Lost in the modeling stage: a comparative analysis of machine learning models for real estate data

Ian Lenaers[a] (0000-0001-9188-3870) and Lieven De Moor[a] (0000-0002-1290-2971)*

[a] Vrije Universiteit Brussel, Faculty of Social Sciences and Solvay Business School, Pleinlaan 2 B-1050 Brussels, Belgium

** corresponding author: Ian.Dave.J.Lenaers@vub.be*

Abstract

Lately, machine learning and artificial intelligence have dominated automated property valuation models. However, an extensive comparison of real estate price prediction models is rarely conducted. Therefore, this research aims to conduct an extensive comparison of 28 rent prediction models, trained on a cleaned dataset of 79,735 Belgian residential rental properties from 2022. The evaluation incorporates both traditional metrics and alternative metrics to assess predictive performance for the train set, the test set, and across the different deciles of the test set. The results indicate and confirm that tree-based ensemble models, outperform other models in predictive performance, suggesting that these models are highly effective for real estate price predictions. However, ensemble models such as stacking and averaging show even better results but with greater computational burden. Further, it is inferred that traditional and alternative metrics generate similar findings. Lastly, this study highlights that predictive performance is better for middle-range rents compared to the extremes (lower and higher deciles). These findings are useful to real estate stakeholders for incorporation into expert systems used for automated valuation models.

***Keywords**: automated valuation models, machine learning, model selection, rent prediction, residential real estate market*

Declaration of Interest

The authors have no relevant financial or non-financial interests to disclose.

1. Introduction

In recent years, the emergence of machine learning (ML) and artificial intelligence (AI) has entered several fields, including real estate. In the real estate sector, it has been automating the property valuation process. Automated valuation gives stakeholders more independence and power (Steurer et al., 2021). Real estate agents can improve their operational efficiency by automating routine valuation tasks with ML models. For instance, automated property valuations free up time to focus on client service and unburdening the clients. Furthermore, buyers, sellers, investors, and renters become less dependent on real estate agents for valuations. Next to that, lenders and insurance companies benefit from accurate valuations because they are critical to assessing risk. ML models can improve the accuracy of valuations, leading to more informed lending decisions and more accurate insurance premium calculations. On top of that, policymakers, urban planners, and contractors can use predictive models to understand the influence of various factors, such as an improvement in energy consumption due to energy renovations, on sale, and rent prices. This understanding can lead to more effective urban planning and policy decisions.

Predictive real estate price research has been increasing, as we will note in *Section 2. Related Work*. However, many studies tend to rely on a limited selection of models, often following what has been previously used in the literature, without rigorously cross-validating these models on their datasets. Researchers seem to favor established models out of convention or convenience, rather than through a unified, data-driven benchmark. This reveals the absence of a standardized framework for model selection, making it challenging to systematically compare studies. Our primary research objective is to address this gap by conducting an extensive comparison of commonly used ML models for rent prediction, aiming to provide a step towards a more comprehensive guide for selecting the appropriate models for real estate prediction problems. Furthermore, we introduce several, primarily linear, ML models that have not yet been applied in real estate predictive modeling.

Additionally, predictive real estate price research typically employs a limited set of evaluation metrics, with no consensus on the most appropriate ones for this type of research. As our second research objective, we will utilize several alternative evaluation metrics, alongside traditional ones, which Steurer et al. (2021) argue are better suited for evaluating real estate price prediction models. These alternative metrics have, to our knowledge, not been adopted in the literature, making our study one of the first to

compare them against traditional metrics to determine if they yield similar or different results.

Moreover, we investigate whether there are differences in evaluation metrics between high, medium, and low real estate prices, in this study rents, by evaluating metrics for the learned models per decile. To our knowledge, this is the first study to examine this issue. Thus, our third research goal is to show whether there are differences in the evaluation metrics across deciles and to assess whether the position of the prediction in the distribution should be considered to better inform about price variability.

Lastly, while much of the focus in real estate prediction research has been on sale prices, rent prices hold significant economic relevance and deserve more in-depth exploration. Rent prices serve as a critical indicator of housing affordability, a growing concern in many urban areas where the cost of living is rising more rapidly than wages. Consequently, policymakers and urban planners increasingly rely on rental market data to evaluate housing accessibility and to inform the development of effective housing policies (ElFayoumi et al., 2021). Additionally, for investors, rental yields are a key factor in assessing returns on investment in residential properties, further underscoring the value of accurate rental price prediction models.

In summary, our study contributes to the body of knowledge by utilizing a comprehensive dataset of Belgian residential rental properties in 2022 and employing data cleaning and preprocessing to ensure data quality. We create a wide range of ML regression models, leveraging hyperparameter tuning and cross-validation to optimize model performance. The evaluation of the learned models, both on the train set, test set, and per decile of the test set, integrating both traditional and alternative measurement methods provides a thorough assessment. Ultimately, there is also the added value of this research to the field as there has been relatively less research on predicting rents than on predicting sale prices.

The remainder of this paper is organized as follows. Section 2 describes related works on prediction models in real estate research. Section 3 describes the data and the methodology. Section 4 presents and discusses the results. Section 5 provides the conclusion with a summary of the key findings, limitations, and directions for future research.

2. Related work

A considerable literature for predicting property prices is now available. However, from Table 1, it is notable that the focus is rather on predicting sales prices than rental prices. We state this because 9 of the 38 studies in Table 1 predicted rents. When prediction models are applied, a handful of models are usually compared but there is a wide range of models used across the different studies. Linear regression models that have been applied in the literature of automated real estate valuation listed in Table 1 are Linear (LR, based on ordinary least squares), Ridge, Least Absolute Shrinkage and Selection Operator (LASSO), and Elastic Net (EN). Other traditional statistical regression models that can capture more complex relations that have been applied are Generalized Linear Models (GLM), Geographically Weighted Regression (GWR), Spatial Autoregressive Regression (SAR), and Generalized Additive Models (GAM). Nonlinear ML regression models that have been applied are Classification And Regression Trees (CARTs), ensembles – including tree-based models –, Artificial Neural Networks (ANNs), K-Nearest Neighbors (KNN), and Support Vector Regression (SVR). Tree-based models include models based on aggregations of CARTs. Bagging, which is the combining of multiple models, of CARTs results in the Random Forest (RF) and Extra Trees (ET). Boosting, which is constructing an ensemble of CARTs sequentially with a focus on the errors made in the previous iteration, results in gradient boosting models amongst them eXtreme Gradient Boosting (XGBoost), Light Gradient Boosting (LightGBM), Category Boosting (CatBoost) and Adaptive Boosting (AdaBoost). Also, there are other ensemble models which include voting, averaging, stacking, bagging, and boosting. These ensemble models combine the predictions of a variety of ML models into a single prediction. Lastly, ANNs include MultiLayer Perceptron (MLP), Long-Short-Term-Memory (LSTM), and Convolutional Neural Networks (CNN).

Best performing and therefore popular models from the studies, which are reported in Table 1, are ensembles – primarily RF and XGB – and ANNs. Many studies use tree-based models, such as RF and GB. They are, as already stated, ensemble models, built from CARTs. These CARTs are inherently prone to overfitting, but capture the complex relationships in the data on which they learn exceedingly well (Hastie et al., 2009). By aggregating the CARTs, overfitting can be mitigated, and the generalizability and robustness of the modeling increases (Kuhn & Johnson, 2013). This is certainly useful for real estate datasets that are noisy and heterogeneous. In other words, the

reasons, which have also been brought forward by Antipov & Pokryshevskaya (2012), for using tree-based models are linked to the characteristics of real estate data. As such, the tree-based models do not require a detailed model specification. The relationships for features in real estate data are often non-linear and interactive (Krämer et al., 2021; Lenaers & De Moor, 2023; Lorenz et al., 2022; Rampini & Re, 2021; Yazdani, 2021; Yilmazer & Kocaman, 2020). Capturing such relationships is something tree-based models excel at because they are constructed from CARTs. In addition, real estate features contain a range of data types, from continuous to categorical. Tree-based models by nature handle these well. Furthermore, irrelevant, and correlated features are often present in real estate data. Tree-based models are capable of automatically filtering out these less informative features while building the CARTs. In addition, there are often missing values in real estate data which is not a problem for tree-based models. Moreover, both RF and ET are also robust to outliers, thanks to bagging (Kuhn & Johnson, 2013). However, classical methods such as LR remain prevalent in the literature for their interpretability and ease of implementation (Valier, 2020). That interpretability is crucial to understanding drivers of real estate prices. For example, traditional LR provides insight into the direction, magnitude, and significance of features.

