis of the impact of elevators on unit_prices, histogram drawing can be clearly displayed. Through the extraction of data samples, it can be seen that the majority of second-hand houses in

Guangzhou are equipped with elevators. There is only a slight difference, but houses with elevators do cost more than those without.

Although only 0.03 positive correlation between unit_price and decoration can be seen from the heat map. But only on the decoration of this attribute to draw a visualization, it can be seen that most second-hand houses are in the fine decoration and the more beautifully decorated and fully furnished the house, the higher the price.

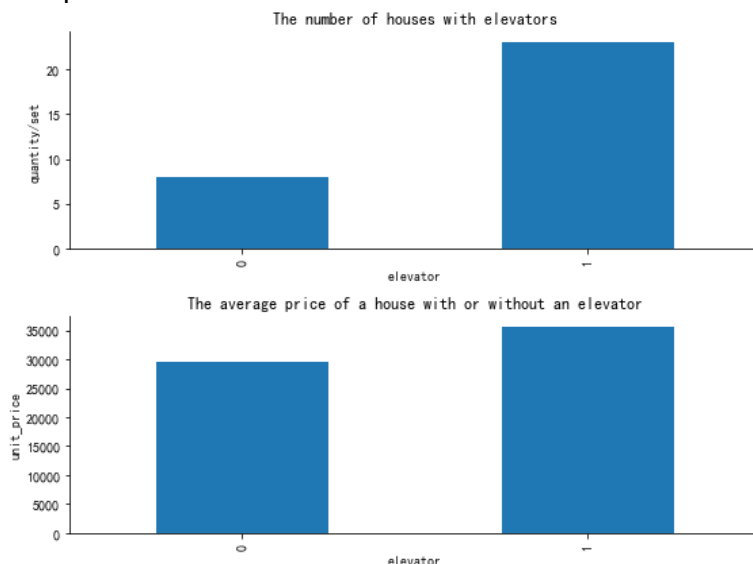


Figure 3. The ratio of elevators and their impact on unit_price

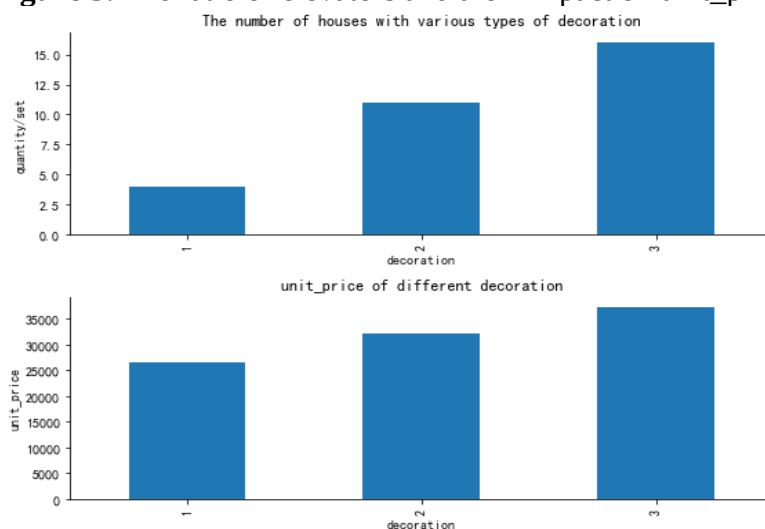


Figure 4. The impact of decoration level on unit_price

2.3. Random forest regression model

2.3.1 Algorithm

Random forest is an algorithm that trains and predicts samples using multiple decision trees. That is, the random forest has an algorithm containing numerous separate decision trees, and the final output of all the decision trees is the same with the category of most decision trees. It is handy to invoke all the functions related to random forest algorithm in Scikit-Learn package which encapsulates the methods in the module called 'ensemble'. While random forest can be flexibly applied to different specific tasks in machine learning including the most common ones like classification and regression, random forest demonstrates an extremely excellent performance in the tasks of classification. It adopts bagging, an integrated idea that generates many parallel classifiers through the autonomous sampling method to determine the final result through the principle of "the minority is subordinate to the majority". Random forest significantly solves the disadvantage of the

weak generalization ability of the decision tree. The randomness of random forest learners functions both in selecting the sample trees and in selecting the features. Random forest is known as "the algorithm representing the computing level of integrated learning" which is inexpensive to compute and quite simple to implement.[8].

