School of Business
OPIM 5604: Predictive Modeling
Preprocessing Project: Group 4

Airbnb Dataset: Bangkok, Central Thailand, Thailand

Vinay Kiran Reddy Chinnakondu vinay_kiran_reddy.chinnakondu@uconn.edu

Table of Contents

Introduction

The purpose of this project is to prepare a dataset for building a model to predict “price” as the target variable. The dataset is sourced from http://insideairbnb.com/get-the-data.html (Inside Airbnb) where listing data for many major cities across the world is available. The Group 4 project was assigned data from Bangkok, Central Thailand, Thailand (Bangkok dataset). It includes seventy-five (75) columns and 18,880 rows. This report documents the preprocessing steps taken to prepare the dataset for modeling using the first three (3) steps of the SEMMA process, i.e., (1) sample, (2) explore, and (3) modify.

Sample

We implemented a random sampling strategy to partition the dataset into distinct subsets for the purpose of model development and evaluation. The key details of our sampling approach are as follows:

→ **Random Sampling:** We employed a random sampling methodology to ensure that the selection of data points was unbiased and representative of the entire dataset.
→ **Data Split:** The dataset was divided into three primary subsets with the following proportions: 60% for training, 20% for validation, and 20% designated for testing.

This method of sampling allowed us to create separate datasets for training, validation, and testing, ensuring that our machine learning model would be effectively trained, assessed, and evaluated with diverse data subsets.

Explore

In this section, the group focused on each of the 75 columns in the Bangkok dataset to determine if we will use the column for modeling – a yes/no decision is made for each column.
Study the Data Dictionary and understand what each variable meant.
Analyze the interconnected relationships between variables.
The main criteria used for selecting columns in the sampling step was checking for missing values.

Missing values

We found twenty-one (21) columns which had approximately 20%, or more, missing values and we decided not to use those columns in our final model. The columns were discovered using the “Explore Missing Values” function in JMP. A snapshot of the results from the “Explore Missing Values” in JMP is shown below – you can see “neighbourhood_group_cleansed” through “host_location” fit our defined missing values criteria, however no data cleansing or modifications were performed.

Explore Missing Values

Commands

Command	Description
Missing Value Report	Number of missing values for each column
Missing Value Clustering	Hierarchical clustering of rows and columns missingness
Missing Value Snapshot	Patterns of missing values with graphical map
Multivariate Normal Imputation	Least squares prediction from the nonmissing variables in each row
Multivariate SVD Imputation	Imputation for wide problems using a singular value decomposition with the power-method adapted for missing values
Automated Data Imputation	Automatically selects best dimension for low-rank approximation based on the data and has streaming imputation capabilities

Automated Data Imputation Controls

Missing Columns

Show only columns with missing

Close

Select columns and choose an action.

Select Rows | Color Cells | Exclude Rows | Color Rows

Column	Number Missing
neighbourhood_group_cleansed	18879
bathrooms	18879
calendar_updated	18879
license	18879
host_neighbourhood	9642
neighborhood_overview	9610
neighbourhood	9610
host_about	7297
review_scores_value	6728
review_scores_location	6727
review_scores_checkin	6726
review_scores_cleanliness	6722
review_scores_communication	6722
review_scores_accuracy	6721
first_review	6598
last_review	6598
review_scores_rating	6598
reviews_per_month	6598
bedrooms	4946
host_is_superhost	4266
host_location	4213
description	559
beds	328
bathrooms_text	96

Some anomalies (i.e., N/A, High Nines, and ###) were found in the dataset and these values were converted accordingly. The screenshot below shows the columns found with High Nines. These Highest Nines were converted to missing values.

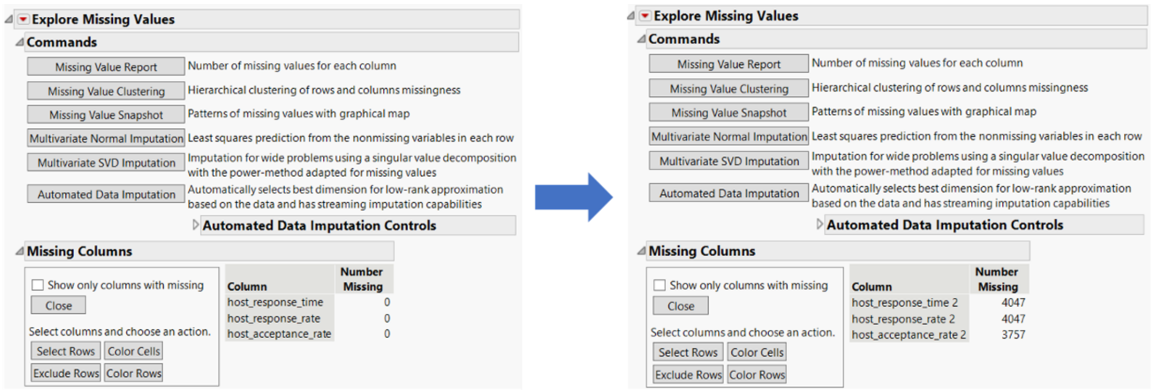
Nines

Column	Count	Highest Nines	90% Quantile
minimum_nights	4	999	30
maximum_nights	1	99999	1125
minimum_minimum_nights	2	99	30
maximum_minimum_nights	2	99	30
minimum_maximum_nights	1	99999	7599.38
maximum_maximum_nights	1	99999	1142.5
minimum_nights_avg_ntm	2	99	30
maximum_nights_avg_ntm	1	99999	1142.5

Nines

Column	Count	Highest Nines	90% Quantile
minimum_nights	2	99	30
minimum_minimum_nights	2	99	30
maximum_minimum_nights	2	99	30
minimum_nights_avg_ntm	2	99	30

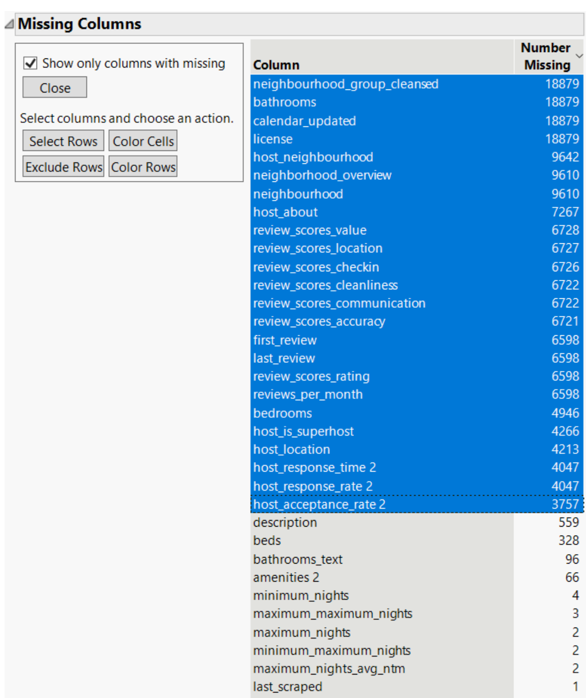
N/A columns were also converted to missing values, seen in the screenshot below.



After converting all anomalies to missing values, another missing values report was run. The table below shows the additional columns that now meet our criteria for ignoring based on 20%, or more, missing values.

No.	Variable	Variable type	Before conversion	After conversion
1	host_response_time	Character → Nominal	0	4,047
2	host_response_rate	Character → Nominal	0	4,047
3	host_acceptance_rate	Character → Nominal	0	3,757

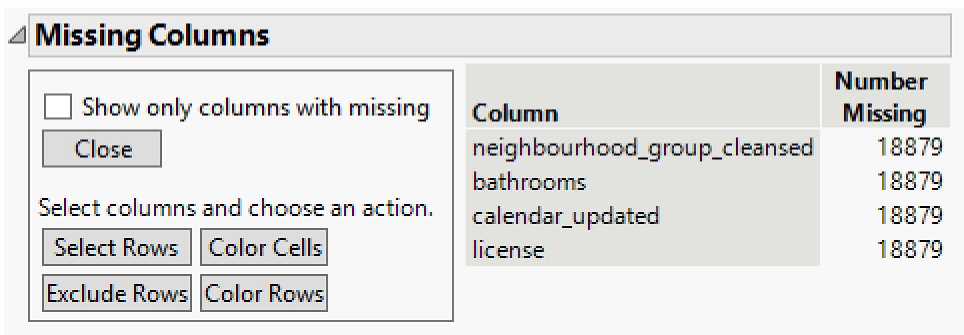
Once all conversions have been made, a final run of "Explore Missing Columns" was executed. The result is now twenty-four (24) columns that meet the criteria. A screenshot shows these columns (highlighted in blue) below.



The table below shows each column that will be ignored, variable type, and number of missing rows.
The tables below show each column that will be ignored, variable type, and number of missing rows.

- The columns listed in the table below exhibit a complete absence of values, indicating a lack of meaningful or pertinent information for the model. Consequently, we chose to eliminate these columns to improve model performance and streamline the dataset. The presence of 100% missing values within these columns suggests significant data quality problems, rendering imputation impractical. This decision to exclude them from the analysis simplifies the model while maintaining performance, as it involves removing variables with both an extensive absence of data and limited predictive relevance.

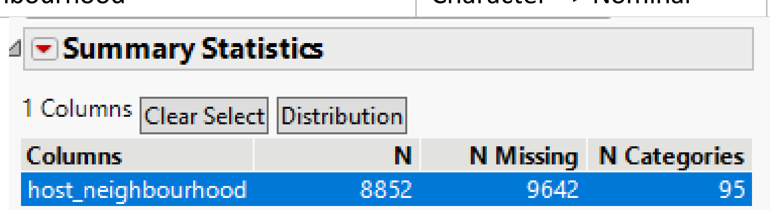
No.	Variable	Variable type	Number missing
1	**neighbourhood_group_cleansed**	Character --> Nominal	18,879
2	**bathrooms**	Character --> Nominal	18,879
3	**calendar_updated**	Character --> Nominal	18,879
4	**license**	Character --> Nominal	18,879



- We decided to eliminate the "**host_neighbourhood**" variable from modeling because of the following reasons

  - The "host_neighbourhood" variable has a substantial number of missing values (9,642 out of the total), which can pose challenges for modeling. Missing data can lead to biased or inaccurate results, and imputing such a large number of missing values may introduce significant uncertainty.
  - The "host_neighbourhood" variable may not directly provide strong predictive power for the model. In this situation where we have 95 categories within a nominal variable, it may not be a meaningful predictor.

No.	Variable	Variable type	Number missing
5	host_neighbourhood	Character --> Nominal	9,642



- The **"neighbourhood_overview "** column contains lengthy, descriptive text that provides information about the neighborhood and local amenities. While this information might be valuable for a human reader, it may not be a good fit for a machine learning model.
  - In our dataset, 9,610 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis
  - The column contains unstructured and free-text descriptions. Machine learning models typically work with structured data, such as numerical or categorical variables. Dealing with unstructured text data would require text processing techniques like natural language processing (NLP), which can significantly complicate the modeling process.
  - The column lacks quantitative or categorical data that can be directly used for modeling. Instead, it contains descriptive language that is more suitable for human understanding than for predictive modeling.

