Prediction of Taipei House Price with Stacked Generalization

A study submitted in partial fulfilment
of the requirements for the degree of
MSc Data Science

at

THE UNIVERSITY OF SHEFFIELD

by

Xuan-Yi Fu
200226767

Word-length:10283 August 2021

Table of content

Abstract

Background House prices in Taipei has been increasing and less affordable to home buyers. The issue has been raised by government and society, and it is essential for the home buyers to know the real value of the house. The prediction of house prices can make the real estate market more transparent and home buyers could know the real value of the properties. Therefore, the methods are conducted in this study to better improve the house prices prediction in Taipei.

Aims The main purpose of this study is to compare stacked generalization with other single and ensemble methods to see if the stacked generalization does improve the prediction. In addition, the performance of single methods is compared to that of ensemble methods when they are applied in the first stage of stacked generalization which could lead to better understanding of the methods used in first stage of stacking.

Methods The data used in this study is retrieved from Kaggle platform. It is pre-processed procedures including missing values, feature engineering and translation. Some further pre-processing procedures are conducted in this study such as removing outliers and label encoding. There are six methods applied individually to compare with stacking firstly. Three single methods are Linear regression, Lasso and Ridge. Three ensemble methods are LightGBM, Random Forest and XGBoost. After that, single methods and ensemble methods are applied in the first stage of stacked generalization to compare the performance. Finally, MAE, RMSE and R^2 are conducted to evaluate the models.

Results The best performance among all the models is LightGBM which the MAE is 30102.05, the RMSE is 41818.69, and the R^2 is 0.62. All the ensemble methods outperform the single methods and the stacking models. The stacking models includes all six methods showed the best performance.

Conclusion Stacked generalization do improve the models when the first stage are all single methods. However, when the stacking method includes all ensemble methods in the first stage, it does not improve the house price prediction in Taipei.

Acknowledgement

I would like to express my appreciation to Dr. Susan Oman for her guidance on this dissertation. Susan always provides very useful and clear information and feedback which leads me to better constructing the dissertation.

I also want to thank all the lecturers and module coordinators in this programme for the knowledge I learnt from these modules have been very useful to this dissertation.

Finally, I would like to make a special acknowledgment to the media in Taiwan which has raised the issue of high house prices in Taipei. This has given me an opportunity to investigate on this topic.

1. Introduction

This section first describes the background of house prices in Taipei. Secondly, the influences of house price fluctuation will be discussed. Thirdly, the reasons for the increasing house prices in Taipei and the government's corresponding solutions to it will be introduced. After that, the benefit of predicting house prices will be illustrated. Finally, the methods used in this study to predict house prices in Taipei will be explained.

1.1. Background of house prices in Taipei

People buy houses for home or for investment. Homeownership can have an impact on the subjective well-being of human beings. In fact, research from China indicates that homeownership is positively associated with one's happiness, and it provides an increased sense of security to people (Hu & Ye, 2020). Hence, it could explain the continuous growth of homeownership in Taiwan in recent decades (Li, 2002). Real estate investment is another purpose of buying houses in Taipei. House prices increase along with the rise of land value (Paris, 2009). Taiwan Ministry of the interior released that Taipei house prices have increased by nearly 50% in the past 10 years. Many investors seize the business opportunities of the increasing real estate in Taipei and made great profits from the investment. However, the increasing house prices in Taipei are making homeownership more unaffordable. The property age and condition, location, home size and public transport are all influential factors to the house prices. Generally, the house owner or the salesperson offer a price and buyers will decide whether to accept it. However, there are no reliable sources to help the buyers identify if the price is reasonable. House price prediction with machine learning can be one solution.

1.2. Problems and solutions

In this section, the cause and effect of the skyrocketing house prices in Taipei and solutions to it will be discussed. The souring Taipei house prices have become a great challenge in society. Taipei is the economic center of Taiwan which offers many great opportunities and attracts people from other cities, however, the unaffordable high house prices stand in their way. The average mortgage rate has been decreasing in the past 10 years in Taiwan, and Taiwan central bank has released that it has reached rock bottom to 1.3% in March 2021(Central bank of Republic of China, 2021). Apart from the inflation of the building materials, mortgage and carrying costs are two causes often raised. Research has shown that low mortgage rates are responsible for the soaring house prices in Taipei city (Yu & Chen, 2018). Another reason for the high house prices is that the cost of carrying a house in Taiwan is extremely low. When the cost of carrying a house is low, people tend to carry it and wait for a good price to sell which is considered an investment. When those wealthy people possess many houses at the same time, the demand for the houses increases which than lead to the rise of house price. Imagine properties with the same condition in the U.S. and Taipei, the carrying costs of the former could be up to 48 times more than the latter. According to the Taiwan Ministry of the interior, 10% of the houses in Taiwan are owned by those who carry more than four properties (Ministry of the interior, 2021). This indicates that many houses in Taiwan are not owner-occupied properties. Taiwan’s government has practiced policies to decrease the chance of flipping properties. One of the most famous policies is Actual Price Registration of Real Estate which allows the prices of the houses to be more transparent. The actual price registration policy has led to a decrease in house prices by 4% to 29 % (Yu & Chen, 2018). Another policy is to tax properties that are not lived by the owner, so those non-self-use properties will have a higher tax rate. Finally, Taiwan’s government tax on those properties being traded shortly after the previous transaction is not only a disadvantage for the investors but create an opportunity for buyers who will live in the house. Machine learning can provide valuable information and has been applied to support decision-making. House price prediction can assist the policymakers to understand the trend of housing prices and thus make policies accordingly. AI has been applied to support policy-making such as in health and education. (Allam et. al, 2020) (Gulson & Webb, 2017), and has a successful contribution. Although there is increasing concern about AI making unethical decisions (Dellinger 2015), it is suggested that the issue could be improved through some ethics bots and legislation (Mars storm, AI ethics and the LHC's big upgrade, 2018) (Etzioni & Etzioni, 2016). Improving the prediction of house prices in Taipei allows the government to analyze the economic situation and assist in policymaking, such as increasing homeownership and reducing house price rises. From the buyer’s perspective, the prediction made by AI provides supporting information to the buyers which allow the buyers to analyze and make decisions accordingly (Allal-Chérif, 2021). The prediction of house prices with higher accuracy provides them more reliable information which helps them make decisions on the real estate investment.

1.3. House price prediction and stacked generalization

House price prediction can be accomplished by machine learning methods. Machine learning methods learn from the property data including lot size, utilities, location, number of bathrooms, parking lot and property condition to do the prediction. The target is to predict the price of the house, which is continuous data, therefore, it belongs to regression. After the model is built, the evaluation methods are used to examine the performance of each model. MAE (Mean Absolute Error) is often used on evaluation which represents the difference between the real value and the predicted value. Many methods can be used to do the house price prediction such as regression-based methods, SVR and deep learning. The ensemble method is to combine many weak learners to get better results. There are three kinds of ensemble methods such as stacking, boosting, and bagging, and the one applied in this dissertation is stacking. Stacked generalization also named stacking is an ensemble method introduced by David Wolpert in 1992, which allows a meta-learner to learn from different learners. It first trains some learners, and then uses the results that these learners generated as the input of the meta-learner and train the new model to reduce biases. Stacked generalization is a method that estimates and corrects the errors of the provided learning set and thus make the prediction more accurate (Wolpert, 1992). Stacked generalization has been applied to house price prediction in many cities such as Beijing, NYC and Melbourne and is proven to outperform many other methods (Zhang, et al., 2020) (Truong et al., 2020) (Xiong, Sun & Zhou, 2020). Research have applied methods such as regression, SVR, MLP and LSTM to the prediction of Taipei house pricing (Lee & Chen, 2008) (Lin & Chen, 2019). However, stacked generalization has not yet been used in predicting house prices in Taipei. Therefore, using the method on Taipei real estate dataset and compare the performance could help people better understand the approaches used on the prediction of house pricing. Although stacking does not always provide best performance (Ribeiro& dos Santos Coelho, 2020)

(Dou et al., 2020), it is a good chance to have further understanding of ensemble methods used on Taipei house prices prediction through this research.

1.4. Aims and objectives

To use stacked generalization methods on the prediction of house pricing in Taipei and compare with single methods to see if stacked generalization is indeed a better method to be used on the prediction of house pricing in Taipei. The findings can provide a further understanding of the methods used on house price prediction and their performance and can thus lead to more reliable prediction model. With such model, the home buyers could know the reasonable house price in Taipei.

1.5. Research questions

This dissertation will provide answers to the following research questions:

1. What is the performance of stacked generalization on prediction of house pricing?
2. Does stacked generalization outperform other single methods on the prediction of house pricing in Taipei?
3. If the ensemble methods are used in the first-level learner, will it outperform the model that uses single methods as first-level learner?

1.6. Structure of dissertation

This dissertation is presented in six sections. The first section, the introduction, provides background regarding house price prediction, cause, effect, and solutions to soaring house prices in Taipei, the methods used on house price prediction and the gaps between the literature. Section two is a literature review that begins with a discussion of house prices and society such as economy, policy and labour market. After that, some relevant literature regarding prices prediction, machine learning theory and ML methods is presented. Section three describes the process of the methodology used in this research. Furthermore, it explains the use and the sources of the dataset. Section four shows the results of the prediction on the house prices with different ML methods. Section five first discusses the results of the research and answers the research questions. After that, it provides some guidance for future research. In the end, the final section concludes the research findings.

2. Literature Review

2.1 House price and society

2.1.1 Economy

Real estate investment contributes significantly to the national economy. Research demonstrated that house prices and consumption are strongly correlated (Benjamin et al., 2004). Rising house prices urge people to spend more money which leads to consumption growth, and thus boost economic development (Miller et al., 2011). The supply of houses is also influential to house prices and can thus have an impact on the economy (Goodhart & Hofmann, 2008). The fluctuation of house prices is one of the key economic factors and has brought policy makers attention (Chandler & Disney, 2014). Moreover, Chandler also demonstrated that unstable house prices could lead to economic and financial instability (Chandler & Disney, 2014). Moreover, house prices are significantly positively correlated with national income. (Kraft & Munk, 2011). Due to the considerable impact house prices bring to the economy, it is important to pay attention to the stability of it. Research around the world has shown the influence that house price brought to the economy. Housing wealth has been proved to have influences on the U.S. economy (Bostic et al., 2009). The data from Eurostat also shows that real estate contributed approximately ten per cent to the European economy in 2014.

2.1.2 Policy

House price and national policy have interacted with each other. On one hand, policymaking led to the rise of house prices. On the other hand, policies are made to prevent the rapid increase in house prices in Taipei. Research demonstrated that monetary policy has an intermediate influence on real estate prices. More precisely, expansionary monetary policy causes the rising of house prices (Xu & Chen, 2012). Research has shown that low mortgage rates were responsible for the soaring house prices in Taipei city (Yu & Chen, 2018), while the actual price registration has made the real estate transaction more transparent and fairer. As a result, buyers can take it as a reference to avoid overpriced properties. In fact, the actual price registration policy in Taiwan has been acknowledged to lead to a decrease of house prices by 4% to 29% (Yu & Chen, 2018) which provided another supportive argument that policies have an impact on house prices.

2.1.3 Labour market

The labour market is also involved in the impact that house prices bring to society. Research by Johnes and Hyclak (1999) demonstrated that house prices are significantly influenced by the labour market and employment rate. Even though Gathergood (2011) illustrated that the rise of the unemployment rate had reduced people's willingness to purchase houses, his research did not find a strong relationship between house prices and the unemployment rate. Apart from the direct impact that the labour market had on the real estate market mentioned above, it also brought indirect effects. The rise of unemployment increases the chances of residential mobility. Actually, research showed that residential mobility increased the demand for houses, and thus influenced the real estate market (Nijkamp et al., 2002) (Dieleman et al., 2000). Evidently, house prices are indeed influenced by the labour market. Therefore, maintaining the stability of house prices can benefit society. Irandoust (2019) stated that the economic and financial stability led by a stable employment rate can benefit from stable house prices.

2.2 Previous work on house prices prediction

2.2.1 House price prediction with machine learning of Taipei

Some research has contributed to the Taipei house prices prediction. Lin & Chen used machine learning methods to the Taipei house dataset for prediction in 2019. They applied linear regression, MLP (Multi-layer perceptron) and LSTM (Long Short-Term Memory) methods on the actual price registration dataset separately and evaluated the models with FastDTW, loss function, MSE and R squared. The result showed that LSTM outperformed the other methods (Lin & Chen, 2019).

Another research by Lee and Chen retrieved data from the Taiwan government official database to do the prediction. In their research, Support Vector Regression (SVR) and Adaptive Network-Based Fuzzy Inference System (ANFIS) were the two methods applied to do the prediction. They compared the performance of the two techniques through MAPE (Mean absolute percentage error) and R squared to do the evaluation (Lee & Chen, 2011).

One special research by Chen et al. used the classification method to forecast house pricing. The only technique used in this research is SVM which aimed to find the decision boundary for classification. The dataset used in this research is from Gigahouse Taiwan's Real Estate Portal data, which is a property website in Taiwan. In order to apply classifiers to numerical prediction, Chen et al. binned the prices into 56 classes. Finally, they evaluated the models with "hit-rate" also called "true positive rate". SVM had shown very high performance that the prediction of some districts in the suburbs reached 82% hit-rate (Chen et al., 2017).

The research has applied many methods on Taipei house price data, including regression-based methods, neural network techniques and classification methods. However, no ensemble methods have been applied to the Taipei house price dataset yet.

2.2.2 House price prediction with stacked generalization

Stacked generalization has been applied in many cities such as Beijing, Melbourne and New York City. The research by Truong et al. used five techniques, Random Forest, XGBoost, LightGBM, Hybrid Regression and Stacking to predict the house price in Beijing. The stacking model being used included Random Forest and LightGBM as the first stage learner and XGBoost as the meta-learner. The evaluation used on this was RMSLE (Root mean squared log error). The five models are compared, and the stacking method had the lowest RMSEL and thus showed the best performance among all. The research showed that the stacking ensemble method outperformed the other single methods. However, the research did not mention why XGBoost was chosen to be the meta-learner while Hybrid regression and Random Forest both showed better performance in the individual model examination than XGBoost (Truong et al., 2020).

Xiong, Sun and Zhou used different combinations of six methods, Lasso, Extra Tree, ElasticNet, XGBoost and GDBT to do the prediction on the house prices in Melbourne. The first stage learners include two to five out of the six methods, and the meta-learner is chosen from one of those which is not included in the first stage. The result showed that the best model is composed of Extra Tree, GDBT, XGBoost, Random Forest and ElasticNet, and the meta-learner is Lasso. Despite the fact that the model outperformed all the others, a high-efficiency ensemble method used on the meta-learner may lead to a better result (Xiong et al., 2020).

The research of New York house price prediction applied Lasso, Ridge, Linear Regression, Adaboost, ElasticNet, Random Forest and Gradient Boosting Machine as the first stage learner and Lasso as the meta-learner (Zhang et al., 2020). It first evaluated all the methods individually and found that Ridge, Random Forest and Gradient Boosting Machine showed better results among all of the seven methods. Therefore, they compared the high-performance selection (Ridge, Random Forest and Gradient Boosting Machine) with a model that includes all of the seven in the first stage. The result showed that the one with high-performance selection outperformed the other. As a result, the more techniques included in the first stage does not mean the better.

