# **Real Estate Valuation Decision-Making System Using Machine Learning and Geospatial Data**

**Ahmet Yagmur**Department of Business Information Systems, Hochschule Furtwangen University, Robert-Gerwig-Platz 1, 78120 Furtwangen im Schwarzwald

## Abstract

Forecasting and analyzing the real estate market is crucial globally due to the presence of investors, prospective property owners, and sellers motivated by various reasons. These motivations are universal and can arise anywhere on the globe. However, specific factors such as geographical and geo-spatial locations, geological attributes, and human foresight regarding the future value of real estate make this a challenging research area. With an abundance of scientific data available, it is indeed feasible to predict prices or establish reasonable valuation intervals. While existing studies have integrated multiple data sources, including house imagery [8, 3] and economic indicators [5], this article will focus solely on geo-spatial data. Economic conditions, property conditions, and construction years will not be examined in this discussion. Although research has acknowledged the predictive capabilities of geo-spatial data, there is a scarcity of studies that isolate its unique aspects. This research aims to fill that gap by exploring how geo-spatial features can be processed and utilized within machine learning algorithms to inform accurate real estate value predictions. Employing machine learning techniques, such as Random Forest Regressor for price prediction and XGBoost Regressor for valuation intervals, this study analyzes geo-spatial data from the 2024-2025 German real estate market. The prediction pipeline computes geo-related scores, like distances to the nearest airport and city center, before running the models. The findings intend to enhance decision-making for real estate stakeholders by offering insights into accurate, geo-spatially informed valuations. This research aspires to inspire future studies incorporating additional factors for a comprehensive understanding of the real estate market.

**Keywords:** Real Estate Price Prediction, Machine Learning in Real Estate, Geo-spatial Data, Spatial Data Analysis, Random Forest Regression, XGBoost Regression, German Real Estate Market, Big Data

## 1. Introduction

### 1.1 Problem Definition

Real estate valuation is a complex and often unreliable process, primarily due to the challenge of determining an accurate market value for properties. Inaccurate valuation can cause disputes and financial losses, and can impede the success of a real estate transaction [7]. Furthermore, incorrect valuations can distort market trends, contributing to inflated property prices or underpriced properties, ultimately affecting the overall stability of the real estate market.

Traditional valuation methods, such as comparative market analysis (CMA) or appraisals based on expert judgment, are often subject to human biases and errors. These approaches may overlook crucial factors or fail to adapt quickly to market fluctuations, making them prone to inaccuracies. Additionally, market dynamics can change rapidly, and relying on historical data or limited geographic scope can lead to outdated and unreliable valuations.

The increasing availability of geospatial data, coupled with advancements in machine learning techniques, offers an opportunity to address these limitations. By incorporating various data sources, such as property characteristics, location information, and proximity to key amenities, machine learning models or automated valuation models (AVMs) can provide a more robust and objective approach to property valuation by analyzing large volumes of data [9].

For example,  
"The inability to accurately value properties leads to financial inefficiencies in the real estate market. Leveraging geospatial data and machine learning algorithms provides a novel approach to address this issue."

### 1.2 Project Objective

This project seeks to automate and enhance the real estate valuation process by using data science techniques and geospatial analysis tools. The primary objective is to develop a machine learning-based model that can predict real estate values more accurately than traditional methods.

By integrating geospatial data (such as location, proximity to amenities, and environmental factors) and machine learning algorithms, the project aims to achieve higher precision in estimating property prices. This approach also aims to minimize human error, providing a data-driven, objective, and scalable solution that can adapt to changing market conditions.

In early years, some of scientists tried to build a machine learning model with combination of geographic information system (GIS) [1]. That is accepted as early attempt of ML real estate valuation [6].

The project will focus on developing a predictive model that can be applied to a variety of real estate markets, taking into account local trends, property features, and geospatial relationships. This will ultimately result in more reliable pricing for both investors and buyers, contributing to better decision-making in the real estate sector.

### 1.3 Research Question and Hypothesis

Research Question  
The central research question guiding this project is:

"Can machine learning models trained with geospatial data outperform traditional methods in real estate valuation? Additionally, how reliable is the price interval valuation produced by the machine learning model?"

This question explores the potential of machine learning models to generate more accurate and reliable price estimates for real estate properties, while also examining the consistency of these predictions within specific price intervals.

Hypothesis  
The hypothesis underlying this research is that:

"Machine learning models, when applied to real estate valuation using geospatial data, will deliver higher accuracy compared to traditional price interval valuation methods."

Traditional methods often involve simple models that fail to capture the complexities of location, amenities, and other geospatial factors. By leveraging machine learning, which can process and analyze large datasets with many variables, the expectation is that these models will provide more reliable and accurate property valuations. This hypothesis will be tested by comparing the performance of machine learning models with traditional valuation techniques across different property types and locations.

