How Much Should I Pay: Modeling the Property Prices in Singapore

Project Report for CS5228 Knowledge Discovery and Data Mining

Ziji Shi
A0230314Y
National University of Singapore
Team Good Miners
zijishi@comp.nus.edu.sg

Yutong Xia
A0249309X
National University of Singapore
Team Good Miners
yutong.xia@u.nus.edu

Ronghua Zhu
A0215599N
National University of Singapore
Team Good Miners
ronghuazhu@u.nus.edu

***Abstract*—In this project, we build a predictive model for the resale price of condominiums in Singapore. Through extensive exploratory data analysis, we investigate the relationship between the features, and we find some problems including missing values and outliers. We then perform data transformation to clean the dataset. We augment the original dataset using the landmarks information, which improves the model performance. We also explore 3 types of models to predict the housing price: gradient boosted trees, random forests, and ridge regression. We observe that, despite XGBoost has the lowest validation loss, random forest outperforms the rest on the test set in our setting. To deal with the exploding hyper-parameter search space, we also explore Bayesian Optimization. We believe this study can provide more insights to the factors that impact the property price, and help the future policy-making. Our codes are available on Github: https://github.com/StevenShi-23/CS5228-Data-Mining.**

***Index Terms*—Data Mining, Bayesian Optimization, Random Forest, Ridge Regression, Gradient Boosted Trees**

I. Introduction and Motivation

Private home prices rose 10.6% in 2021 [3] , according to real estate agencies Knight Frank and JLL. Singapore introduced new measures aimed at cooling the private and public residential property market in mid-December 2021, including higher taxes on second and subsequent property purchases and tighter lending restrictions. Lower interest rates, limited supply, and strong demand are factors driving up house prices.

In this project, we examine the factors that affect the value of condominiums in Singapore. Homeownership in Singapore is expensive, and there are many stakeholders in the condo market. For the government, by understanding the factors that affect the value of apartments, it can better regulate the price of the real estate market effectively. It is conducive to the proposal of relevant measures and better maximizes the impact of suitable measures.

For buyers, by understanding the factors that affect the value of the apartment, they can better choose the type of apartment they are satisfied with according to their property, including the number of rooms in the apartment, the location of the apartment, etc. It promotes buyers to maximize their own interests by influencing their decisions.

For sellers, by understanding the factors that affect the value of an apartment, they can effectively evaluate their own home. The seller can sell the apartment at a reasonable price without incurring a large loss.

Promoting the rationality of transaction prices for the entire apartment market will better promote market activity.

To solve this problem, we explored three regression models: Gradient Boosted Trees, Random Forest, and Ridge regression, and we observed that Random Forest performed the best. We used public facility information to expand the dataset. We also recovered the missing values as much as possible by building a lookup table. In terms of hyperparameter tuning, we chose Bayesian Optimization over grid search.

Our contributions include the following:

- We investigate the importance of factors in determining the housing price, finding that *area size* and *location* are the most important features;
- We expand the dataset with proximity information to the nearby public facilities, and we observe a significant improvement in the model performance;
- We explore 3 types of models for this regression tasks, using grid search and Bayesian Optimization for hyperparameter optimization. We find that random forest performs the best and Bayesian Optimization is a more effective search method when the search space is very large;
- We visualize the housing price with respect to the geographical information, and it is accessible at https://cs5228-demo.netlify.app/.

II. Exploratory data analysis

The core dataset of apartment sales prices is collected from SRX. Contains detailed properties related to the apartment, such as sale price, property name, number of bedrooms, number of bathrooms, floor area, etc.

The integrated final data has 25718 rows and 36 columns. Numerical values include *bathrooms, built year, no of units, area size, date listed, price, bedrooms, tenure remains, tenure year, comme min distance, comme no 1km, comme no 3km,gov-m min distance, gov-m no 1km, gov-m no 3km, shopp min distance, shopp no 1km, shopp no 3km, secon min distance, secon no 1km, secon no 3km, prima min distance, prima no*

1km, prima no 3km, train min distance, train no 1km, train no 3km.

Category values include *listing id, name, street, type, district, region, planning area, subszone*.

Before going into predicting house prices, we first performed a descriptive analysis to explore our input data. This allows us to better understand the distribution of input features and the relationship between features and house prices.

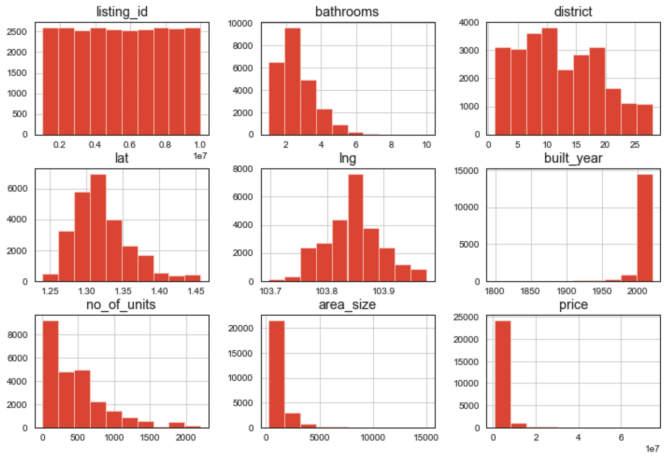

Fig. 1. Distribution of attributes from the original dataset.