2.3 Machine learning

2.3.1 What is machine learning

Machine learning is a part of the computer science field, and it allows machines to learn from the data and conduct predictions accordingly. It provides automation solutions to difficult problems through algorithms and advanced techniques (Rebala et al., 2019). According to the IBM data scientists from IBM, approximately 2.5 quintillion bytes of data were generated each day last year. We are in the era of big data where the amount of data is big, and the velocity is fast. Therefore, we need fast and smart enough solutions to analyze the data. In fact, machine learning is able to provide such solutions (Murphy, 2012). The purpose of machine learning is to conduct forecasting through analyzing the past data and find the best model to explain the dataset (Murphy, 2012), and extracting valuable insights (Kelleher et al., 2015). The best model is the one that provides the best answers to the user's questions, which can be examined through a validation process. There are two steps in the machine learning process. When the data is ready for analysis, it is split into a training set and test set. The first step of machine learning is to train the model with a training set. After the model has learned from the training data, it is applied to the test set and does the prediction (Kelleher et al., 2015). Machine learning has been applied in many fields. It filters those spam in the mailbox through classification. It could be used to do object detection, which can identify faces in pictures. It is used on the crediting system which tells if the customers are high risk. The stock price prediction can also be done through machine learning. In addition, ML is applied to support medical diagnosis through analyzing patient’s data.

2.3.2 Supervised and unsupervised learning

There are two types of machine learning known as “supervised” and “unsupervised”. The difference between these two is the dataset given to the learning algorithm. The labelled dataset indicates that the dataset has included answers in it. Supervised learning uses a labelled dataset to do the training, and it learns from all the features including the prediction target (Rebala et al., 2019). Another feature of supervised learning is that the defined categories are provided in advance. In other words, the supervised learning algorithms know the categories of each data and it aims to find the rule of the classification. The prediction of the survival of Titanic belongs to supervised learning. The dataset of the passengers includes a column named “survived” which indicates if the passenger survived the Titanic sinking, and the learning algorithms learn from the dataset to generate a prediction model. And then, when new data is fed to the model, it would be able to give the right answer. Furthermore, supervised learning can be divided into two categories according to the type of the predicting target, classification for categorical prediction and regression for numerical prediction. On the other hand, the dataset used on unsupervised learning is unlabelled, which means that the algorithms do not know the correct answers. Even though no prediction target or defined categories are provided to unsupervised learning, the algorithms learn from the features to seek the similarity trends and divide the data into groups according to the homogeneity (Rebala et al., 2019). For example, scientists apply unsupervised learning to cluster DNA sequences (James et al., 2018).

2.3.3 What is ensemble method

As the old saying goes, “Two heads are better than one.” Ensemble methods combine many base learning algorithms such as SVM, decision tree, and neutral network together to make precise predictions (Zhou, 2012). The concept of ensemble methods is to improve accuracy through combining methods, and it is usually better than any of its elements (Seni & Elder, 2010). Boosting is an approach of sequential ensemble which contains a group of non-parallel learners. Each learner learns from errors made by the previous learner and the weights of those base learners which have lower error rate are added to improve the new model. Bagging is another often seen approach of ensemble methods. The word “bagging” contains the concept of bootstrap and aggregation. It includes a group of parallel learners, and the learners vote(classification) or average(regression) to get the final result. The voting and the averaging make the models less noisy which then improves the robustness (Shaikhina et al., 2019). Finally, stacking was introduced by Wolpert in 1992. In stacking, a group of base learners are combined as the first-level learner. The first-level learners generate an output dataset which contains new insights, and then the meta-learner uses the new output as the training data to train the model and do the prediction.

Ensemble methods are applied in many fields and have shown better performance than single methods. Ren et al. applied ensemble methods to predict wind and solar power, and ensemble methods outperformed the single methods. Liu et al. predicts the BBB permeability of compounds using ensemble methods, and the performance of ensemble methods are better than most of the basic classifiers. King et al. uses ensemble methods to predict advanced skier days and the error that ensemble methods made are lower than single methods. To sum up, the performance of the ensemble prediction methods and the results have shown that the ensemble methods in general have outperformed the non-ensemble methods.

Stacked generalization

As the previous section mentioned, stacked generalization is a method used to improve the model by combining many base learners output as the input of the meta learner. The base learners in the first stage do the prediction and feed their guesses to the meta learner in the second stage as figure 2.1 shows. After that, the meta learner used the data to do the prediction. In fact, the meta learner improves the model through correcting the errors and reducing the biases made by the first stage learner (Wolpert, 1992).

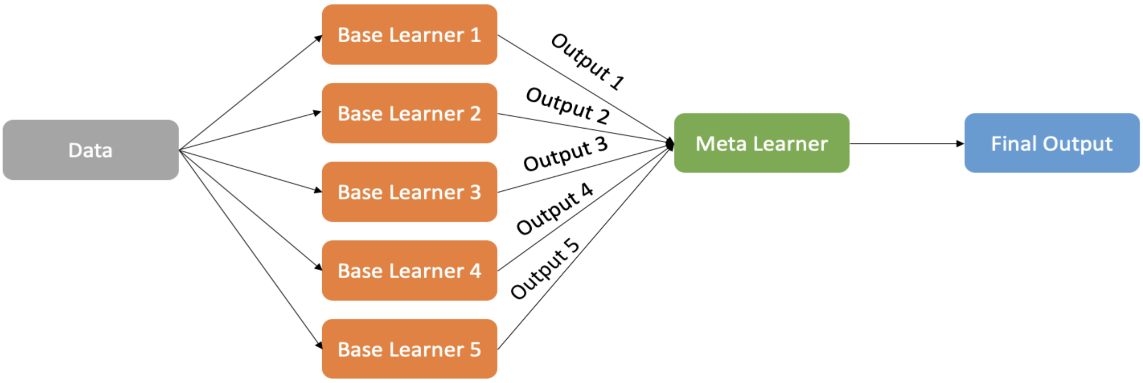

Figure 2.1 – Stacking

3. Methodology

In this section, the process used for the analytics will be introduced. After that, each process will be explained in detail. CRISP-DM is a process widely used in predictive data analytics (Kelleher, 2015), which will be implemented in this dissertation. The six processes of CRISP-DM are “Business understanding”, “Data understanding”, “Data preparation”, “Modeling”, “Evaluation” and “Deployment”. The final process will not be necessary for this dissertation, so there are only five steps. Firstly, the process of “Business understanding” is to find out what is to be addressed, in this case the prediction of house price is the main goal. To accomplish the goal, used data is essential. Therefore, the data collection and the origin of the dataset will be discussed and explained. Secondly, the data understanding process is to understand the characteristics of the features and to check if these characteristics will cause any quality issues which could lead to bad influence on the models (Kelleher, 2015). Thirdly, data preparation will be applied to deal with the issues found in the previous process, such as missing values or outliers. And then, the models used for the prediction are built based on different machine learning algorithms. After that, methods are applied to evaluate the models and compare the performance. Finally, the risk assessment and ethical issues is demonstrated and clarified.

3.1 Research tools

The main tool used in this study is Python 3. Python is a programming language that can be applied in all the main operating systems, and it has been applied in all sizes of projects (Pittard&Li, 2020). There are many powerful Python libraries that can satisfy the need of data science process and will be used in this study such as "Matplotlib" and "seaborn" for data visualization, "numpy" and "pandas" for data structures, and finally, "sklearn" for machine learning. The Integrated Development Environment (IDE) used here is PyCharm which is a powerful and functional software developed by Jetbrains. It is a software that supports the code editing, compiling, deploying, and debugging and makes the development more efficient. Furthermore, Knime is also applied to support data visualization, generating plots for data exploration.

3.2 Data collection and business understanding

The dataset used in this study is collected from Kaggle, which originated from Taiwan government open data platform. Both platforms are free and open sources. Actual price registration policy has been executed by the Taiwan government since 2012, and all the real estate transactions are registered in Taiwan Government's databases with the actual price of the houses. Additionally, all the transactions are available at the Taiwan government open data platform at https://plvr.land.moi.gov.tw/DownloadOpenData. However, most of the data is in Chinese and the format is not well processed or organized. Fortunately, Peiyuan Chieng has released processed Taipei actual prices transaction data in Kaggle website at https://www.kaggle.com/chrischien17/taiwan-taipei-city-real-estate-transaction-records, which has translated all information into English and dealt with all missing values. Other pre-processing procedures are also completed such as removing those columns with 90% missing value, “non_urban_use”, “non_urban_use_code”, “note” and “serial_number”. The missing values of “total_levels” are filled with “0” because there are no shifting levels for land and car park transactions. The column "transaction_number" which is recorded as “土地 1 建物 0 車位 0” contains three pieces of information, "no. of lands", "no. of buildings" and "no. of car parks". These messages are split into three columns separately, {"土地"：1, "建物": 0, "車位": 0} which makes it more understandable. Chieng also transfers civil year to A.D. for the “transaction_year” column.

In this dataset, there are two attributes representing house price, “total_ntd” and “unit_ntd”. The previous one indicates the total price of the property, and the latter represents the price per unit. In Taiwan, the unit price is more often emphasized, therefore, “unit_ntd” will be used as the target attribute for the models to predict.

3.3 Getting to know data

Data table

Feature	Datatype	Variable type	Note
district	object	Categorical	12 administrative districts in Taipei City
transaction_type	object	Categorical	land, building or carpark
land_shift_area	float64	Numerical	shifted land area in a transaction (square meter)
urban_land_use	object	Categorical	registered land use
main_use	object	Categorical	registered property use
main_building_material	object	Categorical	concrete, wood, stone…
complete_year	float64	Numerical	year when the building is completed (in civil year)
building_shift_total_area	float64	Numerical	shifted building area (square meter)
num_room	int64	Numerical	Number of rooms
num_hall	int64	Numerical	Number of halls
num_toilet	int64	Numerical	Number of toilets
total_ntd	int64	Numerical	total price in New Taiwan Dollar
unit_ntd	float64	Numerical	price per square meter (in NTD)
carpark_category	object	Categorical	type of car park
carpark_shift_area	float64	Numerical	Car park area
carpark_ntd	int64	Numerical	price of car park in NTD
transaction_year	int64	Numerical	Transaction year
transaction_month	int64	Numerical	Transaction month
building_age	float64	Numerical	age of the building
number_of_land	int64	Numerical	number of lands in a transaction
number_of_building	int64	Numerical	number of buildings in a transaction
number_of_carpark	int64	Numerical	number of carparks in a transaction

First, there are 5909 rows, which means that there are 5909 transactions in the dataset. There are twenty-two features in the dataset, 6 categorical and 16 numerical variables. There are many ways to see the information that each feature brings. Kelleher (2015) suggested to use bar plots to see the frequencies of categorical data and use mean and standard deviation to understand numerical data. Through the process of exploring the data, the issues and problems are found and can thus be dealt with in the next step. In addition, the relationship between each feature and target feature is also explored in this step and could provide information to support feature selection process. What is worth mentioning is that, although transaction month is int64 data type, it will be discussed in the bar plot to see how transactions appear in each month. In the following section, the single attribute exploration and the correlation with target attribute will be discussed.

Figure 3.1 shows the number of transactions in each district during 2012 to 2021 in Taipei city. As the figure shows, Zhongshan district and Neihu district have the most transactions, and Nangang district has the fewest transactions.

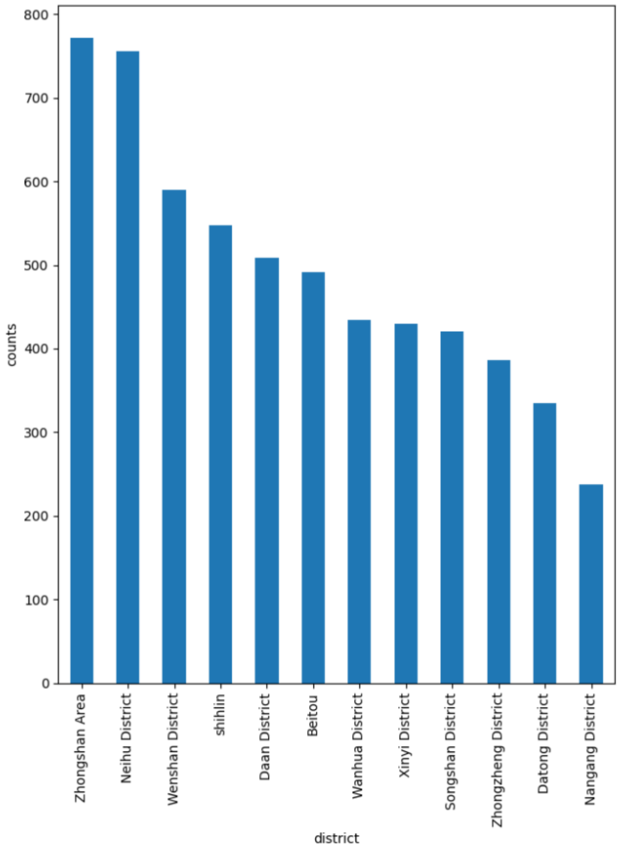

Figure 3.1 - District bar plot

Figure 3.2 shows that most transactions include both land and building, which means that the transactions in Taipei are mostly one of the apartments in a building which might contain a carpark in the basement.

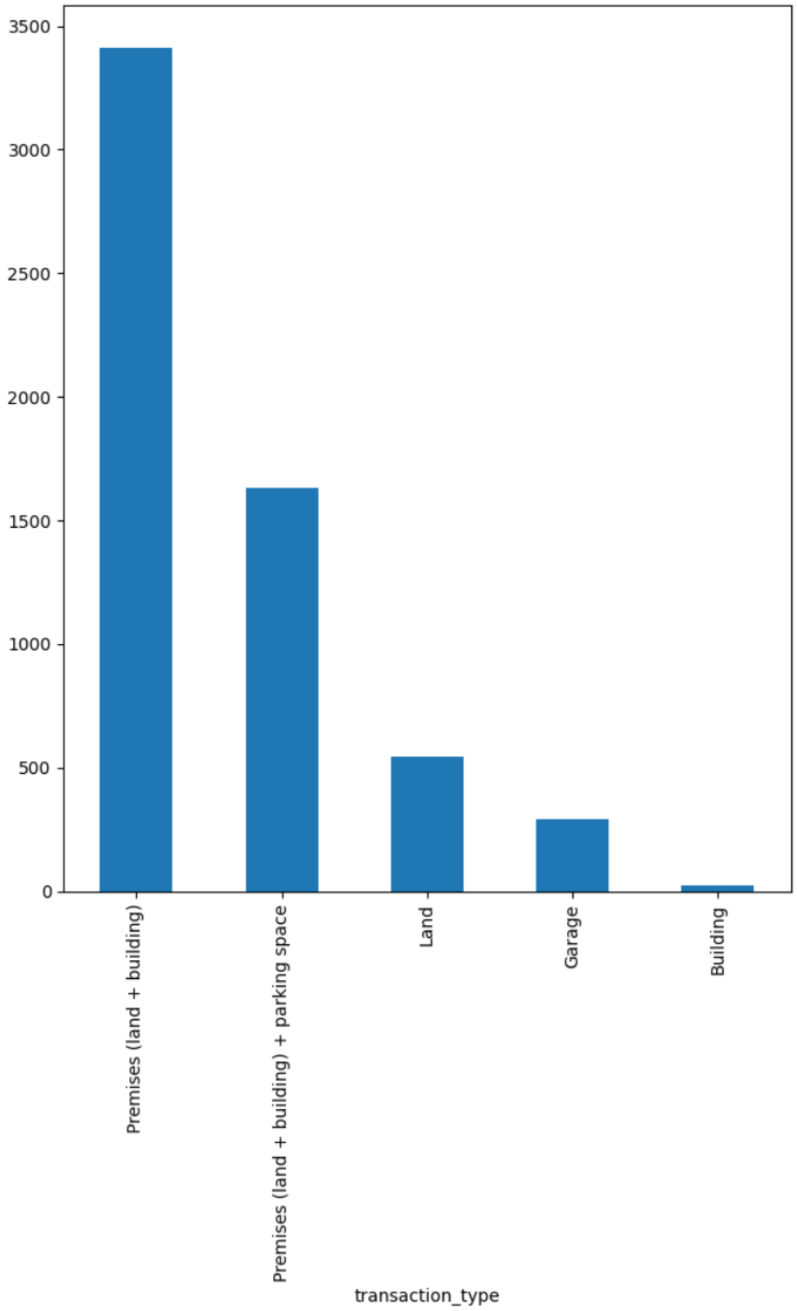

Figure 3.2 - Transaction type bar plot

Figure 3.3 shows the amount of each land use, and most transactions are for residence. There are two similar categories “other” and “others” that will be integrated into one.

