## Exploring Housing Prices through Time Series Analysis and Finding the Most Accurate Prediction Model

Team Members: Tianhao Zhang, Yifan Chen, Yuchu Hou, Muyuan Ma, Xinnuo Li, Tianqi Qian

New York University

Professor Stanislav Sobolevsky

## Abstract

Accurate housing price prediction is crucial as it reflects a region's economic well-being and social fabric. Its impact is far-reaching, from government officials to homebuyers. We aim to leverage Python and machine learning to build a predictive model for housing prices in New York City, a dynamic and complex market. Using Pandas, Numpy, Matplotlib, Sklearn, and Statsmodel, we will process and analyze large datasets to uncover insights and patterns that impact housing prices. Through this project, we hope to develop a reliable and accurate model for predicting housing prices in New York City and push the boundaries of data mining and machine learning in the real estate industry. By exploring new techniques and methods, we aim to enhance our predictions' accuracy and practical value, and ultimately benefit all stakeholders in the housing market.

## 1. Introduction

Housing prices play a significant role in shaping a region's economic well-being and social fabric. Accurate housing price prediction is essential for government officials and prospective homebuyers. The real estate industry is constantly evolving, and it has become increasingly vital to use advanced tools and techniques to predict housing prices. In this regard, we aim to leverage the power of Python and machine learning to build a predictive model for housing prices in New York City, one of the most dynamic and complex real estate markets globally.

This project is mainly divided into two parts. One part focuses on analyzing housing prices using time series analysis. The other aims to find the model with the highest accuracy in predicting housing prices. The primary objective of this project is to develop a reliable and accurate predictive model for housing prices in New York City. To achieve this goal, we will process and analyze large datasets using popular data analysis tools. By analyzing historical housing data, we aim to uncover valuable insights and patterns that impact housing prices in the city. Our project will go beyond the standard approaches used in the real estate industry and explore innovative techniques and methods to enhance the predictions' accuracy and practical value. The outcomes of this research will have far-reaching implications for various stakeholders in the housing market.

In conclusion, our project aims to develop an accurate predictive model for housing prices in New York City using Python and machine learning. By leveraging innovative techniques and methods, we hope to enhance the predictions' accuracy and practical value, benefiting all stakeholders in the housing market. This research will be a significant contribution to the real estate industry's ongoing transformation towards data-driven approaches.

## 2. literature Review

In recent years, the use of machine learning techniques to predict housing prices has been gaining increasing attention. While various studies have been conducted in this field, a few stand out for their unique contributions. For example, Ali Soltani et al. (2019) proposed a framework that combines spatial and temporal features of housing data to build a predictive model using graph convolutional networks, which has shown superior performance compared to other state-of-the-art methods[[1]](#footnote-1). Adetunji et al. (2020) emphasized the importance of feature selection in improving the accuracy of housing price prediction using Random Forest[[2]](#footnote-2). Quang Truong et al. (2020) improved the accuracy of their housing price prediction model by combining linear regression and support vector regression, feature selection, and feature engineering[[3]](#footnote-3). Nur Shahirah Ja’afar (2020) conducted a comprehensive literature review of machine learning techniques for property price prediction and valuation, highlighting the significance of algorithm selection, feature selection, data preprocessing, and model evaluation[[4]](#footnote-4). Collectively, these studies demonstrate the potential of machine learning techniques in predicting housing prices, and the importance of incorporating spatio-temporal features, feature engineering, and graph convolutional networks to enhance the accuracy of predictive models. Researchers and practitioners interested in applying these techniques to the real estate industry can benefit from the insights provided by these studies. Nonetheless, the dearth of studies utilizing housing price data from New York City, coupled with the limited scope of previous research datasets, highlights the need for more comprehensive approaches. Our innovative project addresses this by leveraging a holistic dataset of the city to develop a model for predicting housing prices in New York City.

## 3. Data Description

### 3.1 [311 Service Requests](https://data.cityofnewyork.us/Social-Services/311-Service-Requests-from-2010-to-Present/erm2-nwe9)

The 311 Service Requests dataset contains information about non-emergency service requests made by New York City residents and businesses. This valuable resource allows users to track and analyze various aspects of city services, including response times, types of issues reported, and the geographic distribution of requests. Each record in the dataset corresponds to an individual service request and includes details such as the date and time the request was created, the type of issue (e.g., noise complaints, potholes, or illegal parking), the location of the request, and the current status of the request.

### 3.2 [Zip code Income Data](https://drive.google.com/file/d/1UmL6IcG-ipyEV26NA0h2f4ucGU9ut8cW/view?usp=share_link)

The Zipcode Income dataset provides a comprehensive overview of key income metrics for all zip code areas in the city. This dataset includes mean income, median income, and population data, offering valuable insights into the dynamics of the local economy of New York City neighborhoods.

### 3.3 [NYC Citywide Annualized Calendar Sales Update](https://data.cityofnewyork.us/City-Government/NYC-Citywide-Annualized-Calendar-Sales-Update/w2pb-icbu)

The NYC Annualized Calendar Sales dataset provides comprehensive information on property sales transactions across all boroughs of New York City. This dataset contains details on real property sales, including residential, commercial, and mixed-use properties, for the entire calendar year. Each record in the dataset represents a single property sale transaction and includes information such as the borough, building class category, property address, sale price, and sale date. Additionally, the dataset provides details on the building's characteristics, such as the total number of units, land and gross square footage, and the year built.

