Airbnb Price Prediction

Team 4: Roman Frič, Johana Krčková, Filip Stárek, Petr Šot

Course: Data-X 2023/2024

1 Data Understanding

Reviews

The reviews dataset includes 8066 unique listings over 4379 days. Listings start on 2010-05-07 and end on 2023-09-17. There are only 23 null values in “comments”. Value types are as follows: ids (int), date (datetime), name (str), comments (str). There are no outliers.

	listing_id	id	date	reviewer_id	reviewer_name	comments
0	3884	847944	2012-01-07	1400020	Steve	We had a very comfortable 6 night stay in Regi...
1	3884	1697446	2012-07-13	2021169	Morrin	We ended up staying in Regina's own apartment ...
2	23163	101588	2010-09-20	227165	Nathan	Incredible apartment in an ideal location. The...
3	498646	1445317	2012-06-09	2476967	Lea	We had a great time in Romans flat. The appart...
4	23163	157152	2010-12-22	286036	Hugh	The apartment was huge, we felt like we were s...

The following table shows the language distribution on a sample of 1000 comments. Highlighted languages are those suitable for sentiment analysis via bert-base-multilingual-uncased-sentiment model.

comments	
en	653
de	67
fr	58
cs	49
es	35
it	28
sk	17
ru	17
ko	16
nl	12

Calendar

All “listing_id” have calendar entries with maximum 365 days and minimum 350 days, meaning practically the whole year is covered for all listings. There are no null values. Value

types are: id (int), date (datetime), available (binary str), prices (int), max/min nights (int). High outliers in price detected and kept as legitimate observations.

	listing_id	date	available	price	adjusted_price	minimum_nights	maximum_nights
0	3884	2023-09-17	f	$1,449.00	$1,449.00	5	365
1	3884	2023-09-18	f	$1,449.00	$1,449.00	5	365
2	3884	2023-09-19	f	$1,449.00	$1,449.00	5	365
3	3884	2023-09-20	f	$1,449.00	$1,449.00	5	365
4	3884	2023-09-21	f	$1,449.00	$1,449.00	5	365

Listings

There are in total 7,8% missing cells in Listings dataset.

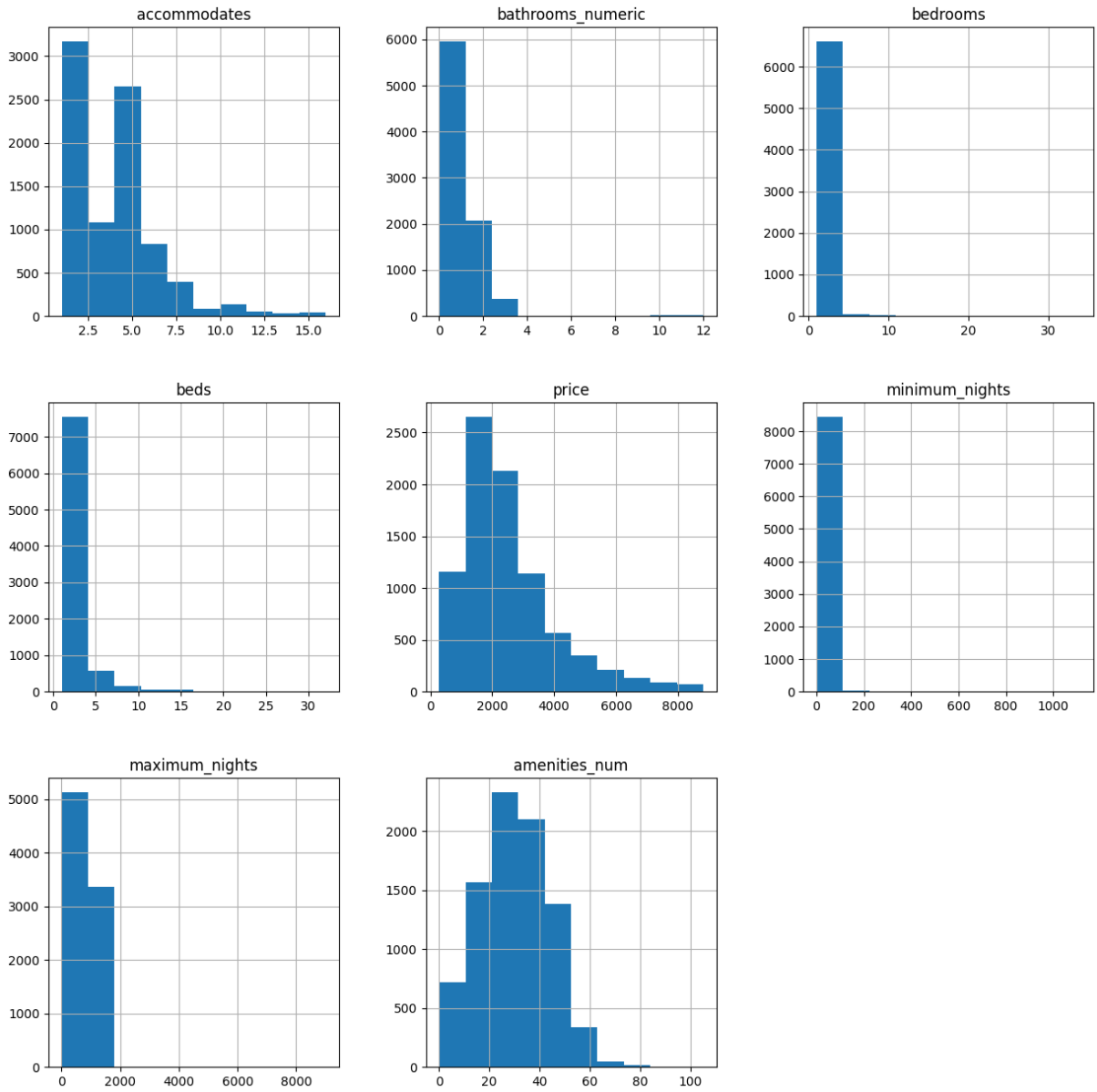
Numeric variables: latitude, longitude, accommodates, bathrooms, bedrooms, beds, minimum_nights, maximum_nights, minimum_minimum_nights, maximum_minimum_nights, minimum_maximum_nights, maximum_maximum_nights, minimum_nights_avg_ntm, maximum_nights_avg_ntm, calendar_updated, host_id, host_response_time , host_acceptance_rate , host_total_listings_count

Categorical variables: neighbourhood_cleansed, property_type, room_type, host_is_superhost, host_neighbourhood, host_verifications, host_has_profile_pic, host_identity_verified

Unstructured variables: name , description, picture_url, host_location , host_about, amenities

Datetime variables: host_since

Histograms show outliers in multiple columns:



2 Data Preparation

Removed columns:

Columns not relevant for our model:

- latitude, longitude – don't include much information by themselves, neighbourhood better describes the location of the listing
- minimum_minimum_nights, maximum_minimum_nights, minimum_maximum_nights, maximum_maximum_nights, minimum_nights_avg_ntm, maximum_nights_avg_ntm – unclear meaning, enough information is included in minimum_nights and maximum_nights

- adjusted_price – same as price
- id (of comment), reviewer_id, reviewer_name – irrelevant
- name,description,picture_url,host_id,host_since,host_location,host_about,host_response_time,host_acceptance_rate,host_is_superhost,host_neighbourhood,host_total_listings_count,host_verifications,host_has_profile_pic,host_identity_verified – These column are either irrelevant or were used for a feature which rendered them obsolete

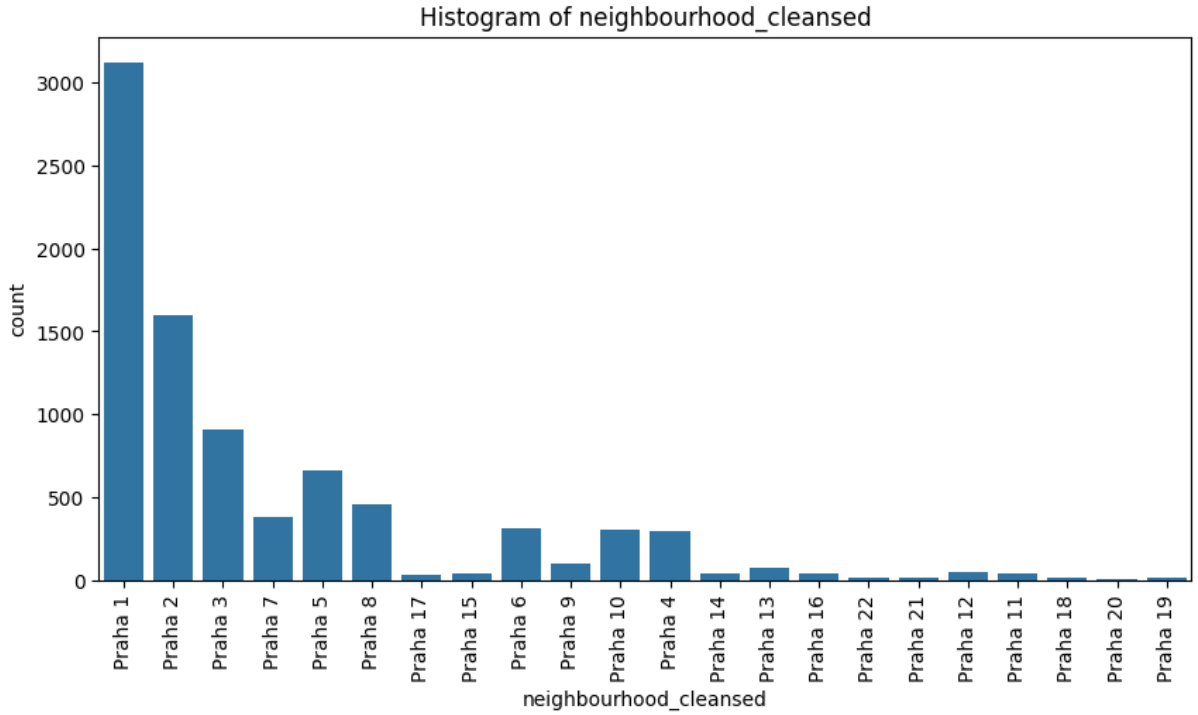
Empty columns: neighbourhood_group_cleansed, bathrooms, calendar_updated

Other columns:

- neighbourhood – bad data quality, a lot of missing values
- bedrooms – a lot of missing values
- property-type – self-selected, therefore there were a lot of inconsistent categories

Modified columns:

We unified the neighbourhood_cleansed column into Praha 1 – Praha 22 neighbourhoods, so it does not include smaller neighbourhoods such as Zličín, Kunratice etc.



Then we extracted the number of bathrooms from bathrooms_text where the values included values such as “1.5 baths, 1.5 shared baths, 2 baths” etc. The result is a column we called “bathrooms_numeric” with numeric values.

The “availability” column modified from binary str to “busy” column with binary int values

Added columns:

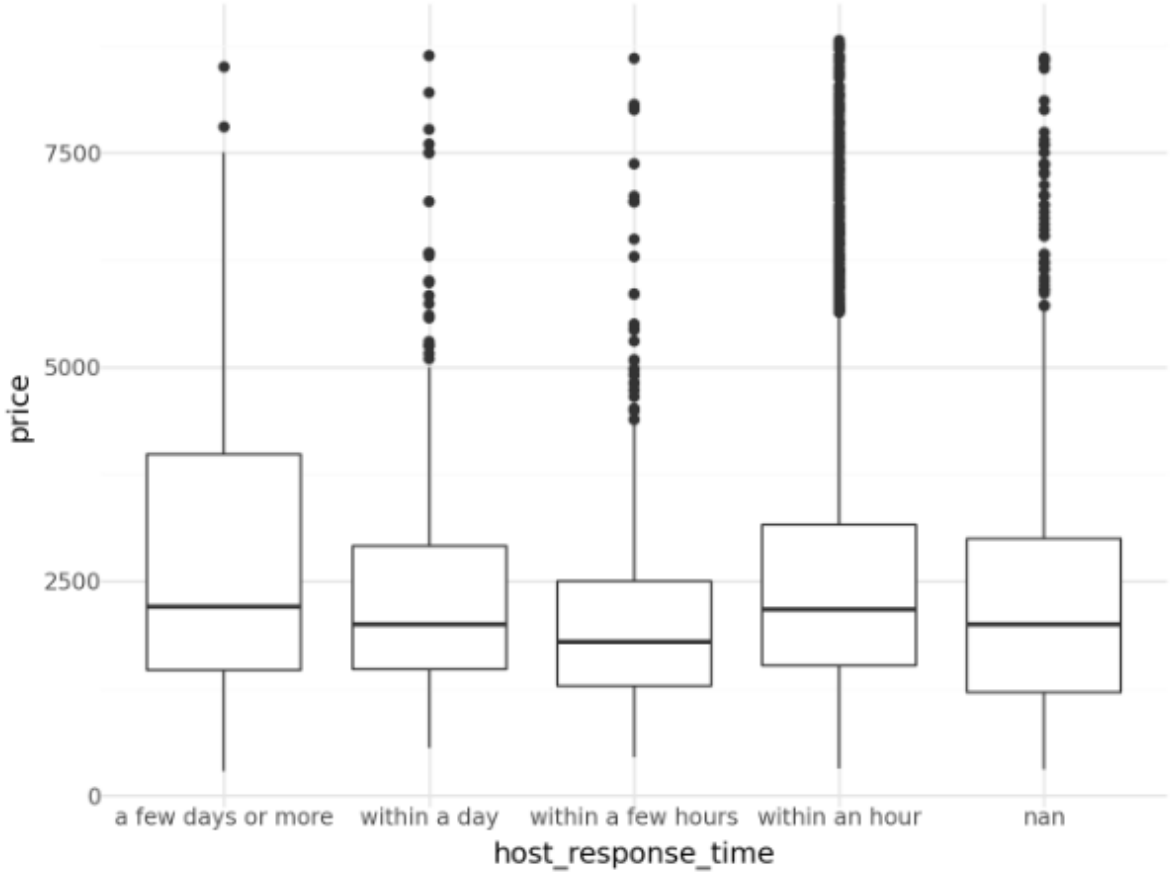
We created some features that we thought might be interesting and useful for the model.

