|  |
| --- |
|  |
| Airbnb Rental Price Estimation Using ML Methods |
|  |

|  |
| --- |
| Gandhar Gokhale  11-21-2024 |

Contents

[Abstract 2](#_Toc183116820)

[Introduction 2](#_Toc183116821)

[Prior work 3](#_Toc183116822)

[Data 4](#_Toc183116823)

[Initial data exploration, preprocessing and cleanup 4](#_Toc183116824)

[Expansion of Amenities feature 7](#_Toc183116825)

[Reviews mapped to numeric sentiment scores 7](#_Toc183116826)

[Method 9](#_Toc183116827)

[Linear Regression 9](#_Toc183116828)

[Decision Tree 9](#_Toc183116829)

[Random Forrest 10](#_Toc183116830)

[SVM 10](#_Toc183116831)

[XGBoost 11](#_Toc183116832)

[Hyperparameters 11](#_Toc183116833)

[Feedforward deep neural network 11](#_Toc183116834)

[Findings and conclusions 12](#_Toc183116835)

[Future Work 14](#_Toc183116836)

[References 15](#_Toc183116837)

# Abstract

The sharing economy has transformed how individuals access goods and services. Platforms like Airbnb have revolutionized the short-term accommodations industry. The prices of the rooms fluctuate widely temporally as well as spatially. This poses challenges both to the consumer and the provider i.e. the guest and the host. For guests it is in predicting whether they are getting the best possible accommodation for their budget whereas for the host the challenge is to maximize the occupancy while getting the best possible rates. Inspired by a Kaggle dataset for the city of Albany, NY, USA this study looked at the aspects that determine the listing prices of the accommodations. The study applies various Exploratory Data Analyses (EDA) techniques to clean up the data and to identify significant influences. Using a Large Language Model (LLM) Natural Language Processing (NLP) for sentiment analysis of the reviews of each of the listings sentiment scores are generated which, are then combined with various other factors of the listings to develop various machine learning models. These models provide varying degrees of accuracy while consuming different computational costs. This project uncovers interesting insights into the factors influencing rental prices. The analysis incorporates diverse variables, including location, amenities, occupancy and other property features. The robust models proposed here are computationally light weight, but they provide high accuracy in price prediction as determined by the Root Mean Squared error. This paper details the methodological framework, key discoveries, and the practical implications of an intelligent pricing prediction system in the context of the Airbnb ecosystem.

# Introduction

Due to the great technological advancements such as cloud computing, mobile handheld devices and mobile networks the new model of shared economy has revolutionized various aspects of life such as cab ride hailing, smart city bike riding or AirBnb. Airbnb had attracted 100 million hosts and guests worldwide, earning $100 million by 2017 [2],[3] (today, by some estimates there are over five million Airbnb hosts all over the world).

The author of this study started to travel often in recent times and finding good temporary accommodation became a repeated problem, for which the shared economy provided an efficient and convenient solution. The AirBnb and similar platforms have revolutionized the hospitality industry. It offers the travelers flexible and multitude of accommodation options and at the same time enables hosts, who are otherwise, not necessarily, business owners, a means of extra stream of reliable income. The platform's widespread adoption stems from its ability to connect guests with hosts seamlessly. Moreover, it creates a dynamic marketplace that fosters choice and convenience. On the other hand though, due the less standardized nature of the offerings it becomes somewhat challenging for the travelers to assess the value that each accommodation offers and its listed price [5]. To add to that, like any other modern marketplace the pricing is very dynamic, changing sometimes daily, if not more. Navigating such a fluctuating pricing landscape of Airbnb listings can often be perplexing. Prices for similar properties can vary significantly, influenced by a myriad of factors ranging from property attributes to external market conditions.

After seeing the [[Kaggle dataset](https://www.kaggle.com/datasets/rhonarosecortez/new-york-airbnb-open-data?resource=download&select=reviews.csv) for AirBnb market listing in Albany, New York](https://www.kaggle.com/datasets/rhonarosecortez/new-york-airbnb-open-data?resource=download&select=reviews.csv), the author’s interest sparked the curiosity about the underlying drivers of price determination because as a frequent traveler relying on Airbnb for accommodations, pronounced variations in rental prices are observed. Having been trained in data analysis and Machine Learning techniques it immediately became an interesting subject of investigation with direct tangible applicability.