## 2. Data and Tools

### 2.1 Data Source and Types of Data

The dataset used in this project was sourced in result of web scraping.. The dataset includes key features that are essential for real estate valuation:

Price: The market value of each property, which is the primary target variable for the valuation model.

Land Size: The size of the property in square meters, which plays a crucial role in determining the overall value.

Number of Rooms: The number of rooms in the property, providing insights into the property's layout and potential market appeal.

Location Information: The geographic coordinates and addresses, which are critical for assessing proximity to amenities such as schools, transportation, and commercial areas.

|  |  |  |
| --- | --- | --- |
| ad\_id | Int | Unique ID for all records. |
| street | Str | The street name of the real estate. |
| city\_code | Int | The zip code and the city name of the real estate. |
| price | Float | The price of the real estate. |
| number\_of\_rooms | Int | The number of rooms of the real estate. |
| living\_area | Float | The size of living area of the real estate. |
| land-size | Float | The size of land area of the real estate (if exist). |
| URL | Str | The URL of the real estate ad. |
| city\_id | Int | Unique city ID for all the unique city, derived for the dataset. It does not have any relation with reality. |
| city\_score | Float | The city score calculated with statistical methods |
| population | Int | The population of the city that real estate located. |
| geo\_spatial | Int | The score of amenities around the real estate |
| center\_distance | Float | The score of closeness of the real estate to the nearest city center |
| airport\_distance | Float | The score of closeness of the real estate to the nearest airport |

Table 1: Columns and Explanations

In the context of real estate price prediction, the effectiveness of machine learning models depends heavily on the nature and quality of the data used. This study utilizes three primary types of data: structured data, and geospatial data. Each type plays a critical role in building a comprehensive and accurate predictive model.

### 2.2 Data Cleaning Process

In order to prepare the data for analysis and model training, several data cleaning procedures were performed:

Handling Missing Values: Missing values in the dataset were addressed through imputation, where reasonable assumptions were made to fill the gaps, or, in some cases, the rows with too many missing values were removed to avoid introducing errors into the analysis.

Categorical Variable Processing: For categorical variables such as city names, encoding techniques were applied. This involves converting city names into numerical representations, making it easier for machine learning models to interpret and process the data. Various encoding methods such as one-hot encoding or label encoding were considered based on the structure of the data.

Outlier Detection: The dataset was also examined for any extreme outliers in variables such as price or land size, which could distort the model's performance. Techniques like the IQR method or z-score calculation were applied to detect and remove or adjust outliers.

Normalization and Scaling: Continuous features like land size and price were normalized or scaled to ensure that the data was within a similar range, which helps improve the performance of certain machine learning algorithms.

A screenshot of a computer

Description automatically generated  
Figure 1: A part of the CSV file of ads  
  
Some of the cleansing done in the CSV. And it is significant to see the what is wrong and what could be occurred as a problem when dealing the dataset. The most important thing can be mentioned as: setting the language encoding “utf-8-sig” for all the reads and writes functions in python web scaring to CSV. Because of the fact that German language has some special letters in German alphabet (ß, ö, ä, ü can be given for instance).

Also all Unknowns have to be deleted. That occurs when the address is not found due to some typos when the address is written into CSV or directly from the ads that on the world wide web.

The numerical with decimal part are also a topic for confusion. Due to the fact that CSV and Python language interpret the decimal separator and thousand separator differently, means one of them has default “,” thousand separator and the other has “.”. The confusion comes from the situation that there is no universal rule for that and every country uses their own logic such as the USA uses “,” as thousand separator and Türkiye uses “.” as thousand separator.

2.3 Structured Data  
Structured data refers to highly organized, numerical information that fits neatly into tables or databases. In this research, structured data includes variables such as property prices, size of living area, size of land area and number of rooms. These features are quantifiable and serve as the backbone of almost all of real estate valuation models, offering direct correlations with property value. For instance, properties with larger size of living area or larger size of land area, typically fetch higher prices than those which has smaller size of the attributes.

A screenshot of a computer screen

Description automatically generated

Figure 2: Correlation Matrix of Key Features

This type of data is well-suited for machine learning algorithms because it allows for straightforward feature extraction and analysis. Furthermore, structured data provides the foundational inputs for models such as Random Forest and XGBoost, which rely on numerical features directly or categorical values that turned into numerical format for training and prediction. However, the predictive power of structured data is limited without the inclusion of more contextual variables, which are addressed through geospatial and textual data.