Sentiment_score – mean sentiment score feature of each comment for given listing (bert-base-multilingual-uncased-sentiment model by npltown was used)

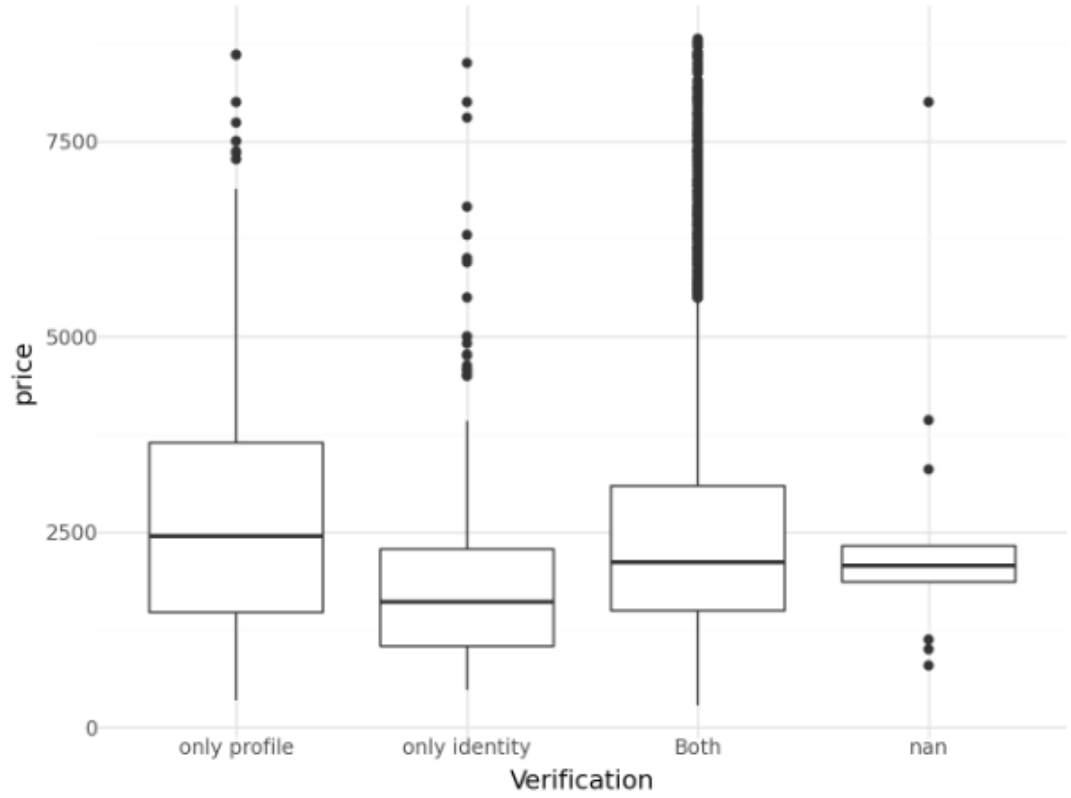
Distance – We've used geolocator libraries to calculate the distance from the host location and their AirBnB.

Weekend, in_season, new_year – binary features describing seasonal price changes derived from date

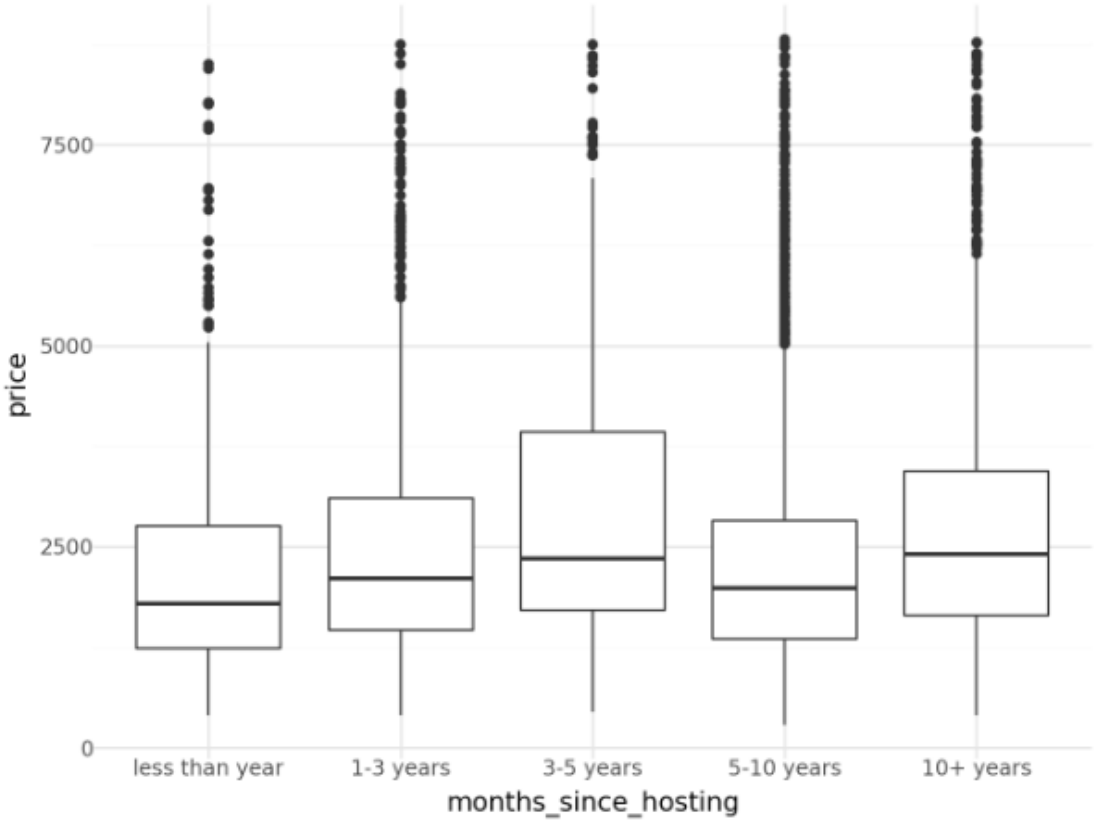
Host_response_time – How long does it take for a host to respond to the client’s request.