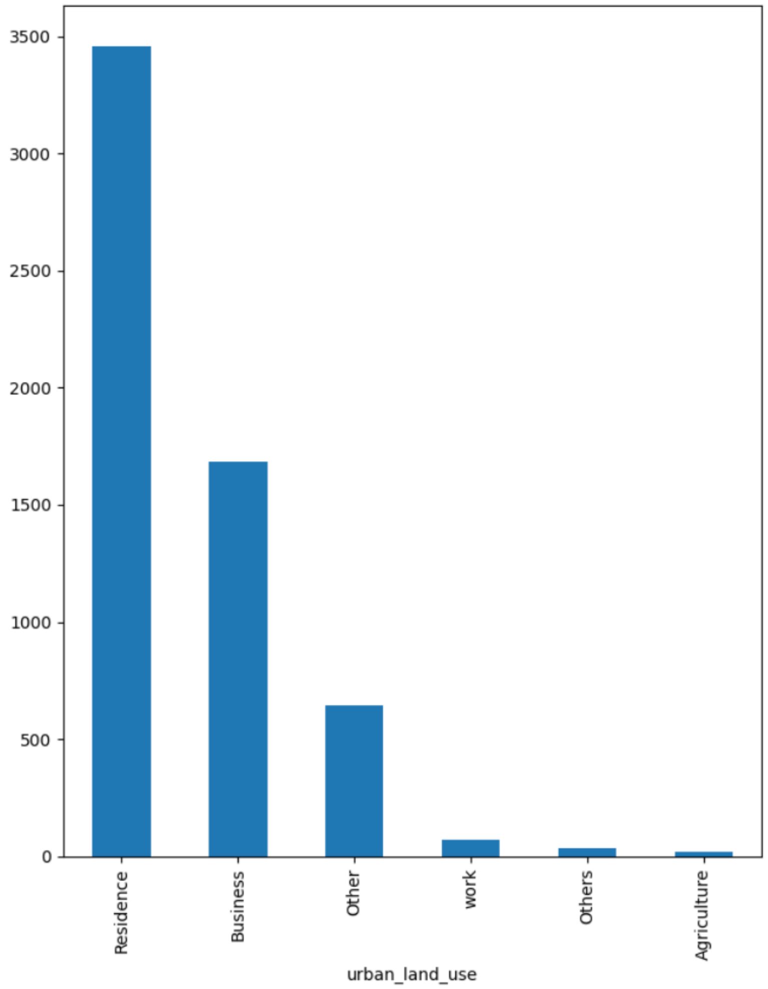

Figure 3.3 - Urban land use bar plot

As figure 3.4 shows, many properties do not come with a carpark, while those do are mostly “Ramp plane”. In Taiwan, ramp plane car parks are more expensive due to the convenience. In addition, machinery car parks include maintenance expenses in the future, which is why they are less favored.

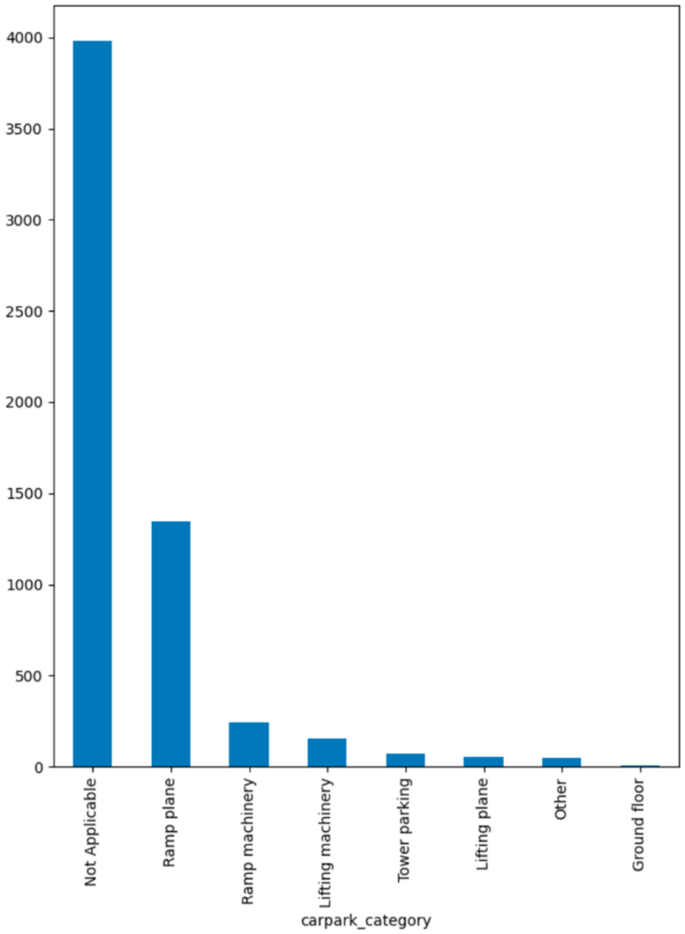

Figure 3.4 - Carpark category bar plot

Figure 3.5 explains that most of the transactions are concrete construction properties.

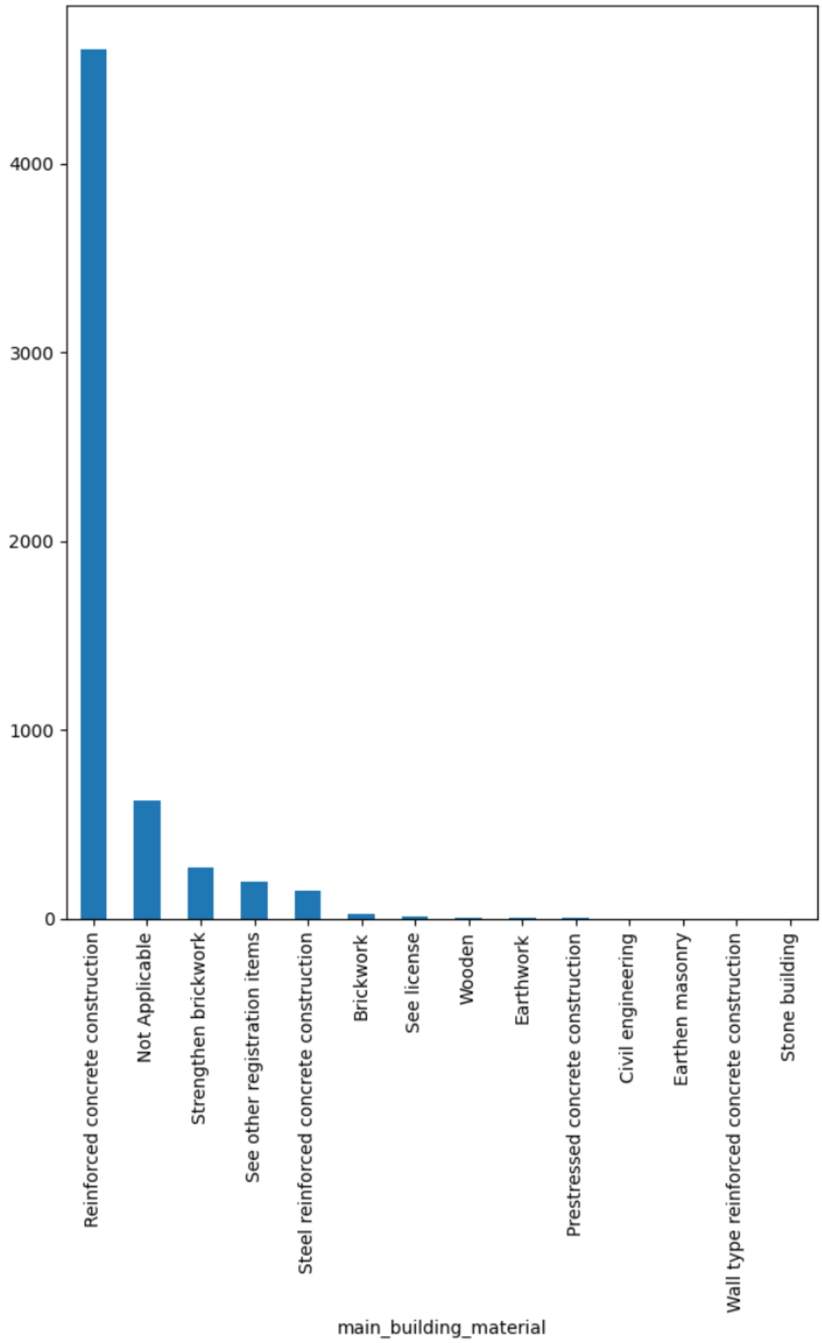

Figure 3.5 - Building material bar plot

As figure 3.6 shows, most of the properties are for residence. The main use of the property greatly influences the house price. Some central business districts in Taipei such as Xinyi District and Shihlin District have higher house prices for commercial use.

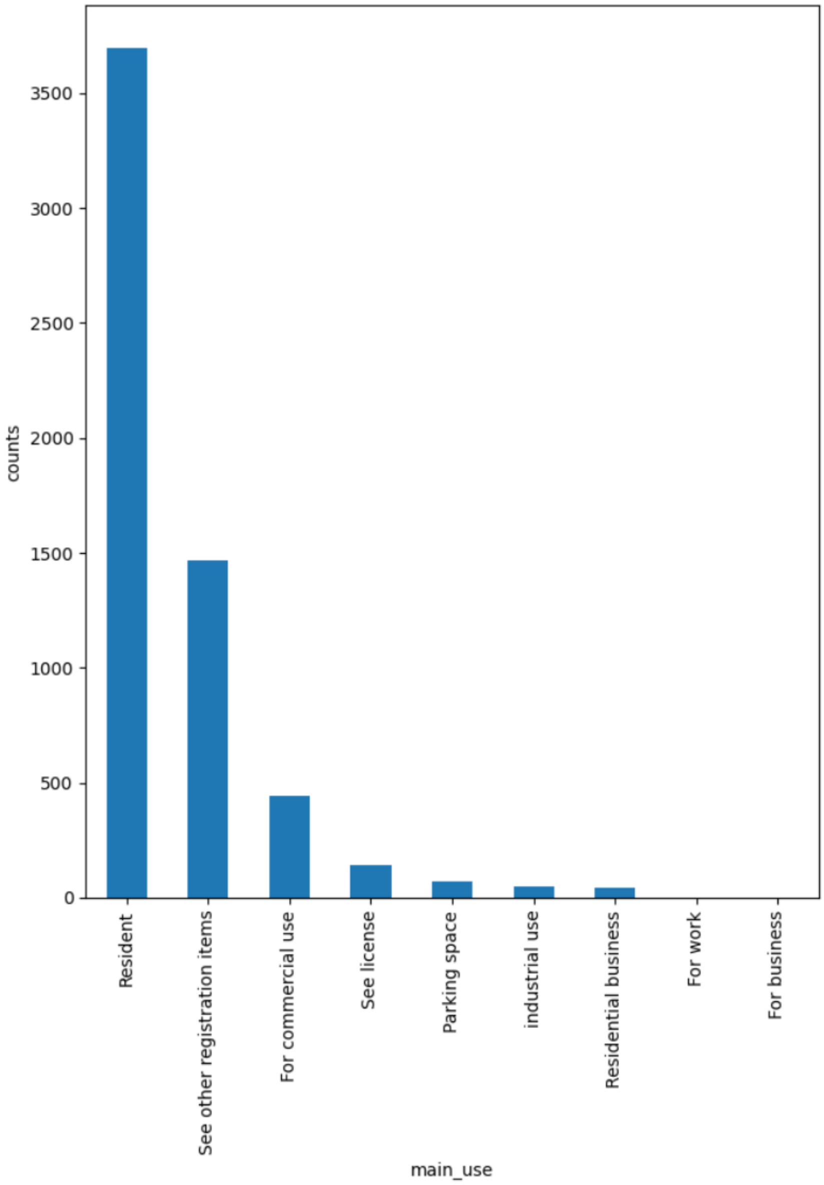

Figure 3.6 - Main use bar plot

Figure 3.7 is the transaction month bar plot which shows the month when the transactions happen. Obviously, most of the transactions happen in October, November, and December. In other words, people buy houses more in winter than in other seasons. Nevertheless, the fact that more transactions happen in winter does not influence the house price as the figure 8 shows that the mean house prices for winter are not particularly high or low.