For these reasons, it's generally more practical and effective to exclude the "neighbourhood_overview" column.

No.	Variable	Variable type	Number missing
6	neighbourhood_overview	Character --> Nominal	9,610

Summary Statistics

1 Columns Clear Select Distribution

Columns	N	N Missing	N Categories
neighborhood_overview	8895	9610	5635

- The **"neighbourhood"** column contains location information of different neighborhoods in Bangkok, Thailand. While this information can be relevant for certain types of analysis or models.
  - In our dataset, 9,610 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis
  - The column has 611 categories, including variable with these many categories can significantly increase the complexity of the model. This may lead to a larger number of parameters and potentially overfitting.

No.	Variable	Variable type	Number missing
7	neighborhood	Character --> Nominal	9,610

Summary Statistics

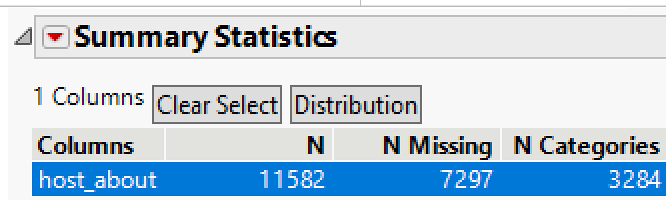
1 Columns Clear Select Distribution

Columns	N	N Missing	N Categories
neighbourhood	9269	9610	611

- The **"host_about"** column contains textual information about the host and their property, which, while informative for humans, may not be well-suited for inclusion in a machine learning model for several reasons.
  - In our dataset, 9,610 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis
  - The column contains unstructured and free-text descriptions. Machine learning models typically work with structured data, such as numerical or categorical variables. Dealing with unstructured text data would require text processing techniques like natural language processing (NLP), which can significantly complicate the modeling process.
  - The column lacks quantitative or categorical data that can be directly used for modeling. Instead, it contains descriptive language that is more suitable for human understanding than for predictive modeling.
  - The column has 3,284 distinct categories, and the presence of such a high number of categories can substantially complicate the model.

For these reasons, it's generally more practical and effective to exclude the "neighbourhood_overview" column.

No.	Variable	Variable type	Number missing
8	host_about	Character --> Nominal	7,297



- The columns listed below, including "**review_scores_value**," "**review_scores_location**," "**review_scores_checkin**," "**review_scores_cleanliness**," "**review_scores_communication**," "**review_scores_accuracy**," "**review_scores_rating**," and "**reviews_per_month**" collectively exhibit a substantial number of missing values, each surpassing 6,500. The decision to exclude these columns is explained in the following reasons:
  - These columns exhibit a significant number of missing values, with more than 6,500 values absent in each of them. Imputing such a large volume of missing data could introduce substantial uncertainty and potential inaccuracies into the model.
  - The correlation coefficients between these columns and the target variable "price" are quite low, with the highest correlation being only 0.1453 for "reviews_per_month." This suggests that these columns may have limited predictive power in explaining the variation in the target variable.
  - To maintain data quality and ensure that the model is trained on meaningful and relevant features, eliminating columns with a high number of missing values can improve the overall quality of the dataset.

No.	Variable	Variable type	Number missing
9	review_scores_value	Numeric --> Continuous	6,728
10	review_scores_location	Numeric --> Continuous	6,727

11	review_scores_checkin	Numeric --> Continuous	6,726
12	review_scores_cleanliness	Numeric --> Continuous	6,722
13	review_scores_communication	Numeric --> Continuous	6,722
14	review_scores_accuracy	Numeric --> Continuous	6,721
15	review_scores_rating	Numeric --> Continuous	6,598
16	reviews_per_month	Numeric --> Continuous	6,598

Summary Statistics

8 Columns Clear Select Distribution

Columns	N	N Missing	Min	Max	Mean	Std Dev
review_scores_rating	12281	6598	0	5	4.610107483104	0.723025777612
review_scores_accuracy	12158	6721	1	5	4.7032891923014	0.5387878880899
review_scores_cleanliness	12157	6722	0	5	4.6625836966357	0.5512096651365
review_scores_checkin	12153	6726	0	5	4.7394091993746	0.5330200729128
review_scores_communication	12157	6722	0	5	4.7618211729868	0.5183332946532
review_scores_location	12152	6727	0	5	4.6110623765635	0.5376703822423
review_scores_value	12151	6728	0	5	4.631195786355	0.5596893998445
reviews_per_month	12281	6598	0.01	53.97	0.9270287435877	1.291908055304

Multivariate

Correlations

	review_scores_rating	review_scores_accuracy	review_scores_cleanliness	review_scores_checkin	review_scores_communication	review_scores_location	review_scores_value	reviews_per_month	price
review_scores_rating	1.0000	0.8496	0.8053	0.7642	0.7800	0.6723	0.8572	0.1453	0.0023
review_scores_accuracy	0.8496	1.0000	0.7875	0.7435	0.7570	0.6412	0.8216	0.1133	0.0209
review_scores_cleanliness	0.8053	0.7875	1.0000	0.6708	0.6730	0.5863	0.7659	0.1073	0.0329
review_scores_checkin	0.7642	0.7435	0.6708	1.0000	0.7919	0.6197	0.7328	0.0880	0.0182
review_scores_communication	0.7800	0.7570	0.6730	0.7919	1.0000	0.6092	0.7480	0.1086	0.0018
review_scores_location	0.6723	0.6412	0.5863	0.6197	0.6092	1.0000	0.6840	0.1159	0.0527
review_scores_value	0.8572	0.8216	0.7659	0.7328	0.7480	0.6840	1.0000	0.1281	-0.0003
reviews_per_month	0.1453	0.1133	0.1073	0.0880	0.1086	0.1159	0.1281	1.0000	0.0258
price	0.0023	0.0209	0.0329	0.0182	0.0018	0.0527	-0.0003	0.0258	1.0000

- The "**First_review**" column contains dates when the first/oldest review was given and the "**Last_review**" gives the date of the last/newest review. We are excluding the "First_review" and "Last_review" columns because of the following reasons:
  - Both columns contain 6,500 missing values each. Such a large number of missing data points can hinder the utility of these columns in analysis and modeling. Imputing such a large volume of missing data could introduce substantial uncertainty and potential inaccuracies into the model.
  - "First_review" and "Last_review" are date columns. While date information can be valuable for time-series analysis or specific temporal modeling, they are not directly relevant to the primary modeling objective, as our primary objective is on predicting property prices.
  - There are 2731 categories in First_review column and 1604 categories in the Last_review column and the presence of such a high number of categories can substantially complicate the model.

No.	Variable	Variable type	Number missing
17	first_review	Numeric --> Continuous	6,598
18	last_review	Numeric --> Continuous	6,598

Summary Statistics

2 Columns Clear Select Distribution

Columns	N	N Missing	N Categories
first_review	11880	6598	2731
last_review	11880	6598	1604

- The **"review_scores_rating"** column contains textual information about the host's ratings with continuous values between zero (0) and (5).
  - In our dataset, 6,598 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis

- The **"bedrooms"** column contains textual information about the number of beds available per property with continuous values between one (1) and fifty (50).
  - In our dataset, 4,946 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis

For these reasons, it's generally more practical and effective to exclude the "**bedrooms**" column.

No.	Variable	Variable type	Number missing
19	bedrooms	Character --> Continuous	4,946

Summary Statistics

1 Columns Clear Select Distribution

Columns	N	N Missing	Min	Max	Mean	Std Dev
bedrooms	13933	4946	1	50	1.520706237	1.8711600789

- The **"host_is_superhost"** column contains textual information showing whether the host is a superhost, or not. This is a nominal column.
  - In our dataset, 4,266 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis
  - The column has 2 categories.

For these reasons, it's generally more practical and effective to exclude the "host_is_superhost" column.

No.	Variable	Variable type	Number missing
20	host_is_superhost	Character --> Continuous	4,266

Summary Statistics

1 Columns Clear Select Distribution

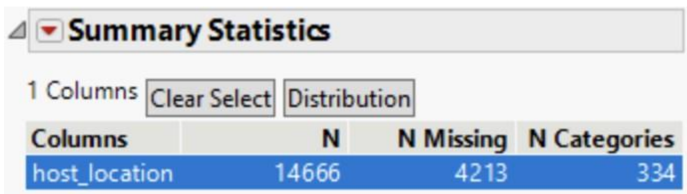
Columns	N	N Missing	N Categories
host_is_superhost	14613	4266	2

➢ The **"host_location"** column contains textual information indicating the location of the host's property.

- In our dataset, 4,213 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis
- The column has 334 categories. Including a variable with these many categories can significantly increase the complexity of the model. This may lead to a larger number of parameters and potentially overfitting.

For these reasons, it's generally more practical and effective to exclude the "host_location" column.

No.	Variable	Variable type	Number missing 4,213
21	host_location	Character --> Nominal	4,213



Summary Statistics

1 Columns Clear Select Distribution

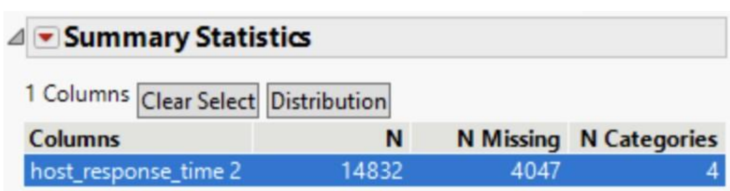
Columns	N	N Missing	N Categories
host_location	14666	4213	334

➢ The **"host_response_time"** column contains textual information indicating how long it takes for the host to respond to customers.

- In our dataset, 4,047 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis

For these reasons, it's generally more practical and effective to exclude the "host_response_time" column.

No.	Variable	Variable type	Number missing
22	host_response_time	Character --> Nominal	4,047



Summary Statistics

1 Columns Clear Select Distribution

Columns	N	N Missing	N Categories
host_response_time 2	14832	4047	4

➢ The **"host_response_rate"** column contains textual information describing their response rate in percentages.

- In our dataset, 4,047 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis

For these reasons, it's generally more practical and effective to exclude the “host_response_rate” column.

No.	Variable	Variable type	Number missing
23	host_response_rate	Character --> Nominal	4,047

Summary Statistics

1 Columns | Clear Select | Distribution

Columns	N	N Missing	N Categories
host_response_rate 2	14832	4047	53

- The **"host_acceptance_rate"** column contains textual information showing the host’s acceptance rates.
  - In our dataset, 3,757 values are absent, and the presence of missing data can potentially result in biased or inaccurate outcomes. Attempting to impute such a substantial number of missing values may introduce considerable uncertainty into the analysis

For these reasons, it's generally more practical and effective to exclude the “host_acceptance_rate” column.

No.	Variable	Variable type	Number missing
24	host_acceptance_rate	Character → Nominal	3,757

Summary Statistics

1 Columns | Clear Select | Distribution

Columns	N	N Missing	N Categories
host_acceptance_rate 2	15122	3757	69

“Eyeballing” the data

The URL (uniform resource locator) columns had no correlation to the target variable (price) since they are just links to their described objective. The same applies to other columns shown in the table below which contain freeform text, or unstructured textual data, which have no correlation to price.

