Prediction of Airbnb listing price per night in the Central Region of Singapore

1. Introduction:

Nowadays, the dominance of the Internet and booking platforms is growing year by year since more and more tourists book their hotels online. Airbnb is a booking platform for listing and renting local homes. Since both hosts and travelers can easily exchange services on Airbnb, a good pricing strategy could create a competitive advantage for hosts.

This report aims to help the listing owner set a reasonable price by predicting the price per night based on host information and the room/flat information and using different machine learning models.

The used dataset is described in the Dataset section. Then, the machine learning problem is explained in more detail in the Problem Formulation section of the report. After that, the methods section describes the Machine Learning regression models. The performances of different methods are compared in the Results section. Finally, the Conclusions section discusses the limitation of the ML model and some directions for future development.

2. Problem Formulation:

The project aims to answer the question: how much could an Airbnb host charge per night in the Central Region of Singapore?

The raw dataset of the project is the detailed listing data in Singapore that are publicly available data from the Airbnb website (http://insideairbnb.com/get-the-data.html). The dataset has been updated since 26th December 2021. There are 3672 data points containing detailed listings information and metrics in the raw data set. The number of columns is 74. Since this project focuses on the listings data in Central Region in Singapore, only listings in this area are taken into account.

One data point has several listing statistics, including the host information and the room/flat information. In particular, the host information includes the host response time, the host response rate, the host acceptance rate, the host-is-super-host, the host-has-a-profile-picture, host-identity-verified. The listing information contains the latitude, longitude, the room type, the accommodation, the number of bathrooms, the number of bedrooms, the number of beds, the minimum and maximum available days, the listing-has-availability, the instant bookable, and the price per night.

The study aims to predict the price per night based on listing information and metrics. Therefore, the value this project is interested in is the price per night of each listing in Singapore dollars. This column is the label, while other columns are considered the features. The features are selected intuitively based on domain knowledge. There are 17 feature columns. Eight columns: host response time, the host is super host, host-has-profile-picture, host-identity-verified, the room type, the accommodates, the instant bookable, and the listing-has-availability are categorical features, while the other nine features have numerical values. Most of the categorical features contain true/false values, while the room type column contains four types of rooms (Private room, Entire home, shared room, hotel room), and the host response time column contains: within a few hours, within a day, within an hour, a few days or more).

3. Preprocessing Data and methods:

There are no duplicated data points. The host acceptance rate and the host response rate columns have incorrect data type objects, so values in these columns have been changed to float. Only numbers have been taken out of the strings of the column the number of bathrooms.

The values in the price column also have been changed to float. Some missing values in the dataset need to be handled before analyzing the data and training Machine Learning models. Those missing values were dropped since there is no logical explanation for them. The cleaned dataset has 2014 data points. The

categorical features require label encoding so that the models can use them. I perform one-hot-encoding using Panda's "get_dummies()" function.
Then, data points were split randomly into training and test sets such that 80% of the data points were in the training set, and 20% of the data points were in the test set in order to assess whether the models could generalize well to unseen data.

a) Linear Regression Model:

The first method used in the project was the Simple Linear regression model since the visualization and correlation table show a linear relationship between some features and the label.

```
price                  1.000000
bedrooms               0.417814
accommodates           0.236382
maximum_nights         0.144654
beds                   0.087751
host_response_rate     0.042566
bathrooms             -0.007504
minimum_nights        -0.136027
host_acceptance_rate  -0.153365
latitude              -0.156096
longitude             -0.184772
Name: price, dtype: float64
```

Figure: Correlation table and visualization

Regarding the scatter plots, using a hypothesis space constituted by linear maps could be helpful in this case. The sklearn Python library and its LinearRegression class were used to perform the linear regression. Then, the k-fold cross-validation with k = 3 was used for the training set due to the limited number of data points. This would help avoid overfitting. (The reason for k = 3 is due to the time limitation).
The mean absolute error loss is chosen as it is robust against a few outliers in the training set. The Linear model is fitted to the training set by minimizing the mean absolute error loss. The mean absolute loss is calculated by using Scikit Learn library. The same loss was also applied for training and test sets with different fits.

b) Polynomial Regression model:

