Prediction And Analysis Of HDB Rental Rates

Ng Wei Jie Brandon
School of Computing
NUS
Singapore
e0315868@u.nus.edu

Low Jia Liang
School of Computing
NUS
Singapore
e0764330@u.nus.edu

Zhou Zhou
School of Computing
NUS
Singapore
e0950207@u.nus.edu

Cai Haozhe
School of Computing
NUS
Singapore
e0950244@u.nus.edu

Zhao Wenzhuo
School of Computing
NUS
Singapore
e0945769@u.nus.edu

***Abstract*—We aim to predict the HDB rental rates in Singapore using publicly available data from the government website. We performed Exploratory Data Analysis to visualize the data to guide us in the subsequent steps for feature engineering. We evaluated the performance of different regression models using k-fold cross-validation with RMSE as the metric. We further optimized the model using grid search to finetune the hyperparameters. We also used auxiliary data of public amenities to improve the model's predictive ability. Our best model is XGBRegressor with an RMSE score of 481.01. We found the important features that contribute to the rental prices are the flat type, rent approval date, and region.**

I. Introduction

A. Motivation

Rents in public and private housing markets in Singapore soared in 2023. HDB and condo rents are higher by 20.8% and 17.3% respectively in August compared to a year ago.[4] Rising rent is one of the biggest financial burdens and is overwhelming tenants. This motivated us to alleviate the rental issue by developing a predictive model and data-driven analysis, helping stakeholders make informed decisions related to rentals in Singapore.

B. Objectives

Our main objective for this project is to create a regression model to predict the rental prices for different types of housing in Singapore. The model helps tenants to search for similar housing in a certain price range. Our secondary objective is to understand factors and trends contributing to the increase in rental prices. This information helps prospective tenants to prioritize factors when choosing houses to rent.

II. Data Analysis & Preprocessing

A. Data Exploration

We used the HDB rental rates obtained from publicly available data from the government website at data.gov.sg as the dataset for analysis. The dataset consists of 60,000 records spanning 16 columns of attributes - 10 categorical attributes and 6 numerical attributes. There are no null values in this dataset. Below is the list of attributes and the description and data type. Table I shows the number of variables for each of the categorical attributes.

- **rent_approval_date** (string): year and month when the rent was approved
- **town** (string): the town of the sold HDB flat
- **block** (string): block number of the flat
- **street_name** (string): street name of the block containing the flat
- **flat_type** (string): type of flat
- **flat_model** (string): model of the flat
- **floor_area_sqm** (string): floor area in square meter
- **furnished** (string): indicator if flat is furnished
- **lease_commence_date** (integer): year the lease for flat commenced
- **latitude** (numeral): latitude of block containing the flat
- **longitude** (numeral): longitude of block containing the flat
- **elevation** (numeral): elevation of block containing the flat in meter
- **subzone** (string): subzone of block containing the flat in meter
- **planning_area** (string): planning area of block containing the flat
- **region** (string): region of block containing the flat in meter
- **monthly_rent** (numeral): rental rate SGD

TABLE I
Number of variables for each attribute

Attribute	Number Of Variables
Town	26
Block	2553
Street Name	1083
Flat Type	5
Flat Model	19
Furnished	1
Elevation	1
Subzone	152
Planning Area	29
Region	5

The dataset also contains additional information about the locations of primary schools, shopping malls, existing MRT stations, and planned MRT stations represented as longitude and latitude. The dataset also has information on stock prices and COE prices dated from 2021.

B. Data Cleaning

The dataset is clean except for the attribute 'flat_type'. For this attribute, 'n-room' and 'n room' are recorded as two separate categorical values when both values refer to the same flat type. We removed the dash to merge 'n-room' and 'n room' together.

We removed the attributes 'furnished' and 'elevation' from the dataset since there is only 1 categorical value, offering no valuable information for the model. On the other hand, we exclude the attributes 'block,' 'street_name,' and 'subzone' because there is a large number of variables that will lead to very sparse columns after one-hot encoding. We also dropped the attributes 'town' and 'planning_area' as they do not provide much variability in the rental prices, and the information is already encoded in the latitude and longitude. Besides, some flats are less common in certain towns and planning areas in Singapore and are not covered in both the training and testing datasets.