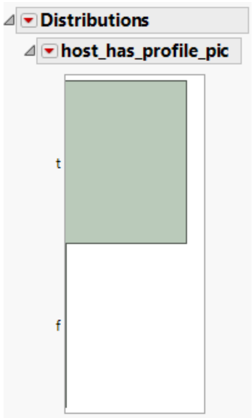
No.	Variable	Variable type	Comments
1	id	Numeric → Continuous	ID
2	listing_url	Character → Nominal	URL
3	scrape_id	Numeric → Continuous	ID
4	description	Character → Nominal	Text
5	picture_url	Character → Nominal	URL
6	host_id	Numeric → Continuous	ID
7	host_url	Character → Nominal	URL
8	host_name	Character → Nominal	Text
9	host_thumbnail_url	Character → Nominal	URL
10	host_picture_url	Character → Nominal	URL
11	host_verifications	Character → Nominal	Because we are using another column, “host_identity_verified,” that has Boolean values, we decided to use that instead of a Text column.
12	latitude	Numeric → Continuous	Since this is a city-specific data, there is not much variation in either latitude or longitude. These columns would be more useful if we were exploring global data.
13	longitude	Numeric → Continuous	

Distribution

The following columns are almost constant with very low variability. Because the variability of these columns is low, we do not see any value in using these columns for the predictive model.

No.	Variable	Variable type	Comments
1	host_has_profile_pic	Numeric → Continuous	1.022% See snapshot below.
2	source	Character → Nominal	It has only two values (city scrape and previous scrape) and is evenly spread across the price. See snapshots below.
3	last_scraped	Character → Nominal	All data revolves around 26^{th} and 27^{th} June and that too does not have any variation over price. See snapshots below.
4	calendar_last_scraped	Numeric → Continuous	
5	instant_bookable	Character → Nominal	The mean is nearly the same for both true and false values and it does not provide much variability. See snapshots below.

The distribution plot of “host_has_profile_pic,” below, shows no variation between true and false values, i.e., 99:1.



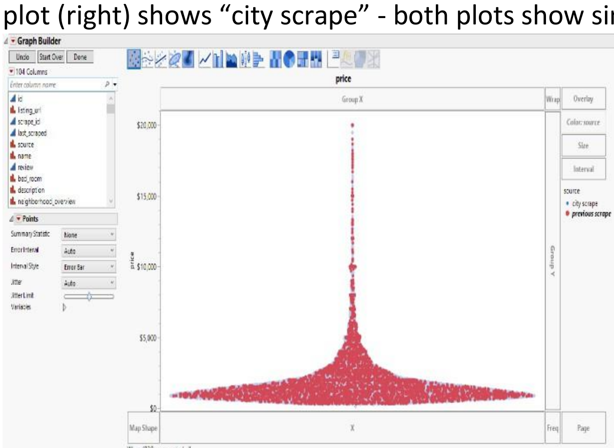
Frequencies

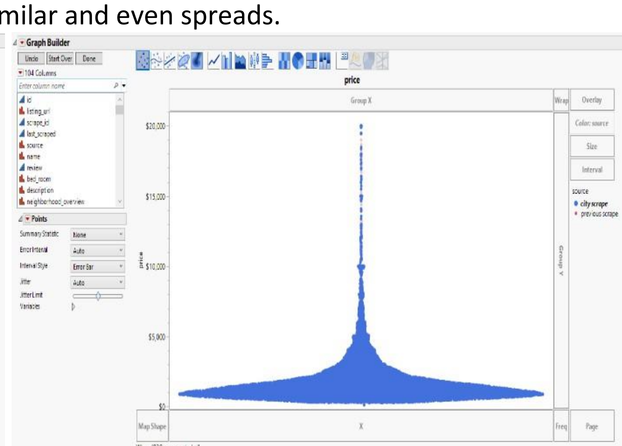
Level	Count	Prob
f	193	0.01022
t	18686	0.98978
Total	18879	1.00000

N Missing 0
2 Levels

Distribution plot: “host_has_profile_pic”

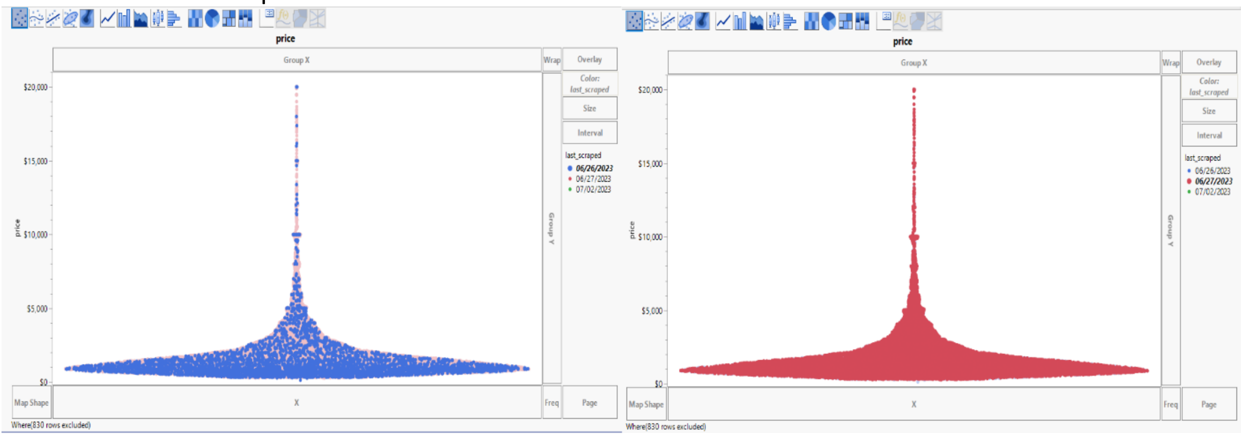
These two scatterplots show “price” vs. “source.” The blue plot (left) shows “previous scrape” and the red plot (right) shows “city scrape” - both plots show similar and even spreads.





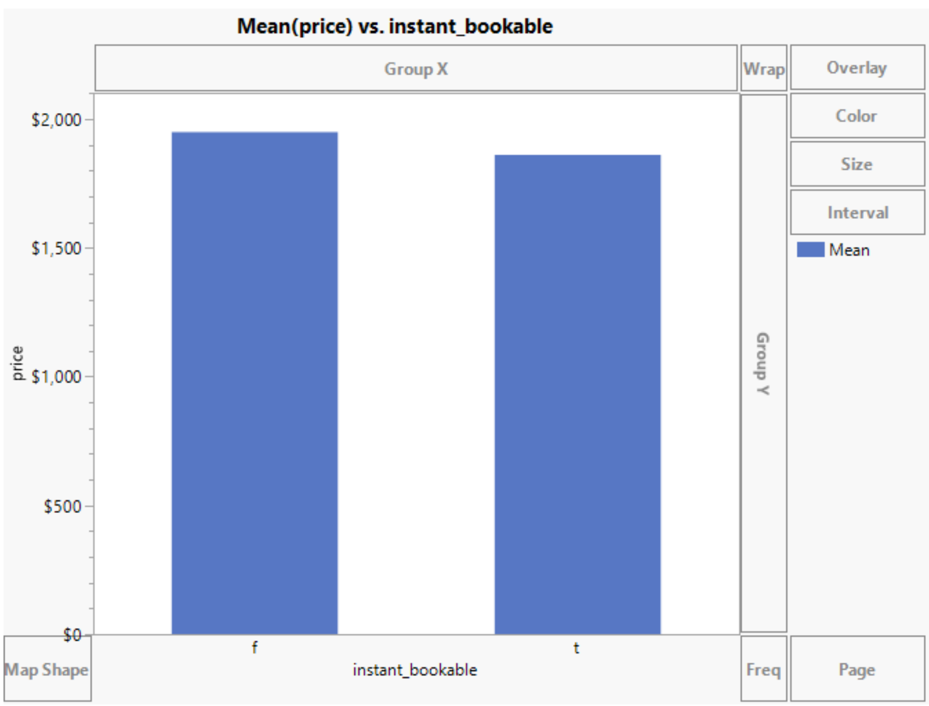
Scatterplots: “price” vs. “source”

These two snapshots show scatterplots for “price” vs. the dates in “last_scraped.” Like “source,” they demonstrate similar spreads.



Scatterplots: “price” vs. “last_scraped”

The final snapshot shows a bar chart shows “mean(price)” vs. “instant_bookable” where it clearly shows little variability between the true and false values.



Bar chart: “Mean(price)” vs. “instant_bookable”

Correlation

The columns in the table below show high correlation with "availability_90" and have redundant information. See snapshot below showing the relationship between "availability_ 30," "availability_ 60," "availability_ 365," and "price."

No.	Variable	Variable Type	Correlation
1	availability_30	Numeric → Continuous	0.8983
2	availability_60	Numeric → Continuous	0.9793
3	availability_365	Numeric → Continuous	0.6992

Multivariate

Correlations

	price	availability_30	availability_60	availability_90	availability_365
price	1.0000	-0.0042	-0.0093	-0.0126	-0.0053
availability_30	-0.0042	1.0000	0.9497	0.8983	0.5966
availability_60	-0.0093	0.9497	1.0000	0.9793	0.6585
availability_90	-0.0126	0.8983	0.9793	1.0000	0.6992
availability_365	-0.0053	0.5966	0.6585	0.6992	1.0000

Similarly, the columns below are highly correlated with "number_of_reviews_ltm" and have redundant information.

No	Variable	Variable type	Correlation
1	number_of_reviews	Numeric → Continuous	0.6367
2	number_of_reviews_l30d	Numeric → Continuous	0.7055

Multivariate

Correlations

	price	number_of_reviews	number_of_reviews_ltm	number_of_reviews_l30d
price	1.0000	-0.0069	-0.0133	-0.0115
number_of_reviews	-0.0069	1.0000	0.6367	0.4093
number_of_reviews_ltm	-0.0133	0.6367	1.0000	0.7055
number_of_reviews_l30d	-0.0115	0.4093	0.7055	1.0000

The column below is highly correlated with "host_listings_count" and has redundant information.

No.	Variable Name	Variable Type	Correlation
1	host_total_listings_count	Numeric → Continuous	0.9477

Multivariate

Correlations

	host_listings_count	host_total_listings_count	price
host_listings_count	1.0000	0.9477	0.0128
host_total_listings_count	0.9477	1.0000	0.0011
price	0.0128	0.0011	1.0000

Relativity

The following columns are related to “calculated_host_listings_count”, “room_type”, “minimum_nights” & “maximum_nights” respectively. The relative columns are correlated better with “price” comparatively and thus we eliminated these to reduce the dimensionality.

No.	Variable Name	Variable type	Relative variable
1	calculated_host_listings_count_entire_home	Numeric → Continuous	calculated_host_listings_count
2	calculated_host_listings_count_private_room	Numeric → Continuous	
3	calculated_host_listings_count_shared_room	Numeric → Continuous	
4	property_type	Character → Nominal	room_type
5	minimum_minimum_nights	Numeric → Continuous	minimum_nights
6	maximun_minimum_nights	Numeric → Continuous	
7	minimum_nights_avg_ntm	Numeric → Continuous	
8	minimum_maximum_nights	Numeric → Continuous	maximum_nights
9	maximum_maximum_nights	Numeric → Continuous	
10	maximum_nights_avg_ntm	Numeric → Continuous	

Modify

New binary columns

Because the following columns consisted of only true and false values, we decided to convert them to binary and keep just the one with true to reduce dimensionality. See the table below showing which columns were converted and ultimately kept for use.

