PREDICTING AIRBNB LISTING PRICES IN ISTANBUL USING MACHINE LEARNING AND SENTIMENT ANALYSIS

MÜGE KESER

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
AT THE SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
OF TILBURG UNIVERSITY

STUDENT NUMBER

2028837

COMMITTEE

dr. M. Jung
dr. Y. Satsangi
dr. D. Shterionov

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

January 14, 2022

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisors dr. Satsangi and dr. Jung for their guidance and valuable feedback. In addition, I would like to thank dr. Shterionov for providing valuable feedback.

PREDICTING AIRBNB LISTING PRICES IN ISTANBUL USING MACHINE LEARNING AND SENTIMENT ANALYSIS

MÜGE KESER

Word count: 7261

Abstract

Airbnb is an online platform on which people can offer or book listings (accommodations). Listings are offered by hosts, and they are responsible for determining the price for their listing. It is important for hosts to correctly determine the listing price because 'price' is a key factor in the decision-making process of consumers. In addition, Airbnb does not control the prices determined by the host. This thesis aims to predict the prices of Airbnb listings in Istanbul using machine learning and natural language processing to provide hosts with a model to determine the price for the listing, and potential (guests) can use this model to determine if the indicated price by the host is fair. Different machine learning models were built to predict the Airbnb listing prices in Istanbul, namely Linear Regression, Random Forest Regression, Support Vector Regression, and XGBoost (eXtreme Gradient Boosting) Regression. As input for the models, guest reviews and various listing-related features were used. The listings characteristics features and reviews are stored in separate data sets, therefore two different data sets were merged to make the predictions. Both data sets are publicly available on Inside Airbnb. Moreover, this study examines the generalizability of the models used for predicting Airbnb listing prices in Istanbul by testing them on two new cities, which are Amsterdam and Rome. For this purpose, a Listings data set and a Reviews data set will be used for both cities. The results show that the XGBoost Regression model predicts the Airbnb listing prices in Istanbul the best. Additionally, the results show that none of the models were able to predict the prices of Airbnb listings in Amsterdam and Rome. This means that the models for Airbnb listings in Istanbul are not generalizable to predict Airbnb listing prices in other cities - Amsterdam and Rome.

1 DATA SOURCE/CODE/ETHICS STATEMENT

Work on this thesis did not involve collecting data from human participants or animals. The data sets used in this thesis were retrieved from Inside Airbnb[1] and are publicly available. The original owner of the data and code used in this thesis retains ownership of the data and code during and after the completion of this thesis. The author of this thesis acknowledges that she does not have any legal claim to this data or code. The code related to this thesis is available as a Python file (.py) on GitHub.[2]

2 INTRODUCTION

Airbnb, established in 2008, is an online platform on which members can publish, offer, search for, or book listings.[3] Listings refer to the different accommodations that are offered, for example, homes, apartments, studios, and villas. Since the launch of the platform, Airbnb has more than 5.6 million listings located in more than 220 countries, including Turkey, and more than one billion guest arrivals from all over the world.[4] In Turkey, the use of Airbnb listings is popular in cities like Istanbul, Antalya, and Fethiye.[5] Once a guest has booked a listing, stayed, and checked out, the guest will be asked to leave a review. This review must be submitted within 14 days from the time of checkout, and these reviews will then be visible on the Airbnb platform.[6] With this opportunity, guests can express their opinion about the listing. Reviews are valuable for customers, and they use reviews in their consumer decision-making process. In addition, reviews are the most important information source that travelers use to make a purchase decision. Furthermore, price is an important factor in the consumer decision-making process, and reviews are supposed to help with evaluating the price of a product (Wang, Fong, & Law, 2020).

Currently, hosts determine the prices for their listings, and Airbnb does not control the prices (Priambodo & Sihabuddin, 2020). However, it is important for Airbnb to have a model that can help hosts indicate the price for a listing. Another benefit could be that this model helps (potential) guests to evaluate if the indicated price by the host is fair. More importantly, there is no standard model that can be used to predict the prices of Airbnb listings in the world. Therefore, this study will focus on developing different machine learning models for Airbnb listings in

[1] http://insideairbnb.com/get-the-data.html
[2] https://github.com/ksr-m
[3] https://www.airbnb.nl/help/article/2908/terms
[4] https://news.airbnb.com/about-us/
[5] https://www.airbnb.com/turkey/stays
[6] https://www.airbnb.com/help/article/13/reviews-for-stays

Istanbul and assess if these models generalize to two other cities in Europe. The aim of this study is to answer the main research question:

> *How can Airbnb listing prices in Istanbul be predicted based on listing characteristics and reviews?*

In order to answer the main research question four sub-questions were formulated:

SQ1 *What is the compound polarity score of each reviews?*

In order to use reviews for predicting the prices of Airbnb listings in Istanbul, sentiment analysis will be performed. By doing so, the compound score of each guest review will be obtained. Accordingly, a new feature will be created. To obtain the compound score for each review a lexicon-and-rule-based sentiment analysis tool, VADER (Valence Aware Dictionary and sEntiment Reasoner), will be used (Hutto & Gilbert, 2014). This tool calculates four different sentiment polarity scores, namely negative polarity score, neutral polarity score, positive polarity score, and compound polarity score, for each review. The compound polarity score is the normalized sum of the negative, neutral, and positive sentiment classes (Hutto & Gilbert, 2014). A further description of VADER will be provided in Sub-section 5.3.

SQ2 *Which features affect the price of Airbnb listings in Istanbul the most?*

To determine which features influence the price of Airbnb listings the most, the 'SelectKBest' method from the Scikit-learn library will be used (Pedregosa et al., 2011). This method will be explained in Sub-section 5.4.

SQ3 *Which model predicts the price of Airbnb listings in Istanbul the best?*

To answer the third sub-question several steps will be taken. First of all, four different machine learning models, namely Linear Regression (baseline model), Random Forest Regression, Support Vector Regression, and XGBoost Regression will be built to predict the prices of Airbnb listings in Istanbul. Subsequently, the prediction performance of each model will be evaluated on the test set by using three evaluations metrics, including the R-Squared (R^2) value, Mean Absolute Error (MAE), and Root Mean Square Error (RMSE). The model with the highest R^2 value and lowest MAE and RMSE values will be considered as the best model. The different evaluation metrics will be described in Sub-section 5.7.

SQ4 *How well do the models generalize to other cities?*

The fourth sub-question will be answered by testing the different models built for predicting Airbnb listings in Istanbul on unseen data. For this purpose, Airbnb data for Amsterdam and Rome will be used.

This study shows that the XGBoost Regressor is the best model to predict Airbnb listing prices in Istanbul using listing characteristics and reviews. The second-best performance is obtained with the Random Forest Regressor. This model performs slightly worse than the XGBoost Regression model. Moreover, the Support Vector Regression model is the worst performing model. In addition, the results show that the models built for predicting Airbnb listing prices in Istanbul do not generalize to other cities - Amsterdam and Rome.

3 RELATED WORK

Sentiment Analysis is a widely performed task in Natural Language Processing (NLP) and is used to extract and analyze people's opinions about a topic (Birjali, Kasri, & Beni-Hssane, 2021; Luo, Chen, Xu, & Zhou, 2013; Yue, Chen, Li, Zuo, & Yin, 2019). By applying sentiment analysis, a polarity score is obtained and used to classify a particular text into the positive class, neutral class, or negative class (Borg & Boldt, 2020; Haselmayer & Jenny, 2017; Hutto & Gilbert, 2014). For sentiment analysis, the compound polarity score is commonly used to classify a text into one of the three classes (Hutto & Gilbert, 2014). Sentiment Analysis is applied in different fields (Mouthami, Devi, & Bhaskaran, 2013). One of these field is business, where companies use sentiment analysis as a quality measure for their products or services through social media monitoring (Lawani, Reed, Mark, & Zheng, 2019; Mouthami et al., 2013).