C. Visualisation

We plotted different types of graphs to visualize and observe the effect of the attributes on the rental prices. To begin, we plotted the distribution of the monthly rent is plotted in Figure 1. Most of the rental prices are between $2000 to $2500 and the graph is skewed left. This gives us a sense of the rental price range.

The effects of rental prices can be observed for different attributes in Figure 2. Below are some observations made.

- Average monthly rent by regions: Rent is fairly uniform across regions, with higher rates in the central and lower rates in the north. The central region is where businesses and people gather while the north is where nature is; hence, rentals in the central region will be more expensive as demand is higher.
- Monthly rent by floor area: Rent is positively correlated with floor area with a correlation score of 0.3064. Rental rates are usually higher when the living space is larger.
- Average monthly rent by flat type: Rent is higher for housing with more rooms. Similar to floor area, rental rates are usually higher when the living space is larger.
- Average monthly rent by flat model: Rent has significant variation across flat models. Different flat models target different groups of people. For example, the 2-room flats are targeted at low-income and elderly groups while 3-room and 4-room are targeted at young couples. These groups belong to different income brackets, driving different rental prices.
- Monthly rent over rent approval date: Rent is significantly higher for recent rent approval dates over the past 2 years due to high inflation and delays in the supply of new flats due to COVID-19. Predictions of recent rental rates are expected to be higher. compared to previous years.
- Monthly rent over lease commence date: Rent is substantially higher compared to 50 years ago and fluctuates over time. The steady upward trend in rental prices should be due to inflation while the fluctuations depend on the supply and demand of houses.

Fig. 1. Distribution of monthly rent

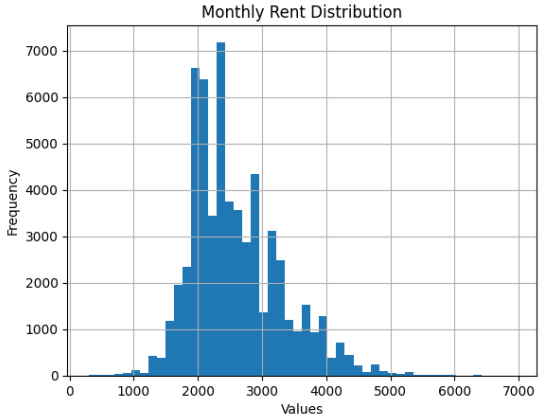
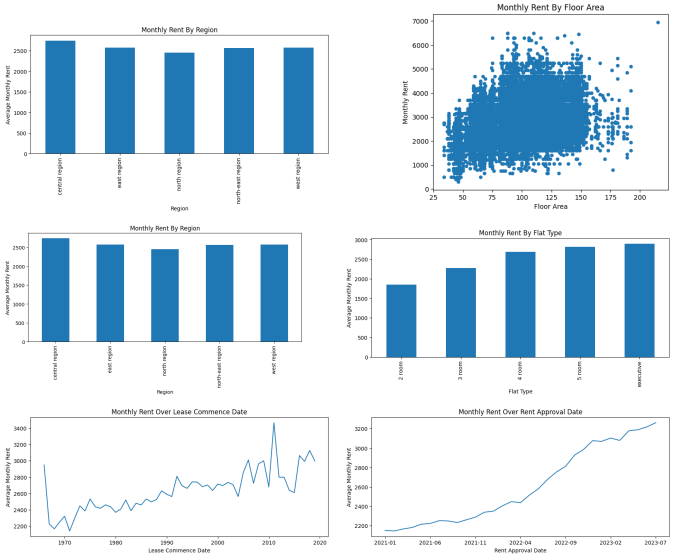


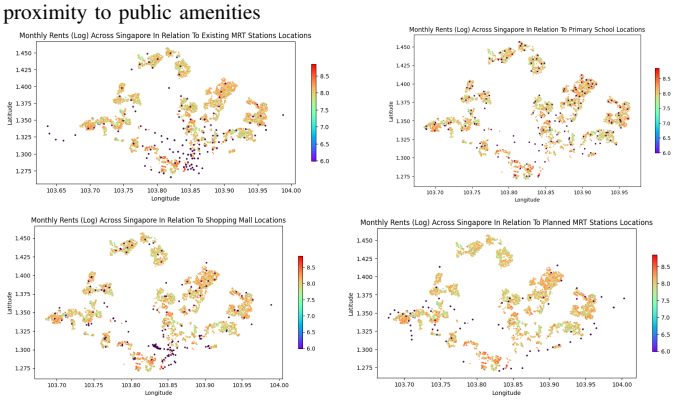
Fig. 2. Effect on rental prices for different attributes



