CS610 – Applied Machine Learning

Year: 2023-2024 Jan Term

Final Project Report
HDB transaction price

Group 2

Aloysius Teng

Dang Thi Mai Vy

Huang Yaping

Lee Wei Zheng

Neel Modha

Contents

1. Project Objective

Public housing is one of the vital needs of most Singapore residents. According to the Department of Statistics Singapore (2023), total HDB dwelling rate is up to 77.9% in year of 2022. According to OrangeTree (2023), the HDB resale volume has also been stable in the range between 23,714 in 2019, up to 27,896 in 2022 with a spike in transaction volume to 31,017 in 2021.

This project aims to help potential HDB buyers and sellers to accurately predict the price for their resale HDB flats, to ensure they are aware of the market pricing of the flat, to be able to finance their sale or purchase, and to potentially time their buy or sell decisions accordingly.

Past HDB resale transaction data, up to 20 years ago, is well recorded and contains quality information on the types of flats being sold, including (but not limited to) attributes such as town area, flat model, floor area, storey ranges, location and HDB block no. Due to some of these attributes being non-numeric, the data having time factor, and the large volume of data available, the Machine Learning approach is more suited for the objective of this project over conventional analytical tools such as Excel or SQL.

In this project, 5 machine learning models are trained to carry out the objective: linear regression, linear regression with lasso and ridge regularization, gradient boosting, random forest and artificial neural network. The model performance is then compared against and analyzed to investigate which model is the most suitable one to be used for the objective.

2. Introduction of Dataset

2.1. Overview of Dataset

HDB resale transaction data from the year 1990 to the year 2023 are obtained from Data.com.sg (2023) with about 900,000 datapoints. To capture the macroeconomic factors (such as demand and supply for flats) and the availability of financing funds, the original dataset is supplemented with interest rates (SGD SIBOR 3-month) (Bloomberg, 2023) and HDB resale price index (Housing & Development Board, 2023) over the years as a proxy to reflect the economic situation of HDB resale. Description of the combined dataset is included in the appendix.

As stated in the project objective, the target variable is ***"resale_price"***, while the independent variables to be explored include the remaining variables in the original dataset, as well as the augmented variables which will be explained in the subsequent sections.

2.2. Dataset Preprocessing

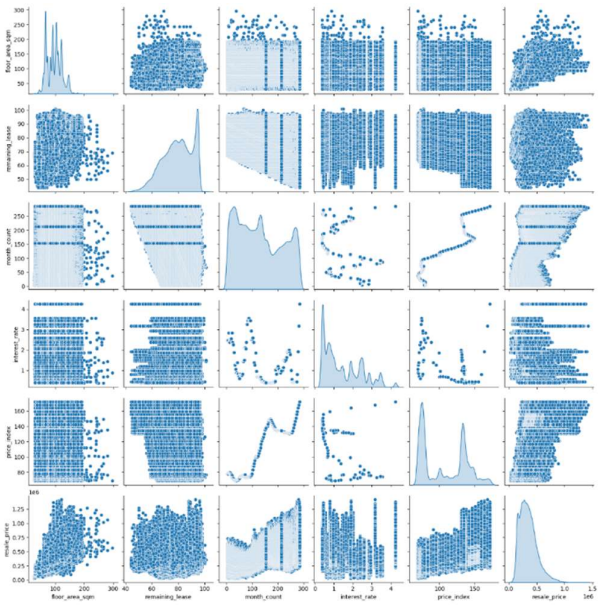
2.2.1. Data cleaning and transformation

To improve model accuracy, the following cleaning steps and transformations are applied:

- Removal of data of the months that do not have ***"interest_rate"*** available i.e. resale transactions before Sep 1999
- Transformation of ***"lease_commence_date"*** into ***"remaining_release"***, which measures the number of lease months left after the resale
- Dummy variables creation from categorical variables with one-hot encoding
- Ensuring the dataset has no missing values
- Normalizing the continuous variables to make them more comparable and improve model training speed