Moreover, another common task is to make predictions by using sentiment analysis. One example is from the field of politics, where sentiment analysis is commonly used to predict elections based on social media posts or news articles (Ibrahim, Abdillah, Wicaksono, & Adriani, 2015; Ramteke, Shah, Godhia, & Shaikh, 2016). To give a specific example, (Rodríguez-Ibáñez, Gimeno-Blanes, Cuenca-Jiménez, Soguero-Ruiz, & Rojo-Álvarez, 2021) applied sentiment analysis on political campaigns based on tweets posted by the political party leader or tweets that contain a mention to a political party in order to characterize the sentiment during an election period. Furthermore, a study conducted by Caetano, Lima, Santos, and Marques-Neto (2018) performed sentiment analysis on tweets and classified Twitter users into six classes.

Another example is from the field of finance, where an extensively performed task is to predict stock prices or cryptocurrency prices based on messages posted on social media platforms, like Reddit and Twitter (Inamdar, Bhagtani, Bhatt, & Shetty, 2019; Jin, Yang, & Liu, 2020; Mohan, Mullapudi, Sammeta, Vijayvergia, & Anastasiu, 2019; Sattarov, Jeon, Oh, & Lee, 2020; Wooley, Edmonds, Bagavathi, & Krishnan, 2019). In this field, sentiment analysis is used to extract whether people's sentiment is positive or negative about, for example, a stock or cryptocurrency to predict the price of a stock or cryptocurrency (Renault, 2020).

Moreover, sentiment analysis is frequently applied to derive the opinions and sentiments of customers towards services or products (Lawani et al., 2019; Ligthart, Catal, & Tekinerdogan, 2021). The importance of guests' reviews on Airbnb listings prices is examined in by Lawani et al. (2019). To do so, they used sentiment analysis to extract the review score of each guest review for Airbnb listings in Boston. This study showed that review scores highly influence the price of an Airbnb listing (Lawani et al., 2019).

In addition, Kokasih and Paramita (2020) applied XGBoost Regression to predict the Airbnb listing prices in Singapore. They used listing features, location data, reviews, and data about the host as the input for the model (Kokasih & Paramita, 2020). In this study, NLP is used to extract polarity scores from reviews to use these with other features. This study showed that XGBoost Regressor is a powerful algorithm to predict Airbnb prices Kokasih and Paramita (2020). Furthermore, Zhu, Li, and Xie (2020) used different machine learning models, namely Linear Regression (baseline model), Deep Neural Network, Random Forest Regressor, and XGBoost Regressor to predict the Airbnb listing prices in New York in combination with NLP. The NLP algorithm 'sentimentr' was used to calculate the sentiment score based on the listing name and used it along with other features from the Airbnb data set as input for the models (Zhu et al., 2020). They used the NLP algorithm 'sentimentr'. In this study, the Random Forest Regressor and XGBoost Regressor achieved the best performance.

Another study that predicts prices of Airbnb listings in New York was conducted by Rezazadeh Kalehbasti, Nikolenko, and Rezaei (2021). They also used machine learning models in combination with NLP. In contrast to Zhu et al. (2020), this study performed sentiment analysis on guest reviews for each Airbnb listing to extract the sentiment polarity score. This study used the TextBlob library to perform sentiment analysis and obtain the polarity score of each review. Subsequently, they used this sentiment polarity score combined with listing features from the Airbnb data set (Rezazadeh Kalehbasti et al., 2021). They applied a Linear Regression model (baseline model), Ridge Regression, K-means Clustering with Ridge Regression, Support Vector Regression, Neural Network, and

Gradient Boosting Tree Ensemble to predict the prices. In this study, the Support Vector Regression model outperformed the other models (Rezazadeh Kalehbasti et al., 2021).

In addition, numerous studies predict the prices of Airbnb listings without performing sentiment analysis. Yang (2021) used an XGBoost Regression model and a Neural Network to predict Airbnb listing prices in Beijing. This study used listings characteristics and created two additional features, namely Beijing house prices and location data of subway stations as input for the models (Yang, 2021). Among the two models, the XGBoost Regression achieved the best result (Yang, 2021). Furthermore, Dhillon et al. (2021) used a Linear regression, Logistic regression, and Random Forest Regression to predict Airbnb prices across different cities in the United States. As the input to the models, several features from the Airbnb data set were used, and they achieved the best performance with the Random Forest Regression model (Dhillon et al., 2021).

The contribution of this study to the existing literature is twofold. The first contribution is building a model to predict the prices of Airbnb listings in Istanbul using machine learning and natural language processing since this is not observed in the existing literature. Second, examining the generalizability of the models for this particular city was not observed. Therefore this study examines the generalizability of the models for Airbnb listing prices in Istanbul on two other cities, namely Amsterdam and Rome.

4 METHOD

Supervised learning algorithms are one category of machine learning algorithms. Supervised learning algorithms require labeled data. In other words, the dependent variable is known (Shmueli, Bruce, Gedeck, & Patel, 2019). In this study, the outcome variable is known, therefore supervised learning algorithms were used to predict the prices of Airbnb listings in Istanbul. From previous studies conducted by Rezazadeh Kalehbasti et al. (2021), Kokasih and Paramita (2020), Zhu et al. (2020), Yang (2021), and Dhillon et al. (2021), it was observed that the XGBoost Regressor, Support Vector Regressor, and Random Forrest Regressor algorithms achieve the best performance in predicting Airbnb listing prices. Based on these findings, the same models were used in this study. In addition, a Linear Regression model was used as the baseline model since this machine learning algorithm is simple and widely used for prediction tasks (Maulud & Abdulazeez, 2020).

4.1 *Linear Regression*

Linear regression is used for supervised learning tasks. A linear regression model fits a linear model and has the objective to minimize the sum of squared errors between the actual dependent variable and the predicted dependent variable (Shmueli et al., 2019). Moreover, multiple linear regression is used when there are two or more independent variables to predict one dependent variable (Uyanık & Güler, 2013).

4.2 *Random Forest*

Another supervised learning algorithm is the Random Forest. This algorithm is proposed by Breiman (2001). The Random Forest is suitable for classification tasks and regression tasks (Breiman, 2001). This algorithm is an ensemble machine learning method that combines different decision trees by using a random feature subset from the data set (Biau, 2012; Biau & Scornet, 2016). The model outputs the majority class, in the case of a classification task. In the case of a regression problem, the output is an average of the predictions found by each tree (Biau & Scornet, 2016).

4.3 *Support Vector Regressor*

The Support Vector Machine is a supervised machine learning algorithm. The Support Vector Machine is used for classification tasks. However, it is also possible to implement this algorithm for a regression task by using a Support Vector Regression (Awad & Khanna, 2015). This algorithm uses thresholds values to fit the hyperplane (best fit line) (Awad & Khanna, 2015).

4.4 *XGBoost Regressor*

The XGBoost (eXtreme Gradient Boosting) is a supervised learning algorithm. This algorithm is a tree-based ensemble method and the purpose of this algorithm is to minimize the objective function during each iteration. The objective function is defined by the loss function and regularization term (Chen & Guestrin, 2016; Cherif & Kortebi, 2019).

5 EXPERIMENTAL SETUP

5.1 *Data set*

The data sets used in this study were obtained from Inside Airbnb[7]. Inside Airbnb provides separate data sets for Airbnb listings in different cities and are publicly available in comma-separated values (CSV) formatted files. In this study, the Listings data set and Reviews data set for Airbnb listings in Istanbul, Amsterdam, and Rome were used. The data sets for Istanbul (compiled on 30 September 2021) were the main data sets. Furthermore, the Amsterdam data sets (compiled on 7 September 2021) and Rome data sets (compiled on 12 September 2021) were used solely to test the generalization performance of the price prediction models for Airbnb listings in Istanbul. A further description of the Listings data set and Reviews data set, including the pre-processing and Exploratory Data Analysis (EDA), will be provided in Sub-section 5.2 and Sub-section 5.3, respectively.