The scatter plots of the HDB flats in Singapore are plotted using their longitude and latitude in Figure 3. The intensity of the dots is the logarithmic of the rental prices with red showing the higher end of rental prices while blue showing

the lower end of the rental prices. A logarithmic scale is applied to the rental prices as rental prices can be very extreme for some flats as shown in Figure 1. Besides, Figure 2 shows that the rental prices for recently approved HDB flats are more expensive. We only plotted rental prices since 2023 January for a fairer comparison of recent rental prices, reducing the number of records from 60,000 to 11,971. The dark blue dots represent public amenities such as primary schools, MRT stations, and shopping malls.

Fig. 3. Scatter plot of HDB flats in Singapore show the monthly prices and proximity to public amenities



- Monthly rents (log) across Singapore in relation to existing MRT station locations: Existing MRT stations are mostly concentrated in the central region. The rental prices of HDB flats in the central region are also more expensive. There are more red dots indicating rental prices in the central region are on the higher end. HDB flats nearer to MRT stations are expected to be higher due to the convenience of commuting.

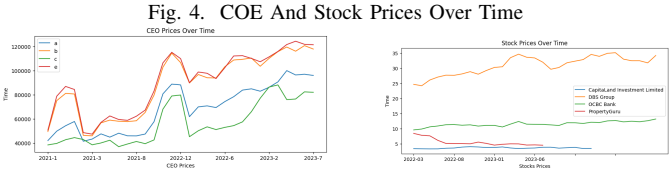
- Monthly rents (log) across Singapore in relation to shopping mall locations: Similar to existing MRT stations, the majority of the shopping malls are concentrated in the central region where rental prices of HDB flats are also higher. The central region is where the business districts are, and we expect more shopping malls located in the central region. The proximity to shopping malls should have a profound impact on rental prices due to the convenience of shopping.

- Monthly rents (log) across Singapore in relation to primary school locations: Primary schools are surprisingly well distributed across Singapore. Unlike MRT stations and shopping malls, primary schools are not concentrated in the central region. The distribution of primary schools is likely due to Singapore's urban planning, putting primary schools in areas where HDB flats are common. We can see the clustering of higher rental prices around primary school locations. Besides, Singapore's primary school gives first priority to students who reside within 2km.[3]

- Monthly rents (log) across Singapore in relation to planned MRT station locations: Some of the planned MRT stations are located far away from the residential areas or business districts since these are MRT stations that are planned for the future. The scatter plot does not show any noticeable relationship between the location and rental prices of HDB flats, and the locations of the planned MRT stations.

The dataset also contains auxiliary data about the COE prices since 2021 January and stock prices since 2022 March. We explored a few stock prices, focusing on stocks related to banks (mortgages) and property (investors). We plotted line graphs to observe the change in COE and stock prices over time. The COE prices show an increasing trend; however, the price movements in COE prices differ from those in HDB rental rates. Even though both prices are driven by supply and demand, COE prices are driven by different factors, such as government policies and available units after construction, etc. We explore 4 stock prices (DBS, OCBC, Property Guru, and CapitalLand Investment Limited). The price movements of these stock prices do not show any noticeable relationship with the HDB rental rates. The stock prices are likely diversified and affected by other variables not just the property market in Singapore. Both the COE and stock prices offer little information about the rental prices.

Fig. 4. COE And Stock Prices Over Time



D. Data Preprocessing

Since most ML models can only handle data in numeric form, we need to preprocess categorical attributes and any other attributes with the data type string for the models to work on. We also enriched the current dataset with auxiliary datasets of the locations of shopping malls, existing MRT stations, and primary schools. Note that when we compute the distance between two points, we use the latitude and longitude in the EPSG:4326 geo-coordinate system and project and compute the distance in meters with the modules provided by GeoPandas and Shapely libraries. We counted the number of public amenities within a certain radius by utilizing the module BallTree from the sklearn.neighbors library. Below is the list of attributes and the corresponding steps applied to preprocess the data.