2.2.2. Exploration of continuous variables

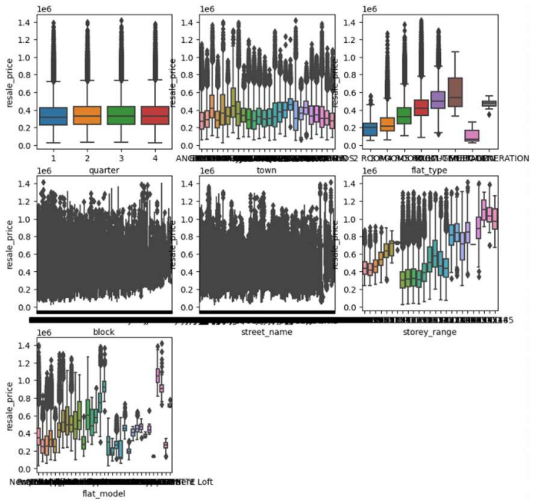
Continuous variables include ***"floor_area_sqm", "remaining_lease", "month_count", "interest_rate", "price_index"*** and ***"resale_price".*** Pair plots were plotted for all pairs of continuous variables to visualize their relationships. From the pair plots, it is observed that ***"price_index"*** and



"month_count" had a linear relationship. Hence, the decision was made to drop ***"price_index"*** as ***"month_count"*** is a more generic variable, and captured information regarding time's influence on the resale price (more on this in section 2.2.5).

2.2.3. Exploration of discrete variables

Histograms were plotted to explore the categorical variables, which included ***"town", "flat_type", "block", "street_name", "storey_range", "flat_model", "quarter".*** The decision was made to drop ***"quarter"*** as there was no significant trend in the ***"resale_price"*** across quarters of the year. ***"block"*** and ***"street_name"*** were dropped because ***"town"*** provided sufficient location information, and instead, used the ***"block"*** and ***"street_name"*** to mine geospatial data to supplement location information (see section 2.2.4).



2.2.4. Dataset augmentation with geographical data

The existing HDB transaction data is augmented with various additional variables to improve the accuracy of the models. Variables are added that account for distances (in km) between a transacted HDB flat and its nearest points of interest (POIs), including: (1) MRT/LRT Station, (2) Hawker Centre, (3) Supermarket, (4) PHPCs/Polyclinics. The hypothesis here is that the closer a HDB flat is to these amenities, the higher its resale price. In addition to this, the distance of a transacted HDB flat to CBD (also in km) are also measured, using Raffles Place MRT as a proxy for the city centres. A mixture of Google Maps and the Singapore OneMaps API are used to acquire the coordinates (Latitude and Longitude) of the transacted HDB flats, and the various POIs. The Haversine Distance formula is then used to calculate the distance between each HDB flat and the POIs, to find out what are the nearest distances to the POIs across the above categories/city centres for each HDB flat.

2.2.5. Modifying the time variable to fit regression models

The original time variable in the dataset was in the form of 'YYYY-MM'. This makes it an unsuitable input into the regression models. Two techniques were experimented with to preserve the information provided by the time variable, such as potential trends and seasonality patterns, and at the same time, transform the string data into a viable regression model input. In the first technique, the ***"quarter"*** was extracted from the data to check if there were any seasonality patterns in HDB transactions across the year. However, no such relation was found (see section 2.2.1) and hence, it was decided to drop the ***"quarter"*** from the final dataset. In the second technique, the time variable was converted into ***"month_count"***, which took a count of the number of months from the designated start time of the dataset (Sept 1999, which is the date from which interest rate data was available) till the latest datapoint (Feb 2023). This technique maintained the ordinality of the time variable and allowed testing of the hypothesis that HDB resale prices have slowly risen with time.

3. Model Building

3.1. Linear Regression

Linear model is a widely used machine learning approach where a best-fitted linear relationship is established between input variables and target variable using least squared error method. In linear regression, the commonly used cost function is Mean Squared Error (MSE). MSE is defined as the average of the squared differences between the predicted and actual values of the output variables. For linear regression learning, overfitting might occur when the model fits training data very well but not for test data. The simple linear regression model gave an output R^2 of 0.68, suggesting that either (I) the number of features in the model needed to be adjusted to potentially reduce overfitting – L1 Regularization (LASSO) was used to do this (ref. Section 3.2) or (II) More complex models were required to address theoproblem statement.