No.	Variable converted to binary	Column used
1	host_identity_verified	host_identity_verified_t
2	has_availability	has_availability_t

Formula based columns

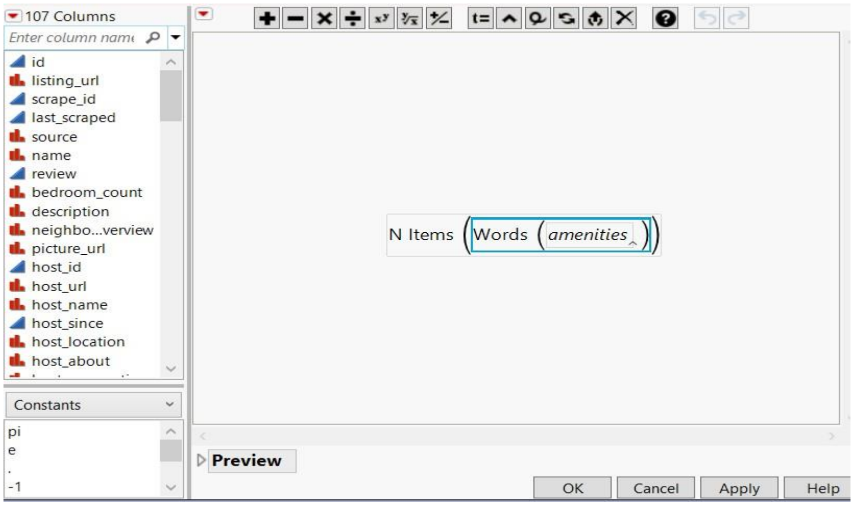
No.	Variable name	Variable type	Extracted from
1	bedroom_count	Numeric → Continuous	name
2	amenities_count	Numeric → Continuous	amenities
3	bathrooms_text	Character → Nominal	Recoded

bedroom_count

We created bedroom_count to get the count of bedrooms in a listing. The “bedrooms” variable consists of 4946 missing values whereas the bedroom_count variable that is extracted from name contains 16 missing values only. So, we are using bedroom_count variable instead of bedrooms. The screenshot below shows the formula we used to extract bedroom_count from name.

amenities_count

We created amenities_count to get the count of amenities in a listing. We extracted the variable from amenities by using the formula mentioned in the below screenshot.



Other ideas on amenities

As there are many categories in the form of text in amenities variable, it increases the complexity of model. But we can't exclude the variable based on this difficulty. So, we had some other ideas as well on handling the amenities which are explained briefly below, along with the reasons for not considering them.

Idea 1:

- The first idea was to make indicator columns using a delimiter, which gave us 2,430 new columns. There were many types of TVs, TV sizes, refrigerators, refrigerator colors etc., as categories so we thought of combining those into one category (for example Sharp, Beko, fridge and refrigerator can be combined to "refrigerator" category) and excluding the categories like shampoo, conditioner, body soap etc., which does not influence the price much according to the business sense. This reduces many indicator columns and only a few important amenities would be left over.
  Reason for not selecting: Although we are reducing data, we are being biased by selecting what amenities to choose and what not to choose. This was also labor intensive. In addition, even if we chose to, the dimensionality would increase.

Idea 2:

- The second idea was to list some important amenities that are important to predict the price pool, air conditioner etc., and use the weighted model theory by weighing them according to their level of importance. Then create a column of count by quantifying how many amenities each listing has.
  Reason for not selecting: It is difficult to put the quantification of the categories into the dataset.

Idea 3:

- The third idea was to list a few important amenities, according to what makes sense to the business, that would impact price and create indicator columns for those.
  Reason for not selecting: If we list amenities based on business sense, then it will become biased.

Idea 4

- The next idea was to combine a few amenities into high, medium and low based on their importance and then create indicator columns.
  Reason for not selecting: This is labor intensive and if you do this, it becomes subjective. It's hard to justify what's considered important and what's not.

bathrooms_text_2

Since it had many categories, and most of the values were concentrated amongst 1 and 2 bathrooms, we decided to recode all the values greater than 2 as "2+ beds" to reduce the complexity of the column.

Data type changes

The following columns consist useful data, but their data type is character, therefore we converted them to continuous.

No.	Variable converted	Type before conversion	Type after conversion
1	host_since	Character	Continuous
2	price	Character	Continuous
3	amenities_count	Character	Continuous

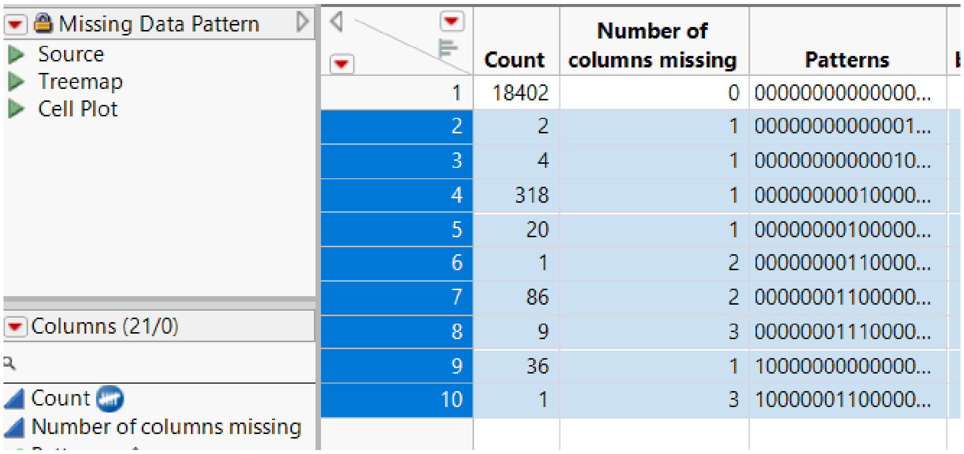
Standardized variables

We standardized the following four (4) variables as they have large scales and skew the result.

No.	Variable	Variable type	New column name
1	amenities_count	Numeric → Continuous	Standardize[amenities_count]
2	minimum_nights	Numeric → Continuous	Standardize[minimum_nights]
3	maximum_nights	Numeric → Continuous	Standardize[maximum_nights]
4	availability_90	Numeric → Continuous	Standardize[availability_90]

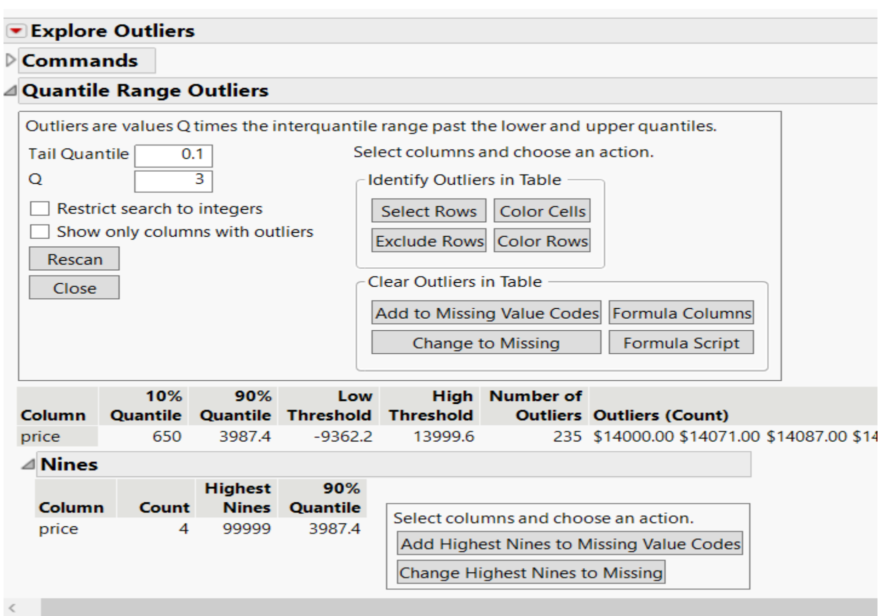
Missing data pattern

We excluded 477 rows in which one (1) or more columns were missing by observing the missing data pattern.



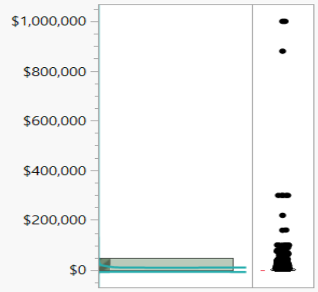
Outlier Analysis on target variable

We identified outliers through a distribution analysis and subsequent outlier analysis. It became evident that the data primarily adhered to a Johnson Su distribution. We then transformed this distribution into a normal one. Following this process, we pinpointed 148 outliers within the "price" column. These detected outliers were subsequently excluded from the analysis, revealing that the values in the "price" column exceeded $20,000.



Distributions

price



Compare Distributions

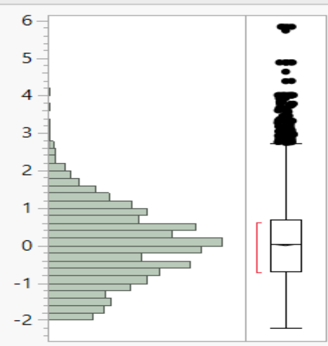
Show	Distribution	AICc	AICc Weight	BIC	-2*LogLikelihood
☑	Johnson Su	309434.51	1	309465.79	309426.51
☐	Lognormal	311238.89	0	311254.53	311234.89
☐	SHASH	312325.17	0	312356.45	312317.17
☐	Student's t	315935.71	0	315959.17	315929.71
☐	Normal 2 Mixture	323265.42	0	323304.52	323255.41
☐	Normal 3 Mixture	323267.12	0	323314.04	323255.11
☐	Weibull	324299.4	0	324315.04	324295.4
☐	Exponential	326369.85	0	326377.67	326367.85
☐	Gamma	326370.84	0	326386.48	326366.84
☐	Normal	412449.7	0	412465.34	412445.7
☐	Cauchy	762580.81	0	762596.45	762576.81

Quantiles

100.0%	maximum	$1,000,000
99.5%		$26,541
97.5%		$8,999
90.0%		$3,987
75.0%	quartile	$2,166
50.0%	median	$1,349
25.0%	quartile	$890

Distributions

Johnson Su Transform to Normal price



Quantiles

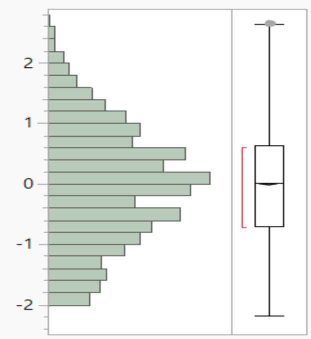
100.0%	maximum	5.834046
99.5%		2.9447226
97.5%		2.0452399
90.0%		1.3101319
75.0%	quartile	0.6693507
50.0%	median	0.0396484
25.0%	quartile	-0.707411
10.0%		-1.293884
2.5%		-1.797388
0.5%		-1.899594
0.0%	minimum	-2.191808