- **rent_approval_date**: Split the date string into year and month for the model to capture seasonal patterns or yearly trends in rental rates
- **town**: Drop column due to lack of variability as discussed in the data cleaning section
- **block**: Drop column due to sparse data as discussed in the data cleaning section
- **street_name**: Drop column due to sparse data as discussed in the data cleaning section
- **flat_type**: Apply one-hot encoding into numerical columns for the model to train on and predict rental prices based on the type of flat
- **flat_model**: Apply one-hot encoding into numerical columns for the model to train on and predict rental prices using flat model
- **floor_area_sqm**: No change since values are already in numerical form for the model to train on and predict for different floor areas
- **furnished**: Drop column due to single value as discussed in the data cleaning section
- **lease_commence_date**: No change since values are already in numerical form for the model to train on and predict for different lease commence date for HDB flats
- **latitude**: No change since values are already in numerical form for the model to predict rental prices using geographical location
- **longitude**: No change since values are already in numerical form for the model to predict rental prices using geographical location
- **elevation**: Drop column due to a single value as discussed in the data cleaning section
- **subzone**: Drop column due to sparse data as discussed in the data cleaning section
- **planning_area**: Drop column due to lack of variability as discussed in the data cleaning section
- **region**: Apply one-hot encoding into numerical columns for the model work on and predict prices by region
- **monthly_rent**: No change since values are already in numerical form and are the target value
- **n_schools**: Compute the number of primary schools within 2km radius from the HDB flat since the students living within 2km are given higher priority to the school
- **n_malls**: Compute the number of shopping malls within 2km radius from the HDB flat; the radius proximity currently defaults to 2km since the improvement in the model is not significant when we evaluate the feature importance for this attribute
- **n_mrt_existing**: Compute the number of existing MRT stations within 2km radius from the HDB flat; the radius proximity currently defaults to 2km since the improvement in the model is not significant when we evaluate the feature importance for this attribute

The following table in Table II shows the data for a sample in the preprocessed dataset before one-hot encoding. After the above preprocessing steps and one-hot encoding, we standardized all numerical fields to help the ML learn the training data and converge faster. We used the module StandardScalar from sklearn.preprocessing.StandardScalar to standardize the values. We did not extract features for planned MRT stations, COE prices, and stock prices as the visualization plots do not show any noticeable relationship with rental prices.

TABLE II
NUMBER OF VARIABLES FOR EACH ATTRIBUTE

Attribute	Sample Data
rent_approval_date-year	2021
rent_approval_date-month	7
flat_type	3 room
flat_model	new generation
floor_area_sqm	67.0
lease_commence_date	1979
latitude	1.366600
longitude	103.855579
region	north-east region
monthly_rent	29
n_schools	9
n_malls	5
n_mrt_existing	4

III. DATA MINING METHODS

We experimented with a variety of data mining techniques to find the best regression model. We evaluated our regression models using k-fold cross-validation with k=10 while optimizing the major hyperparameters using grid search. We followed the metric RMSE to evaluate the model prediction. The following subsections briefly discuss the list of regression models that we have experimented with.

A. XGBRegressor

XGBRegressor is an additive model based on Gradient Boosted Decision Tree (GBDT) and uses ensemble learning that combines multiple weak learners in the form of decision trees to create a strong learner. The difference is that XGBRegressor is packed with many other optimizations, such as penalizing large coefficients, dropout to prevent overfitting, parameters to reduce the model complexity, etc. The training method optimizes each base learner inside using a forward step-by-step algorithm.[2]

In this project, XGBRegressor is our best-performing model. To push the model performance further on the current dataset, we look into fine-tuning the model hyperparameters while reducing overfitting since the training dataset is small. XGBoost allows us to adjust L1 and L2 normalization and we set alpha and lambda values to 1. We also decreased the depth of trees and the value of the subsample to make the algorithm more conservative. As for min_child_weight, it controls the number of samples in the leaf node, we increased the number to balance the samples of leaf nodes to give a more robust prediction. The parameters for XGBRegressor

after finetuning are learning_rate = 0.1, n_estimators = 550, max_depth = 4, min_child_weight = 5, seed = 27, subsample = 0.7, colsample_bytree = 0.7, gamma = 0.1, reg_alpha = 1, and reg_lambda = 1.

