Price prediction for Airbnb Listings in Bangkok

Bruno Helmeczy

01/02/2021

Abstract

This report investigates how to predict the prices at which apartments in Bangkok can be rented out on Airbnb. Below I discuss a number of features taken into account, compare 6 prediction models of varying complexity, & after choosing & re-estimating my final model, I investigate expected performance across property types, the number of people accommodated, while investigating most important accommodation features. Based on 5-fold cross validation, I found the expanded Random Forest model predicting log-transformed prices to perform best in terms of all metrics. After re-estimation, the final model boasted 54.1% R-squared, & performed with 21 dollar MAE & 51.9 dollar RMSE, ca. 95% of the average price during out-of-sample testing. Finally, model diagnostics showed the model to perform fairly consistently given accommodation size (2-6 people) & property type (apartment or condominium), yet RMSE decreased by ca. 40%, to 28 dollar RMSE, when considering the most frequent neighborhood in the dataset: Khlong Toei.

Data, Cleaning & Feature Engineering

The raw dataset was downloaded from- & is available at AirBnB-s website (from here), & comprises over 19.7K observations & 74 Variables, summarizing all accommodations in the Bangkok area available for rental, as of 23rd-24th December, 2020. After keeping only apartments-, & condominiums hosting between 2-6 people, 9962 observations remained to clean for analysis. To manage this feature space, i.e. to keep what's important & drop what is not, while looking to maintain intuition throughout the process, I grouped variables into 7 subjects (excluding ID variables): **Host** information, **Geo-Spatial**-, & **Property** information, **Sales**-, & **Availability**-related data, & **Reviews**-, & **Listings** data.

Some of these groups proved rather not useful, e.g. from **Geo-spatial** variable group only the **neighbourhood_cleansed** variable is retained, as latitudinal-, & longitudinal metrics are indirectly already captured by the neighbourhood they fall into, while the raw **neighbourhood** variable held 500+ values, also posing a linguistic challenge. Similarly, **Listings**-group variables were in essence all dropped, as they contain information regarding listings of different room/apartment types, or contain information already present in **Host**-related variables. My substantive rationale for that is **a)** the number of listings attributed to a host above all are a proxy for the likelihood of him/her being a scammer, rather then an attribute impacting rentals' pricing, & **b)** even on AirBnb, this information is presented only as a host having either 1, or multiple listings online. As such, seeing as prices are rather market-, then inventory-based, a binary variable showing whether the host has only 1 listing seems sufficient. To be safe, the integer variable denoting how many listings a host has, is also kept, so a LASSO model could capture if either is less important by dropping it.

Speaking of **Host**-variables, I view their primary purpose is indicating whether a listing is a scam or not, as these are what both Airbnb & prospective guests use this information for. Thus, 6 of 8 Host-variables are logical / binary variables, while host verifications are a list of credential details (similarly to amenities being a list of objects in the rental property). Any single 1 of these credential details is not important, say whether a host has an email address or Facebook profile, however the total number of credentials may indicate hosts' credibility, & professionalism, so I simply counted the **number of verifications** a host provided.

Guest Reviews-related variables: As proxy for a hosts' experience & recent activity, Nr. of days since their 1st of last reviews were calculated. Ca. 34% NAs, for hosts without reviews - seen from the same observations also missing review scores, & Nr. of reviews per month. In terms of review ratings, only the overall review scores are kept, being an aggregation of sub-scores, & for Nr. of days since 1st & last reviews, instead of the median-, the maximum values were imputed, signifying the quesi-infinite number of days the host has had a review. To impute NAs for review scores, I took the arithmetic mean instead of the median. Though the mean is less robust to extreme values, there is some level of uncertainty arising in guests' minds, if seeing there are no reviews what so ever for a listing. I intend to take this into account by using the mean instead.

Sales & Availability-related variables are what hoteliers would consider most obviously relating to prices, & besides price, also include minimum-maximum stay-restrictions, as well as the number of available room nights in the coming periods of different length. I argue the number of room nights a host has sold more closely relates to what price he asks for, rather then the number of room nights he has available in the future, given the confidence it may instill. So, I transformed these to denote the number of rooms sold in future periods of different lengths, & eventually reduced these to factor variables. With stay-restrictions, I 1st wrote a function to calculate how identical these columns are, allowing me eventually to retain only minimum stay nights, as maximum nights contained many extreme-, & missing values.