A. Dependent Variable

We have the numerical variable '*price*' as our dependent Variable. It represents the sale price of the house.

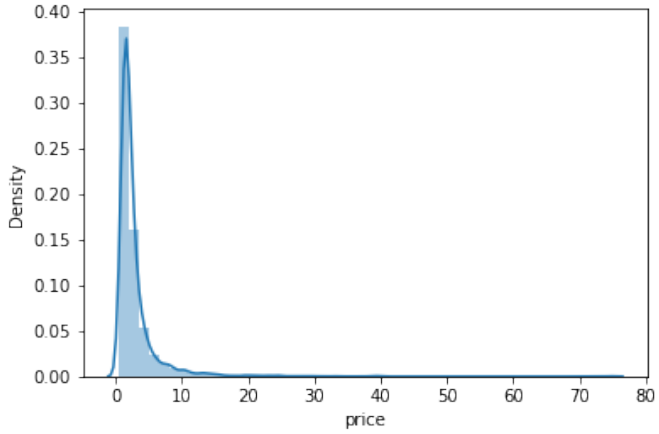

Fig. 2. Distribution of price.(Unit: million)

TABLE I
DESCRIPTION OF THE VARIABLE PRICE

count	*mean*	*std*	*min*	*max*
26048	2.99	4.32	0.56	74.80

[a]Unit: million singapore dollars.

By calculating the Kurtosis and Skewness of the price distribution, the Kurtosis is 7.793932, and the Skewness is 89.321184. It is found that the distribution of price is steeper than the peak of the normal distribution, and the Skewness value is more significant, which is right-skewed.

B. Independent Variable

We can see that as the bathroom increases, the house price increases. When *bathroom*=10, the average price is lower than *bathroom*=8. Judging him as a possible outlier can be checked in the following analysis. At the same time, when *bathroom*=3,4,5,6, there are many outliers that are higher than the upper limit.

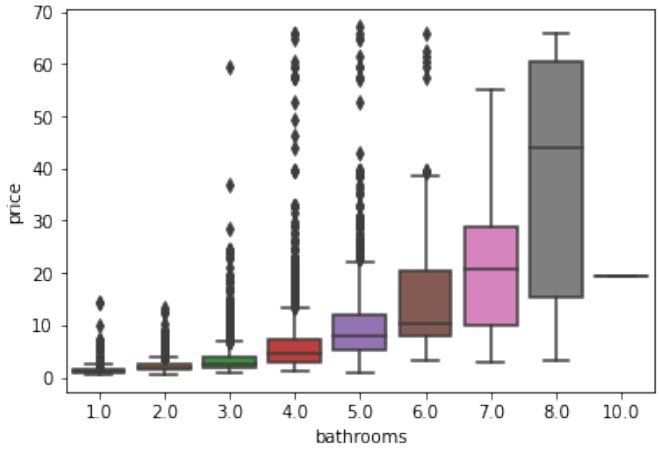

Fig. 3. Distribution of bathrooms.

We can see that the classification of *bedrooms* is a bit mixed, and there are values such as '3+1'. The type of a bedroom needs to be re-divided according to the meaning of a bedroom.

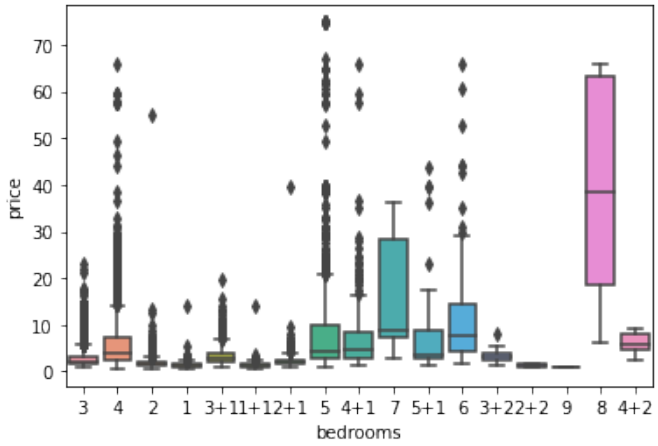

Fig. 4. Distribution of bedrooms.

We found that the average price of houses in different districts is not very different. But there are also some phenomena that the price of this house in the same district is much higher than the price of other houses. There may be other factors that need to be further explored.