### 2.4 Geospatial Data

Geospatial data is a critical component in modern real estate valuation, offering insights into the geographic and locational attributes of a property. This data includes street, zip code, city name, and some other city and state identity determination number are derived for specifically for the dataset of the real estate. There are four types of geospatial data in the dataset. They are saved as numerical scores to corresponding addresses. They are, City Score, Geospatial Score, Distance to City Center Score and Distance to Airport Score. City Score: A score that is calculated with statistical analysis on mean real estate prices for each city that be part of the dataset. Geospatial Score: Distances to key amenities (super markets, convenience stores, shopping centers, Variety Stores, Parks, restaurants), and proximity to public transportation hubs or major stations. The same stations (arrivals and departures are counted as one station). Distance to City Center Score: A score that is calculated upon the distance from the nearest city center in km. Distance to Airport Score: A score that is calculated upon the distance from the nearest city center in km. However, the In particular cases particular cities has a small airport serving for small plane or educational and hobby purpose flights. The influence of geospatial data is intense. That is the reason expectance from the geospatial data to impact real estate values cannot be neglected.

|  |  |  |
| --- | --- | --- |
| Supermarket, Convenience Store, Variety Store | 100 m | 20 |
| 300 m | 15 |
| 600 m | 10 |
| 1000 m | 5 |
| Bus Stop, Train Station | 200 m | 25 |
| 500 m | 15 |
| 1000 m | 5 |
| Park, Fast Food | 100 m | 10 |
| 300 m | 7 |
| 1000 m | 5 |

Table 2: Geospatial Scoring Logic

A graph of a distribution of city score

Description automatically generated

Figure 3: Distribution of City Score

That distribution in Figure 3 indicates that there is a density on city score 2.5 to 4. Since the same amount of the real estates have been taken from 233 cities, most of on scope city has a city score 2.5 to 4.

A graph of a distribution of a number of people

Description automatically generated

Figure 4: Distribution of Distance to Airport Score

Figure 4 indicates that, most of the real estates has a location far from airport. In contrast, there are some which are pretty close to airport. However, those are small airport for hobby and education purpose.

A graph of a distribution of a number of people

Description automatically generated

Figure 5: Distribution of Distance to Center Score

According to Figure 5 the most of the real estates located in around the center and in a close diameter. However there are minority of real estates which located rural areas

A graph of a distribution of geo spatial score

Description automatically generated

Figure 6: Distribution of Geo Spatial Score

According to distribution in Figure 6, it confidently can be said, that majority of the real estates have at least decent amount of amenities and public services provided by German Government nearby.

A graph of a distribution of land area

Description automatically generated

Figure 7: Distribution of Land Area

From the distribution shown in Figure 7, all of the real estates used to train the model, have an special area for home owner to use, there can be a garage or a green field.

A graph of distribution of living area

Description automatically generated

Figure 8: Distribution of Living Area

Depends on the sample visualized on the graph in Figure 8 indicates, in Germany real estates have 125 m2 of living area size, most frequently.

A graph of a distribution of a number of objects

Description automatically generated with medium confidence

Figure 9: Distribution of Price per m2

As shown in the Figure 9, in German real estate market, The distribution appears fairly symmetrical with a slight tail to the right (positive skew), which suggests a small number of very high-priced properties affecting the average. Due to the wide variability in price per square meter, the MAE metric remains relatively high, even after thorough data cleansing.

Moreover, geospatial data enhances the ability to create location-specific insights, such as analyzing regional price trends or identifying hotspots for real estate development. When integrated with structured data, geospatial variables significantly boost the predictive accuracy of machine learning models by capturing spatial dependencies often overlooked in traditional valuation methods.  
  
This research aims to develop a robust and comprehensive valuation model by combining two types of data that are merged into the dataset: structured and geospatial data. Each data type complements the other to create a comprehensive data set covering both quantitative and qualitative aspects of real estate valuation, providing more accurate and actionable predictions.

### 2.5 Tools Used

This project employed a variety of tools and technologies for data analysis, machine learning, web scraping, and user interface development:

Python and Open-Source Software Libraries

Python, along with its vast collection of open-source libraries, provides considerable benefits in data cleaning, analysis, and modeling, especially regarding real estate valuation. The flexibility of Python and the presence of specialized libraries render it an economical and adaptable option for managing intricate datasets and creating strong machine learning models.

Cost-Effectiveness: Open-source libraries like Pandas, NumPy, Tkinter, Scikit-learn, and XGBoost remove the necessity for costly proprietary software, allowing researchers and practitioners with constrained budgets to access advanced data science methods. The democratization of resources promotes innovation and enables quick model prototyping without substantial financial obstacles.

Flexibility: The libraries in Python offer unmatched versatility for data manipulation and analysis. For instance, Pandas provides robust data structures for managing and sanitizing extensive datasets, whereas NumPy enables effective numerical calculations. This adaptability is essential when managing the varied and heterogeneous data types commonly found in real estate valuation, such as structured and geospatial information.

Integration and Extensibility: The smooth incorporation of diverse Python libraries facilitates the development of efficient workflows. For example, data may be refined and preprocessed through Pandas and NumPy, geospatial attributes can be created using Geopy or GeoPandas, and machine learning models can be constructed and optimized with Scikit-learn or XGBoost. This cohesive strategy guarantees that every phase of the data pipeline is fine-tuned for efficiency and precision.

