How can an Airbnb host know if their listing is optimally priced? It’s not difficult to get a *general* idea, but a host knows that no two listings are exactly the same. When it comes to getting it exactly right, the best they can do is hazard a guess based on its unique blend of features. But if the price is set too low, a host will miss out on revenue; if set too high, they’ll be undercut by competitors.

To create a model that can accurately predict the price of a listing, we used a collection of Airbnb listings in Amsterdam, Netherlands. It consists of 19,619 listings with 105 features, and the target variable, “price.”

Several regression models were trained, and their performance was compared.

	Linear Regression	Decision Tree	Random Forest
MAE	33.757	35.920	32.705
MSE	2087.021	2312.952	1953.390
RMSE	45.684	48.093	44.197
R-squared	0.470	0.412	0.504
Time to Train (s)	0.170	27.470	1027.610

Some of the 105 features were unusable, and others were not suspected to be predictive of price. Indeed, the first iteration of the data that was fed to the models consisted of just 31 of the original features. Later iterations of the data were able to improve performance through feature engineering.

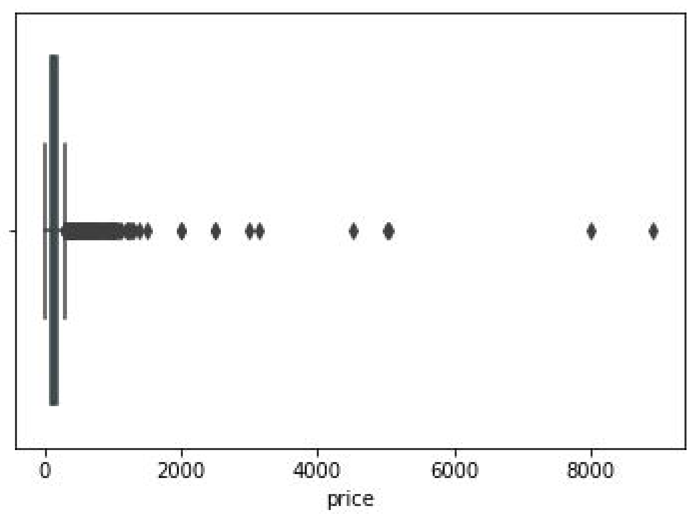
Data Preprocessing

The Target Variable

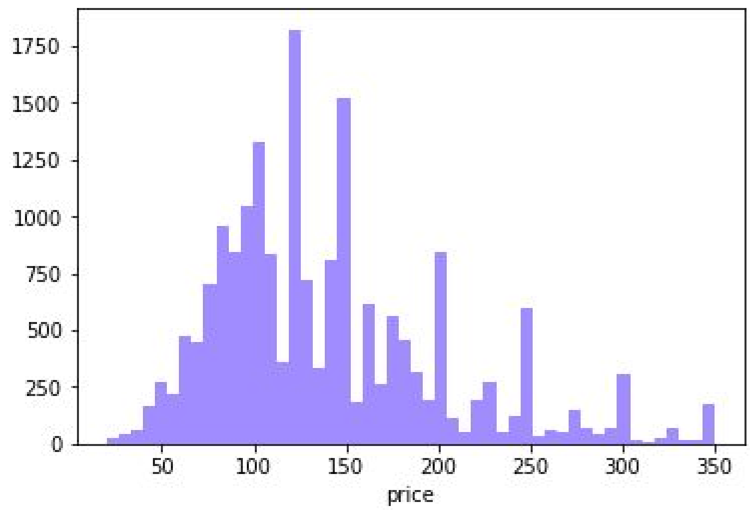
After converting the target variable to numeric format, its structure was examined. The box and whisker plot below reveals that the majority of the listings are less than $350 per night.

Anything priced higher than this should be treated with skepticism. Often, the more expensive listings have unique and novel features. For example, this Airbnb is at the top of a crane (https://www.airbnb.com/rooms/5341871). Keeping these listings in the dataset would lead to overfitting. Others list their Airbnb with a high price so that it will remain active but not rented, in an effort to gain the benefits of having active listings on the site. For these reasons, any listings priced higher than $350 were removed.

Similarly, any listings priced unusually low would not be useful for the model. Any listings priced below $20 were also removed.



Here is the price distribution with outliers removed.



Missing Data

Many features were not fit to be used in the model. Any columns that contained more than 25% null values were removed. This was in an effort to preserve as many rows as possible, considering the dataset's relatively small size (20K observations). It is possible that the absence of data could, in and of itself be predictive of price, in which case we could fill NaNs with zeros. No columns in this state immediately jump out as strong predictors of price, so they were left out of the model for now.

Text-based Features

A number of features, such as the listing's description, contain textual data that cannot be easily fed to Sci-kit Learn. It is possible that some value could be extracted from these features by identifying keywords that contribute most to price movement, but that is beyond the scope of this analysis.

