Airbnb Pricing Analysis in Berlin

Junqian Zhang

University of Milan
Data Science and Economics

Abstract. This work aims at analyzing the Airbnb Pricing in Berlin. It employs supervised learning methods, multiply linear regression and decision tree, to develop a price prediction model. In the meantime, it intends to explore the main contributing factors in pricing.

Keywords: Statistical Learning · Multiple Linear Regression · Decision Tree.

1 Introduction

Berlin is one of the most important cities in Germany and therefore Airbnb business there deserves to explore. This project is designed to develop the price prediction model for listings on Airbnb Berlin and explore how different features influence the price. It tries to interpret the pricing and figure out the contributing factors by regression tools, linear models and decision trees. Moreover, it looks for the most powerful predictive model among them.

2 Data Description

The dataset used for the analysis is Airbnb listings in Berlin and was compiled on July 12th, 2021, available in *Inside Airbnb*. It contains **18,204** observations and **16** variables. The variables in the dataset are listed as following with type in brackets:

- ***ID***
- ***Name***
- ***Host_id***
- ***Host_name***
- ***Neighbourhood_group***: location information (character)
- ***Neighbourhood***: location information (character)
- ***Latitude***: location information (number)
- ***Longitude***: location information (number)
- ***Room_type***: description of room (character)
- ***Price***: response variable (number)
- ***Minimum_nights***: amount of nights minimum (number)
- ***Number_of_reviews***: number of reviews (number)
- ***Last_review***: latest review date (date)
- ***Reviews_per_month***: number of reviews per month (number)
- ***Calculated_host_listings_count***: number of listings per host (number)
- ***Availability_365***: number of days when listing is available for booking (number)

Since latitude and longitude can provide sufficiently precise location information, neighbourhood and neighbourhood_group are abandoned. Also, reviews_per_month tells similar feature as number_of_reviews does. So, in this analysis, only ***Latitude***, ***Longitude***, ***Room_type***, ***Minimum_nights***, ***Number_of_reviews***, ***Calculated_host_listings_count*** and ***Availability_365***, **7** variables are selected for price prediction. All the samples with null values are removed.

2.1 Response Variable

The response variable here is ***price*** which is numeric, and therefore all the analysis will be focus on regression. From **Table 1**, the minimum value of price is zero which is impossible. And maximum value is far larger than the 3rd quantile 81, which is quite unusual and should be cleaned in the beginning. To prevent extremely large values which can be viewed as outliers from impacting models, I remove the samples with price larger than mean by 3 standard deviations and the ones with price equal to zero. Now the maximum is 481 with respect to 3rd quantile 80.

Variable	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Price	0.0	35.0	52.0	73.3	81.0	8000.0
Latitude	52.34	52.49	52.51	52.51	52.53	52.66
Longitude	13.10	13.37	13.41	13.40	13.44	13.76
Minimum_nights	1	2	3	9.106	5	1124
Number_of_reviews	0	1	4	21.64	17	620
Calculated_host_listings_count	1	1	1	3.136	2	76
Availability_365	0	0	0	91.27	175	365

Table 1: Summary of Numeric Variables

Before the analysis, it is important to check the normality of price distribution before using linear regression tools against violating the assumptions. **Fig.1** presents the density plots for price and log of price respectively. Apparently, price is not normally distributed but after taking logarithm, the situation improves. Since the dependent variable price is not normally distributed while logarithm is nearly normally distributed, all the models will set log of price as target in the following analysis.