Tuning Hyperparameters using GridSearchCV: Improving the precision of machine learning models is crucial for dependable real estate assessment. Methods for hyperparameter tuning, like GridSearchCV, are essential for enhancing model performance by methodically exploring the optimal combination of hyperparameters.



Figure 10: The result of GridSearchCV

GridSearchCV Overview: GridSearchCV is a technique provided by Scikit-learn that simplifies the hyperparameter optimization procedure. It comprehensively explores a defined set of hyperparameters, evaluating each combination via cross-validation to identify the optimal configurations that improve model performance.

Improving Model Accuracy: By tuning hyperparameters, GridSearchCV helps increase the forecasting accuracy of machine learning models. For example, in Random Forest models, parameters such as the number of trees, maximum depth, and minimum samples for splitting can be adjusted to reduce overfitting and improve generalization. Similarly, modifying parameters like learning rate, boosting rounds, and tree depth in XGBoost models can improve prediction accuracy.

Efficiency and Robustness: GridSearchCV not only improves accuracy but also contributes to the robustness of the models. By validating each hyperparameter combination through cross-validation, it ensures that the model performs well across different subsets of the data, thereby reducing the likelihood of overfitting and increasing the reliability of the predictions.  
  
Implementation in Research: In this study, GridSearchCV was employed to optimize the hyperparameters of both Random Forest and XGBoost models. This meticulous tuning process ensured that the selected models achieved the best possible performance metrics, such as Mean Absolute Error (MAE) and R² scores, thereby enhancing the overall effectiveness of the real estate valuation framework.

Python  
Python is the primary programming language used for data manipulation, model building, and analysis. Python's extensive libraries make it an ideal choice for this data science project. Libraries like Pandas and NumPy were utilized for data processing, while Matplotlib and Seaborn were used for visualizing results.

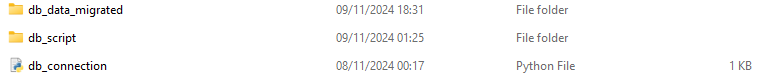
Scikit-learn  
Scikit-learn, a robust machine learning library, was used for training and evaluating various algorithms like Linear Regression, Random Forest, and XGBoost. It provides an intuitive interface for model training, evaluation, and tuning, which was crucial for implementing machine learning-based real estate valuation.

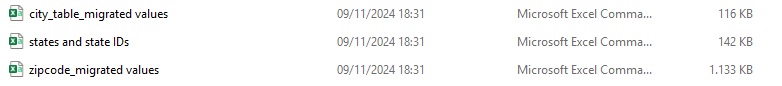
Geopandas and Matplotlib  
Geopandas was used for handling and analyzing geospatial data, as it is built specifically for working with geographical information. This was essential for the project, where spatial information plays a key role in understanding location-based patterns. Matplotlib was used to visualize geographical distributions and property data to derive valuable insights.

SQL  
SQL was used to store and manage large datasets efficiently. It allowed for fast querying and manipulation of the data, facilitating easy access to relevant property details for model training and analysis.

A screenshot of a computer

Description automatically generated  
Figure 11: A table from Database

  
Figure 12: Database files

  
Figure 13: Migrated .CVSs

A number of days and months

Description automatically generated with medium confidence  
Figure 14: Database scripts

A screenshot of a computer screen

Description automatically generated  
Figure 15: Database design last version

Tkinter: Tkinter was employed to create a user-friendly graphical interface for interacting with the model. This interface allowed users to input data and easily view predictions without directly engaging with the underlying code, improving the accessibility of the model.

A screenshot of a computer

Description automatically generated  
Figure 16: Price interval predictions

A screenshot of a computer

Description automatically generated  
Figure 17: Price predictions

Selenium and Web Scraping  
Selenium was utilized for automating web browser actions and scraping data from real estate websites, particularly Immowelt. This allowed the extraction of relevant real estate ad information, such as prices, locations, and property sizes, which are essential for building the valuation model. Additionally, threading was employed to handle multiple requests simultaneously, which sped up the scraping process significantly. The following web scraping tools were used:

Selenium: To automate browser interactions and scrape dynamic content from real estate websites.

WebDriver Manager: This tool automatically manages the necessary ChromeDriver for Selenium, ensuring seamless operation across various environments.

Threading: Multiple threads were used to simultaneously scrape different zip codes, improving the efficiency of the data collection process.

CSV (Comma-Separated Values)  
Data Storage: CSV files were used to store and exchange raw data due to their simplicity and compatibility.

Intermediary Format: Processed data and intermediate outputs were saved as CSV files for seamless use with other tools.

Excel  
Data Inspection and Cleansing: Excel’s filtering, sorting, and conditional formatting features helped identify errors, missing values, and duplicates.