The choice between ML or more traditional models is hence determined by predictive power and interpretability (Valier, 2020). On the one hand, ML has better predictive power but provides lower interpretability. On the other hand, traditional models provide more interpretability but capture fewer complex relationships, so they have lower predictive power. However, with the emergence of Explainable AI (XAI), the tradeoff between interpretability and predictive power will prevail less because the black-box ML models can be interpreted and explained (Tekouabou et al., 2024; Valier, 2020). The black-box nature can thus be cracked open. This will create a synergy between the superior predictive power of ML models and the ease of interpretability of those ML models with XAI (Tekouabou et al., 2024).

However, there are weaknesses in prior research for real estate valuation with ML. First, a limited number of models are often compared. This is coupled with the consideration of a limited number of evaluation metrics to study the performance of the models. As remarked in Table 1, often used metrics are root mean squared error (RMSE), mean absolute percentage error (MAPE), coefficient of determination (R^2), and median absolute error (MAE) which provide insight into the predictive power and reliability of the models. However, Steurer et al. (2021) express the criticism that some of them are not

necessarily the optimal metrics because some of the popular metrics – MAPE and R^2 – do not symmetrically treat prediction errors. Steurer et al. (2021) therefore propose the use of log median prediction error (LMDPE), MAE, max-min mean absolute prediction error (mmMAPE), log root mean squared error (LRMSE), RMSE, max-min percentage error range (mmPER(x)), and inter-quartile range in ratios (IQRat) in the evaluation of real estate valuation models.

Beyond that, from Table 1 emerges that there is also the issue that sample sizes between studies vary considerably, from a few hundred to millions of observations. Large data sets can have the advantage of being representative and as such generate more robust and reliable models and deeper insights (Kuhn & Johnson, 2013). This is important for ML models, as they benefit from larger sample sizes of heterogeneous data to learn complex relationships and patterns in data and thus have better predictive power, as they generalize better. Traditional models can work with relatively smaller homogeneous data sets to generate valuable insights by focusing on a specific location or specific segment, for example. It is important to note that overfitting can be a problem with complex models trained on small samples, making the learned relationships and patterns have little generalizability (Kuhn & Johnson, 2013). In addition to the influence of data quantity, there is also the influence of data quality which is also a common issue in training real estate prediction models (Iban, 2022; Tekin & Uçal Sarı, 2022).

Furthermore, it is also quite pertinent that the comparison of existing research is not self-evident. After all, there is no unified framework for reporting metrics. First, different metrics are used. In addition, it is often unclear on which data the performance was checked, whether on the train, or test dataset. In this regard, often only the performance on one of these datasets is reported. Another issue is that target variables differ due to specific factors such as predicting rent or sales price, predicting prices per square meter, working with or without inflation corrections, applying logarithmization of the target variable, or differing currencies. In addition, data collection, data cleaning, and data pre-processing are not standardized and as such have their respective influences on modeling real estate prices.

On a final note, besides the variety in modeling, which has an impact on performance, there is also the variety of features and types of features included in research that have an impact. Following Potrawa & Tetereva (2022), there are three major categories into which you can classify features. First, structural (S) features include characteristics of the property, such as the size and age of the property. Location (L)

factors are for example proximity to supermarkets and schools. Socioeconomic (N) factors are considered factors about the neighborhood of the property and can include demographics and income levels. The line between location and socioeconomic factors is sometimes blurred, so they are often treated together or interchangeably. This is because they are often related to each other. Integrating relevant factors about location and socioeconomics into real estate prediction models often leads to improved predictive power (Gao et al., 2022; Talaga et al., 2019; Zhou et al., 2019). In addition, there is a fourth category, namely the macroeconomic features that attempt to include the impact of the economic cycle, such as mortgage rates. However, macroeconomic characteristics are used less in predictive real estate price research as remarked by Zulkifley et al. (2020) and supported by Table 1.

3. Data and methodology

3.1. Data collection and data cleaning

This study uses a dataset that is provided by Realo nv. Realo nv. is a Belgian real estate data platform that provides real estate professionals and other stakeholders with data-driven insights that assist buyers, sellers, renters, and landlords in buying, selling, and renting their homes. The provided dataset contains 38 features and the target variable, monthly rent, of 87,188 Belgian residential real estate property listings from January 1st, 2022, until December 31st, 2022.

This raw data is cleaned. First, less relevant features, including the date of first and last publication, and type of address are removed. High-cardinal features that contain similar content are also removed. Second, duplicate observations are removed from the data. Next, observations for which there is no price or name of the municipality are dropped. Then, missing values for each observation are examined. If it exceeds 50% for an observation, the observation is dropped. In addition, features that have more than 80% missing values are dropped. This whole data-cleaning process results in a sample of 79,735 observations with 25 features and the target variable, monthly rent (in EUR). To provide insight into the cleaned dataset, an overview and summary statistics are provided in Table 2.

Table 1 Previous real estate valuation research with ML

Author(s)	Sample size	Feature Types	Target Variable	Evaluation Metrics	Models	Best Model
Alkan et al. (2022)	200	S, L	rent prices	MAE, R^2, RMSE	KNN, RF, SVR	SVR
Baur et al. (2023)	63,828	S, L	rent and sale prices	MAE, MAPE, RMSE	EN, LightGBM, LR, RF, SVR	Tree-based
Bilgilioğlu & Yılmaz (2023)	1,982	S, L, N	sale prices	AIC, COD, MAPE, MSE, PRD, R^2, RMSE	ANN, CART, CHAID, SVR	ANN
Birkeland et al. (2021)	18,795	S, L, N	sale prices	MAPE, MedAPE	CART, Ensemble (Stacking), ET, RF, XGBoost	Ensemble
Chou et al. (2022)	13,220	S, L, N	sale prices	MAE, MAPE, R^2, RMSE, SI	ANN, CART, Ensembles, LR, SVR	Ensemble
Choy & Ho (2023)	24,317	S, L	sale prices	MAPE, MSE, R^2, RMSE	ET, KNN, LR, RF	RF
Çılgın et al. (2023)	16,578	S, L, N	sale prices	MAE, MAPE, MSE, RMSE	ANN, LASSO, LR, Ridge, XGBoost	XGBoost
Fedorov & Petrichenko (2020)	5,491	S	sale prices	R^2	AdaBoost, CatBoost, LR, XGBoost	CatBoost
Fourkiotis & Tsadiras (2023)	1,458	S	sale prices	R^2, RMSLE	Ensembles (Averaging, Voting), GB, LASSO, LightGBM, RF, ridge, SVR, XGBoost	Ensemble
Gao et al. (2022)	63,426	S, L, N	sale prices	MAPE, MedAPE, PE(x), R^2	CART, EN, GB, GWR, LASSO, LR, MLP, RF, Ridge, SVR, XGBoost	RF, XGBoost
Hinrichs et al. (2021)	±190,000	S, L, N	sale prices	MAPE, MedAPE, RMSE	EN, LASSO, LR, Ridge	Ridge
Hjort et al. (2022)	126,719	S, L, N	sale prices	MedE, PE(x), R^2, RMSE	ANN, Ensembles, LR, RF, XGBoost	Ensemble
Ho et al. (2021)	39,554	S, L	sale prices	MAPE, MSE, RMSE	GB, RF, SVR	Tree-based
Iban (2022)	1,002	S, L	sale prices	COD, MAPE, PRD, R^2, RMSE	GB, LightGBM, RF, XGBoost	XGBoost
Kiely & Bastian (2020)	12,012,780	S, L, N	sale prices	MAE, MSE, R^2, RMSE	ANN, GB, GLM, RF	ANN
Krämer et al. (2021)	81,166	S, L, N	sale prices	MAPE, MedAPE, PE(x)	LR, XGBoost	XGBoost
Krämer et al. (2023)	1,212,546	S, L, N	sale prices	MAPE, MedAPE, PE(x), R^2	ANN, GAM, LR, XGBoost	XGBoost
Lenaers & De Moor (2023)	78,788	S, L	rent prices	MAE, MAPE, MedAE, MedAPE, RMSE	CatBoost, Ridge, XGBoost	CatBoost
Lenaers et al. (2023)	18,935	S, L	rent prices	MAE, MAPE, RMSE	LR, RF, XGBoost	XGBoost
Lorenz et al. (2022)	52,966	S, L, N	rent prices	/	GAM, LR, SAR, XGBoost	XGBoost
Mora-Garcia et al. (2022)	39,943	S, L, N	sale prices	MAE, MSE, R^2, RMSE	ET, GB, LightGBM, LR, RF, XGBoost	Ensemble
Pai & Wang (2020)	32,215	S, L	sale prices	MAPE, NMAE	ANN, CART, SVR	SVR
Potrawa & Tetereva (2022)	1,844	S, L, N	rent prices	R^2, RMSE	LR, RF	RF
Rampini & Re (2021)	1,228	S	sale prices	MAE	ANN, EN, XGBoost	ANN
Sapakova & Sapakov (2024)	3,882	S	sale prices	MAE, MSE, R^2, RMSE	CART, EN, LASSO, LR, RF, Ridge, SVR	RF
Sevgen & Tanrivermiş (2024)	1,315,675	S, L	sale prices	adj. R^2, MAE, MSE, R^2, RMSE	ANN, KNN, LR, RF, SVR	RF
Sharma et al. (2024)	2,930	S, L, N	sale prices	adj. R^2, MAE, MSE, R^2	LR, MLP, RF, SVR, XGBoost	XGB
Stang et al. (2022)	1,212,546	S, L, N	sale prices	MAPE, MedAPE, PE(x), R^2	GAM, LR, XGBoost	XGBoost