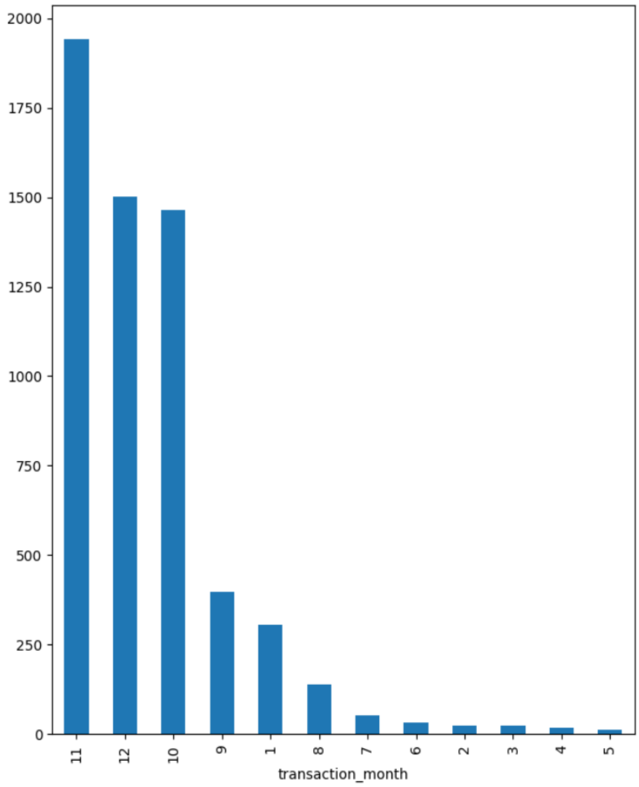

Figure 3.7 - Transaction month bar plot

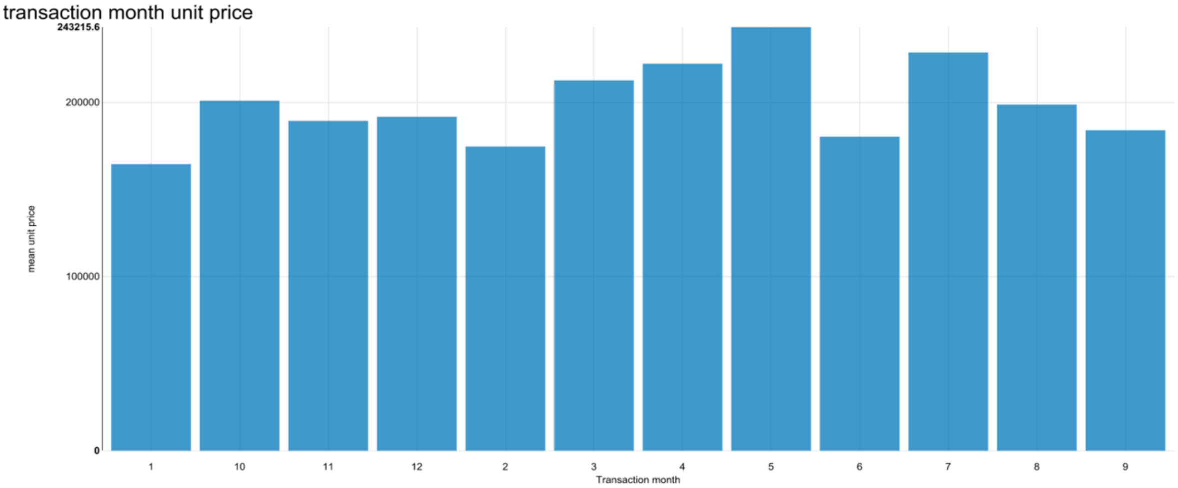

Figure 3.8 transaction month & mean unit price

Figure 3.9 explains that most of the properties among the transactions were built between 1910 and 1960, which means that most of the properties are 60 to 110 years old. House ages also provide important information for the algorithms to do the training for older houses that tend to have lower prices.

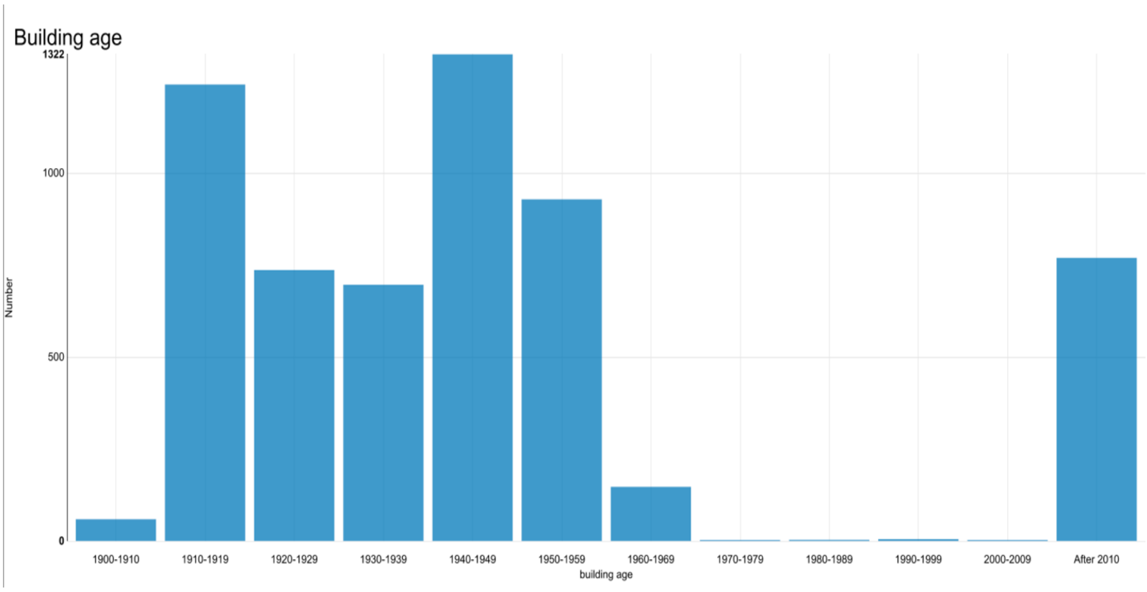

Figure 3.9 - Building bar plot

Figure 3.10 is the box plot of the building shift area which includes many outliers and may greatly decrease the reliability of the trained models or even lead to wrong conclusions. Figure 3.11 is the distribution for building shift areas which is obviously skewed and will need improvement on the normality (Zhang & Luo, 2015) (Chakravarty, Demirhan & Baser, 2020).

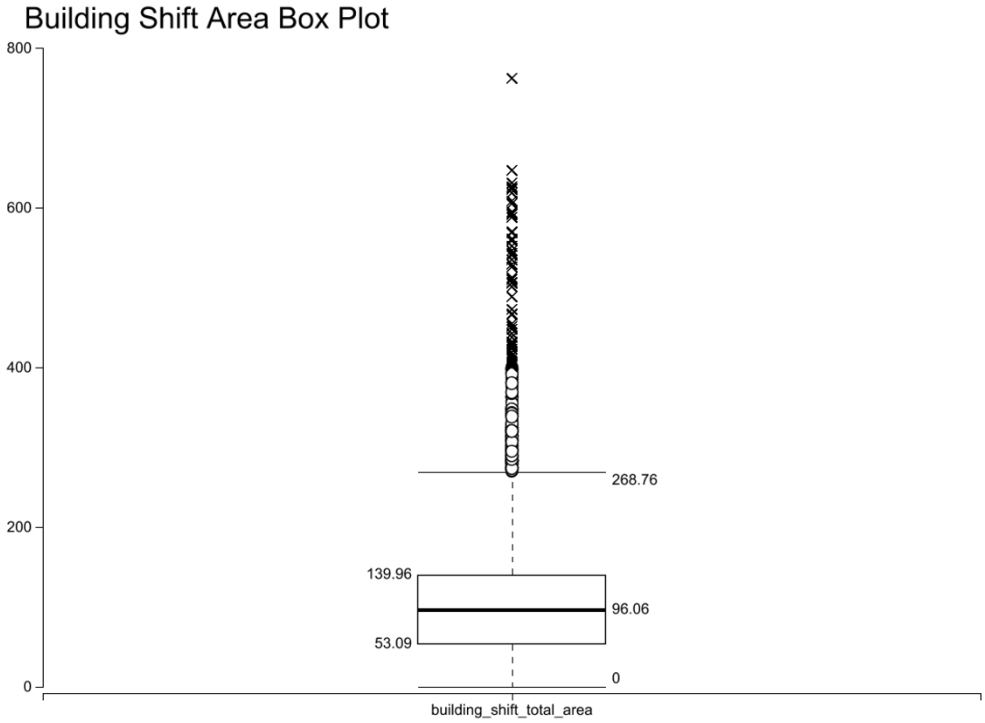

Figure 3.10 - Building shift area box plot

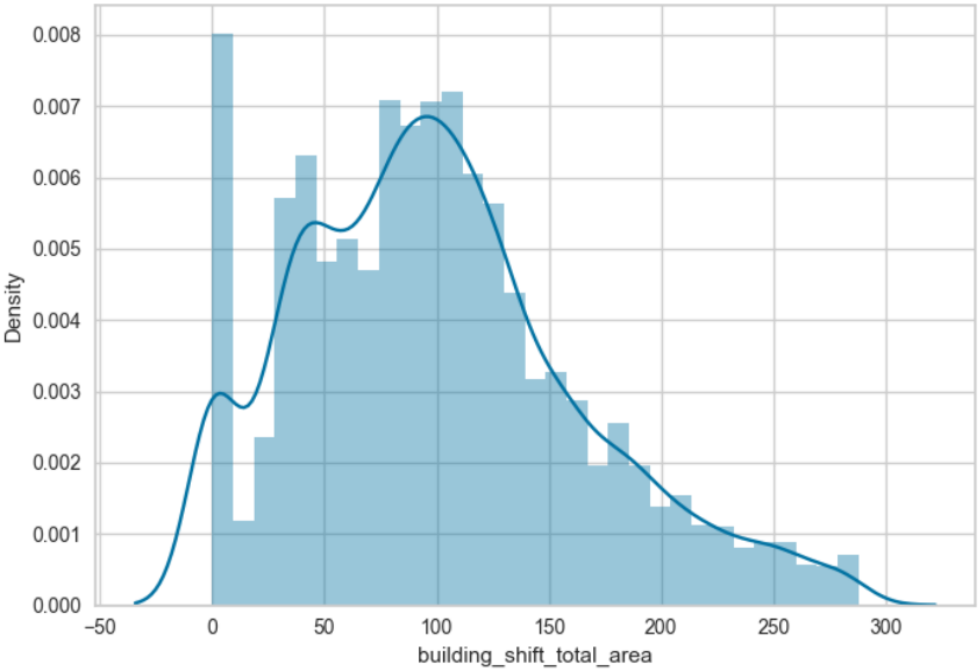

Figure 3.11 – building shift total area distribution

Figure 3.12 also shows that there are many outliers for the price per unit. The instance with 25 million NTD per unit is not very often seen in the Taipei house price transactions and the dataset shows that it is not for residence but for garage. The second instance where the unit price is 8 million is also for a garage. Since these extremely high “garage prices” increase the skewness of the data and could provide very limited information to help the model learn and predict “house prices”, they should be considered outliers and be removed.

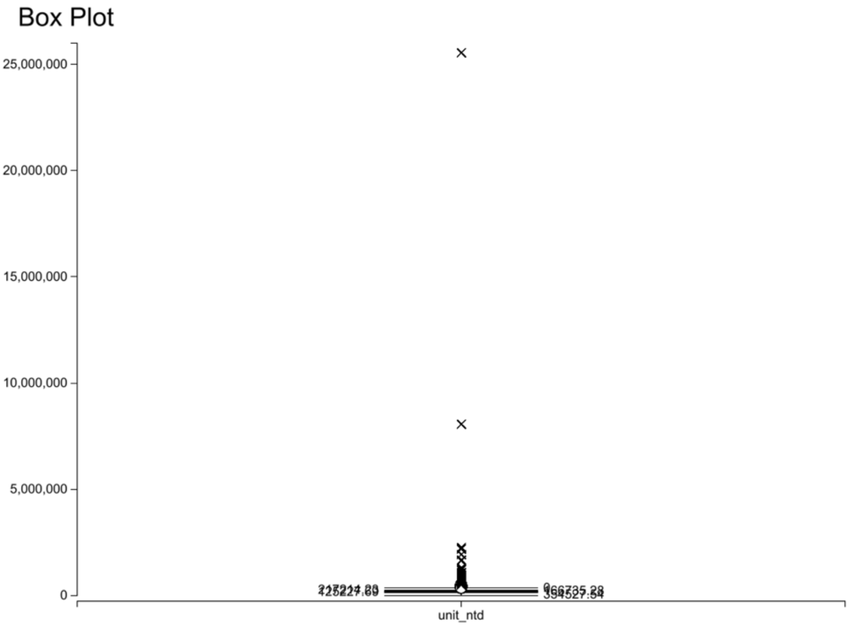

Figure 3.12 - Property unit price box plot

Figure 3.13 shows the box plot of unit price for each district, and the properties in Daan district have the most expensive unit price among all the districts in Taipei city which is 233k NTD per unit. On the other hand, Beitou district has the lowest unit price which is 130k NTD.

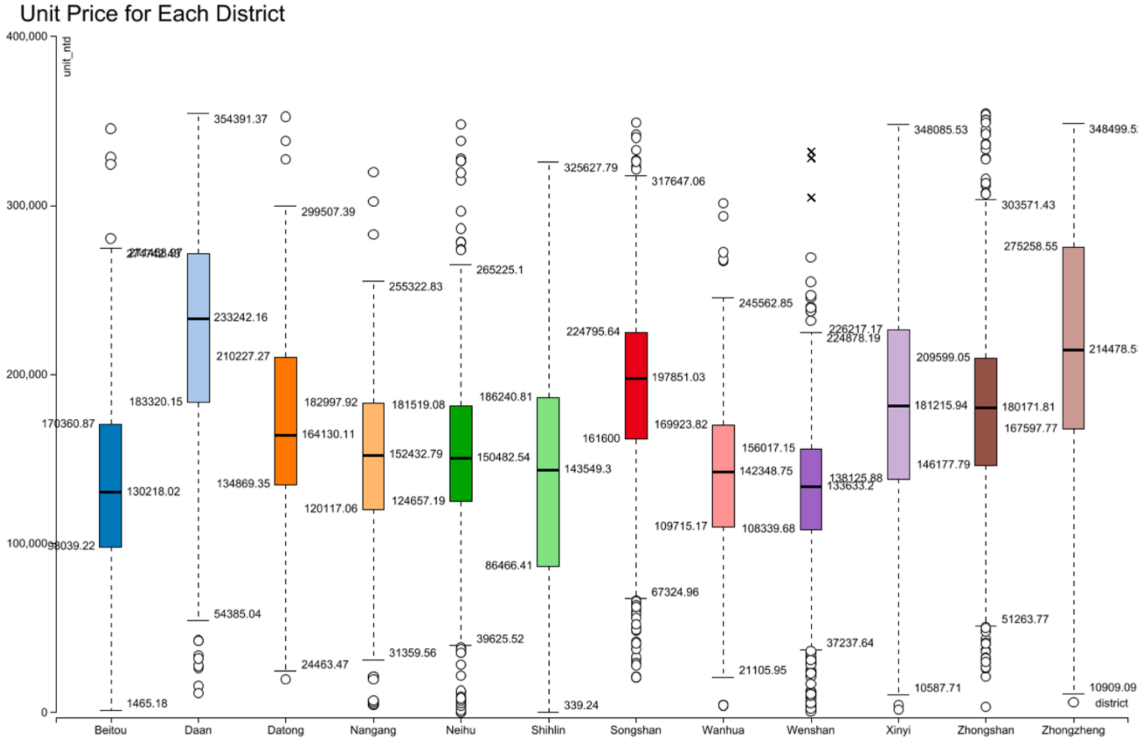

Figure 3.13 - District price per unit

Figure 3.14 shows the box plot for each transaction type, and the unit price for Premises type (with or without parking space) is more expensive than the others. In addition, there appears to be some outliers for transaction types, especially some instances of “Garage” where the unit prices are extremely high. These instances might influence the accuracy of the models and should be taken into consideration in the data cleaning process.