Property Variables The final, & perhaps most important feature engineering decision is regarding **Property**-related variables, most strongly contributing the dropping observations, & 48 dummy variables standing for amenities. To obtain data only for apartments, I 1st filtered the room_type variable to only include entire homes or apartments, & also removed listings where the property type was described as e.g. Room, Castle, hostel, or treehouse & other properties of similarly different nature. This still left some non-trivial property types, like Bungalows, Guest Suites, Lofts & Villas, but I view these as legitimate alternatives in potential guests' minds, whereas I hazard, the type of guest looking to book a Castle or a Treehouse is materially different then someone looking to stay in apartments.

The greatest cleaning challenge however, is of handling the **amenities** column. After creating a dummy-variable table of all possible amenities, I wrote 3 functions to handle this variable: 1st to coerce columns whose names match certain sets, & combinations of keywords; 2nd to remove duplicate columns, & 3rd to check the frequency of Yes-es in the remaining dummy columns, latter looking to remove amenities with imbalanced values, by myself defined with having less then 1% Yes values. All this resulted in reducing the number of amenities dummies from 262 to 48.

Modelling

Eventually, I grouped my variables into **Property-**, **Host-**, **Reviews-**, & **Amenities**-related, seeing these as the varying degrees, & order of additional information a guest might be looking for, to decide whether to stay at a given listed rental or not. For OLS & LASSO models, I also incorporated interactions between key property-related variables & all amenities, & predicted log-transformed price in 5 of 6 models, seeing how skewed it is. Please see the finalized variable groups, & derived predictor groupings in bulleted lists below:

Price distribution	Logged−Price distribution
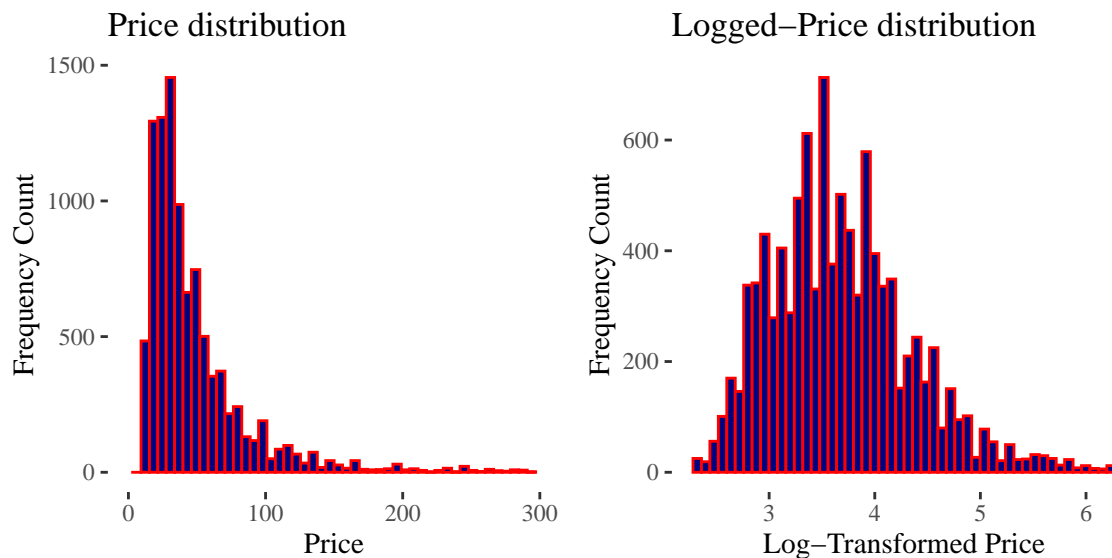
- **Property**-related: Property Type, Nr. of Guests accommodated (level, log-,level squared, log-squared), Nr. of Beds-, Bedrooms & Bathrooms, Neighbourhood Cleansed, Has Availability, Nr. of Rooms Sold in coming 30-, 60-, 90-, & 365 days.
- **Host**-related: Host is owner, Nr. of hosts' listings & verifications, & whether he greets you, accept every reservation, has a profile pic, & his/her identity is verified.
- **Reviews**-related:
- **Amenities:**
- **Interactions:**