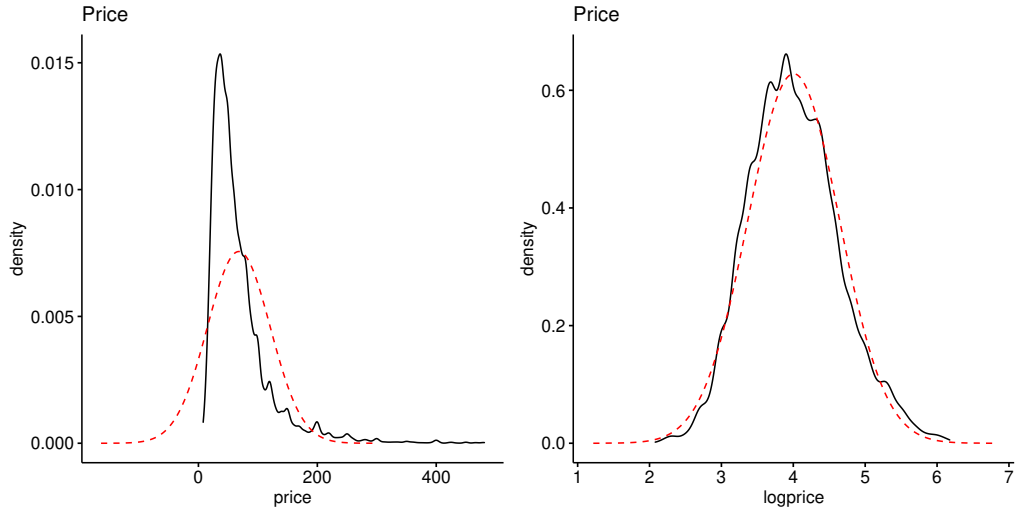

Fig. 1: Normality Check of Price

2.2 Variable Description

Table 1 also presents the summary of independent numeric variables. Maximum value of minimum_nights is 1124 which is more three years, tending to a permanent residence. Moreover, number_of_reviews, and calculated_host_listings_count all have extremely large maximum values. To prevent these values from impacting regression and simultaneously from losing too much information, **99%** quantile value of the four independent variables are calculated, which are 100, 244, and 44, and the observations whose variable values higher than the quantile threshold are removed.

The correlation between variables illustrated in **Fig.2** gives an overview of the possible mutual relationship of variables. Price is positively correlated to availability_365. But surprisingly, price is not highly correlated to location. The price distribution in **Fig. 3** also implies that the price variation may not depend on the location. Moreover, calculated_host_listings_count and number_of_reviews are significantly correlated to availability_365.

Room_type is categorical variable which has four values which are Entire home/apartment, Private room, Shared room and Hotel room. From **Fig.4**, obviously, price is correlated to room type. Mean price of hotel room is much higher than that of the other types, while share room has the lowest mean price. Entire home/apartment is more expensive than private room. However, the difference between shared room and entire home/apartment have close confidence interval of price. To be more precise, ANOVA test is implemented between room type and price to prove the observation