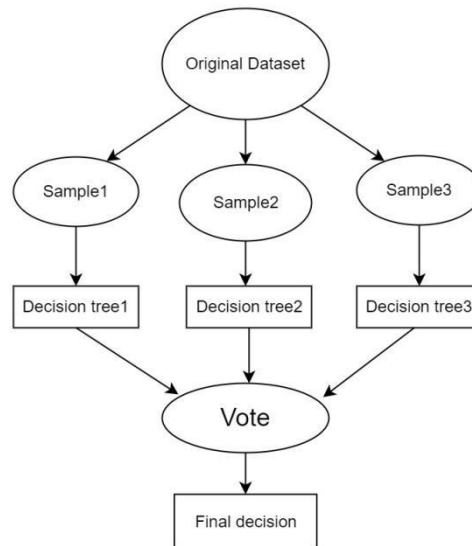


Figure 5. Building Process of Random Forest

This is how Random Forest works:

Choose N samples at random from the training sample, and train with these N samples yields a decision tree which serves as a sample at the node of the decision root.

Randomly select m features when each node of the decision tree needs to be split, and adopt a certain criterion (such as information entropy) to select one feature which gains the majority votes as the split feature of the node.

Repeat the procedures mentioned above until the features can not be split which represent the leaf nodes.

The advantages of Random Forest:

- (1) The results of classification task are more accurate.
- (2) It can handle the attributes of high dimensions, and avoid making feature selection.
- (3) Even if a large part of the data is lost, a high degree of accuracy can still be maintained.
- (4) The training efficiency is better than that of bagging.
- (5) Randomization only needs to consider a subset of attributes, and bagging needs to examine all attributes.
- (6) Easy to realize parallelization computing.
- (7) The interaction between features can be collected during training during training.

The disadvantages of Random Forest:

(1) Since it does not produce consecutive outputs, random forest does not has a great performance when addressing the regression task. When regression is conducted, random forests are unable to predict outcomes outside the bounds of the training set, which may result in over-fitting when modeling some specific noise data.

(2) For small data or low-dimensional data (less characterized data), it might not result in accurate classification since random forests are better at processing high-dimensional data, missing feature data, and unbalanced data.

2.3.2 Training

2.3.2.1 Fitting procedure

The supervised learning technique, which involves passing two parameters via fit() function, can be used to train the model using sample test set and sample training set that have been separated

during data format processing. The model's score() method is then invoked to obtain the model score. Similar to this, the predict() function can be used to forecast the outcomes of data training in order to assess the effectiveness of the regression model's fitting.

2.3.2.2 Parameters modification—— Grid Search Algorithm (GridSearchCV)

When building random forest model, the grid search algorithm is implemented to automatically select the best parameters (such as n_estimators). Grid Search is a method of executing an exhaustive search in which every possible parameter combination is tested, with the best performing parameter being the end result. GridSearch CV is a common approach of parameter evaluation that combines grid search and cross-validation. As a result, Scikit-Learn built a function called GridSearchCV in specifically for model auto tuning.

Table 4. The best parameters of Random Forest regression model

Parameter	Description	Value
n_estimators	The amount of decision trees	200
max_depth	Decision tree maximum depth	15
min_sample_split	Minimum number of samples required to split an internal node	5
min_sample_leaf	Minimum number of samples required per leaf node	5
criterion	A method to evaluate the quality of decision tree's branch	MSE

2.4. XGBoost regression model

2.4.1 Algorithm

As a kind of integrated learning, XGBoost is also composed of many decision trees, but it is different from random forest in many ways. XGBoost mainly trains the next tree on the basis of training the previous tree. By predicting the gap between the tree and the real distribution, it continuously trains the trees used to bridge the gap, in order to finally realize the simulation of the real distribution by the combination of trees. XGBoost's approach of using one model to rely on the previous model to approach the correct answers together is called Boosting.