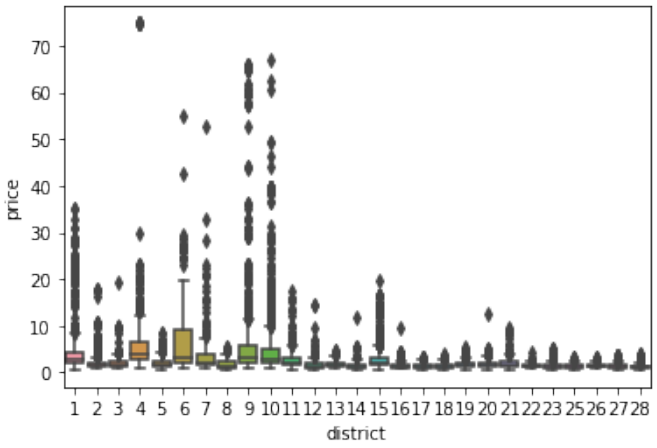

Fig. 5. Distribution of district.

We can see that the built year of houses is concentrated between 2010 (upper quartile) and 2016 (lower quartile). The earliest year was 1799, and there were also some house sales between 1900 and 1950. At the same time, the existence of an outlier, which is higher than all other prices in the year, needs to be investigated.

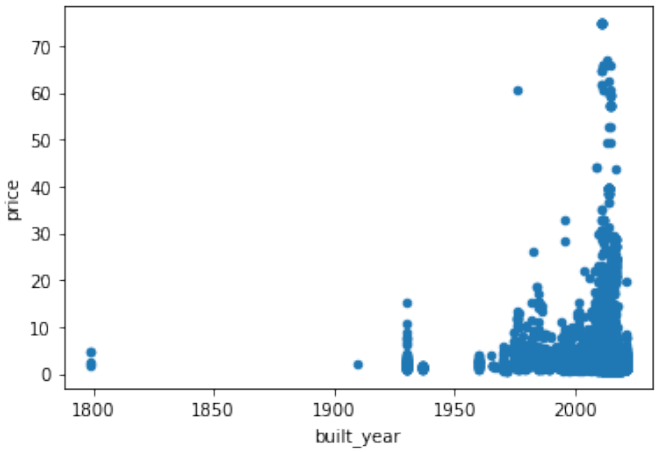

Fig. 6. Distribution of built year.

We can find that the larger the *area size*, the higher the transaction price of the house. But there are also outliers with area sizes between 10,000 and 12,000 and prices lower than 10 million. There are also outliers with area sizes between 0 and 1,000 and prices higher than 10 million. We need further investigation.

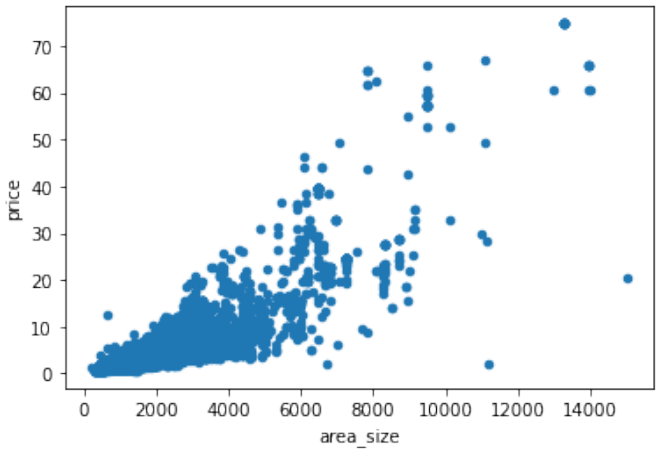

Fig. 7. Distribution of area size.

We found that the number of *units* is concentrated between 0-1000. With the increase in the number of units, there is no apparent positive correlation between the transaction price of the house. Between 0-100units, there is a phenomenon that the house price is much higher than other houses.

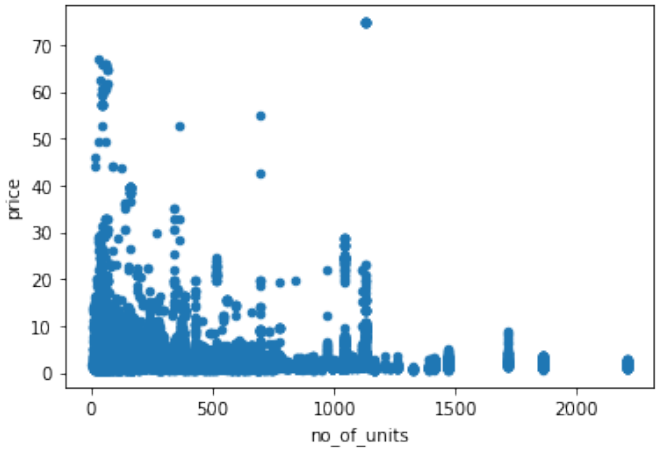

Fig. 8. Distribution of number of units.

Finally, we observed the correlation between variables, and the lighter the color block, the stronger the correlation. It is found that *area size*, *bathrooms*, and *bedrooms* are strongly correlated with price. *District* and *number of units* have a weak correlation with price.