The primary objective of this study is to analyze the pricing patterns of Airbnb rentals within a specific city in relation to various aspects of the listings, identify the relationships of the factors with occupancy and pricing, correlate how the reviews written by the guests relate to other price and occupancy of the listings and finally build a machine learning model to accurately predict rental prices based on various attributes. The project employs comprehensive data analysis and state-of-the-art machine learning techniques to provide actionable insights.

The dataset encompassed features such as location, property type, amenities, host ratings, and seasonal trends. By employing statistical techniques and visualizations, initial patterns and correlations were identified. Subsequently, machine learning models, including regression algorithms, neural networks and ensemble methods, were developed and evaluated to predict rental prices effectively.

The rest of the paper describes the prior work done in this area, provides the details of the data and steps to clean up the data. Then it describes the discovery of the features or the factors of importance to prepare the data suitable for inputting into machine learning models. Various machine learning models are used to discover efficient and robust models with tuned hyperparameters. The performance of the models is determined by using the root mean squared error on the validation dataset. Then the paper discusses conclusions as well as direction for the future work in this area.

The python implementation that includes a jupyter notebook and a couple of scripts uses the standard libraries such as pandas, scikit-learn, numpy, seaborn, matplotlib and also the transformer and LLM from HuggingFace. The code is shared at <https://github.com/3of7/CSML.git>.

# Prior work

Due to the explosive growth in the shared economy industry and Airbnb in particular a lot of interest is present in finding ways to increase the price predictability for the hosts as well as for the guests. Various Machine Learning techniques are applied over the course of time for like in [4],[6],[7],[8],[9],[10],[11]. For instance, [12] looks at big Kaggle data set of AirBnb prices across the entire US. This dataset has more than 200k entries. The number features, though, are limited to mainly location and room type. The RMSE for regression remained high so the authors defined classes of prices to convert the regression problem into a classification problem. However, the classification accuracy too had a high variance. The small number of parameters appears to be the reason for not achieving better results. [13] used dataset from [public.opendatasoft.com](https://public.opendatasoft.com/explore/dataset/airbnb-listings/table/?disjunctive.host_verifications&disjunctive.amenities&disjunctive.features&dataChart=%3D). There are a total of 494,954 records each of which contains details of one Airbnb listing across various countries in the US, Europe and Australia. The total size of dataset is 1.89 GB. It’s a single flat dataset with very similar features like the dataset used by the present work. However, it doesn’t consider the reviews as well as occupancy levels. The RMSE score shows that the price prediction is less than satisfactory. The probable improvement seems to be further data pre-processing. [14] dataset provides more than 67k entries of AirBnb listings across 20k+ cities. The features are limited but with large variations. E.g., the room types are more than 2000 or bed types are more than 1300. The amenities are not listed in a granular fashion and reviews are absent. Due to the log pricing the RMSE can’t be directly compared with the present work.

# Data

The dataset is available [1]. There are three .csv file containing the data for AirBnb listings in Albany, NY.

1. The calendar.csv file has data about occupancy for one year.
2. The listings.csv file consist of the descriptive details of 426 properties. There are 75 features in this file while some features such as amenities are in fact aggregated lists of features.
3. The reviews.csv file contains all the reviews for each of the listed entries. There are 6k+ unique reviewers providing 24k+ reviews in total.

## Initial data exploration, preprocessing and cleanup

Listings.csv is the most important table here. It has a lot of features but many of them may not be very relevant for price prediction. As discussed in [15] “Feature subset selection is the process of identifying and removing as much of the irrelevant and redundant information as possible. The role of feature selection in machine learning is (1) to speed up the prediction process, (2) to improve the correlation coefficient of a regression algorithm, and (3) to improve the comprehensibility of the learning results”. There are many sophisticated algorithms investigated for feature selection in high dimensional regression such as [17],[18],[19]. In this work we keep it simply based on relevance and correlation with the target variable [20]. Table 1is the list of features and the apparent relevance from their descriptions provided. Then the correlation values are used to eliminate the uncorrelated features. It is noted that amenities are given as word lists. One of the hypotheses was that amenities do influence the demand by the guests as well as the costs borne by the host and hence are expected to correlate with the listing price. Therefore, the amenities list was expanded to individual features. While it exploded the number of columns these new columns are all binary valued which computationally not too heavy.