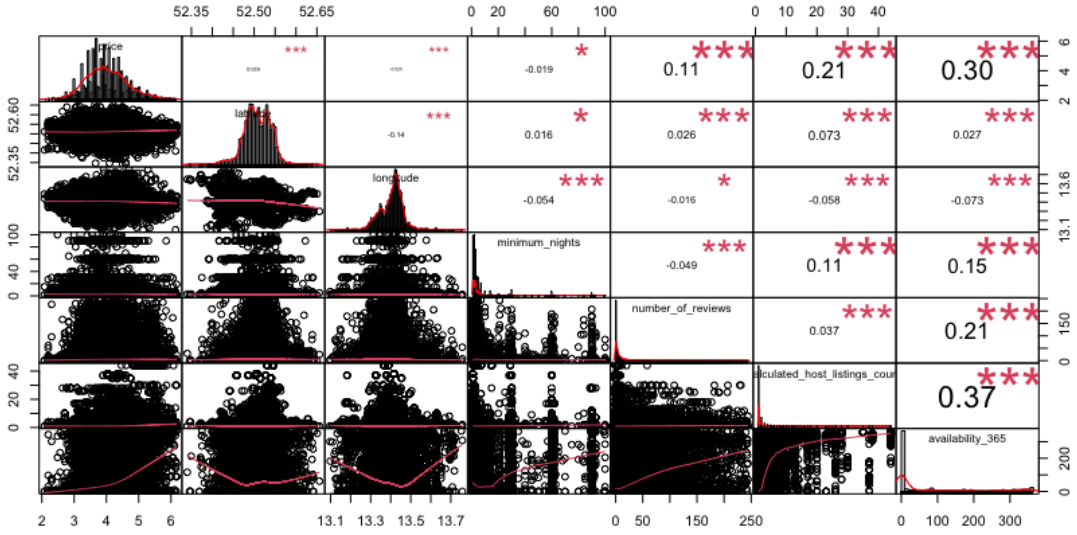

Fig. 2: Correlation between Numeric Variables

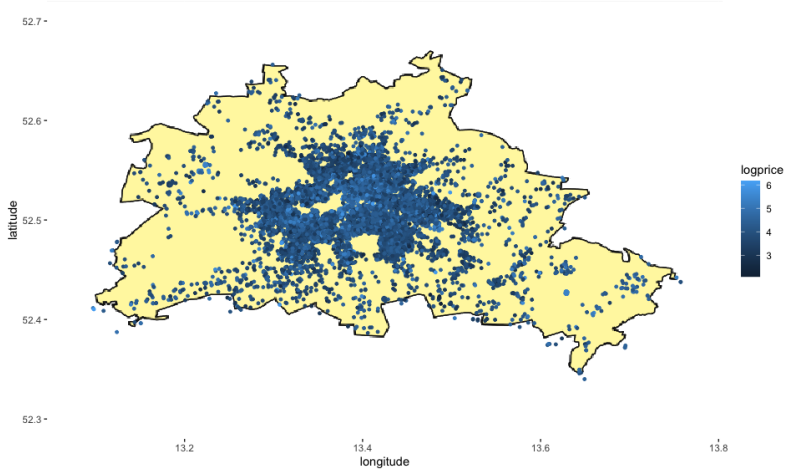

Fig. 3: Price distribution in Berlin

from the boxplot. The test result rejects the null hypothesis which the mean of each group is equal as expected. The final p-value is less than 2 $\times 10^{16}$, meaning that at least one group is significantly correlated to the price. In the **Fig.4**, apparently, the mean and standard deviation of Private room and Shared room is close to each other. It is necessary to test whether the two category value have the same impact, or in the other words, to test whether Private room and Shared room can be combined into one group. Since Private room have 7,835 samples while Shared room has only 206, I sample 206 samples from Private room to generate a balanced group and implement unpaired t-test on the two groups. The result p-value is far smaller than 0.05, rejecting the hypothesis that their difference in means is equal to zero, and therefore we do not need to combine the categories.