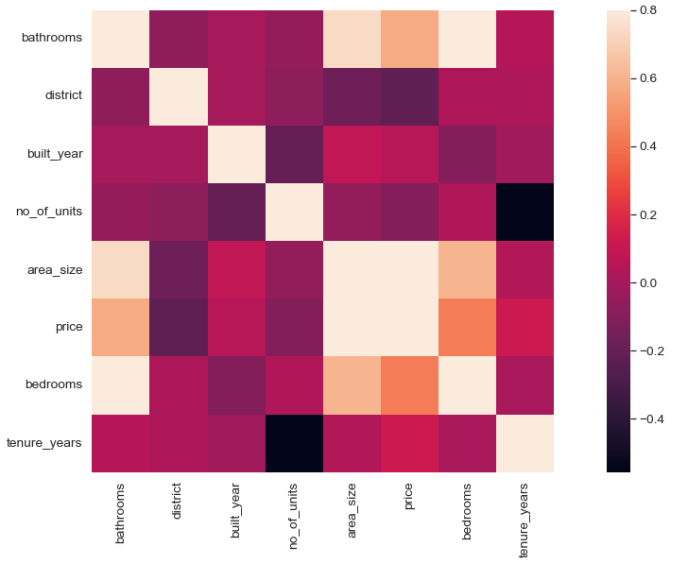

Fig. 9. Correlation plot between variables.

C. Geographical information visualisation

To further explore the relationship between target variable, i.e. *price*, and the geographical location of different landmarks, including *commercial centres*, *primary schools*, *secondary schools*, *shopping malls*, *government markets hawker centres* and *train stations*, we generated an interactive web page[1] as in Figure 10 showing the geographical distribution of properties and six different landmarks in Singapore. Via exploring the distribution of landmarks, there are several intuitive findings. For example, according to the distribution of train stations, they are concentrated in the central area of Singapore. At the same time, the mean housing price in this region is the highest among the other five regions, which indicates the correlation between the locations of landmarks and the housing price. These intuitive points emphasize the importance of adding landmark-related attributes for modelling and predicting price (see in Sec. III-F).

III. Data pre-processing

A. De-duplication

Duplicate in the training set is undesirable since it may contaminate the evaluation. For instance, if 4 out of 5 records were duplicates, even if we could only correctly predict half of the unique records, we may still get an high accuracy of 80%. Therefore, we must remove duplicate records to make sure the evaluation is not biased towards frequent records.

In this project, we first count the duplicate records, finding that initially there seems to be no perfect duplicates because every record has a different *listing_id*. However, if we ignored the *listing_id* attribute, some records are duplicate to each other. We think it is very unlikely to have everything to be the same for a unit, and then remove those duplicates based on comparison on the rest of the attributes.

After deduplication, the number of records in training set becomes 25718 from 26048.

[1]The web page is accessible at https://cs5228-demo.netlify.app/ You can change the map mode and the landmark types showing in the map by clicking the button in the upper right corner. To check the details of a point, please click it.

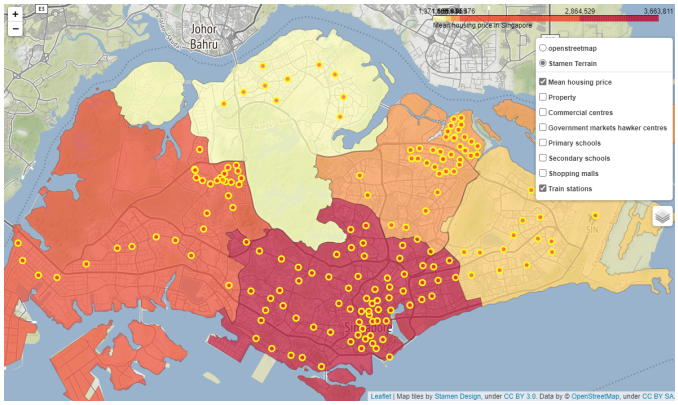

Fig. 10. Interactive web page that provides the locations of properties and six different landmarks.

B. Missing values

After de-duplicatation, we find that many of the records have missing values. Although some model are capable of handling missing values (for instance, decision tree can treat missing value as a unique value), having too many missing values may not provide enough useful information to the model and lead to worse result. We therefore decide to recover the missing values as much as possible by building an lookup table.

We take the *built year* attribute as an example. The completion algorithm works in the following way:

1) Build a lookup table for all unique values with respect to property *name*
2) Check if the missing value can be found from the lookup table
   - If true, use the value found
   - Else, go to step 3
3) Check if a non-NaN mode number exists from lookup table
   - If true, use the mode number
   - Else, go to step 4
4) If no such record can be found, we manually collect the value from online.
   - If the property has a matching record from SRX, we use the value
   - If there are very few transactions or multiple mode numbers, but a similar property exists with close proximity and built year (< 10 years), we choose the median value of that property to fill
   - If no transaction records are found, or the property name is just a road name, we leave it as NaN.

This way, we manage to reduce the number of missing *built year* from 10043 to 73.