Summary Statistics

Mean	0.0149884
Std Dev	1.0038152
Std Err Mean	0.0073998
Upper 95% Mean	0.0294927
Lower 95% Mean	0.0004841
N	18402

Distributions

Johnson Su Transform to Normal price



Quantiles

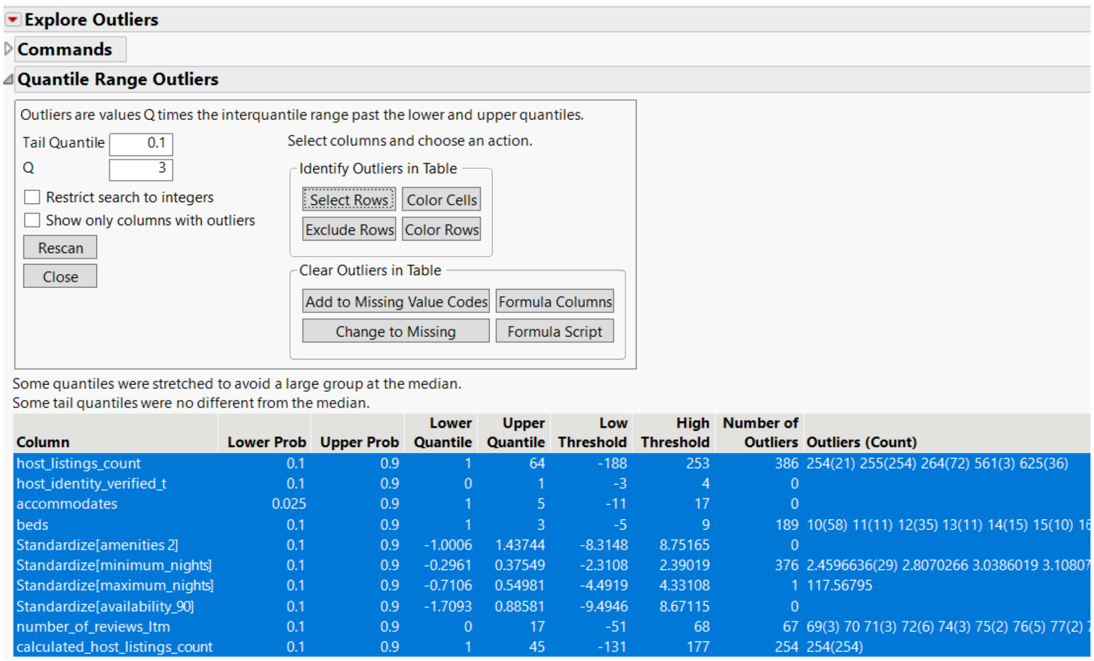
100.0%	maximum	2.6569284
99.5%		2.377456
97.5%		1.9065389
90.0%		1.2632338
75.0%	quartile	0.6329869
50.0%	median	0.0199974
25.0%	quartile	-0.707411
10.0%		-1.293884
2.5%		-1.799274
0.5%		-1.899594
0.0%	minimum	-2.191808

Summary Statistics

Mean	-0.012235
Std Dev	0.9587778
Std Err Mean	0.0070964
Upper 95% Mean	0.001675
Lower 95% Mean	-0.026144
N	18254

Outlier analysis on all continuous variables

We identified outliers through distribution analysis, multivariate and explore outliers' analysis. We found that there are 1017 potential outliers in total from all the continuous variables. These potential outliers were excluded from the dataset.



Column	Lower Prob	Upper Prob	Lower Quantile	Upper Quantile	Low Threshold	High Threshold	Number of Outliers	Outliers (Count)
host_listings_count	0.1	0.9	1	64	-188	253	386	254(21) 255(254) 264(72) 561(3) 625(36)
host_identity_verified_t	0.1	0.9	0	1	-3	4	0	
accommodates	0.025	0.9	1	5	-11	17	0	
beds	0.1	0.9	1	3	-5	9	189	10(58) 11(11) 12(35) 13(11) 14(15) 15(10) 16
Standardize[amenities 2]	0.1	0.9	-1.0006	1.43744	-8.3148	8.75165	0	
Standardize[minimum_nights]	0.1	0.9	-0.2961	0.37549	-2.3108	2.39019	376	2.4596636(29) 2.8070266 3.0386019 3.10807
Standardize[maximum_nights]	0.1	0.9	-0.7106	0.54981	-4.4919	4.33108	1	117.56795
Standardize[availability_90]	0.1	0.9	-1.7093	0.88581	-9.4946	8.67115	0	
number_of_reviews_ltm	0.1	0.9	0	17	-51	68	67	69(3) 70 71(3) 72(6) 74(3) 75(2) 76(5) 77(2)
calculated_host_listings_count	0.1	0.9	1	45	-131	177	254	254(254)

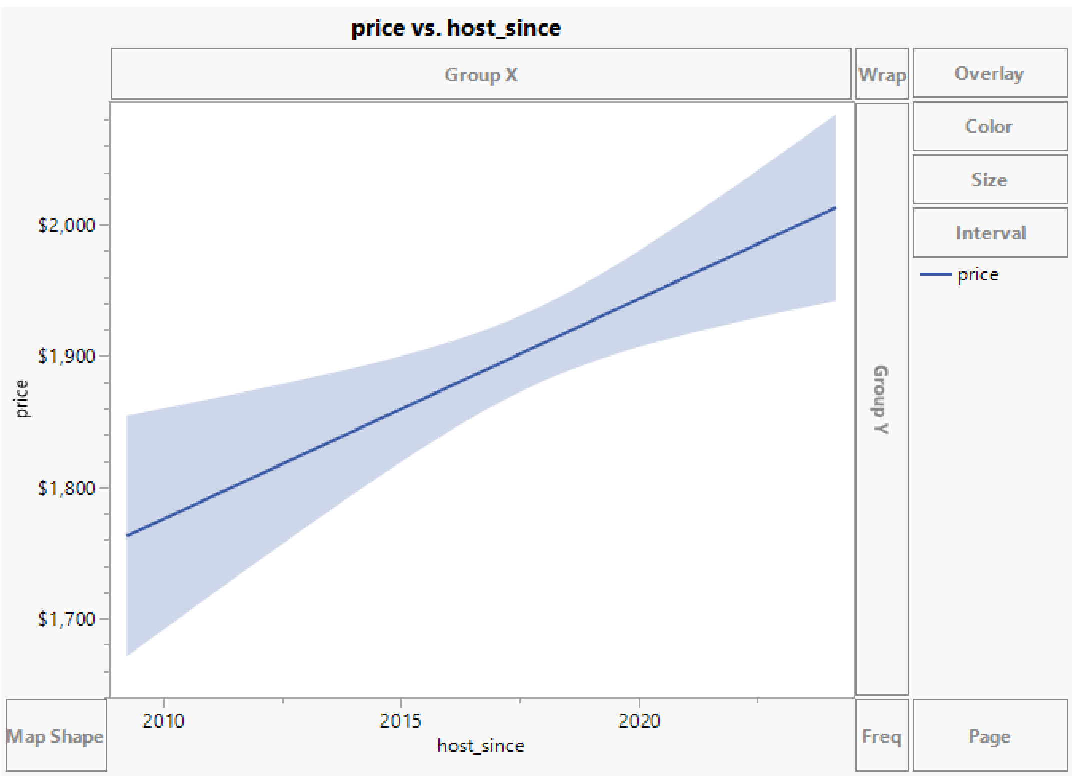
Selected variables

bedroom_count

Instead of using bedroom column, which was available in the data, we decided to get this data from the name column using text extraction. The original column had 4,689 missing values and the column we created has only sixteen (16) missing values and those were taken out too during the Missing values analysis for Rows.

host_since

This column reflects data for the amount of time that the host has been active. It is an important column to determine price and we are not dropping this because it is directly correlated to "price" (refer to line of fit below)



host_listings_count

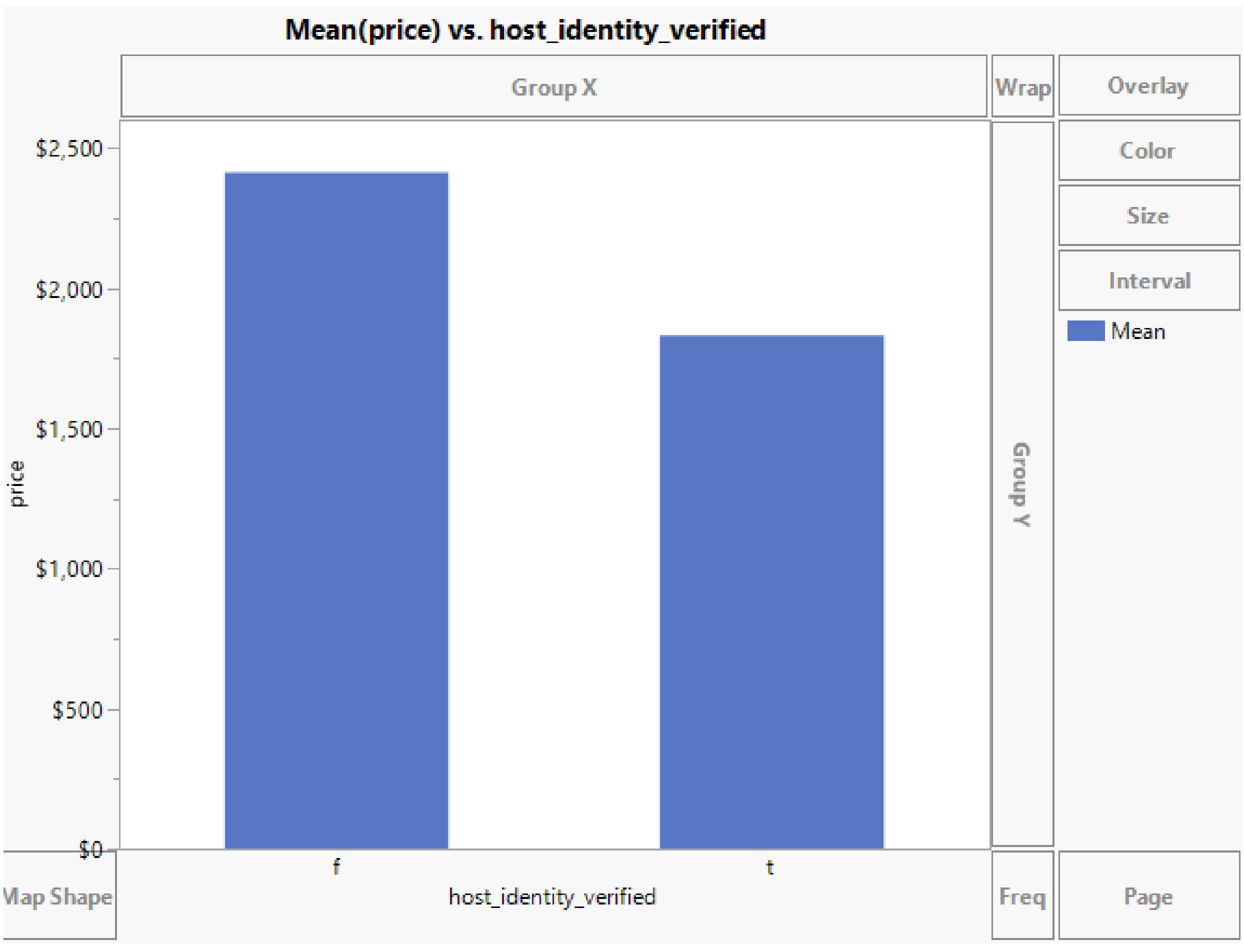
We had another column with redundant information: "host_total_listings_count." We tried to understand the difference between these two (2) columns and referred to data dictionary for that as well, but they had the same definition. At last, we concluded that total would have listings throughout Airbnb all over the world and "host_listings_count" will have information just for the specific city. Please refer to the snapshot below to understand the correlation with "price," and redundancy.