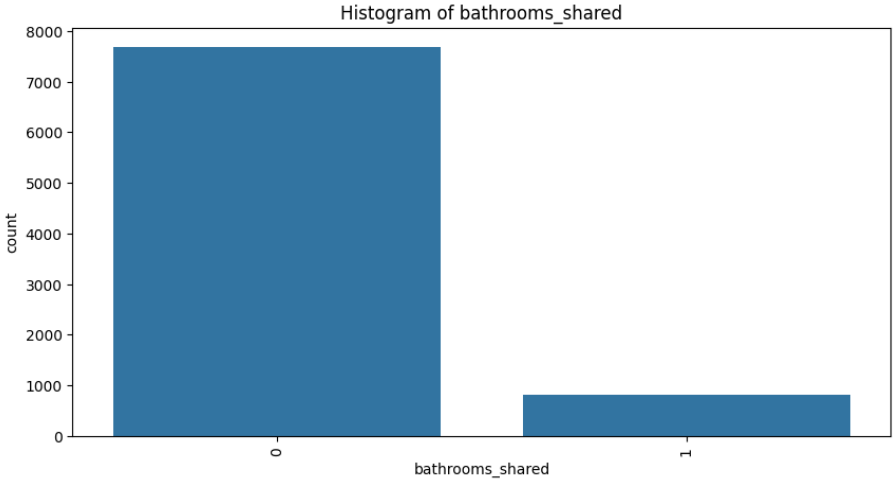
Verification – How is the host verified. Boxplot showing the effect on price:



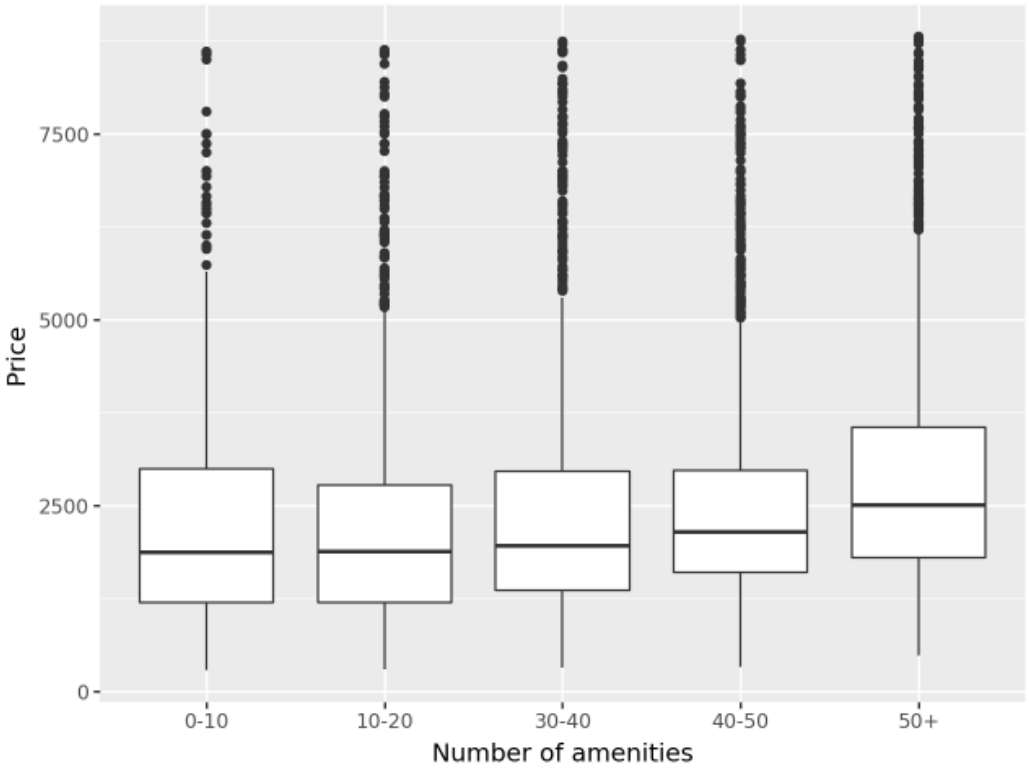
Host_since – For how long the host has been a host.



Bathrooms_shared – a binary column indicating whether the bathroom is shared (1) or not (0). It was extracted from bathrooms_text based on the presence of keyword “shared” in the value for each listing.

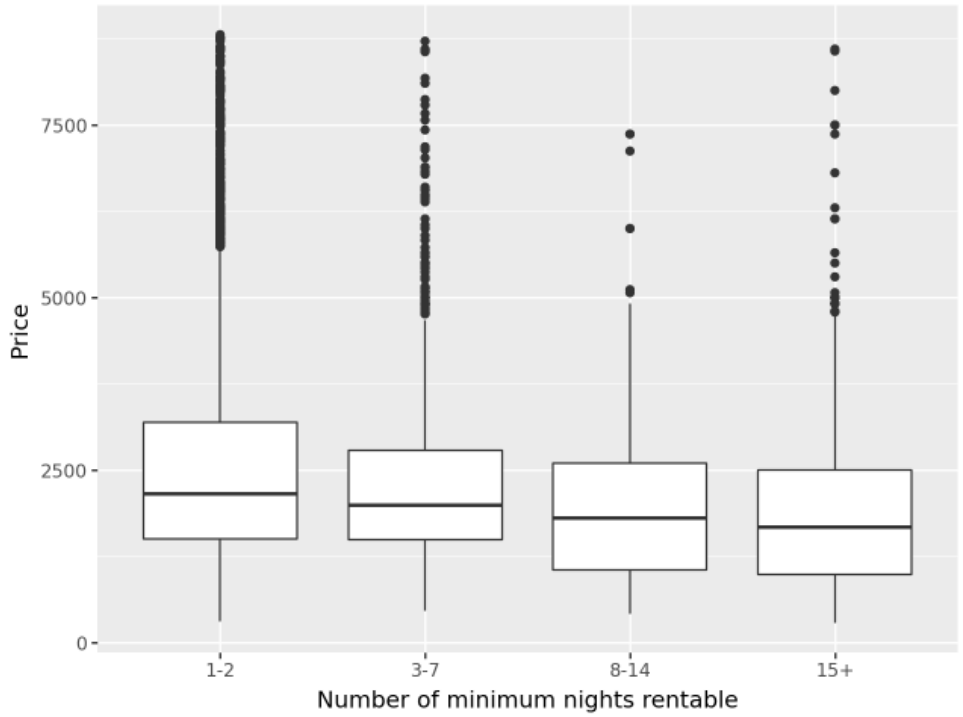


The amenities column included lists of all the self-selected amenities the listing includes. There are 1933 unique amenities, the names are not unified, therefore there are very unique values such as 'Fast wifi – 158 Mbps' which is difficult to work with. We still wanted to use this column somehow, so we created “amenities_num” which indicates how many unique amenities are listed for each property. The number of amenities seems to affect the price, so this column will be useful in the model.