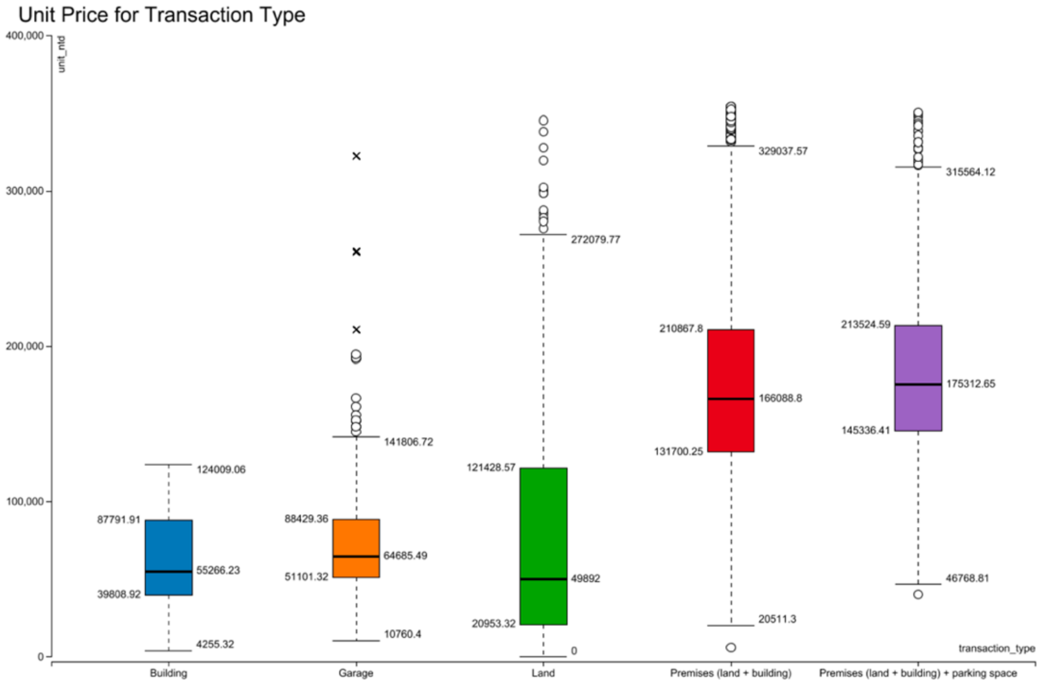

Figure 3.14 - Unit price for transaction type

Figure 3.15 shows the unit price boxplot for each land use. As the figure shows, agriculture has unit prices lower than others. Besides, agricultural land is not considered as “house” which should not be included in the house price prediction data. Therefore, the transactions with agriculture land use will be removed from the data.

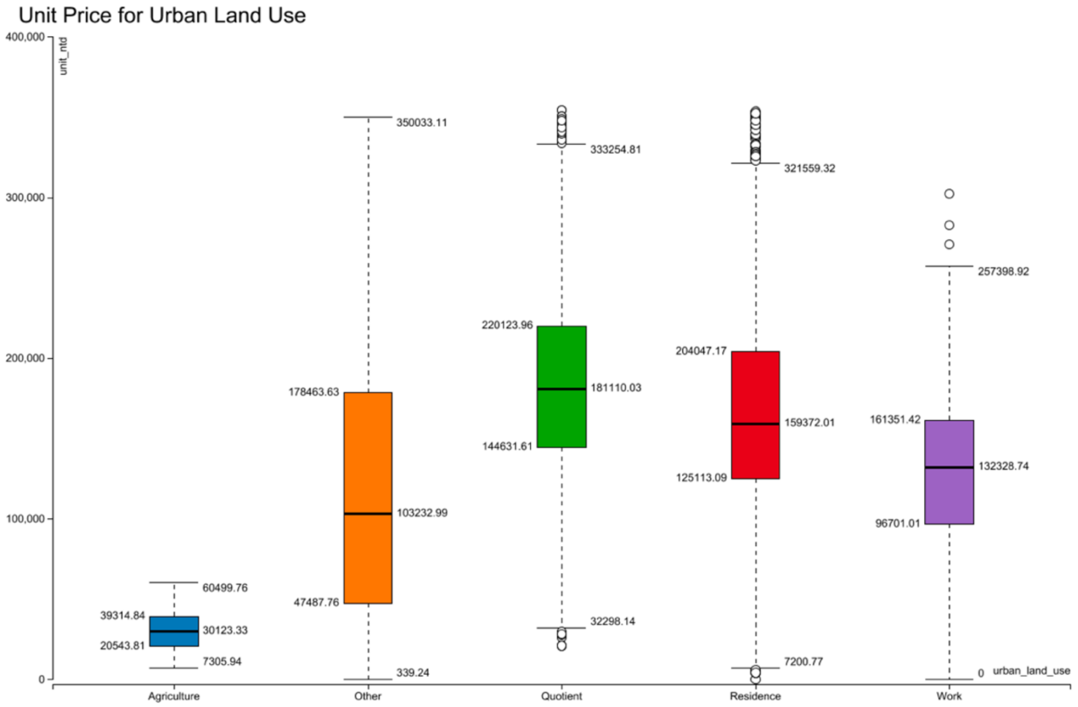

Figure 3.15 - Unit price for land use

Figure 3.16 shows the correlation of each attribute. Firstly, four features having higher correlation with the target feature “unit_ntd” are “total_ntd”, “transaction type”, “complete_year” and “number_of_building”. Actually, “total_ntd” is not being discussed in this part because it is not reasonable or meaningful to use total house to predict unit house price because the model will be applied to predict the price of houses that have not been sold yet and the price would be unknown. Secondly, transaction type tells whether the property is land, car park or building which could influence the price. Thirdly, the number of buildings tells how many buildings are included in the transaction. Finally, the high correlation between complete year and unit price indicates that the age of the property can also influence the house price.

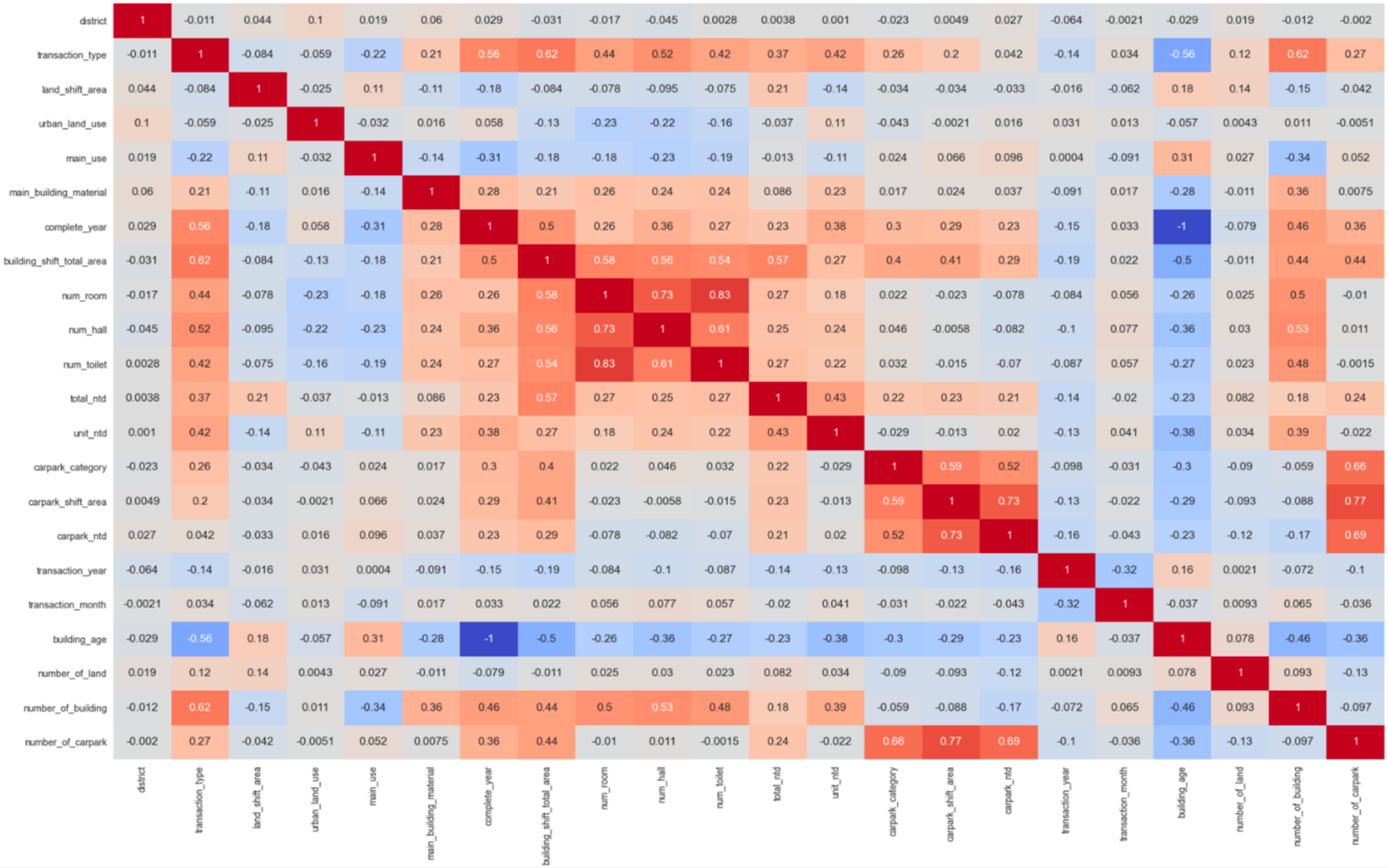

Figure 3.16 - Correlation heat map

3.4 Data processing

The pre-processing procedures will be discussed in this section. The dataset was found with some problems such as outliers and skewness according to the data discovery in the previous section. For skewness, Osborne (2013) suggested that data transformation should be applied to improve the normality of the data and should be regularly used in the data cleaning process. For outliers, Perez & Tah (2020) demonstrated that outliers not only decrease the accuracy but also lead to inaccurate or misleading result. As a result, removing outliers and transformation will both be necessary steps applied to the data. In box plot, the instances lower than (Q1- 1.5 * IQR) or upper than (Q3 + 1.5*IQR) are regarded outliers (Liu, 2008). The outliers shown in figure 10 and figure 11 are removed to prevent the reduction of accuracy on the models. Data transformation is applied to the “building_shift_total_area” column to improve the normality. Finally, **Label Encoder** from **sklearn** library is applied to the categorical columns to label the categories with numbers so the algorithms will be able to process them with mathematics.

3.5 Feature selection

Feature selection is an important step in the data mining process which is to decide which attributes to include to train the models (Witten et al., 2017, p. 61). It was demonstrated that feature selection can improve machine learning performance (Xue et al., 2016). Filter and wrapped are two methods commonly used for feature selection. Filter methods evaluate the features based on their characteristics and are demonstrated to be more computationally efficient than wrapper methods (Li et al., 2018), therefore, it is used in this study. In this study, there are five features not included in the training model and the following section will illustrate the reasons. Firstly, “building age” is calculated by “current year” and “complete year”, and it conveys the same information as “complete year”. As figure 16 showed, these two attributes have “-1” correlation, which is perfect negative correlation. Therefore, only “building age” will remain in the feature list. Secondly, “total_ntd” is not included because it is not reasonable to predict unit price by total price. Finally, “carpark_shift_area”, “carpark_ntd” and “number_of_carpark” all have very high correlation with “carpark_category”, and therefore will be excluded from the list.

3.6 Prediction model

Before training, the data is split into training and test sets using 70/30 ratio, 70% of the transactions were grouped as training data and 30 % was grouped as test data. Additionally, k-fold cross validation was also conducted to the training data. Cross validation divides the dataset into K subsets to avoid the potential bias caused by relying on only one subset. Cross validation has been applied to regression and other methods and outperformed those models without applying cross validation (Andrews, 1991; Rohani, et al., 2018; Kang, 2019). In this study, 10-fold cross validation was applied, and the mean of the 10 scores are used for evaluation on each model.

Single methods	Ensemble methods
Linear Regression	LightGBM Regressor
Ridge	Random Forest Regressor
Lasso	XGBoost Regressor

Table 1 - Machine learning methods

As Table 1 shows, there are three single methods and three ensemble methods being used in this study. Single methods are Linear regression, Ridge and Lasso. Ensemble methods are LightGBM Regressor, Random Forest Regressor and XGBoost Regressor. The first stage, all the methods were applied to do the prediction, and the best performance model is chosen to be the meta learner in the next stage. In the second stage, three different combinations of stacking methods will be applied as Table 2 shows, and their first stage learners are as follows: “all single methods”, “all ensemble methods” and “all six methods”.

Model	First stage learner	Meta learner
1	All single methods	Best performance method
2	All ensemble methods	Best performance method
3	All methods	Best performance method

Table 2 – Stacking methods combination

3.7 Evaluation

The evaluation methods used in this study are MAE, RMSE and R^2. The previous two methods use different metrics to calculate the error of the actual value and the prediction value while the latter evaluate the model ability to explain data variation.

Evaluating errors

MAE and RMSE formula:
The $\mathbf{n}$ in the below formula represents the number of the models, and the $\mathbf{e}$ represents the errors made by the model.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|e_i|$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}e_i^2}$$

Both MAE (Mean Absolute Error) and RMSE (Root Mean Square Error) are commonly used to evaluate regression models (Chai & Draxler, 2014; Swalin, 2018). MAE shows the mean of the errors made by the model. Although RMSE is not always recommended due to its sensitivity to outliers and could thus lead to misleading conclusions (Willmott & Matsuura ,2005), the outliers have been removed in the data processing procedure and the concern has been reduced. On the other hand, RMSE is raised when evaluating how close the model is to the real values (Liemohn et al., 2021) and it also better reflects large errors made by the models (Chai & Draxler, 2014). Although research argued about how RMSE and MAE outperform each other, Chai and Draxler (2014) demonstrated that it is necessary to assess models with a combination of metrics. MAE is good at showing the average error of the models, while

RMSE is good at reflecting large errors made by the model. Therefore, the model evaluation in this study will include both MAE and RMSE.

Here is an example to illustrate how MAE and RMSE can be interpreted. There are two models predicting a value, and the prediction they made are as below:

Prediction	1	2	3	4
Model X	2	4	2	4
Model Y	7	3	3	3

The correct answer is 3, and below is the error they make (absolute value):

Error	1	2	3	4
Model X	1	1	1	1
Model Y	4	0	0	0

MAE for model X:

$$\frac{|2-3|+|4-3|+|2-3|+|4-3|}{4}=1$$

MAE for model Y:

$$\frac{|7-3|+|3-3|+|3-3|+|3-3|}{4}=1$$

RMSE for model X:

$$\sqrt{\frac{(2-3)^2+(4-3)^2+(2-3)^2+(4-3)^2}{4}}=1$$

RMSE for model Y:

$$\sqrt{\frac{(7-3)^2+(3-3)^2+(3-3)^2+(3-3)^2}{4}}=2$$

The MAEs for both models are the same, which is 1. The RMSE for model X is 1, and the RMSE for model Y is 2. These two models have the same MAE, but model Y has a larger RMSE which indicates that model Y makes larger error in the prediction.

Evaluating ability to explain data variation

R squared formula:
SST = total sum of squares of dependent variable
SSR = regression sum of squares

$$R^2 = \frac{SSR}{SST}$$

Apart from evaluating the error, variation is another way to assess models, and R^2 is a common metric to use. The value of R^2 shows the variation in the data that is explained by the regression models (Sahay, 2016; Wang, 2013). In fact, it explains how reasonable a model is. The R^2 value lies between 0 and 1, and the R^2 value closer to 1, the better the model is.