The process of training a model is usually to define an objective function and then optimize it further. XGBoost's objective function consists of a loss function and a regular term. The loss function represents the degree of model fitting data, and its first derivative is usually used to indicate the direction of gradient descent[9]. XGBoost also calculates its second derivative, which further considers the trend of gradient change, so the fitting is faster and the accuracy is higher. The regular term is used to control the complexity of the model. The more leaves there are, the bigger the model will be. Not only will the operation time be long, but the classification effect will be reduced when the operation time exceeds a certain limit. The regular term of XGBoost is a penalty mechanism, the more number of leaf nodes, the greater the penalty. Thus limiting the number of leaf nodes.

$$\text{Objective function: } L(t) = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (1)$$

The biggest improvement in XGBoost outside of math is that it dramatically increases the speed of computation. The most time-consuming part of tree construction is the sorting of feature values to determine the optimal split point, whereas XGBoost sorts features before training and stores them in Block structures, which are reused later to reduce computation.

The advantages of XGBoost:

- (1) Process a strong ability to capture relationships between complicated data and the capacity to create efficient models from massive data sets.
- (2) Support a variety of systems (Linux, Win, macOS...)

(3) Support multiple languages (C++, Java, Python, R...)

The disadvantages of XGBoost is that it does not perform well on high dimensional sparse feature data sets and small scale data sets.

2.4.2 Training

2.4.2.1 Parameters modification

Based on the detailed description and range Settings of XGBoost parameters, XGBoost is more of a manual callback within a limited range. By taking special values and many experiments, the parameters are set as follows:

Table 5. The best parameters of XGBoost regression model

Parameter	Description	Value
seed	random number seed	40
lr	(alias: learning_rate) must be set to 1 when training random forest regression	0.01
n_estimators	the amount of decision trees	900
colsample_bytree	the ratio of subsample columns used to build each tree	0.8
max_depth	maximum depth of a tree	10
reg_alpha	L1 regularization term on weights	0.01
reg_lambda	L2 regularization term on weights	0.6
subsample	subsample ratio of the training instances	0.5

2.4.2.2 Feature Importance

The next table displays the analysis findings for the feature importance utilising well-known machine learning algorithms as Random Forest Classifier and XGBoost Classifier.[10]. Each input is given a value using feature importance techniques based on its relative importance. Thus, the top three important features in XGBoost regressor model are total_level, building_year and level. These not only show the correlation between the independent variables and the target variables, but also serve as the basis for further optimization of the callback combination.

Table 6. Feature importance ranking by XGBoost Model

Feature	Relative importance
total_level	0.2194
building_year	0.2002
level	0.1584
room	0.1299
area	0.1132
decoration	0.0748
elevator	0.066
livingroom	0.0381

3. Results and Discussion

3.1. Evaluation metrics

The evaluation indexes of the performance of the regression model mainly include: MAE (average absolute error), MSE (mean squared error), and R^2_score (coefficient of determination).

MAE: Mean-squared error is the most common loss function in linear regression. MSE can evaluate the degree of change in the data, and the smaller the value of MSE indicates that the prediction model describes the experimental data with better accuracy.

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i| \quad (2)$$

MSE: The average absolute error is the square of the difference between the true value and the predicted value, and then the average of the sum value, which is generally used to detect the deviation between the predicted value and the true value of the model.

$$MSE = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (3)$$

R^2_score : The coefficient of determination, which reflects the accuracy of the model fitting data, generally R^2 ranges from 0 to 1. The closer the value is to 1, the better the variable of the equation explains y, and this model fits well to the data.

$$R^2_score = 1 - \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{\sum_{i=1}^n (\bar{y}_i - y_i)^2} \quad (4)$$

3.2. Comparison results

Table 7. The performance of Random Forest regressor model and XGBoost model

	MSE	MAE	R^2_score
Random Forest	2802	534	0.89
XGBoost	3108	643	0.87

According to the difference of importance of each index in random forest model and XGBoost model, the predicted values can be obtained, and the predicted data and the real data can be substituted into the formula of MSE, MAE and R^2_score to obtain the values of MSE, MAE and R^2_score shown in Table 7. The Random Forest model obtains MSE and MAE of 2802 and 534, respectively, while the XGBoost model obtains these two figures of 3108 and 643, respectively. The R^2_score obtained from the random forest is 0.89, while the R^2_score of XGBoost is 0.87. In terms of all three indicators, the prediction accuracy of the Random Forest model is greater than that of XGBoost[11], which shows that the Random Forest model has better precision and is more suitable as a training model.