The most promising columns for the model appear to be:

- accommodates (min 1, median 4, max 16, mean 3.9) - the higher number of people who can rent the property, the higher the price.
- beds - min 1, median 2, max 50, mean 2.61
- room_type (Entire home/apt 7185, private room 1477, hotel room 161, shared room 126) - more privacy and exclusivity mean higher prices, with hotel rooms being a special category which is significantly more expensive on average.
- Minimum_nights – higher number of minimum nights the customers can rent the property for means the property is meant for longer stays and is therefore cheaper. Maximum_nights doesn't seem to affect the price, as most listings have very high maximum rentable nights.

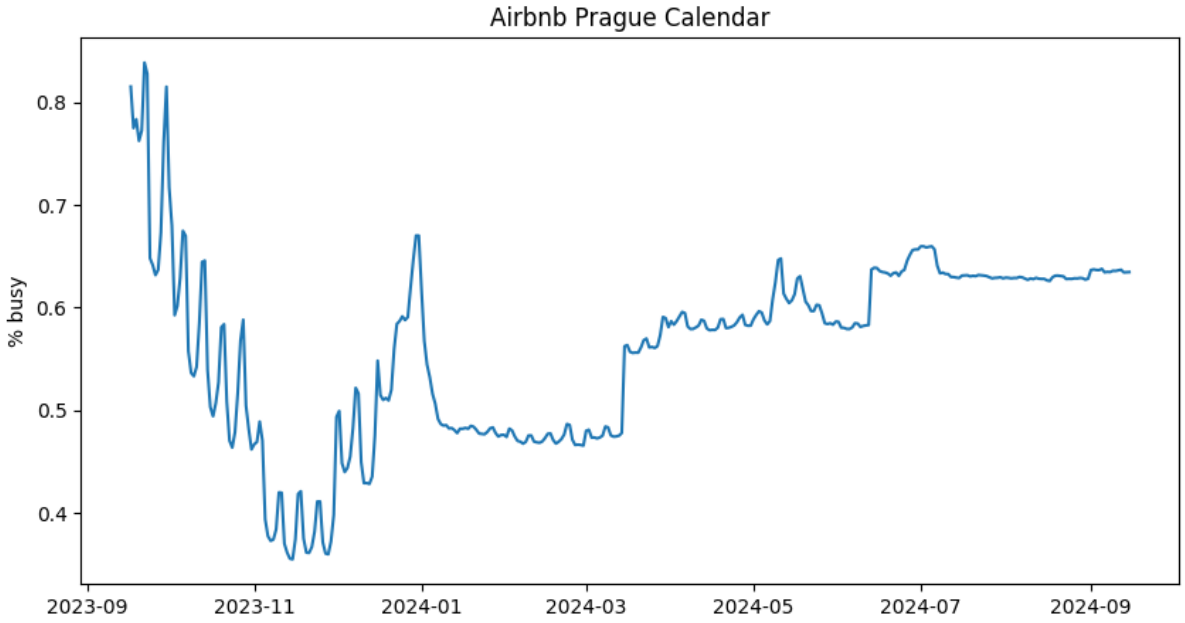


- neighbourhood - most listings are in Praha 1 and 2, and generally near the center, where they tend to be more expensive
- bathrooms_shared - when bathrooms are shared, it significantly affects the price
- room_type - category significantly affects price
- amenities_num - the more amenities, the pricier
- Verification
- Seasonality related columns
- Distance of host

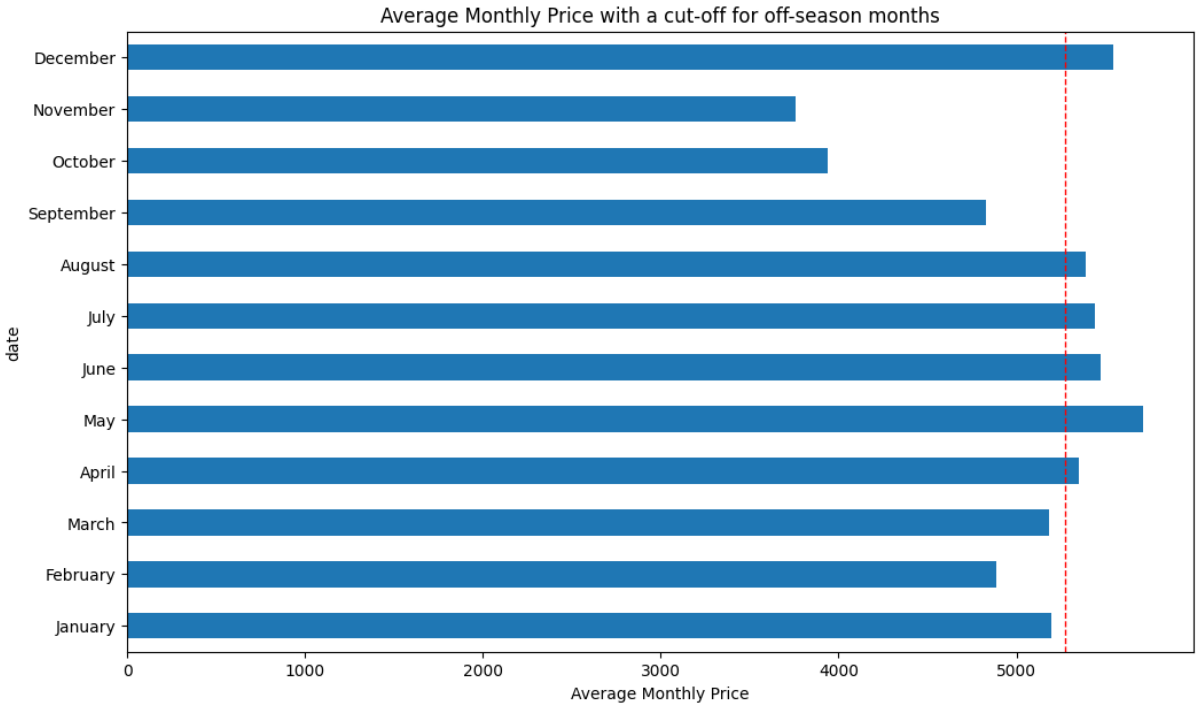
- Response time of host and last reviews – the more active a host is, the better rating and higher price