Other Features Removed

Any columns that were unique row identifiers, such as "listing_url", were removed. As the data consisted entirely of listings in Amsterdam, columns like "country" and "country_code" would have no predictive power, and were thus removed. Some were left out due to their close relation to other variables, as in the case of "host_listings_count" and "host_total_listings_count". Often a boolean or categorical column would appear to be a viable, but some exploration would reveal that it was heavily skewed towards one category, and thus would not have much predictive power.

Additionally, any features that would not be relevant in analogous datasets were removed (e.g. “host_name”).

Review-based Features

Review-based features are a wildcard. Some were unusable due to their homogeneity, while others were sufficiently stratified. They were ultimately left out of the model due to null values, which made up roughly 20%. But it’s possible that these features have some predictive power. Future model improvements could explore adding these features back in. Null values would either need to be imputed, or these rows would need to be dropped altogether; reducing the size of an already small dataset. For this reason, imputing nulls is recommended.

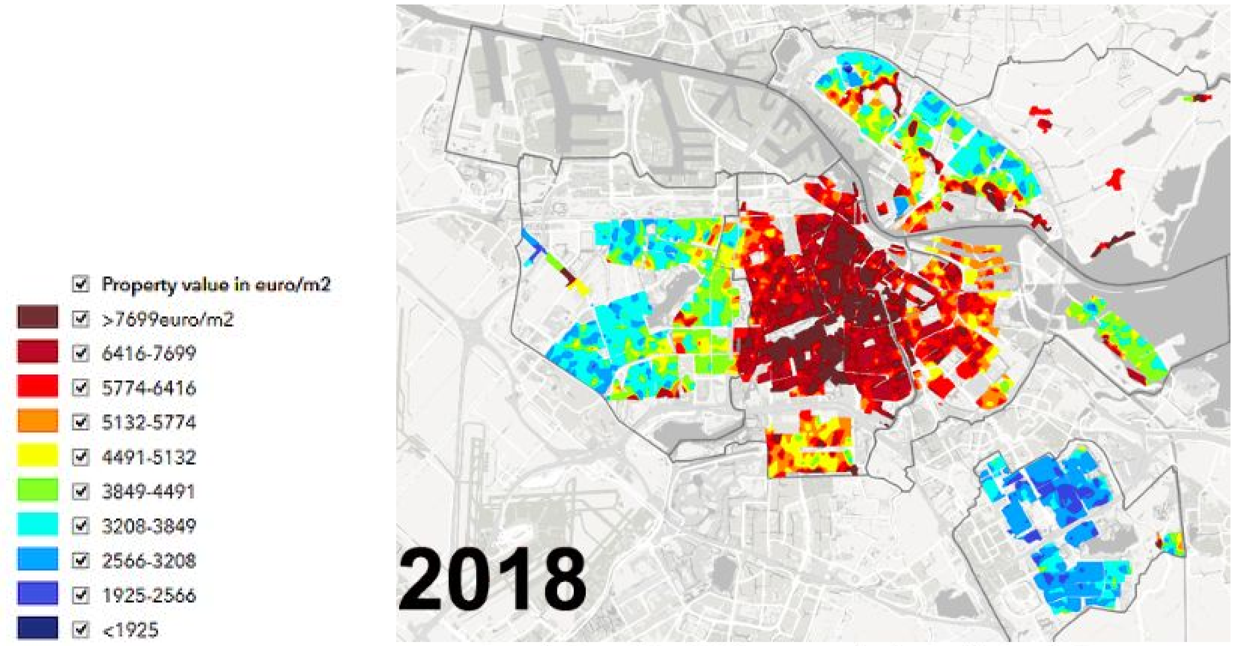
Feature Engineering

During the course of this analysis, the data underwent three iterations of feature engineering. After each iteration, model performance was recorded.

- *Iteration 1:* The feature “distance_to_city_center” created
- *Iteration 2:* Four categorical columns encoded and added to dataset
- *Iteration 3:* Amenities columns created and added to dataset

Iteration 1

Home prices generally increase as the distance to the city center decreases, and research confirmed that Amsterdam holds true to this assumption. See the graphic below for details. The distance to the city center was calculated using the latitude and longitude coordinates for each listing, which were included in the original data. The latitude and longitude of The Rijksmuseum were used as the city center. The distance between the two points (as the crow flies) was calculated using the geopy.distance.vincenty library.



Amsterdam Property Values 2014-18, source: amsterdam.nl, animated by AmsterdamTips.com

Iteration 2

Four categorical variables were identified as potentially useful features. These were one-hot encoded and added to the data. Some categories have very few observations, and thus will have little effect on the model. These can be omitted, reducing the time needed to train the model. To this end, any column with less than 100 observations was removed as a feature.

“Neighbourhood (cleansed)”

It is presumed that the neighborhood in which a listing is located has a significant impact on its price. The cleaned version of this column reduced the number of neighborhoods from 44 to 22. The number of listings in each neighborhood are listed below.

