Group 2 - DBM301 - Assignment

AirBnB Market Analysing and Pricing

Team members:
SE150579 - Đinh Hoàng Lâm (Leader)
SE150440 - Trần Duy Ngọc Bảo
SE150380 - Đặng Chí Thanh
SE150454 - Nguyễn Cao Trí

I. Introduction

AirBnB is one of the most popular home rental apps today, they have provided many travellers a great, easy and convenient place to stay during their travels. Airbnb optimises the interests of both the lessor and the lessee by listing their properties for residents to stay. They help hosts know which properties they should invest in if they want to list their home on the app or choose competitive pricing. And help travellers can search by keywords that suit their prices, such as "free parking, balcony"...

The reason we chose this topic is because it is suitable for requirements such as the dataset is not clean, suitable for analysis based on data to predict the price by machine learning…

II. Implementation:

1. Data Preparation and Analysing:

a) Seattle AirBnB Open Data (Kaggle):

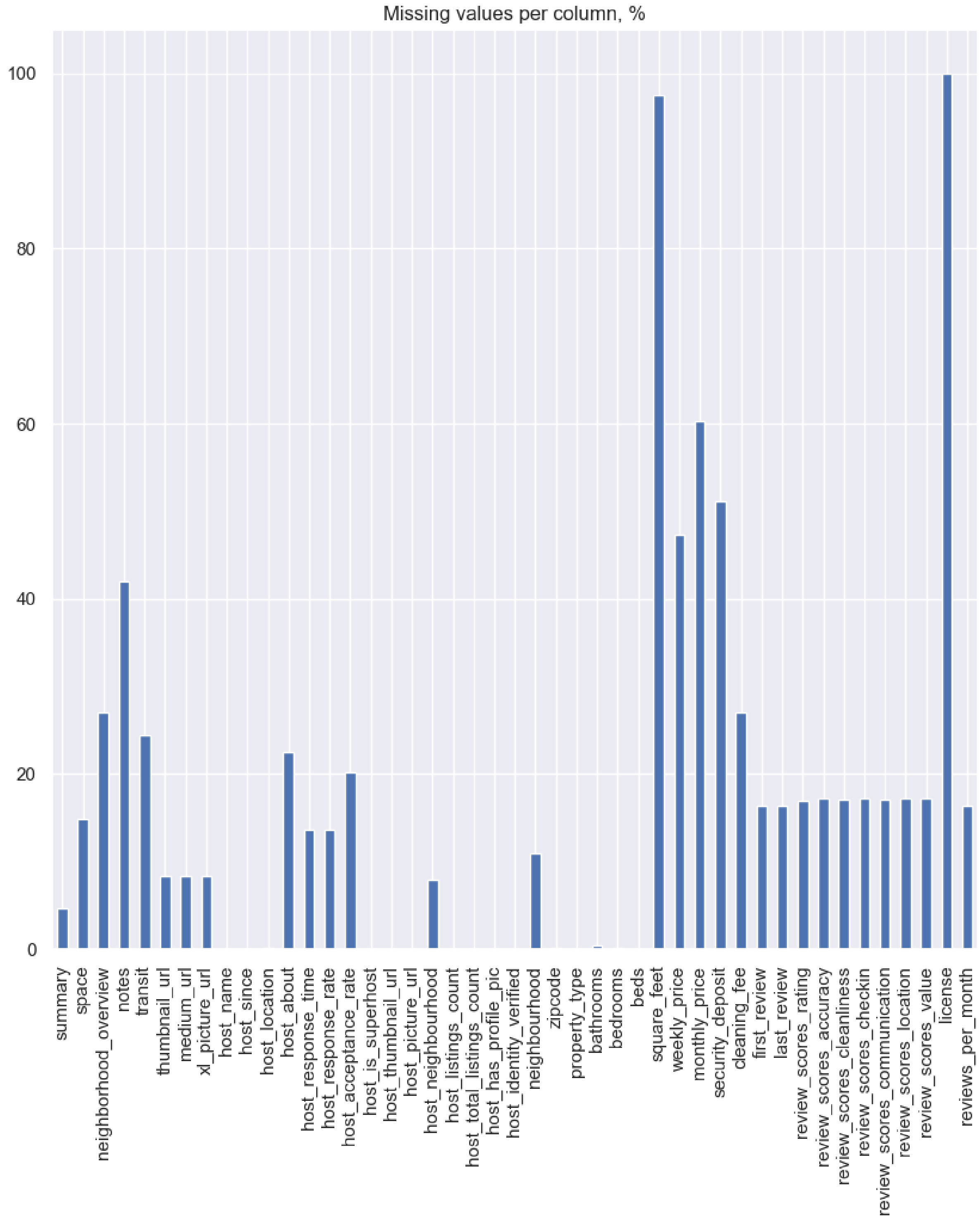
The dataset is divided into 3 smaller datasets:

- Listing: descriptions and average review score.
- Reviews: details in comments of each unique reviewer.
- Calendar: available date for property in listing.

⇒ Because of the limitation of level in data mining, we will choose Listing as our main dataset for analysing and predicting. Besides, there are some reasons for us not choosing 2 remaining datasets: the Reviews dataset needs a deep understanding in emotion and positive/negative investigation, and the Calendar dataset has a period that is too old to be calculated.

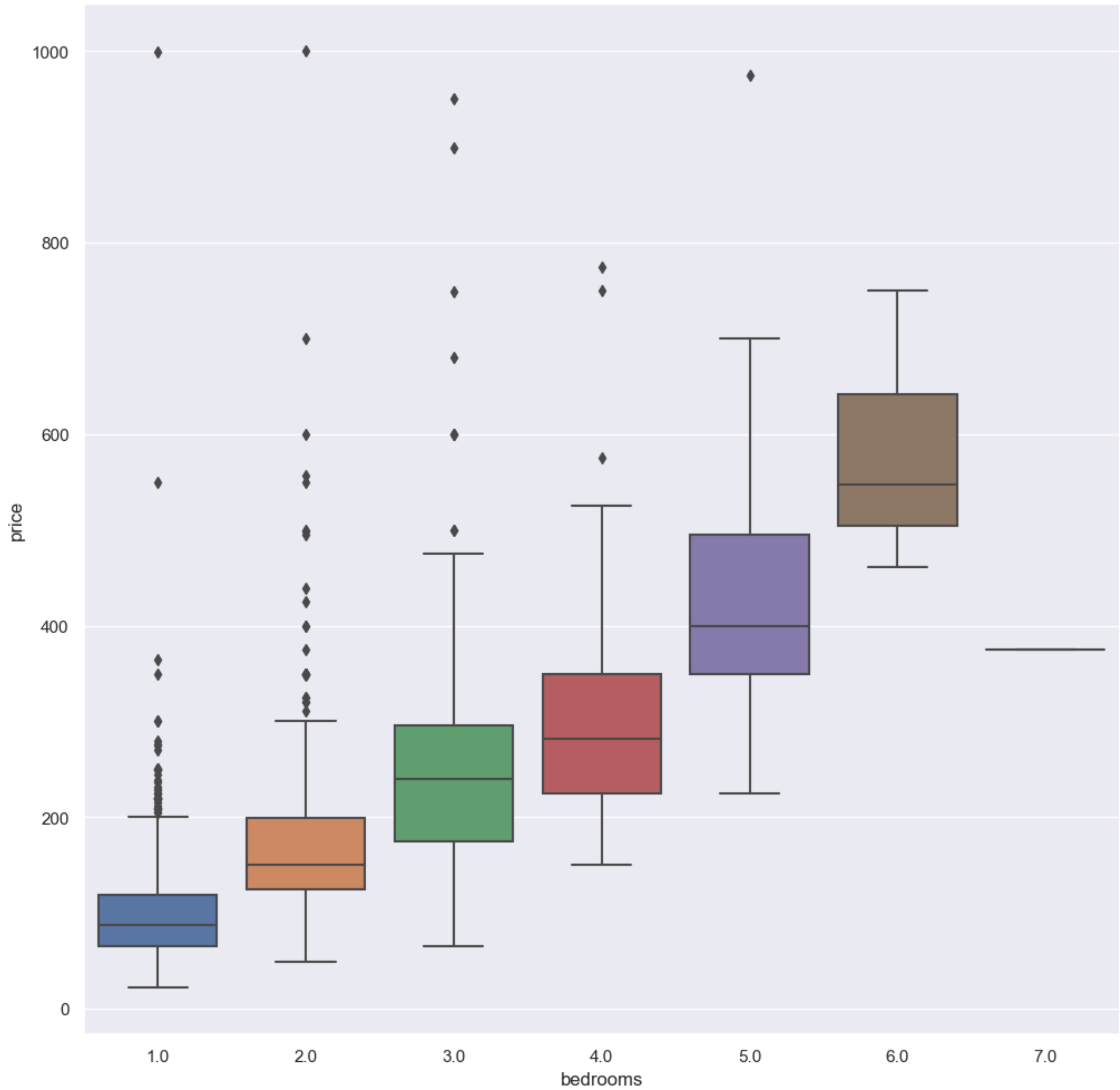
Overview of Listing dataset:

- Dimension: $3818\ values\ \times 92\ variables$.
- Missing values per column are recorded as:



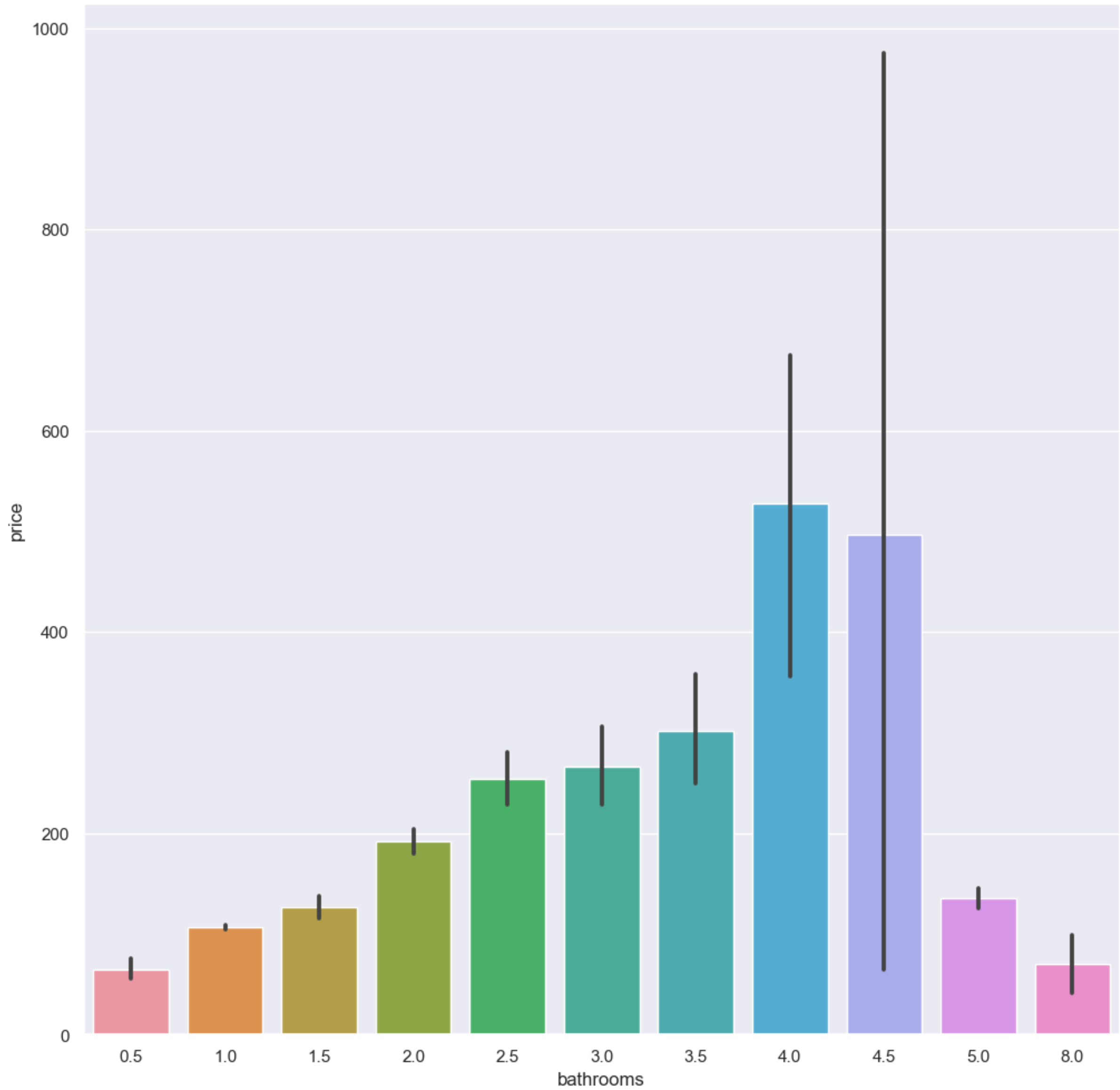
[Figure1] Missing values per column

- Price is proportional to the number of bedrooms.



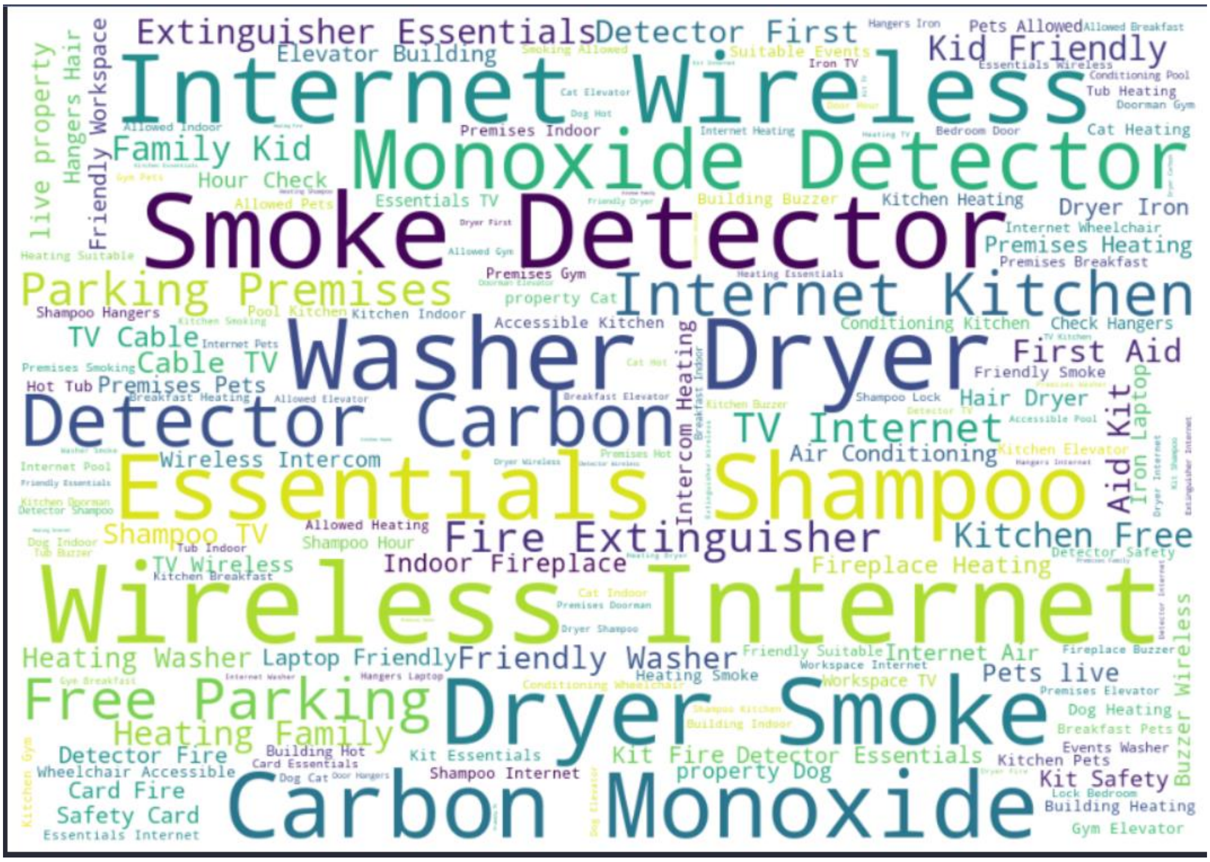
[Figure2] Price based on bedrooms

- The number of bathrooms that are most concentrated between 2 and 4.5 indicates an increase of price with the increase of bathrooms.