Basic Analysis: Built-in formulas were used for simple calculations like averages and counts to validate data.

Data Organization: Pivot tables and multiple sheets were used to structure data during cleansing.

VBA (Visual Basic for Applications)  
Automation: VBA scripts automated repetitive tasks such as merging CSV files, flagging duplicates, and standardizing formats.

Basic Query Replacement: VBA allowed lightweight filtering and calculations, reducing the need for complex database queries.

Custom Functions: Custom VBA scripts applied project-specific transformations and scoring rules.

Real Estate Ads Dataset  
The dataset used in the project was created from scratch by gathering real-time data from Immowelt, one of Germany's leading real estate platforms. The dataset comprises over 2,000,000 rows, each representing a real estate ad, and includes crucial information such as:

* ad\_id: Unique identifier for each ad
* street: The street address of the property
* city\_code: Code representing the city of the property
* price: Price of the property
* number\_of\_rooms: Number of rooms in the property
* living\_area: Size of the living area in square meters
* land\_size: Size of the land in square meters
* URL: Link to the ad's page for direct access
* city\_id: Unique identifier for the city
* city\_score: Calculated score for the city based on factors such as livability, amenities, etc.
* population: Population of the city
* geo\_spatial: Geospatial data related to the property
* center\_distance: Distance from the city center
* airport\_distance: Distance from the nearest airport

In the first stage of web scraping, the following information was gathered for each zip code:

* ZipCode
* City\_ID
* State\_ID
* URL

  
Figure 18:

The scraping process works by iterating over each zip code and fetching the maximum number of pages available for that zip code. Then, it scrapes all the real estate ads from those pages, collecting detailed information like ad price, number of rooms, land size, and more. This data is then stored in a CSV file for easy access and further analysis.

## 3. Methods

### 3.1 Algorithms

In this project, several machine learning algorithms were tested and applied to evaluate their effectiveness in predicting real estate prices. The algorithms chosen are designed to capture the complexity of real estate data, including both numerical and categorical variables, and to offer insights into the importance of different features.

Within the literature view, old researches on finding the best machine learning model for real estate valuation is not finalized since that purpose of search is relatively new. Even though there is no best suitable machine learning algorithm for that specific purpose, according to researches [2, 4] SVM is underperformed compared to RF (Random Forest) and GBM (Gradient Boosting Machine).

Linear Regression  
Linear regression is a fundamental and widely used statistical method for predictive modeling. It is simple, interpretable, and serves as a baseline model for comparison. Linear regression works well for problems where the relationship between input variables (features) and the target variable (price) is assumed to be linear. While effective in certain contexts, its simplicity may limit its ability to model more complex relationships inherent in real estate data.

Why Used: Linear regression was used as a baseline model to assess its performance against more sophisticated algorithms. It provides a good starting point for understanding how various features (such as square footage, number of rooms, etc.) impact the predicted price.

Tested Data: Linear regression was tested using a dataset containing properties with various features, including price, square footage, number of rooms, age of the property, and location data.

Random Forest  
Random Forest is an ensemble learning method that uses multiple decision trees to make predictions. It is known for its ability to handle large datasets with complex relationships and interactions between features. One of the key advantages of Random Forest is its ability to evaluate feature importance, helping to understand which factors are most influential in determining the property price. This makes it particularly useful for real estate valuation where multiple features interact with each other in non-linear ways.

Why Used: Random Forest was selected as the primary algorithm for this project due to its robustness and accuracy in predicting real estate prices. It is particularly effective when dealing with a large number of features and complex, non-linear relationships. Additionally, its ability to evaluate feature importance allowed for better understanding of the variables influencing property prices.

Tested Data: Random Forest was tested on a dataset with various property features such as price, square footage, location, and amenities. The model was trained to predict price intervals, allowing for a more detailed understanding of predicted property values (e.g., low, medium, high price intervals).

XGBoost  
XGBoost (Extreme Gradient Boosting) is an advanced machine learning algorithm based on the gradient boosting framework. It is known for its high predictive accuracy and efficiency, especially in complex problems with large datasets. XGBoost can handle both regression and classification tasks, and it is particularly well-suited for scenarios where there are intricate relationships between features that linear models cannot capture.

Why Used: XGBoost was used in this project to capture more complex relationships and interactions within the data that may not be well represented by Random Forest. It is particularly useful for fine-tuning and optimizing the model’s performance, improving prediction accuracy.

Tested Data: XGBoost was tested using the same dataset as Random Forest. It was applied to both price interval prediction and direct price prediction, offering a deeper level of insight into property valuations. The model was fine-tuned to assess its performance in terms of accuracy and generalization.

Model Focus: Random Forest Regression and Interval Prediction

Among the algorithms tested, Random Forest Regression proved to be the best fit for the case of real estate price prediction. This algorithm was particularly well-suited to handle the complexities of real estate data, such as non-linear relationships and the importance of different features.