3.2. Linear Regression with L1 (LASSO) Regularization

Regularization of linear regression penalizes a complex linear model by adding model parameters as part of cost function. LASSO uses the sum of the absolute values of the coefficients multiplied by a tuning parameter as penalty term. LASSO is suitable for models with a high number of features. For feature selection and analysis of feature importance in the model, LASSO model is tested with alpha = {0.01, 0.1, 1, 10, 100, 1000}. The model identified ***"month_count"*** and ***"floor_area_sqm"*** as the most important features, with these features' coefficients being penalized the least as alpha increased. However, R^2 decreased slightly (from 0.68 to around 0.59-0.65) with increasing alpha, suggesting that most features were indeed important to varying extents in predicting resale price, and more complex models would be needed to improve forecasting accuracy.

3.3. Random Forest

Random forest is a tree ensemble model that is based on the decision tree machine learning model. Decision tree-based model works by splitting the training data into different sub-trees or leaves using Gini impurity calculation. Like decision trees, random forest models can perform both classification and regression tasks, with the regression prediction being done by taking the mean of the target variables in the same leaf once the splitting of training data has been completed. The model trained uses default setting of 100 trees as the R^2 of this model is already above 0.95 (to be shown in later section). No further hyperparameter tuning is done to avoid overfitting.

3.4. Gradient Boosting

Gradient boosting is a tree ensemble model which can be applied to both regression and classification tasks. It employs decision trees with the trees being built repeatedly on remaining residuals from the previous iteration. As such, gradient boosting can fit very well on the training data. Hence, as the number of trees increases, the train score can become higher but at the same time it is more likely overfitting on test data. To avoid overfitting, Grid Search is used with 100, 300 and 500 trees as the parameter search and 5-fold cross-validation. The results show that as the number of trees increases, better validation scores are obtained. The best average score is 0.87 from the model with 500 trees.

3.5. ANN

Artificial Neural Networks (ANN) is a computing system inspired by the structure of neurons in a human brain. It consists of layers made of nodes that sum the output of nodes from the previous layer and applies an activation function. A fully connected ANN was used, this means that each node receives the output from all nodes from the previous layer and sends its output to all nodes in the next layer. Grid search was used to find the best learning rate (0.1, 0.01, 0.001) and batch size (16, 32, 64, 128), this was done using 5-fold cross-validation with 10 epochs each. The result showed that the best parameters were learning rate of 0.1 and batch size of 16. However, further testing with 100 epochs showed that the second-best set of parameters (learning rate=0.01, batch size=64) performed better as it managed to

converge at a lower loss value, hence this set of parameters was used in the analysis. The model consists of 4 hidden layers with 128, 256, 128, 64 neurons.

4. Result Analysis

4.1. Base Case Model – Estimate Current Prices

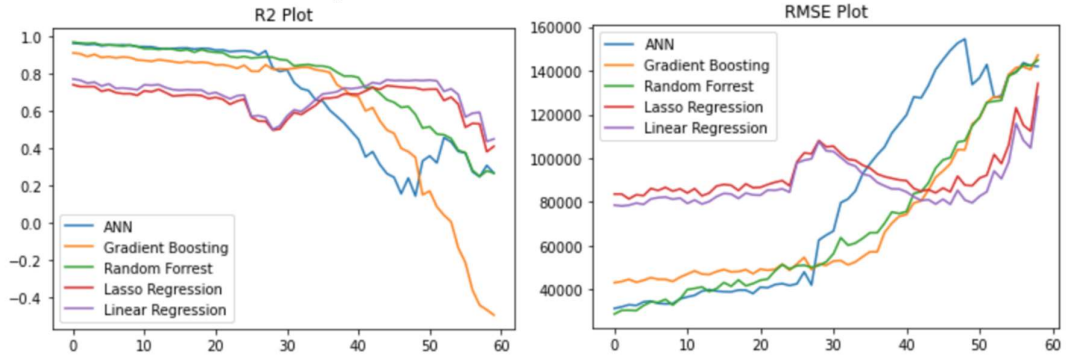
The models were first tested across the entire dataset, having a train-test split (with np.random.seed(2023)) across all time periods, to test if the models performed well in estimating current HDB resale prices. Using R^2 (overall accuracy of model) and Root Mean Squared Error (RMSE, proxy for error vs. actual observed resale price) as performance measures, all of the models performed well with $R^2 > 84\%$. It was further noted that Random Forest and ANN models performed best, with an accuracy of 98%.