Talaga et al. (2019)	21,000	S, L	sale prices	MAPE	CART, Ensembles, MLP, LR, RF, XGBoost	Ensemble
Tekin & Uçal Sarı (2022)	3,514	S, L	sale prices	MAPE, R^2	CART, LR, RF, XGBoost	RF
Trawiński et al. (2017)	12,439	S	sale prices	MAE	CART, Ensembles, MLP	Ensemble
Waddell & Besharati-Zadeh (2020)	363,010	S, L, N	rent prices	RMSE	LR, RF	RF
Xu & Li (2021)	± 148,000	S, L	sale prices	RMSE	AdaBoost, Ensemble, GB, LightGBM, RF, XGBoost	Ensemble
Yazdani (2021)	1,061	S, L, N	sale prices	MAE, R^2, RMSE	ANN, KNN, LR, RF	RF
Yilmazer & Kocaman (2020)	1,162	S, L	sale prices	adj. R^2, R^2, RMSE	LR, RF	RF
Yoshida & Seya (2021)	4,588,632	S, L	rent prices	MAE, MAPE, RMSE	ANN, LR, RF, XGBoost	XGBoost
Zhan et al. (2023)	1,898,175	S, L, M	sale prices	adj. R^2, EVS, MAD, MAE, MAPE, ME, MGD, MPD, MSE, PL, R^2, RMSLE, RMSE	AdaBoost, CatBoost, CNN, ensembles, GB, KNN, LSTM, RF, SVR, XGBoost	CatBoost
Zhou et al. (2019)	76,487	S, L	rent prices	MAE, MAPE, RMSE	ET, GB, KNN, LASSO, LR, MLP, RF, Ridge	RF

With *L = Location, M = Macroeconomic, N = Socioeconomic/Neighborhood, S = Structural;*

AIC = Akaike Information Criterion, adj. R2 = Adjusted Coefficient Of Determination, COD = Coefficient Of Dispersion, EVS = Explained Variance Score, MAE = Mean Absolute Error, MAD = Median Absolute Deviation, MAPE = Mean Absolute Percentage Error, MGD = Mean Gamma Deviance, ME = Max Error, MedAE = Median Absolute Error, MedAPE = Median Absolute Percentage Error, MedE = Median Error, MPD = Mean Poisson Deviance, MSE = Mean Squared Error, PE(x) = Percentage Predicted Error within x%, PL = Pinball Loss, PRD = Price-Related Differential, R2 = Coefficient Of Determination, RMSE = Root Mean Square Error, RMSLE = Root Mean Square Log Error, and SI = Synthesis index;

AdaBoost = Adaptive Boosting, ANN = Artificial Neural Network, CART = Classification And Regression Tree, CatBoost = Category Boosting, CHAID = Chi-Squared Automatic Interaction Detection, CNN = Convolutional Neural Networks, EN = Elastic Net, ET = Extra Trees, GAM = Generalized Additive Model, GB = Gradient Boosting, GLM = Generalized Linear Models, GWR = Geographically Weighted Regression, KNN = K-Nearest Neighbours, LASSO = Least Absolute Shrinkage and Selection Operator, LightGBM = Light Gradient Boosting, LR = Linear Regression, LSTM = Long-Short-Term-Memory, MLP = MultiLayer Perceptron, RF = Random Forest, SAR = Spatial Autoregressive Regression, SVR = Support Vector Regression, and XGBoost = eXtreme Gradient Boosting.

Table 2 Summary statistics for the target variable and features

Feature name	Data type	Nr. of observations	Min.	Max.	Mean	Median	St. Dev.	Skew.	Kurt.	Cor. with target var.
Monthly rent price (in EUR)	Continuous	79,735	370	4,100	903.07	800	366.27	2.77	12.37	
Area (in m2)	Continuous	60,520	62	5,332	706.84	360	890.70	2.53	6.90	0.00
Build year	Continuous	43,241	1,700	2,023	1,990.95	2,006	35.33	-1.72	4.45	-0.12
Building area (in m2)	Continuous	59,106	25	2,637	237.04	128	316.42	3.71	16.79	0.02
Distance to bus stop (in m)	Continuous	70,247	1	5,008	185.12	142	195.49	5.41	51.02	0.08
Distance to school (in m)	Continuous	70,247	0	7,857	353.70	246	365.34	3.70	25.49	0.06
Distance to train station (in m)	Continuous	70,247	0	13,876	1,926.79	1,213	1,947.13	1.87	3.31	0.00
Distance to the village centre (in m)	Continuous	70,247	1	13,948	1,488.03	1,019	1,503.20	2.34	7.03	0.00
Double glass (0 = no, 1 = yes)	Binary	79,735	0	1	0.72	1	0.45	-0.98	-1.04	0.07
Energy consumption (in kWh/m2 per year)	Continuous	50,490	8	1,482	225.44	188	157.62	1.85	5.75	-0.06
Floor	Integer	66,816	0	38	1.44	1	1.84	3.37	24.83	-0.08
Garden (0 = no, 1 = yes)	Binary	79,735	0	1	0.65	1	0.48	-0.63	-1.60	0.08
Garden area (in m2)	Continuous	46,696	24	3,970	483.61	223	663.63	2.65	7.91	-0.02
Habitable area (in m2)	Continuous	78,393	24	510	102.09	91	45.50	1.55	3.92	0.55
Housing type (0 = apartment, 1 = house)	Binary	79,735	0	1	0.26	0	0.44	1.11	-0.78	0.29
*Locality****	Nominal	78,788	/	/	/	/	/	/	/	/
*Mobility score**	Continuous	79,551	0.25	1	0.79	0.82	0.12	-0.94	0.64	-0.02
New build (0 = no, 1 = yes)	Binary	79,735	0	1	0.13	0	0.34	2.17	2.71	0.03
Number of bathrooms	Integer	73,140	1	4	1.26	1	0.52	2.13	5.04	0.46
Number of bedrooms	Integer	79,735	1	6	2.02	2	0.90	0.86	0.93	0.53
Number of floors in the building	Integer	45,975	0	43	2.96	3	2.57	4.55	44.46	0.09
Number of parking spaces	Integer	39,801	0	15	1.37	1	1.32	3.60	24.41	0.16
Number of sides (1, 2, 3, 4)	Ordinal	45,417	1	4	2.45	2	0.78	0.74	-0.22	0.17
Number of toilets	Integer	63,706	0	6	1.29	1	0.59	1.30	3.65	0.51
*Proneness to flooding***	Nominal	78,788	/	/	/	/	/	/	/	/
Solar panels (0 = no, 1 = yes)	Binary	79,735	0	1	0.08	0	0.27	3.09	7.53	0.03

** The mobility score indicates the environmental impact of travel from one's property. It shows how well facilities such as schools, stores, and public transport, ... are accessible by bike or on foot.*
*** Categories are 'yes', 'possible', and 'no'. *** This feature had 578 categories*

3.2. Data pre-processing

Before model training, we pre-process the data in four steps. First, the dataset was randomly split into two parts using the train-test data split: training (80%) and test (20%) datasets, respectively for model training and model evaluation. Second, we handled missing values with a straightforward imputation method, which consists of replacing missing values with either the mean for numeric (integer and continuous) features or the mode for categorical (binary and nominal) features. Third, we one-hot encode the nominal feature 'Proneness to flooding'. Fourth, for the feature 'Locality', we use the target encoder from the Python package category_encoders 2.6.3 because one-hot encoding this high cardinal categorical feature with 578 categories would lead to high dimensionality.

3.3. ML algorithms

In our research, we test a variety of regression algorithms selected by previous research and all available algorithms in the Python package PyCaret (Ali, 2023). As such, the following algorithms were selected: LR, LASSO regression, ridge regression, EN regression, least angle regression (LAR), LASSO least angle regression (LLAR), orthogonal matching pursuit regression (OMP), Bayesian ridge regression (BR), automatic relevance determination regression (ARD), passive aggressive regressor (PAR), random sample consensus regression (RanSaC), TheilSen regressor (TR), Huber regressor (Huber), SVR, KNN regressor, CART regressor, RF regressor, ET regressor, AdaBoost regressor, GB regressor, MLP regressor, XGBoost regressor, LightGBM regressor, and CatBoost regressor. Furthermore, we add a GAM, symbolic regressor (SR), and ensembles (averaging model and stacking model based on RF, XGBoost, and CatBoost) into the mix.

3.4. Model training

For ML, hyperparameter optimization is also important for solving the problem of choosing a set of optimal hyperparameters for a learning algorithm. The models are trained using the train dataset with hyperparameter tuning and 10-fold cross-validation to reduce the likelihood of overfitting We use the tree-structured Parzen estimator, a

Bayesian optimization approach, for hyperparameter tuning that has been shown to outperform traditional methods (Yang & Shami, 2020). During the hyperparameter tuning process, we use MAPE as a scoring metric.

3.5. Model evaluation

To evaluate the models, we calculate traditional – as deduced by Table 1 – and alternative metrics for real estate applications, as proposed by Steurer et al. (2021). The selected metrics are defined in Table 3.

The evaluation metrics are calculated on both the train and test datasets, to obtain an impression about the degree of overfitting. Furthermore, metrics per decile of the test data are also considered for the evaluation of the models.

Table 3 Selected evaluation metrics and their definition

Metric	Definition	
Coefficient of determination	$R^2 = 1 - \left(\frac{\sum_{i=1}^{n}(y_i - f_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}\right),$	(1)
Coefficient of dispersion	$COD = \frac{1}{n}\sum_{i=1}^{n}\left\lvert\left[\left(\frac{y_i}{f_i}\right)/med\left(\frac{y_i}{f_i}\right)\right] - 1\right\rvert$	(2)
Inter-quartile range in ratios	$IQRat = \left[\ln\left(\frac{y_i}{f_i}\right)\right]_{75} - \left[\ln\left(\frac{y_i}{f_i}\right)\right]_{25},$	(3)
Root mean square error	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - f_i)^2},$	(4)
Mean absolute error	$MAE = \frac{1}{n}\sum_{i=1}^{n}\lvert y_i - f_i\rvert,$	(5)
Mean absolute percentage error	$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left\lvert\frac{y_i - f_i}{y_i}\right\rvert,$	(6)
Max-min mean absolute prediction error	$mmMAPE = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\max(y_i, f_i)}{\min(y_i, f_i)} - 1\right),$	(7)
Max-min percentage error range	$mmPER(x) = \frac{1}{n}\sum_{i=1}^{n}\left\lvert\frac{\max(y_i, f_i)}{\min(y_i, f_i)} - 1\right\rvert > x,$	(8)

with n the number of observations in the respective dataset, y_i the actual rent for property i and f_i the predicted rent for property i.

4. Results and discussion

The results were generated in Python, using the Python package Pycaret 3.2.0 for data pre-processing and model training of LR, LASSO regression, ridge regression, EN regression, LAR, LLAR, OMP regression, BR regression, ARD regression, PAR, RanSaC regression, TR, Huber regressor, SVR, k–nearest neighbors regressor, CART regressor, RF regressor, ET regressor, AdaBoost regressor, GB regressor, MLP regressor, XGBoost regressor, LightGBM regressor, CatBoost regressor, averaging and stacking that uses the Python packages numpy 1.26.0, pandas 2.1.1, sklearn 1.2.2, xgboost 2.0.2, catboost 1.2, lightgbm 4.1.0 for the models. The Python package optuna 3.4.0 was used for hyperparameter tuning. For the GAM and SR, the Python packages pygam 0.9.0, and gplearn 0.4.2 were respectively used.

First, it is noticeable that the evaluation metrics of the models in panel (A) of Table 4, which contains mainly linear models, are worse for both the test and train set than those in panel (B) of Table 4, which contains mainly (tree-based) ensemble models and some more complex ML models. When looking at the differences between the evaluation metrics for the train and test set, it is noticeable that for the linear models in panel (A) of Table 4, there are smaller differences. For panel (B) of Table 4, those differences are larger, which could indicate overfitting for the latter panel. Looking deeper into panel (B), we infer that the Adaboost model does not perform as well as the other members in the tree-based ensemble model family. As also highlighted from the literature in Table 1, this study confirms that XGBoost, CatBoost, and RF, score well on real estate datasets. The performance discrepancy between the linear and tree-based models can be attributed to the inherent strengths of these tree-based models to, among others, capture complex, non-linear, and interactive patterns within real, while linear models, assume a linear relationship between the target variable and the features, which oversimplifies the relations in the data.

Among the tree-based ensembles in panel (B) of Table 4, the CatBoost model has the majority of the better evaluation metrics except mmPEr and IQRat. For mmPEr and IQRat, ET performs slightly better. However, the question is whether the magnitude of this better performance is noteworthy. For the difference between the metrics of the train and test set, the CatBoost model seems to perform well in this regard because the differences are small. However, what is noticeable is that the ET model seems to overfit

on the train set because it has nearly perfect metrics on that set, but the metrics on the test set have similar values to the other members of the tree-based ensemble model family.

Note further in panel (B) of Table 4 that for LightGBM the differences in the metrics between the train and test set are very small, indicating that it overfits the least of the tree-based models. Less overfitting could lead to less frequent retraining of the model because it could generalize better. Furthermore, the differences between the evaluation metrics of the tree-based models are not particularly large. For example, for MAE the difference between the LightGBM and the CatBoost model is just 4.6 (EUR) to the LightGBM's disadvantage. The question thus arises as to whether these slightly worse metrics outweigh whether the model has less overfitting.

Furthermore, the metrics in panel (A) of Table 4 report the PAR as by far the worst model. The PAR is followed by the SVR, which contradicts Alkan et al. (2022) and Pai & Wang (2020) who found in their research that SVR performed better than RF and NN respectively. In addition, it is inferred that for the linear models – LR, LASSO, ridge, EN, LAR, LLAR, OMP, BR, ARD, PAR, Ransac, TR, and Huber – the metrics are almost all the same. This suggests that those linear models, that are variants of the original LR model, are thus toning down to that LR model. Additionally, when looking at the differences between evaluation metrics of linear models between the train and test set, it is noted that they are small, suggesting that the linear models have little overfitting on the training data, which is also observed by Lenaers et al. (2023).

Furthermore, note in Table 4 that traditional metrics, such as RMSE, MAPE, COD, R2, and MAE draw similar conclusions as the alternative metrics – mmMAPE, mmPER, and IQRat – put forward by Steurer et al. (2021). The best-performing models in this study, according to the evaluation metrics, which are nearly equal for the two ensembles, are averaging and stacking (of the RF, XGBoost, and CatBoost models). However, it is noted that differences between train and test sets are larger than for the CatBoost model, which would indicate that the stacking and averaging ensembles have more overfitting. In addition, one can criticize whether for the small improvements in evaluation metrics compared to the metrics of the CatBoost, ET, and XGBoost it is justifiable to make this more complex model.

Evaluation metrics by decile for the test dataset for some of the best-performing models – stacking, CatBoost, XGBoost, and RF – are shown in Fig. 1 (a), 1 (b), 1 (c), 1 (d), 1 (e), 1 (f), and 1 (g) for MAE, MAPE, COD, RMSE, mmPER, IQRat, and mmMAPE respectively. Notice visually that the differences between the selected models are not

large. It is also inferred that all the metrics have similar trends. There is lower performance of the models for the extremes, i.e., deciles 1, 2, and 3 and deciles 8, 9, and 10. The best metrics are obtained for the observations in the middle range. However, the strength of the side where the performance is lower depends on the metric. Thus, it is observed that the absolute evaluation metrics, RMSE and MAE, have a lower performance in decile 10 because the error is higher there. Note that the differences for RMSE and MAE between decile 4 and decile 10 differ by about a factor of 5. With MAPE, a relative metric, this is the opposite, since decile 1 gives poorer evaluation metrics there. This contradiction is logical, since deviations in decile 1 have a small absolute impact, but can be relatively large. This is different for mmMAPE and mmPER, where both decile 1 and decile 10 have similarly worse evaluations relative to the middle-range deciles. For IQRat and COD, the lower deciles are worse in terms of evaluation metrics than relative to the middle-range deciles. However, the higher deciles are even worse because they have inferior values than the metrics of the lower deciles. The difference between the deciles with the highest value and the lowest value for MAPE, COD, mmPER, IQRat, and mmMAPE is about a factor of 2.

It is also important to critically consider the results and the time to train the models with hyperparameter tuning and 10-fold cross-validation. After all, hyperparameter tuning and training the random forest took over 45 hours, and logically the ensembles, containing the RF, require even more computational time. This is considerably more than, for example, the LightGBM which took little more than 10 minutes. Hence, the question arises here whether, for a similar performance between the tree-based models, one can defend the very long training time of the RF. In addition, the training time of the linear models which took on average less than a minute, is relatively seen a lot less time than that for the hyperparameter tuning and training of the tree-based ensembles.

The choice of whether a model is appropriate for one's real estate application depends on several considerations in which the modeler has a share. For example, the focus may be on the differences between the test and train set for the metric or better said, minimal overfitting. On the other hand, there is a possible trade-off between training time and the performance of the models. Another interesting consideration is the one towards interpretation with XAI techniques, including SHapley Additive exPlanations (SHAP). SHAP is a model-agnostic technique that helps to understand the importance of features in making a prediction. For tree-based models, there is a speed-up version for calculating SHAP via TreeSHAP (Molnar, 2022), and for linear models with LinearSHAP. Although

the averaging and stacking models have the best evaluation on the test set, for both traditional and the alternative metrics proposed by Steurer et al. (2021), in that case, an argument can be made in favor of tree-based and linear models at the expense of the complex averaging and stacking ensembles in this study.

5. Conclusions

This study compared traditional and alternative evaluation metrics among 28 ML models for predicting rents based on 25 features and the target variable monthly rent from cleaned Belgian residential real estate data of 2022.

It follows from the comparison that averaging and stacking ensemble models based on RF, XGBoost, and CatBoost performed best. However, the results of the averaging and stacking ensemble models are close to those of the tree-based ensemble models, including RF, XGBoost, and CatBoost. The good performance of the tree-based models confirmed previous research. The practical implication is that real estate price modelers are well advised to look toward tree-based ensemble models. Depending on the computational capacity, albeit time or computer resources, at hand, modelers can choose between (tree-based) ensemble models. Accurate models will help real estate agents in delivering precise valuations, and focus on other matters linked to the rental of a property. It will also help other stakeholders such as tenants to determine what a realistic rent price is. Also, with accurate models, investors can check whether it is profitable to make certain investments. Moreover, the government can intervene if investors do not make the necessary investments that are desired, think about energetic investments.

Further as was already made apparent from the literature, more complex ML models outperform simple linear models in terms of evaluation metrics on both train and test sets. Practitioners should favor these complex models when predictive accuracy is crucial. In addition, it would also be interesting for researchers to interpret and study the models to gain new insights, compared to classical econometric research. However, this research does not confirm the good performance of SVR found by some authors.

In addition, we were able to infer from the results that the traditional metrics – RMSE, MAPE, MAE, COD, R^2 – and alternative metrics – mmPER, mmMAPE, IQRat – proposed by Steurer (2022) yield approximately the same findings. Future research should continue to validate these metrics across different real estate datasets and contexts to ensure their robustness.

Table 4 Evaluation metrics for the ML models

(A)

Model	SR	LR	LASSO	Ridge	EN	LAR	LLAR	OMP	BR	ARD	PAR	RanSaC	TR	Huber
	Train dataset													
MAE	177.8	139.7	139.7	139.7	139.6	187.8	139.7	140.4	139.7	139.7	328.3	176.3	140.1	148.8
MAPE	0.182	0.154	0.154	0.154	0.153	0.217	0.154	0.155	0.154	0.154	0.411	0.196	0.153	0.155
COD	0.192	0.151	0.151	0.151	0.151	0.275	0.151	0.152	0.151	0.151	0.787	0.197	0.148	0.158
RMSE	286.9	206.7	207.5	206.7	207.7	264.3	207.5	207.9	206.7	206.7	410.3	262.7	211.3	238.6
R^2	0.387	0.682	0.679	0.682	0.679	0.480	0.679	0.678	0.682	0.682	0.254	0.486	0.667	0.576
mmMAPE	0.222	0.172	0.171	0.172	0.171	0.278	0.171	0.172	0.172	0.172	0.702	0.227	0.169	0.178
mmPER	0.640	0.589	0.588	0.589	0.587	0.695	0.588	0.591	0.589	0.589	0.858	0.659	0.580	0.578
IQRat	0.141	0.119	0.118	0.119	0.118	0.168	0.118	0.119	0.119	0.119	0.142	0.148	0.115	0.117
TT	00:02:43	00:00:27	00:00:28	00:00:28	00:00:28	00:00:28	00:00:27	00:00:27	00:00:28	00:00:33	00:01:17	00:04:16	00:08:06	00:00:38
	Test dataset													
MAE	179.6	141.4	141.3	141.4	141.3	191.1	141.3	142.0	141.4	141.4	330.1	178.4	141.9	150.9
MAPE	0.184	0.155	0.155	0.155	0.155	0.219	0.155	0.156	0.155	0.155	0.412	0.197	0.155	0.157
COD	0.193	0.152	0.152	0.152	0.151	0.283	0.152	0.153	0.152	0.152	0.365	0.196	0.150	0.160
RMSE	288.0	211.2	211.7	211.2	211.9	272.1	211.7	212.1	211.2	211.2	411.9	264.7	215.2	242.5
R^2	0.379	0.666	0.665	0.666	0.664	0.446	0.665	0.663	0.666	0.666	0.270	0.475	0.653	0.560
mmMAPE	0.224	0.173	0.172	0.173	0.172	0.312	0.172	0.174	0.173	0.173	0.447	0.229	0.171	0.181
mmPER	0.642	0.593	0.589	0.593	0.586	0.696	0.589	0.591	0.593	0.593	0.862	0.665	0.584	0.580
IQRat	0.142	0.119	0.119	0.119	0.119	0.166	0.119	0.120	0.119	0.119	0.143	0.150	0.115	0.119

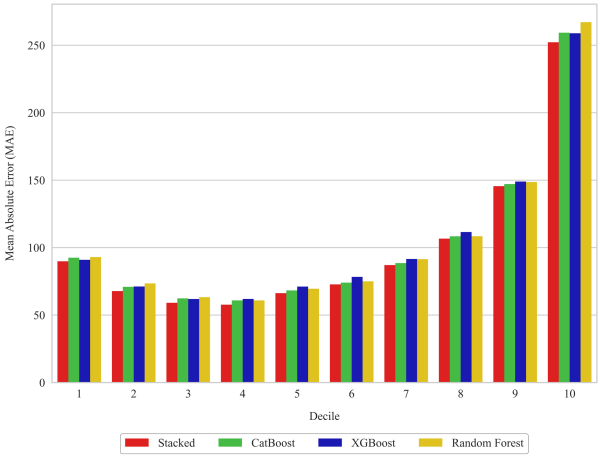
With TT = Training Time in hours, minutes and seconds

(B)

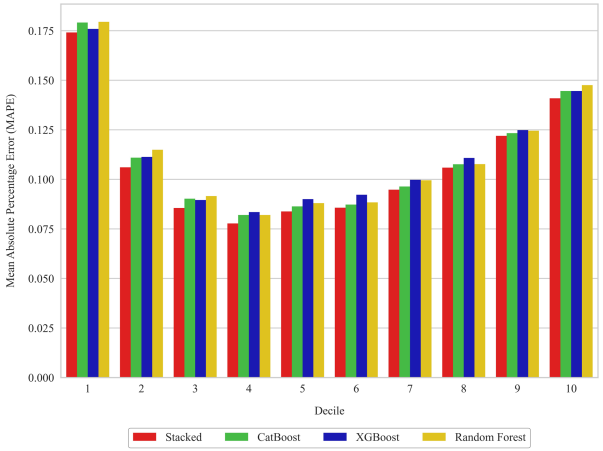
Model	ET	Adaboost	GB	XGBoost	CatBoost	LightGBM	MLP	RF	GAM	Averaging	Stacking	SVR	KNN	CART
	Train dataset													
MAE	0	152.3	82.5	78.0	92.1	107.0	111.0	38.1	129.5	67.3	70.8	176.4	0	100.8
MAPE	0	0.168	0.094	0.090	0.103	0.120	0.126	0.042	0.143	0.076	0.080	0.178	0	0.109
COD	0	0.166	0.092	0.088	0.101	0.116	0.120	0.040	0.140	0.074	0,.078	0.195	0	0.110
RMSE	1.6	224.5	113.6	107.9	131.2	153.4	160.0	58.9	192.5	95.1	99.7	295.1	1.5	159.2
R^2	1	0.625	0.904	0.913	0.872	0.825	0.803	0.974	0.724	0.933	0.926	0.351	1	0.811
mmMAPE	0	0.188	0.101	0.096	0.111	0.130	0.135	0.043	0.158	0.080	0.084	0.217	0	0.121
mmPER	0	0.612	0.385	0.361	0.421	0.483	0.498	0.083	0.558	0.284	0.309	0.629	0	0.439
IQRat	0	0.128	0.073	0.069	0.079	0.090	0.091	0.030	0.110	0.058	0.061	0.136	0	0.079
TT	17:03:34	00:06:00	00:20:20	00:16:39	00:14:37	00:10:29	00:36:35	45:06:33	00:02:46	62:29:42	62:33:29	01:58:40	00:01:49	01:14:01
	Test dataset													
MAE	104.4	153.1	105.0	104.0	102.6	107.2	119.6	104.4	132.0	100.3	**99.9**	178.2	150.4	120.4
MAPE	0.113	0.168	0.114	0.113	0.112	0.117	0.130	0.113	0.145	0.109	**0.108**	0.180	0.162	0.127
COD	0.110	0.167	0.112	0.111	0.110	0.115	0.127	0.110	0.142	**0.107**	**0.107**	0.196	0.164	0.129
RMSE	163.7	227.0	161.1	159.3	157.3	163.9	184.0	164.1	200.1	155.1	**154.3**	296.1	241.3	190.0
R^2	0.800	0.614	0.806	0.810	0.815	0.799	0.747	0.798	0.700	0.820	**0.822**	0.344	0.564	0.730
mmMAPE	0.123	0.189	0.124	0.123	0.121	0.127	0.142	0.123	0.161	**0.118**	**0.118**	0.219	0.186	0.143
mmPER	0.442	0.615	0.454	0.451	0.448	0.466	0.509	0.444	0.556	0.431	**0.430**	0.630	0.573	0.509
IQRat	0.082	0.128	0.085	0.084	0.084	0.087	0.096	0.082	0.110	**0.080**	**0.080**	0.138	0.116	0.097

With TT = Training Time in hours, minutes and seconds

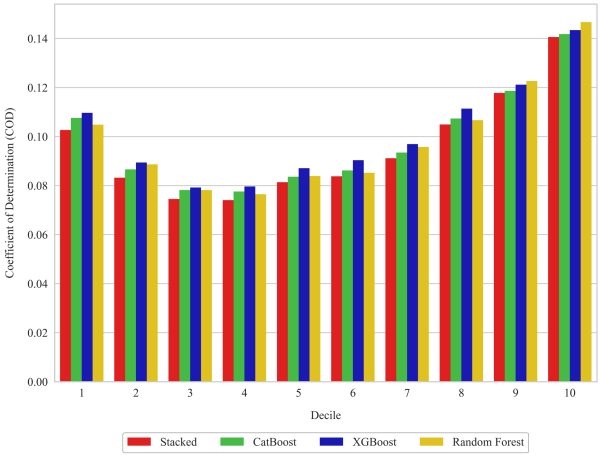
Fig. 1 Evaluation metrics on the test set per decile for selected models



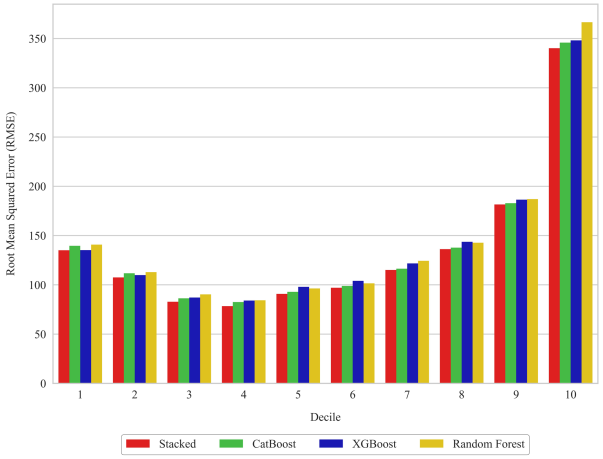
(a)