3 Data Visualization

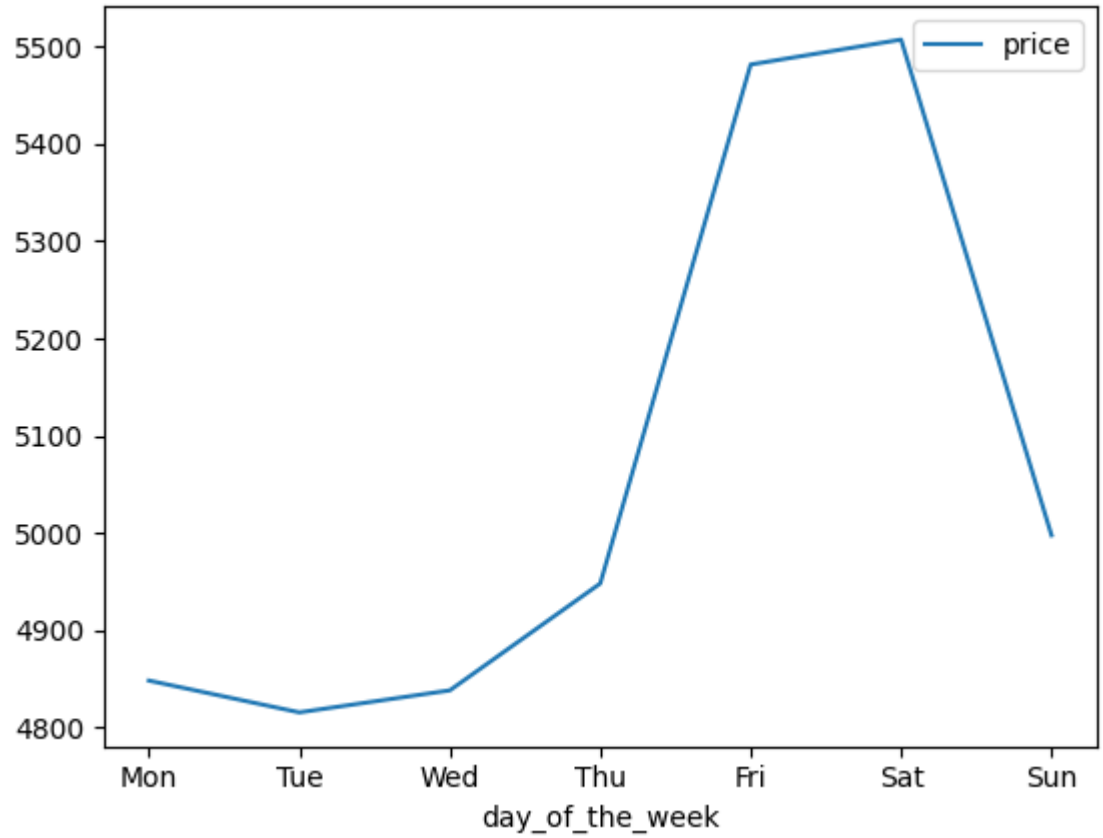
Here we see that there is an apparent trend of booking last minute, which means that indicators further into the future are underestimated. Two spikes – New Years Eve and hockey championship in May are clearly defined, and summer holiday season is also indicated.



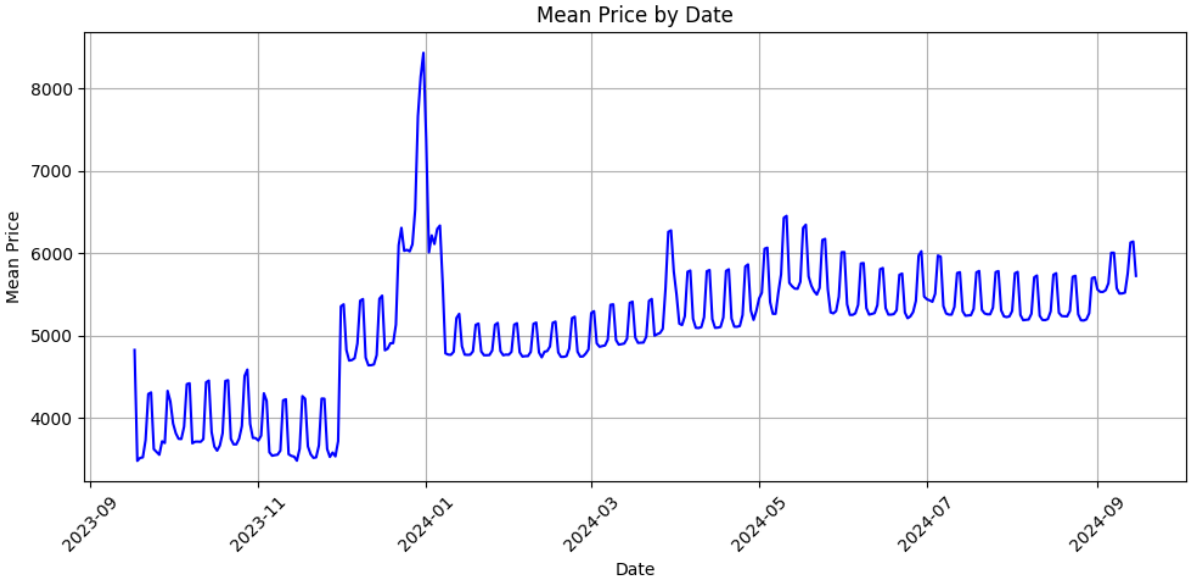
Looking deeper into seasonality, we discover that there is a slight outline of average price difference between in-season and off-season months, typically defined by winter and summer with the expected exception of December.



Seasonality is also clearly present in every week, where Friday and Saturday logically have higher average prices.



Mean price by date graph shows both of the aforementioned seasonal trend combined.



All seasonality trends are further discussed in Bonus Tasks at the end of the report.