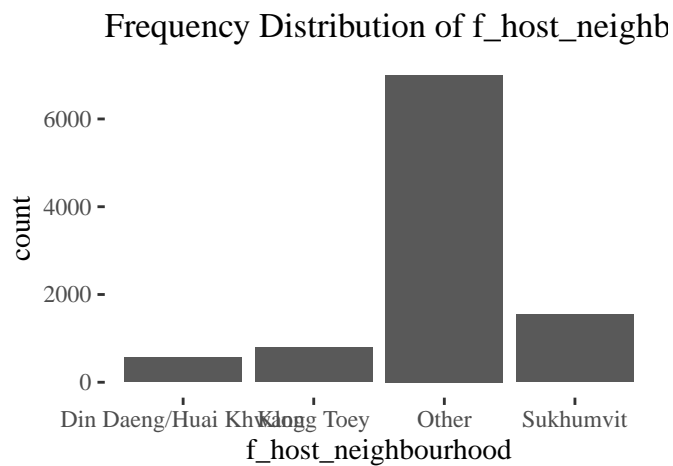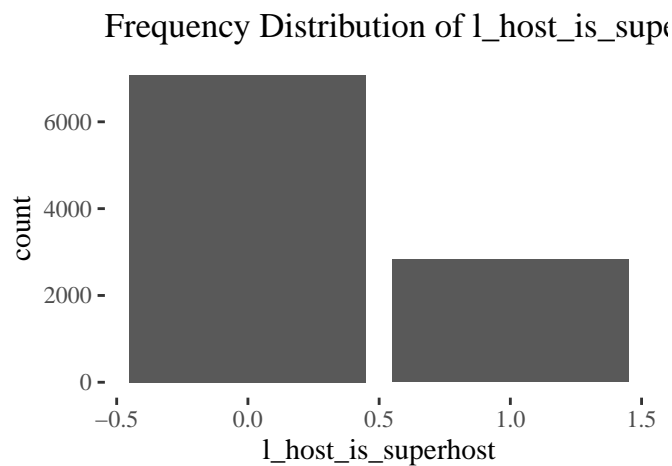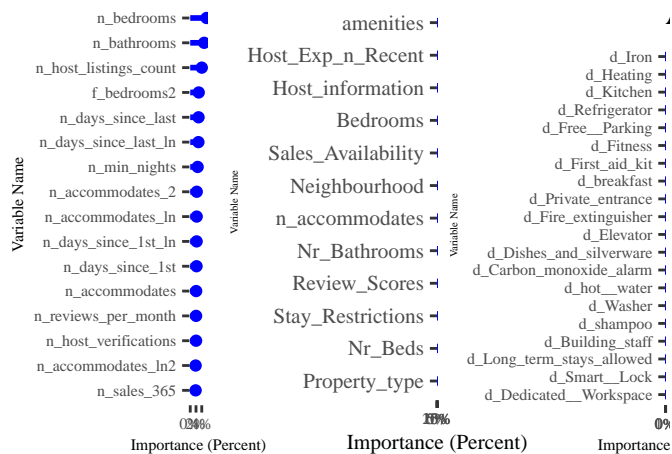| models	Rsq	MAE	RMSE	RMSE_norm
ols_model	0.446	20.9	41.1	0.792
lasso_model	0.479	19.7	39.6	0.763
rf_model_1	0.516	11.5	29	0.559
rf_model_2	0.551	10	26.6	0.513

Rsq	MAE	RMSE	RMSE_norm
0.557	18	36.3	0.721

Model Implications

Actual versus Predicted Prices
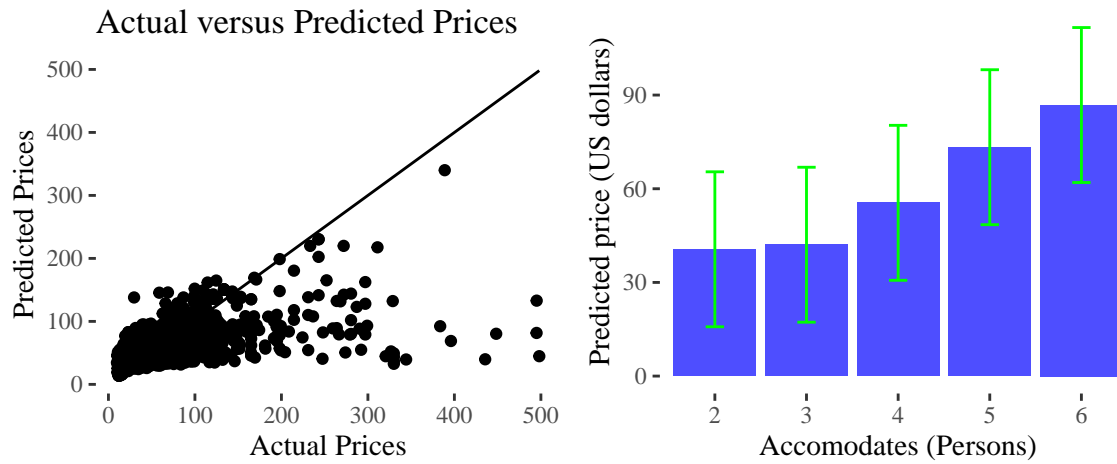
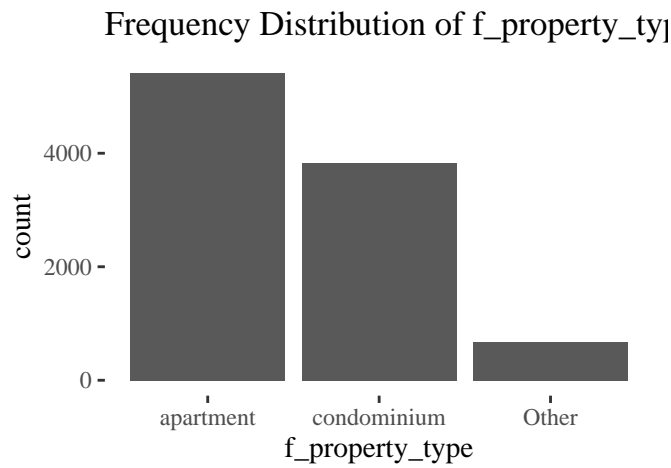

Var.1	RMSE	Mean.price	RMSE.price
Apartment size			
2	34.8	43.5	0.8
3	27.2	43.7	0.62
4	43.8	59.2	0.74
5	43.5	74.8	0.58
6	40.7	84.6	0.48
Type			
apartment	32	49.2	0.65
condominium	36.5	47.8	0.76
Neighbourhood			
Khlong Toei	19.3	42.9	0.45
Vadhana	38	62.5	0.61
Huai Khwang	24.9	40.5	0.61
Ratchathewi	29.1	51.5	0.57
Sathon	27.5	48.9	0.56
All	36.3	50.4	0.72

Variable Importance

Frequency Distribution of l_host_is_supe...

Frequency Distribution of f_host_neighb...

Frequency Distribution of n_host_listing...

Frequency Distribution of n_host_verific...

Frequency Distribution of l_host_has_p

Frequency Distribution of l_host_identity

Frequency Distribution of f_property_typ

Frequency Distribution of n_accommoda

Frequency Distribution of n_bedrooms

Frequency Distribution of n_beds

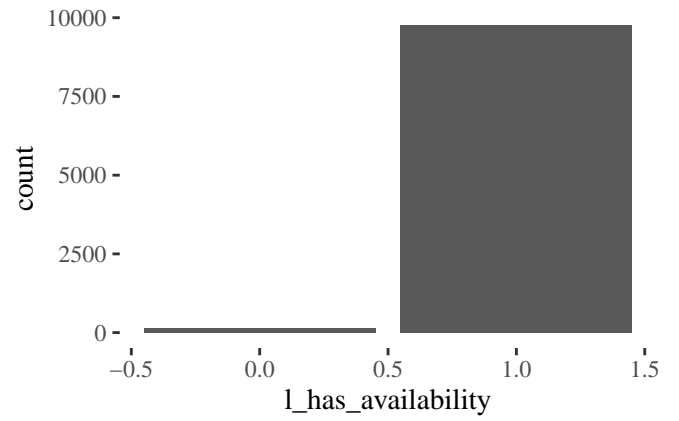## Frequency Distribution of n_min_nights
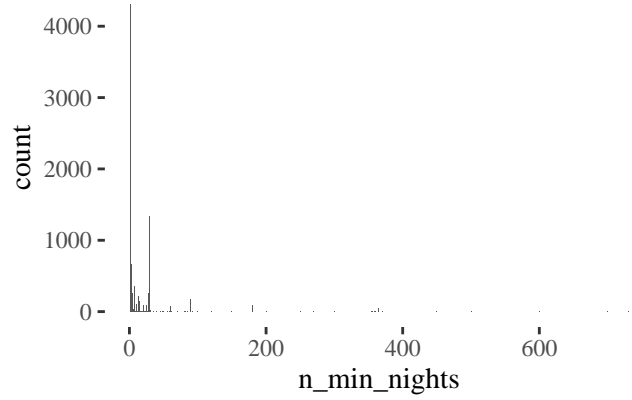
Frequency Distribution of l_has_availab

Frequency Distribution of n_sales_30
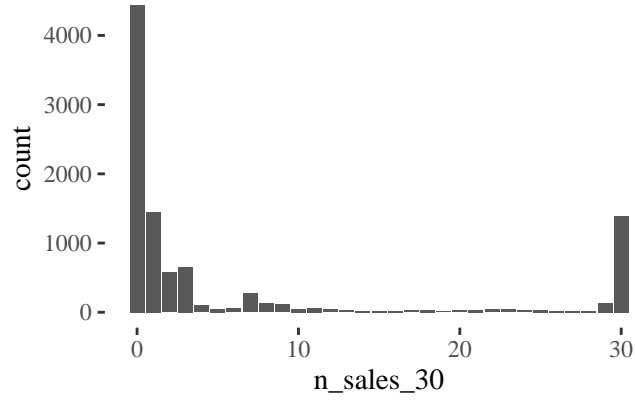
Frequency Distribution of n_sales_60
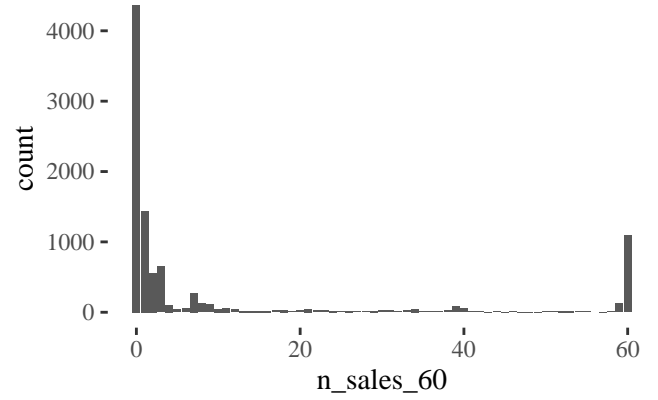
Frequency Distribution of n_sales_90
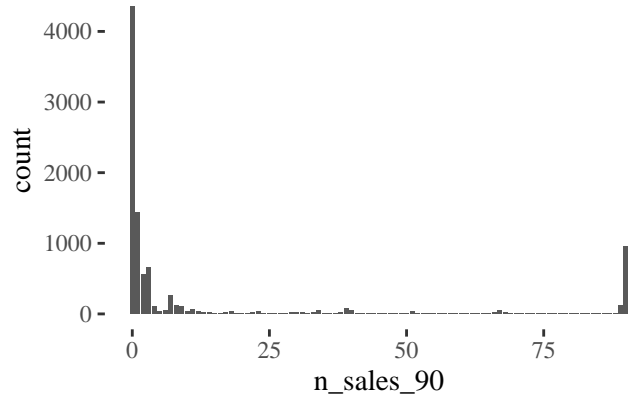
Frequency Distribution of n_sales_365
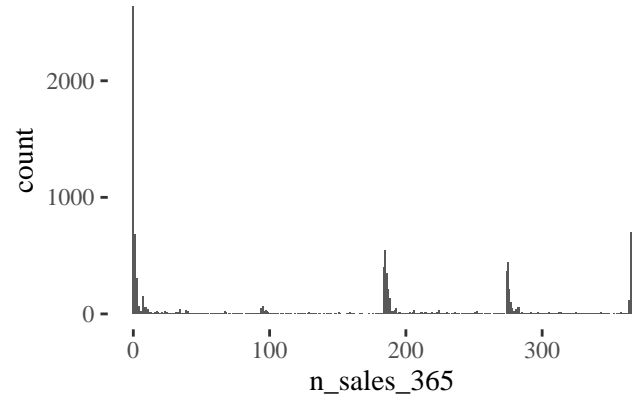
Frequency Distribution of n_number_of_
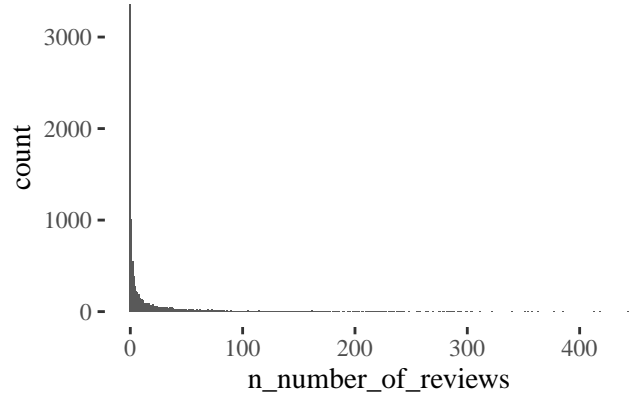
Frequency Distribution of n_review_scor
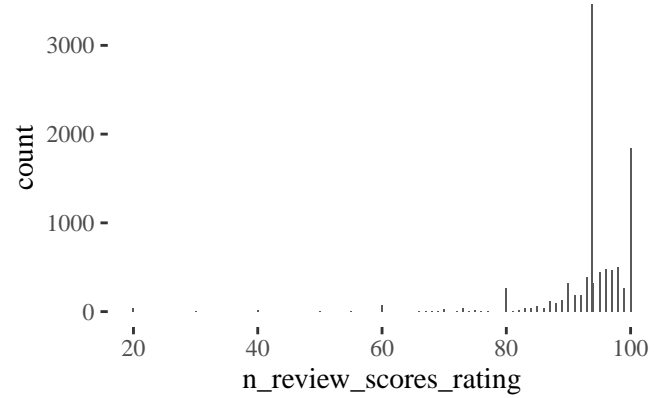
Frequency Distribution of l_instant_bool
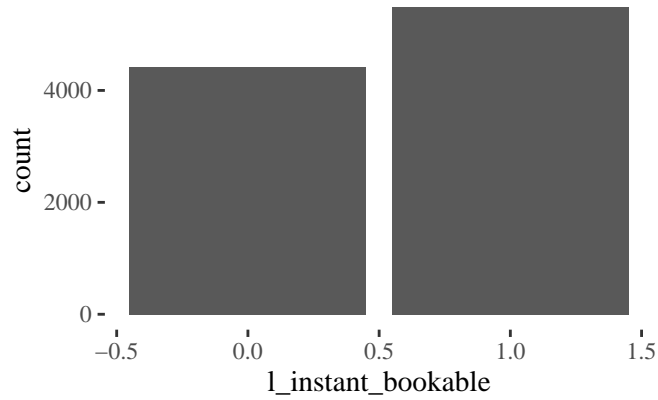
Frequency Distribution of n_reviews_per
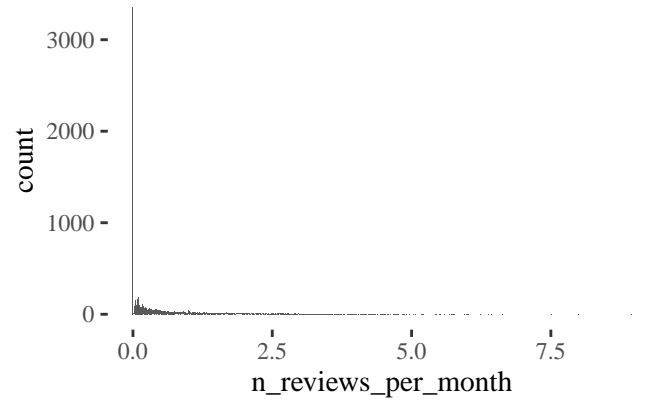
Frequency Distribution of f_neighbourho
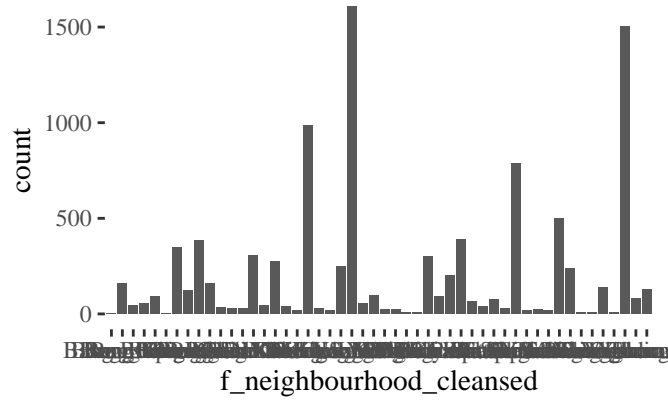
Frequency Distribution of n_bathrooms