Table 1 Feature relevance from the Listings

The price column had extra “$” symbol which was removed to make it a numerical value. Also, it was observed that there are a few outliers which could skew the overall model’s performance. The statistical distribution of price values indicates that there are outliers.

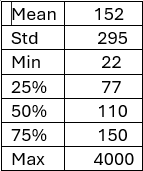
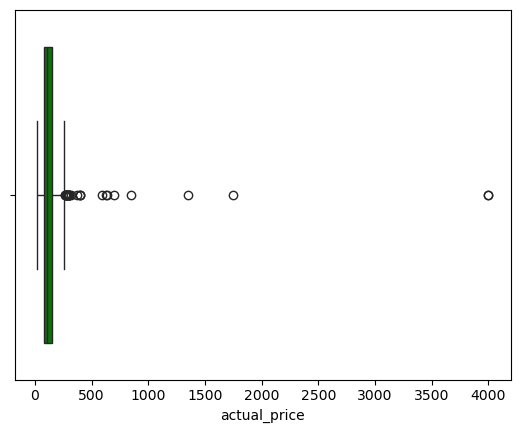


Figure 1Price distribution and outliers

A map of a city

Description automatically generated

Figure 2 Listing price clusters

A map of a city

Description automatically generated

Figure 3: Listing prices after zooming in into a cluster

The price is moderately to strongly correlated with the number of beds, persons it can accommodate and number of bedrooms. These are certainly important factors. However, they mask the importance of other features if taken at absolute values. Hence their effect is normalized by considering per unit prices for number of beds, number of bedrooms and number of people accommodated.

## Expansion of Amenities feature

The hypothesis of this work was that there some amenities affect the prices more than the other amenities which is not studied extensively in literature [8]. Since the data provided doesn’t include individual amenities as features but only a list of amenities as comma separated strings they were converted into one-hot-encoded feature vectors. Many amenities have been described with different strings but would differ in small details which is why such strings were aggregated to single class. E.g. *HDTV, HDTV\_32\_inches, HDTV\_48* inches etc. were aggregated to a single amenity *HDTV*. Similarly for *Wi-Fi* routers and their speed based descriptions were combined together. Even after all the aggregations the amenity expansion added a very large number of new columns. It was observed that some of the amenities were too common. That is, they are so common that one can expect that their presence doesn’t affect the choice of the renters. Here’s the list of such amenities *'Carbon monoxide alarm', 'Dishes and silverware', 'Cooking basics', 'Essentials', 'Fire extinguisher', 'Hair dryer', 'Hangers', 'Hot water', 'Iron', 'Kitchen', 'Microwave', 'Refrigerator', 'Refrigerator, 'Self check-in', 'Smoke alarm*'. These features were dropped from the model training and inference. The overall count of features (columns) used became 451 after the entire processing.

## Reviews mapped to numeric sentiment scores

The reviews.csv provided a different, and in this work was hypothesized to be more reliable, way of assessing the evaluation of the house. There have been some effort in using the reviews for extracting sentiment scores [6] but it was not very granular to see the comparison with the review scores. The listsings.csv does contain review scores and more granular review scores for various aspects such as cleanliness. In this work we converted the verbal reviews to a numeric sentiment score of 1 to 5 where 1 is the worst and 5 is the best score. With easily available lightweight NLP models encouraged the idea to be tried. I selected a pre-trained model “distilbert-base-uncased-finetuned-sst-2-english” from HuggingFace used with AutoTokenizer and AutoModelForSequenceClassification modules from the Transformer library by HuggingFace after considering some prior work and studying the alternatives [28],[29],[30]. The model is cheaper, faster and lightweight version of BERT model. It has 40% less parameters than BERT but still retains 95% performance on GLUE score which is used to asses language understanding. Since the requirement is to find out the sentiment this model is a great fit.

The model’s logits were passed to a softmax function to get real valued pair from the classifier. The first value represents the confidence about being a member of the positive class and second value represents that of negative. Thresholds of 0.9 were used to assign the most positive (5) and most negative (1) scores, thresholds of 0.6 were used to assign moderately positive (4) and moderately negative (2) scores while the reset were scored as 3. Absent values were set to score (3) to keep neutrality. Every review was scored individually scored. Then the average of the sentiment scores per listing\_id were computed and were set as the review score for that listing\_id. The sentiment scores were then appended to the feature list after matching the listing\_ids. From Figure 4 we can see the sentiment analysis has provided better scores at lower levels of satisfaction among the renters.

Figure 4: Review score distribution

# Method

This is a classic regression problem statement, where we predict the real valued price of the Airbnb listing unit given the listing’s refined features from the previous pre-processing steps. There are many different potential algorithms of which some have been used in the past to attack this problem [21],[22],[23],[24],[25],[26],[27] .The Root Mean Square Error (RMSE) and Mean Absolute Error are the evaluation criteria used to judge the performance of every model on the test set. The pre-processed dataset was split into 80:20 as training vs test data. The dataset was shuffled with different random seeds to ensure robustness of the model. The following models were selected to explore and identify the best model

### Linear Regression

Linear Regression models try to find a linear relationship between input features and the target variable, making it simple, interpretable, and computationally efficient. For regression problems with many features, it provides a good baseline, especially when the relationships between predictors and the target are approximately linear. Regularization techniques like Lasso (L1) or Ridge (L2) regression improve performance in high-dimensional settings by penalizing large coefficients, thereby preventing overfitting and automatically selecting relevant features. Therefore, as per previously described exploratory data analysis Linear regression (with regularization) appeared to be a good candidate algorithm.

No hyper parameters tuned for the base model. With default parameter values this was used more like a baseline model. RMSE accuracy achieved was 106.35 and MAE was 78.

As the Linear Regression model is expected to overfit, Lasso was used to reduce the overfitting.

Hyper parameters tuned for Lasso:

alpha: 0.1-0.25

max\_iterations:1000 - 10000

Lasso with alpha 0.1 and max\_iterations=1000 improved the validation RMSE dramatically. The RMSE achieved with Lasso is 19.83 while MAE is 16.64. There was no significant sacrifice on computation cost for prediction.

### Decision Tree

Decision Trees divide the data into subsets by learning simple rules based on feature values. They capture non-linear relationships and feature interactions. They are interpretable and provide insights into which features are most important. For high-dimensional datasets trees are particularly useful because they perform inherent feature selection i.e. they focus only on the most relevant ones. With large number of features owning to data preprocessing with expanded amenities decision tree model is a good prospective model. However, they can overfit the data when many features are involved, unless regularization techniques like pruning are applied to limit tree depth and complexity which was observed in the results.

Hyper parameters tuned:

max\_depth : 3-500

This was found to be a robust model. RMSE accuracy of 18.64 and MAE of 14.98 achieved which are the best amongst all the models compared in this study.

### Random Forrest

Random Forests are ensembles of decision trees. Each of the trees is trained on a random subset of features and data samples. It usually helps mitigate overfitting and improves generalization. They are effective at handling high-dimensional data as the random feature selection prevents reliance on irrelevant predictors. Additionally, Random Forests naturally handle non-linear relationships and complex interactions, providing robust and accurate predictions. Feature importance metrics [Figure 5] generated by the algorithm also help in understanding which features contribute most to the prediction.

Hyper parameters tuned:

n\_estimators (number of decision trees in the forest): 3-900

max-depth: 3-100

max\_leaf\_nodes=3-1000

Less max\_depth, higher n\_estimators give better RMSE accuracy. However, it resulted in increased runtime. The benefit of improved accuracy didn’t outweigh the cost of the increased runtime. RMSE and MAE scores were 19.08 and 15.04 respectively.

### SVM

Support Vector Machines (SVM) aim to find the optimal hyperplane that minimizes the prediction error while separating data points. They are well-suited for high-dimensional datasets, as their performance depends more on the number of support vectors than the dimensionality of the feature space. Using kernel functions, SVMs can model complex, non-linear relationships effectively. With a very large number of features the SVMs may become computationally expensive as can be seen observed in Figure 6.

Hyper parameters tuned:

Kernel – RBF, Linear, sigmoid, poly

C: 0.01: 100

Gamma: 0.001- 10

The best RMSE and MAE scores of 20.92 and 17.14 respectively were achieved with linear kernel.