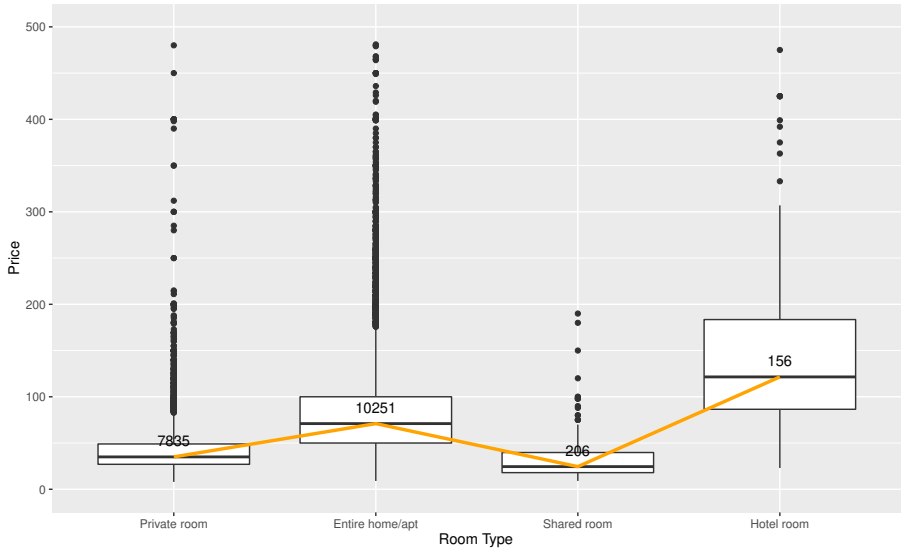

Fig. 4: Boxplot between Room Type and Price

2.3 Data Splitting

After previous operations, the dataset now contains **18,448** samples and it is randomly split to training set and test set, **70%** and **30%** of the data respectively.

3 Metrics

To evaluate the performance of the models and compare the models, metrics **Root Mean Squared Error** (RMSE) and **R Squared** (R^2) are employed.

4 Multiple Linear Regression

Firstly, I train the linear model with all the features. Since room_type is categorical variable, it will be treated as dummy variables, setting Private room as baseline. From the result in **Table 2**, longitude has high p-value 0.95 and the data does not reject the null hypothesis that the independent variable is not correlated to the dependent variable, so longitude is not significant for price prediction.

4.1 Subset Selection and Regression

To remove the irrelevant features and improve the performance of regression model, this work employs subset selection method to grow the model. Here I use **Forward Stepwise** method to enforce subset selection. This algorithm starts from a null model containing no predictors, and then adds the predictor which gives the greatest additional improvement to the model at each step, until all of the predictors are in the model. Bayesian information criterion is used for comparing the models

and pick the optimal one.

As the left plot of **Fig.4** illustrates, the model with 7 predictors is the optimal model, having the lowest BIC values. The right plot depicts that the optimal model picks the variables: room_type, minimum_nights, number_of_reviews, calculated_host_listings_count and availability_365. Location information variables are abandoned.

Table 2: Full Model

	Coefficients	t value	P-value
Intercept	-1.537e+01	-2.143	0.03214*
latitude	3.768e-01	2.830	0.00465***
longitude	-6.545e-02	-0.949	0.34249
room_type-Entire hotel	6.942e-01	77.179	$<$ 2e-16***
room_type-Shared room	-3.882e-01	-9.334	$<$ 2e-16***
room_type-Hotel room	9.525e-01	19.703	$<$ 2e-16***
minimum_nights	-7.159e-03	-22.405	$<$ 2e-16***
number_of_reviews	3.899e-04	3.258	0.00112**
calculated_host_listings_count	8.746e-03	9.904	$<$ 2e-16***
availability_365	1.134e-03	29.629	$<$ 2e-16***

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

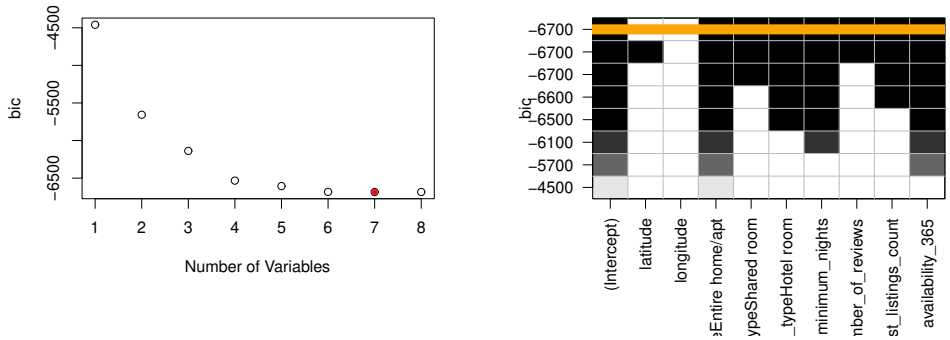

Fig. 5: Results of Variable Selection

After variable selection, I retrained the linear model and the results are presented in **Table 3**. All the variables in he model are significant now. This linear model now achieves RMSE 0.4764and R^2 0.414 on test set.