3.8 Risk assessment and ethical issues

The data used in this study is collected without any human participants which is classified as “No Risk”. It is secondary data that originated from Taiwan government open data platform which contains the data of real estate transactions. In addition, it was preprocessed by Peiyuan Chieng and was released on the Kaggle platform. It is completely publicly available. The dataset does not contain any sensitive or personal information and it could not be used to track any person. Finally, the Taiwan government open data platform has released the licenses that apply to open data and all the data on this platform can be copied, modified, and used without asking for permission. In conclusion, there are no issues of laws and ethics in the use of this dataset.

4. Results

In this section, the scores of each model will be introduced separately including single, ensemble and stacked generalization models. After that, the comparison of them all will be discussed. The evaluation methods used in this study are MAE, RMSE and R^2. MAE and RMSE calculate the error between actual values and predictions made by the models. In other words, a model with smaller MAE or RMSE means that it has better performance. Additionally, RMSE is usually larger than MAE, and as the variability of error distribution grows, RMSE gets even larger than MAE (Chai & Draxler, 2014). MAE shows the average errors of the models and RMSE can suggest when larger errors are made by the models. R^2 on the other hand, tells how much the variability of the data is explained by the model, and a greater R^2 indicates a better model.

4.1 Single methods performance

Single methods	MAE	RMSE	R^2
Linear Regression	45246.42	57903.47	0.27
Ridge	45246.96	57903.00	0.26
Lasso	45246.56	57903.34	0.26

Table 3 – Single methods evaluation

In this section, the implications of each evaluation will be explained, and the performance of the single methods will be discussed according to table 3. Table 3 shows the MAE, RMSE and R^2 for the three single methods. The MAE of linear regression is 45246.42 and the RMSE is 57903.47. Ridge has an MAE of 45246.96 and RMSE of 57903.00. Finally, Lasso has an MAE of 45246.56 and RMSE of 57903.34. Apparently, the three methods have very similar scores; the scores of MAE are around 45246 and scores of RMSE are around 57903. The differences of both MAE and RMSE between any two methods are less than one. It is very hard to tell which single method is the best. On one hand, linear regression has the lowest MAE but the highest

RMSE. On the other hand, Ridge has the highest MAE but the lowest RMSE, which also indicates that this model is less likely to make large errors. Finally, the linear regression model has the highest R^2 which indicates that its ability of explaining the variation of the data is the best among all three. Nevertheless, the R^2 values of all three single methods are quite small, which means that all the three models do not explain much data variation.

4.2 Ensemble methods performance

Ensemble methods	MAE	RMSE	R^2
LightGBM Regressor	30102.05	41818.69	0.62
Random Forest Regressor	30146.66	42466.59	0.60
XGBoost Regressor	30890.56	43315.63	0.59

Table 4 - Ensemble methods evaluation

In this section, the implications of each MAE, RMSE and R^2 will be explained, and the performance of the ensemble methods will be discussed according to table 4. Table 4 shows the MAE and RMSE for the three ensemble methods. LightGBM has the best performance of 30102.05, and Random Forest comes after with MAE of 30146.66. Surprisingly, XGBoost Regressor has the worst performance of all, with a MAE of 30890. The RMSE of these three shows the same result as MAE that LightGBM has the best performance while XGBoost has the worst performance. In addition, LightGBM is also the best of all six methods, therefore, it is chosen as the meta learner in stacked generalization models. The R^2 values of the three ensemble methods are relatively high. LightGBM especially has the highest R^2, and 62 % of the variation is explained by the model.

4.3 Stacked generalization models performance

Model	First stage learner	Meta learner	MAE	RMSE	R^2
1	All single methods	LightGBM	45106.17	57894.81	0.26
2	All ensemble methods	LightGBM	31378.75	44083.54	0.56
3	All methods	LightGBM	31005.82	44151.94	0.57

Table 5 – Stacked generalization evaluation

In this section, the implications of each MAE, RMSE and R^2 will be explained, and the performance of all the stacked generalization models will be discussed according to table 5. Table 5 shows the performance of the three stacked generalization models. Model 3 using all six methods as first stage learner in stacked generalization shows the best performance in MAE. Model 2 using three ensemble methods as first stage learners shows the best performance in RMSE. This explains that although the mean error that model 3 makes is the smallest, it has a larger RMSE than model 2 and could have made larger errors in the prediction than model 2. However, model 3 still outperforms model 2 in R^2, which means that it has a better ability to explain the data variation.

4.4 Comparison

Ensemble methods VS Single methods

Ensemble methods do perform better than single methods as table 3 and 4 show. The average MAE of ensemble methods is around 30k while the average MAE of single methods is around 45k. The average RMSE of ensemble methods is around 42k while the average RMSE of single methods is around 57k. Both evaluation metrics indicate that ensemble methods outperform single methods in the Taipei house price prediction. The R^2 values of the single methods ranging from 0.26 to 0.27, while the R^2 values of ensemble methods ranging from 0.59 to 0.62 which indicates that the ensemble methods have much better ability in explaining the variability of the data.

Model 1 VS single methods

Model 1 in table 5 is the stacked generalization using all three single methods, “Linear regression”, “Ridge” and “Lasso” as the first stage learner in stacked generalization, and the meta learner is the best-performance ensemble method, “LightGBM”. The MAE of model 1 is 45156 which is slightly better than the MAE of the three single methods. The RMSE of model 1 is 57894, which is also slightly better than the RMSE of the three single methods and that the stacking model could make smaller errors compared to the single methods. This indicates that the stacked generalization using single methods as first stage learners does improve the prediction. Nevertheless, the R^2 value of linear regression is higher than model 1 which suggested that it has better ability in the explanation of data variability than model 1.

Model 2 VS ensemble methods

Model 2 uses all the ensemble methods “LightGBM”, “Random Forest” and “XGBoost” as the first stage learner and uses “LightGBM” as the meta learner. The MAE of model 2 is 31378 and the RMSE is 44083 which are both higher than all the three ensemble methods, which means that the mean of the errors made by model 2 is larger than all the ensemble methods and model 2 could make larger individual errors than all the ensemble errors. The results indicate that stacked generalization using ensemble methods in the first stage does not improve the prediction. Furthermore, the R^2 value of model 2 is 0.56 which is relatively low compared to the three ensemble methods (**LightGBM**: 0.62, **Random Forest**: 0.60, **XGBoost**: 0.59). This suggested that the three ensemble models all have better ability to explain the data variation compared to model 2.

Comparison of stacking models

Model 3 shows the best performance and model 2 comes after. It can be inferred from the performance comparison between single methods and ensemble methods that stacking models with ensemble methods as first stage learners should be able to do better prediction. Evidently, model 2 which uses ensemble methods in the first stage indeed has lower MAE and RMSE than model 1. On the other hand, model 3 with all the six methods included has the best MAE of all three, which is 31005. As section 4.3 mentioned, although the mean of the errors made by model 3 is the smallest, it has higher RMSE than model 2 which indicates that model 3 could make larger errors than model 2. The R^2 values of the three models are 0.26, 0.56 and 0.57. Model 3 (R^2 = 0.57) performed the best in explaining the variation in the data, and model 2 (R^2 = 0.56) comes after. Out of the three evaluations, model 3 has the lowest

MAE and the best R^2, therefore, model 3 is considered the best model among all the three stacking models. Stacked generalization using different methods in the first stage can lead to different results and seems that the ones with ensemble methods included do have better prediction. In addition, the model using the combination of single methods and ensemble methods as first stage learners performs particularly well. In the discussion part, the implication of the comparison will be further discussed and concluded.

5. Discussion

In this section, the findings are firstly presented according to the results. Secondly, the research questions are answered, and the rationality is explained. Thirdly, the connection between the findings and the previous research is discussed. After that, some potential limitations and improvements for this study are provided. Finally, future research questions for this topic are recommended.

5.1 Findings and Impact

According to the result of this study, stacked generalization did improve the prediction. When the single methods (Linear regression, Ridge and Lasso) were applied separately, linear regression had the best performance in MAE and the ability to explain the data variation while Ridge had the lowest RMSE. Model 1 using all single methods as the first stage learner in stacking showed lower error and better ability to explain the data variation, which indicated that it did improve the prediction. When the ensemble methods (LightGBM, Random Forest and XGBoost) were individually conducted, LightGBM showed the best performance of both error wide and variation evaluation which is why it was chosen as the meta-learner in the second stage of stacked generalization. Overall, all three ensemble methods had much better results than the single methods in all the evaluations. On the other hand, model 2 using all ensemble methods in the first stage showed worse performance than any of the ensemble methods. Although model 2 shows a relatively good performance, the result indeed implies that stacking does not always make better predictions. Finally, model 3 which included all six methods in the first stage showed the best result among the three stacking models. However, all the stacking models only outperformed single models, not any of the ensemble methods. To sum up, stacked generalization can improve the prediction when the first stage learners are single methods, but it does not show any improvement on the one using ensemble learners as first stage learners in this study. The findings can contribute to the Taiwan government for policy making based on Taipei house price prediction that stacked generalization can provide relatively good performance, however, it is not suggested to use all single methods in the first stage.

5.2 Answering research question and rationality

What is the performance of stacked generalization on prediction of house pricing?

The performance on the evaluation and performance on the improvement will be discussed separately in this section. In general, two out of three stacking models showed relatively good performance, and the two models are model 2 and model 3. Compared to the single methods, stacked generalization did perform better. However, stacking models with good performance did not mean that it has great improvement. The stacking model using ensemble methods in the first stage showed good performance but did not improve the prediction compared to the performance of each ensemble method. Whether a stacking model could improve the prediction is related to the methods used in the first stage. In this study, the stacking model using single methods in the first stage showed obvious improvement while the one using ensemble methods did not. According to previous studies, the selection of the methods used is likely to be the main reason for this which will be discussed in the next section.

Does stacked generalization outperform other single methods on the prediction of house pricing in Taipei?

Yes, stacked generalization does outperform the single methods on the prediction of house pricing in Taipei in both prediction error and ability to explain the data variation. Stacking is an ensemble method which makes prediction based on the previous predictions made by the first stage methods. The generalizability is increased, and the uncertainty is decreased through the process of stacking (Alkenani et al., 2021). Therefore, stacking makes more reliable predictions than single methods.

If the ensemble methods are used in the first-level learner, will it outperform the model that uses single methods as first-level learner?

Yes, the stacking model using ensemble learners in the first stage does outperform the one using single methods. When ensemble methods were applied individually, they performed better than single methods. Healey et al. (2018) demonstrated that when new inputs are fed to the stacking model, further information is provided. Therefore, when ensemble methods provided stronger inputs to the stacking model, the predictions could make better predictions based on the strong inputs and thus lead to a better stacking model.

5.3 Connection with literature review

The findings of this study showed consistency with previous studies that ensemble methods have higher performance than single methods on the prediction most of the time (King, Abrahams & Ragsdale, 2014; Ren, Suganthan & Srikanth, 2015; Liu et al., 2021). In this study, LightGBM, Random Forest, XGBoost and Stacked generalization all outperformed the single methods, and it is because these ensemble methods improve the models through different approaches. Random Forest lowers the impact of data bias through combining the result of many decision trees (Shaikhina et al., 2019). XGBoost creates a series of trees sequentially and each learns from the previous tree to improve the models (Basak et al., 2019). LightGBM works on similar principles with XGBoost. Compared to XGBoost, LightGBM takes up less memory consumption which makes it work faster, and it reduces errors for better accuracy and thus makes better prediction (Zhang, Deng & Jia, 2020).

Furthermore, the research of Xiong et al. (2020) showed that the stacking model which included the most methods had the best performance which aligned with the result of this study that model 3 using all six methods in the first stage outperformed the other staking models. Although the study of Zhang et al. (2020) indicated that models that include the most methods do not always have the best performance. Different combinations of methods used in the stacking model could also lead to different results, and it could be a new topic to find out. The exploration of the methods used could also lead to better understanding of stacked generalization. When ensemble methods are compared to stacked generalization, the research of Truong et al. (2020) showed that stacked generalization model slightly outperformed the XGBoost, Random Forest and LightGBM. Xiong et al. (2020) also found that stacked generalization model using those ensemble methods in the first stage outperforms the models applying ensemble methods individually. However, the result of this study showed that when ensemble methods are applied to the Taipei house data individually, each of the performance is better than the stacked generalization model that includes all these methods in the first stage. There could be many reasons, the different dataset used, selected features or the methods included could all lead to different results. On the other hand, ensemble methods are improved through the new inputs which provide further information (Healey et al., 2018), and the selection of the methods are important. Healey et al. (2018) demonstrated that the stability of the algorithms can influence the prediction accuracy, and therefore, choosing the right algorithms used in ensemble models can affect the extent of improvement on the models. In addition, Ting (1999) suggested that the confidence of the methods used in the first stage of stacked generalization can greatly influence the result of the prediction and should be taken into consideration when selecting the methods used. Finally, Breiman (1996) illustrated that non-negativity constraint and the similarity of the first stage learners could both influence the stacking regression on the predictive accuracy. That is to say, to improve the stacking model, it is important to constrain the regression coefficients to be non-negative. Besides, the stacking models improves the prediction when diverse and unlike methods are used in the first stage (Breiman, 1996). This implies that when similar methods are used in the first stage of stacking, the improvement made by stacking could be limited. Additionally, Wang et al. (2006) demonstrated that it is more desirable that each base-learner provides a unique point of view so the information used to feed the meta-learner is not redundant.

This could explain why model 2 showed worse performance than the ensemble methods because LightGBM, Random Forest and XGBoost are all tree-based algorithms. Random Forest is based on the “bagging” method and random feature selection to build a group of decision trees (Mantas et al., 2019). XGBoost is a boosting-based method which constructs several trees, and each tree learns from the residual that previous tree made to make better predictions (Sagi & Rokach, 2021). LightGBM on the other hand is based on Gradient Boosting Decision Tree algorithm, or a modified version of XGBoost. LightGBM technique has been demonstrated to be 20 times faster than XGBoost (Zhang et al., 2019). LightGBM, Random Forest and XGBoost are all tree-based methods and work similarly. As a matter of fact, these three tree-based methods applied in the first stage of model 2 could be the reason that it does not improve the performance of stacking according to the study of Breiman (1996).

5.4 Limitation and improvements

There are some limitations of data pre-processing and method selection in this study. The dataset used in this study is retrieved from Kaggle which is well pre-processed and translated. It indeed provides convenience and reduces time consuming. However, it has dropped some columns which contain useful information, such as shifting level and total level. Peiyuan Chieng who provided the pre-processed data in Kaggle explained that these columns are “too unstructured” so that they are excluded. Sifting level indicates the level of the property, and house prices increase with the rise of floor level. Wong et al. (2011) demonstrated that high floor levels are less influenced by the environmental quality problems such as air pollution and noise (as shown in Figure 5.1), and the house prices are thus higher than the low-level ones. In addition, the fourth level is always with a lower price due to the traditional Chinese culture. As a matter of fact, if these two columns are well organised and processed, they could absolutely bring valuable information to the prediction.