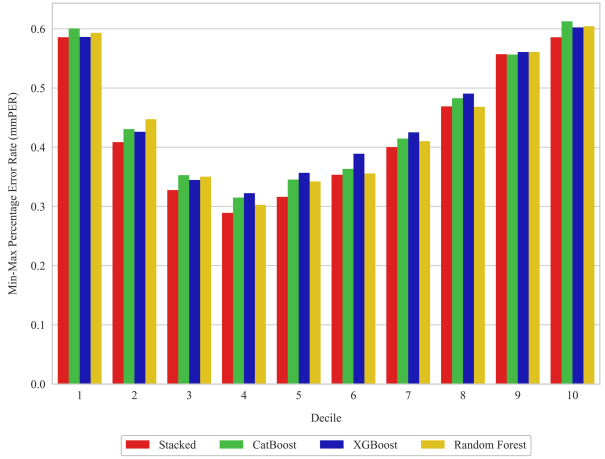
(b)



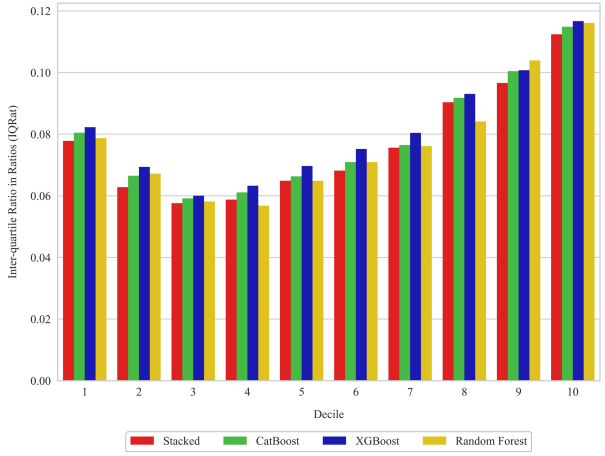
(c)



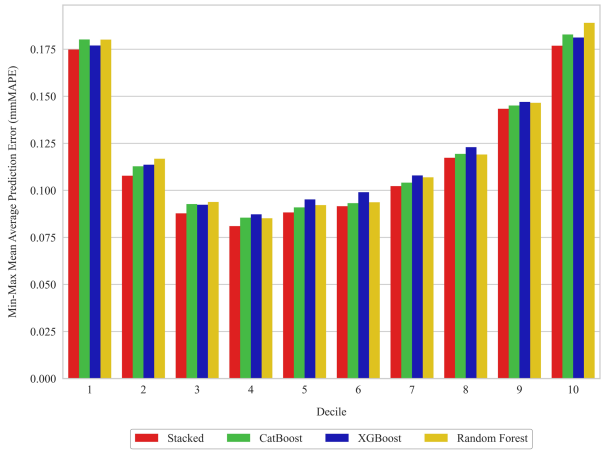
(d)


Min-Max Percentage Error Rate (mmPER)
0.0
0.1
0.2
0.3
0.4
0.5
0.6
1
2
3
4
5
6
7
8
9
10
Decile
Stacked
CatBoost
XGBoost
Random Forest

(e)


Inter-quartile Ratio in Ratios (IQRat)
0.00
0.02
0.04
0.06
0.08
0.10
0.12
1
2
3
4
5
6
7
8
9
10
Decile
Stacked
CatBoost
XGBoost
Random Forest

(f)


Min-Max Mean Average Prediction Error (mmMAPE)
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
1
2
3
4
5
6
7
8
9
10
Decile
Stacked
CatBoost
XGBoost
Random Forest

(g)

Furthermore, an important remark is to be made for the quality of the predictions per decile, indeed it was deduced from the obtained results that the predictions are very good for the observations in the central deciles – i.e. with mean rent prices –, but when looking at the low or high deciles – the tails – it is noticeable that the errors on the predictions are worse. Indeed, the differences between the best and worst deciles differ by a factor of 2 for MAPE, COD, mmPER, mmMAPE, and IQRat and differ by a factor of 5 for RMSE and MAE. So, this whole, that the metrics are better for observations in the middle-range deciles, will have an influence that must be considered during the modeling of rent prediction. Thus, it will be possible to better indicate how accurate the predictions are and provide a better impression of rents. Practitioners should consider this variability and possibly apply different models or additional preprocessing steps to improve overall prediction accuracy. Linked to this, it may be interesting to provide a price range or confidence bounds when predicting real estate prices so that variability can be captured and conveyed.

However, there are also limitations to this research. For example, tree-based models are also good at training on data with missing values. This study did not take advantage of that because the missing data was imputed. Another limitation is data preprocessing, for which no general framework is yet available for real estate research and applications. In addition, there is the possibility and thus limitation that some of the models in this study overfit, given the differences that are noticed for some models in the metrics between the train and test set. Thus, this would not lead to good generalizability to unseen data. There is also a limitation of the data, for example, we are working with the advertised rent prices and not the actual rent prices. In addition, the data is also data that is fed into websites by real estate agents and individuals. Therefore, it is not immune to input errors. Although a data cleaning was held beforehand, it is always possible that some errors slipped through the net.

Considering potential future research, there are many avenues, a few of which we highlight here. First, the setup should be applied to other datasets to validate the results. After all, there is the possibility that real estate data from, for example, properties for sale or other regions, have different characteristics and therefore obtain different findings. Further, although the tree-based models are black-box, follow-up research may involve deriving the relationships, evolutions, and trends between property prices and their determinants from tree-based models using XAI techniques. After all, interpreting the tree-based models is relatively quick to perform via SHAP via TreeSHAP, for example,

compared to an averaging or stacking ensemble. The interesting characteristic is that the tree-based models allow nonlinear and interactive relationships, which is not the case in classical econometric research based on linear regression. Optionally, after interpreting the tree-based models with XAI techniques, it is also feasible to derive the specification of a classical LR model. Finally, there is the opportunity for the study of other, unapplied algorithms. These could include existing neural network algorithms such as TabNet or customized neural networks for predicting real estate prices.

Declaration of Interest

The authors have no relevant financial or non-financial interests to disclose.

Data Availability Statement

The data that has been used is confidential and was provided by Realo N.V. which is located at Poel 16, 9000 Ghent, Belgium.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

CReDiT Author Statement

Ian Lenaers: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – Original Draft, Writing – Review & Editing

Lieven De Moor: Resources, Supervision, Writing – Original Draft, Writing – Review & Editing

References

Ali, M. (2023). *Pycaret: An open source, low-code machine learning library in python* (Version 3.0.0) [Computer software]. https://www.pycaret.org

Alkan, T., Dokuz, Y., Ecemiş, A., Bozdağ, A., & Durduran, S. (2022). *Using Machine Learning Algorithms for Predicting Real Estate Values in Tourism Centers*. https://doi.org/10.21203/rs.3.rs-1757533/v1

Antipov, E. A., & Pokryshevskaya, E. B. (2012). Mass appraisal of residential apartments: An application of Random Forest for valuation and a CART-based approach for model diagnostics. *Expert Systems with Applications*, *39*(2), 1772–1778. https://doi.org/10.1016/j.eswa.2011.08.077

Baur, K., Rosenfelder, M., & Lutz, B. (2023). Automated real estate valuation with machine learning models using property descriptions. *Expert Systems with Applications*, *213*, 119147. https://doi.org/10.1016/j.eswa.2022.119147

Bilgilioğlu, S. S., & Yılmaz, H. M. (2023). Comparison of different machine learning models for mass appraisal of real estate. *Survey Review*, *55*(388), 32–43. https://doi.org/10.1080/00396265.2021.1996799

Birkeland, K. B., D'Silva, A. D., Füss, R., & Oust, A. (2021). The Predictability of House Prices: "Human Against Machine." *International Real Estate Review*, *24*(2), 139–183.