C. Feature selection

1) Discovery of variable meanings.
   a) The meaning of "*bedroom*": Because there is a form similar to "3+1" in the data, according to the meaning, turn this form into a numerical variable. "3+1" means 3 bedrooms, 1 study, 40-100 square feet on average, more than 3 bedrooms. "3+2" means 3 bedrooms, 2 dens, which are usually small. So we regard the area of 1 study as 0.5 study, and turn "3+1" into 3.5 bedrooms.
   b) The meaning of "*bedroom*": The '*tenure*' variable takes values like '99 years from 18/04/2016' or 'freehold'. We turn the '*tenure*' variable into a numerical value by studying its meaning. 3 types of condo tenures in Singapore are Freehold, 99-year leasehold, 999-year leasehold. The 999-year leasehold developments are in essence freehold properties. So the '*tenure*' variable can be changed into two categories, 99 and 999 years. Freehold properties can be held indefinitely by the buyer, while 99-year leasehold properties will revert back to the state after the tenure ends.
   If the tenure value contains a specific date, subtract the specific year from the current 2022 to get the year difference. Finally, subtract the the years difference from the year displayed by the tenure value to get the real tenure of the property in 2022.
   c) The meaning of *model* and *type*: *model* is a subdivision of type. However, except for condo and apartment, the values of model are very small and can be ignored. So we can delete the *model* and use the type variable.
2) Data reduction.
   We observe unique values in the data. There is only one value for the *market segment*, *type of area*, *eco category*, and *accessibility*. We delete these variables.

D. Outlier removal

We plot the numerical value and output the maximum and minimum values. Go through google max and min to see if it's due to a typo etc. or if it's the real situation.

The reasonable value is *street* ='marina way',*built year* =1799,*area size*=15000. Outliers are *bedrooms*=9, *bathrooms*=10, *no of units*=1, *area size*=226. We should delete them.

E. Feature encoding

Features in the dataset contain both numerical and categorical variables. Numerical variables can be modelled directly in regression. While categorical variables are transformed into dummy variables by the one-hot encoding technique, including *type*, *region*, *district*, and *tenure_years*.

F. Landmarks information

In addition, considering the convenience of an accommodation can influence the price, we used the locations of landmarks,including commercial centres, primary schools, secondary schools, shopping malls, government markets hawker centres and train stations, to represent the convenience of a listing. Vincenty distance is a method used in geodesy to calculate the distance between two points on the surface of a spheroid [7]. By calculating the Vincenty distance between each property and each landmark, we got six distance matrices. Based on these matrices, we generated three new attributes for each kind of landmark:

- Distance to the nearest 'landmarks'
- The number of 'landmarks' within 1 kilometer
- The number of 'landmarks' within 3 kilometers

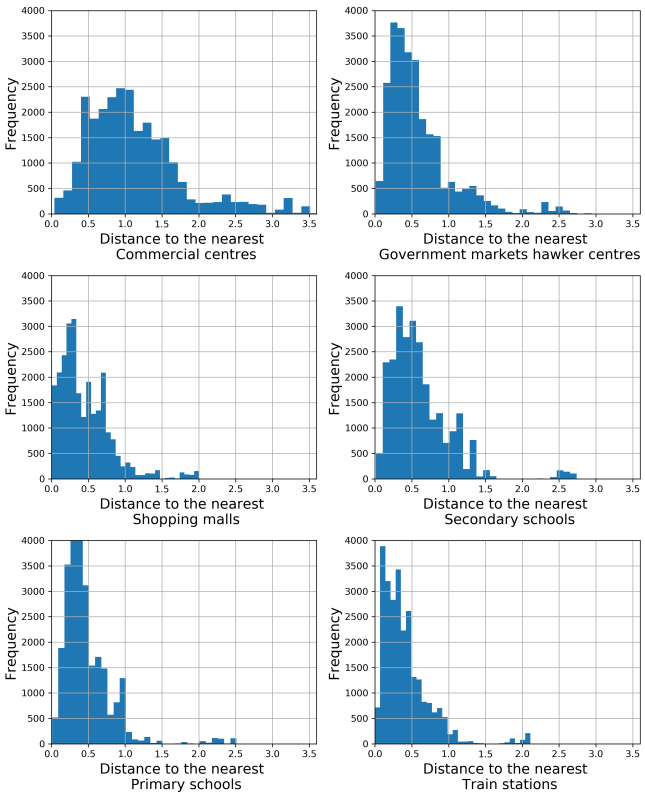

Fig. 11. Distribution of the distance to the nearest landmarks

Fig. 11 shows the distribution of distances to the nearest landmarks. According to the histograms, the commercial centres are relatively evenly distributed, indicating that the difference between properties to the nearest commercial centres is not very considerable. However, for the other five landmarks, although the most frequent distance (roundly 0.3 kilometres) is smaller than that of commercial centres (1 kilometre), there are some properties located in extreme inconvenient locations.

IV. Data Mining Methods

A. Gradient Boosted Tree

Gradient Boosted Tree (GBT) is a type of decision tree algorithm that uses sample reweighting to adaptively select samples and recompute the tree, which is also called boosting.

Samples that were misclassified will have a higher chance of getting selected in the future iterations.

XGBoost [1] is an efficient library of GBT algorithm. We use it in our project.