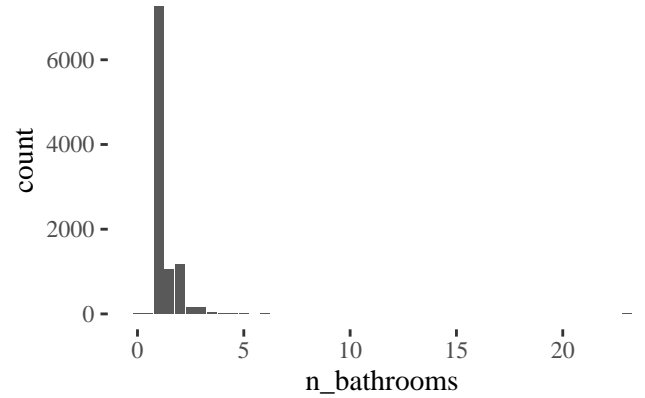
Frequency Distribution of usd_price
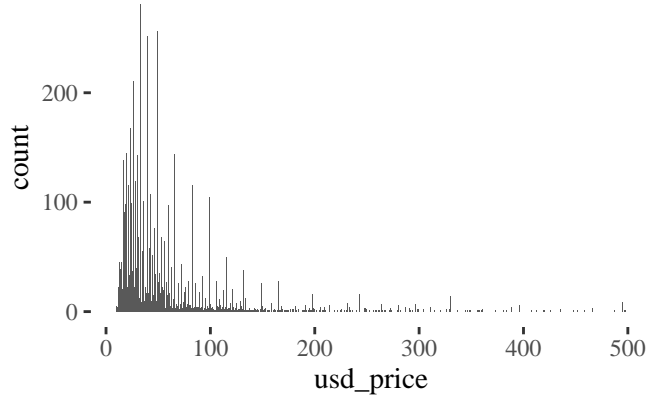
Frequency Distribution of flag_reviews_
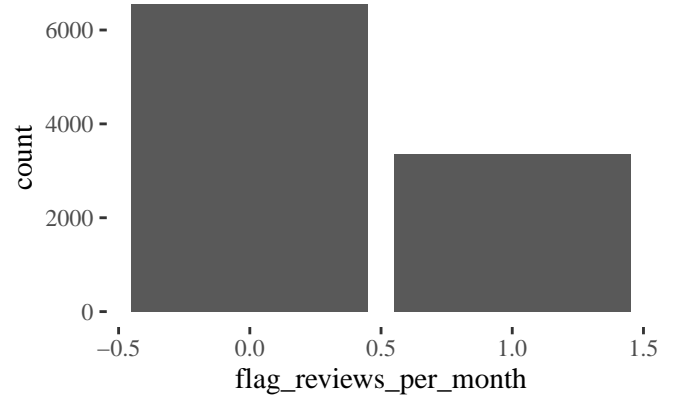
Frequency Distribution of flag_review_s
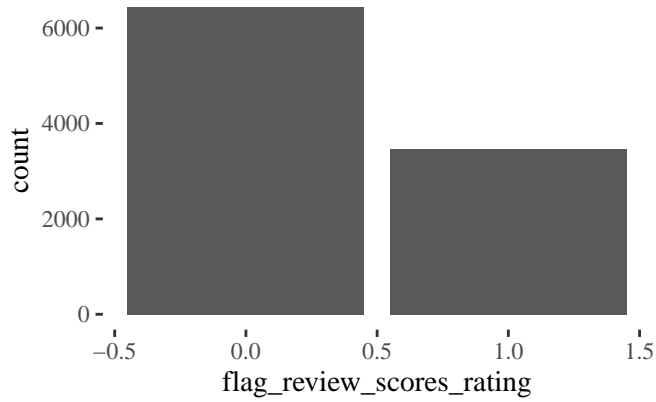
Frequency Distribution of n_days_since_
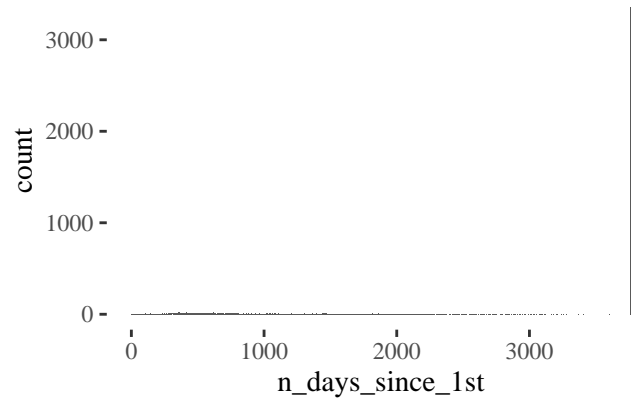
Frequency Distribution of n_days_since_