## 4. Methods and Analysis Process

### 4.1 Analyzing Housing Prices Using Time Series Analysis

#### 4.1.1 Data Preprocessing

Our project started with a series of data preprocessing and exploratory data analysis (EDA) steps. We first read the data from the “311 Service Requests” file and indexed it by the 'Zip' column. After filtering out data from New York City (Zip codes between 10000 and 11500) and converting the 'Zip' column data to a numeric format, we used the pd.pivot\_table() method to pivot the 311 service request data by 'Zip' and 'Complain' columns. We then created a new data frame named data311, which shows the number of complaints for each type in each Zip code. Next, we read the data from the income-related file and converted the data in the 'Mean' column to numeric format. We then removed any data with values less than or equal to 0. Moving on, we read the data from the file “NYC Citywide Annualized Calendar Sales Update” and indexed it by the 'ZIP CODE' column. We filtered out rows containing the keyword 'FAMILY DWELLINGS' in the 'BUILDING CLASS CATEGORY' column and removed any missing values. The data in the 'GROSS SQUARE FEET' and 'SALE PRICE' columns were converted to a numeric format, and we removed any data with values greater than 1e+04 in the 'PRICE PER SQFT' column and values less than or equal to 0 in the 'GROSS SQUARE FEET' and 'SALE PRICE' columns. Finally, we converted the data in the 'SALE DATE' column to date format and analyzed the data using visualization techniques.

#### 4.1.2 Time Series Analysis

Appropriate AR and MA terms were determined using autocorrelation and PACF plots, followed by training an ARIMA (2,1,2) model on the data from 2016 to 2020 and then predicting the housing price data for 2021. Finally, the prediction results were evaluated, and it was found that the ARIMA model had a relatively acceptable predictive performance.

### 4.2 Find Model with Highest Accuracy

#### 4.2.1 Normalize the Data

Next, we merged the data using the merge() method. We merged the three datasets mentioned in part 3 based on the same zip code. After merging, we performed some operations on the 'SALE DATE' column in the new dataset, including converting it to date format, selecting all data before 2021, and calculating the data length. We take the “SALE PRICE” as the prediction object and the rest columns as features. Then, we normalize the features. Normalizing data can improve the accuracy of machine learning models by eliminating measurement scale effects and reducing the impact of outliers. It also enables easier interpretation and comparison of data across different datasets.

#### 4.2.2 Data Split and PCA Dimensionality Reduction

We use the train\_test\_split() method to randomly split the dataset into training and testing data. Due to the high dimensionality of the data, we performed dimensionality reduction on the data using PCA. We calculated the cumulative explained variance ratio of the first n principal components to determine how many principal components to select as input features for the model. The final selection consists of 36 principal components that represent 90% of the information in the original dataset.

#### 4.2.3 Model Training

##### 4.2.3.1 linear regression model

First, we built a linear regression model with intercept using the OLS function in the statsmodels library based on our experience and use this model to make predictions. We also calculated in-sample and out-of-sample R-squared, MSE, and MAE values using both the original data and the data after PCA, to check the performance of the model.

##### 4.2.3.2 Neural Network Model

Then, we defined a more complex neural network model using PyTorch. The model consists of three fully connected hidden layers with ReLU activation functions and dropout layers between each layer to prevent overfitting. The model is trained using mean squared error loss and the Adam optimizer. The code prints out the loss and accuracy of the training and testing set for each training epoch, allowing users to monitor the model's training progress and performance. We also calculated in-sample and out-of-sample R-squared, MSE, and MAE values using both the original data and the data after PCA, to check the performance of the model.

##### 4.2.3.3 XGBoost Model

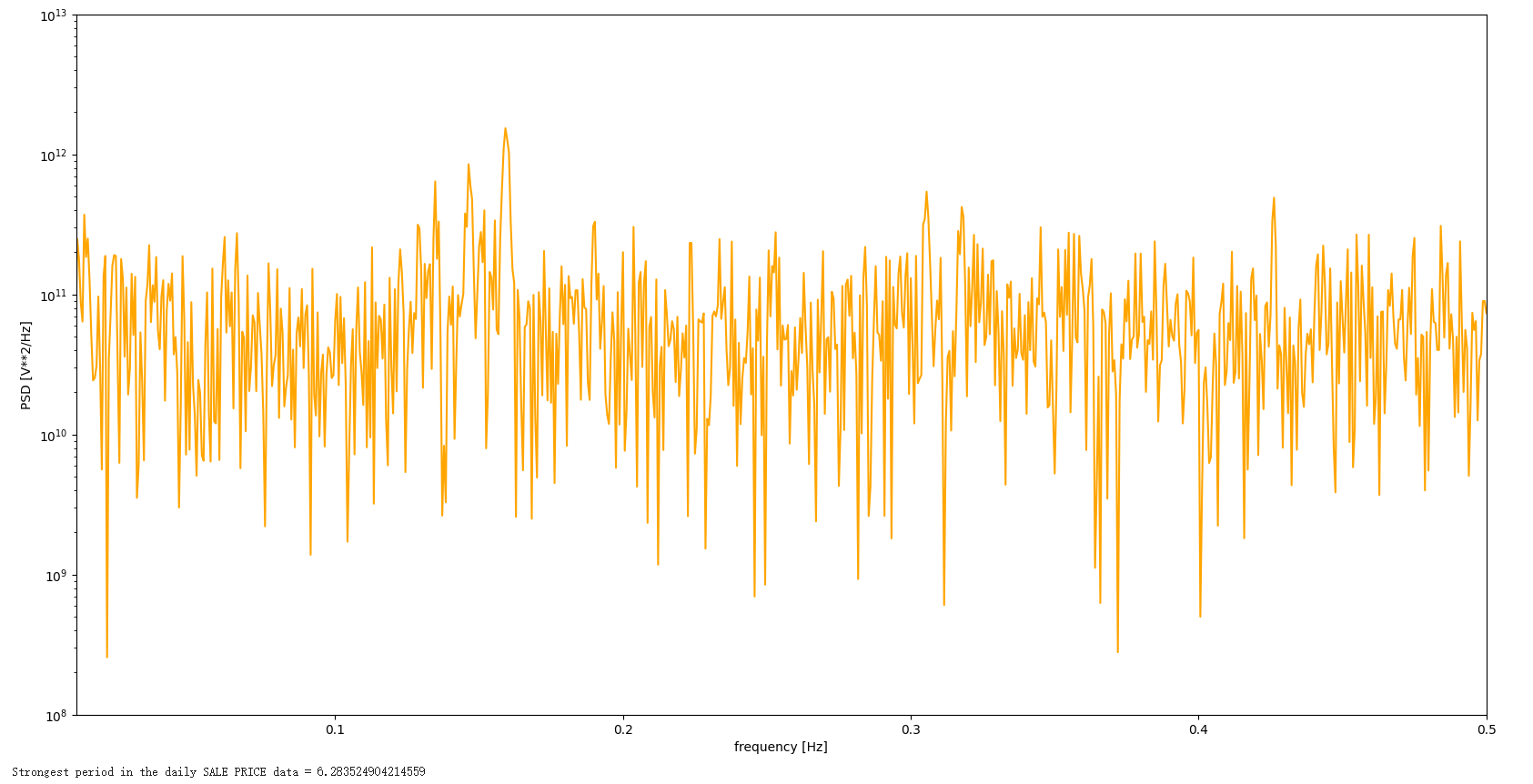
Due to the less excellent results from the first two models tested, we introduced the XGBoost model to improve our predictions' accuracy in-sample and out-of-sample R-squared, MSE, and MAE values using both the original data and the data after PCA, to check the performance of the model.

## 5. Results

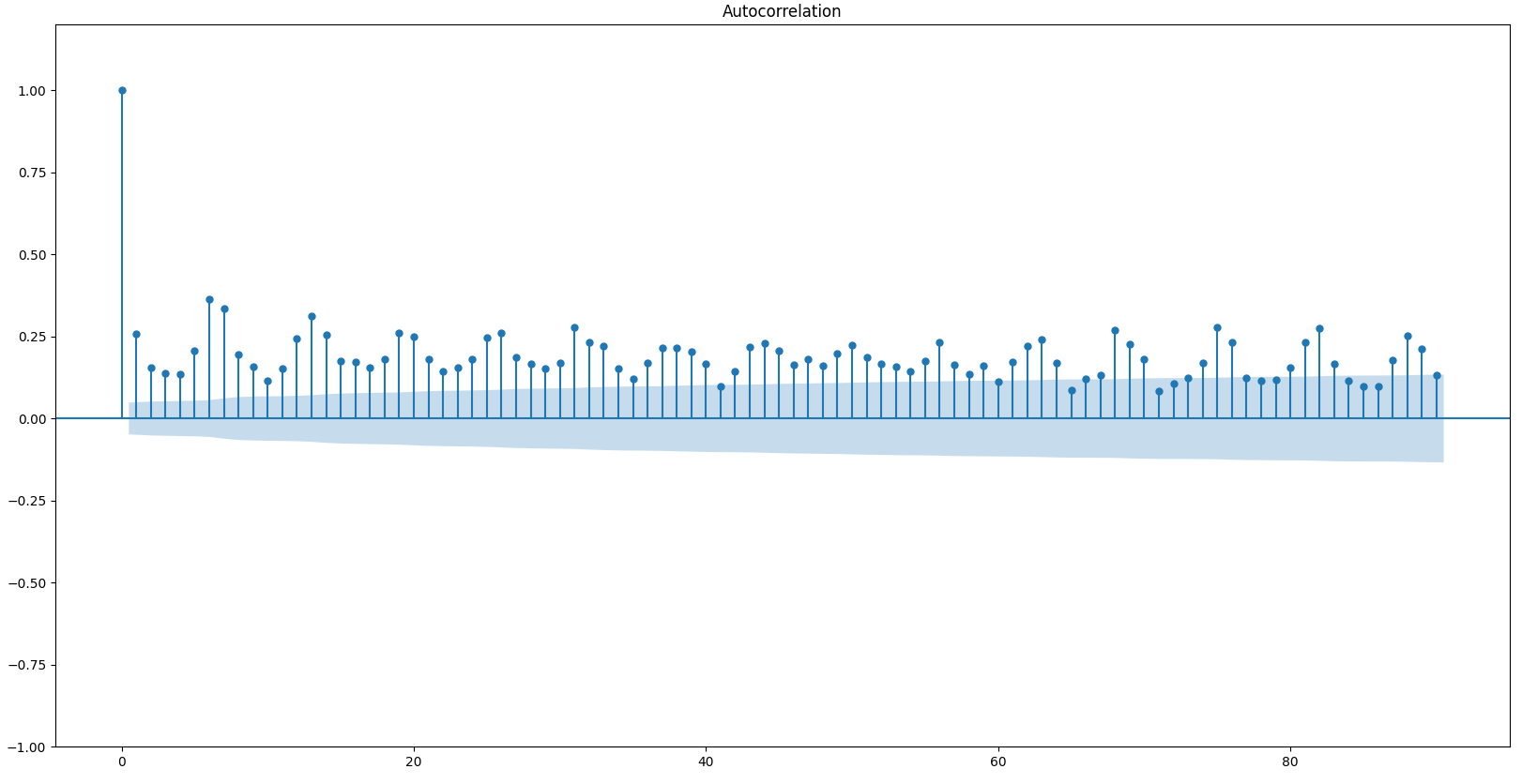
In this section, we will structure our discussion into four primary segments. The initial segment will present the outcomes of the Time Series Analysis conducted prior to data integration. Subsequently, the second segment will emphasize the results of implementing a Linear Regression model on the merged data, both with and without the application of Principal Component Analysis (PCA). The third segment will shift its focus towards the outcomes derived from the Neural Network model after data merging, again considering cases with and without PCA. Lastly, the final segment will delve into the output and performance evaluation of the XGBoost model.