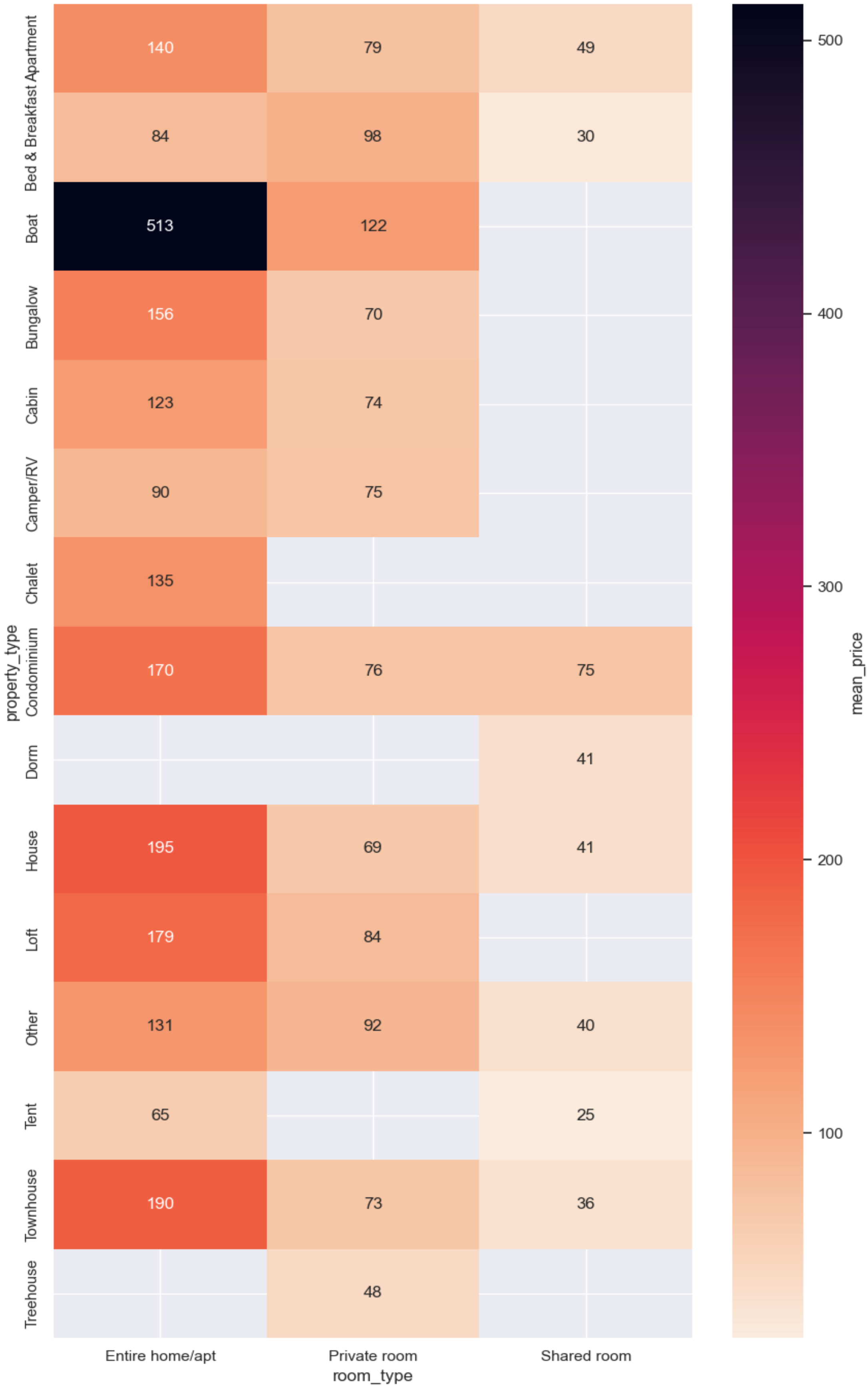
[Figure3] Price based on bathrooms

- Furniture and materials also affect the rental price quite a lot.



[Figure4] Amenities

- We can see that shared rooms have the lightest colour hence cheapest. Private rooms have a slightly darker colour so they are in the middle, and entire houses are the darkest thus the most expensive. Noting that the highest number of listings which was house and apartments actually have very similar prices for each of the room_type category. Therefore, room_type and property_type play an important role in affecting the price in listing dataset.