Performance Metric	Linear regression	LASSO (alpha = 0.01)	Gradient Boosting	Random Forest	ANN
R2	0.85	0.84	0.96	0.98	0.98
RMSE (SGD)	61,693	63,259	29,839	21,233	21,731

4.2. Prediction Model – Predict Future Prices

4.2.1. R^2 and RMSE graphs

To predict the future prices of HDB resale flats using historical data, the models are trained on historical transactions and tested on future transactions. The dataset is still split on 80:20 ratio, which will lead to training data having ***"month_count"*** < 221 (historical transactions including and before Dec 2017) and test data having ***"month_count"*** >= 221" (future transactions including and after Jan 2018). This results in training dataset having 221 months of transaction data, and testing dataset having 60 months of data. The R^2 and RMSE are used as performance measurement.



.

R^2 has a range from 0 to 1 and shows how much information is captured by the models. RMSE can be interpreted as the average prediction errors in terms of monetary value. Predicting future prices is harder than estimating the current prices because of unforeseen factors that are not captured in the training data. This issue is clearly reflected in the model results, measured by R^2 and RMSE against 60 future months as test data:

R^2 analysis: ANN and Random Forest are the best among the 5 models with R^2 above 0.9 for 22 and 27 future months respectively, however R^2 for ANN deteriorate rapidly around month 25 (July 2020). Gradient Boosting is better than the regression models but only has 3 months with R^2 above 0.9. Simple Linear Regression performs slightly better than Lasso Linear Regression, although both are the worst performers among the 5 models.

RMSE analysis: ANN and Random Forest are the best among the 5 models, with the lowest RMSE below SGD48,000 for 22 and 27 future months respectively, which is equal to 10% of the average resale prices of SGD480,000 from the test data. Similar to the trend with R^2, RMSE for ANN deteriorate

rapidly around month 25. Gradient Boosting is the third best with the RMSE below SGD48,000 for 14 future months and below SGD60,000 for 37 future months, the most among the models. RMSE of Simple Linear Regression and Lasso Linear Regression is above SGD78,000 from the first future month onwards, making it less useful for users.

As shown in the R^2 and RMSE plots, the model performance quickly deteriorates after month 25 (most prominent with ANN), which coincides with rapid price rises of resale HDB because of Covid. It implies that COVID-19 is a potential factor that is not captured in the training data. However, 2 models are still able to produce good results within 10% price error in the first 20 months. From the result, it can be concluded that ANN and Random Forest are the best models to be used for prediction purposes.

4.2.2. Feature importance analysis

Feature importance from each model is analyzed to identify the most important feature affecting future prices. The 5 most important features according to each are recorded in the below table:

Importance Ranking	Linear Regression	LASSO	Gradient Boosting	Random Forest	ANN
1	floor_area_sqm	floor_area_sqm	month_count	month_count	floor_area_sqm
2	month_count	month_count	floor_area_sqm	floor_area_sqm	flat_model_terrace
3	remaining_lease	dist_city	dist_city	dist_city	dist_mrtlrt
4	storey	storey	storey	flat_type_executive	remaining_lease
5	flat_type_executive	int_rate	remaining_lease	remaining_lease	int_rate

The table shows that models are mostly consistent in evaluating the importance of features: ***"floor_area_sqm"***, ***"month_count"***, ***"dist_city"*** are in the top 3 for 3 out of 5 models. However, there are two main limitations observed:

- Variations are due to different feature importance calculations employed by each model. Feature importance was derived via coefficient comparison for Linear Regression and LASSO Regression, mean decrease in impurity of tree splits for Gradient Boosting and Random Forest, and1st layer weight magnitudes for ANN.
- Dummy variables created from each one-hot encoding of the original categorical feature is treated as a single feature.