4.2 Diagnostics

Variance Inflation Factor (VIF) measures how much the variance of an estimated regression coefficient is increased because of collinearity. The existence of collinearity will not reduce the power of the model, but it inflates the variance and thus the standard errors of the estimates, so leading to wider confidence intervals and less statistically reliable estimates. Variables having the square root of VIF higher than 2 is considered as collinear with other variables. Fortunately, all the variables do not exceed the threshold, so we do not need to worry about multicollinearity in this model.

Table 3: Linear Regression after Variable Selection

	Coefficients	**t value**	**P-value**
Intercept	3.536e+00	486.271	$<$ 2e-16 ***
room_type-Entire home	6.944e-01	77.216	$<$ 2e-16***
room_type-Shared room	-3.853e-01	-9.264	$<$ 2e-16***
room_type-Hotel room	9.555e-01	19.776	$<$ 2e-16***
minimum_nights	-7.137e-03	-22.350	$<$ 2e-16***
number_of_reviews	3.995e-04	1 3.338	0.000846***
calculated_host_listings_count	8.952e-03	10.163	$<$ 2e-16***
availability_365	1.134e-03	29.645	$<$ 2e-16***

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Besides multicollinearity, it is also to check whether the data violate the assumptions made in regression. In **Fig. 5**, Residual vs Fitted plot does not show any curve shape, meaning that the assumption that the error terms have constant variance is not violated. The Normal QQ plot shows a nearly straight line, suggesting that the residuals have a normal distribution. Also, Scale Location plot gives no specific pattern, telling the absence of heteroskedasticity. From Residuals vs Leverage plot, the spread of standardized residuals does not change as a function of leverage. Moreover, we can find several points with high leverage which may influence the model. To be more precise, we can check Cook's distance which is signed as dotted red line in the plot. The points outside the dotted line have high influence, it is necessary to remove them and check the difference.

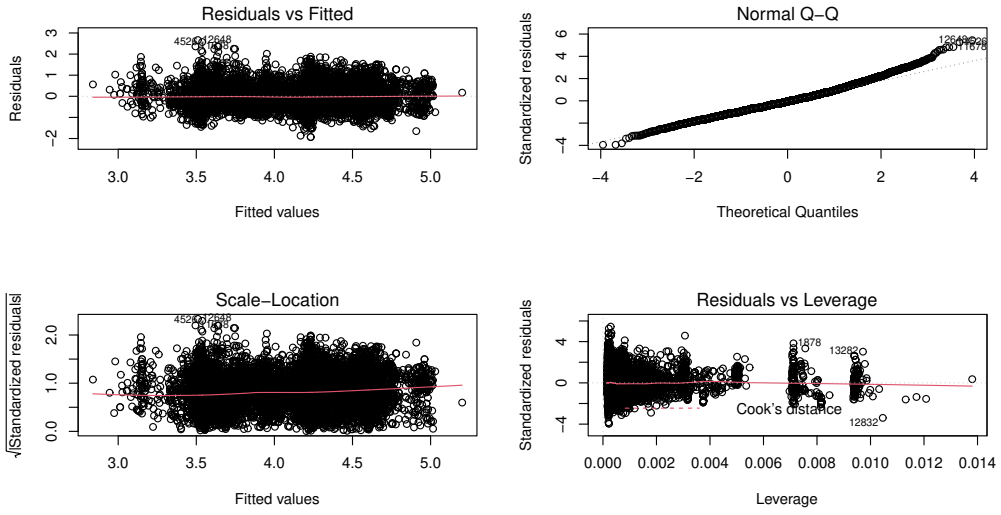

Fig. 6: Residual Plots

4.3 Robust Regression

As we can see in the Residuals vs Leverage plot, some influential observations exist in the model. So in order to get a robust model, robust regression is employed. If removing the outliers directly, they may take away too much information and harm the predictive power of the model. But remaining these data points and treating them the same as other points may impact the estimations. Robust regression is a good choice since it makes a compromise between the two situations. The idea of robust regression is to weigh the observations differently based on how well behaved these observations are. All the data are assigned a weight, the more influential points will get a smaller weight while the normal points will tend to get a weight 1.