De Baarsjes - Oud-West	3288
De Pijp - Rivierenbuurt	2343
Centrum-West	2069
Centrum-Oost	1591
Westerpark	1418
Zuid	1277
Oud-Oost	1235
Bos en Lommer	1099
Oostelijk Havengebied - Indische Buurt	911
Oud-Noord	550
Watergraafsmeer	540
IJburg - Zeeburgereiland	437
Slotervaart	379
Noord-West	332
Noord-Oost	253
Buitenveldert - Zuidas	223
Geuzenveld - Slotermeer	211
Osdorp	143
De Aker - Nieuw Sloten	132
Gaasperdam - Driemond	127
Bijlmer-Centrum	105
Bijlmer-Oost	100

"Property Type"

Apartment	14579
House	1411
Townhouse	557
Bed and breakfast	539
Loft	332
Boat	310
Condominium	279
Houseboat	233
Guest suite	140
Aparthotel	108
Serviced apartment	52
Guesthouse	43
Other	42
Boutique hotel	35
Villa	28
Cottage	12
Bungalow	11
Cabin	11
Tiny house	9
Hotel	8
Hostel	7
Casa particular (Cuba)	5
Chalet	3
Camper/RV	2
Campsite	2
Earth house	1
Dome house	1
Island	1
Tent	1
Nature lodge	1

“Room Type”

Entire home/apt	14706
Private room	4002
Shared room	55

“Cancellation Policy”

strict_14_with_grace_period	7253
moderate	7017
flexible	4451
super_strict_60	24
super_strict_30	18

Iteration 3

The “amenities” column contained a list of all the amenities offered at each Airbnb. The dataset was found to contain 124 unique amenities. Of those, 14 were identified as potentially strong indicators of price. Boolean columns were created for each of the 14 amenities such that if an amenity was offered at an Airbnb, that column would have the value “True”. The list of amenities is below.

- Indoor fireplace
- Long term stays allowed
- Dishwasher
- Washer
- Private entrance
- TV
- Pets allowed
- Hot tub
- Beachfront
- Bathtub
- Patio or balcony
- Waterfront
- Cable TV (or Satellite TV in US)
- Family/kid friendly

Machine Learning

Model Selection and Evaluation

In comparing model performance, three common algorithms for regression problems were tested; linear regression, decision tree, and random forest. A neural network was determined not necessary due to the data’s small size, and relatively low complexity. The models were scored based on both mean absolute error (MAE) and root mean squared error (RMSE). R-squared was also calculated for each model. In each iteration of the features, the random forest performed the best on both MAE, RMSE, and R-squared. However in the final iteration, the difference in both MAE and RMSE between linear regression and the random forest was very small. Only R-squared was notably better in the random forest. Considering it took the random forest nearly 10^4 times as long to train as linear regression, there is a strong argument that linear regression is the model of choice for this problem. It is worth noting that the linear regression model improved, even as the complexity of the data increased in later iterations.

Evaluation of Individual Predictions

About 58% of predictions were within $30 of the actual price. Before this model can be used in production, it is suggested that at least 90% of predictions fall within this range.

Feature Importances

In comparing the feature importances from the random forest model to the coefficients from the linear regression model, we do see some overlap, as well as some notable differences. Our created column "distance_to_city_center" was the second most important feature in the random forest, but in linear regression it was the fifth strongest overall feature.

Random Forest

	feature	importance
5	accommodates	0.259204
29	distance_to_city_center	0.112423
0	host_since	0.054805
62	Entire home/apt	0.049246
20	availability_90	0.042724
22	number_of_reviews	0.038348
21	availability_365	0.036788
9	extra_people	0.033600
7	bedrooms	0.029579
6	bathrooms	0.028326

Linear Regression (positive)

	feature	coefficient
62	Entire home/apt	76.310771
63	Private room	44.397271
39	Gaasperdam - Driemond	37.360949
41	IJburg - Zeeburgereiland	30.590906
67	indoor_fireplace	19.299585
59	Houseboat	18.516507
31	Bijlmer-Oost	15.596112
5	accommodates	15.507844
7	bedrooms	14.891070
35	Centrum-West	14.615807

Linear Regression (negative)

	feature	coefficient
29	distance_to_city_center	-19.520960
48	Slotervaart	-17.192927
53	Apartment	-15.274404
32	Bos en Lommer	-15.022482
56	Condominium	-14.006873
38	De Pijp - Rivierenbuurt	-12.788246
37	De Baarsjes - Oud-West	-12.624574
75	beachfront	-12.322881
33	Buitenveldert - Zuidas	-11.742051
47	Oud-Oost	-10.777648

Suggestions for Future Analysis

The initial data fed to Sci-kit Learn was able to do a reasonable job predicting Airbnb prices. Several iterations of feature engineering led to marginal improvements. Further research could revisit columns that were initially dropped, but may hold value. It is worth noting that in this particular dataset, most Airbnbs did not have square footage listed, which is why it was not used in this analysis. It is likely that other datasets will have this useful piece of information. The data on host ratings may also be useful. From an intuitive standpoint, it would not be surprising if a highly-rated host with many reviews could list their Airbnb at a higher price. After determining the best way to address null values, these columns can be added to the model. One final point is that there are a number of columns that contain textual data. It may be possible to extract commonalities among words that are most associated with price movement.