### 5.1 Time Series

Initially, we attempted to identify an appropriate period by employing scipy.signal.periodogram and autocorrelation methods. Figures 5.1.1 and 5.1.2 depict the outputs of the periodogram and autocorrelation analyses, respectively. These results suggest that the most prominent period in the daily SALE PRICE data is approximately 7 days.



5.1.1 Result of Periodogram



5.1.2 Result of Autocorrelation

Subsequently, utilizing the ARIMA model, we trained our model on the price per square foot data from 2016 to 2020, and then generated forecasts and performed testing on the 2021 data. Figure 5.1.3 illustrates the results of these predictions. The performance metrics of this model include an out-of-sample Mean Absolute Error (MAE) of 184,953.53, an out-of-sample Root Mean Square Error (RMSE) of 244,673.58, and an out-of-sample R-squared (R2) value of -0.1031.

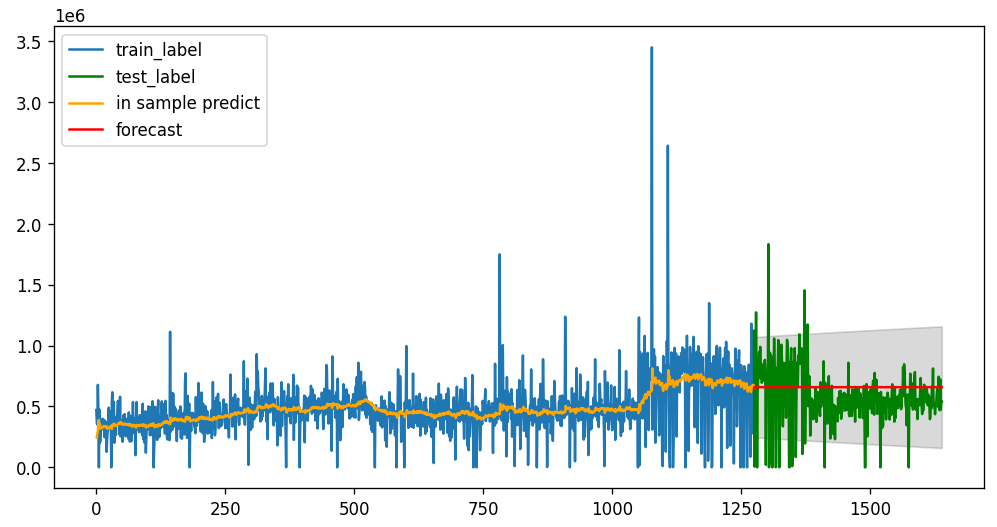
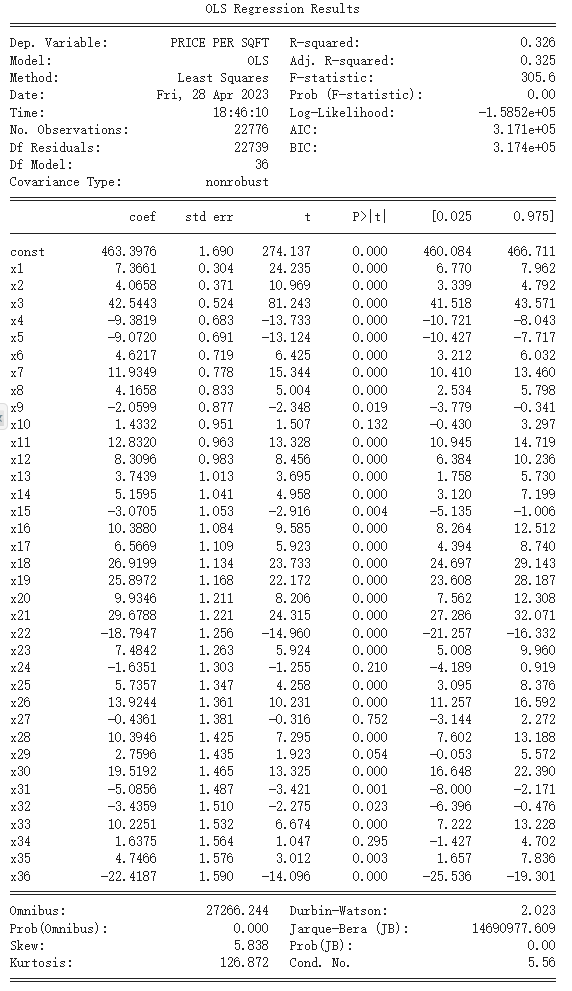
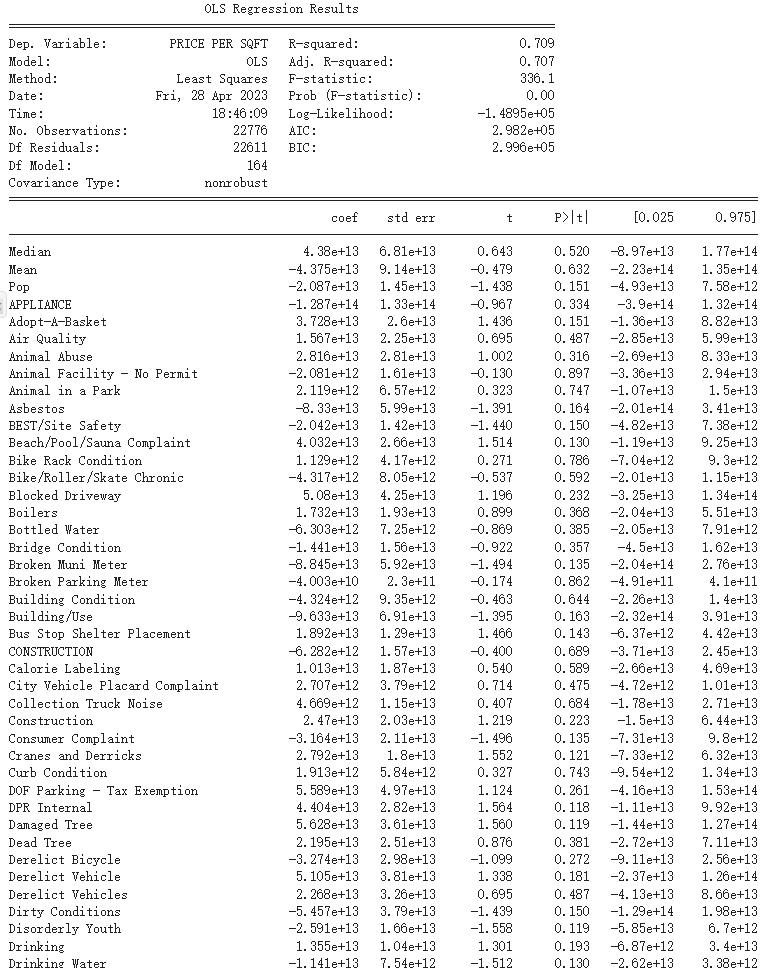


Figure 5.1.3 Prediction with ARIMA

### 5.2 Linear Regression

After merging and processing the data, we employed a Linear Regression model for predictions and explored the use of PCA to enhance the model. Figures 5.2.1 and 5.2.2 display the OLS model before and after PCA, respectively. In terms of performance metrics, the model before PCA exhibited an in-sample R2 score of 0.7091, an out-of-sample R2 score of -1.0035 × 10^18, a mean squared error of 8.7906 × 10^22, and a mean absolute error of 3,929,156,474.95. Conversely, the model after applying PCA showed an in-sample R2 score of 0.3261, an out-of-sample R2 score of 0.3432, a mean squared error of 57,530.02, and a mean absolute error of 151.70.



5.2.1 OLS before PCA 5.2.2 OLS after PCA

### 5.3 Neural Network

In this section, we implemented a Neural Network model using the same process. Initially, we examined the model without PCA, consisting of an input layer, three hidden layers with 128, 64, and 32 neutrons respectively, an output layer, ReLU activation functions, and dropout layers with a rate of 0.2. We used a batch size of 64, shuffled the training data, and trained the model for 100 epochs. The in-sample R2 score was 0.7560, and the out-of-sample R2 score was 0.7542, with MSE and MAE values of 21532.35 and 95.55, respectively.

Next, we applied PCA for dimensionality reduction, retaining 36 principal components. After transforming the training and testing datasets using the PCA model, we employed the same architecture and settings. The model was trained for 100 epochs and evaluated on the test dataset at the end of each epoch. The in-sample R2 score was 0.3550, and the out-of-sample R2 score was 0.3339, with MSE and MAE values of 58351.78 and 152.15, respectively.

### 5.4 XGBoost

To further enhance accuracy, we applied the XGBoost algorithm, experimenting with both the original data and the data transformed using PCA. For the XGBoost model without PCA, we obtained an MSE of 1537.28, an MAE of 8.82, an in-sample R2 score of 0.9995, and an out-of-sample R2 score of 0.9825. For the XGBoost model with PCA, we achieved an MSE of 55543.89, an MAE of 148.46, an in-sample R2 score of 0.7195, and an out-of-sample R2 score of 0.3659.

## 6. Conclusion

In conclusion, this project has demonstrated the effectiveness of utilizing diverse analytical methods for housing price analysis in New York City. By employing time series analysis, PCA, linear regression, and neural network analysis, we were able to uncover complex relationships and trends in the housing market. The model’s performance declined after applying PCA. A possible explanation is that some less important features, removed by PCA, are still vital for the model, indicating the pricing model is more complex than it seems to be. The incorporation of the XGBoost model further enhanced the predictive capabilities, yielding unprecedented accuracy in forecasting housing prices.

Our findings revealed key factors influencing housing prices and their relative importance. The study also highlighted the potential of machine learning techniques, such as XGBoost, in improving the performance of traditional statistical models.

Considering these results, we recommend that future research should continue to explore the integration of advanced machine learning algorithms with traditional analytical methods to further improve the prediction of housing prices. Additionally, as new data becomes available, it is crucial to reevaluate and update the models to ensure their continued relevance in the ever-changing real estate market.

Ultimately, this project has showcased the power of data-driven approaches in providing a comprehensive understanding of the housing market in New York City. By leveraging such knowledge, stakeholders can make more informed decisions to optimize investments, guide urban planning, and shape policies to promote sustainable and equitable growth in the city's housing market.