4 Modelling

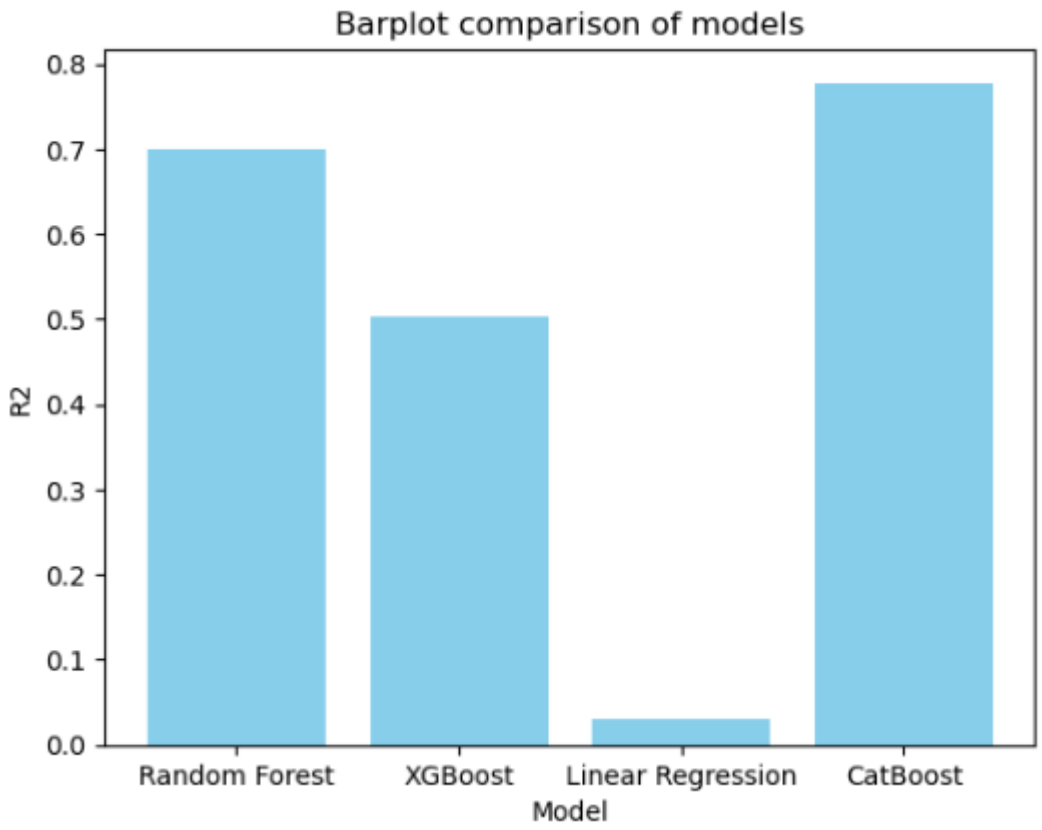
Chosen models

Linear Regression – Its performance was overall poor.

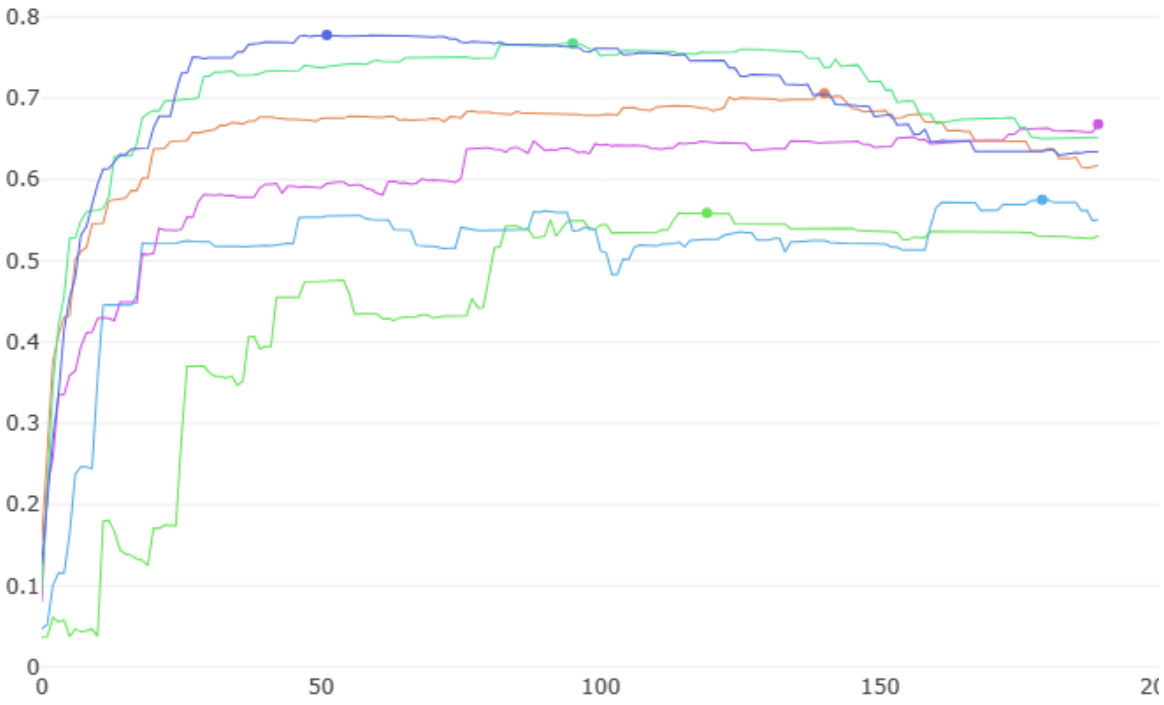
XGBoost was overfit – It has R2 of 50%, but it was most likely because of the lack of hyperparameter tuning.

Random forest – Was second best model according to R2.

Catboost – This model is one of the most powerful boosting algorithms but also one of the slowest. It is more immune to overfitting and does not require one hot encoding, as the model chooses the best encoder by itself. The most powerful tool is the ability to use RFE (Recursive Feature Eliminator), which allowed us to choose the most influential features.



The following graph shows the learning curve of CatBoost models.



Feature selection

We've used loss function to determine which features were the most significant to the model. This shows that the model performed better as more features were deleted.

Model limitations and considerations

The main issue is the teaching period. The computational complexity of the model is really noticeable, especially as we add more features.

The interpretability. CatBoost, like other ensemble methods, can be seen as a black box model, meaning it might be challenging to interpret how exactly it makes predictions.

Hyperparameters. Fine-tuning hyperparameters for optimal performance can be tricky and computationally expensive.

What to improve

We could further optimize the model's hyperparameters to improve model performance with methods such as grid search. Techniques such as oversampling and undersampling could help handle imbalanced data. More data and more diverse data would also improve prediction.

Hyperparameters

As an evaluation metric we've used R2 (because of the need for interpretability). We've also used learning rate of 0.282558. And lastly, we've used the standard way for a catboost to handle categorical variables (target encoding), out of which we've kept 11 (optimal number).

Validation Strategy

We utilized walk-forward validation to assess the performance of our predictive model for. We sequentially split our dataset into training and testing sets, updating the model with new data points at each step and evaluating its performance iteratively. This approach ensured that our model was tested on unseen data.

5 Model Interpretation

Interactions between the most influential features: In the final model out of 24 selected features, only 6 were determined as influential and used.