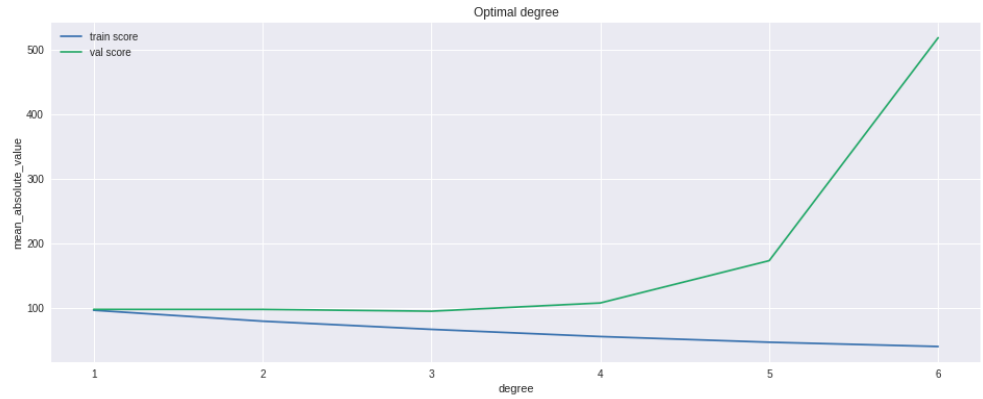
Although the scatter plots show the linear relationship between some features and the label, the correlations seem not high since the highest correlation was around 0.4. Therefore, the polynomial regression models with ridge regression as regularization were also applied to predict the price.
In order to find the best alpha of the ridge regression, I used the grid search cross-validation because this is the most common technique for tuning hyper parameters of a model. After that, the k-fold cross-validation with k = 3 was used on the training set to avoid overfitting and to train polynomial regression models with different polynomial degrees (degree from 1 to 6). The models are trained on the training dataset. After that, the optimal poly degree is chosen with the help of validation errors. The mean absolute error was also used in this case.
The model is used to predict the test set, and error is recorded. The cross-validated error is the average error on the K test sets. This process is repeated for each model. The model with the best cross-validation error is selected by comparing training errors and validation errors of different models.

Finally, the final error is computed on the test set using the learned polynomial regression model with an optimal degree since the comparison of loss values on the validation set indicates how well the model fits the data, and overfitting can be suspected when the difference between these loss values starts to increase.

4. Results:

For the Linear Regression model, the training mean absolute error was 97.023. The validation mean absolute error of the linear model when tested against the validation set was 98.4137. The training and validation errors being the same means there was no overfitting. The validation error is slightly larger than the training error. That means that the model faces a little overfitting. That can be solved by increasing the adequate number of training data points.

For the polynomial Regression Model, the optimal alpha for the regularization is 0.0616. The mean absolute loss calculated for training and validation data sets with six different models is also shown below.



	average_train_error	average_val_error	degree
0	96.228914	97.593042	1
1	79.284963	97.536691	2
2	66.619236	94.690485	3
3	55.506505	107.483915	4
4	46.705719	172.958154	5
5	40.088393	517.951431	6

We can observe that the polynomial regression model with a degree of 3 has the lowest validation loss of 94.69. For the higher degrees, the higher degree the model has, the larger the validation loss is. For the training set, the error constantly decreases. This means that the more complex the model is, the more likely the model is overfitting the training set. Based on the K-fold CV average validation errors of both the Linear Regression model and Polynomial Regression Model, the 3-degree polynomial regression model is chosen as the final model.

In order to assess the model's performance with unseen data, I applied the best model to the test set. The mean absolute error for the final model and the test data is roughly 92.14. Although this number is lower than that of the validation set, the minimum price, the median price, and the maximum price are around 15, 209, and 4159 Singapore dollars, respectively. We could not conclude that the model performs relatively well with the test data since the loss is still considered to be large compared to the true price.

5. Conclusion and Limitation

Based on the test error, there could be room for improvement since the models used in this project did not perform well. The project focused on the regression model, so the full potential of other models for predictions in the data set at hand should be explored in future research endeavors, such as Decision Tree Regressor or Deep Learning methods. Instead of choosing features based on domain knowledge, other methods of feature selections, such as PCA or Lasso, could be tried in this problem since there were a total of 17 features, but they only have little correlation to the label values. The model's performance was also affected by the limitation of the dataset. The data need to be gathered more to improve the models' performance.

6. References:

[1] Scikit Learn, https://scikit-learn.org/stable/index.html
[2] A. Jung, "Machine Learning: The Basics," Springer, Singapore, 2022

7. Appendices: