Assignment 2

Balint Parragi, 2023-02-11

Predicting Airbnb prices in small and medium-size apartments

Data collection

Data is from the Inside Airbnb site, I selected the city of *Vienna* for the most recent time period available. The raw data consists of 11955 observations. After filtering for size (small and medium) and number of guests (2-6), and cleaning the data sufficiently, the sample size consists of 8772 observations.

Data exploration and dropping features

Several features of the data is not needed as

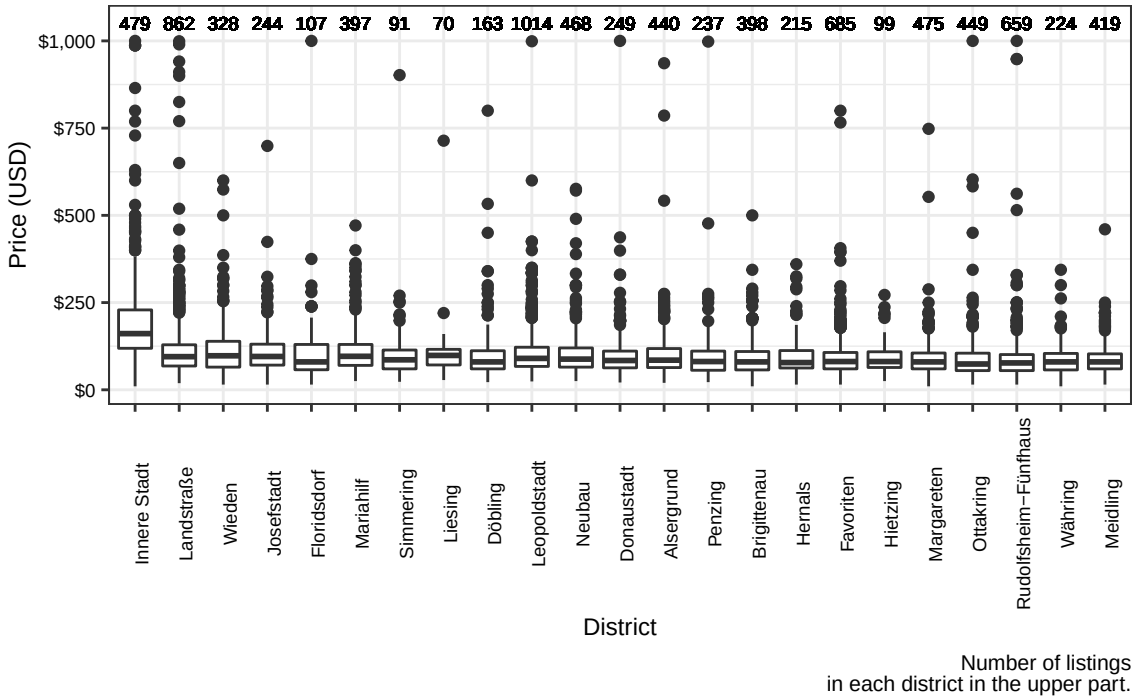
1. They are not conveying any information (large number of missing values which cannot be computed, very low variation, etc.)
2. They are not valid as this task is designed for new listings on the market
3. There are (or there can be created) other features that incorporate the characteristics of that feature as well

Based on these points, the following more important variables were dropped:

- review scores measures, listing's number of days on the market, host response rate, host acceptance rate, bedrooms
- some features needed imputed values - either with the median value, or if categorical, then by negative (false) value
- number of `NA`-s were very low in the remaining features –> required little imputation -some variables could also be dropped from or united among amenities as they have high (0.6,0.7) correlation with each other, but at large samples this causes less severe problems.

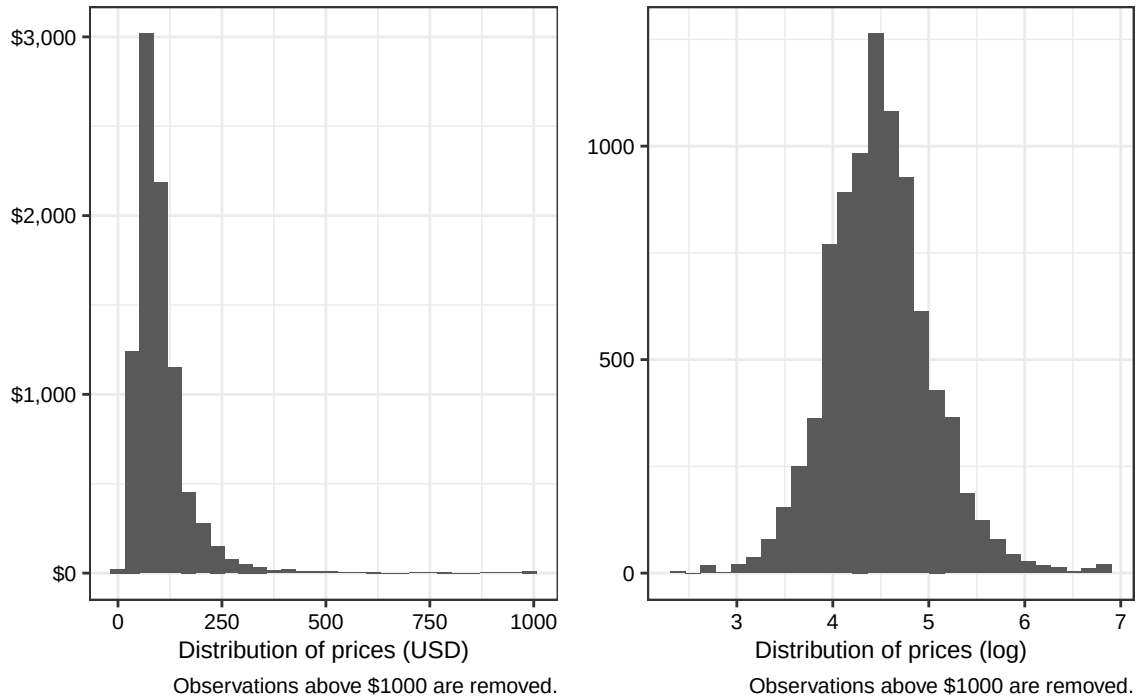
Finally, the target variable (*price usd*) needed filtering as there were some massive outlier in the data. However, a threshold of 1000$ was sufficient, which resulted in the loss of less than 1% of the data.

Location



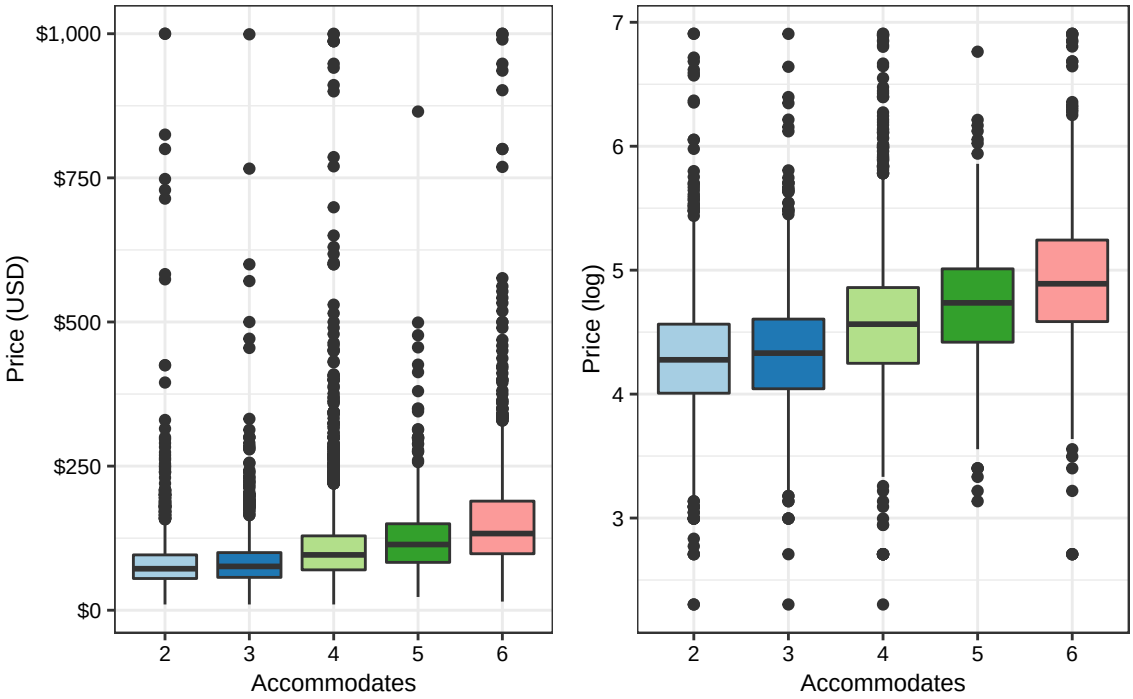
Using domain knowledge, I assume that location should be some of the most important factors in deciding the price. Unsurprisingly, Innere Stadt (first district) clearly stands out, it is much more expensive on average than any other districts in Vienna. Landstrasse (3rd), Wieden (4th), Josefstadt (8th) Mariahilf (6th) and Neubau (7th) are also on top as they constitute the inner, historic part of the city, close to the canal and several sights. Some others are on the shore of the Danube (Florisdorf, Leopoldstadt), which can come with better views. Also, Leopoldstadt contains the most observations, as it is both close to the Danube and to the center, ideal for people who are seeking entire apartments.

Prices

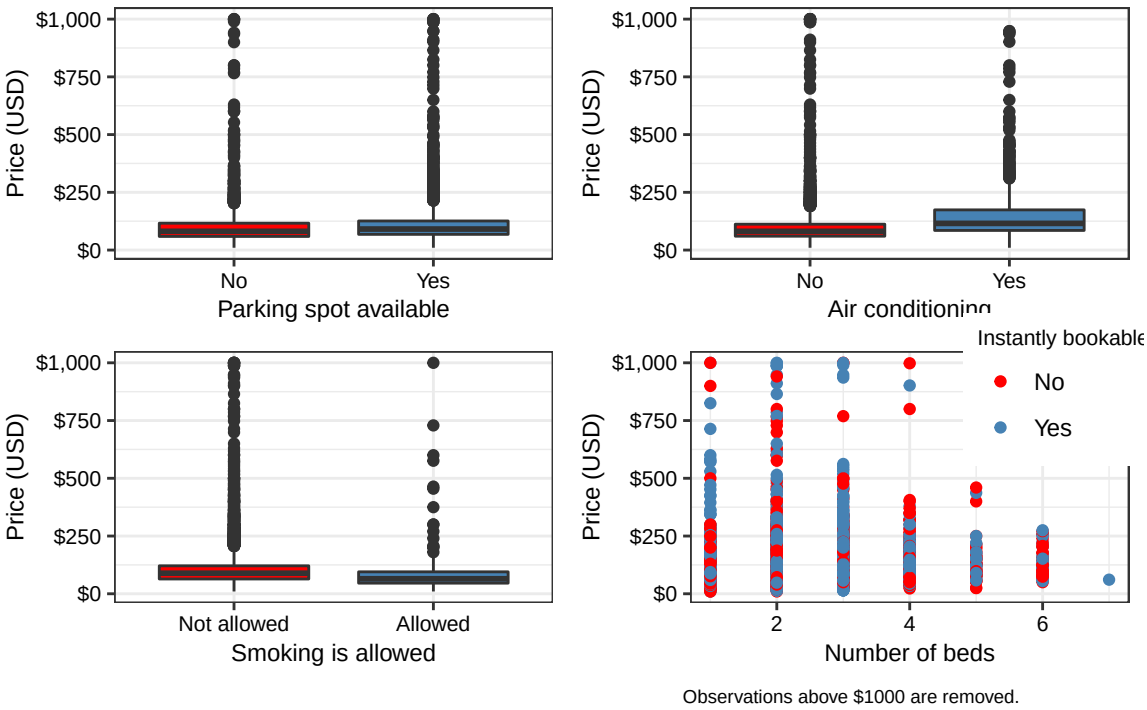


Prices are as usual in the data: very skewed to the left, and mostly below 500 USD/night with a mean around 104.3707752. The log price plot shows a much clearer view, more resembling to normally distributed.

Accommodates



Prices are increasing with number of accommodates, which is again not surprising. The nominal price plot show a larger variability, differences between the log categories are more subtle.



Observations above $1000 are removed.

Other features show that listings having air-conditioning, (bit counterintuitively) being non-smoking are more expensive. Parking spots seemingly do not lead to any price difference. Number of beds - most likely because it is linked to number of accommodates - has a positive relationship with prices as well as the possibility of instant booking.

Models and methods used

The target variable is **nightly price** across all models.

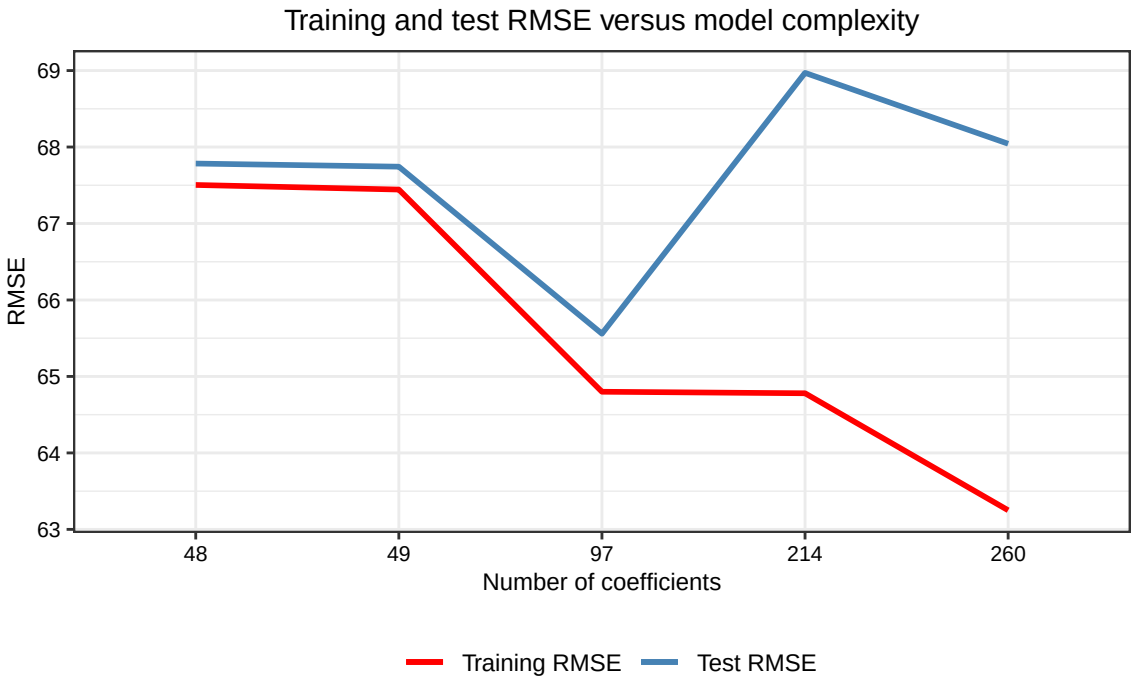
1. Model 1: OLS model; $accommodates+beds+propertytype+bathrooms+neighbourhood+instantbookable+hostattributes$
2. Model 2: OLS model; $Model_1 + accommodates^2$
3. Model 3: OLS model; $Model_2 + interactions$
4. Model 4: OLS model; $Model_1 + amenities$ ($Model_1$ sic!)
5. Model 5: OLS model; $Model_2 + amenities$
6. Model 6: LASSO, based on $Model_4$
7. Model 7: Random Forest with basic tuning

The interactions are based on domain knowledge and include:

- $accommodates \times property\ type$, $airconditioning \times property\ type$, $pets \times property\ type$
- $accommodates \times neighbourhood$, $property\ type \times neighbourhood$

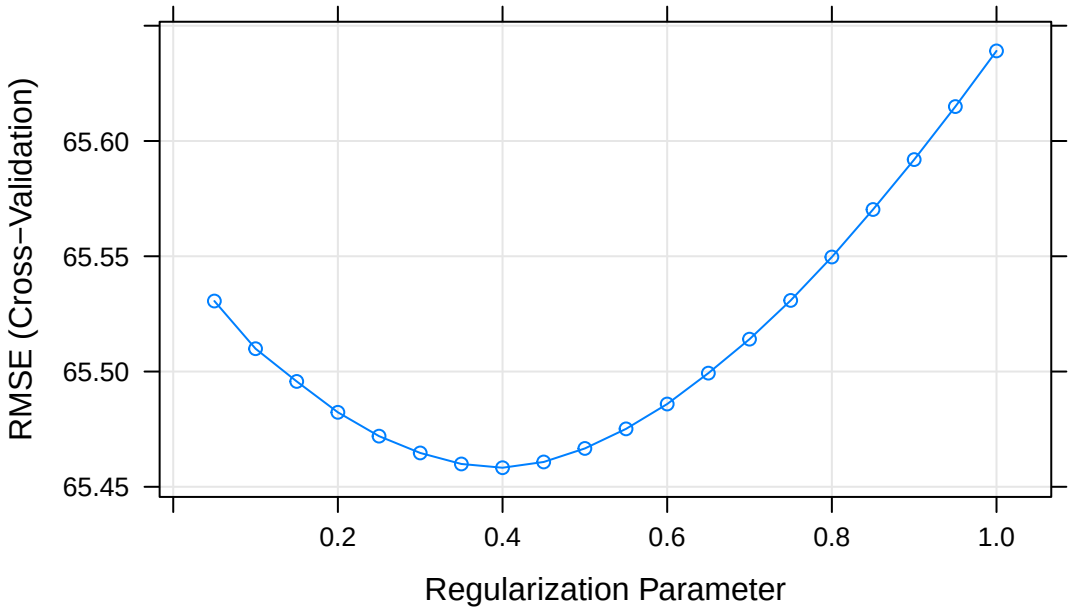
Model	Coefficients	R_squared	BIC	Training_RMSE	Test_RMSE
M1	48	0.242	74538.9	67.504	67.784
M2	49	0.244	74535.9	67.444	67.743
M3	214	0.302	75456.2	64.780	68.970
M4	97	0.302	74431.7	64.800	65.559
M5	260	0.335	75546.0	63.250	68.043

$Model_4$ stands as the best model of these five with lowest Test RMSE and BIC. It indicates that amenities are needed in general, but the interactions do not improve the model fit (even more so, worsening the fit by much), but still, the most complex model's ($Model_5$) performance is not bad either. Plotting Test and Train RMSE-s shows the same.



LASSO based on the best OLS

Next model (LASSO) is based on $Model_4$. After a basic tuning (only parameter here is λ), the following plot shows which tuning parameter is the most optimal.



The best λ is 0.4, this is where the cross-validated RMSE is the lowest.

Many coefficients - 22 out of 98 - has been shrunk to zero, mostly property types (because of low sample size each), a few districts and several amenities dummy (f.e all the ones related to babies-children). These features apparently are among the ones that did not improve the fit as much as they increased the variance according to the algorithm.

Model	Coefficients	R_squared	BIC	Training_RMSE	Test_RMSE
M1	48	0.242	74538.9	67.504	67.784
M2	49	0.244	74535.9	67.444	67.743
M3	214	0.302	75456.2	64.780	68.970
M4	97	0.302	74431.7	64.800	65.559
M5	260	0.335	75546.0	63.250	68.043
LASSO	76	0.287	NA	NA	65.458

The LASSO model produces a better Test RMSE, indicating that it might be a better model than the previously seen OLS models.

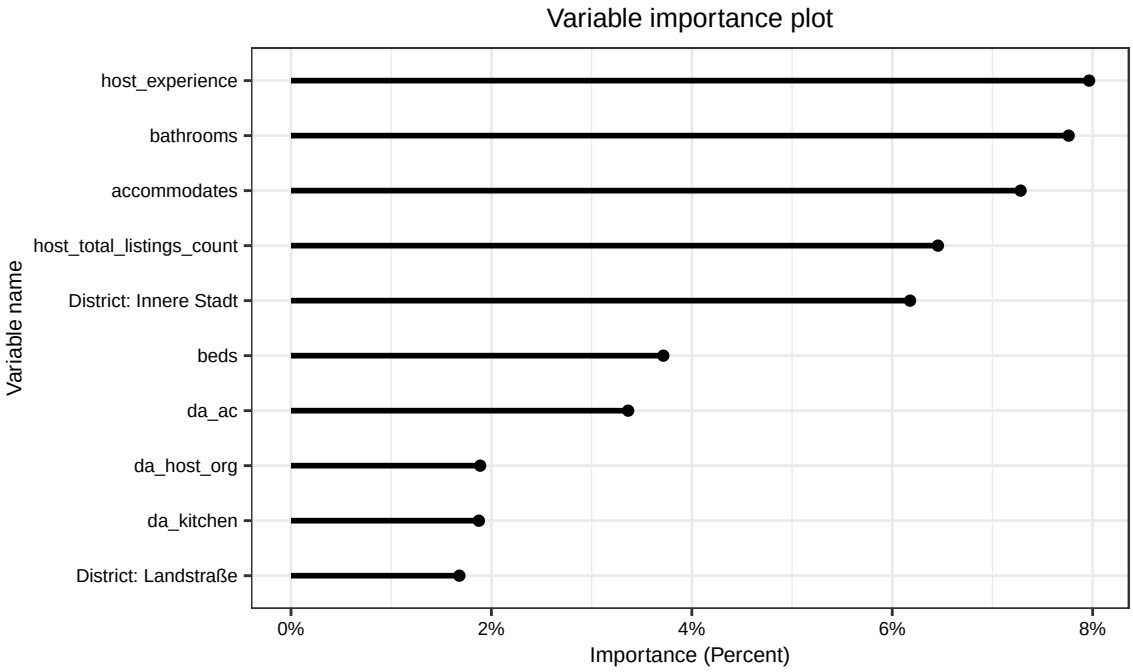
Random Forest (RF)

Going one step further, now using RF in order to predict Airbnb prices. Then, there is no need to define functional form and choose which variables to include. It only needs a stopping rule (depth rule, minimum node size), splitting rule (square root of features) and number of trees to grow, I define two models with somewhat different parameters to decide which provides a better fit.

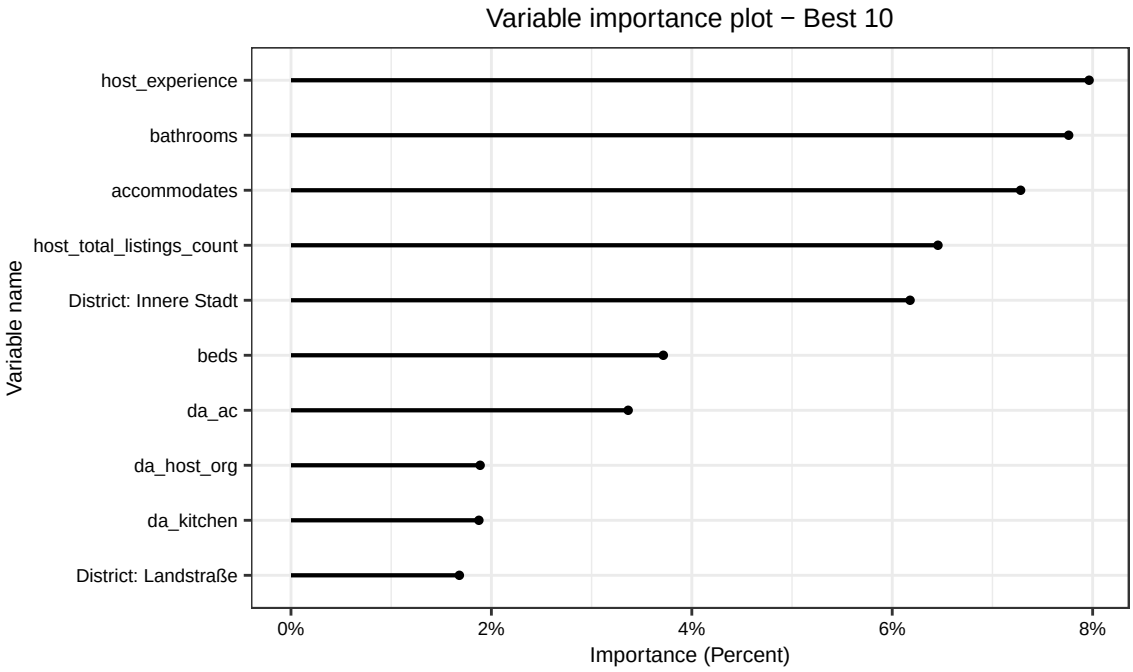
	Min vars	Min nodes	RMSE
Model 1	5	10	67.694
Model 2	12	5	62.036

Decision: model 2 is selected with minimum node size at 5 and selected variables to split equal to 12. With this RF model, I plot the variables with their respective importance to decrease the variance of the model in three ways:

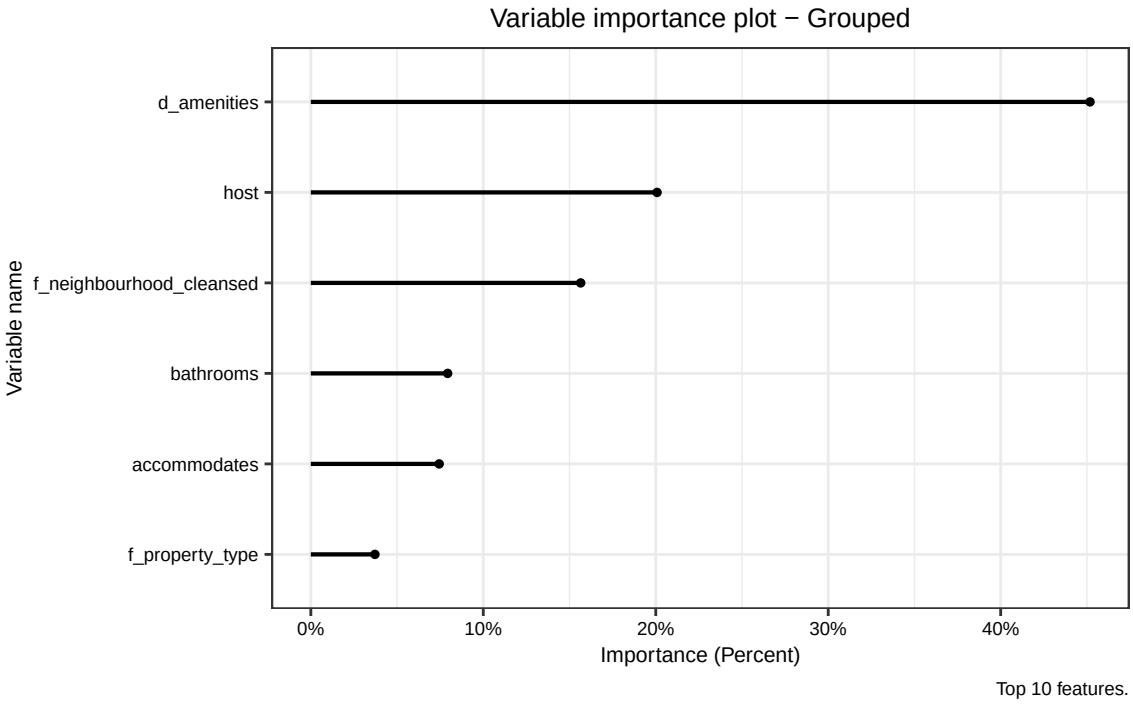
- Total (above an importance threshold)
- Best 10
- Grouped according to broader features and factor types



Importance threshold above 500.



