Christopher Kevin Siswanto

26002204494

31997: Data Science

Data Science Final Report

Machine Learning for Property Price Prediction in Greater London

***Abstract*—This study explores the application of machine learning models to properly predict property prices in Greater London using the UK Price Paid dataset. The dataset presented significant challenges due to the absence of key property specific features such as the number of bedrooms, floor count, or square footage. To address these limitations, feature engineering techniques were employed to extract meaningful patterns from available data, including transaction dates, geographical information, and property types. Four models, Linear Regression, Random Forest, Gradient Boosting, and CatBoost were trained and evaluated to assess their predictive performance. Among these models, CatBoost emerged as the most effective model, achieving the highest accuracy and the lowest prediction error. The results highlight the potential of advanced machine learning algorithms to handle complex datasets and provide actionable insights into property price prediction. Despite these advancements, the study identifies the possible areas for improvement, particularly in enhancing predictions for high value properties, by incorporating additional external data and exploring deep learning approaches. This work underscores the importance of combining robust preprocessing, model selection, and evaluation to achieve reliable predictions in the real estate analytics world.**

Keywords—Data Science, Machine Learning, Price Prediction, Real Estate Analytics

I. Introduction

The realistic price prediction of properties is a crucial aspect for buyers, sellers, investors, and data analysts in the real estate world. With an objective to predict house prices, there exist a lot of factors that contribute to deciding a price, ranging from location to property type to market trends to economic conditions, making the job tough. Nowadays, with ever-growing, transaction level data and state of the art machine learning algorithms, it becomes an important task to conduct efficient and much more accurate predictions through a data-driven approach.

The Greater London housing market exemplifies this complexity, as it represents one of the most dynamic and competitive real estate landscapes in the United Kingdom. Characterized by high property demand, substantial price variation across districts, and consistent market growth over the years, understanding the key drivers of property prices in this region is essential for effective decision-making.

This project deals with the prediction of property prices in Greater London using machine learning models. The data used is the UK Price Paid Data [1], a very detailed dataset on the transactions of properties, including sale price, property type, transaction date, and location. It does not mention any specifics about the property which is different and gives data analysts a hard challenge. Given the size and complexity of the dataset, some preprocessing steps were necessary to handle categorical features, log transformation of the target variable, and scaling of numerical inputs in order to prepare the data for analysis.

The various predictive models developed and evaluated in the paper include a baseline model of Linear Regression, tree-based models such as Random Forest, Gradient Boosting, and an advanced tree model CatBoost. These were selected for their capabilities of non-linearity and capturing complex patterns in the data. Performance evaluation was done based on some key metrics, such as MAE, MSE, and R^2, with MAE in the original price scale being the main metric for comparison. Finally a sample test was done to show how the model performed on actual data.

This project will have two main objectives, to identify the best model that describes house price prediction and show the important factors affecting house prices in Greater London. The results will be useful to the stakeholders in the real estate sector and will also provide a data analysis task with meaningful insights.

II. Proposed Method

The proposed system for predicting property prices in Greater London uses a dataset that closely relates to the problem and trains machine learning models that will be compared using evaluation metrics. The methodology involves five stages as seen on Figure 1: data acquisition, data preprocessing, feature engineering, model development, and model evaluation.

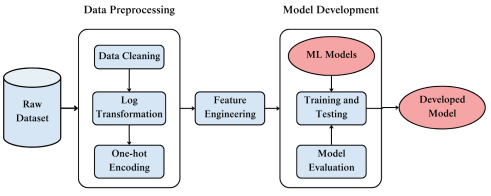

Fig. 1. Predictive modelling Workflow

2.1 Data Acquisition

The dataset used for this project is the UK Price Paid Data, which is publicly available and made by the UK's HM Land Registry and provides transaction-level information on property sales in England and Wales since 1995 up until

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

2017. This dataset includes details such as the sale price, transaction date, property type (e.g., detached, semi-detached, terraced, flats), whether the property is newly built or old, and tenure (freehold or leasehold). Additionally, geographical data covering the town/city, district, and county is provided, allowing for location-based analysis. Since it is a dataset made by the government, specific locations and home details such as floor type, bathroom, bedroom, and other details are not shared to the public. This makes a data analysis task further needed to tackle this unconventional dataset.

To make the analysis more specific and relevant, the dataset was filtered to include only transactions in Greater London. This filtering ensures the models capture the unique characteristics of the region's real estate market, offering insights specific to this dynamic area.

2.2 Data Preprocessing

Before the machine learning models are trained, preparation for the dataset is required to ensure each model understands the data inserted. Several preprocessing steps were implemented. Since the dataset is made by the government themselves, missing values in critical columns, such as price and transaction date, were not needed for removal since the data integrity is correctly formatted beforehand. Transactions with extreme property prices in the top and bottom 1% were identified as outliers and excluded to reduce noise and prevent skewed results. This is caused by buyers that aim for high value properties, however in this dataset, there is no specific mention of either high or low value properties directly, which propose a challenge for the regression task. The target variable chosen will be the property price and other features will be used for learning the relationships.