The prediction approach used in this project was interval-based, where prices were predicted within certain intervals (e.g., low, medium, high), rather than as exact numerical values. This method offers a more robust way to handle the uncertainty and variability in real estate prices. However, both price interval prediction and direct price prediction were performed using Random Forest, allowing for a comparative analysis of both approaches.

Interval Prediction: Predicting price ranges or intervals helps account for the inherent uncertainty in real estate valuation, providing a more reliable range of potential prices rather than an exact figure.

Price Prediction: In addition to interval prediction, direct price prediction was also tested to assess how accurately the model could estimate the actual property price. This approach was more precise but also more sensitive to outliers and data inconsistencies.

By combining both interval and direct price prediction, the model provides a comprehensive view of real estate valuation, balancing the need for accuracy with the flexibility to account for market variability.

### 3.3 Data Splitting

In predictive modeling, the division of data into training, validation, and test sets is a critical step to ensure the robustness and generalizability of the model. The dataset is divided as follows:

Training Set (70%): Used to train the model and learn patterns from the data.

Validation Set (15%): Used to tune hyperparameters and avoid overfitting by evaluating the model’s performance on unseen data during the training phase.

Test Set (15%): Reserved for final evaluation, providing an unbiased assessment of the model's performance.

The splitting is performed using stratified sampling to maintain the proportional distribution of the target variable across all subsets. This approach ensures that each subset is representative of the entire dataset, which is particularly important when dealing with imbalanced data.

### 3.4 Evaluation Metrics

To assess the performance and accuracy of the predictive model, the following evaluation metrics are utilized:

Mean Absolute Error (MAE)  
MAE measures the average magnitude of errors in the predictions, without considering their direction. It is calculated as follows:

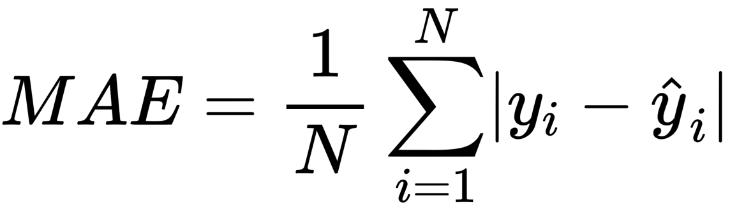
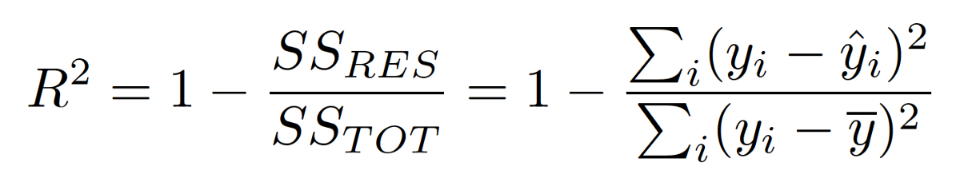


Figure 19:

Here, yi​ represents the actual values, yi (with hat) denotes the predicted values, and n is the number of observations. A lower MAE indicates better model performance as it signifies minimal deviations from the actual values.

R-squared (R²)  
The R²metric, also known as the coefficient of determination, quantifies the proportion of variance in the dependent variable that is predictable from the independent variables. It is expressed as:

  
Figure 20:

Where y (with bar)​ is the mean of the observed data. The R² value ranges from 0 to 1, with higher values indicating better model fit. A value of 1 signifies a perfect prediction, whereas a value of 0 suggests that the model fails to explain any variability in the target variable.

Mean Absolute Deviation (MAD)  
MAD quantifies the average deviation of observed values from their mean and provides a measure of data dispersion. It is particularly useful for understanding the variability within the data. MAD is calculated as:

A black background with a black square

Description automatically generated with medium confidence  
Figure 21:

Here, xi represents individual observations, and m is the mean of the observed values. While MAD does not directly evaluate model performance, it provides a reference for comparing the spread of the data to the model errors (e.g., via MAE).

By combining these metrics, a comprehensive evaluation of the model's predictive accuracy, reliability, and the underlying data's variability can be conducted. MAE and MAD together allow for a nuanced understanding of the error distribution, while R² offers insights into the explanatory power of the model.

## 4. Results

### 4.1 Algorithm Performances

The performance of the predictive models is evaluated using several statistical metrics, and visualizations are used to compare predicted and actual values. The results for each model are summarized below:

Price Prediction Model  
Performance Metrics for price prediction:

* Mean Absolute Error (MAE): €92,333
* Mean Squared Error (MSE): €18,152,598,354
* Root Mean Squared Error (RMSE): €134,731
* Mean Absolute Deviation (MAD): [Value to be inserted]
* R-squared (R²): 0.62
* Accuracy Rate (within 20% tolerance): 63.17%

These results indicate that the model achieves moderate accuracy with an R² value of 0.62, signifying that 62% of the variability in the target variable (price) is explained by the model. However, the relatively high MAE and RMSE values reflect substantial deviations between predicted and actual prices, emphasizing the need for further improvements, such as incorporating additional features or tuning hyperparameters.