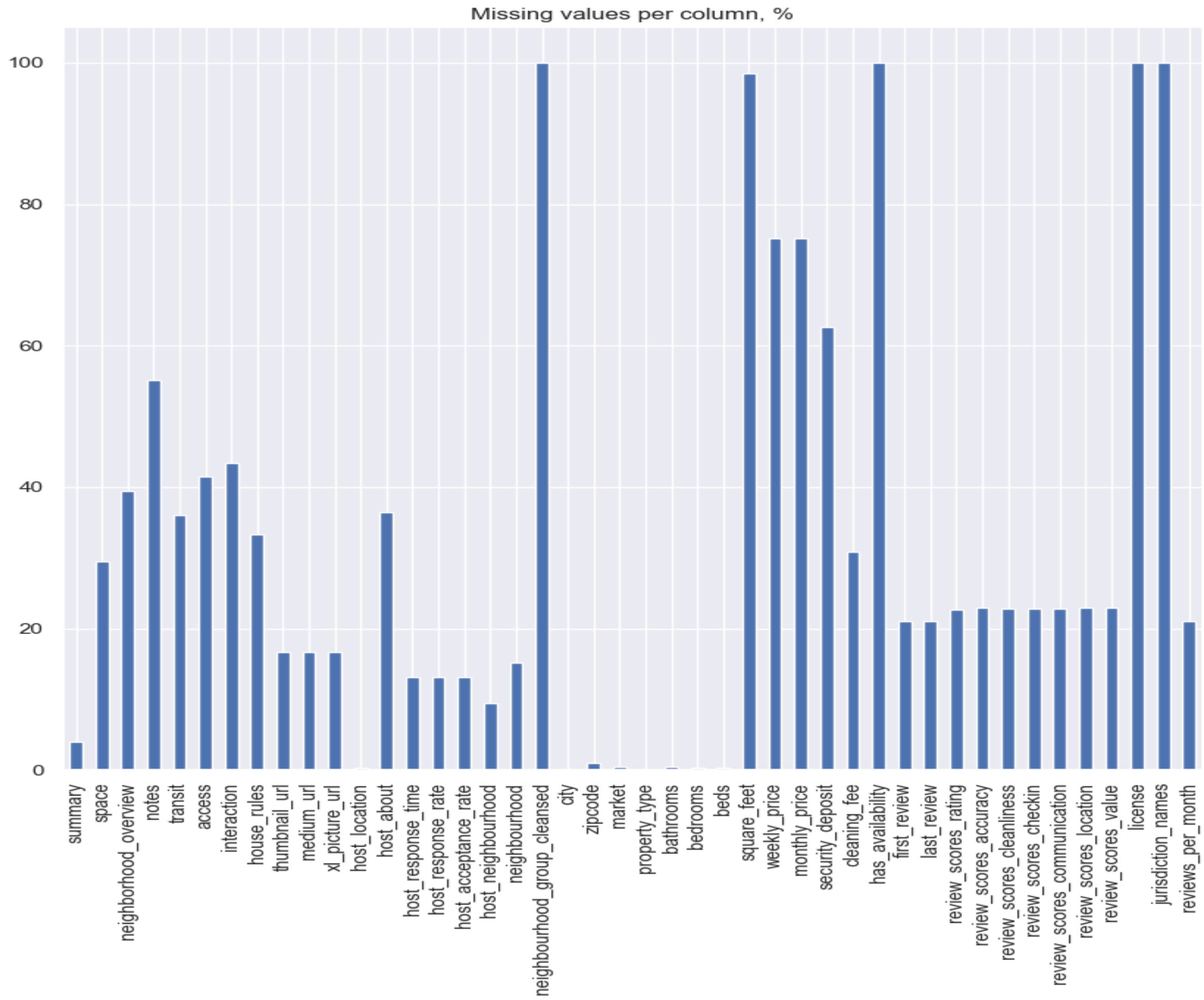
[Figure5] Property and room type

b) Boston AirBnB Open Data (Kaggle):

In order to experiment objectively, we decided to choose another data set with a similar structure to the city of Seattle. Fortunately, we consider that the Boston dataset satisfied our requirements.

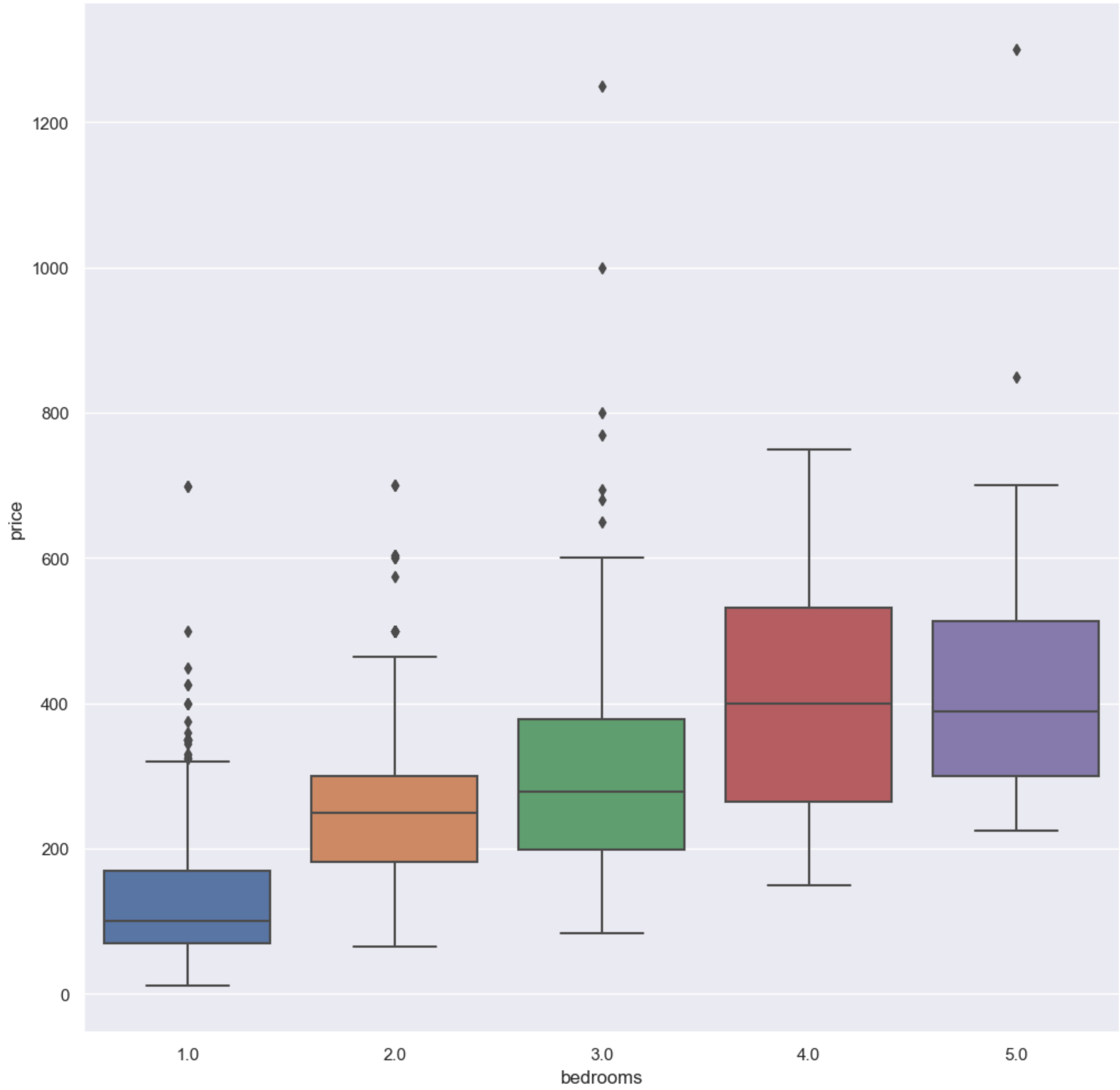
So, we can overview some characteristics of listing.csv from the Boston dataset:

- Dimension: $3585\ values\ \times 95\ variables$. (More variables than Seattle, but we focus only on the most influential common characteristics (28 attributes) of the two datasets.)
- Missing values per column:



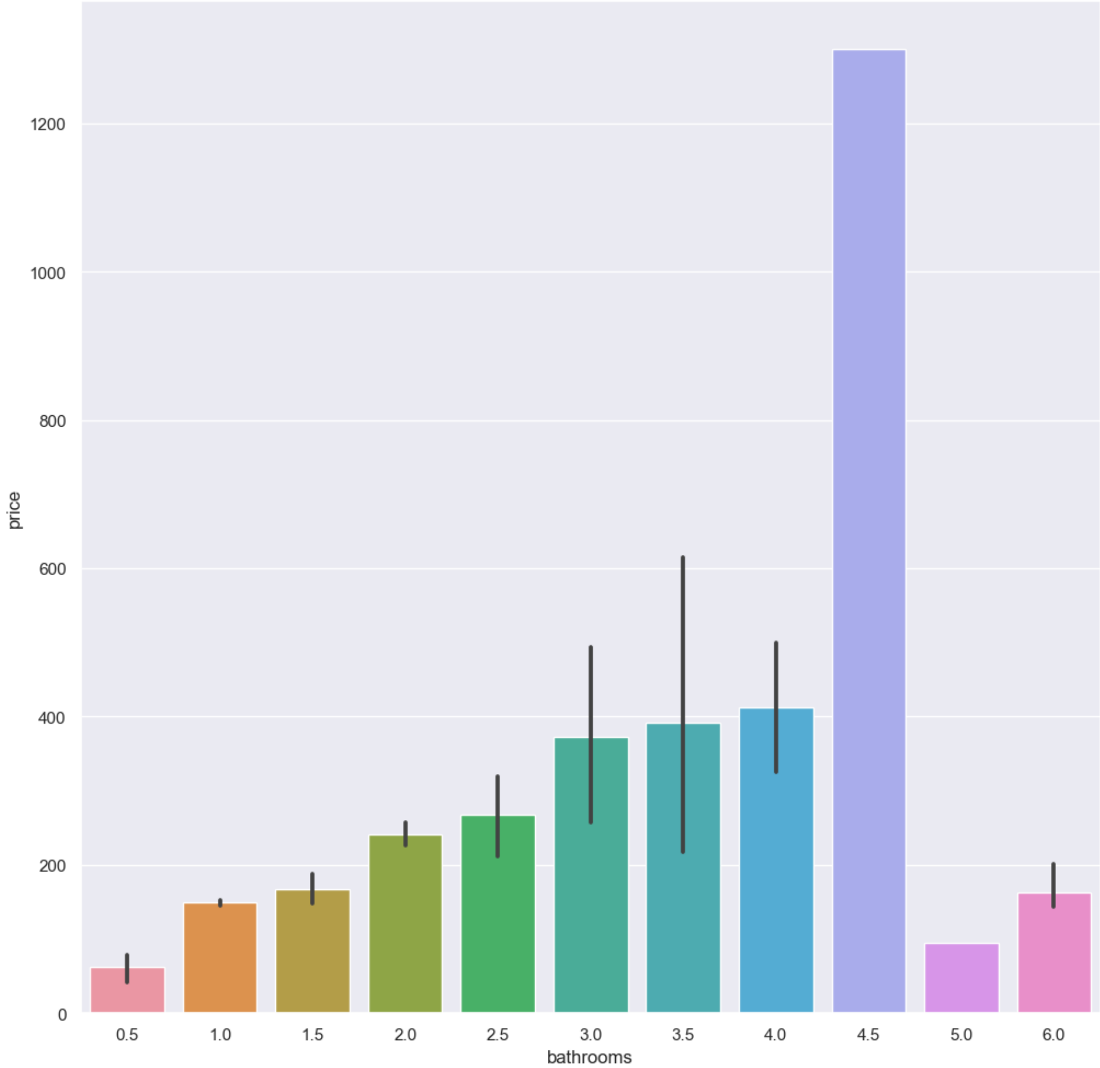
[Figure6] Missing values per column

- Price based on the number of bedrooms:



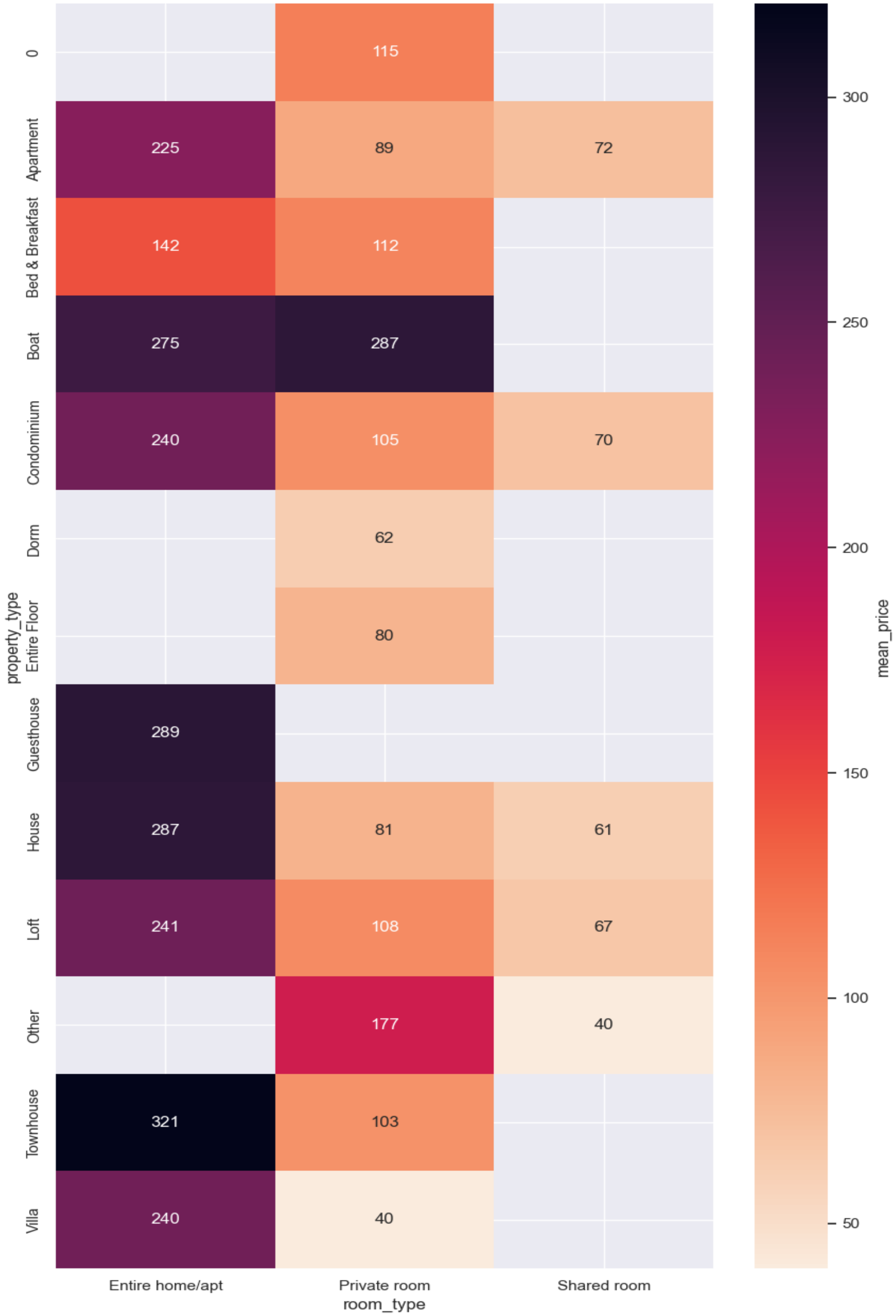
[Figure7] Price based on bedrooms

- Price based on the number of bathrooms:



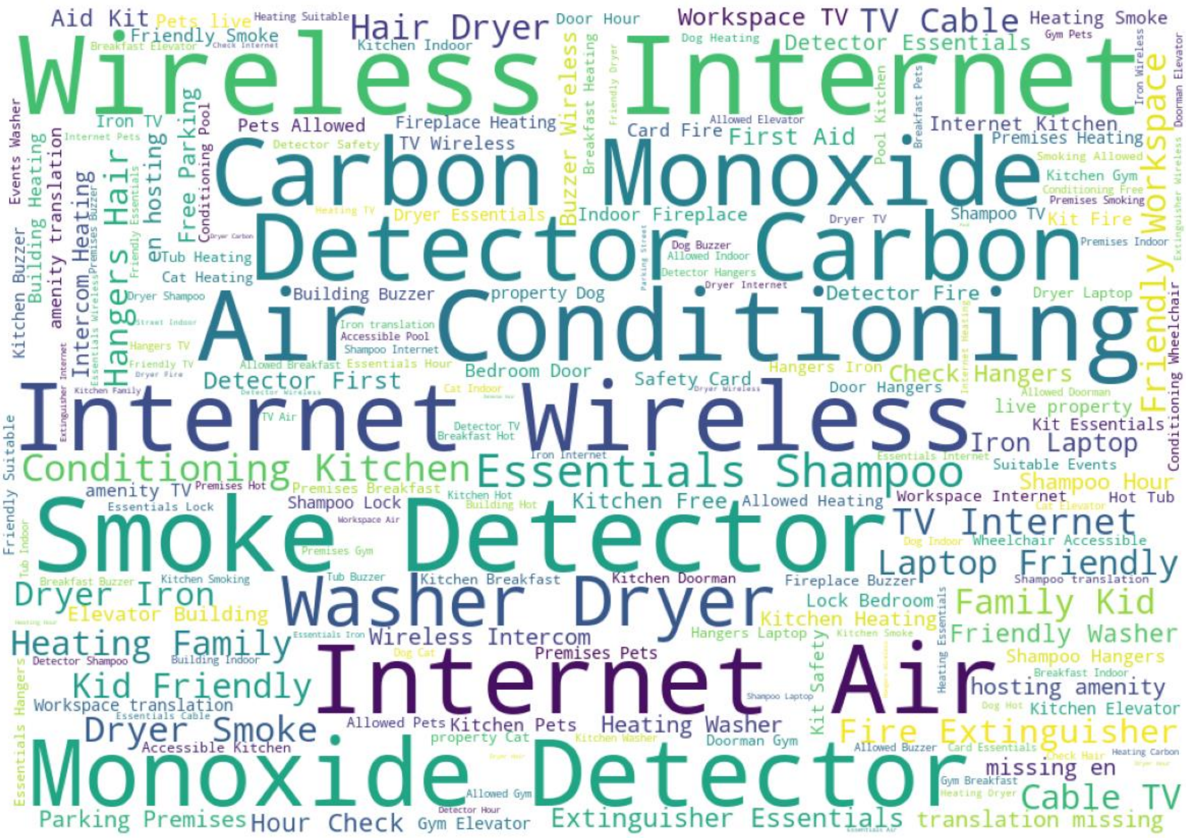
[Figure8] Price based on bathrooms

- Prices for different room and property types:



[Figure7] Property and room type

- Amenities:



2. Choose model (Linear regression, Random Forest):

   In this article, we use 2 algorithms:

   - **Linear Regression** is a machine learning algorithm that is based on supervised learning. It performs the regression task to predict a dependent variable value (in this case, price) based on given independent variables (in this case, the identified predictor variables). It then tries to find a linear relationship between the variables andredicts the price based on the linear line.
   - **Random Forest** is an ensemble technique that is able to perform both Regression and Classification tasks with the use of multiple decision trees and a technique that is called Bootstrap Aggression. The idea behind this technique is to combine multiple decision trees in its prediction rather than relying on individual decision trees.
     Here, we use the RandomForestRegressor to help predict the price.

3. Input for Training and Testing model:

By analysing the features of common variables in the Listing dataset. We choose the input for training and testing that satisfies these conditions:

- The number of values is large enough to include in the model (no or very few Null values).
- The characteristics can meet the needs of users who want to know.
- All of them have a big influence on the price during the above analysis (To make it have enough information for providing to models).

Our input variables can be described as some kind of questions below:

- How many bedrooms do you need?
- How many bathrooms do you need?
- Do you want to rent an apartment with television?
- Do you want to have elevators in your apartment?
- Do you want to rent an apartment with a gym?
- Do you want some saunas or pools in your apartment?
- Do you want your apartment to have the internet?
- Will the apartment allow people to come with their pets?
- What type of room do you want to live in? (3 types: Shared room, Private room or Entire home apartment).
- What kind of property do you want to hire? (6 types: Condominium, House, Loft, Townhouse, Bed and Breakfast or Other).

4. Processing data to clean it to a suitable for training

Amenities

- First we displays the different types of amenities available for each listing, separating the different amenities and creating a dedicated column for each amenity.

```
each of the different amenities and adding them into the original dataframe
f['amenities'].str.contains('24-hour check-in'), 'check_in_24h'] = 1
f['amenities'].str.contains('Air conditioning|Central air conditioning'), 'air_conditioning'] = 1
f['amenities'].str.contains('Amazon Echo|Apple TV|Game console|Netflix|Projector and screen|Smart TV'), 'high_end_electronics'] = 1
f['amenities'].str.contains('BBQ grill|Fire pit|Propane barbeque'), 'bbq'] = 1
f['amenities'].str.contains('Balcony|Patio'), 'balcony'] = 1
f['amenities'].str.contains('Beach view|Beachfront|Lake access|Mountain view|Ski-in/Ski-out|Waterfront'), 'nature_and_views'] = 1
f['amenities'].str.contains('Bed linens'), 'bed_linen'] = 1
f['amenities'].str.contains('Breakfast'), 'breakfast'] = 1
f['amenities'].str.contains('TV'), 'tv'] = 1
f['amenities'].str.contains('Coffee maker|Espresso machine'), 'coffee_machine'] = 1
f['amenities'].str.contains('Cooking basics'), 'cooking_basics'] = 1
f['amenities'].str.contains('Dishwasher|Dryer|Washer'), 'white_goods'] = 1
f['amenities'].str.contains('Elevator'), 'elevator'] = 1
f['amenities'].str.contains('Exercise equipment|Gym|gym'), 'gym'] = 1
f['amenities'].str.contains('Family/kid friendly|Children|children'), 'child_friendly'] = 1
f['amenities'].str.contains('parking'), 'parking'] = 1
f['amenities'].str.contains('Garden|Outdoor|Sun loungers|Terrace'), 'outdoor_space'] = 1
f['amenities'].str.contains('Host greets you'), 'host_greeting'] = 1
f['amenities'].str.contains('Hot tub|Jetted tub|hot tub|Sauna|Pool|pool'), 'hot_tub_sauna_or_pool'] = 1
f['amenities'].str.contains('Internet|Pocket wifi|Wifi'), 'internet'] = 1
f['amenities'].str.contains('Long term stays allowed'), 'long_term_stays'] = 1
f['amenities'].str.contains('Pets|pet|Cat(s)|Dog(s)'), 'pets_allowed'] = 1
f['amenities'].str.contains('Private entrance'), 'private_entrance'] = 1
f['amenities'].str.contains('Safe|Security system'), 'secure'] = 1
f['amenities'].str.contains('Self check-in'), 'self_check_in'] = 1
f['amenities'].str.contains('Smoking allowed'), 'smoking_allowed'] = 1
f['amenities'].str.contains('Step-free access|Wheelchair|Accessible'), 'accessible'] = 1
f['amenities'].str.contains('Suitable for events'), 'event_suitable'] = 1
```

[Figure8] Grouping amenities column

We have columns as "Beach view | Beachfront | Lake access | Mountain view | Ski-in/Ski-out | Waterfront |" into one column as 'nature_and_views'

- Removing amenities which have NULL values for all listings:

```
Data columns (total 16 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   room_type             3818 non-null   object
 1   property_type         3817 non-null   object
 2   bedrooms              3812 non-null   float64
 3   bathrooms             3802 non-null   float64
 4   number_of_reviews     3818 non-null   int64
 5   price                 3818 non-null   object
 6   breakfast             291 non-null    float64
 7   tv                    2741 non-null   float64
 8   white_goods           3134 non-null   float64
 9   elevator              785 non-null    float64
 10  gym                   442 non-null    float64
 11  hot_tub_sauna_or_pool  159 non-null   float64
 12  internet              3692 non-null   float64
 13  pets_allowed          1169 non-null   float64
 14  secure                727 non-null    float64
 15  accessible            300 non-null    float64
dtypes: float64(12), int64(1), object(3)
```

[Figure9] Remaining non-nul variables

Property type

- Grouping property types whose low counts might be insignificant and not provide us with enough information
- Thus, grouping property types that have counts that are < 30 into ‘other’ column

```
House            1733
Apartment        1708
Townhouse         118
Condominium        91
Loft               40
Bed & Breakfast    37
Other              22
Cabin              21
Camper/RV          13
Bungalow           13
Boat                8
Tent                5
Treehouse           3
Dorm                2
Chalet              2
Yurt                1
Name: property_type, dtype: int64
```

```
House            1733
Apartment        1708
Townhouse         118
Other              91
Condominium        91
Loft               40
Bed & Breakfast    37
Name: property_type, dtype: int64
```

[Figure10] Original property [Figure11] After grouping

Price

	room_type	property_type	bedrooms	bathrooms	number_of_reviews	price	breakfast	tv	white_goods	elevator	gym	hot_tub_sauna_or_pool	internet
0	Entire home/apt	Apartment	1.0	1.0	207	$85.00	NaN	1.0	1.0	NaN	NaN	NaN	1.0
1	Entire home/apt	Apartment	1.0	1.0	43	$150.00	NaN	1.0	1.0	NaN	NaN	NaN	1.0
2	Entire home/apt	House	5.0	4.5	20	$975.00	NaN	1.0	1.0	NaN	NaN	NaN	1.0
3	Entire home/apt	Apartment	0.0	1.0	0	$100.00	NaN	NaN	1.0	NaN	NaN	NaN	1.0
4	Entire home/apt	House	3.0	2.0	38	$450.00	NaN	1.0	NaN	NaN	NaN	NaN	1.0