A system for housing price forecasting have been developed. The system makes optimal use of Forest regression and XGBoost Regressor. The random forest model outperforms the XGBoost model, primarily in R^2_score (approximately 89 percent than with 87 percent for XGBoost). Their precision was proven to be greater than 80%, demonstrating the excellent performance of both models when used to housing price prediction tasks. By using Matplotlib, a two-dimensional graphics library for Python, the predicted values of the two models are compared with the actual values to create a line graph of the difference between the predicted and actual values for the random forest model and the XGBoost model presented in Figure 6 and Figure 7.

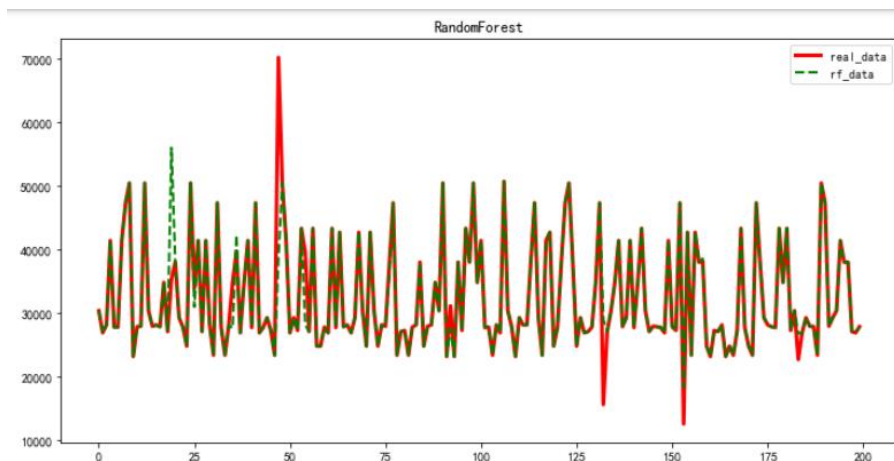


Figure 6. The data comparison between the real data vs prediction data of the random forest model

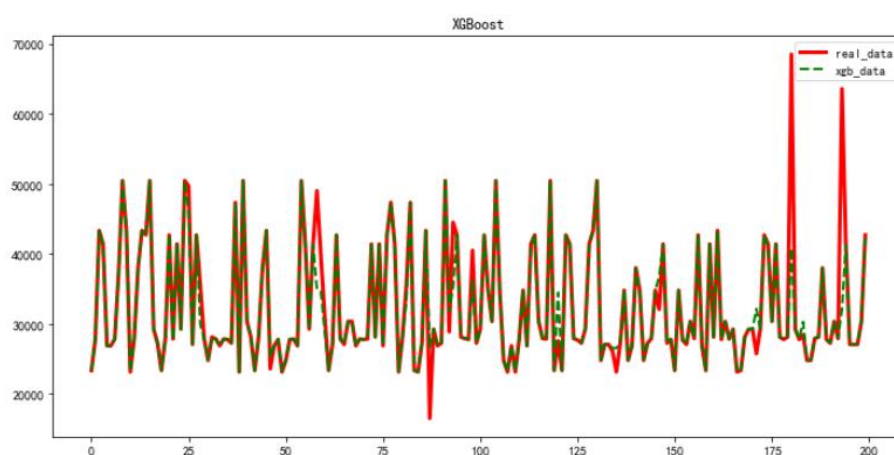


Figure 7. The data comparison between the real data vs prediction data of the XGBoost model

As the graph shows, both the Random Forest model and the XGBoost model perform accurately when analyzing house prices between 20,000 and 50,000 per square meter, with the predicted values being close to the actual values. However, the Random Forest model outperforms the XGBoost model when dealing with house prices per square meter between 10,000 and 20,000 and above 50,000, which shows that the Random Forest model has better performance when dealing with extreme values. The model can be improved in the future by adding more cities to the database and increasing the number of indices used to predict house prices in order to get more accurate predictions.