Correlations

	host_listings_count	host_total_listings_count	price
host_listings_count	1.0000	0.9477	0.0128
host_total_listings_count	0.9477	1.0000	0.0011
price	0.0128	0.0011	1.0000

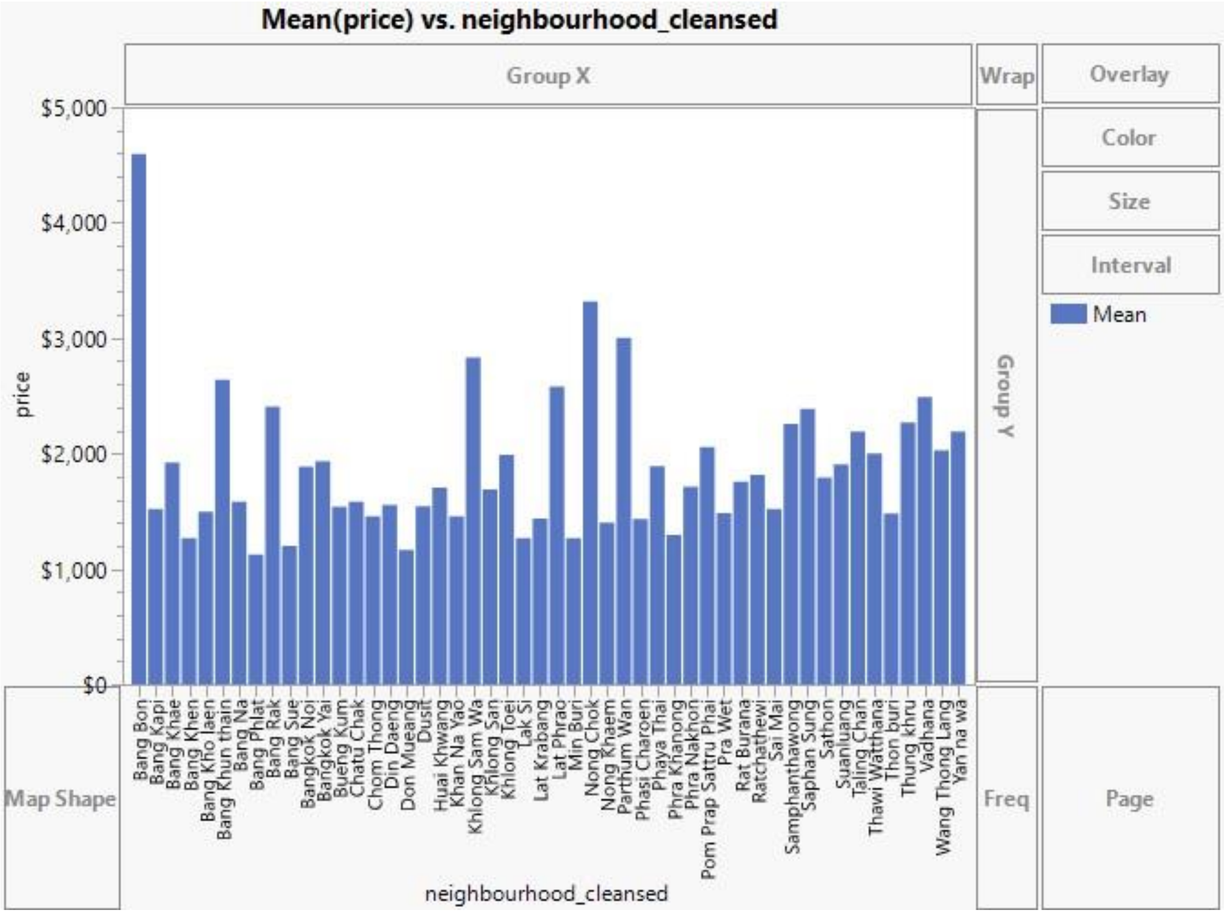
host_identity_verified_t

Since “host_identity_verified” was a Boolean column, we decided to create indicator columns for these and to reduce the dimensionality we kept only one with true values as 1. Please refer to the snapshot below to confirm the clear difference between mean price and verified status.



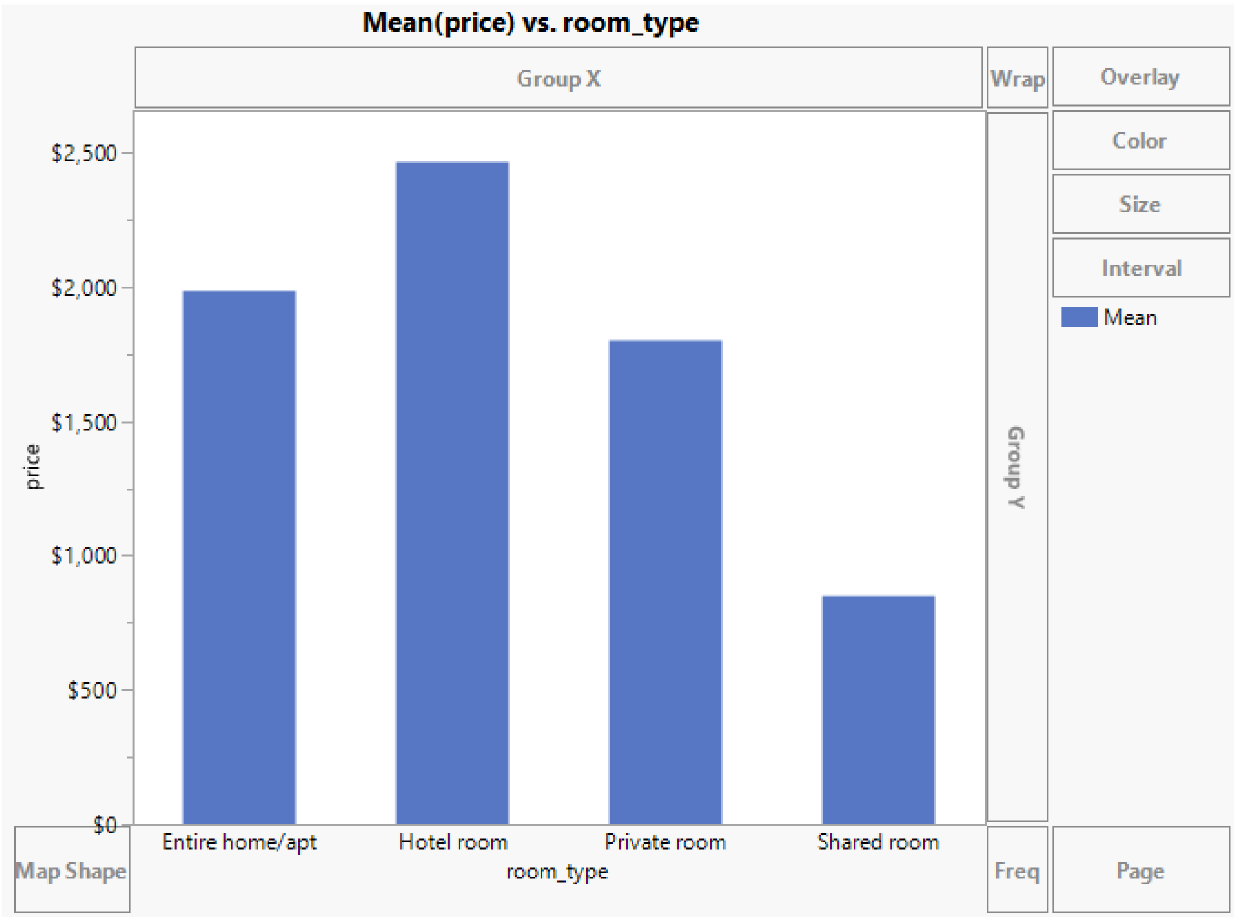
neighbourhood_cleansed

We have decided to use this column because every neighbourhood has a different mean price. This would be a useful indicator in predicting "price" and thus we have decided to use this column as-is.