B. Random Forest

The random Forest (RF) algorithm, first proposed by [1], is a collection of decision trees, presenting a random perturbation while training, which has been widely employed in lots of research fields, such as predicting bank failures [6] and firm vulnerability analysis [5].

C. Ridge Regression

In addition to above two ensemble algorithms, for comparison purpose we select the ridge regression to predict the target value as well, since this method is also commonly used when it comes to predicting housing price [2]. In ridge regression, a tuning parameter α is added to address the effect of multiple variables in linear regression which is usually referred as noise in statistical context.

V. Experiments

A. Dataset preparation

1) Normalization: Normalizing data before training the model can speed up the learning process [4]. Therefore, we normalized the numerical variables before training the models using zero-centered mean and unit variance.

2) Training and Validation set: We partitioned the training set into the training set and the validation set at the ratio of 4:1. After splitting, the training set and the validation set have 20574 and 5144 samples, respectively.

B. Hyper-Parameters Optimization (HPO)

The selection of hyper-parameters is of great significance and on which the predictive performance of machine learning models largely depends.

Table II shows the crucial hyper-parameters in the three models, their searching space and the best combinations of hyper-parameters adopting the Grid Search method.

1) Grid Search: For XGBoost model, the search space for hyperparameter is very large because there are various types of hyperparameters, each of them having multiple candidate values. Together they constitute a very large search space.

We first tackle the hyperparameter search problem using Grid Search. But we found that this approach will lead to exploding number of hyperparameters to find. For example, if we search for 8 hyperparameters, each of them having 10 values, we will have in total 10^8 candidates. Together with k-fold cross validation (where we set k=5), we will have 500 Million experiments. Considering each experiment takes approximately 0.3 seconds, it will cost us 1736 days, which is 4.75 years. And we only touch a small portion of hyperparamters. This is unrealistic to tune within feasible time.

We must find a more efficient way to tune the hyperparameters. Therefore, we choose Bayesian Optimization.

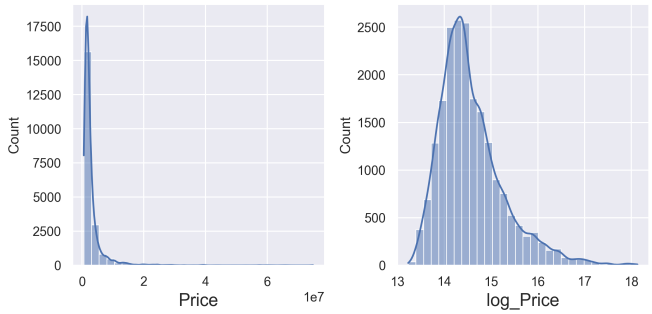

Fig. 12. Distribution of target variable *price* before and after log-transformation.

2) Bayesian Optimization: Bayesian Optimization (BO) is a method that leverage Bayesian Probability theory to find the best parameter for a machine learning algorithm. We used the Python library *HyperOpt* for this tasks.

It consists of 4 steps:

1) Initialize search space. We define the search space to be the possible values of our model's hyperparameters.
2) Define objective function. We define it to be the negative RMSE loss.
3) Choose optimization algorithm. We use *tpe.suggest*, which uses a surrogate model that estimate the final performance without actually running the experiments.
4) Execute BO.

We observe that one benefit of BO over grid search is that we can better control the the computational resources quota for HPO, which automates the "choose hyper-parameter $\rightarrow$ experiment $\rightarrow$ analyse result" feedback loop.

C. Log-transformation of the target variable

According to Figure II-A, it is evident that there is a considerable left-skewness in the distribution of dependent variable, i.e. *price*. For the regression task, a normal-like distributed variable may help linearize the relationship between independent variables and dependent variables. Figure 12 shows the distribution of the target variable before and after log-transformation, which indicates that applying log-transformation on the target variable makes it closer to normally distributed. To test the efficiency of the log-transformation on the dependent variable, we use both *price* and *log_price* to trained the models.

VI. Results

A. Model evaluation and comparison

We adopted root mean square error (RSME) values to evaluate and compare models' predictive performances. According to Table III, there are several interesting observations. First, the performance of the XGBoost and random forest models is considerably better than that of Ridge regression models, which indicates the advantage of using ensemble methods in this case that ensemble methods may reduce the generalization error of the prediction and minimise the errors caused by

TABLE II
HYPER-PARAMETERS TUNING

Model	Dependent Var	Hyper Param	Searching Space	Best Param
Gradient Boosted Tree	y	learning_rate	[0.07, 0.09, 0.11, 0.13]	0.11
		max_depth	[8,10,12,14]	10
		gamma	[6,8,10]	8
		colsample_bytree	[0.9, 0.95, 1.0, 1.05]	1
		n_estimators	[100, 150, 200]	150
Ridge	y	alpha	[0.01, 0.1, 0.5, 1, 2, 5, 10, 50, 100]	100
	log_y	alpha	[0.01, 0.1, 0.5, 1, 2, 5, 10, 50, 100]	10
Random Forest	y	n_estimators	[10, 50, 200, 300, 500]	300
		max_depth	[10, 30, 50, 70, 100]	50
		max_features	[0.1, 0.3, 0.5, 0.7, 0.9, 0.99]	0.9
	log_y	n_estimators	[10, 50, 200, 300, 500]	500
		max_depth	[10, 30, 50, 70, 100]	30
		max_features	[0.1, 0.3, 0.5, 0.7, 0.9, 0.99]	0.7