As such, another method is explored to calculate feature importance to address such limitations. After careful research, ***Feature_importance_permutation*** from ***mlxtend*** Python package was chosen. Using this method, the importance of a feature is measured as R^2 loss when a target feature data is selected through random shuffling while keeping the other feature data intact. Furthermore, dummy variables created from categorical features ***"town", "flat_type"*** and ***"flat_model"*** for model training were recombined for assessment. The new top 5 most important features are recorded below:

Importance Ranking	Linear Regression	LASSO	Gradient Boosting	Random Forest	ANN
1	flat_type	town	town	flat_type	town
2	flat_model	flat_model	flat_model	town	flat_type
3	town	flat_type	flat_type	flat_model	flat_model
4	month_count	floor_area_sqm	floor_area_sqm	month_count	month_count
5	floor_area_sqm	month_count	month_count	floor_area_sqm	floor_area_sqm

4.2.3. Further Analysis on Individual Attributes

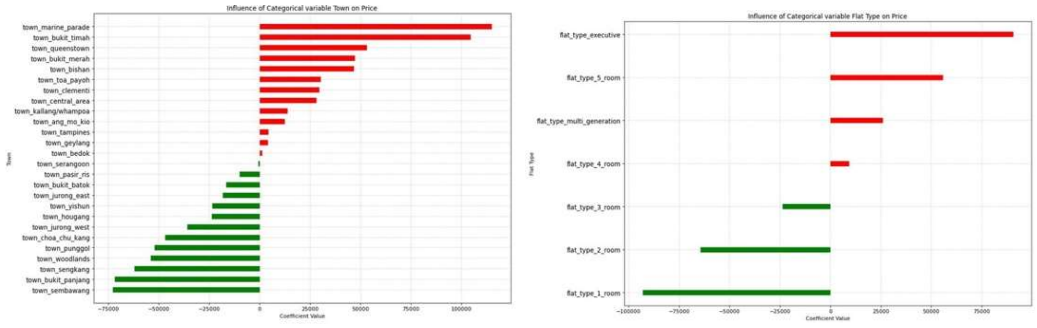
If a closer look is taken at factors that are usually considered during an HDB resale transaction, observations made are mostly aligned with expectations:

Geospatial Variables: In general, the closer a HDB is to a Point of Interest (POI), the more expensive it gets.

Distance to	Nearest School	CBD	Nearest Supermarket	Nearest Hawker Centres	Nearest MRT/LRT	Nearest PHPC/ Polyclinic
Change in HDB Price per +1KM	+$15.5K	-$3.8K	-$5.6K	-$6.5K	-$9.2K	-$12.6K

- Distance to Nearest School remains an outlier. Potential reasons could be that most schools are in the heartlands
- Influence of Distance to PHPCs/Hawker Centres seems to be unusually high compared to Distance to CBD, or Distance to Nearest MRT/LRT. Potential reasons could be inaccuracies in the data due to missing POI opening dates. Currently, opening dates were only scraped for MRT/LRT stations.

Town: Central and mature towns come with a premium. The most expensive towns are Marine Parade, Bukit Timah, Queenstown, which are more mature estates located in prime areas. The cheapest towns are Sembawang, Bukit Panjang and Sengkang, which also fits in with the hypothesis that less mature towns which are far away from the city centre are cheaper.

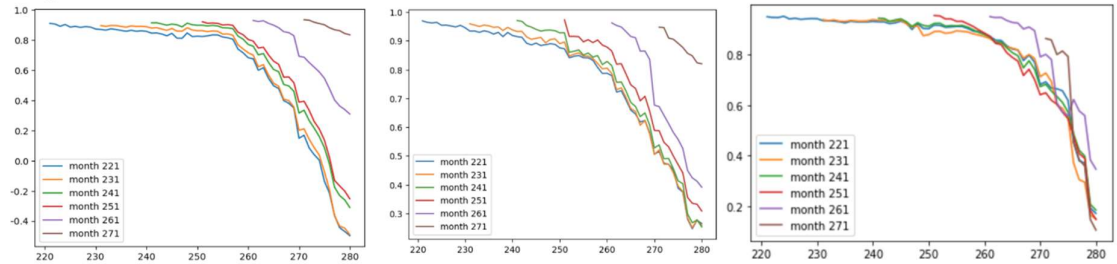


Flat Type: The bigger the size, the better the resale price, evident by 4-room, 5-room and executive flat types having a greater price premium, most likely due to larger floor areas.

