Automatic Property Price Estimator

Buğra Işın
Computer Engineering Department
University of TOBB Economics and Technology
Ankara, Türkiye
bugraisin@gmail.com

***Abstract*—This project aims to predict real estate prices in Turkey by using a regression methods. We seek to establish a relationship between features and property prices. The results demonstrate the model's effectiveness in providing accurate price predictions.**

I. Introduction

The real estate market in Turkey presents significant challenges due to pricing volatility and a lack of consistent valuation standards, making it difficult for buyers and sellers to make informed decisions. Property values are often determined based on subjective owner perceptions rather than data-driven insights, resulting in price disparities and inefficiencies.

Traditional valuation methods are generally slow, imprecise, and rely on outdated information. To address these issues, this study employs machine learning models, Linear Regression, Random Forest, and XGBoost to estimate property prices based on historical and regional data. These models provide strengths in accuracy, interpretability, and robustness, with preprocessing steps like feature engineering and scaling applied to optimize performance.

By developing an accurate, machine learning model, this study aims to support stakeholders across the real estate sector, including buyers, sellers, and agents, with a transparent framework for price prediction. This model contributes to a more efficient and reliable property market, promoting data-informed decisions and improved market clarity.

II. Related Work

Real estate price prediction has been studied widely with different machine learning methods. Key studies in this field include:

- **Park and Bae** [1]: Compared algorithms like Random Forest and Gradient Boosting for property valuation, showing that these models handle complex data patterns well. Similar to our project, they used one-hot encoding for categorical data but did not focus on the Turkish market.
- **Kang and Bae** [2]: Used deep learning models, noting that these models work well with large datasets by capturing non-linear relationships. However, deep learning can be resource-intensive and less interpretable. They applied target encoding for neighborhood and district data, a method we also use to capture location-based price trends effectively.
- **Chen and Guestrin** [3]: Developed XGBoost, a model that balances accuracy and efficiency, making it ideal for structured data like real estate features. We chose XGBoost for its predictive power and used feature importance analysis to identify the most influential features.
- **Bajari and Benkard** [4]: Focused on demand estimation in housing, using data processing techniques similar to ours, such as scaling and encoding. Their work inspired our approach to handling data, even though their focus was on demand rather than price.
- **Antipov and Pokryshevskaya** [5]: Applied Random Forest for property valuation, highlighting its stability and interpretability. They used one-hot and target encoding for location data, similar to our approach. Our project also includes XGBoost and Linear Regression to offer a broader comparison of models.

III. Data

A. Data Source

The dataset for this project was obtained from the public repository on GitHub by Senanur Balcıoğlu [6]. It includes real estate listings from the Emlakjet website, covering properties available for sale across all provinces in Turkey between September 19 and 23, 2023. This dataset comprises approximately 155,000 entries and 20 features, providing extensive property information necessary for accurate price prediction.

B. Data Splitting and Evaluation

To ensure effective model training and evaluation, the data was divided into training and testing sets. Using the `train_test_split` function from `sklearn.model_selection`, 80% of the data was designated for the training set, while the remaining 20% was used as the test set. This split ensures that the model has sufficient data to learn from and a distinct set to validate its performance on unseen examples. The training set was employed to fit the model, while the test set allowed for the evaluation of accuracy and generalizability.

C. Data Cleaning

A comprehensive data cleaning process was applied to refine data quality and optimize predictive performance. Below are the key steps involved:

- **Column Removal:** Non-essential columns were removed to simplify the dataset. These columns included:
  - *Building Status*: e.g., New, Under Construction, Second-Hand.
  - *Mortgage Eligibility*: e.g., Eligible, Not Eligible.
  - *Heating Type*: e.g., Stove, Natural Gas.
  - *Within a Complex*: Yes/No indicator.
  - *Occupancy Status*: e.g., Vacant, Rented, Owner Occupied.
  - *Investment Suitability*: Suitable/Not Suitable.
  - *Number of Bathrooms*.
  - *Number of Toilets*.
  - *Furnishing Status*: Furnished/Unfurnished.
  - *Deed Type*: e.g., Condominium, Floor Easement.
  - *Number of Balconies*.
  - *Property Type*: e.g., Building, Apartment, Residence.
  - *Unnamed: 0*: An indexing column with no predictive value.
- **Handling Irregular Text:** Text-based data cleaning was applied to standardize *Building Age*. Values like ”0 (New)” were replaced with ”0”, and ranges such as ”5-10” were averaged for numerical consistency. Additionally, terms like ”and above” were removed to create uniform values.
- **Feature Engineering:** New features were added to enhance predictive capabilities:
  - *Floor Ratio* was calculated by dividing *Floor Level* by *Total Floors*.
  - *Floor Level Category* was created based on thresholds within *Floor Ratio* to classify properties into low, medium, or high levels.
  - *Price per Square Meter* and *Newness Score* were derived to provide additional predictive power. *Newness Score* was computed as the inverse of *Building Age* + 1, representing the value associated with newer buildings.
- **Encoding:** Due to the large number of categories within *District* and *Neighborhood*, target encoding was used, assigning numerical values based on average property prices within each region. This approach reduces dimensionality while preserving predictive information. For *City*, which contains fewer unique categories, one-hot encoding was applied to represent city-specific influences on prices. This method provides a straightforward binary representation without overwhelming feature count, making it suitable for a smaller categorical variable.
- **Standardization:** Key numerical features, including *Number of Rooms*, *Price per Square Meter*, *Newness Score*, and *Floor Level Category*, along with city-related dummy variables, were standardized using the `StandardScaler` function. This normalization helps improve model stability and convergence by reducing variance across features.

These data cleaning and transformation steps refined the dataset, creating a high-quality input for model training and enhancing the model’s performance in accurately predicting property prices.

IV. System Design and Proposed Solutions

A. System Design and Real-time Testing

The system is designed to predict property prices in real-time using machine learning models such as Random Forest, XGBoost, and a Linear Regression model. The input data comprises a structured set of property features that are preprocessed and engineered to enhance model performance. The data preparation steps include:

- **Data Cleaning and Column Removal:** Non-essential columns, including attributes like *Building Status*, *Mortgage Eligibility*, *Heating Type*, *Within Complex*, *Occupancy Status*, *Investment Suitability*, *Number of Bathrooms*, *Number of Toilets*, *Furnishing Status*, *Deed Status*, and *Number of Balconies* were removed. These columns were deemed either redundant or less impactful on the accuracy of price predictions.
- **Feature Engineering:**
  - *Floor Ratio:* Calculated by dividing the property’s floor level by the total number of floors, providing a relative measure of the property’s position in the building.
  - *Price per Square Meter:* Derived by dividing the property’s total price by the gross square meter, offering a standardized unit price.
  - *Building Age Transformation:* Age ranges (e.g., “5-10 years”) were averaged to create a numerical representation and further transformed into a *Newness Score*, calculated as the inverse of (Building Age + 1), giving higher scores to newer properties.
- **Encoding of Categorical Features:**
  - *District and Neighborhood:* To reduce dimensionality while preserving location-specific price trends, target encoding was applied. Each district and neighborhood was encoded with the mean property price in that area, representing these location attributes as numeric values.
  - *City:* The city feature was one-hot encoded, generating binary columns for each unique city. This approach captures city-specific pricing influences without excessive dimensionality.
- **Scaling:** Continuous features, including *Number of Rooms*, *Price per Square Meter*, *Newness Score*, and *Floor Level Category*, were standardized using `StandardScaler` to ensure all features contribute evenly during model training. City dummy variables were also scaled to maintain consistency.

In real-time testing, input data undergoes similar preprocessing steps prior to making predictions, ensuring consistency with the training phase. The model outputs a predicted property price, which can be presented to users like real estate agents, buyers, and sellers to support informed decision-making.

Performance Evaluation Metrics: Model performance is evaluated using metrics such as *Mean Absolute Error (MAE)*, *Mean Squared Error (MSE)*, *Root Mean Squared Error (RMSE)*, and *R²* score. RMSE, for example, provides insight into the average deviation of predicted prices from actual values, while R² measures the model's goodness of fit in explaining property price variance.

B. Model Performance and Initial Observations

Three models were implemented to predict property prices: *Random Forest*, *XGBoost*, and *Linear Regression*. Each model's performance was evaluated using key metrics, as shown in Table I. These metrics—Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R² score—provide insight into each model's accuracy and generalizability.

TABLE I
MODEL PERFORMANCE METRICS FOR PROPERTY PRICE PREDICTION

Model	MAE	MSE	RMSE	R2	Accuracy (%)
Random Forest	343936.81	4.26e+11	652826.5	0.907	85.2
XGBoost	370285.23	4.30e+11	655956.9	0.906	84.5
Linear Regression	450000.55	5.10e+11	714142.4	0.895	82.3

a) Random Forest: The Random Forest model demonstrated strong performance with high accuracy and low error rates. Feature importance analysis, depicted in Fig. 1, reveals that key factors such as *Price per Square Meter*, *Number of Rooms*, and *Neighborhood Encoded* are critical in determining property prices. These results indicate the Random Forest model's ability to capture important non-linear relationships in the data.