TABLE III
RMSE ON PREDICTED PRICE

	Dependent var	Validation RMSE
XGBoost	y	600748.105
Ridge Regression	y	1779979.618
	log_y	3124882.378
Random forest	y	689665.068
	log_y	788818.042

noise. Second, as we discussed in Sec. V-C, we also explored the effectiveness of log transformation on the target variable. The higher RSME values for the Ridge Regression model and Random forest model when applying for a log-transformation show that it is not effective in this case, which may be because even the overall price range is large, but most prices are in a small range, i.e., there are several outliers leading to the overall large range which have little influence on training these models.

We also observe that although XGBoost outperforms Random Forest and Ridge on the validation set, Random Forest actually performs better on the test set (i.e., the Kaggle rank). We conjecture that this is because our XGBoost model may overfit on the training set. We noticed that some properties from the test set cannot be found from the training set, therefore, the distribution shift may hinder the performance of the learned XGBoost model. A more detailed evaluation on the overfitting problem requires extensive HPO, and we leave it as future work.

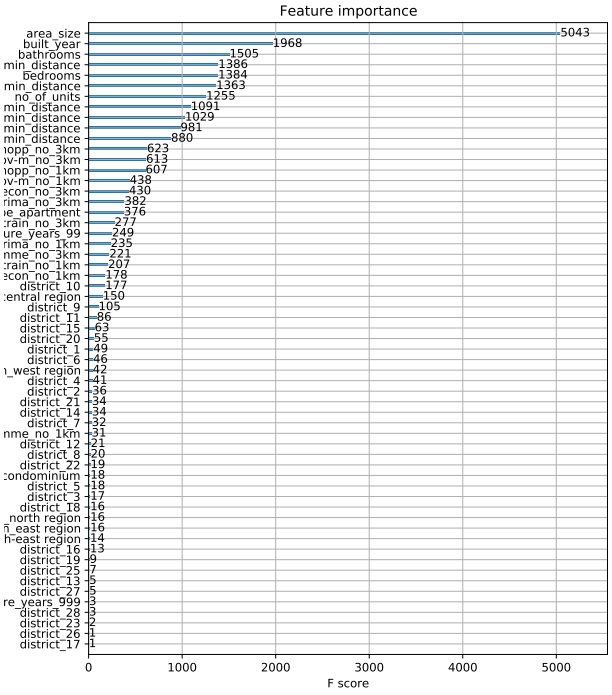

Fig. 13. Feature importance of XGBoost Model.

B. Importance of features

Fig. 13 shows the feature importance of our learned XGBoost model. Out of all attributes, *area_size*, *built_year*, *bathrooms* are top-3 most important features. This highlights the importance of locality in determining the property price in Singapore.

C. Ablation study of feature engineering

In order to understand the effectiveness of our feature engineering efforts, we compare the model performance of the original dataset, feature-engineered dataset, and landmark-augmented dataset and summerize the results in Table VI-C.

We found that both feature engineering and additional landmark information help to improve the final performance. It indicate that feature quality has a significant impact on the model performance.

D. Effect of Different HPO Techniques

We examine the best hyper-parameter found using the two HPO techniques described above. Given similar range and computational budget (4 hours), both of them found a plausible set of hyper-parameters that have a lowest validation loss. We observe that the found hyper-parameter values are very

TABLE IV
ABLATION STUDY OF DIFFERENT VERSIONS OF DATASET. ORIGINAL DATASET IS THE ORIGINAL VERSION FROM KAGGLE WEBSITE. FEATURE ENGINEERED DATASET REFERS TO THE ONE AFTER DATASET PREPARATION. LANDMARK AUGMENTED DATASET REFERS TO THE FEATURE ENGINEERED DATASET WITH ADDED LANDMARK INFORMATION (SEC. III-F).

Dataset	Feature Encoding	De-duplication	Fill Missing Value	Outliers Removal	Add Landmark Info	Validation RMSE score
Original	✓					516711.27175
Feature Engineered	✓	✓	✓	✓		480603.08752
Landmark Augmented	✓	✓	✓	✓	✓	456718.58349

TABLE V
BEST PARAMETER FOUND USING DIFFERENT HPO TECHNIQUES.

HPO Method	Hyperparameter Value				
	learning_rate	gamma	max_depth	colsample_bytree	n_estimators
Grid Search	0.13	6	10	0.95	150
Bayesian Optimization	0.11	9	11	0.98	85

close, except *n_estimators*. This is because when we use BO, we also explore strong model regularization terms (*reg_alpha, reg_lambda*) to prevent overfitting. Therefore, the size of BO-found best model is much small in size (85 v.s. 150) compared to that of the grid search-found counterpart.