5.2 *Listings data set*

The Listings data set contains detailed information about Airbnb listings, including the target variable which is 'Price'. All Listings data sets have the same 74 features, which can be found in Appendix A (page 27). However, the number of observations is different per data set, as the number of Airbnb listings offered varies by city. The number of listings in the data sets for Istanbul, Amsterdam, and Rome is 23,019, 16,116, and 26,097, respectively. After obtaining the data sets, EDA and pre-processing were performed for all data sets. These steps were the same for each data set. The first step was to remove columns from the data set that were not relevant for analysis, for example, 'Listing_url', 'Last_scaped', and 'Picture_url'. Furthermore, there were features that contained similar information, for example, 'host_listings_count' and host_total_listings_count. Therefore, one of the two features was removed. Moreover, several numerical features were converted from object to float by removing the text, for example, the dollar symbol was removed from the 'Price' feature (target variable) or the percentage sign was removed from the 'Host_response_rate' feature.

The second step was to determine whether there were missing values in the data set. It was observed that several columns contained missing values. From these columns, four contained only missing values and were removed from the data set. Moreover, data imputation was used to replace the missing values in numerical columns. There are two commonly used

[7] http://insideairbnb.com/get-the-data.html

imputation techniques, namely, mean imputation and median imputation (Hadeed, O'Rourke, Burgess, Harris, & Canales, 2020). Mean and median imputation means replacing the missing values in a particular feature with the mean value or median value of that feature (Jadhav, Pramod, & Ramanathan, 2019). To decide which imputation method to use the distribution of the numerical variables was analyzed by using a distribution plot. From this, it was observed that the features are highly skewed. Therefore, median imputation was used since this method is suitable for highly skewed data (Hadeed et al., 2020). Moreover, Boolean variables were observed and converted into binary variables. An example of a Boolean variable from the data set is the 'Host_is_superhost' feature which contains only 'f' (False) or 't' (True) values. In addition, categorical variables were one-hot-encoded, except the 'amenities' column. This column was used to create a new feature ('Nr_amenenities') by counting the number of amenities. Furthermore, after one-hot-encoding the 'property_type', feature it was found that not all data sets contain the same property types. Therefore, the property types that were not present in all data sets were removed. Subsequently, the original features which were one-hot-encoded and the 'amenities' feature were removed.

The final pre-processing step for the Listings data set was applying log-transformation on features with a right-skewed distribution to make the distributions more normal (Curran-Everett, 2018). One of the features on which log-transformation was applied is the 'Price' feature (target variable). Figure 1 illustrates the distribution of the 'Price' feature before and after applying log-transformation. It is shown that the distributions became more normal after applying log-transformation.

5.3 *Reviews data set*

The Reviews data set contains detailed information about guest reviews for Airbnb listings. All data sets have the same six features. An overview of all features can be found in Appendix C (page 30). The number of observations is different per data set, as these correspond to the number of reviews. The Istanbul data set contains 216,729 reviews, the Amsterdam data set contains 397,185 reviews, and the Rome data set contains 1,044,497 reviews.

After obtaining the data sets EDA, and pre-processing were performed for each data set. The first step was to remove unnecessary features since only the 'Listing_id' and 'Comments' were relevant for this study. The 'Listing_id' feature includes the unique ID of an Airbnb listing, and the 'Comments' feature includes the guest reviews. The second step was to detect missing values. As a result, missing values were identified for the

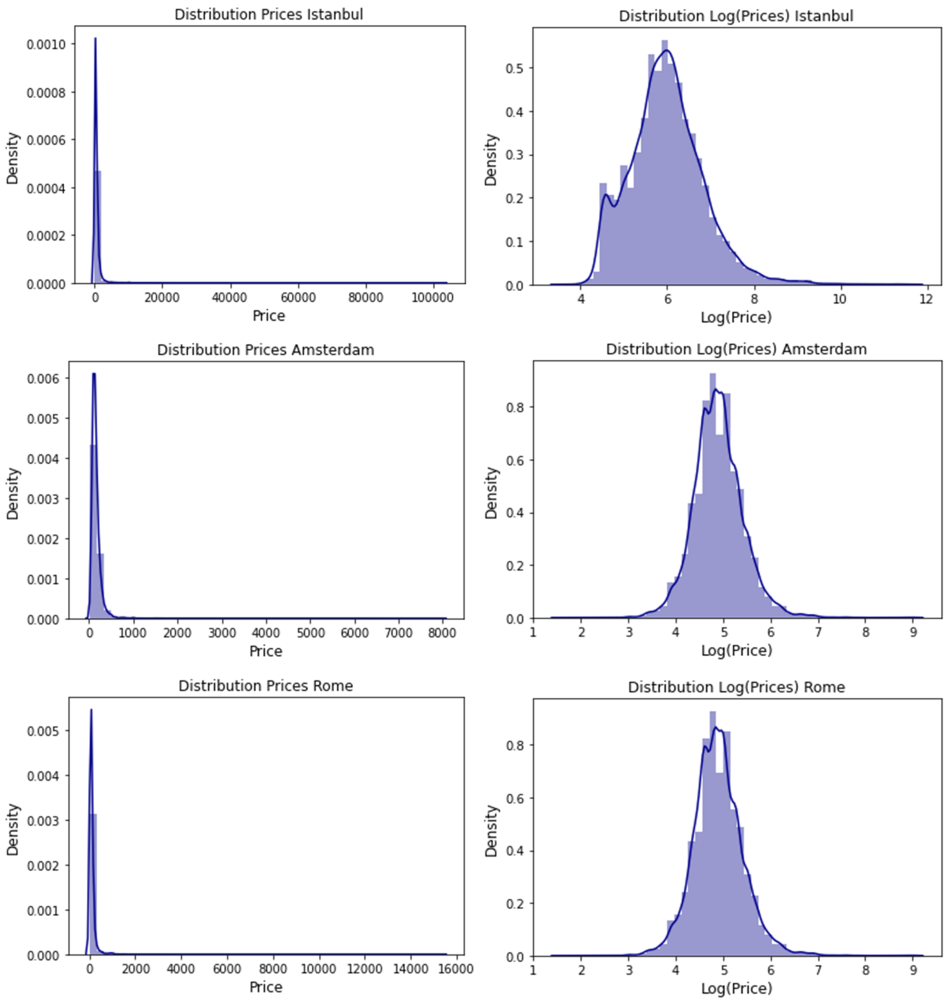

Figure 1: Distribution of the 'Price' feature of the Istanbul, Amsterdam, and Rome data sets before log-transformation and after applying log-transformation.

'Comments' feature in all data sets. The Istanbul Reviews data set contained 409 missing values which is 0.19% of the entire data set. Furthermore, the Amsterdam Reviews data set had 289 missing values which represents 0.07% of the observations. In the Rome data set, 412 missing values, were identified which amounts to 0.04% of the total observation. The missing values were removed from the data sets since the number of missing values is small and the data sets are relatively large (Cokluk & Kayri, 2011). In addition, automatically posted reviews were removed since these reviews were not useful for analysis. Consequently, 1,543 observations from the Istanbul data set, 5,558 observations from the Amsterdam data set, and 5,401 observations from the Rome data set were removed. Furthermore, the reviews in the 'Comments' feature were written in different languages,

for example, Turkish, English, French, and Russian. However, only reviews written in English were used for analysis. In order to detect these reviews, the Python package 'Langdetect' was used (Danilák, 2021). After applying this tool, 149,423 English reviews were detected in the Istanbul Reviews data set, 307,168 English reviews were identified in the Amsterdam Reviews data set, and 684,656 English reviews were detected in the Rome Reviews data set. The final step of this process was to store the detected English reviews. As a result, a data set with only English reviews were created for each city.