5. Extended Analysis

5.1. Impact of Including COVID-19 Data in Training Data

From section 4.2.1 analyzing R^2 and RMSE against 60 future months, it is observed that the test performance drops quickly after month 30 due to COVID, which is not captured in the training data. To further explore this macro issue, the best performing models were retrained and retested using a different train-test data split ratio, split by the ***"month_count"*** of 221 (same as the prediction model in 4.2), 231, 241, 251, 261 and 271. For the last 2 thresholds, the month thresholds are after COVID-19, meaning COVID-19 data is included as part of training data. The R^2 of test data against ***"month_count"*** using different thresholds are shown below:

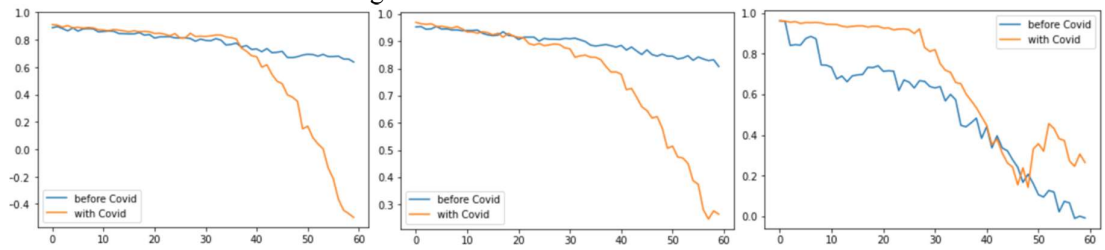


*R^2 plots of (from left to right) Gradient Boosting, Random Forest and ANN by shifting **"month_count"** for train-test split*

The models are consistent in showing that R^2 graphs using the split thresholds after COVID-19 (i.e., the graph for "month 261" and "month 271") have a similar trend, of being initially stable, and dropping off. Despite starting with a comparable R^2 above 0.8, all models tested had a shorter stability and deteriorated earlier as the prediction window is pushed back (from month 221 to 271). The group theorized that the models may be overfitted to spike in training data during COVID-19, and hence unable to predict future prices after COVID-19.

5.2. Model Performance Without the Impact of COVID-19

To further explore the impact of COVID-19 on the models' prediction stability, only data from 1999 to 2019 is considered for training and testing, which is before COVID-19. For this experiment, the reserve 60 months are still maintained as test data, which now includes data from 2015 to 2019. Data from 1999 to 2014 will be used as training data. The model performance is then compared between data before COVID-19 and with data including COVID-19. The R^2 of the models on test data are shown below:



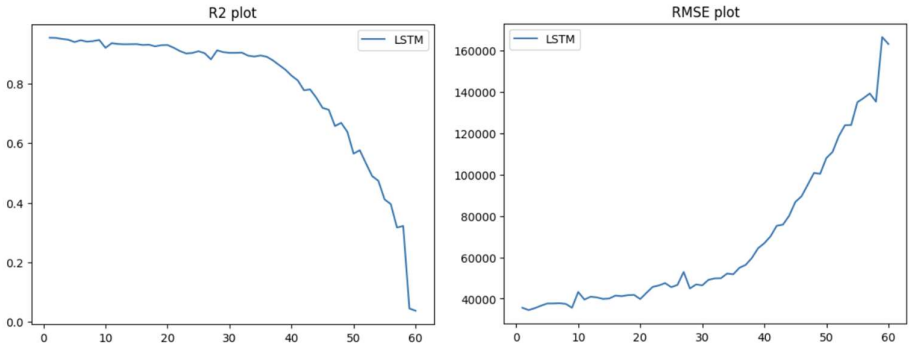
R^2 plots of (from left to right) Gradient Boosting, Random Forest and ANN using testing data before v.s. during COVID-19

The graphs for Gradient Boosting and Random Forrest show that without COVID-19, the predictions are more stable and relatively accurate for a longer time period. For example the Gradient Boosting model, at 60th month forecast, the R^2 value of "before COVID-19" model only drops to around 0.6 compared to dropping to negative value using COVID-19 data. As such, this can prove that COVID-19 and other macro factors plays a significant role in the prediction accuracy. Incorporating or predicting macro factors remains a challenging task and a limiting factor to prediction model.