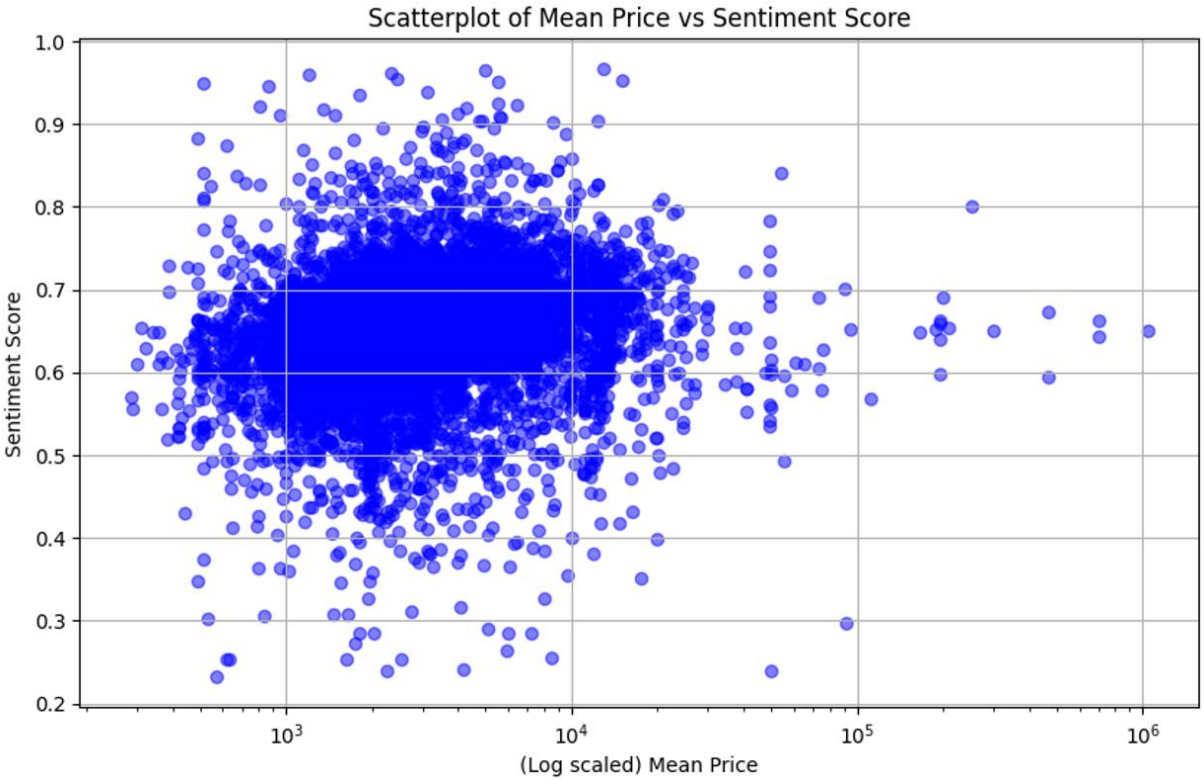
'maximum_nights', - The more nights you can book, the cheaper it is. 'amenities_num', - The more amenities AirBnB has the more expensive it is. 'Distance', - We can see, that the hosts who live nearby tend to have more expensive AirBnBs. 'Verification', - Hosts who are fully verified (both profile pic and their identity) tend to have cheaper AirBnBs. 'last_review', - AirBnBs with newer reviews tend to be cheaper.

'has_availability' - If it's listed as available, it is cheaper.

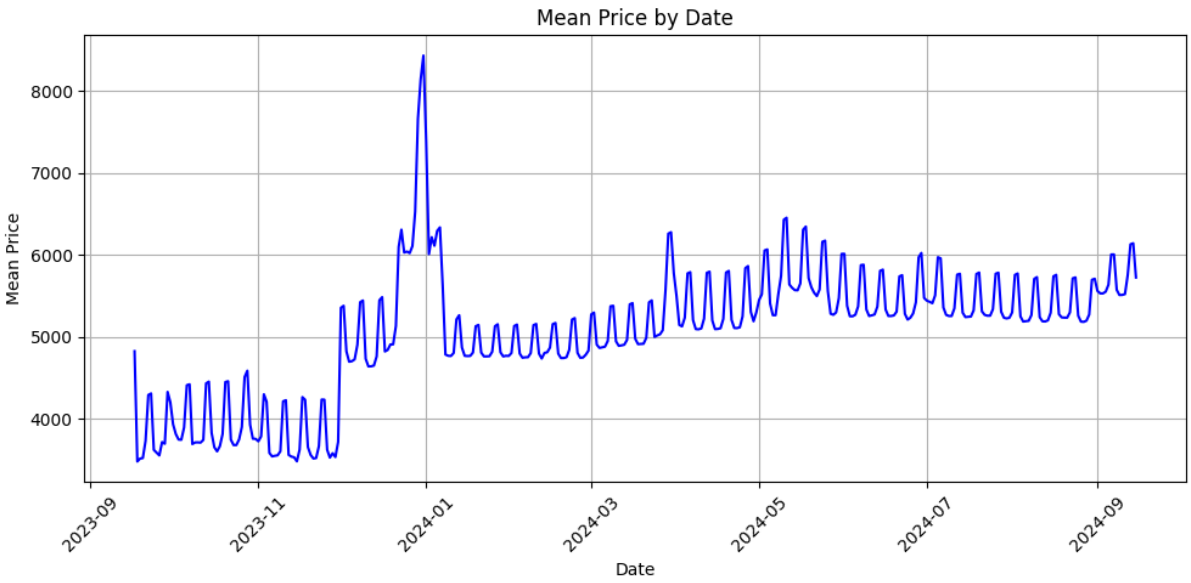
Bonus tasks

○ **Analyse the relation between the sentiment and price. Were people who paid more also more satisfied?**

At first, there is no evident trend of sentiment score and mean price, so those who paid more do not on average leave significantly better comments. Later on, when choosing features for the final model, sentiment score was a strong contender, so there is some prediction power, but it did not make the final cut.



○ What high seasons did you identify? How do the seasons differ for different locations and estate types?



There is a distinction when it comes to in-season (april,may,june,july,aug,dec) and off-season months (jan, feb, mar, sep, oct, nov), with a distinct price peak in may most likely due to hockey championship taking place.

Overall rising trend of inflation is also apparent.

There is also a clear distinction between weekday and weekend pricing, with Fri and Sat having much higher prices, and Sun (and perhaps even Thu) slightly higher prices than workdays.

The basic trend of dividing between in-season/off-season months and weekend/workdays remains more or less consistent across all neighbourhoods and room types, with inconsistencies occuring mostly due to lower numbers of observations in peripheral locations in Prague.

The weekend pricing does not really apply for Sundays in Hotels and Private rooms, which makes sense when we consider tourists mostly leaving on sundays.

Hotels utilize monthly seasonal price adjustments the most, whereas shared rooms nearly ignore them altogether.

Streamlit app