### XGBoost

XGBoost is a gradient-boosted decision tree algorithm. It’s known for its efficiency and scalability in handling large datasets with many features. Boosting algorithms iteratively build models by correcting errors made by the previous ones. XGBoost does so using decision tree models. Thus allowing it to capture non-linear relationships and complex interactions. Regularization techniques built into XGBoost help prevent overfitting even when dealing with high-dimensional data. It supports sparse data well. Considering these aspect this is considered a good candidate algorithm for the regression problem here. The computational cost of the XGBoost model turned out to be very high relative to the other models as can be seen in [Figure 6]

### Hyperparameters

objective – squarederror

n\_estimators: 5 - 500

Best RMSE and MAE scores achieved 20.21 and 16.65 respectively with 5 estimators.

### Feedforward deep neural network

Deep Neural Networks (DNNs) are highly versatile models capable of learning complex patterns and non-linear relationships from large, high-dimensional datasets. They use multiple layers of neurons to create hierarchical representations. They can capture intricate interactions between the features. The best accuracy we see here is not the best amongst all the models. Though heavier models were also tried the validation accuracy didn’t improve beyond what the smaller models delivered at much less computational costs. One explanation for this could be that the data is fairly linear and is small in size which allows simpler models to perform equally well.

The pytorch based model that produced the best accuracy was

RegressionModel(

(layer1): Linear(in\_features=451, out\_features=16, bias=True)

(layer2): Linear(in\_features=16, out\_features=8, bias=True)

(layer3): Linear(in\_features=8, out\_features=8, bias=True)

(layer4): Linear(in\_features=8, out\_features=8, bias=True)

(output): Linear(in\_features=8, out\_features=1, bias=True)

)

Hyper parameters tuned:

Optimizers: Adam, RMSProp

Number of hidden layers: 1 - 3

Size of hidden layers: 4 - 1024

Number of Epochs to train: 20 – 500

Learning rate: 0.0001 – 0.1

Best RMSE: 19.82, and MAE: 16.7

# Findings and conclusions

The best accuracy (lowest **RMSE**) achieved was **18.64** while best **MAE** was **14.98**. That is the predicted price would differ from the actual listing by +/- $19. Since the price data has an overall **Standard Deviation** of **295** (even after the elimination of the outliers the Standard Deviation is 65), the RMSE and MAE values achieved by the models are quite good in comparison. Moreover, the **Mean Absolute Percentage Error** (MAPE) scores remain **below 2%.** Therefore, we can conclude that the models have achieved rather good regression accuracy even with the small dataset at hand.

In addition to the goal of predicting the price of the listings as accurately as possible, the study also started out with a couple of hypotheses:

1. That the Reviews written by the renters provide more valuable predictor of the favorability measure of a listing. The thought behind this was that numerical ratings often get done with default biases (pre-selected options, order of options, etc.) and, also, done relatively quickly. On the contrary, when someone spends time and effort to verbally provide a review it is likely to capture the actual favorability sentiment due to the likely effort put into verbalize the experience. What the study found [Figure 4] out was that the sentiment scores differ at the lower level of ranking while close the gap at the higher level of rankings. Moreover, the average minimum sentiment score is much higher (>2) compared to the review score (0) which could be possible either because the writeups represent the actual perceived value more accurately or those who tend to write the reviews are more favorably biased to the listing. These observations support the possibility of biases.
2. That the amenities provided in the listings can be a good predictor of the price – at least a few of them and we can identify them. This hypothesis too turned out to be true. As can be seen from the top 15 features [Figure 7] almost half of them are amenities. Some interesting insights can be gained by fine analysis of amenities choices and expanding the lists to convert them to features is a useful pre-processing step

Additional insights were also derived. For instance, the location, as defined by longitude and latitude, though influence significantly but is not the top criterion to decide the price.

This work demonstrates that the price can be fairly accurately judged or predicted given the data available for these listings.

Figure 5: Accuracy of each of the model

Figure 6: Comparative computational cost of the models considered in this work

Features in the descending order of importance for prediction as given by the random forest model.