Visualization  
A scatter plot comparing actual and predicted prices highlights the spread and alignment between the values. Deviations are observed particularly in higher price ranges, suggesting challenges in capturing extreme price variations.

Price Interval Prediction Model  
Performance Metrics

* Mean Absolute Deviation (MAD): €1,596
* Scaled MAD: €319

For Minimum Price Interval Prediction

* Mean Absolute Error (MAE): €738
* Root Mean Squared Error (RMSE): €1,013

For Maximum Price Prediction

* Mean Absolute Error (MAE): €734
* Root Mean Squared Error (RMSE): €1,037

This model effectively predicts price intervals with smaller deviations compared to the absolute price prediction model. The relatively lower MAE and RMSE values for minimum and maximum price predictions indicate improved accuracy for capturing price ranges.

Visualization  
Line plots depicting predicted and actual minimum and maximum price intervals demonstrate close alignment, particularly for mid-range values. However, slight discrepancies are noted at the boundaries of the data range.

#### Visual Mapping

  
Figure 22:

After giving the address into algorithm, algorithm will start.

A screenshot of a computer

Description automatically generated  
Figure 23:

The algorithm finds the correct latitude and longitude. Then it can calculate the distance to amenities. It allows to calculate the score of each bullet point.

A graph with numbers and points

Description automatically generated with medium confidence  
Figure 24:

That is the map corresponds to the latitude and longitude of the given address. The address is signed as red dot, and the bullet points as purple dots. It depict the real location without satellite view.  
  
Note that the visuals are a debug mechanism, and cannot use by User. It is counted as internal tool for debugging and gaining further insights from the address.

### 4.2 Algorithm Comparison

A comparative analysis of the two predictive models reveals insights into their respective strengths and limitations:

Price Prediction Model  
The price prediction model performs moderately well in terms of overall accuracy, as evidenced by its R² value (0.62). However, the model struggles with high variability in the data, leading to higher error metrics such as MAE (€92,333) and RMSE (€134,731). These results suggest that the model's ability to capture non-linear relationships and handle extreme outliers is limited.

Price Interval Prediction Model  
The interval prediction model outperforms the absolute price prediction model in terms of precision and consistency. Lower MAE values (€738 and €734 for minimum and maximum prices, respectively) and scaled MAD (€319) demonstrate that this model is better suited for tasks requiring precise range estimations. Furthermore, the reduced RMSE values highlight its robustness in minimizing large errors.

Key Observations  
The Random Forest algorithm used in the interval prediction model likely outperformed linear approaches due to its capacity to model non-linear relationships and interactions between variables effectively.

On the other hand, the price prediction model may benefit from incorporating ensemble methods or feature engineering to improve performance further.

In summary, while the price interval prediction model delivers superior results in terms of accuracy and error minimization, the price prediction model provides valuable insights into overall price trends and patterns.

## 5. Discussion and Evaluation

### 5.1 Specific Challenges Faced

Several challenges were encountered throughout the project, reflecting the complexities of real-world data acquisition, processing, and analysis:

Fake Advertisements: Many listings on the real estate platforms were promotional rather than genuine ads, created to market realtors or their services. These "fake ads" were difficult to identify and filter out due to the lack of an advanced outlier detection mechanism, leading to potential noise in the dataset.

Inaccurate or Outdated Geospatial Data: Geospatial data quality was another significant hurdle. While the project required precise and up-to-date location data, budget constraints prevented the use of premium services like Google Maps API. Instead, a free API was utilized, which occasionally provided inaccurate or incomplete data. Additional preprocessing and validation were necessary to mitigate this limitation.

Complexity of Web Scraping: Conducting web scraping was a labor-intensive, multi-step process:

Zip Code Acquisition: The first step involved scraping all the available German ZIP codes to structure the geographical queries.

URL Generation for Ads: Using the ZIP codes, queries were sent to the real estate website to retrieve URLs associated with listings in each region.

Batch Processing of Data: Over 200 batches were created for different ZIP codes, eventually compiling over 2 million advertisements. Each batch underwent a scoring process to calculate features like the city center score and city score.

Feature Engineering and Integration: Scoring mechanisms were based on statistical calculations of average price per square meter for each ZIP code, integrated with a database for city scores. The final dataset, after cleansing and filtering, comprised 40,000 ads.

This end-to-end process required significant effort to handle data in batches, merge functionalities, and ensure robust feature engineering for accurate modeling.

### 5.2 Lessons Learned

The project provided valuable insights into the importance of location and other factors in real estate valuation:

Significance of City and Location: The city in which a property is located is the most critical determinant of its value. Even small villages within high-demand cities like Munich exhibited prices significantly higher than equivalent properties elsewhere.

Living Space as a Key Factor: The second most influential factor was the size of the living space, reinforcing its importance in property valuation.

These lessons highlighted the complex interplay of geographical and spatial data in predictive modeling for real estate.

### 5.3 Recommendations for Improvement

To enhance the project's scope and effectiveness in future iterations, the following improvements are suggested:

Broader Valuation Criteria: While location is a dominant factor, housing valuation should also account for socio-economic and psychological dynamics, such as the nation's economic health, public morale, and market trends. These factors play a significant role in influencing housing demand and pricing.

Incorporating Construction Year: The age and construction year of properties should be factored into the valuation model, as they directly affect both market value and buyer preferences.

Advanced Outlier Detection: Developing a robust mechanism for identifying and removing fake or misleading advertisements would significantly enhance data reliability and model accuracy.

Real-Time Data Sources: Integrating real-time data feeds could provide more dynamic insights into market trends and improve the relevance of predictions.

Enhanced APIs and Data Sources: Exploring affordable alternatives to premium APIs or optimizing the use of free APIs could improve geospatial data quality without exceeding budget constraints.

### 6. Conclusion

Algorithm Performance: The predictive algorithms achieved competitive accuracy rates, demonstrating their capability to estimate property values effectively. The integration of geospatial data further improved the model's reliability, providing a more holistic understanding of property value factors.

Impact of Geospatial Data: Geospatial features like proximity to city centers and average prices within ZIP codes added substantial value to the valuation process. This enriched the predictive power of the model by incorporating location-based nuances.

#### Real-World Applications

Investor and Buyer Assistance:

This system empowers investors and buyers to make well-informed decisions, utilizing insights derived from data rather than relying solely on intuition or anecdotal information.

Transparency in Real Estate Transactions:

By providing objective, data-backed valuations, the project enhances trust between stakeholders in the real estate market.

Final Remark  
The methodology and tools employed in this project demonstrate the potential of data science in transforming traditional real estate practices, making them more transparent, reliable, and accessible. Further advancements, such as real-time data integration and improved location-based features, could push these benefits even further.

## Abbreviations and Glossary

AVMs: Automated Valuation Models

GIS: Geographic Information System

GBM: Gradient Boosting Machines

MAD: Mean Absolute Deviation

MAE: Mean Absolute Error

VBA: Visual Basic for Application

XGBoost: eXtreme Gradient Boosting

## Resources

[1] Gold, C. M. (2006). What is GIS and What is Not?. *Transactions in GIS*, *10*(4), 505-519.

[2] 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>. Retrieved date: 06/01/2025

[3] Kang, Y., Zhang, F., Peng, W., Gao, S., Rao, J., Duarte, F., & Ratti, C. (2021). Understanding house price appreciation using multi-source big geo-data and machine learning. Land use policy, 111, 104919. Available at: <https://shorturl.at/8UIuj>. Retrieved date: 02/12/2024

[4] Mete, M. O., & Yomralioglu, T. (2022). Mass valuation of Real Estate Using GIS-based nominal valuation and machine learning methods. *European Real Estate Society, ERES*, 1-7.

[5] Plakandaras, V., Gupta, R., Gogas, P., & Papadimitriou, T. (2015). Forecasting the US real house price index. *Economic Modelling*, *45*, 259-267. Available at: <https://arxiv.org/pdf/1707.04868>. Retrieved at: 07/01/2025

[6] Root, T., H. Strader, T., J. Huang, Y., J. (31/07/2023). A Review of Machine Learning Approaches for Real Estate Valuation. Journal of the Midwest Association for Information Systems. Page:16. Available at: <https://jmwais.org/wp-content/uploads/sites/8/2023/07/V2023.I2.A2.pdf> Retrieved date: 06/01/2025

[7] Tereshchenko, S. (27/01/2024). Common real estate problems and their solutions. Greenm.io. Available at: <https://greenm.io/common-real-estate-problems-and-their-solutions> Retrieved date: 06/01/2025

[8] Vaddi, S. S., Yousif, A., Baraheem, S., Shen, J., & Nguyen, T. V. (2022). House Price Prediction via Visual Cues and Estate Attributes. In International Symposium on Visual Computing (pp. 91-103). Cham Springer Nature Switzerland. Available at: <https://par.nsf.gov/servlets/purl/10428263>. Retrieved date: 07/01/2025

[9] Yarotska, Y. (14/11/2024). AI Property Valuation Software: A New Disruption of Real Estate Appraisals? Ascendix.com. Available at: <https://ascendixtech.com/ai-property-valuation-tools-appraisal> Retrieved date: 06/01/2025