Top 10 features.



Model comparison and evaluation

The final and arguably the most important part is the model evaluation, how the specified models perform on the holdout set and how accurate the prices predictions are. But before, let us have a final look at the Test RMSE comparison of all the models.

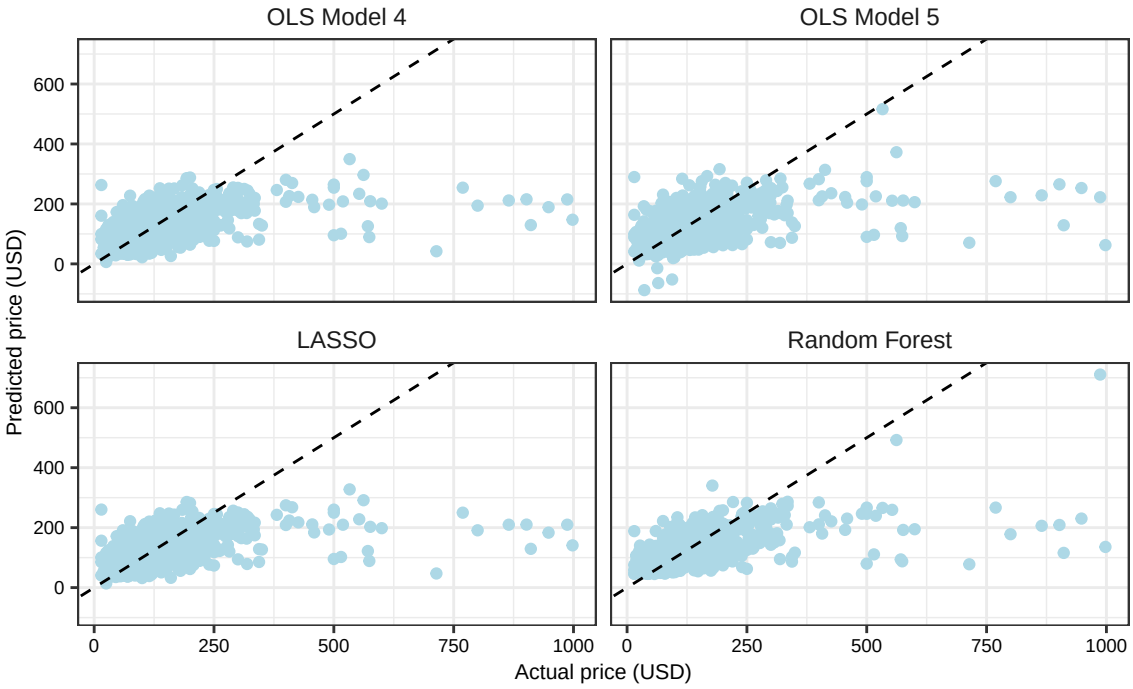
	M1	M2	M3	M4	M5	LASSO	Random Forest
Test RMSE	67.784	67.743	68.97	65.559	68.043	65.458	62.036

	RMSE on hold-out sample
Model 4	68.300
Model 5	68.420
LASSO	68.273
Random Forest	63.236

The Random Forest model provides a clearly lower RMSE on the holdout sample fit, meaning it can predict prices with a little lower error on average. As a conclusion, this should be the chosen model for price prediction.

Additional plots on prediction

Price predictions can be shown via scatter plots as well - we might discover some unique patterns.


OLS Model 4
OLS Model 5
LASSO
Random Forest
Predicted price (USD)
Actual price (USD)
0
200
400
600
0
250
500
750
1000