A log transformation was applied to address the skewness in the property price, using the formula:

$$Log\ Price = log(P + 1)$$

Where P represents the original property price (measured in GBP) for each transaction in the dataset. The constant 1 is added to P to handle cases where the property price might be zero, as the logarithm of zero is undefined.

Finally, numerical features were standardized using StandardScaler to ensure compatibility with models that require scaled inputs.

2.3 Feature Engineering

Feature engineering was performed to enhance the predictive power of the dataset. Temporal features were derived through the extraction of Transaction Year, Month, and Quarter from the transaction date column. These variables capture time related trends and seasonality in the property market, therefore providing additional context to the models.

Categorical variables, including Property Type, Town/City, and District, were one-hot encoded to create binary columns for each unique category. One-hot encoding is a process of converting categorical data into a numerical format that machine learning models can understand. Each category within a feature is represented as a separate binary column. If a particular data point belongs to a category, the corresponding binary column is assigned a value of 1, and all other columns are assigned 0. Ensuring that machine learning algorithms treat these categories independently, this method is done for this data since unlike factorization method where assigning arbitrary numerical values to categories could lead to unintended ordinal relationship.

2.4 Model Development

The project involved the training and evaluation of four machine learning models. Linear Regression was implemented as a baseline model to establish a performance benchmark. Random Forest, an ensemble model that builds multiple decision trees and averages their predictions [4], was chosen for its ability to capture non-linear relationships and feature interactions. Gradient Boosting [3], a boosting algorithm that iteratively minimizes residual errors, was used to improve prediction accuracy. Finally, CatBoost [2], a gradient boosting algorithm optimized for categorical data and high-cardinality features, was selected for its advanced capabilities in handling complex datasets. Detailed network architectures of each model are continued in the experiments section.

The data was split into training and testing subsets, with 80% of the data used for training and 20% reserved for testing. Hyperparameter tuning was applied to optimize the performance of advanced models such as Random Forest, Gradient Boosting, and CatBoost. This process ensured that the models achieved the highest possible accuracy on unseen data.

2.5 Model Evaluation

The performance of the models was assessed using three key metrics:

- Mean Absolute Error (MAE):

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |yi - \widehat{yi}|$$

The MAE measures the average absolute difference between the actual and predicted value. It is calculated by summing the absolute differences $yi - \widehat{yi}$ across all data points and dividing by the total number of observations (n). MAE provides a straightforward and interpretable metric in the same unit as the target variable. It represents the average error in predicting property prices in GBP when not scaled.

- Mean Squared Error (MSE):

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (yi - \widehat{yi})^2$$

The MSE measures the average squared difference between the actual and predicted values. It penalizes larger errors more heavily due to the squaring of residuals $(yi - \widehat{yi})^2$ .While MSE emphasizes large prediction errors, it is less interpretable because it is expressed in squared units of the target variable. However, it is useful for understanding model sensitivity to large deviations.

- R^2 Score:

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

The R^2 score evaluates the proportion of variance in the target variable (yi) explained by the model. It compares the model's residual errors to the variance of the target, the value ranges from 0 meaning the model performs no better than predicting the mean. 1 represents the model that perfectly fits the target variable. If the value is lower than 0, the model is worse than predicting the mean.

Residual analysis was conducted to evaluate prediction errors across different price ranges and identify any systematic biases. These evaluations provided a comprehensive understanding of the models' performance and highlighted the strengths and weaknesses of each.

III. EXPERIMENTS

The experiments were designed to evaluate the performance of baseline and advanced machine learning models, consisting of Linear Regression, Gradient Boosting, Random Forest, and CatBoost respectively to predict the housing prices. This section describes the experimental setup, the conducted experiments, and the training and evaluation process.

3.1 Experimental Setup

Experiments were conducted on a NVIDIA GeForce RTX 4060, 8GB VRAM GPU-enabled system to accelerate training for computationally intensive models that allows training and support GPU usages. Other models that do not support it will be trained on a normal CPU. Key software components included Python 3.10.9 to ensure capability along with libraries such as Scikit-learn, CatBoost, PyTorch, Pandas, and Matplotlib.

The final dataset consists of approximately 2.9 million rows and 132 features after feature engineering, was split into 80% training data and 20% testing data. This split was used to evaluate the models on unseen data and ensure generalizability.

3.2 Experiments Conducted

The experiments were carried out in stages, beginning with data visualization to explore the dataset, followed by training the models and evaluation. The training processes were monitored for convergence and stability to ensure the models were trained properly.