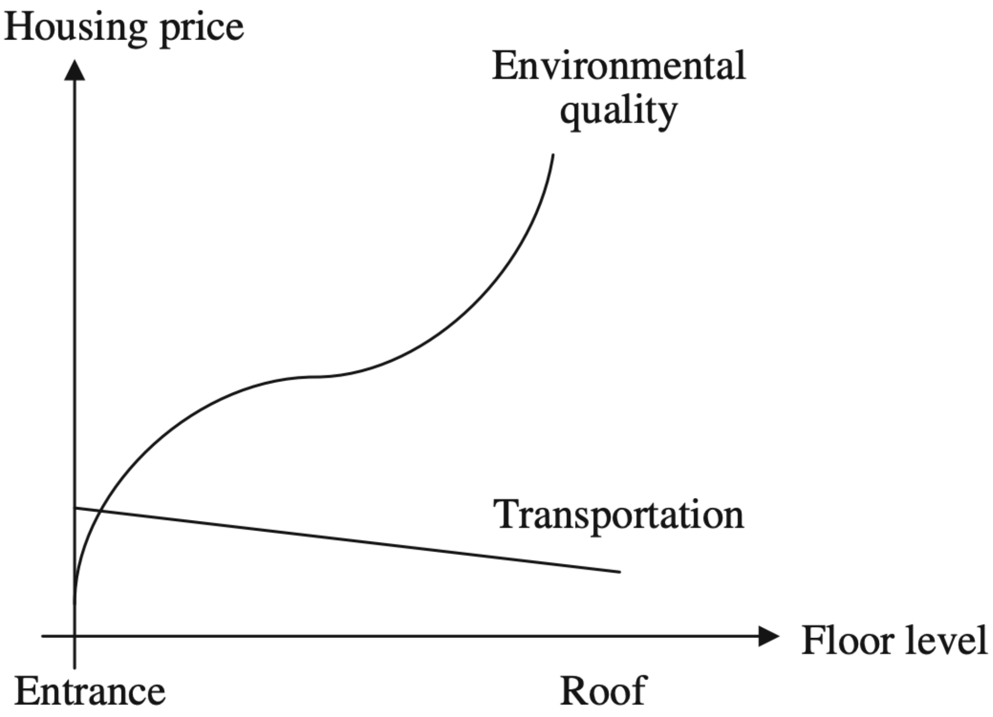

Figure 5.1

Another limitation about this study is the combination of the learners in the first stage of stacking models. As the previous section mentioned, Breiman (1996), Healey et al. (2018), Ting (1999) all had demonstrations supporting the method selection in stacked generalization which could improve the stacking model. The stability of the algorithms raised by Healey et al. (2018), the confidence of the methods used in first stage of stacked generalization suggested by Ting (1999) and the non-negativity constraint and the similarity of the first stage learners illustrated by Breiman (1996) could all influence the stacking models. These were not addressed in this study and could thus miss some chances to further improve the stacking models.

5.5 Recommendation for future research

In this study, the three different combinations of first stage learners in stacking were compared and the one with all six methods outperformed the other two. The first stacking model includes “Linear regression”, “Lasso” and “Ridge” in the first stage. The second model includes “Random Forest”, “LightGBM” and

“XGBoost”. The third model includes all the six methods mentioned which showed the best performance. However, it would be interesting to find out a better combination of first stage learners which could lead to further understanding of stacked generalization. According to Breiman (1996), Healey et al. (2018), Ting (1999), there are some approaches to evaluate and choose methods to be applied in stacking which leads to better results. Hence, future research could try to improve the stacking model through these approaches.

Secondly, this study only measures the models through the prediction error and ability to explain the variation while there are many other metrics to evaluate the models. When the stacking models are conducted, the time consuming is also important for it might contain some ensemble methods such as bagging which takes a long time to process. Hence, evaluating the calculating speed could help find a more efficient model.

6. Conclusion

The main aim of this study is to find out how stacked generalization performs on the prediction of Taipei house prices and to compare the performance of ensemble methods and single methods. This study can lead to better understanding of the use of stacked generalization and how it works on the prediction of Taipei house prices. Previous literature has applied many methods to predict Taipei house price, however, stacked generalization which has been conducted to do house price prediction of other cities have not been applied to Taipei house price dataset yet. Therefore, this study conducts the stacked generalization to predict Taipei house price data retrieved from Kaggle and aims to find out how stacked generalization can contribute to the prediction. Three research questions were formulated accordingly. The first question was to find out the performance of stacked generalization on prediction of house pricing in Taipei. The second question was to reveal if stacked generalization outperforms the single methods on the prediction of house pricing in Taipei. Finally, the third question was to discover the performance of different combinations used in the first stage of stacked generalization. The Kaggle used in this study had finished most of the data processing. Although there are some parts of the data processing procedure that were different from what it was expected, it was good enough for the analysis and has reduced time consuming.

The findings of this study also showed that when all the ensemble methods are applied in the first stage of the stacking model, it does not improve the prediction compared to the result of each ensemble method. However, when all the single methods are applied in the first stage of the stacking model, the model showed great improvement compared to those single methods, which answered the first research question that Stacked generalization does not always improve the prediction and it depends on the methods used in the first stage. Stacked generalization has shown relatively good performance compared to the single methods in this study, which answered the second research question that Stacking models do outperform single methods. Model 1 which includes all the single methods in the first stage showed the worst performance. Model 2 which includes all the ensemble methods and model 3 which includes all the six methods both showed good performance, and model 3 is slightly better. This answered the third research question that the stacking models including ensemble methods in the first stage did show better performance on the prediction. Although other ensemble methods such as LightGBM, Random Forest and XGBoost have slightly outperformed model 2 and model 3, the difference was little.

To sum up, the best performance showed up when LightGBM is applied to the Taipei house price dataset. The ensemble methods made the least error and showed best performance in data variation explanation. Although the stacking models are not as good as the ensemble methods, the performance of the ones with ensemble methods included are pretty good. Stacking improves the prediction when the first level methods are single methods. However, it did not show any improvement on the prediction of house prices when first stage learners are all ensemble methods.

The findings could support the method selection when the prediction of house prices in Taipei is conducted. In the future, there could be platforms designed to predict house prices and present to home buyers to help them better understand the real value of the houses. In addition, the government could conduct house price prediction to support policy making. The findings allow them to select the methods used in the prediction, and it is suggested to use ensemble methods over single methods. If the stacked generalization is selected, it is recommended to include ensemble methods in the first stage.

References

Allam, Zaheer, Dey, Gourav, & Jones, David S. (2020). Artificial Intelligence (AI) Provided Early Detection of the Coronavirus (COVID-19) in China and Will Influence Future Urban Health Policy Internationally. AI, 1(2), 156–165. https://doi.org/10.3390/ai1020009

Alkenani, A. H., Li, Y., Xu, Y., & Zhang, Q. (2021). Predicting Alzheimer’s Disease from Spoken and Written Language Using Fusion-Based Stacked Generalization. Journal of Biomedical Informatics, 118, 103803–103803. https://doi.org/10.1016/j.jbi.2021.103803

Andrews, Donald W.K. (1991). Asymptotic optimality of generalized CL, cross-validation, and generalized cross-validation in regression with heteroskedastic errors. Journal of Econometrics, 47(2), 359–377. https://doi.org/10.1016/0304-4076(91)90107-O

Basak, Suryoday, Kar, Saibal, Saha, Snehanshu, Khaidem, Luckyson, & Dey, Sudeepa Roy. (2019). Predicting the direction of stock market prices using tree-based classifiers. *The North American Journal of Economics and* Finance, 47, 552–567. https://doi.org/10.1016/j.najef.2018.06.013

Benjamin, John D, Chinloy, Peter, & Jud, G Donald. (2004). Real Estate Versus Financial Wealth in Consumption. The Journal of Real Estate Finance and Economics, 29(3), 341–354. https://doi.org/10.1023/B:REAL.0000036677.42950.98

Bing Xue, Mengjie Zhang, Browne, Will N, & Xin Yao. (2016). A Survey on Evolutionary Computation Approaches to Feature Selection. IEEE Transactions on Evolutionary Computation, 20(4), 606–626. https://doi.org/10.1109/TEVC.2015.2504420

Bostic, Raphael, Gabriel, Stuart, & Painter, Gary. (2009). Housing wealth, financial wealth, and consumption: New evidence from micro data. Regional Science and Urban Economics, 39(1), 79–89. https://doi.org/10.1016/j.regsciurbeco.2008.06.002

Breiman, L. Stacked regressions. Mach Learn 24, 49–64 (1996). https://doi.org/10.1007/BF00117832

Central Bank of the Republic of China. (2011, March 23) *Average lending rate* https://www.cbc.gov.tw/tw/cp-302-133666-a49be-1.html

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

Chakravarty, Srinivas, Demirhan, Haydar, & Baser, Furkan. (2020). Fuzzy regression functions with a noise cluster and the impact of outliers on mainstream machine learning methods in the regression setting. Applied Soft Computing, 96, 106535. https://doi.org/10.1016/j.asoc.2020.106535

Chandler, Daniel, & Disney, Richard. (2014). The Housing Market in the United Kingdom: Effects of House Price Volatility on Households. Fiscal Studies, 35(3), 371–394. https://doi.org/10.1111/j.1475-5890.2014.12034.x

Chen, Jieh-Haur, Ong, Chuan Fan, Zheng, Linzi, & Hsu, Shu-Chien. (2017). Forecasting spatial dynamics of the housing market using Support Vector Machine. International Journal of Strategic Property Management, 21(3), 273–283. https://doi.org/10.3846/1648715X.2016.1259190

Dellinger, A. J. (2015) Tim Wu says Google is degrading the Web to favor its own products. The Daily Dot. June 29. http://www. dailydot.com/technology/google-search-tim-wu-yelp/.

Dieleman, Frans & Clark, William & Deurloo, Marinus. (2000). The Geography of Residential Turnover in Twenty-seven Large US Metropolitan Housing Markets, 1985-95. International Journal of Geographical Information Science - GIS. 37. 223-245. 10.1080/0042098002168

Dietterich, T.G. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Mach. Learn*. 2000, 40, 139–157.

Dou, Jie, Yunus, Ali P, Dieu Tien Bui, Merghadi, Abdelaziz, Sahana, Mehebub, Zhu, Zhongfan, Chen, Chi-Wen, Han, Zheng, & Binh Thai Pham. (2020). Improved landslide assessment using support vector machine with bagging, boosting, and stacking ensemble machine learning framework in a mountainous watershed, Japan. Landslides, 17(3), 641–658. https://doi.org/10.1007/s10346-019-01286-5

Etzioni, Amitai, & Etzioni, Oren. (2016). AI assisted ethics. Ethics and Information Technology, 18(2), 149–156. https://doi.org/10.1007/s10676-016-9400-6

Gathergood, John. (2011). Unemployment risk, house price risk and the transition into home ownership in the United Kingdom. Journal of Housing Economics, 20(3), 200–209. https://doi.org/10.1016/j.jhe.2011.03.001

Goodhart, Charles, & Hofmann, Boris. (2008). House prices, money, credit, and the macroeconomy. Oxford Review of Economic Policy, 24(1), 180–205. https://doi.org/10.1093/oxrep/grn009

Gulson, Kalervo N, & Webb, P Taylor. (2017). Mapping an emergent field of 'computational education policy': Policy rationalities, prediction and data in the age of Artificial Intelligence. Research in Education (Manchester), 98(1), 14–26. https://doi.org/10.1177/0034523717723385

Healey, S. P., Cohen, W. B., Yang, Z., Kenneth Brewer, C., Brooks, E. B., Gorelick, N., Hernandez, A. J., Huang, C., Joseph Hughes, M., Kennedy, R. E., Loveland, T. R., Moisen, G. G., Schroeder, T. A., Stehman, S. V., Vogelmann, J. E., Woodcock, C. E., Yang, L., & Zhu, Z. (2018). Mapping forest change using stacked generalization: An ensemble approach. Remote Sensing of Environment, 204, 717–728. https://doi.org/10.1016/j.rse.2017.09.029

Hu, Mingzhi, & Ye, Wenping. (2020). Home Ownership and Subjective Wellbeing: A Perspective from Ownership Heterogeneity. Journal of Happiness Studies, 21(3), 1059–1079. https://doi.org/10.1007/s10902-019-00120-y

Irandoust, Manuchehr. (2019). House prices and unemployment: an empirical analysis of causality. International Journal of Housing Markets and Analysis, 12(1), 148–164. https://doi.org/10.1108/IJHMA-03-2018-0021

James, Benjamin T, Luczak, Brian B, & Girgis, Hani Z. (2018). MeShClust: an intelligent tool for clustering DNA sequences. Nucleic Acids Research, 46(14), e83–e83. https://doi.org/10.1093/nar/gky315

Johnes, Geraint, & Hyclak, Thomas. (1999). House prices and regional labor markets. The Annals of Regional Science, 33(1), 33–49. https://doi.org/10.1007/s001680050091

Kelleher, J. D., Mac Namee, Brian, & D'Arcy, Aoife. (2015). Fundamentals of machine learning for predictive data analytics : algorithms, worked examples, and case studies. The MIT Press.

Kang, Kyeonghwan, Qin, Caiyan, Lee, Bongjae, & Lee, Ikjin. (2019). Modified screening-based Kriging method with cross validation and application to engineering design. Applied Mathematical Modelling, 70, 626-642.

King, Michael A, Abrahams, Alan S, & Ragsdale, Cliff T. (2014). Ensemble methods for advanced skier days prediction. Expert Systems with Applications, 41(4), 1176–1188. https://doi.org/10.1016/j.eswa.2013.08.002

Kraft, Holger, & Munk, Claus. (2011). Optimal Housing, Consumption, and Investment Decisions over the Life Cycle. Management Science, 57(6), 1025–1041. https://doi.org/10.1287/mnsc.1110.1336

Lee, T.-W., & Chen, K. (2008). Prediction of House Unit Price in Taipei City Using Support Vector Regression. http://apiems2016.conf.tw/site/userdata/1087/papers/0307.pdf

Li, Jundong, Cheng, Kewei, Wang, Suhang, Morstatter, Fred, Trevino, Robert, Tang, Jiliang, & Liu, Huan. (2018). Feature Selection. ACM Computing Surveys, 50(6), 1–45. https://doi.org/10.1145/3136625

Li, William Der-Hsing. (2002). The growth of mass home ownership in Taiwan. Journal of Housing and the Built Environment, 17(1), 21–32. https://doi.org/10.1023/A:1014812822953

Liemohn, Michael W, Shane, Alexander D, Azari, Abigail R, Petersen, Alicia K, Swiger, Brian M, & Mukhopadhyay, Agnit. (2021). RMSE is not enough: Guidelines to robust data-model comparisons for magnetospheric physics. Journal of Atmospheric and Solar-Terrestrial Physics, 218, 105624. https://doi.org/10.1016/j.jastp.2021.105624

Lin, Y.-R., & Chen, C.-C. (2019). House Price Prediction in Taipei by Machine Learning Models. International Journal of Design, Analysis and Tools for Integrated Circuits and Systems, 8(1), 89–94. http://search.proquest.com.upc.remotexs.xyz/docview/2316725356?accountid=43860

Lin, Hong & Chen, Kuentai. (2011). Predicting price of Taiwan real estates by neural networks and support vector regression. 220-225.