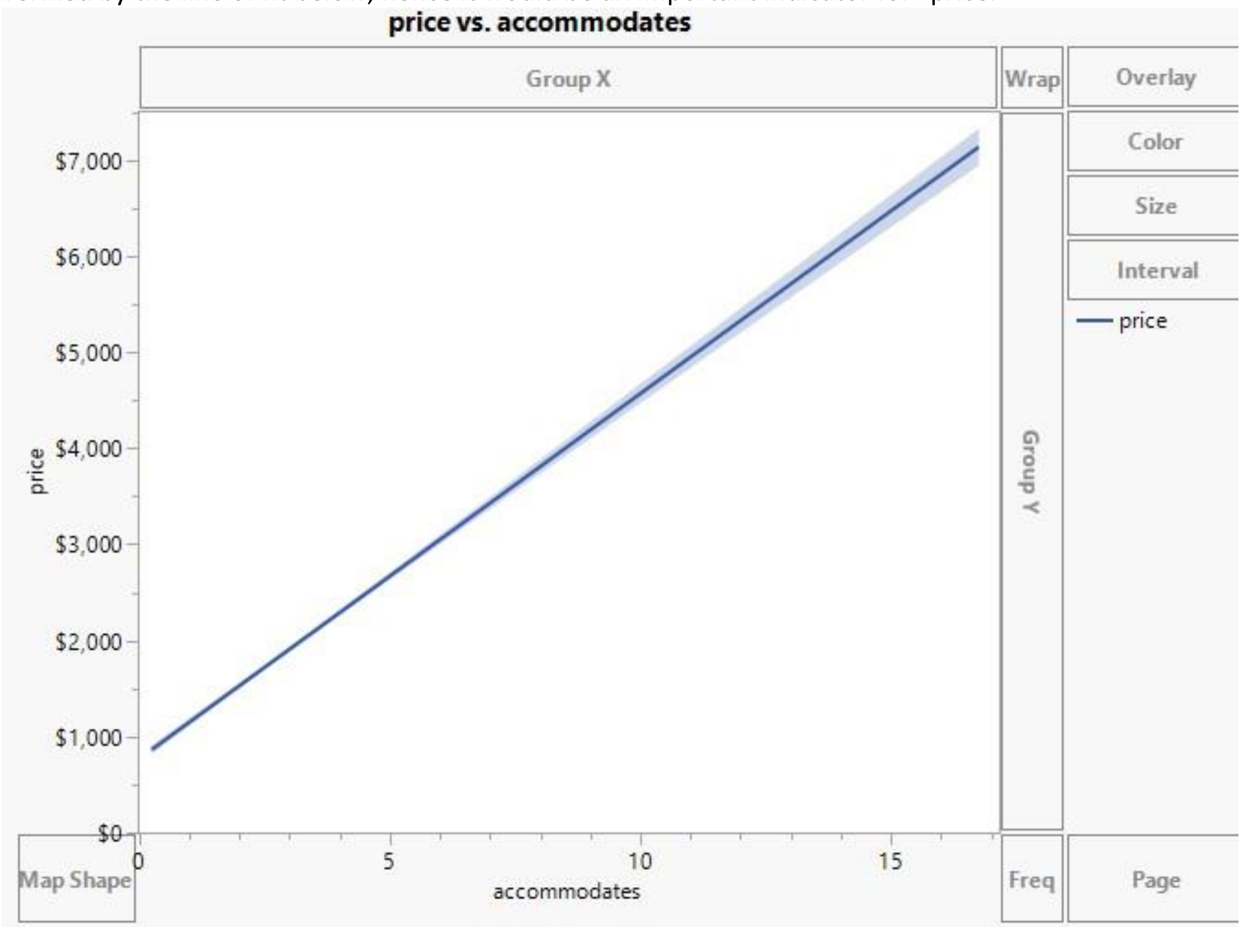
room_type

We have one similar column “property_type,” but we have decided to use “room_type” because it is consolidated and has fewer values. “property_type” contained a lot more values which would have increased the complexity. We can also see that mean price varies with every category in “room_type” and thus it is a useful indicator to predict “price.”



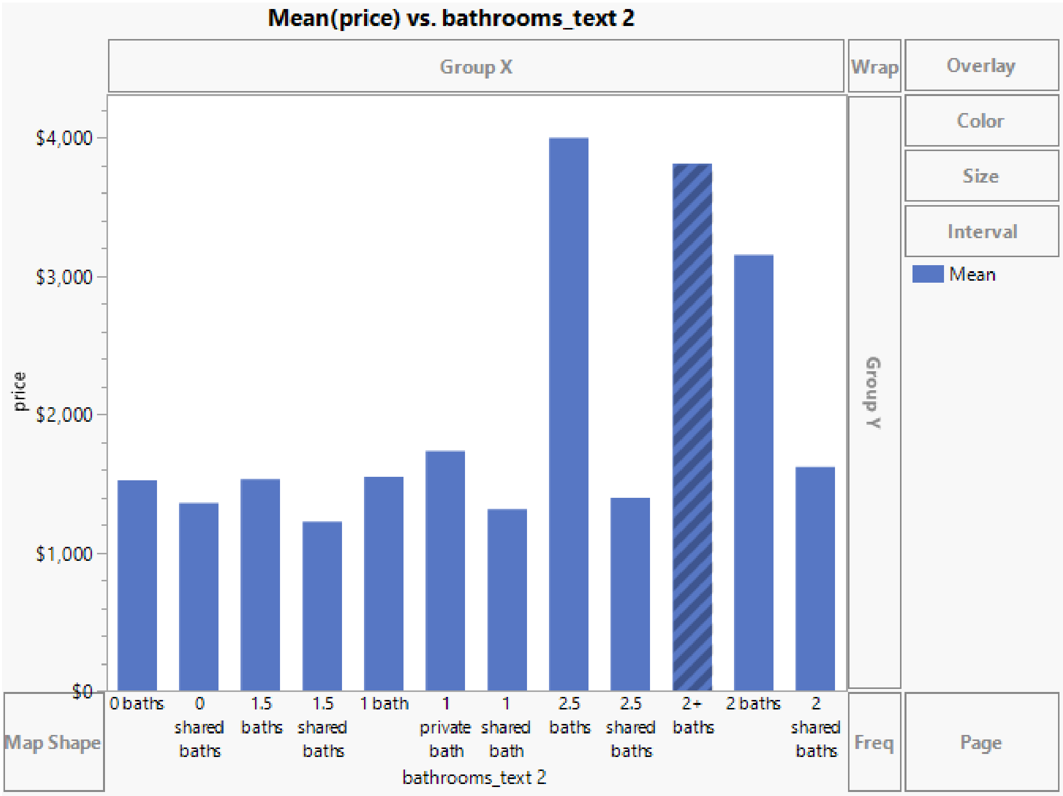
accommodates

We decided to keep this column since “accommodates” is directly proportional to “price” which can be verified by the line of fit below, hence it would be an important indicator for “price.”



bathrooms_text_2

We had "bathrooms_text" column, but this column contained a lot of categories, thus we decided to recode all of the categories with three (3), or more than three (3) bathrooms under 2+ categories. Recoding reduced the complexity, and we can also use this as every category has a different mean price which makes it a useful indicator for "price."

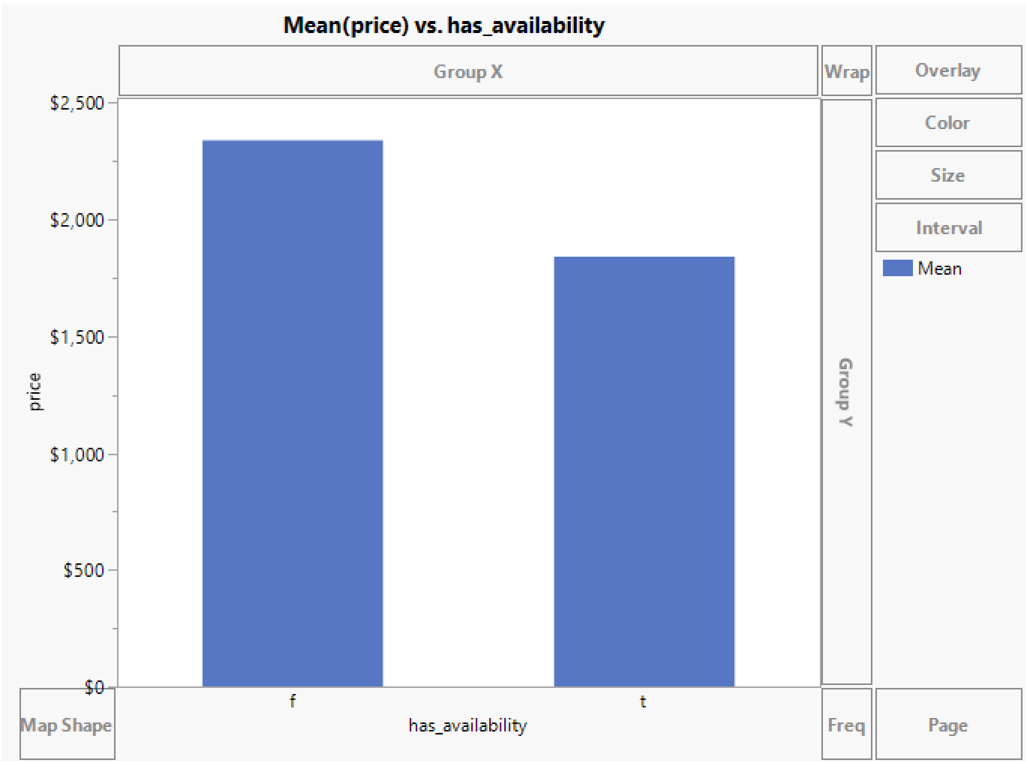


Beds

We have decided to keep this column as-is because it holds the numeric values of beds available per property which is a significant indicator of "price."

has_availability_t

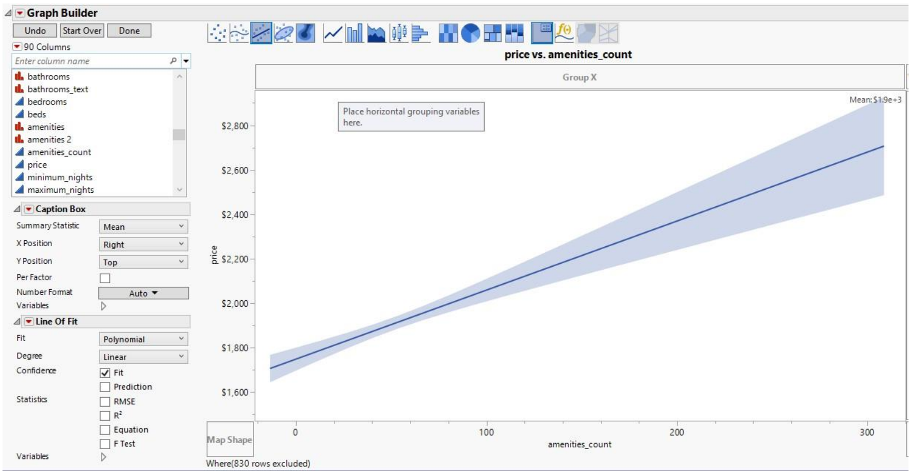
Since “has_availability” is a Boolean column, we decided to create indicator columns for these and to reduce the dimensionality we kept only one (1) with true values as 1. See snapshot below to verify the clear difference between mean price (i.e., “Mean(price)”) and verified status (i.e., “has_availability”).



amenities_count

As mentioned in the *Formula based columns* section, we have already mentioned the reasoning for elimination of the "amenities" column and all other variations associated with it. We also standardized the column since it had large values in comparison to all the other values in the dataset.

The reason we are keeping count of amenities (i.e., "amenities_count") is because it is directly proportional to "price," as can be seen in the line of fit below.