Frequency Distribution of d_Cleaning_b

Frequency Distribution of d_Extra_pillow

Frequency Distribution of d_First_aid_k

Frequency Distribution of d_High_chai

Frequency Distribution of d_Iron

Frequency Distribution of d_Patio_or_ba

Frequency Distribution of d_Private_ent

10

Frequency Distribution of d_Room_dark

Frequency Distribution of d_Single_leve

Frequency Distribution of d_Waterfront

Frequency Distribution of d_Window_g

Frequency Distribution of d_wifi

Frequency Distribution of d_HDTV

Frequency Distribution of d_Dedicated_

Frequency Distribution of d_Paid__Park

Frequency Distribution of d_Free__Park

Frequency Distribution of d_Refrigerator

Frequency Distribution of d_Fitness

Frequency Distribution of d_stove

Frequency Distribution of d_shampoo

Frequency Distribution of d_hot__water

Frequency Distribution of d_Washer

Frequency Distribution of d_air__condi

Frequency Distribution of d_Smart__Lo

Frequency Distribution of d_Dryer

13

Frequency Distribution of d_Kitchen

Frequency Distribution of d_Oven

Frequency Distribution of d_Children

Frequency Distribution of d_Microwave

Frequency Distribution of d_garden

Frequency Distribution of d_breakfast

Frequency Distribution of d_Bed_linens

Frequency Distribution of d_Bread_mal

Frequency Distribution of d_Building_st

Frequency Distribution of d_Carbon_mo

Frequency Distribution of d_Coffee_mak

Frequency Distribution of d_Cooking_ba

Frequency Distribution of d_Dishes_and

Frequency Distribution of d_Elevator

Frequency Distribution of d_Essentials

Frequency Distribution of d_Fire_exting

Frequency Distribution of d_Hangers

Frequency Distribution of d_Heating

Frequency Distribution of d_Host_greets

Frequency Distribution of d_Indoor_firep

Frequency Distribution of d_Keypad

Frequency Distribution of d_Long_term_

Frequency Distribution of d_Luggage_d

Frequency Distribution of usd_price_ln

Frequency Distribution of n_days_since_

Frequency Distribution of n_days_since_

Frequency Distribution of n_accommoda

Frequency Distribution of n_accommoda
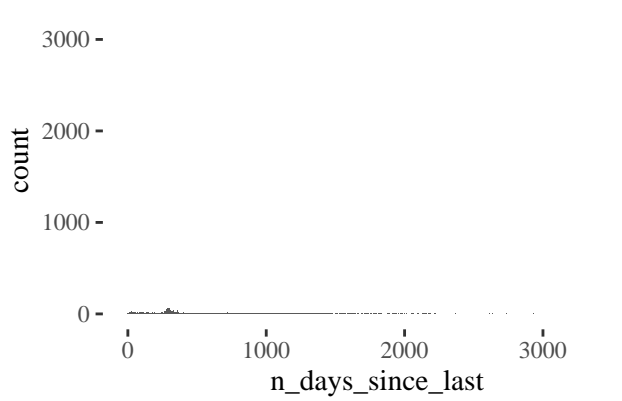
Frequency Distribution of n_accommoda

Frequency Distribution of f_has_1_revie

Frequency Distribution of f_review_scor

Frequency Distribution of f_number_of_

Frequency Distribution of f_sales_365

Frequency Distribution of f_sales_90

Frequency Distribution of f_sales_60

Frequency Distribution of f_sales_30

Frequency Distribution of f_min_nights

Frequency Distribution of f_beds

Frequency Distribution of f_bedrooms

Frequency Distribution of f_bathrooms

Frequency Distribution of f_host_listing

Frequency Distribution of flag_n_days_s

Frequency Distribution of f_host_accept

NULL