Subsequently, the data sets containing only the English reviews were used to create a new feature for each Listings data set. This new feature was created by extracting the compound polarity score of each guest review by using VADER (Hutto & Gilbert, 2014). VADER was used as the sentiment analysis tool since it performs well on social media text or reviews (Bonta & Janardhan, 2019; Hutto & Gilbert, 2014). As mentioned earlier, VADER calculates different sentiment scores, including the compound polarity score. This score is the normalized sum of valence scores and ranges from -1 to +1, where -1 means extremely negative and +1 means extremely positive (Hutto & Gilbert, 2014). A compound polarity score of >= 0.05 indicates a positive sentiment, a compound polarity score between > -0.05 and < 0.05 indicates a neutral sentiment, and a sentence has a negative sentiment when the compound polarity score is <=-0.05. Furthermore, listings can have multiple reviews. Therefore, the average compound polarity score is computed per listing and used as a new feature which was called 'Compound_score'.

As a final step, the Listings data set and Reviews data set for each city were merged based on listing ID. After merging the data sets, missing values were observed in the new feature 'Compound_score'. These missing values can be explained by the fact that there are no guest reviews for these listings. As mentioned earlier, a compound polarity score between > -0.05 and < 0.05 indicates a neutral sentiment, therefore, the missing values were replaced with zeros (Hutto & Gilbert, 2014).

By performing all steps, one data set containing the features from the Listings data set and the new feature 'Compound_score' was obtained for each city.

5.4 *Feature selection*

Another important part of pre-processing is feature selection and is used to determine the (minimum) input features of a model Kuhn and Johnson (2013). As mentioned earlier, the Istanbul data set was considered the main data set. Therefore, feature selection was performed based on the

Istanbul data set. To identify the most important features for predicting Airbnb listings prices in Istanbul the 'SelectKBest' method from Scikit-learn was used (Pedregosa et al., 2011). This method calculates the correlation between the dependent variables and the independent variable. Subsequently, a p value for each variable was obtained (Pedregosa et al., 2011). After applying this method to the Istanbul data set, it was observed that features related to property, such as 'property type' and features related to the room, for example, 'room type' highly influence the price of an Airbnb listing in Istanbul. Table 1 shows the 10 most important features for predicting Airbnb listing prices in Istanbul. The newly created feature 'Compound_score' is listed on place 12 and has a p value of .044. It can be concluded that reviews influence the price of Airbnb listings in Istanbul. Another feature created during the pre-processing was 'Nr_amenities' this feature has a moderate influence on the price of Airbnb listings in Istanbul since it was listed on place 30 ($\mathbf{p}$ value = .017).

In addition, the 'SelectKBest' was also used to determine the appropriate number of features to predict Airbnb listing prices in Istanbul. This method selects the top features based on the p value for a given k, where k represents the number of features to select (Pedregosa et al., 2011). In this study different values for k were determined, namely $k = 10$, $k = 15$, $k = 20$, $k = 25$, $k = 30$, $k = 35$, $k = 40$, $k = 45$, $k = 50$, $k = 55$, and $k = 60$. Subsequently, each set of k features was used to predict the prices of Airbnb listings in Istanbul. After evaluating the model performances, it was observed that the optimal number of features is $k = 60$. Appendix D (page 31) includes the 60 selected features for predicting prices of Airbnb listings in Istanbul. In addition, the same features were selected for the Amsterdam data set and Rome data set. As a result, the final size of the Istanbul data set is 23,006 observations and 60 features, the Amsterdam data set contains 16,096 observations and 60 features, and the Rome data set includes 26,068 observations and 60 features.

In Machine Learning data is commonly split into a training set, validation set, and test set (Kuhn & Johnson, 2013). The training set is used for training the models, the validation set is used for tuning the hyperparameters of a model, and the test set is used to evaluate the performance of the models on unseen data (Kuhn & Johnson, 2013). Therefore, the last pre-processing step involved splitting the Istanbul data set (23,006 observations and 60 features) into a training set, validation set, and test set, 70%, 15%, and 15%, respectively. The Amsterdam and Rome data sets were entirely used (100%) as test sets to assess the generalizability of the models built for predicting Airbnb listing prices in Istanbul.

Table 1: The first 10 most important features for predicting the price of Airbnb listings in Istanbul.

Feature	Score (p value)
Accommodates	.204
Entire vila	.175
Bedrooms	.117
Beds	.097
Bathrooms	.095
Room type: Private room	.087
Room type: Entire home/apt	.084
Private room in rental unit	.074
Private room in boat	.052
Boat	.045

5.5 *Hyperparamater tuning*

Hyperparameter tuning is an important task and is used to find the optimal set of hyperparameters for a model (Kuhn & Johnson, 2013). In this study, hyperparameter tuning was performed for all models except the Linear regression model (baseline model) because this model has no hyperparameters to tune (Kuhn & Johnson, 2013). The hyperparameter tuning process for the XGBoost Regression model, Random Forest Regression model, and Support Vector Regression model started by defining a set of hyperparameters to tune for each model. First, the set of hyperparameters to tune was defined for the XGBoost Regression model. The defined set included four hyperparameters, namely the 'n_estimators' which is the number of gradient boosted trees or the number of boosting rounds, 'max_depth' which defines the maximum depth of the tree, 'learning_rate' which is the boosting learning rate, and 'gamma' which is a regularization parameter (Pedregosa et al., 2011). Moreover, for the Random Forest Regression model, the maximum depth of the tree ('max_depth') and the number of trees in the forest ('n_estimators') were the selected hyperparameters to tune (Pedregosa et al., 2011). The last set of hyperparameters was defined for the Support Vector Regression model and included two hyperparameters, namely 'C' which is the regularization parameter, and 'gamma' which defines the kernel coefficient (Pedregosa et al., 2011). Subsequently, the validation set was used to find the optimal set of hyperparameters by using grid search which was implemented by a for-loop. Table 2 shows the defined set of hyperparameters and the best set of hyperparameters found for each model. After finding the best set of hyperparameters for each model, these were used to build the final models.

Table 2: Selected hyperparameters and optimal parameters obtained per model

Model	Defined set of hyperparameters	Best set of hyperparameters
XGBoost Regressor	n_estimators = [50, 100, 150] max_depth = [1, 5, 10, 15, 20] learning_rate = [0.1, 0.2, 0.3] gamma = [0.1, 0.5, 1]	n_estimators = 150 max_depth = 10 learning_rate = 0.1 gamma = 0.5
Random Forest Regressor	max_depth = [1, 5, 10, 15, 20] n_estimators = [100, 150]	max_dept = 20 n_estimators = 100
Support Vector Regressor	gamma = [0.001, 0.01] C = [30, 40, 50]	gamma = 0.001 C = 50

5.6 *Implementation*

In this study, the experiments were performed in Python (version 3.7) by using various packages. An overview of all packages and versions used for each task can be found in Table 3.

Table 3: Python packages and versions used per task

Task	Python package	Version
Pre-processing	Pandas (Wes McKinney, 2010)	1.1.5
	NumPy (Harris et al., 2020)	1.19.5
Sentiment analysis		
Language detection	Langdetect (Danilák, 2021)	1.0.9
Compound polarity score	VADER (Hutto & Gilbert, 2014)	3.3.2
Feature selection	Scikit-learn (Pedregosa et al., 2011)	1.0.1
Data partitioning	Scikit-learn (Pedregosa et al., 2011)	1.0.1
Model building	Scikit-learn (Pedregosa et al., 2011)	1.0.1
Model evaluation	Scikit-learn (Pedregosa et al., 2011)	1.0.1
Visualization	Seaborn (Waskom, 2021)	0.11.2
	Matplotlib (Hunter, 2007)	3.2.2

5.7 *Evaluation metrics*

R^2, MAE and RMSE are the most commonly used metrics for regression tasks (Chicco, Warrens, & Jurman, 2021; Kuhn & Johnson, 2013; Li, 2017). Therefore, these metrics were used to assess the prediction performance of each model based on the test set. R^2 indicates the fraction of variance in

the dependent variable that is explained by the independent variables in the model Kuhn and Johnson (2013). For example, if a model has an R^2 value of 0.40 this implies that the model explains 40% of the variance in the dependent variable given the independent variables in the model. For this metric high values means a better fit Chicco et al. (2021). Moreover, the MAE measures the average absolute errors between the actual value and the predicted value (Willmott & Matsuura, 2005). Furthermore, the square root of the errors between the predicted values and actual values is represented by the RMSE (Hunter, 2007; Willmott & Matsuura, 2005). In terms of the MAE, the best value for these metrics is 0 (zero) and the worst value is $+\infty$, therefore lower values are desirable (Chicco et al., 2021). The R^2, MAE, and RMSE are defined as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{1}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

where y_i is the actual value of the i^{th} observation, $\hat{y}_i$ is the predicted value for the i^{th} observation, $\bar{y}$ denotes the mean of y, and n represents the sample size.