By means of robust regression, the estimations are presented in **Table 4**. As expected, robust regression has a better performance on test set. Its R^2 is 0.4141, slightly higher than the simple linear regression.

Table 4: Robust Regression

	Coefficients	**t value**
Intercept	3.5349	493.26
room_type-Entire home	0.6896	78.8234
room_type-Shared room	-0.4314	-10.6631
room_type-Hotel room	0.9416	20.0323
minimum_nights	-0.0071	-22.8197
number_of_reviews	0.0004	3.3696
calculated_host_listings_count	0.0099	11.5228
availability_365	0.0011	29.3675

4.4 Interpretation

In linear regression model, the magnitude of coefficients does not represent the importance of variables but t-value does. Room_type has the largest t-value. If someone transfers from private room to entire home or hotel room in Berlin, the price he suffers will have a significant increase, especially hotel home. But if transferring to a shared room, he will pay less. If someone spend more nights in a room, he will pay less too. And when the room owner has more listings, he tends to offer a higher price. Also, when the listing has longer days available for booking, the price will goes up.

5 Decision Tree

Another tools for the regression task is **decision tree** which is easy for interpretation, but one of main drawbacks is that it is difficult to get satisfactory performance as linear models. So in this work, **Random Forest** is employed to increase the accuracy.

5.1 Single Tree

Here I take a recursive binary splitting to build a regression tree and use cost complexity pruning to prevent a large tree with high variance. In **Fig. 7**, the dashed line represents the highest cross validation error minus the minimum cross validation error, plus the standard deviation of the error at the tree. After 4 splits (5 nodes), the Complexity Parameter (cp) to restrict the size of tree goes to a point where cross validation error smaller than the dashed line value, suggesting that 0.016 can be the optimal point for pruning the tree.

With this cp value, the trained tree is built as showed **Fig.8** and all the data are categorized into 5 groups. 30%of the training data have mean log of price at 3.5. They are Hotel room or Entire apartment and have availability_365 less than 66 days. Another 13% of listings with same room type but longer available days than 66, have higher mean log price which is 3.8. For the listings with private room or shared room, 30% of training data have mean log price 4.2 with availability shorter 18 days, 5% have minimum_nights no less than 18 days and availability_365 less than 18 days, and 22% have minimum_nights less than 11 days and availability_365 no less than 17 days. This tree is built only with 3 variables, room type, availability_365 and minimum_nights.

According to the tree, we can conclude that rooms with type of private room and shared room tend to have a lower price. Less available days also decrease the price while less minimum_nights help drive price down.