[Figure12] Original Seattle dataset

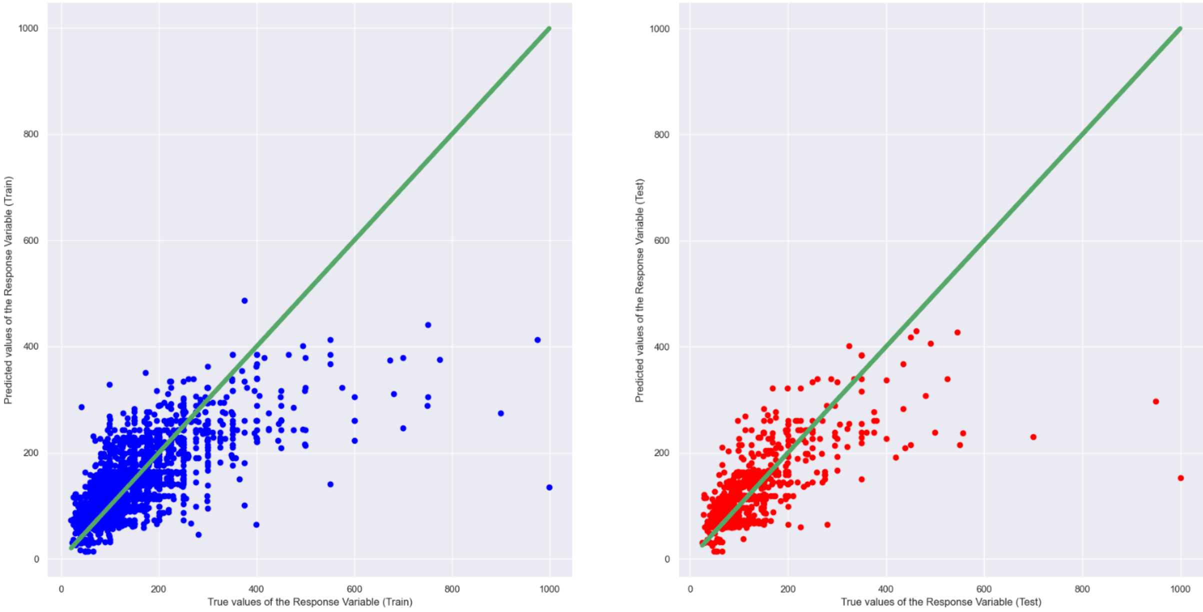
- Since the Price variable is currently a string (with the "$" symbol), the variable is thus converted into an integer to suitable for the training process.
- We ensuring that there are no NULL entries in the data.

	room_type	property_type	bedrooms	bathrooms	number_of_reviews	price	breakfast	tv	white_goods	elevator	gym	hot_tub_sauna_or_pool	internet	pe
0	Entire home/apt	Apartment	1.0	1.0	207	85	0.0	1.0	1.0	0.0	0.0	0.0	1.0	
1	Entire home/apt	Apartment	1.0	1.0	43	150	0.0	1.0	1.0	0.0	0.0	0.0	1.0	
2	Entire home/apt	House	5.0	4.5	20	975	0.0	1.0	1.0	0.0	0.0	0.0	1.0	
3	Entire home/apt	Apartment	0.0	1.0	0	100	0.0	0.0	1.0	0.0	0.0	0.0	1.0	
4	Entire home/apt	House	3.0	2.0	38	450	0.0	1.0	0.0	0.0	0.0	0.0	1.0	

[Figure13] After cleaning dataset

5. Results & Evaluate

a. Seattle



[Figure14] Linear Regression

Points that lie on or near the diagonal line means that the values predicted by the CatBoost Regression model are highly accurate. If the points are away from the diagonal line, the points have been wrongly predicted.

Goodness Fit on the Models (Train/Test Split):

Test

	MSE	R^2
Linear Regression	**3376.5635**	**0.5194**
Random Forest Regression	**3182.4845**	**0.5470**

Train

	MSE	R^2
Linear Regression	**3936.6088**	**0.5326**
Random Forest Regression	**3387.1315**	**0.5978**

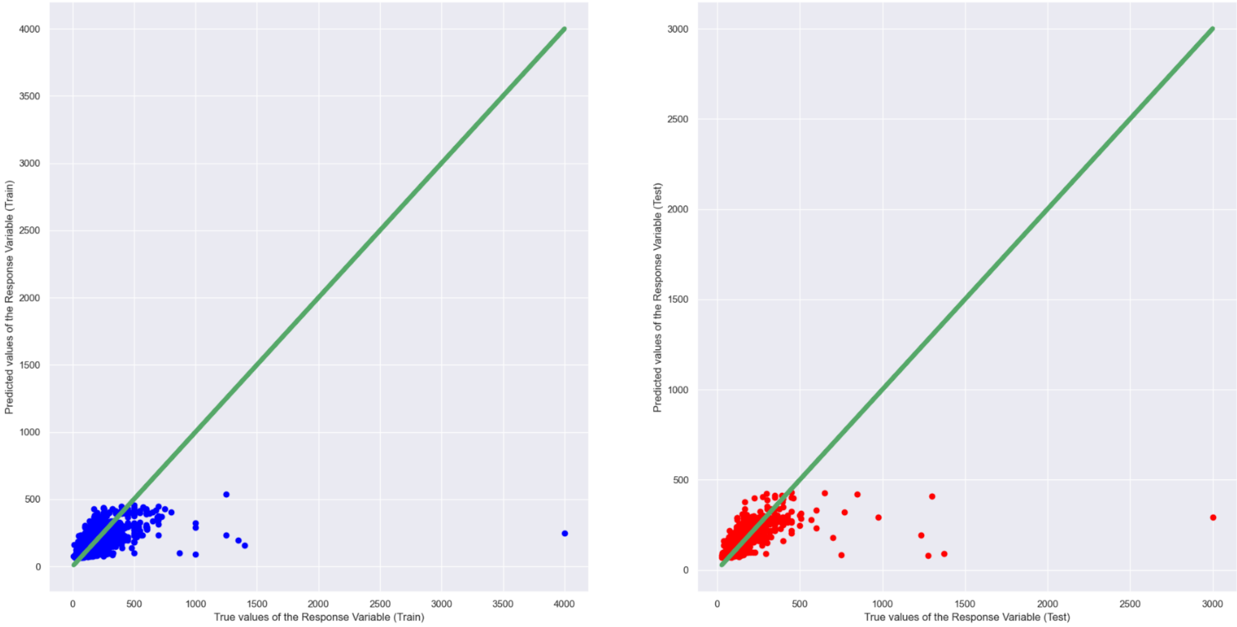
Note:

- MSE has a non-negative value, and a smaller value indicates that the prediction model is closer to the actual value. MSE is commonly used as an evaluation metric in regression and prediction tasks and is often utilised during model training and evaluation.
- R^2 Best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse). In the general case when the true y is non-constant, a constant model that always predicts the average y disregarding the input features would get a R^2 of score 0.0

=> The results show **the mean square error between the predicted values and the actual values in the large regression or prediction model**. This often indicates that the model **does not fit the data well and that there is a large discrepancy between the prediction and the actual value**. The cause of **influence is that the data distribution area is not uniform, so it affects the results**

=> When the value of **R^2** (coefficient of determination) reaches **~~ 0.5,** it means t**hat the regression model is able to explain about 50% of the variation of the dependent variable by the independent variables**. This shows that the model has a **moderate fit to the data and is reasonably predictive.**

b. Boston



[Figure15] Linear Regression

Points that lie on or near the diagonal line means that the values predicted by the CatBoost Regression model are highly accurate. If the points are away from the diagonal line, the points have been wrongly predicted.