Figure 7: Feature importance as discovered by the Random Forest model

# Future Work

This work focused on a very specific dataset of a single city in the US. The dataset, though representative of Airbnb listings in the US there are possibly regional quirks captured in this particular model which may not be so applicable to other cities. For instance, we can see that “indoor fireplace” is the second most important predictor of the price among the listed amenities. This could be a very specific feature of the cold northern climes of NY state whereas warmer places might have some other amenity more predictive of price. Despite that the framework proposed in this work is robust to be applied to any other such dataset.

There are certainly ways to further refine the data in the pre-processing steps that can reduce the number of features while still retaining the accuracy achieved. Applying techniques like Principle Component Analysis (PCA) is another approach to try here for reducing the number of features. However, PCA may improve the price prediction accuracy it could lose the insights into feature importance.

# References

1. [Kaggle dataset](https://www.kaggle.com/datasets/rhonarosecortez/new-york-airbnb-open-data?resource=download&select=reviews.csv) for AirBnb market listing in Albany, New York <https://www.kaggle.com/datasets/rhonarosecortez/new-york-airbnb-open-data/data>
2. Zervas, G., Proserpio, D., & Byers, J. W. (2017). The rise of the sharing economy: Estimating the impact of Airbnb on the hotel industry. *Journal of marketing research*, *54*(5), 687-705.
3. Hati, S. R. H., Balqiah, T. E., Hananto, A., & Yuliati, E. (2021). A decade of systematic literature review on Airbnb: the sharing economy from a multiple stakeholder perspective. *Heliyon*, *7*(10).
4. A. Zhu, R. Li and Z. Xie, "Machine Learning Prediction of New York Airbnb Prices," 2020 Third International Conference on Artificial Intelligence for Industries (AI4I), Irvine, CA, USA, 2020, pp. 1-5, doi: 10.1109/AI4I49448.2020.00007.,
5. A. Lektorov, E. Abdelfattah and S. Joshi, "Airbnb Rental Price Prediction Using Machine Learning Models," 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 2023, pp. 0339-0344, doi: 10.1109/CCWC57344.2023.10099266.
6. Rezazadeh Kalehbasti, P., Nikolenko, L., Rezaei, H. (2021). Airbnb Price Prediction Using Machine Learning and Sentiment Analysis. In: Holzinger, A., Kieseberg, P., Tjoa, A.M., Weippl, E. (eds) Machine Learning and Knowledge Extraction. CD-MAKE 2021. Lecture Notes in Computer Science(), vol 12844. Springer, Cham. <https://doi.org/10.1007/978-3-030-84060-0_11>
7. M. Mahyoub, A. A. Ataby, Y. Upadhyay and J. Mustafina, "AIRBNB Price Prediction Using Machine Learning," 2023 15th International Conference on Developments in eSystems Engineering (DeSE), Baghdad & Anbar, Iraq, 2023, pp. 166-171, doi: 10.1109/DeSE58274.2023.10099909.
8. Ghosh, Indranil;Jana, Rabin K.;Abedin, Mohammad ZoynulInternational, “An ensemble machine learning framework for Airbnb rental price modeling without using amenity-driven features”, International journal of contemporary hospitality management, 29 Aug 2023, Vol. 35, Issue 10, pages 3592 – 3611 ISSN: 09596119 DOI: 10.1108/IJCHM-05-2022-0562
9. Alharbi, Z. H. (2023). A Sustainable Price Prediction Model for Airbnb Listings Using Machine Learning and Sentiment Analysis. *Sustainability*, *15*(17), 13159. <https://doi.org/10.3390/su151713159>
10. S. Yang, "Learning-based Airbnb Price Prediction Model," 2021 2nd International Conference on E-Commerce and Internet Technology (ECIT), Hangzhou, China, 2021, pp. 283-288, doi: 10.1109/ECIT52743.2021.00068.
11. J. Dhillon et al., "Analysis of Airbnb Prices using Machine Learning Techniques," 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), NV, USA, 2021, pp. 0297-0303, doi: 10.1109/CCWC51732.2021.9376144.