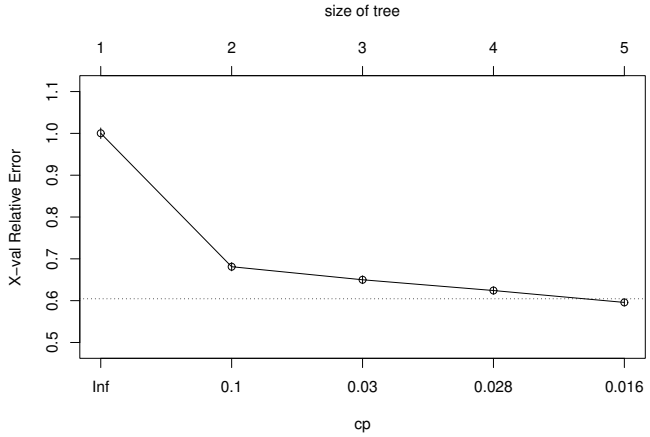

Fig. 7: Complex Parameter

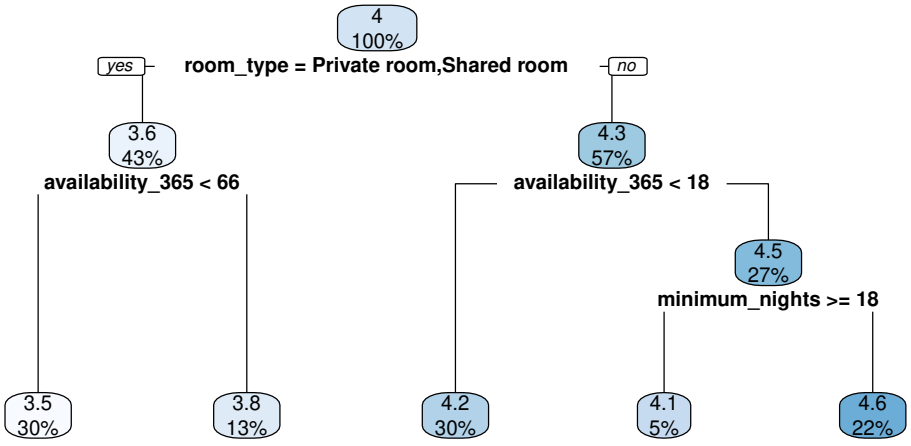

Fig. 8: Result of Decision Tree

5.2 Random Forest

Although the pruned single tree gives a good interpretation for the data on how the variables impact on the price, the accuracy is not high. To address this issue, **Random Forest** is employed. **1000** trees are grown to develop random forest. Finally, it gets 0.4405 on RMSE and 0.5112 on R^2, which is much better both linear model and single tree.

Fig. 9 depicts the importance of variables in the modeling. Same as the linear model, random forest takes room_type and availability_365 as most contributing factors in pricing. However, random forest stresses location more than the linear model.

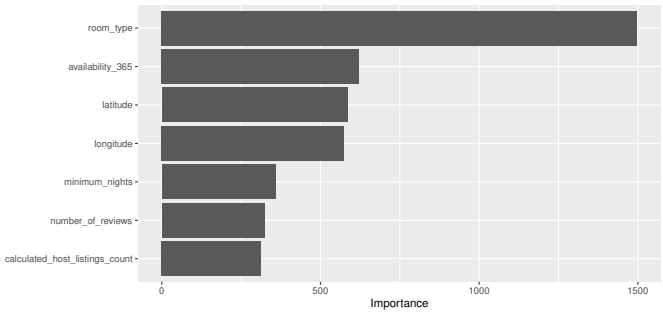

Fig. 9: Variable Importance of Random Forest

Table 5: Summary of the Model Performance

	RMSE	**Squared R**
Full Linear Model	0.4837	0.4106
Subset Linear Model	0.4836	0.4108
Robust Regression	0.4836	0.4109
Single Tree Model	0.4873	0.4019
Random Forest	0.4405	0.5112

6 Conclusions

Table 5 summarizes the 5 models in this project. After removing the location information, subset linear model gets a higher R^2 and lower RMSE. Robust regression does bring stability and hence it has highest R^2 among the three linear models. Single tree model is explainable and takes the same top 3 variables as most contributing factors. But it fails to explain more variance than linear models. Random forest undoubtedly achieves the best performance. The main difference between random forest and the other models is that it concerns more the information.

Apparently, room type and availability_365 are the contributing factors of Airbnb pricing in Berlin. For location information, although the linear models take it as redundant variable, it does not mean the location does not matter for pricing. Through the fact that random forest concerns location and obtains a higher performance, location has its role in pricing but the linear models fails to catch it, i.e. the relation between location and price is not linear.

7 Appendix

This project is performed through R, and code book is shared on Github.