Goodness Fit on the Models (Train/Test Split):

Test

	MSE	R^2
Linear Regression	**23602.8307**	**0.2438**
Random Forest Regression	**23703.0547**	**0.2406**

Train

	MSE	R^2
Linear Regression	**13388.4883**	**0.3197**
Random Forest Regression	**12416.0568**	**0.3692**

Note:

- MSE has a non-negative value, and a smaller value indicates that the prediction model is closer to the actual value. MSE is commonly used as an evaluation metric in regression and prediction tasks and is often utilised during model training and evaluation.
- R^2 Best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse). In the general case when the true y is non-constant, a constant model that always predicts the average y disregarding the input features would get a R^2 of score 0.0

=> The results show **the mean square error between the predicted values and the actual values in the large regression or prediction model**. This often indicates that the model **does not fit the data well and that there is a large discrepancy between the prediction and the actual value**. The cause of **influence is that the data distribution area is not uniform, so it affects the results**

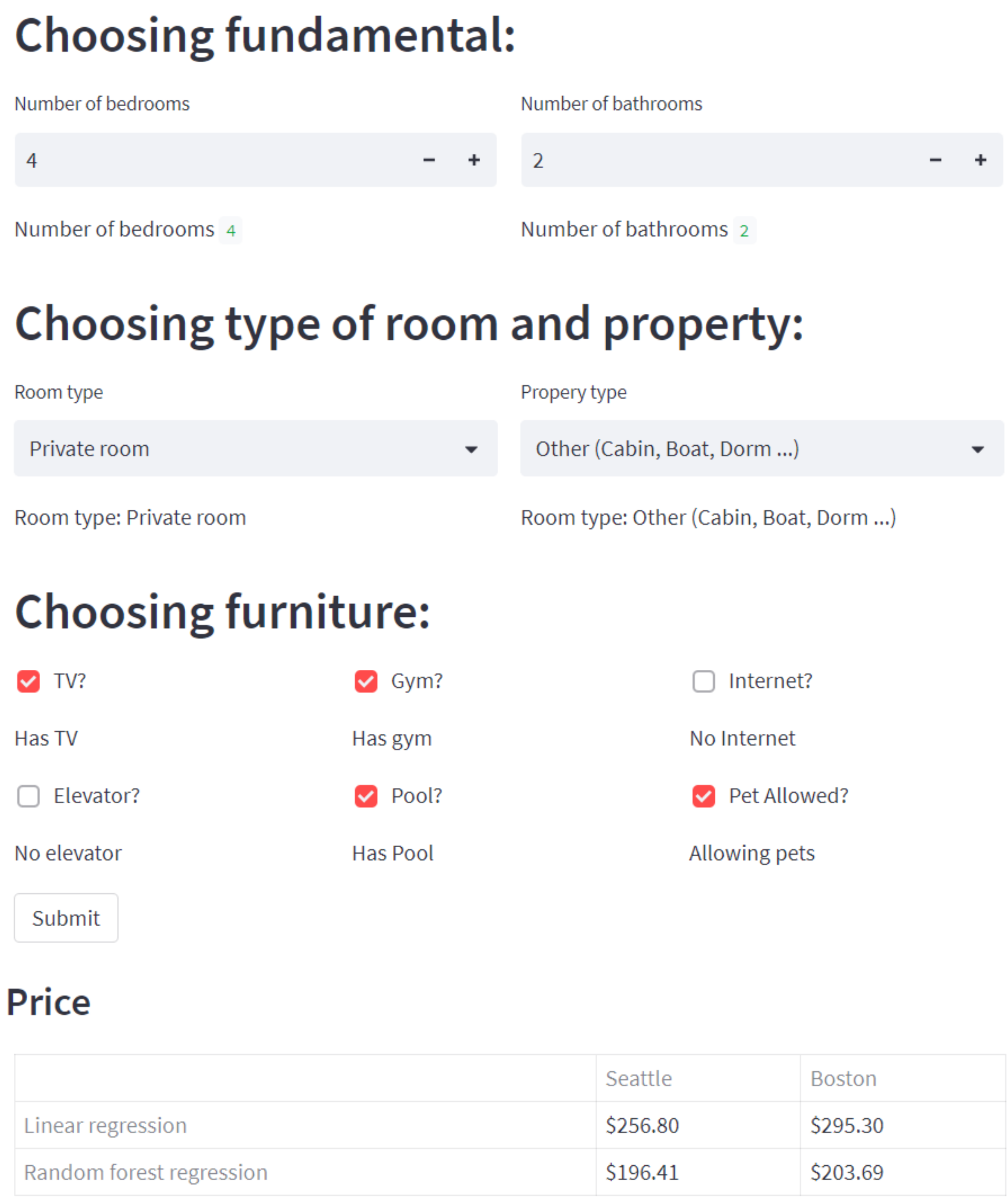
=> The values of **R^2 = 0.24** for the test data set and **R^2 = 0.32** for the training dataset show that the model has the ability to partially **explain the variation in the dependent variable**. However, t**he performance of the model is still limited and there is a disparity between the training data and the test data.**

=> **Overfitting**

III. Reference:

1. irfansh, mohamed. (2016). Airbnb Data Science Project. https://mohamedirfansh.github.io/Airbnb-Data-Science-Project/?fbclid=IwAR0hkKBqnYSecl75zMI3eePWltyo4yslXHX4zdaBKzqLsrDqFO3mtGAHmt4
2. www.kaggle.com. 2023. Seattle Airbnb Open Data | Kaggle. [ONLINE] Available at: https://www.kaggle.com/datasets/airbnb/seattle?resource=download&select=reviews.csv&fbclid=IwAR1JFV4kMRj89rSVkJvrXPhENDgnylFz53Vt59g0IfYjRn9N2w4NpcXlX8s. [Accessed 12 July 2023].
3. www.kaggle.com. 2023. *Boston Airbnb Open Data | Kaggle.* [ONLINE] Available at: https://www.kaggle.com/datasets/airbnb/boston?fbclid=IwAR1oApKkwRkVPQ6YyIIQh0VDkqFndYjVWT-qFiSMWb4fsnpBidU-dVuJv0M. [Accessed 12 July 2023].
4. scikit-learn.org. 2023. *sklearn.linear_model.LinearRegression — scikit-learn 1.3.0 documentation.* [ONLINE] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html. [Accessed 12 July 2023].
5. scikit-learn.org. 2023. *sklearn.ensemble.RandomForestRegressor — scikit-learn 1.3.0 documentation.* [ONLINE] Available at: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html. [Accessed 12 July 2023].
6. www.simplilearn.com. 2023. *Mean Squared Error : Overview, Examples, Concepts and More | Simplilearn.* [ONLINE] Available at: https://www.simplilearn.com/tutorials/statistics-tutorial/mean-squared-error. [Accessed 12 July 2023].
7. statisticsbyjim.com. 2017. *How To Interpret R-squared in Regression Analysis - Statistics By Jim.* [ONLINE] Available at: https://statisticsbyjim.com/regression/interpret-r-squared-regression/. [Accessed 12 July 2023].
8. www.theforage.com. 2023. *What Is a Data Analyst? - Forage.* [ONLINE] Available at: https://www.theforage.com/blog/careers/data-analyst. [Accessed 12 July 2023].
9. joshuakeating.com. 2023. [ONLINE] Available at: https://joshuakeating.com/res/pdfs/airbnb_paper.pdf. [Accessed 12 July 2023].

IV. Contributions & Demo



[Figure16] Demo

STT	Content	Member	Deadline	Status	Note
1	Data Preparation	Bảo	12/07/2023	Done	
2	What are the features /enities of a property that affects its price?	Lâm	28/06/2023	Done	
3	Are there particular locations in Seattle where AirBnb listings fetch higher prices?	Thanh	28/06/2023	Done	
4	Does textual data in the summary and sentiments of reviews affect price?	Trí	28/06/2023	Done	
5	Choose Model	All member	28/06/2023	Done	
6	Training, analysing result	Thanh, Bảo	06/07/2023	Done	
7	Report	All member	12/07/2023	Done	
8	Demo	Thanh	12/07/2023	Done	
9	Reference	Trí	12/07/2023	Done	

[Figure17] Contributions