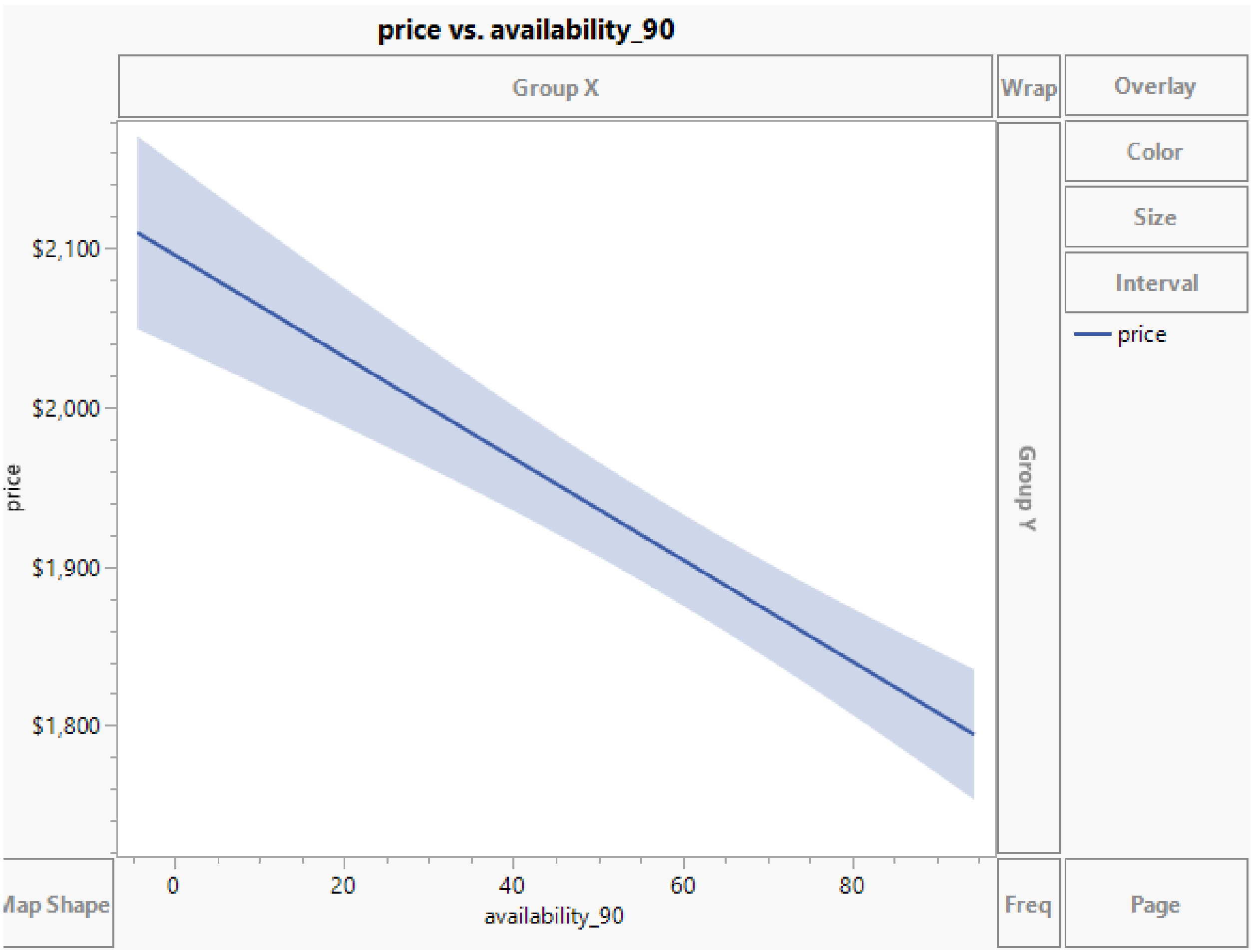


minimum nights & maximum nights

In the *Correlation* section, we have mentioned the reasoning for keeping "minimum_nights" & "maximum_nights" and drop all other related columns to reduce dimensionality.
We have also standardized these columns since they had very extreme values compared to other columns in the dataset.

availability_90

In the *Correlation* section, we have mentioned the reasoning for keeping the "availability_90" column and eliminating all other related columns to reduce dimensionality.
This column is also inversely proportional to "price" which makes it a good indicator to predict price.
Additionally, we have also standardized the column because it has very extreme values compared to other columns in the dataset.

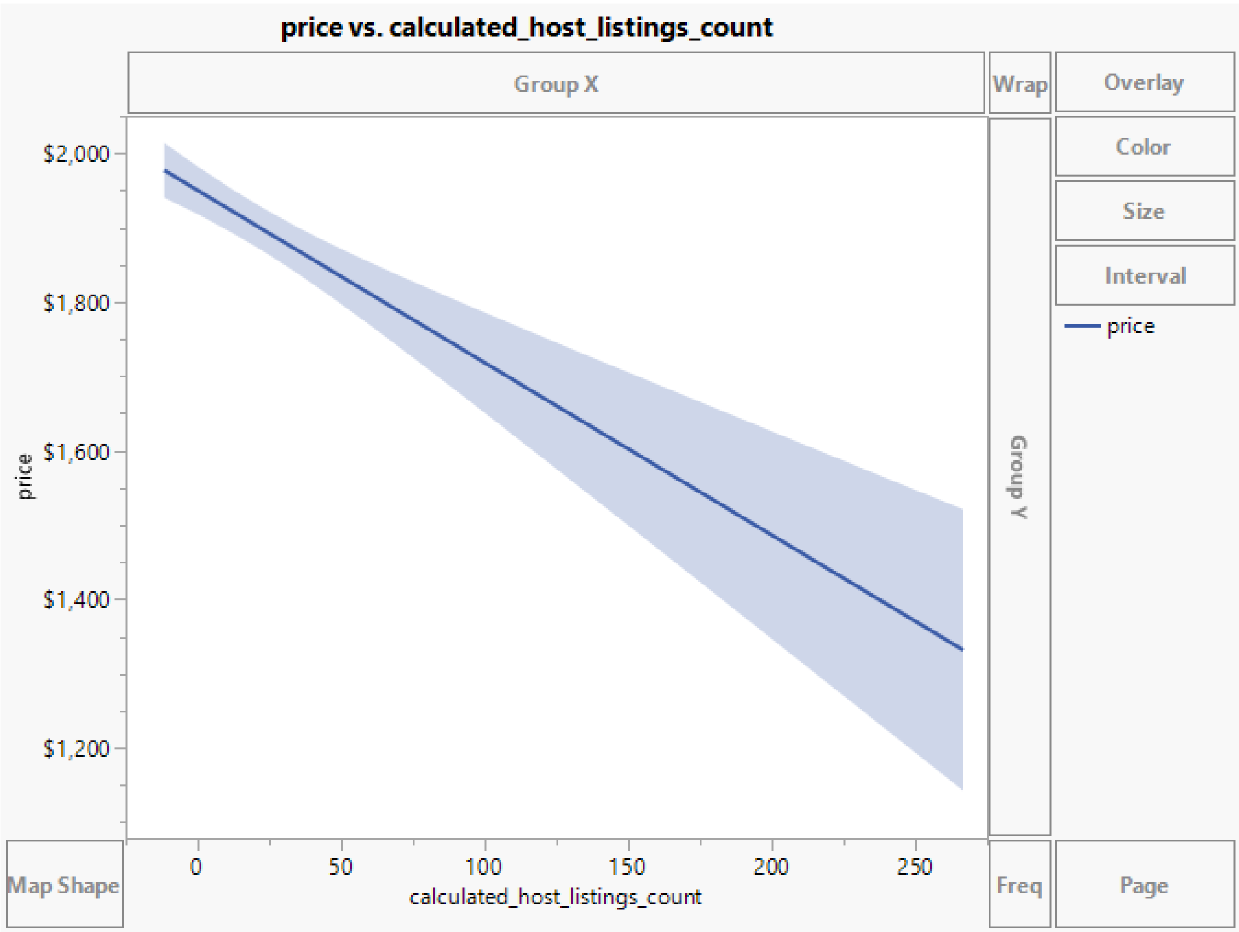


number_of_reviews_ltm

In the *Correlation* section, we have mentioned the reasoning for keeping reviews just for the last twelve months (i.e., "number_of_reviews_ltm") and dropping all other related columns to reduce dimensionality.
This column is a better indicator of price, with no extreme values, and therefore we decided to keep it as-is.

calculated_host_listings_count

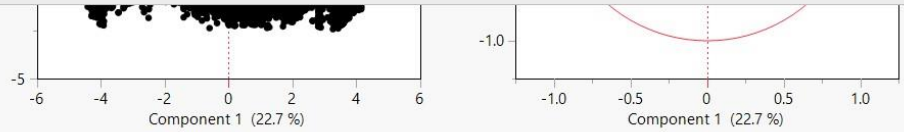
As mentioned in the *Relativity* section, we have justified why we chose to keep "calculated_host_listings_count" over other related columns. It can also be seen that this column is inversely proportional to "price" which makes it a useful indicator for prediction.



Principal Component Analysis (PCA)

Principal Components: on Correlations

Summary Plots



Select component Component 1 Component 2

Eigenvalues

Number	Eigenvalue	Percent	Cum Percent
1	2.9507	22.698	22.698
2	2.0936	16.105	38.802
3	1.6329	12.560	51.363
4	1.2911	9.931	61.294
5	1.0998	8.460	69.754
6	0.9992	7.686	77.441
7	0.8142	6.263	83.704
8	0.7318	5.629	89.333
9	0.5962	4.586	93.919
10	0.3834	2.949	96.868
11	0.2487	1.913	98.782
12	0.1091	0.839	99.621
13	0.0493	0.379	100.000

Eigenvectors

	Prin1	Prin2	Prin3	Prin4	Prin5	Prin6	Prin7	Prin8	Prin9	Prin10	Prin11	Prin12	Prin13
host_since	0.03155	-0.07466	0.00256	-0.59956	-0.25768	0.22373	0.51038	0.49834	0.07492	-0.05127	0.00609	0.01960	0.00236
host_listings_count	0.31933	-0.09188	0.60836	0.10548	0.05633	0.01782	0.06384	0.01949	-0.03154	-0.03159	-0.01060	0.70569	-0.01748
host_identity_verified_t	0.51106	-0.00116	-0.28731	-0.02779	0.10138	-0.03997	-0.02272	-0.01824	0.02821	-0.37823	0.01519	0.01796	0.70595
accommodates	0.05496	0.60889	0.08055	-0.05639	0.13166	0.06157	-0.12798	0.20314	-0.01900	-0.00669	-0.73258	-0.02509	-0.00894
beds	0.04289	0.57997	0.05141	-0.10170	0.16178	0.00921	-0.23728	0.30908	-0.17129	0.08328	0.65831	0.03276	0.00093
price	-0.04705	0.42423	0.11597	-0.17457	0.07152	0.10165	0.41801	-0.64585	0.37923	-0.06422	0.13542	-0.00913	0.01508
has_availability_t	0.51224	0.00111	-0.28285	-0.02891	0.10291	-0.04857	-0.00448	-0.02690	0.04191	-0.37360	0.02989	-0.02125	-0.70739
Standardize[amenities 2]	0.21698	0.14535	-0.16137	0.30943	-0.33435	0.50079	0.25599	-0.17427	-0.56042	0.19337	-0.00246	-0.00723	-0.00400
Standardize[minimum_nights]	-0.00828	-0.16095	-0.11694	0.20874	0.53838	0.65993	-0.00934	0.18003	0.36565	0.15559	0.02914	0.01235	-0.00144
Standardize[maximum_nights]	-0.09514	0.08992	-0.10165	0.43841	0.33327	-0.41744	0.64526	0.26201	-0.09849	0.00661	0.01010	-0.00856	0.00843
Standardize[availability_90]	0.42227	-0.06753	-0.15481	-0.25884	0.14746	-0.25672	0.02356	-0.10541	0.04594	0.78820	-0.05856	0.00435	0.00493
number_of_reviews_ltm	0.16777	0.17238	-0.05562	0.42130	-0.56902	-0.06392	-0.04445	0.22561	0.59779	0.13919	0.06283	0.01032	0.00434
calculated_host_listings_count	0.32206	-0.09745	0.60634	0.08822	0.05191	0.03034	0.05455	0.05353	-0.01981	-0.01889	0.04484	-0.70615	0.02201

Because we have thirteen (13) numeric columns in the seventeen (17) columns that we finalized, we have thirteen (13) total PCAs in this dataset. We believe that we can take 9 or 10 PCAs for our modeling. We have decided to use ten (10) PCAs and see which combination will yield the best results during modeling.