12. Predict AirBnb Prices (Airbnb US 2020), https://www.kaggle.com/code/fireballbyedimyrnmom/predict-airbnb-prices/input
13. Predictive Price Modeling for Airbnb listings <https://www.deepakkarkala.com/docs/articles/machine_learning/airbnb_price_modeling/about/index.html>
14. Airbnb price prediction, <https://www.kaggle.com/datasets/stevezhenghp/airbnb-price-prediction>
15. Karagiannopoulos, M., Anyfantis, D., Kotsiantis, S. B., & Pintelas, P. E. (2007). Feature selection for regression problems. *Proceedings of the 8th Hellenic European Research on Computer Mathematics & its Applications, Athens, Greece*, *2022*.
16. Kolleck, A. S., & Teubner, T. (2024). How Ratings Shape the Supply on Sharing Economy Platforms: A Longitudinal Cohort Study on Listing Survival on Airbnb. *Information Systems Management*, 1–17. https://doi.org/10.1080/10580530.2024.2339234
17. J. Ting, A. D'Souza, S. Vijayakumar and S. Schaal, "Efficient Learning and Feature Selection in High-Dimensional Regression," in Neural Computation, vol. 22, no. 4, pp. 831-886, April 2010, doi: 10.1162/neco.2009.02-08-702.
18. H. Chamlal, A. Benzmane and T. Ouaderhman, "A Two-Step Feature Selection Procedure to Handle High-Dimensional Data in Regression Problems," 2023 International Conference on Decision Aid Sciences and Applications (DASA), Annaba, Algeria, 2023, pp. 592-596, doi: 10.1109/DASA59624.2023.10286637.
19. Dewi, C., & Chen, R. C. (2019). Random forest and support vector machine on features selection for regression analysis. *Int. J. Innov. Comput. Inf. Control*, *15*(6), 2027-2037.
20. Hall, M. A. (1999). *Correlation-based feature selection for machine learning* (Doctoral dissertation, The University of Waikato).
21. Vabalas A, Gowen E, Poliakoff E, Casson AJ (2019) Machine learning algorithm validation with a limited sample size. PLoS ONE 14(11): e0224365. <https://doi.org/10.1371/journal.pone.0224365>
22. Saleh, H., & Layous, J. A. (2022). *Machine Learning–Regression* (Doctoral dissertation, PhD thesis, Jan. 2022. doi: 10.13140/RG. 2.2. 35768.67842).
23. S. Ray, "A Quick Review of Machine Learning Algorithms," 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), Faridabad, India, 2019, pp. 35-39, doi: 10.1109/COMITCon.2019.8862451.
24. Ho, W. K. O., Tang, B. S., & Wong, S. W. (2020). Predicting property prices with machine learning algorithms. *Journal of Property Research*, *38*(1), 48–70. <https://doi.org/10.1080/09599916.2020.1832558>
25. Xu, X., & Zhang, Y. (2021). Second-hand house price index forecasting with neural networks. *Journal of Property Research*, *39*(3), 215–236. <https://doi.org/10.1080/09599916.2021.1996446>
26. Krämer, B., Nagl, C., Stang, M., & Schäfers, W. (2023). Explainable AI in a Real Estate Context – Exploring the Determinants of Residential Real Estate Values. *Journal of Housing Research*, *32*(2), 204–245. <https://doi.org/10.1080/10527001.2023.2170769>
27. Bilgilioğlu, S. S., & Yılmaz, H. M. (2021). Comparison of different machine learning models for mass appraisal of real estate. *Survey Review*, *55*(388), 32–43. <https://doi.org/10.1080/00396265.2021.1996799>
28. Jeremy Howard and Sebastian Ruder. 2018. Universal Language Model Fine-tuning for Text Classification. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). DOI: https: //doi.org/10.18653 / v1 / p18-1031
29. Samuel Kofi Akpatsa, Hang Lei, Xiaoyu Li, Victor-Hillary Kofi Setornyo Obeng, Ezekiel Mensah Martey, Prince Clement Addo and Duncan Dodzi FiawooOnline 2022. News Sentiment Classification Using DistilBERT. Journal of Quantum Computing Tech Science Press DOI: 10.32604/jqc.2022.026658
30. A. F. Adoma, N. -M. Henry and W. Chen, "Comparative Analyses of Bert, Roberta, Distilbert, and Xlnet for Text-Based Emotion Recognition," 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 2020, pp. 117-121, doi: 10.1109/ICCWAMTIP51612.2020.9317379.