6 RESULTS

In this section, the prediction performance of the models described in Section 4 on the Istanbul data set will be presented. In addition, the generalizability of these models on the Amsterdam data set and Rome data set will be presented. The models were compared in terms of the evaluation metrics described in 5.7.

6.1 *Prediction performance*

Several models were used to predict Airbnb listings prices in Istanbul. The predictive performance per model is illustrated in Table 4.

The baseline model (Linear Regression) achieved an R^2 score of 0.461, the MAE and RMSE values of this model are 0.456 and 0.616, respectively. By comparing the different models, it was observed that all models outperformed the baseline model. The highest predictive performance is achieved with the XGBoost Regressor since this model has the highest R^2 score

and the lowest MAE and RMSE values which are 0.570, 0.395, and 0.550, respectively. The Random Forest Regressor is the second best algorithm with an R^2 score of 0.546, MAE value of 0.408, and an RMSE value of 0.566. However, by comparing the XGBoost Regressor and Random Forest Regressor, it was observed that there is a small difference in terms of predictive power. The R^2 score, an MAE value, and RMSE value obtained with the Support Vector Regressor are 0.465, 0.438, and 0.614, respectively. With this performance, the Support Vector Regressor performs the worst, compared to the baseline model. The result of the Support Vector Regressor is notable since it shows similar results as the baseline model. It can be concluded that the Random Forest Regressor and XGBoost Regressor performed considerably better than the baseline model and the Support Vector Regressor.

Table 4: Performance metrics for each model on the test set

Model	R^2	MAE	RMSE
Linear Regression (baseline)	0.461	0.456	0.616
Random Forest Regressor	0.546	0.408	0.566
Support Vector Regressor	0.465	0.438	0.614
XGBoost Regressor	**0.570**	**0.395**	**0.550**

Figure 2 illustrates the actual values and predicted values per model for Airbnb listings in Istanbul.

6.2 *Generalizability of the models to other cities*

After predicting the Airbnb listings prices for Istanbul, the generalizability of the models was evaluated by predicting prices of Airbnb listings in two other cities, namely Amsterdam and Rome. The same features and models used for the Istanbul data set were tested on the Amsterdam data set and Rome data set. Table 5 shows the prediction performance of the models on the Amsterdam data set and Rome data set.

The results showed that none of the models generalize on data of Airbnb listings in Amsterdam or Rome. All models achieved a negative R^2 score which implies that the fit of a model is worse than the mean value of the data (Chicco et al., 2021). Although the models do not generalize, the performance of the models differs per data set. The XGBoost Regressor achieved the best performance on the Amsterdam data set. This model obtained the highest R^2 value (-3.796) and the lowest MAE (1.068) and RMSE (1.150) values. Whereas, the Support Vector Regressor obtained the best performance on the Rome data with an R^2 score of -4.689, MAE value of 1.575, and RMSE value of 1.665. This result is notable because the

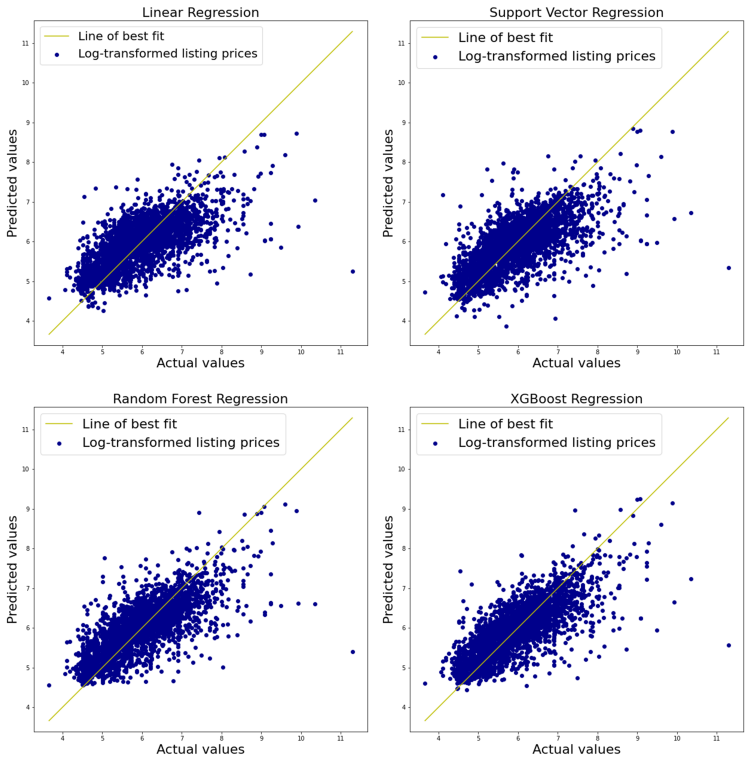

Figure 2: Prediction performances of the different models (based on test set)

Support Vector Regressor was the worst-performing model on the Istanbul data set. Moreover, the models trained on the Istanbul data set showed slightly better performance on the Amsterdam data set, in general. Overall, it can be concluded that the features and the data of Airbnb listings in Istanbul were insufficient to predict Airbnb prices in Amsterdam and Rome.

Table 5: Performance metrics for each model tested on Amsterdam and Rome data sets

Model	R^2	MAE	RMSE	City
Linear Regression (baseline)	-4.641	1.140	1.247	Amsterdam
Random Forest Regressor	-3.894	1.079	1.162	Amsterdam
Support Vector Regressor	-4.049	1.096	1.180	Amsterdam
XGBoost Regressor	**-3.796**	**1.068**	**1.150**	Amsterdam
Linear Regression (baseline)	-5.057	1.640	1.718	Rome
Support Vector Regressor	**-4.689**	**1.575**	**1.665**	Rome
Random Forest Regressor	-5.356	1.681	1.760	Rome
XGBoost Regressor	-5.307	1.676	1.753	Rome

Figure 3 shows the actual values and predicted values obtained with the best models for Airbnb listings in Amsterdam and Rome. The predictive performance of the remaining models can be found in Appendix B (page 29).

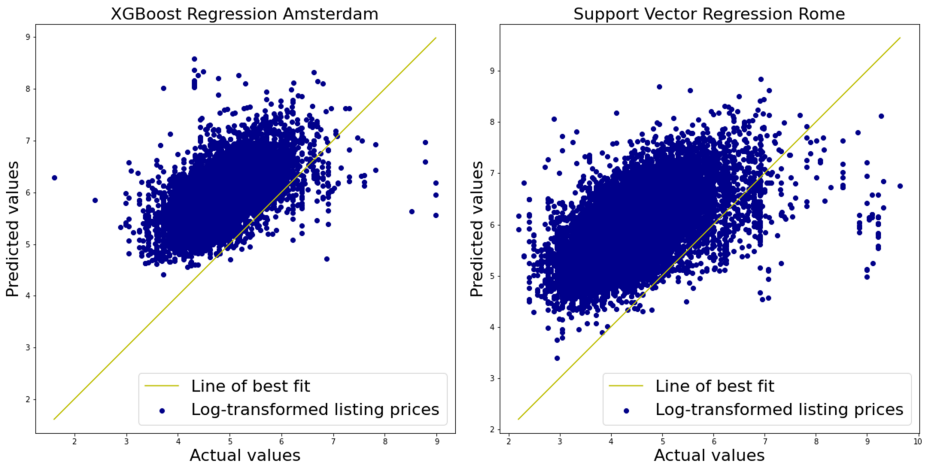

Figure 3: The actual values and predicted values of the best models for the Amsterdam data set and Rome data set.