On the other hand, the graph for ANN shows the opposite result; the predictions deteriorate quicker for "before COVID-19" compared to "with COVID-19". A possible explanation is that a trend change occurred for general HDB resale prices in Jan 2015 (from steep decrease to moderate decrease). Hence the rapid decrease in R^2 value could be that the accuracy of the ANN model more sensitive to trend changes compared to Gradient Boosting or Random Forest. As mentioned previously, the rapid deterioration of R^2 value for ANN at month 25 of the "with COVID-19" plot was due to rapid increase in HDB prices in July 2020 (flat to steep increase) due to COVID-19, this further supports the hypothesis that the accuracy of the ANN model is more sensitive to trend changes.

5.3. LSTM Exploration

An additional analysis that was done through training an LSTM model, which is a model suited for time series data, and investigating the performance of the model in forecasting prices. Using the train-test split ratio as specified in section 4.2, the LSTM model was trained and tested on forecasting test data. The R^2 and RMSE of the forecasts are:



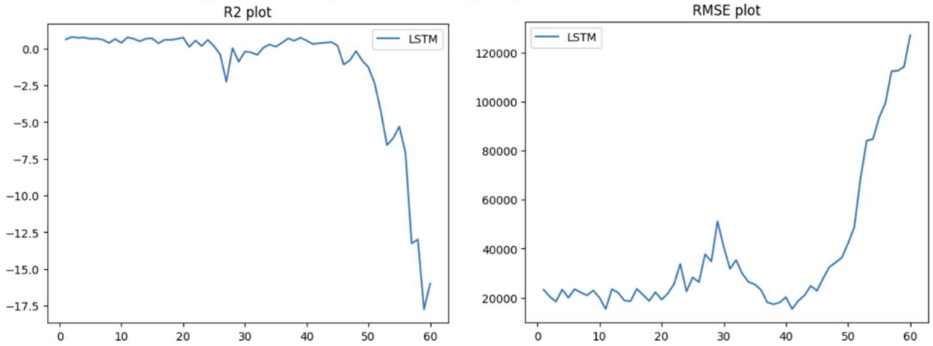
LSTM R^2 plot and RMSE plot vs months of forecast

The results from LSTM forecast are consistent with the results from the 2 best models trained in section 4.2.1, in that the number of months of forecast in which R^2 can be maintained above 0.9 is around 25 to 30 months. This observation lends credit to the conclusion that was drawn from section 4.2.1 in that the trained model has failed to adapt to fluctuations from major events, thus resulting in decreasing R^2. It also shows that the random forest and ANN model have been trained to perform as good as LSTM model using the same data as it also produces similar R^2 and RMSE plots. However, the most important insight to be drawn from training LSTM on the dataset is that there is a possibility that the dataset is not well suited for a time series forecasting as even LSTM failed to account for time-linked major event and R^2 starts to fall off after around 30 months of forecasting.

Using the same LSTM architecture, one more analysis is done to further investigate the result obtained in section 4.2.2, feature importance. From the analysis, it is shown that ***"town"***, ***"flat_type"*** and ***"flat_model"*** are the most important features in output prediction. Since all three are categorical features, the analysis is to be done through on isolation of each feature down to 1 type of input for LSTM training and testing. The train-test split ratio remains the same as per section 4.2.1, and the isolation is done based on that which gives the highest amount of applicable data. The selected input filtering for the features is:

town	flat_type	flat_model	***"mount_count"*** for train-test split
"ang_mo_kio"	"3_room"	"new_generation"	221

The above selection of specific categorical features, and the train-test split of specified ***"month_count"***, still gives 18,574 data samples that meet the criteria and training-to-testing ratio of 86:14. As the ratio is still workable for training and testing, the training is proceeded and results are recorded below:



LSTM R^2 plot and RMSE plot vs months of forecast (specific feature isolation)

From the result, it is quite clear that LSTM trained on specific isolation of categorical features did not perform well at all. This proves that conclusion drawn from section 4.2.2 feature importance in correctly identifying ***"town"***, ***"flat_type"***, and ***"flat_model"*** as the most important features, as the model failed to learn properly once these features lack generalization. Furthermore, by isolating data samples that

only meet specific categories, the sequence information of the data may be further broken and thus making the model unable to correctly predict data point of related categories in future dates.