B. KNeighborsRegressor

KNeighborsRegressor is a regression model that uses the k-nearest neighbors in the dataset to predict a value. The regression model computes the distance between the features of the input to the other features in the dataset and selects the top-k most similar.[1] The predicted value is the interpolation of the target value of the top-k similar points. The hyperparameter tuned is 'n_neighbors' [3, 4, 5, 6].

C. AdaBoostRegressor

Adaboost regression uses an ensemble of weak learners to create a robust regression model. The model assigns higher weights to misclassified samples, allowing the algorithm to focus on challenging data that was predicted wrongly. The model combined the weighted sum of the outputs from weak learners as the final output for prediction. The ensemble learning technique enhances the model's overall performance and robustness.[1] The hyperparameters tune is 'n_estimators' [50, 100, 150, 200].

D. BaggingRegressor

BaggingRegression uses ensemble learning and fits the weak learners on random subsets of the original dataset depending on the bootstrap sampling method. Finally, a voting mechanism or averaging combines the prediction results of all weak learners to obtain the final ensemble prediction result. Also, unlike AdaBoostRegressor where weak learners have different says or weights depending on their performance, the weak learners in BaggingRegressor always have equal say.[1] The hyperparameters tune is 'n_estimators' [10, 20, 30, 40].

E. RandomForestRegressor

Random forest regression also uses ensemble learning to construct a linear regression model. Random forest regression creates bootstrap samples and random feature selections to create variation in the bootstrap dataset, helping to train weak learns that are less correlated and achieving a regression model with less variation in performance.[1] The hyperparameters tune is 'n_estimators' [50, 100, 150, 200].

F. ExtraTreesRegressor

Extra trees regression is similar to random forest regression in that both use an ensemble of learning techniques and aggregate the predictions of multiple decision trees. The difference is that extra tree regression uses the original sample instead of subsampling with replacement. Another difference is that extra tree regression chooses a selection of splits randomly instead of an optimal split. The difference reduces the computation cost and is faster than random forest regression.[1] The hyperparameters tune is 'n_estimators' [50, 100, 150, 200].

G. GradientBoostingRegressor

Gradient boosting regression also uses ensemble learning but uses the loss function to calculate the gradient and update the model. Gradient boosting builds an additive model with other weak learners to minimize the loss function and make predictions.[1] The hyperparameters tune is 'n_estimators' [50, 100, 150, 200].

H. HistGradientBoostingRegressor

Histogram gradient boosting regression is very similar to gradient boosting regression, except that it performs much faster on larger datasets. For continuous values, every point in between is considered a potential split in gradient boosting. However, histogram gradient boosting binned these values, reducing the number of splitting candidates.[1] The hyperparameters tune is 'n_estimators' [50, 100, 150, 200].

I. LinearRegression

Linear regression is a model that learns the linear relationship between features and target value. It learns the relationship with the objective function of minimizing the L2 error. Although this model assumes a linear relationship, it is simple, has high interpretability, and is widely applicable.[1] The hyperparameter tuned is 'fit_intercept' [True, False].

J. Ridge

Ridge regression is similar to the linear regression model except for an additional L2 regularization on the weight coefficient. The L2 regularization helps to penalize large coefficients to avoid overfitting on one feature.[1] The hyperparameter tuned is 'alpha': [50, 100, 150, 200].

K. Neural Network

Neural network can also be designed as a linear regression model where the input in the features and the output is a single prediction head for the rental price. MSE loss function supervises the model and backpropagates the gradients to help the model learn the training data.

IV. Evaluation & Interpretation

We evaluated the different regression models discussed in the data mining section using k-fold cross-validation and finetuned the hyperparameters. Table III shows the RMSE scores for each model. The experiments show that the XGBRegressor with RMSE score of 487.9223, GradientBoostingRegressor with score of 499.1174, and HistGradientBoostingRegressor

with score of 485.3123 are performing better than the other regression models. These 3 models are adaptations of the Gradient Boosted Decision Tree (GBDT) but are optimized differently. We also noticed that the Neural Network model also performed similarly well with score of 494.7823 to the gradient boosting methods. The worst-performing model is the linear regression with score of 2642.0951 as modelling the relationship between rental price and features of the HDB flats as a linear regression is not complex enough.