7 DISCUSSION

This study aims to predict Airbnb listing prices in Istanbul using listing-related features and reviews. To answer the main research question, four sub-questions were formulated:

SQ1 *What is the compound score of each reviews?*

SQ2 *Which features affect the price of Airbnb listings in Istanbul the most?*

SQ3 *Which model predicts the price of Airbnb listings in Istanbul the best?*

SQ4 *How well do the models generalize to other cities?*

The answer to each sub-question will be discussed below.

SQ1 *What is the compound score of each reviews?*

To use reviews for predicting prices of Airbnb listings in Istanbul sentiment analysis was performed. In contrast to prior studies conducted by Rezazadeh Kalehbasti et al. (2021) and Zhu et al. (2020) this study used VADER, as the sentiment analysis tool, to obtain the compound sentiment scores of each guest review. Accordingly, these compound scores were

used to calculate a mean compound score per listing since a listing can have multiple reviews. As a result, a new feature, 'Compound_score' was created for each listing and merged with feature set from the Listings data set.

SQ2 *Which features affect the price of Airbnb listings in Istanbul the most?*

Feature selection is an important part of the pre-processing process. To determine the feature importance the p value was computed for each feature. The results showed that the top 10 most important features consist of property-related features, such as 'property type' and room-related features, such as the 'Beds'. The newly created feature 'Compound_score' has a p value of .044 and was the 12 most important feature for predicting Airbnb listing prices in Istanbul.

Therefore, it can be concluded that reviews have an influence on the price of Airbnb listings in Istanbul. This result supports the study conducted by Lawani et al. (2019) since they also stated that review scores highly influence the price of an Airbnb listing.

Moreover, the number of features was determined using the 'SelectKBest' that selects the features based on the importance given k, where k is the number of features to select. In this study, 11 different values for k were determined. Subsequently, each set of features was used as the input to the different models. After training, tuning, and testing the different models it was found that the highest performance was achieved with $k = 60$. However, this was a computationally expensive process. Therefore, it was not possible to try every possible number for k. One solution could be to include and try the remaining possible numbers of k. Another suggestion would be to use other feature selection methods in order to obtain the most important features for predicting Airbnb listing prices in Istanbul. Moreover, a comparison of the features used for producing Airbnb listings in Istanbul could not be made, since there are no prior studies for predicting Airbnb listings in Istanbul.

SQ3 *Which model predicts the price of Airbnb listings in Istanbul the best?*

In this study, several machine learning models were built to predict the price of Airbnb listings in Istanbul. First of all, a Linear Regression model was built and was the baseline model in this study. Subsequently, a Random Forest Regression, Support Vector Regression, and XGBoost Regression were built. To compare the performance of the models, the R^2 value, MAE value, and RMSE values were analyzed. The results show that all models outperformed the baseline model. However, among these models, the XGBoost Regressor achieved the best performance. This result supports

previous studies conducted by Kokasih and Paramita (2020), Zhu et al. (2020), and Yang (2021), since they also obtained the best performance with the XGBoost Regressor. Among all models, the Support Vector Regression model achieved the lowest performance.

SQ4 *How well do the models generalize to other cities?*

The final sub-question was answered by testing the models built for predicting prices of Airbnb listings in Istanbul on Airbnb data from other cities. In this study, the Amsterdam data set and Rome data set were entirely used to assess the generalizability of the models built for the Istanbul data set. The results showed that none of the models generalize to other cities. All models had a negative R^2 score and high MAE, and RMSE values. The obtained result is interesting since this implies that the features and the data for a specific city, in particular Istanbul, are not sufficient to predict prices in other cities - Amsterdam and Rome. Despite, the models do not generalize to other cities the model performance differs per city. The XGBoost Regressor performs the best on the Amsterdam data set and the Support Vector Regressor on the Rome data set.

In this study, the approach was to test the models built for Istanbul directly on the Amsterdam data set and Rome data set to assess the generalizability of the features and the data of Airbnb listings in Istanbul. Since previous studies did not compare the generalization results these results can not be compared to previous performances. However, the results obtained regarding the generalizability of the models built for Istanbul are worse, therefore there is room to improve the generalizability of these models in future studies. Another direction could be to assess the generalizability of only the features from the Istanbul data set. An approach for this could be to first split the data sets of the other cities into a training set and test set. Next, train the models built for predicting Airbnb listings in Istanbul again on these new data sets. Subsequently, assess the performance of these models on the created test sets.

This study has several limitations. First, as mentioned earlier only English reviews were used to perform sentiment analysis because the used sentiment analysis tool (VADER) is only compatible with English text. Due to time constraints, it was not possible to use machine translation in this study. A suggestion would be to use the DeepL translator API or Google Translate API to translate the non-English reviews and include them in the sentiment analysis process. By using the aforementioned machine translation tools it needs to be taken into account that these tools have a limit of characters to translate per day/per request, therefore some time is necessary to accomplish this. However, since the majority of the reviews were written in English, there was enough data to conduct sentiment anal-

ysis on. Second, during the hyperparameter process, only a limited set of possible candidates were defined. Therefore, an improvement of this process would be to add more candidates and/or tune other hyperparameters and investigate the model performances. Moreover, an extension to the used models would be to use a Neural Network to predict Airbnb listing prices and examine the performance of this model.

This study contributes to the existing literature by developing a model to predict prices of Airbnb listing in Istanbul since this was not observed in the existing literature. The proposed model can be used by hosts to determine the best price for their Airbnb listing. In addition, guests can also use this model to assess whether the indicated price for a listing is fair. Moreover, this study investigated if the features and data of Airbnb listings in Istanbul generalize to other cities - Amsterdam and Rome. Assessing the generalizability of these models to these cities was also not observed in the literature.

Overall, Airbnb listing prices in Istanbul can be predicted with listing characteristics and reviews by using natural language processing and machine learning. The models built to predict Airbnb listing prices are performing well, and the features and data of Airbnb listings in Istanbul are insufficient to predict Airbnb listing prices in Amsterdam and/or Rome.

8 CONCLUSION

This thesis aims to predict the prices of Airbnb listings in Istanbul using various machine learning models (Linear Regression, XGBoost Regression, Random Forest Regression, and Support Vector Regression) based on Airbnb data sets containing listing related features and reviews. In order to answer the main research question, four sub-questions were formulated.

The first sub-question involved computing the compound score of each review. For this sub-question, the Reviews data set was used to perform the sentiment analysis. First, the compound score of each review was obtained by using a lexicon-and-rule-based sentiment analysis tool, namely VADER (Hutto & Gilbert, 2014). Subsequently, an average of the compound scores was taken because several listings contained multiple reviews. As a result, a new feature 'Compound_score' was created for each data set.

Moreover, the second sub-question was to determine which features influence the prices of Airbnb listings in Istanbul the most. The results show that property-related features influence the price the most. In addition, the newly created feature 'Compound_score' also affects the price of Airbnb listings in Istanbul. Furthermore, the third sub-question was to identify the best model (Linear regression, XGBoost Regression, Random Forest Regression, and Support Vector Regression) for predicting Airbnb listing

prices. The results show that all models outperformed the baseline model (Linear regression). However, the XGBoost Regression model achieves the best result. The last sub-question examined if the models built for the Istanbul data sets generalizes to other cities, in this study to Amsterdam and Rome. The results show that none of the models generalize to other cities. This implies that using only the features and data of Airbnb listings in Istanbul is insufficient to predict Airbnb listing prices in Amsterdam and Rome. This study implies that reviews affect the Airbnb listing prices in Istanbul, and the models for this particular city do not generalize to other cities.

There are multiple future research directions identified. The first direction would be to use machine translation to include also non-English reviews for sentiment analysis. Moreover, other feature selection methods can be used. In addition, the set of hyperparameters can be extended or other hyperparameters can be tuned. Another direction is to focus on the generalizability of price prediction models for Airbnb listings. Currently, the models for Istanbul do not generalize to other cities - Amsterdam and Rome. Therefore, one suggestion would be to combine Airbnb data from different cities to investigate whether one general model can be developed to predict the prices of Airbnb listings in different cities.