Liu, Yang. (2008). Box plots: use and interpretation. Transfusion (Philadelphia, Pa.), 48(11), 2279–2280. https://doi.org/10.1111/j.1537-2995.2008.01925.x

Liu, Lili, Zhang, Li, Feng, Huawei, Li, Shimeng, Liu, Miao, Zhao, Jian, & Liu, Hongsheng. (2021). Prediction of the Blood–Brain Barrier (BBB) Permeability of Chemicals Based on Machine-Learning and Ensemble Methods. Chemical Research in Toxicology, 34(6), 1456–1467. https://doi.org/10.1021/acs.chemrestox.0c00343

Mantas, C. J., Castellano, J. G., Moral-García, S., & Abellán, J. (2019). A comparison of random forest based algorithms: random credal random forest versus oblique random forest. Soft Computing (Berlin, Germany), 23(21), 10739–10754. https://doi.org/10.1007/s00500-018-3628-5

Mars storm, AI ethics and the LHC's big upgrade. (2018). Nature (London), 558(7710), 348–349. https://doi.org/10.1038/d41586-018-05468-4

Miller, Norman G, Peng, Liang, & Sklarz, Michael A. (2011). The Economic Impact of Anticipated House Price Changes—Evidence from Home Sales. Real Estate Economics, 39(2), 345–378. https://doi.org/10.1111/j.1540-6229.2010.00292.x

Ministry of the Interior, R.O.C. (2011, March 5) *The proportion of multiple houses owned by legal persons and short-term transactions are significantly higher than those of natural persons* https://www.moi.gov.tw/News_Content.aspx?n=4&s=213457

Murphy, K. P. (2012). Machine learning [electronic resource] : a probabilistic perspective. MIT Press.

Nijkamp, Peter & Rietveld, Piet & Vlist, Arno & Gorter, Cees. (2002). Residential Mobility and Local Housing-Market Differences. Environment and Planning A. 34. 1147-1164. 10.1068/a34176.

Osborne, J. W. (2013). Best practices in data cleaning [electronic resource] : a complete guide to everything you need to do before and after collecting your data. SAGE.

Paris, Chris. (2009). Re-positioning Second Homes within Housing Studies: Household Investment, Gentrification, Multiple Residence, Mobility and Hyper-consumption. Housing, Theory, and Society, 26(4), 292–310. https://doi.org/10.1080/14036090802300392

Perez, H., & Tah, J. H. M. (2020). Improving the Accuracy of Convolutional Neural Networks by Identifying and Removing Outlier Images in Datasets Using t-SNE. Mathematics (Basel), 8(5), 662. https://doi.org/10.3390/math8050662

Pittard, W. Stephen, & Li, Shuzhao. (2020). The Essential Toolbox of Data Science: Python, R, Git, and Docker. Computational Methods and Data Analysis for Metabolomics, 2104, 265–311. https://doi.org/10.1007/978-1-0716-0239-3_15

Rebala, G., & Churiwala, Sanjay & Ravi, Ajaay. (2019). An introduction to machine learning. Springer.

Ren, Ye, Suganthan, P.N, & Srikanth, N. (2015). Ensemble methods for wind and solar power forecasting—A state-of-the-art review. Renewable & Sustainable Energy Reviews, 50, 82–91. https://doi.org/10.1016/j.rser.2015.04.081

Ribeiro, Matheus Henrique Dal Molin, & dos Santos Coelho, Leandro. (2020). Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series. Applied Soft Computing, 86, 105837. https://doi.org/10.1016/j.asoc.2019.105837
Riley, Sarah F, Nguyen, Giang, & Manturuk, Kim. (2015). House price dynamics, unemployment, and the mobility decisions of low-income homeowners. Journal of Housing and the Built Environment, 30(1), 141–156. https://doi.org/10.1007/s10901-014-9400-y

Rohani, Abbas, Taki, Morteza, & Abdollahpour, Masoumeh. (2018). A novel soft computing model (Gaussian process regression with K-fold cross validation) for daily and monthly solar radiation forecasting (Part: I). Renewable Energy, 115, 411–422. https://doi.org/10.1016/j.renene.2017.08.061

Sagi, O., & Rokach, L. (2021). Approximating XGBoost with an interpretable decision tree. Information Sciences, 572, 522–542. https://doi.org/10.1016/j.ins.2021.05.055

Sahay, A. (2016). Applied regression and modeling : a computer integrated approach (First edition.). Business Expert Press.

Seni, G., & Elder, John F. (2010). Ensemble methods in data mining [electronic resource] : improving accuracy through combining predictions. Morgan & Claypool Publishers.

Shaikhina, Torgyn, Lowe, Dave, Daga, Sunil, Briggs, David, Higgins, Robert, & Khovanova, Natasha. (2019). Decision tree and random forest models for outcome prediction in antibody incompatible kidney transplantation. *Biomedical Signal Processing and Control*, 52, 456–462. https://doi.org/10.1016/j.bspc.2017.01.012

Swalin, A. (2018). Choosing the Right Metric for Evaluating Machine Learning Models — Part 1. Medium, 1–11. Retrieved from https://medium.com/usf-msds/choosing-the-right- metric-for-evaluating-machine-learning-models-part-2-86d5649a5428

Ting, K. M., & Witten, I. H. (1999). Issues in stacked generalization. The Journal of Artificial Intelligence Research, 10, 271-289. doi:http://dx.doi.org.sheffield.idm.oclc.org/10.1613/jair.594

Truong, Q., Nguyen, M., Dang, H., & Mei, B. (2020). Housing Price Prediction via Improved Machine Learning Techniques. Procedia Computer Science, 174(2019), 433–442. https://doi.org/10.1016/j.procs.2020.06.111

Wang, S.-Q., Yang, J., & Chou, K.-C. (2006). Using stacked generalization to predict membrane protein types based on pseudo-amino acid composition. Journal of Theoretical Biology, 242(4), 941–946. https://doi.org/10.1016/j.jtbi.2006.05.006

Wang, Yun. (2013). On efficiency properties of an R-square coefficient based on final prediction error. Statistics & Probability Letters, 83(10), 2276–2281. https://doi.org/10.1016/j.spl.2013.06.021

Witten, I. H., Frank, Eibe, & Hall, Mark A. (2011). Data mining [electronic resource] : practical machine learning tools and techniques (3rd ed.). Elsevier/Morgan Kaufmann.

Wolpert, David H. (1992). Stacked generalization. Neural Networks, 5(2), 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1

Wong, S. K., Chau, K. W., Yau, Y., & Cheung, A. K. C. (2011). Property price gradients: the vertical dimension. Journal of Housing and the Built Environment, 26(1), 33–45. https://doi.org/10.1007/s10901-010-9203-8

Willmott, Cort J, & Matsuura, Kenji. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Climate Research, 30(1), 79–82. https://doi.org/10.3354/cr030079

Xiong, Shilong, Sun, Qibo, & Zhou, Ao. (2020). Improve the House Price Prediction Accuracy with a Stacked Generalization Ensemble Model. In Internet of Vehicles. Technologies and Services Toward Smart Cities (pp.

382–389). Springer International Publishing. https://doi.org/10.1007/978-3-030-38651-1_32

Xu, Xiaoqing Eleanor, & Chen, Tao. (2012). The effect of monetary policy on real estate price growth in China. Pacific-Basin Finance Journal, 20(1), 62–77. https://doi.org/10.1016/j.pacfin.2011.08.001

Yu, Chien-Ming, & Chen, Pei-Fen. (2018). House prices, mortgage rate, and policy: Megadata analysis in Taipei. Sustainability (Basel, Switzerland), 10(4), 926. https://doi.org/10.3390/su10040926

Zhang, X., Deng, T., & Jia, G. (2020). Nuclear spin-spin coupling constants prediction based on XGBoost and LightGBM algorithms. Molecular Physics, 118(14), e1696478. https://doi.org/10.1080/00268976.2019.1696478

Zhang, J., Mucs, D., Norinder, U., & Svensson, F. (2019). LightGBM: An Effective and Scalable Algorithm for Prediction of Chemical Toxicity–Application to the Tox21 and Mutagenicity Data Sets. Journal of Chemical Information and Modeling, 59(10), 4150–4158. https://doi.org/10.1021/acs.jcim.9b00633

Zhang, Hang, Wang, Kehan, Li, Mengchu, He, Xinchen, & Zhang, Ruibo. (2020). House Price Prediction with An Improved Stack Approach. Journal of Physics. Conference Series, 1693(1), 12062. https://doi.org/10.1088/1742-6596/1693/1/012062

Zhang, Kai, & Luo, Minxia. (2015). Outlier-robust extreme learning machine for regression problems. Neurocomputing (Amsterdam), 151(3), 1519–1527. https://doi.org/10.1016/j.neucom.2014.09.022

Appendix

Python code

```
import pandas as pd
from mlxtend.regressor import StackingRegressor
from sklearn.linear_model import LinearRegression, Ridge, Lasso
from sklearn.model_selection import train_test_split
import numpy as np
import matplotlib.pyplot as plt
import lightgbm as lgb
import xgboost as xgb
import seaborn as sns
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import cross_val_score, GridSearchCV,
KFold


# read csv file, datatype = Dataframe
df = pd.read_csv('taipei_city_real_estate_transaction_v2 2.csv')

categorical_col = ['district', 'transaction_type', 'urban_land_use',
'main_building_material',
                 'transaction_year',
'carpark_category','transaction_month']

numerical_col =
['land_shift_area','building_shift_total_area','num_room','num_hall',
'num_toilet','unit_ntd','carpark_ntd','building_age','number_of_land'
,'number_of_building','number_of_carpark','carpark_shift_area']

# selected features
features = ['main_use','district', 'transaction_type',
'urban_land_use',
'main_building_material','transaction_year','transaction_month',
'carpark_category','building_age','number_of_building','building_shif
t_total_area','num_hall','num_toilet','number_of_land',
'num_room','land_shift_area']
```

```
### getting to know data ###
print(df.shape)

# checking missing values => no missing values
def MissingValue(df):
    miss_value = df.isnull().sum()
    miss_percentage = miss_value / df.shape[0]
    miss_df = pd.concat([miss_value, miss_percentage], axis=1)
    miss_df =
miss_df.rename(columns={0:'MissingValue',1:'%MissingPercent'})
    miss_df = miss_df.loc[miss_df['MissingValue']!=0, :]
    miss_df = miss_df.sort_values(by='%MissingPercent', ascending =
False)
    return miss_df

print(MissingValue(df))

# exploring the relationship between attributes and the target
attribute
g = sns.pairplot(x_vars=['land_shift_area'], y_vars=['district'],
data=df)
g.fig.set_size_inches(15,10)
plt.show()

### data pre-processing ###
# find and deal with outliers
def outlier_treatment(datacolumn):
     sorted(datacolumn)
     Q1,Q3 = np.percentile(datacolumn , [25,75])
     IQR = Q3 - Q1
     lower_range = Q1 - (1.5 * IQR)
     upper_range = Q3 + (1.5 * IQR)
     return lower_range,upper_range

lowerbound,upperbound = outlier_treatment(df.unit_ntd)
building_shift_area_lowerbound,building_shift_area_upperbound =
outlier_treatment(df.building_shift_total_area)

# upper_threshold = df['unit_ntd'].quantile(0.99)
# lower_threshold = df['unit_ntd'].quantile(0.01)
# building_shift_area_upper_threshold =
df['building_shift_total_area'].quantile(0.99)
# building_shift_area_lower_threshold =
df['building_shift_total_area'].quantile(0.01)
#
print(building_shift_area_upper_threshold,building_shift_area_lower_t
hreshold)

# # remove outliers
df = df[(df.unit_ntd>lowerbound)&(df.unit_ntd<upperbound)]
df =
df[(df.building_shift_total_area>building_shift_area_lowerbound)&(df.
building_shift_total_area<building_shift_area_upperbound)]

# remove agriculture land use form data
df = df[df.urban_land_use != 'Agriculture']

# one hot encoding to the categorical attributes
categorical_feature_mask = df.dtypes==object
categorical_cols = df.columns[categorical_feature_mask].tolist()
labelencoder = LabelEncoder()
df[categorical_cols] = df[categorical_cols].apply(lambda col:
labelencoder.fit_transform(col.astype(str)))
# print(df[features].head())

# correlation heatmap
plt.figure(figsize=(30,15))
sns.heatmap(df.corr(),cmap='coolwarm',annot = True)
plt.show()

# scatter plot
plt.figure(figsize=(12,6))
plt.scatter(x=df.land_shift_area, y=df.unit_ntd)
plt.xlabel("land_shift_area", fontsize=13)
```

```python
plt.ylabel("total_ntd", fontsize=13)
plt.ylim(0,800000)
plt.show()

# normalization
cols_to_norm = ['land_shift_area']
df[cols_to_norm] = StandardScaler().fit_transform(df[cols_to_norm])

# change the calendar from Taiwan local calendar to Gregorian
calendar
df['complete_year'] = df['complete_year'].astype(int) + 1911



def change_word(x):
    if x == 'Address':
        return 'Residence'
    elif x == 'Quotient':
        return 'Business'
    else:
        return x

df['urban_land_use'] = df['urban_land_use'].apply(lambda x:
change_word(x))

# check distribution
sns.distplot(df['building_shift_total_area'])
plt.show()

# # print the unique values of each attributes
for ind, col in enumerate(categorical_col):
    print("Unique values of {}: {} \n".format(col, set(df[col])))

# bar plots
plt.figure(figsize = (20, 30))
for ind, col in enumerate(categorical_col):
    plt.subplot(3, 3, ind+1)
    df[col].value_counts().plot(kind='bar')
    plt.xlabel(col, size=10)
    plt.ylabel("counts")
    plt.tight_layout() # to avoid graph overlapping
plt.show()

# define features and target
X=df[features]
y=df['unit_ntd']

# split dataset into train and test data (7:3)
x_train, x_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=7)
# K-fold cross validation
def CV(model):
    folds = KFold(n_splits = 10, shuffle = True, random_state = 11)
    scores_MAE = cross_val_score(model, x_train, y_train,
scoring='neg_mean_absolute_error', cv=folds)
    scores_RMSE = cross_val_score(model, x_train, y_train,
scoring='neg_root_mean_squared_error', cv=folds)
    scores_R2 = cross_val_score(model, x_train, y_train,
scoring='r2', cv=folds)

    score_MAE=0
    score_RMSE=0


    for i in scores_MAE:
        score_MAE=i+score_MAE

    for i in scores_RMSE:
        score_RMSE=i+score_RMSE




    print("MAE_CV:"+str(abs(score_MAE/10)))
    print("RMSE_CV:" + str(abs(score_RMSE / 10)))
    print("RMSE_R2:" + str(scores_R2))
    print("-----------------")
```

```
# #
# #
######################################################################
#############################
# # models
# single methods
lr = LinearRegression()
ridge = Ridge(random_state=2019)
las = Lasso()

# ensemble methods
lgb = lgb.LGBMRegressor()
xgb = xgb.XGBRegressor()
rf = RandomForestRegressor()

models = [lr, ridge, las, lgb, rf, xgb]
print('base model')
for model in models:
    model.fit(x_train, y_train)
    pred = model.predict(x_test)
    print(model)
    CV(model)


sclf = StackingRegressor(regressors=models, meta_regressor=lr)
sclf.fit(x_train, y_train)
pred = sclf.predict(x_test)

print('stacking model')
CV(sclf)
```