To gain insights into the dataset, visualizations were used to analyze trends and distributions. Figure 2 shows the distribution of property prices after applying a log transformation. This transformation effectively reduced the skewness, compressing extreme values and stabilizing variance. The average property prices over time, revealing trends and seasonal variations in the Greater London property market. These visualizations provided valuable context for model training and evaluation.

Fig. 2. Distribution of Property Prices in Greater London

The training process for each model was designed to ensure stability and convergence. Linear Regression was implemented as a baseline model and trained using stochastic gradient descent (SGD) for 100 epochs. The training loss, measured using Mean Squared Error (MSE), consistently decreased, as shown in the iteration plots, demonstrating proper convergence.

Random Forest and Gradient Boosting required hyperparameter tuning to optimize performance. The key parameters tuned included the number of trees, maximum tree depth, and learning rates. Random Forest, trained with 100 trees and a maximum depth of 10, completed its training in approximately 27 minutes using CPU. Gradient Boosting, with a learning rate of 0.1 and a maximum tree depth of 3, achieved convergence in about 26 minutes utilizing GPU.

CatBoost utilized GPU acceleration for efficient training and was optimized with 1000 iterations, a learning rate of 0.1, and a tree depth of 6. The ordered boosting algorithm demonstrated stable convergence, as reflected in the reduction of the loss function over iterations.

3.3 Model Architectures and Explanations

Linear Regression was implemented as a single-layer neural network with no hidden layers. The model takes 132 features as input and uses a linear transformation to predict the log-transformed property price. The architecture is defined as:

$$\hat{y} = XW + b$$

Where X represents the input features, W is the weight matrix, and b is the bias term. This simple model served as a baseline to compare against more complex algorithms.

The Random Forest model is an ensemble of decision trees, where each tree is trained on a random subset of the data. Predictions from individual trees are averaged to produce the final output. This approach captures non-linear relationships and interactions between features effectively. The prediction for a given input x is shown as:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} h_t(x)$$

Where T represents the number of trees in the forest, and ht(x) is the prediction of the t-th tree for input x.

Gradient Boosting builds decision trees sequentially, where each tree attempts to correct the residual errors of the previous trees, this ability allows it to minimize errors effectively, producing accurate predictions. The model starts with an initial prediction and iteratively updates it using the predictions from weak learners:

$$F_m(x) = F_{m-1}(x) + \gamma \cdot h_m(x)$$

Where $F_m(x)$ is the current ensemble model prediction, $F_{m-1}(x)$ is the prediction from the previous iteration, γ is the learning rate and $h_m(x)$ is the decision tree trained on the residuals.

CatBoost, a gradient boosting algorithm optimized for categorical data, eliminates the need for explicit one-hot encoding. It uses ordered boosting to avoid data leakage, ensuring unbiased predictions during training. The model is particularly adept at handling high-cardinality categorical features such as Town/City and District. Its GPU acceleration further enhances training efficiency, making it highly suitable for large datasets.

IV. Results And Discussions

This section evaluates the performance of the models based on the experiments conducted, providing insights into the findings. The results are discussed in the context of the dataset and the prediction task, highlighting model strengths and limitations.

The evaluation metrics used to assess model performance include Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2. Table 1 summarizes the results of the four models on the testing dataset.

TABLE 1: Model Performance Metrics

Model	MAE	MSE	R^2
Linear Regression	1.639882	2.919212	NaN
Random Forest	0.290326	0.188120	0.717633
Gradient Boosting	0.329182	0.223026	0.665240
CatBoost	0.290023	0.184153	0.723587

Based on the results of the performance metrics, Linear Regression performed the worst among all models. It exhibited the highest MAE and MSE, with no meaningful *R*^2 value due to its inability to capture non-linear relationships inherent in the dataset. The other baseline models Random Forest and Gradient Boosting performed significantly better. Random Forest demonstrated slightly superior results compared to Gradient Boosting, achieving a lower MAE and a higher *R*^2. Among all models, CatBoost achieved the best performance with the lowest MAE (0.2900), lowest MSE (0.1842), and the highest *R*^2 (0.7236). This result highlights the model's ability to handle categorical features and high-dimensional data effectively.

An analysis of feature importance was conducted to determine the most important features driving property price predictions. Among the top features identified by the CatBoost model, as seen on Figure 3, geographical variables such as District and Town/City played a pivotal role. These features captured the regional variations in property prices, providing significant predictive value. Time variables, including Transaction Year and Quarter, also emerged as critical features, reflecting time-based trends and seasonality in the real estate market. Property characteristics, such as Property Type and tenure (Duration), contribute meaningfully to the predictions, highlighting their relevance in understanding price differences among property categories.