REFERENCES

Awad, M., & Khanna, R. (2015). Support vector regression. In *Efficient learning machines* (pp. 67–80). Springer. doi: 10.1007/978-1-4302-5990-9_4

Biau, G. (2012). Analysis of a random forests model. *The Journal of Machine Learning Research*, *13*, 1063–1095.

Biau, G., & Scornet, E. (2016). A random forest guided tour. *Test*, *25*(2), 197–227. doi: 10.1007/s11749-016-0481-7

Birjali, M., Kasri, M., & Beni-Hssane, A. (2021). A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, 107134.

Bonta, V., & Janardhan, N. K. N. (2019). A comprehensive study on lexicon based approaches for sentiment analysis. *Asian Journal of Computer Science and Technology*, *8*(S2), 1–6.

Borg, A., & Boldt, M. (2020). Using vader sentiment and svm for predicting customer response sentiment. *Expert Systems with Applications*, *162*, 113746.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Caetano, J. A., Lima, H. S., Santos, M. F., & Marques-Neto, H. T. (2018). Using sentiment analysis to define twitter political users' classes and their homophily during the 2016 american presidential election. *Journal of internet services and applications*, *9*(1), 1–15.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794).

Cherif, I. L., & Kortebi, A. (2019). On using extreme gradient boosting (xgboost) machine learning algorithm for home network traffic classification. In *2019 wireless days (wd)* (pp. 1–6).

Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation. *PeerJ Computer Science*, *7*, e623.

Cokluk, O., & Kayri, M. (2011). The effects of methods of imputation for missing values on the validity and reliability of scales. *Educational Sciences: Theory and Practice*, *11*(1), 303–309.

Curran-Everett, D. (2018). Explorations in statistics: the log transformation. *Advances in physiology education*, *42*(2), 343–347.

Danilák, M. (2021). *Langdetect - pypi.* Python Software Foundation. Retrieved from https://pypi.org/project/langdetect/

Dhillon, J., Eluri, N. P., Kaur, D., Chhipa, A., Gadupudi, A., Eravi, R. C., & Pirouz, M. (2021). Analysis of airbnb prices using machine learning

techniques. In *2021 ieee 11th annual computing and communication workshop and conference (ccwc)* (pp. 0297–0303).

Hadeed, S. J., O'Rourke, M. K., Burgess, J. L., Harris, R. B., & Canales, R. A. (2020). Imputation methods for addressing missing data in short-term monitoring of air pollutants. *Science of The Total Environment*, *730*, 139140.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... Oliphant, T. E. (2020, September). Array programming with NumPy. *Nature*, *585*(7825), 357–362. doi: 10.1038/s41586-020-2649-2

Haselmayer, M., & Jenny, M. (2017). Sentiment analysis of political communication: combining a dictionary approach with crowdcoding. *Quality & quantity*, *51*(6), 2623–2646. doi: 10.1016/j.knosys.2021.107134

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. doi: 10.1109/MCSE.2007.55

Hutto, C., & Gilbert, E. (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international aaai conference on web and social media* (Vol. 8).

Ibrahim, M., Abdillah, O., Wicaksono, A. F., & Adriani, M. (2015). Buzzer detection and sentiment analysis for predicting presidential election results in a twitter nation. In *2015 ieee international conference on data mining workshop (icdmw)* (pp. 1348–1353).

Inamdar, A., Bhagtani, A., Bhatt, S., & Shetty, P. M. (2019). Predicting cryptocurrency value using sentiment analysis. In *2019 international conference on intelligent computing and control systems (iccs)* (pp. 932–934).

Jadhav, A., Pramod, D., & Ramanathan, K. (2019). Comparison of performance of data imputation methods for numeric dataset. *Applied Artificial Intelligence*, *33*(10), 913–933. doi: 10.1080/08839514.2019.1637138

Jin, Z., Yang, Y., & Liu, Y. (2020). Stock closing price prediction based on sentiment analysis and lstm. *Neural Computing and Applications*, *32*(13), 9713–9729.

Kokasih, M. F., & Paramita, A. S. (2020). Property rental price prediction using the extreme gradient boosting algorithm. *International Journal of Informatics and Information Systems*, *3*(2), 54–59.

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling* (Vol. 26). Springer.

Lawani, A., Reed, M. R., Mark, T., & Zheng, Y. (2019). Reviews and price on online platforms: Evidence from sentiment analysis of airbnb reviews in boston. *Regional Science and Urban Economics*, *75*, 22–34.

Li, J. (2017). Assessing the accuracy of predictive models for numerical data: Not r nor r2, why not? then what? *PloS one*, *12*(8), e0183250.

Ligthart, A., Catal, C., & Tekinerdogan, B. (2021). Systematic reviews in sentiment analysis: a tertiary study. *Artificial Intelligence Review*, 1–57.

Luo, T., Chen, S., Xu, G., & Zhou, J. (2013). Sentiment analysis. In *Trust-based collective view prediction* (pp. 53–68). Springer.

Maulud, D., & Abdulazeez, A. M. (2020). A review on linear regression comprehensive in machine learning. *Journal of Applied Science and Technology Trends*, *1*(4), 140–147.

Mohan, S., Mullapudi, S., Sammeta, S., Vijayvergia, P., & Anastasiu, D. C. (2019). Stock price prediction using news sentiment analysis. In *2019 ieee fifth international conference on big data computing service and applications (bigdataservice)* (pp. 205–208).

Mouthami, K., Devi, K. N., & Bhaskaran, V. M. (2013). Sentiment analysis and classification based on textual reviews. In *2013 international conference on information communication and embedded systems (icices)* (pp. 271–276).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Priambodo, F. N., & Sihabuddin, A. (2020). An extreme learning machine model approach on airbnb base price prediction. *International Journal of Advanced Computer Science and Applications*, *11*(11). doi: 10.14569/IJACSA.2020.0111123

Ramteke, J., Shah, S., Godhia, D., & Shaikh, A. (2016). Election result prediction using twitter sentiment analysis. In *2016 international conference on inventive computation technologies (icict)* (Vol. 1, pp. 1–5).

Renault, T. (2020). Sentiment analysis and machine learning in finance: a comparison of methods and models on one million messages. *Digital Finance*, *2*(1), 1–13.

Rezazadeh Kalehbasti, P., Nikolenko, L., & Rezaei, H. (2021). Airbnb price prediction using machine learning and sentiment analysis. In *International cross-domain conference for machine learning and knowledge extraction* (pp. 173–184).

Rodríguez-Ibáñez, M., Gimeno-Blanes, F.-J., Cuenca-Jiménez, P. M., Soguero-Ruiz, C., & Rojo-Álvarez, J. L. (2021). Sentiment analysis of political tweets from the 2019 spanish elections. *IEEE Access*, *9*, 101847–101862.

Sattarov, O., Jeon, H. S., Oh, R., & Lee, J. D. (2020). Forecasting bitcoin price fluctuation by twitter sentiment analysis. In *2020 international conference on information science and communications technologies (icisct)* (pp. 1–4).

Shmueli, G., Bruce, P. C., Gedeck, P., & Patel, N. R. (2019). *Data mining for*

business analytics: concepts, techniques and applications in python. John Wiley & Sons.

Uyanık, G. K., & Güler, N. (2013). A study on multiple linear regression analysis. *Procedia-Social and Behavioral Sciences*, *106*, 234–240.

Wang, E. Y., Fong, L. H. N., & Law, R. (2020). Review helpfulness: The influences of price cues and hotel class. In *Information and communication technologies in tourism 2020* (pp. 280–291). Springer.

Waskom, M. L. (2021). seaborn: statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. doi: 10.21105/joss.03021

Wes McKinney. (2010). Data Structures for Statistical Computing in Python. In Stéfan van der Walt & Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (p. 56 - 61). doi: 10.25080/Majora-92bf1922-00a

Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate research*, *30*(1), 79–82.