6. Conclusion and Future Works

6.1. Conclusion and Model Limitation

During the project, a few limitations to the prediction model were recognized:

- Geospatial Data Inaccuracy – Missing opening dates of supermarkets, PHPCs, hawker centers and schools
- Missing macro event information such as COVID-19
- Require retraining models with the latest data to stay accurate as R^2 still falls off as the models tried to predict further into the future

Given the limitations, regression models using Random Forest and ANN are still able to predict future resale prices with less than 5% error if the forecast horizon is within 10 months, and within 10% error if the forecast horizon is from 10 to 30 months. Beyond that, the model error is too large to be useful for buyers and sellers.

6.2. Future Works

For future improvements, feature selection can be further explored to simplify the model so that the model can be more versatile. More data related to macro factors such as CPI, Salary Index, Economic downturns, can be augmented to make the model more comprehensive. Furthermore, LSTM can be explored on how it can incorporate event-based data to address the limitation of the 5 models trained in the main project since it is a sequence-based model.

7. References

Bloomberg L.P. (2023). SG SIBOR Interest Rate (3-month) Historical Data. Retrieved March 14, 2023, from Bloomberg Terminal

Data.com.sg. (2023). *Resale Flat Prices*. Retrieved February 10, 2023, from https://data.gov.sg/dataset/resale-flat-prices

Department of Statistics Singapore. (2023, February 9). *Household: Statistics on resident households compiled by the Singapore Department of Statistics*. Retrieved April 5, 2023, from https://www.singstat.gov.sg/find-data/search-by-theme/households/households/latest-data

Housing & Development Board. (2023, April 3). *Resale Price Index from 1st Quarter 1990 to 1st Quarter 2023 (Flash Estimate)*. Retrieved February 26, 2023, from https://www.hdb.gov.sg/residential/selling-a-flat/overview/resale-statistics

OneMap SG (2023, March 31). Singapore OneMaps Visualization and APIs. Retrieved March 14, 2023 from https://www.onemap.gov.sg/

OrangeTree. (2022). *HDB Market Pulse: Real Estate Data Trend & Analytics Q4 2022*. https://www.orangetee.com/Home/ResearchPath/OrangeTee%20-%20HDB%20Market%20Pulse%20Q4%202022.pdf

8. Appendix

8.1. Dataset Description

Column	Format	Description	Min	Max	Mean	Std	Categories
month	'YYYY-MM'	Month at the time of resale	1990-01	2023-02	-	-	-
town	string	Town location of the resale flat	-	-	-	-	27 different towns across Singapore
flat_type	string	Type of the resale flat	-	-	-	-	6 categories: '1 ROOM': 1294 '2 ROOM': 10925 '3 ROOM': 287911 '4 ROOM': 337785 '5 ROOM': 187878 'EXECUTIVE': 67691 'MULTI-GENERATION': 537
block	string	Block of the resale flat	-	-	-	-	2.660 different blocks
street_name	string	Street name of the resale flat	-	-	-	-	582 different street names
storey_range	string	Storey range of the resale flat	-	-	-	-	25 groups, ranging from level 1 to level 51
floor_area_sqm	float	Floor area of the resale flat in squared meter	28.0	307.0	95.7	25.9	-
flat_model	string	HDB model of the resale flat	-	-	-	-	34 different flat models
lease_commence_date	'YYYY'	Year when the resale flat started the 99-year lease	-	-	-	-	-
interest_rate	float	Interest rate recorded for each quarter over the years, starting from Sep 1999	0.373	4.25	1.42	0.96	-
price_index	float	Price index of HDB for each quarter over the years	69.1	171.9	106.7	31.36	-
resale_price	float	Price of resale flat at the time of transaction, SGD	50,000	1,418,000	311,076	162,289	-