TABLE III
RMSE SCORES FOR REGRESSION MODELS

Model	RMSE
KNeighborsRegressor	556.0766
AdaBoostRegressor	572.3338
BaggingRegressor	543.7338
ExtraTreesRegressor	576.9002
GradientBoostingRegressor	499.1174
RandomForestRegressor	531.0873
HistGradientBoostingRegressor	485.3123
XGBRegressor	487.9223
LinearRegression	2642.0951
Ridge	509.1826
Neural Network	494.7823

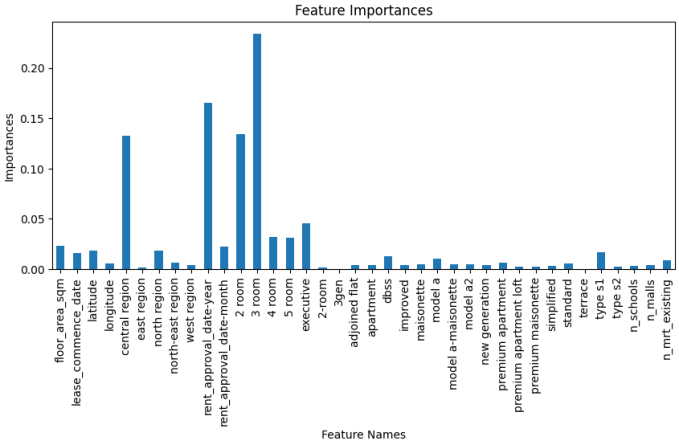
Since XGBRegressor has a lot of additional optimizations to perform better than the other two gradient boosting algorithms. We decided to finetune XGBRegressor using the parameters discussed in the data mining section. We trained on the whole training dataset without the k-fold cross validation, and the RMSE score is obtained from the submission to the Kaggle competition on HDB rental prices. Before finetuning, we obtained an RMSE score of 484.4591 and after finetuning we successfully reduced the RMSE score to 481.0113.

TABLE IV
RMSE SCORES FOR FINETUNED XGBREGRESSOR

Model	RMSE
XGBRegressor (before finetuned)	484.4591
XGBRegressor (after finetuned)	481.0113

After training the XGBRegressor regression model, the model also provides a score for the feature importance as shown in Figure 5. The feature importances provide useful information on what the model uses to predict rental prices. This also provides us insights into important factors that contribute to HDB rental prices. From Figure 5, we observe that the flat type, especially the 2-room, and 3-room, are strong features used by the model to predict rental prices. These are probably the more popular HDB flats in demand and the prices are mostly set at the market rate. The second most important feature is the rent approval year. This is expected since the visualization has shown that rental prices have increased most dramatically in recent years when plotted against the rent approval date. Inflation is a serious contribution to rental prices. Another contributing factor is the HDB located in the central region due to its proximity to many shopping malls and existing MRT stations.

Fig. 5. Feature Importances



V. CONCLUSION

We successfully created a regression model to predict HDB rental prices with an RMSE score of 481.01. We also found the important factors in determining the rental prices are the flat type, rent approval date, and region where the HDB flats are. One limitation of the model is that if new public amenities like shopping malls and MRT stations are constructed, we will need to retrain and update the model since the model only has knowledge of information up till the date it was collected. Besides, there are a lot of other factors not included in the dataset such as salaries and wages, suicide cases affecting rental prices, and government policies.

VI. APPENDIX

The code relevant to this project can be found and reproduced at https://github.com/nwjbrandon/CS5228_Project

VII. REFERENCES

[1] Scikit-learn: Machine Learning in Python, scikit-learn, Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V. and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P. and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E., Journal of Machine Learning Research, 2011

[2] XGBoost: A Scalable Tree Boosting System, ACM, Tianqi Chen and Carlos Guestrin, 2016

[3] Understand how balloting works, Ministry of Education, https://www.moe.gov.sg/primary/p1-registration/understand-balloting, Accessed on November 9, 2023

[4] HDB rents rise 0.1% in August, lowest in almost 2 years; condo rents down 1%, The Straits Time, Esther Loi, https://www.straitstimes.com/singapore/hdb-rents-rise-01-in-august-lowest-in-2-years-condo-rents-down-1, Accessed on November 9, 2023