Wooley, S., Edmonds, A., Bagavathi, A., & Krishnan, S. (2019). Extracting cryptocurrency price movements from the reddit network sentiment. In *2019 18th ieee international conference on machine learning and applications (icmla)* (pp. 500–505).

Yang, S. (2021). Learning-based airbnb price prediction model. In *2021 2nd international conference on e-commerce and internet technology (ecit)* (pp. 283–288).

Yue, L., Chen, W., Li, X., Zuo, W., & Yin, M. (2019). A survey of sentiment analysis in social media. *Knowledge and Information Systems*, *60*(2), 617–663.

Zhu, A., Li, R., & Xie, Z. (2020). Machine learning prediction of new york airbnb prices. In *2020 third international conference on artificial intelligence for industries (ai4i)* (pp. 1–5).

APPENDIX A

Table 6: All features of the original Listings data set with corresponding data type.

Feature	Data type
Id	Integer
Listing_url	Object
Scrape_id	Integer
Last_scraped	Object
Name	Object
Description	Object
Neighborhood_overview	Object
Picture_url	Object
Host_id	Integer
Host_url	Object
Host_name	Object
Host_since	Object
Host_location	Object
Host_about	Object
Host_response_time	Object
Host_response_rate	Object
Host_acceptance_rate	Object
Host_is_superhost	Object
Host_thumbnail_url	Object
Host_picture_url	Object
Host_neighbourhood	Object
Host_listings_count	Float
Host_total_listings_count	Float
Host_verifications	Object
Host_has_profile_pic	Object
Host_identity_verified	Object
Neighbourhood	Object
Neighbourhood_cleansed	Object
Neighbourhood_group_cleansed	Float
Latitude	Float
Longitude	Float
Property_type	Object
Room_type	Object
Accommodates	Integer
Bathrooms	Float
Bathrooms_text	Object
Bedrooms	Float

Continuation of Table 6

Feature	Data type
Beds	Float
Amenities	Object
Price	Object
Minimum_nights	Integer
Maximum_nights	Integer
Minimum_minimum_nights	Integer
Maximum_minimum_nights	Integer
Minimum_maximum_nights	Integer
Maximum_maximum_nights	Integer
Minimum_nights_avg_ntm	Float
Maximum_nights_avg_ntm	Float
Calendar_updated	Float
Has_availability	Object
Availability_30	Integer
Availability_60	Integer
Availability_90	Integer
Availability_365	Integer
Calendar_last_scraped	Object
Number_of_reviews	Integer
Number_of_reviews_ltm	Integer
Number_of_reviews_l30d	Integer
First_review	Object
Last_review	Object
Review_scores_rating	Float
Review_scores_accuracy	Float
Review_scores_cleanliness	Float
Review_scores_checkin	Float
Review_scores_communication	Float
Review_scores_location	Float
Review_scores_value	Float
License	Object
Instant_bookable	Object
Calculated_host_listings_count	Integer
Calculated_host_listings_count_entire_homes	Integer
Calculated_host_listings_count_private_rooms	Integer
Calculated_host_listings_count_shared_rooms	Integer
Reviews_per_month	Float

APPENDIX B

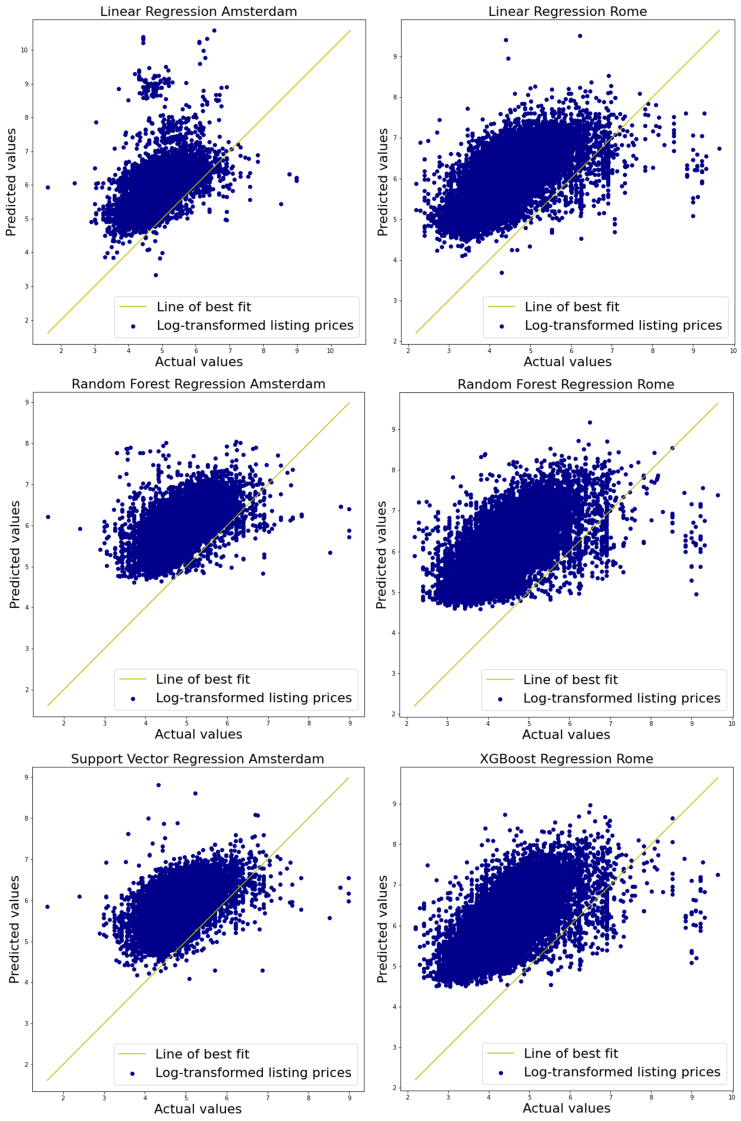

Figure 4: The actual values and predicted values of the other models for the Amsterdam data set and Rome data set.

APPENDIX C

Table 7: All features of the original Reviews data set with corresponding data type.

Feature	Data type
Listing_id	Integer
Id	Integer
Date	Object
Reviewer_id	Integer
Reviewer_name	Object
Comments	Object

APPENDIX D

Table 8: Features - in order of importance - used for predicting Airbnb listing prices.

Feature	Score (p value)
Accommodates	.204
Entire villa	.175
Bedrooms	.117
Beds	.097
Bathrooms	.095
Room type: Private room	.087
Room type: Entire home/apt	.084
Private room in rental unit	.074
Private room in boat	.052
Boat	.045
Unknown response time	.045
Compound_score	.044
Availability_365	.043
Host_total_listings_count	.035
Private room in residential home	.028
Host_response_rate	.028
Entire chalet	.027
Number_of_reviews	.023
Entire residential home	.023
Response: a few days or more	.022
Shared room in chalet	.021
Response: within an hour	.020
Private room in townhouse	.020
Room type: Hotel room	.019
Review_scores_cleanliness	.018
Entire rental unit	.018
Room type: Shared room	.017
Review_scores_communication	.017
Review_scores_location	.017
Nr_amenities	.017
Room in hotel	.017
Earth house	.017
Instant_bookable	.016
Entire townhouse	.015
Review_scores_checkin	.015
Shared room in hostel	.014

Continuation of Table 8	
Feature	Score (p value)
Entire condominium (condo)	.014
Private room in serviced apartment	.013
Private room in condominium (condo)	.013
Entire serviced apartment	.013
Private room in loft	.013
Ranch	.013
Room in aparthotel	.012
Room in bed and breakfast	.011
Private room in hostel	.009
Review_scores_accuracy	.009
Shared room in rental unit	.008
Room in boutique hotel	.008
Shared room in residential home	.008
Host_identity_verified	.007
Private room in casa particular	.007
Room in hostel	.007
Review_scores_value	.007
Entire place	.0067
Response: within a few hours	.006
Shared room in bed and breakfast	.006
Minimum_nights	.006
Tiny house	.006
Private room in tiny house	.005
Private room	.004