4. Conclusion

This paper analyzed the overall house price trend during the COVID-19 pandemic from 2017 to 2022 in China and predicted the house price in Guangzhou in 2022. This paper mainly focused on comparing the performances between two advanced machine learning regression models Random Forest and XGBoost on house price prediction. The results show that though two models have achieved excellent prediction accuracy as well as high coefficient determination score. The random forest regression forecast model's MSE score is 2802, the MAE score is 534, and the R^2 _score is 0.89. The XGBoost regression forecast model's MSE score is 3108, the MAE score is 643, and the R^2 _score is 0.87. Random Forest scores slightly higher than XGBoost on all the evaluation indices. It is because the data sample used in this paper for model training is not balanced since the 'unit_price' feature is largely distributed in the range of 30000 and 40000 while Random Forest is well known for dealing with extremely unbalanced data sets. As a result, though XGBoost is acknowledged superior in handling larger sets of data and has more flexible parameters to modify, Random Forest performs somehow more efficiently and accurately in specific regression tasks. In future work, in order to

acquire more objective and general conclusions about two models, different quantities of data sets will be trained to increase the control groups. At the same time, balanced data sets will be applied to test other advanced functions of two models including dealing with insufficient features and so on. Moreover, the parameters set in this paper were modified manually, whose effects were more directly perceived than auto tuning. However, hyper parameter tuning method will be applied in the future work to reduce the deviation and achieve better results in the task of predicting house price.

References

- [1] Chen, B., Yang, X., & Zhong, N. (2020). Housing demand and household saving rates in China: Evidence from a housing reform. *Journal of Housing Economics*, 49, 101693. <https://doi.org/10.1016/j.jhe.2020.101693>
- [2] Zhou, Y., "The Impact of US Subprime Mortgage Crisis on the Chinese Real-estate Market" (2013). *Applied Economics Theses*. 3. https://digitalcommons.buffalostate.edu/economics_theses/3
- [3] Wong, C. (2011), "The Fiscal Stimulus Programme and Public Governance Issues in China", *OECD Journal on Budgeting*, Vol. 11/3. <http://dx.doi.org/10.1787/budget-11-5kg3nhljqrjl>
- [4] Eftimoski, M., & McLoughlin, K. (2019, March 21). Housing policy and economic growth in China: Bulletin-March Quarter 2019. Retrieved July 8, 2022, from <https://www.rba.gov.au/publications/bulletin/2019/mar/housing-policy-and-economic-growth-in-china.html>
- [5] Bloomberg Businessweek. (2022, June 22). China's Property Slump Is a Bigger Threat Than Its Lockdowns. Retrieved July 8, 2022, from <https://www.bloomberg.com/news/articles/2022-06-22/china-housing-market-slowdown-drags-economy>
- [6] Cheung, K. S., Yiu, C. Y., & Xiong, C. (2021). Housing Market in the Time of Pandemic: A Price Gradient Analysis from the COVID-19 Epicentre in China. *Journal of Risk and Financial Management*, 14(3), 108. MDPI AG. Retrieved from <http://dx.doi.org/10.3390/jrfm14030108>
- [7] He, L. (2022, May 20). China slashes key interest rate as housing sales collapse. CNN. Retrieved July 15, 2022, from <https://www.cnn.com/2022/05/20/economy/china-record-lpr-cut-property-market-intl-hnk/index.html>
- [8] Strobl, C., Malley, J., & Tutz, G. (2009). An introduction to recursive partitioning: Rationale, application, and characteristics of classification and regression trees, bagging, and random forests. *Psychological Methods*, 14(4), 323–348. <https://doi.org/10.1037/a0016973>
- [9] Anders, H., Johan, P., Ida, S. & Dag, E.S. (2022) House price prediction with gradient boosted trees under different loss functions, *Journal of Property Research*, DOI: 10.1080/09599916.2022.2070525
- [10] Jha, S.B., Babiceanu, R.F., Pandey, V., & Jha, R. (2020). Housing Market Prediction Problem using Different Machine Learning Algorithms: A Case Study. *ArXiv*, abs/2006.10092
- [11] Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), 1247–1250. <https://doi.org/10.5194/gmd-7-1247-2014>