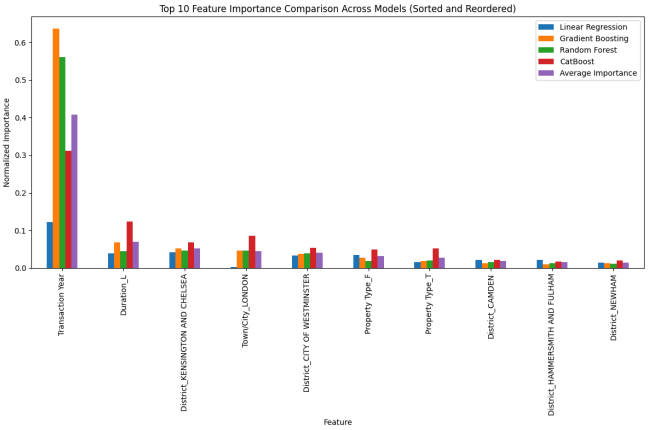

Fig. 3. Top 10 Feature Importance Comparison of all Models

Residual analysis was conducted to evaluate the errors in predictions made by the models. For Linear Regression, the residuals displayed systematic patterns, indicating that the model failed to account for complex relationships in the data. This limitation is due to its nature, which struggles with non linear data.

For the other baseline models, the residuals were more evenly distributed across the range of property prices. Random Forest displayed smaller deviations, especially in mid-range predictions, due to its robust averaging mechanism across multiple decision trees. Gradient Boosting, showed slightly higher residuals for high value properties, due to its iterative approach that can amplify errors in some cases.

CatBoost exhibited the most balanced residual distribution, with residuals concentrated around zero shown in Figure 4. This indicates that the model was able to minimize prediction errors effectively across all price ranges. The performance of CatBoost can be attributed to its ordered boosting mechanism, which enhances prediction accuracy by avoiding data leakage and ensuring unbiased predictions during training.

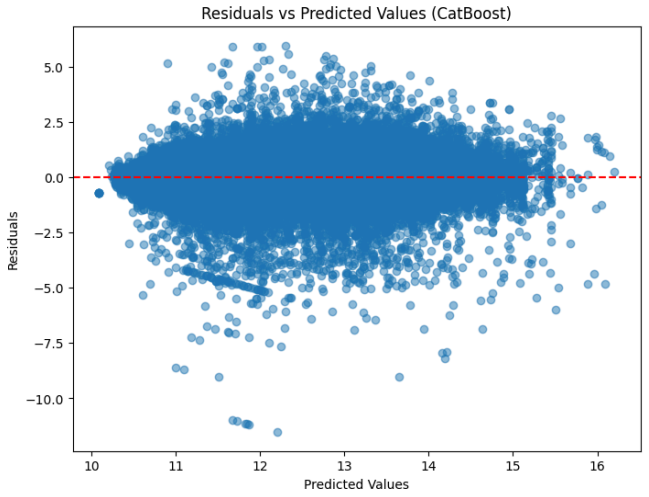

Fig. 4. Residual Plot for Advanced Model CatBoost Regression

The results emphasize the importance of selecting models that can effectively capture the characteristics of the

dataset. CatBoost's ability to natively handle categorical data without requiring one-hot encoding gave it a distinct advantage over the other models. By preserving data structure and avoiding dimensionality issues, it was able to achieve the best overall performance.

Despite CatBoost's superior performance, the study highlights areas for potential improvement. The high MAE values, especially for high value properties, indicate that additional external features, such as socioeconomic or infrastructural data, might still be needed to further enhance prediction accuracy. Future studies could also explore hybrid models or deep learning architectures for more nuanced predictions.

V. Conclusions

This study demonstrated the application of machine learning models to predict property prices in Greater London using the UK Price Paid dataset. The dataset presented unique challenges, as it lacked specific features such as the number of bedrooms or floors, which are typically strong predictors of property prices. Despite these limitations, the models were able to extract meaningful patterns from available features such as location, property type, and transaction date. Among the four models trained and evaluated, CatBoost emerged as the most effective, outperforming others in terms of accuracy and interpretability, due to its ability to handle categorical data natively. While the results highlight the potential of machine learning in real estate price prediction, the relatively high MAE values suggest there is room for improvement, especially for high value properties. Incorporating additional features or combining external data sources in future studies could further enhance the accuracy of the models, enabling more reliable predictions in the real estate market.

References

[1] UK Housing Prices Paid Dataset https://www.kaggle.com/datasets/hm-land-registry/uk-housing-prices-paid

[2] Dorogush, A. V., Ershov, V., & Gulin, A. (2018). CatBoost: gradient boosting with categorical features support. arXiv preprint arXiv:1810.11363.

[3] Jerome H. Friedman. "Greedy function approximation: A gradient boosting machine.." Ann. Statist. 29 (5) 1189-1232, October 2001. https://doi.org/10.1214/aos/1013203451

[4] Breiman, L. (2001). Random forests. Machine learning, 45, 5-32. https://doi.org/10.1023/A:1010933404324