VII. CONCLUSION AND DISCUSSION

A. Conclusion

This report mainly explores the housing market in Singapore. We first do the exploration analysis of the SRX apartment sales price dataset used in this report by visualising the distributions and calculating the mean values of attributes. In addition, to further explore the relationship between the housing prices and the geographical distributions of landmarks, we visualized them on an interactive web page (https://cs5228-demo.netlify.app/). Before we set up the experiments, we pre-processed data through de-duplicating, filling missing values, selecting features, removing outliers, encoding the features, and adding landmark-related features. Then we chose three types of models: XGBoost, Random Forest, and Ridge regression models, to predict the price value in the test set and explore the feature importance.

In addition to the prediction task, we aim to answer the following questions: (1) Whether the ensemble models have better predictive performance or not? (2) Do the log transformation on target variables effective and lead to better predictive performance? (3) How does feature selection affect the model's performance and to what extent? (4) Is Bayesian Optimization (BO) a good way for searching the hyperparameters?

In conclusion, we found that (1) Ensemble models perform considerably better than the ridge regression model. (2) The log transformation on target variables may not have a positive influence when predicting property prices in Singapore. (3) Location is one of the most important features when determining the housing price. (4) When the search space is very large, Bayesian Optimization can be an effective search method to speed the searching process.

B. Limitations

1) Hyper-parameter search: Although Bayesian Optimization can find a plausible set of hyper-parameters for a given computational budget, it still takes quite some time. For instance, 3000 trials of HPO took 4 and half hours. As model size and dataset scale beyond this toy example, it will get increasingly harder to find a good solution.

Furthermore, Bayesian Optimization is essentially a sequential process, where parameters for the future rounds depend on the parameters of the past. Therefore, we must explore efficient ways to parallels this process so that it can leverage the multi-core architecture to run in parallel.

On the other hand, although grid search is more computationally heavy, it is embarrassingly easy to parallelize (since the trials do not have inter-dependency). We may leverage a combination of both approaches by first restricting the search space to a smaller one using BO, then do an extensive grid search to find the optimal hyper-parameters.

2) Feature selection: In this study, we encode *region* and *district* using one-hot encoder, and drop other categorical values to keep the feature size to a reasonably small size.

However, location is one of the most important features when determining the property price. We may explore more advanced encoder to encode more detailed locality information, meanwhile keeping the feature size compact.

REFERENCES

[1] CHEN, T., AND GUESTRIN, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (2016), pp. 785–794.

[2] MANASA, J., GUPTA, R., AND NARAHARI, N. Machine learning based predicting house prices using regression techniques. In *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)* (2020), IEEE, pp. 624–630.

[3] NG, A. Singapore property prices are set to keep climbing in 2022, but at a slower pace. In *https://www.cnbc.com/2022/02/07/singapore-property-prices-rents-to-rise-in-2022-but-at-a-slower-pace.html* (2022).

[4] RASCHKA, S. About feature scaling and normalization. *Sebastian Racha. Disques, nd Web. Dec* (2014).

[5] TANAKA, K., HIGASHIDE, T., KINKYO, T., AND HAMORI, S. Analyzing industry-level vulnerability by predicting financial bankruptcy. *Economic Inquiry 57*, 4 (2019), 2017–2034.

[6] TANAKA, K., KINKYO, T., AND HAMORI, S. Random forests-based early warning system for bank failures. *Economics Letters 148* (2016), 118–121.

[7] VINCENTY, T. Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. *Survey review 23*, 176 (1975), 88–93.

APPENDIX A
DISTRIBUTION OF WORKLOAD

A. Shi Ziji

1) Dataset Preparation: He cleaned the dataset by removing the duplicates, filling missing values, and feature encoding based on initial feedback from EDA. He came up with the algorithm for detecting partial duplicates and progressively filling missing values.

2) Model and Experiment: He was in charge of the XGBoost model, including an end-to-end training and evaluation pipeline. He also explored the Bayesian Optimization method for HPO. He repressively tested the different datasets using the XGBoost model to understand the effect of feature engineering.

3) Report Writing: Ziji drafts the experiment and evaluation part related to XGBoost model.

B. Xia Yutong

1) Geographical Information Visualisation: She visualized the locations and other information (e.g., name and street) of properties and landmarks via an interactive web page.

2) Dataset Preparation: She generated three new distance-related attributes via calculating the Vincenty distance between property and landmarks.

3) Model and Experiment: She was responsible for ridge regression and random forest models, including training and evaluating. She also explored the effectiveness of log transformation on the target variable.

4) Report Writing: She wrote the parts in the report related to the above work and the conclusion part.

C. Zhu Ronghua

1) Introduction and Motivation: She introduced the background of this project and motivations to stakeholders.

2) Exploratory data analysis: She visualized data including dependent variable and independent variables. Found some problems of outliers and data transformation for further data pre-processing.

3) Dataset Preparation: She cleaned the dataset by removing outlier and feature selection which includes discovery of variable meanings and data reduction.

4) Report Writing: She wrote I.Introduction, II.Exploratory data analysis and III. Data pre-processing part related to the above work.