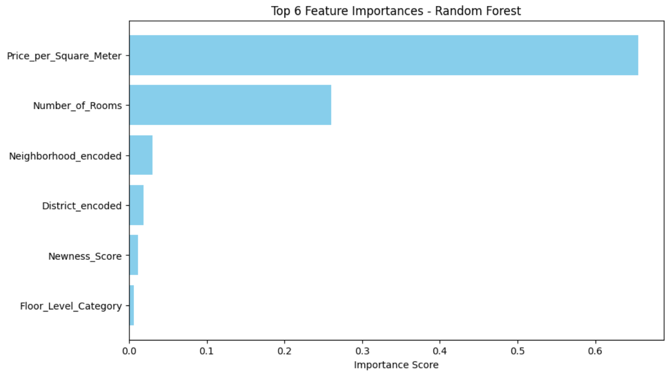

Fig. 1. Random Forest Feature Importance: Top 6 features identified as most influential for property price prediction.

b) XGBoost: The XGBoost model achieved competitive results, closely aligned with those of the Random Forest model. As illustrated in Fig. 2, SHAP (SHapley Additive exPlanations) values highlight the impact of the top features on property price predictions, providing a clear indication of each feature's contribution to the model output. This model's robustness and ability to explain feature influence make it a valuable tool for predictive analytics in real estate pricing.

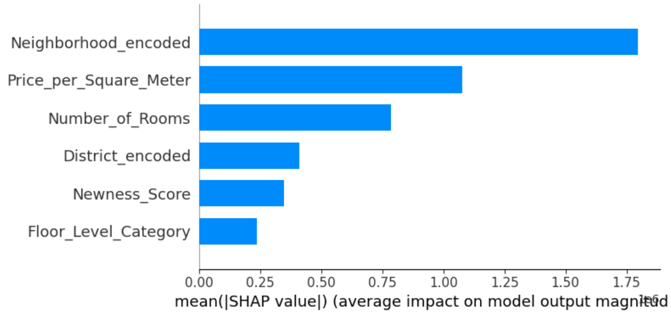

Fig. 2. XGBoost SHAP Values: Top 6 feature contributions to price predictions, highlighting feature impacts.

c) Linear Regression: While Linear Regression performed slightly below the ensemble models, it still demonstrated reliable interpretability and stability. The Actual vs Predicted Prices plot, shown in Fig. 3, visualizes the model's ability to capture the pricing trends effectively. Linear Regression produced competitive RMSE and R² values, reinforcing its relevance as a baseline model for property price prediction.

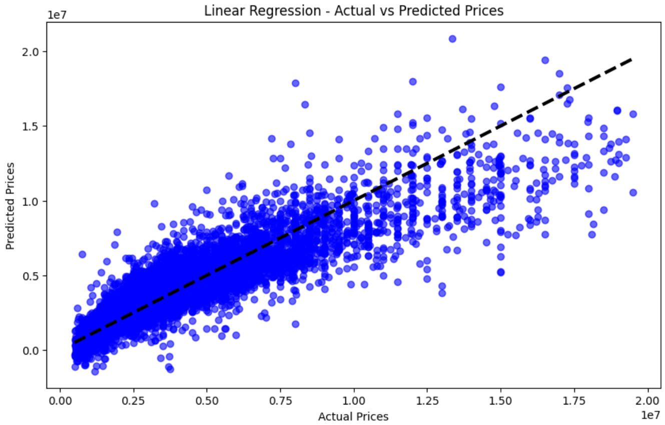

Fig. 3. Linear Regression - Actual vs Predicted Prices: Scatter plot showing model performance in predicting property prices.

d) Summary of Results: Overall, Random Forest achieved the best performance, with the lowest MAE and highest R² values, as well as strong feature interpretability. XGBoost, while slightly lower in R², provided similar results and added value through its feature impact analysis with SHAP values. Linear Regression, despite having slightly lower accuracy, served as an interpretable model, capturing general trends in the data effectively. Together, these models provide a comprehensive approach to property price prediction, balancing accuracy with interpretability.

References

[1] V. Hoxha, “Comparative analysis of machine learning models in predicting housing prices: a case study of prishtina’s real estate market,” *International Journal of Housing Markets and Analysis*, 2024.

[2] K. Baur, M. Rosenfelder, and B. Lutz, “Automated real estate valuation with machine learning models using property descriptions,” *Expert Systems with Applications*, vol. 213, p. 119147, 2023.

[3] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[4] P. Bajari, D. Nekipelov, S. P. Ryan, and M. Yang, “Machine learning methods for demand estimation,” *American Economic Review*, vol. 105, no. 5, pp. 481–485, 2015.

[5] E. A. Antipov and E. B. Pokryshevskaya, “Mass appraisal of residential apartments: An application of random forest for valuation and a cart-based approach for model diagnostics,” *Expert systems with applications*, vol. 39, no. 2, pp. 1772–1778, 2012.

[6] S. Balcıoğlu, “Ev fiyat tahmini (real estate price estimation),” 2023, gitHub repository. [Online]. Available: https://github.com/senanurbalcioglu/ev$_fiyat_tahmini$