## 7. Tables and Figures

Figure 1: 5.1.1 Result of Periodogram, page 6

Figure 2: 5.1.2 Result of Autocorrelation, page 6

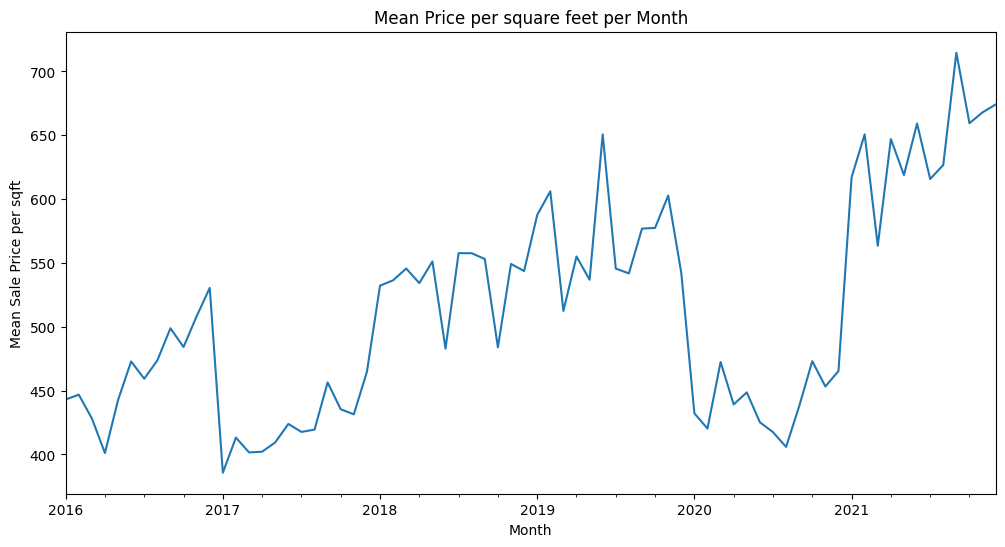
Figure 3: Figure 5.1.3 Prediction with ARIMA, page 6

Figure 4: 5.2.1 OLS before PCA, page 7

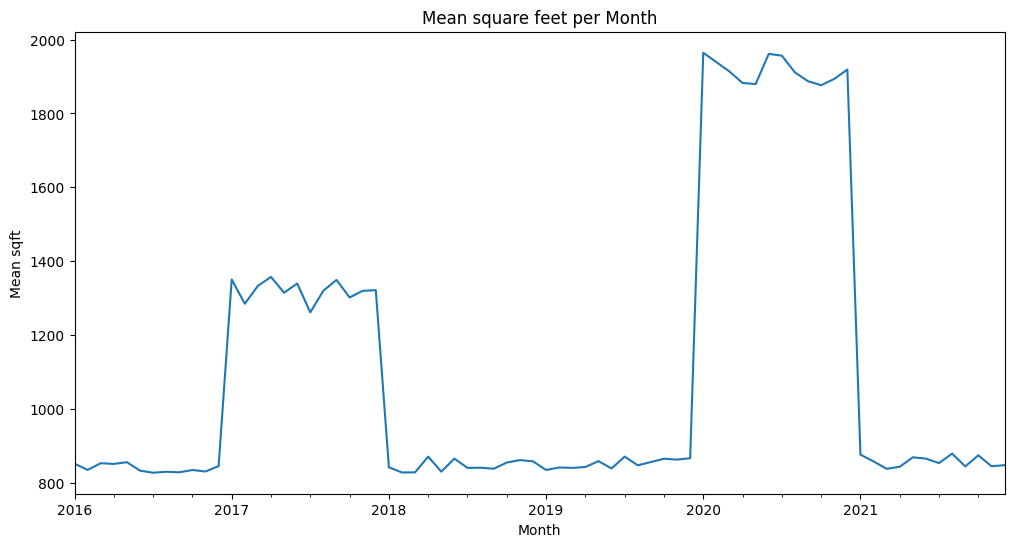
Figure 5: 5.2.2 OLS after PCA, page 7

## 8. Appendix

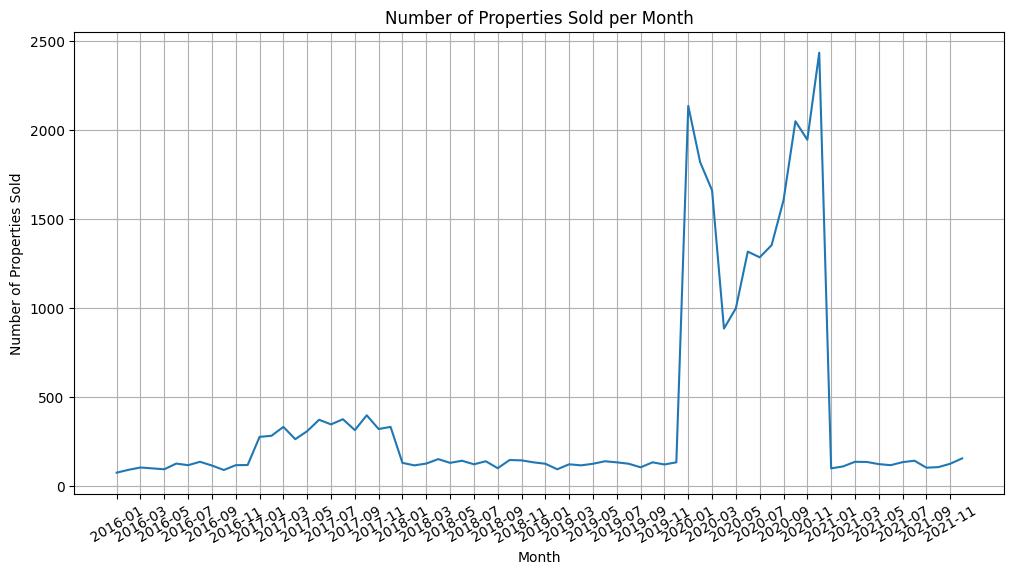
The visualization of the mean price per square feet per month from 2016 to 2021:



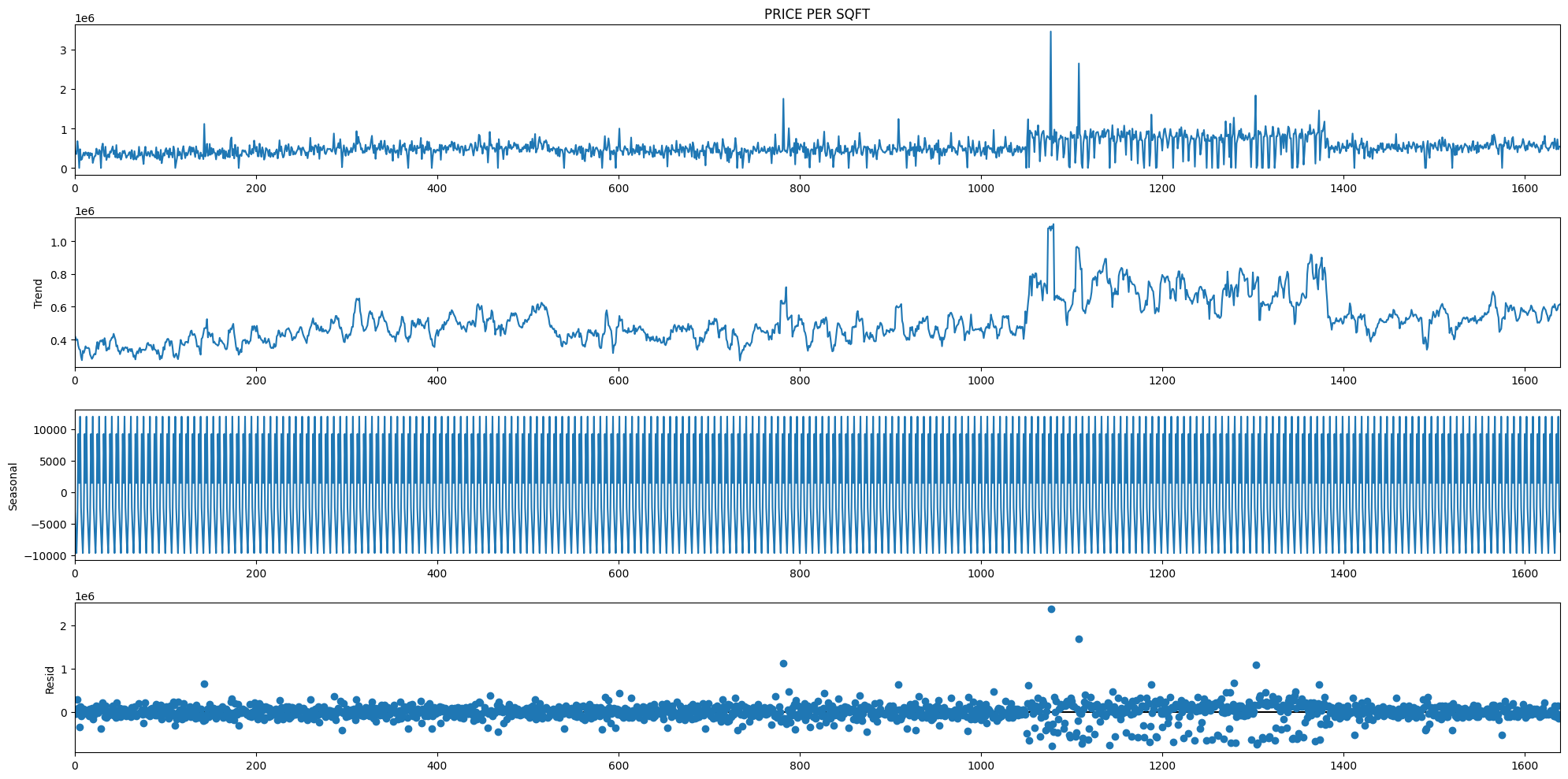
The visualization of the mean square feet of houses per month from 2016 to 2021:



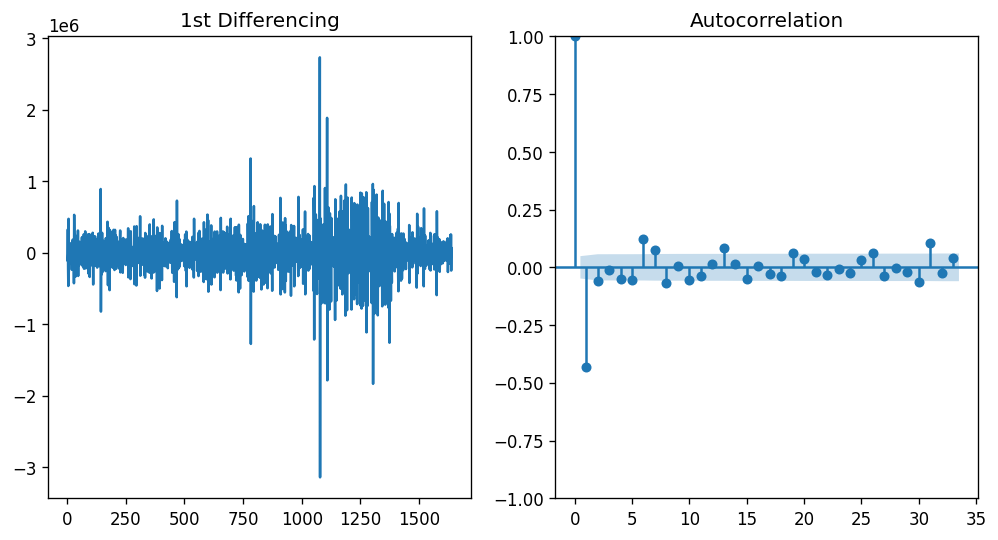
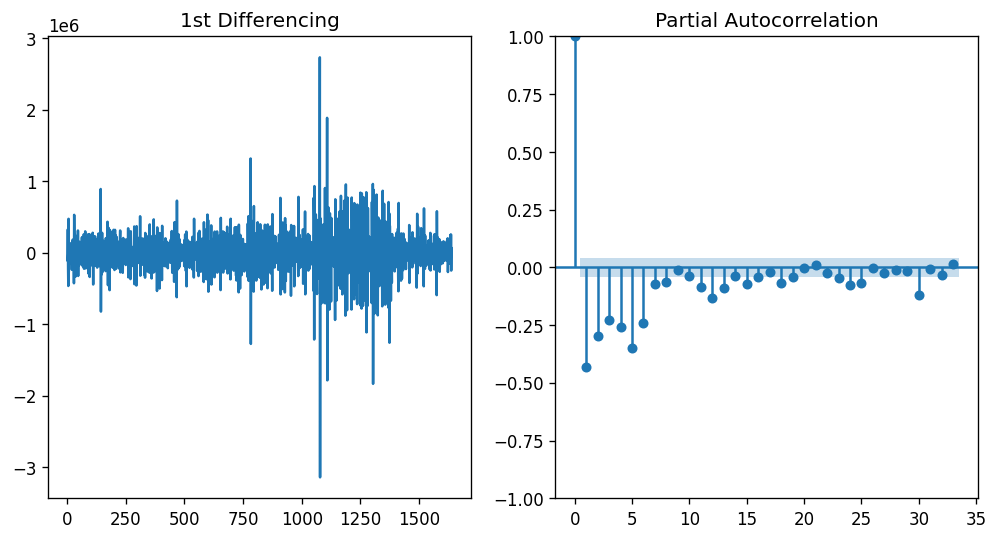
The visualization of monthly sold properties:



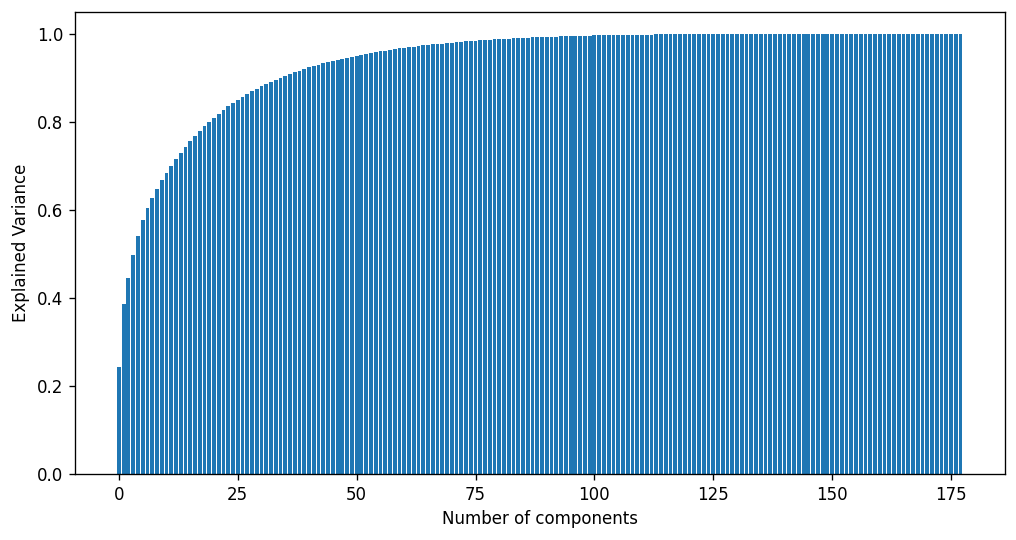
A visualization of the seasonal decomposition analysis on the first five years of the data, for the original time series, the trend component, and the residual component plots, there is a significant anomaly in the fourth year:



Fit PACF and ACF plots using the first differencing model to determine the AR and MA terms. From the PACF graph, 2 AR is suggested and from the ACF plot, 2 MA is suggested.



The PC decomposition plot shows the proportion of total variance explained by each of the 36 principal components retained from applying PCA to the high-dimensional dataset.



Tables of results R^2, MSE, MAE for the linear regression and neural network before and after PCA:

|  |  |  |  |
| --- | --- | --- | --- |
|  | | Linear Regression | Neural Network |
| Before PCA | R2 | -1.00e+18 | 0.75 |
| MSE | 8.79e+22 | 2.15e+4 |
| MAE | 3.93e+9 | 95.54 |
| After PCA | R2 | 0.34 | 0.33 |
| MSE | 5.75e+4 | 5.84e+4 |
| MAE | 1.52e+2 | 152.15 |

## Reference

1. Soltani, A., Heydari, M., Aghaei, F., & Pettit, C. J. (2022). Housing price prediction incorporating spatio-temporal dependency into machine learning algorithms. Cities, 131, 103941.

2. Adetunji, A. B., Akande, O. N., Ajala, F. A., Oyewo, O., Akande, Y. F., & Oluwadara, G. (2022). House price prediction using random forest machine learning technique. Procedia Computer Science, 199, 806-813.

3. Truong, Q., Nguyen, M., Dang, H., & Mei, B. (2020). Housing price prediction via improved machine learning techniques. Procedia Computer Science, 174, 433-442.

4. Ja’afar, N. S., Mohamad, J., & Ismail, S. (2021). Machine learning for property price prediction and price valuation: a systematic literature review. Planning Malaysia, 19.

1. Soltani, A., Heydari, M., Aghaei, F., & Pettit, C. J. (2022). Housing price prediction incorporating spatio-temporal dependency into machine learning algorithms. Cities, 131, 103941. [↑](#footnote-ref-1)
2. Adetunji, A. B., Akande, O. N., Ajala, F. A., Oyewo, O., Akande, Y. F., & Oluwadara, G. (2022). House price prediction using random forest machine learning technique. *Procedia Computer Science*, *199*, 806-813. [↑](#footnote-ref-2)
3. Truong, Q., Nguyen, M., Dang, H., & Mei, B. (2020). Housing price prediction via improved machine learning techniques. Procedia Computer Science, 174, 433-442. [↑](#footnote-ref-3)
4. Ja’afar, N. S., Mohamad, J., & Ismail, S. (2021). Machine learning for property price prediction and price valuation: a systematic literature review. Planning Malaysia, 19. [↑](#footnote-ref-4)