Chou, J.-S., Fleshman, D.-B., & Truong, D.-N. (2022). Comparison of machine learning models to provide preliminary forecasts of real estate prices. *Journal of Housing and the Built Environment*, *37*(4), 2079–2114. https://doi.org/10.1007/s10901-022-09937-1

Choy, L. H. T., & Ho, W. K. O. (2023). The Use of Machine Learning in Real Estate Research. *Land*, *12*(4), Article 4. https://doi.org/10.3390/land12040740

Çılgın, C., Gökçen, H., Çılgın, C., & Gökçen, H. (2023). Machine learning methods for prediction real estate sales prices in Turkey. *Revista de La Construcción*, *22*(1), 163–177. https://doi.org/10.7764/rdlc.22.1.163

ElFayoumi, K., Salas, J., & Tudyka, A. (2021). Affordable Rental Housing: Making It Part of Europe's Recovery. *Departmental Papers*, *2021*(013). https://doi.org/10.5089/9781513570204.087.A001

Fedorov, N., & Petrichenko, Y. (2020). *Gradient Boosting–Based Machine Learning Methods in Real Estate Market Forecasting*. 203–208. https://doi.org/10.2991/aisr.k.201029.039

Fourkiotis, K. P., & Tsadiras, A. (2023). Comparing Machine Learning Techniques for House Price Prediction. In I. Maglogiannis, L. Iliadis, J. MacIntyre, & M. Dominguez (Eds.), *Artificial Intelligence Applications and Innovations* (pp. 292–303). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-34107-6_23

Gao, Q., Shi, V., Pettit, C., & Han, H. (2022). Property valuation using machine learning algorithms on statistical areas in Greater Sydney, Australia. *Land Use Policy*, *123*, 106409. https://doi.org/10.1016/j.landusepol.2022.106409

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer.

Hinrichs, N., Kolbe, J., & Werwatz, A. (2021). Using shrinkage for data-driven automated valuation model specification – a case study from Berlin. *Journal of Property Research*, *38*(2), 130–153. https://doi.org/10.1080/09599916.2021.1905690

Hjort, A., Pensar, J., Scheel, I., & Sommervoll, D. E. (2022). House price prediction with gradient boosted trees under different loss functions. *Journal of Property Research*, *39*(4), 338–364. https://doi.org/10.1080/09599916.2022.2070525

Ho, W. K. O., Tang, B.-S., & Wong, S. W. (2021). Predicting property prices with machine learning algorithms. *Journal of Property Research*, *38*(1), 48–70. https://doi.org/10.1080/09599916.2020.1832558

Iban, M. C. (2022). An explainable model for the mass appraisal of residences: The application of tree-based Machine Learning algorithms and interpretation of value determinants. *Habitat International*, *128*, 102660. https://doi.org/10.1016/j.habitatint.2022.102660

Kiely, T. J., & Bastian, N. D. (2020). The spatially conscious machine learning model. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *13*(1), 31–49. https://doi.org/10.1002/sam.11440

Krämer, B., Stang, M., Doskoč, V., Schäfers, W., & Friedrich, T. (2023). Automated valuation models: Improving model performance by choosing the optimal spatial training level. *Journal of Property Research*, *40*(4), 365–390. https://doi.org/10.1080/09599916.2023.2206823

Krämer, B., Stang, M., Nagl, C., & Schäfers, W. (2021). *Explainable AI in a Real Estate Context—Exploring the Determinants of Residential Real Estate Values* (SSRN Scholarly Paper 3989721). https://doi.org/10.2139/ssrn.3989721

Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer. https://doi.org/10.1007/978-1-4614-6849-3

Lenaers, I., Boudt, K., & De Moor, L. (2023). Predictability of Belgian residential real estate rents using tree-based ML models and IML techniques. *International Journal of Housing Markets and Analysis*, *ahead-of-print*(ahead-of-print). https://doi.org/10.1108/IJHMA-11-2022-0172

Lenaers, I., & De Moor, L. (2023). Exploring XAI techniques for enhancing model transparency and interpretability in real estate rent prediction: A comparative study. *Finance Research Letters*, *58*, 104306. https://doi.org/10.1016/j.frl.2023.104306

Lorenz, F., Willwersch, J., Cajias, M., & Fuerst, F. (2022). Interpretable machine learning for real estate market analysis. *Real Estate Economics*, *0*(0), 1–31. https://doi.org/10.1111/1540-6229.12397

Molnar, C. (2022). *Interpretable Machine Learning: A Guide For Making Black Box Models Explainable*. Independently published.

Mora-Garcia, R.-T., Cespedes-Lopez, M.-F., & Perez-Sanchez, V. R. (2022). Housing Price Prediction Using Machine Learning Algorithms in COVID-19 Times. *Land*, *11*(11), Article 11. https://doi.org/10.3390/land11112100

Pai, P.-F., & Wang, W.-C. (2020). Using Machine Learning Models and Actual Transaction Data for Predicting Real Estate Prices. *Applied Sciences*, *10*(17), Article 17. https://doi.org/10.3390/app10175832

Potrawa, T., & Tetereva, A. (2022). How much is the view from the window worth? Machine learning-driven hedonic pricing model of the real estate market. *Journal of Business Research*, *144*, 50–65. https://doi.org/10.1016/j.jbusres.2022.01.027

Rampini, L., & Re, C. F. (2021). Artificial intelligence algorithms to predict Italian real estate market prices. *Journal of Property Investment & Finance*, *40*(6), 588–611. https://doi.org/10.1108/JPIF-08-2021-0073

Sapakova, S., & Sapakov, A. (2024). Features of modeling the online real estate market in Almaty. *Procedia Computer Science*, *231*, 409–414. https://doi.org/10.1016/j.procs.2023.12.226

Sevgen, S. C., & Tanrivermiş, Y. (2024). Comparison of Machine Learning Algorithms for Mass Appraisal of Real Estate Data. *Real Estate Management and Valuation*, *0*(0). https://sciendo.com/article/10.2478/remav-2024-0019

Sharma, H., Harsora, H., & Ogunleye, B. (2024). An Optimal House Price Prediction Algorithm: XGBoost. *Analytics*, *3*(1), Article 1. https://doi.org/10.3390/analytics3010003

Stang, M., Krämer, B., Nagl, C., & Schäfers, W. (2022). From human business to machine learning—Methods for automating real estate appraisals and their

practical implications. *Zeitschrift Für Immobilienökonomie*. https://doi.org/10.1365/s41056-022-00063-1

Steurer, M., Hill, R. J., & Pfeifer, N. (2021). Metrics for evaluating the performance of machine learning based automated valuation models. *Journal of Property Research*, *38*(2), 99–129. https://doi.org/10.1080/09599916.2020.1858937

Talaga, M., Piwowarczyk, M., Kutrzyński, M., Lasota, T., Telec, Z., & Trawiński, B. (2019). Apartment Valuation Models for a Big City Using Selected Spatial Attributes. In N. T. Nguyen, R. Chbeir, E. Exposito, P. Aniorté, & B. Trawiński (Eds.), *Computational Collective Intelligence* (pp. 363–376). Springer International Publishing. https://doi.org/10.1007/978-3-030-28377-3_30

Tekin, M., & Uçal Sarı, İ. (2022). Real Estate Market Price Prediction Model of Istanbul. *Real Estate Management and Valuation*, *30*, 1–16. https://doi.org/10.2478/remav-2022-0025

Tekouabou, S. C. K., Gherghina, Ş. C., Kameni, E. D., Filali, Y., & Gartoumi, K. I. (2024). AI-Based on Machine Learning Methods for Urban Real Estate Prediction: A Systematic Survey. *Archives of Computational Methods in Engineering*, *31*(2), 1079–1095. https://doi.org/10.1007/s11831-023-10010-5

Trawiński, B., Lasota, T., Kempa, O., Telec, Z., & Kutrzyński, M. (2017). Comparison of Ensemble Learning Models with Expert Algorithms Designed for a Property Valuation System. In N. T. Nguyen, G. A. Papadopoulos, P. Jędrzejowicz, B. Trawiński, & G. Vossen (Eds.), *Computational Collective Intelligence* (pp. 317–327). Springer International Publishing. https://doi.org/10.1007/978-3-319-67074-4_31

Valier, A. (2020). Who performs better? AVMs vs hedonic models. *Journal of Property Investment & Finance*, *38*(3), 213–225. https://doi.org/10.1108/JPIF-12-2019-0157

Waddell, P., & Besharati-Zadeh, A. (2020). *A Comparison of Statistical and Machine Learning Algorithms for Predicting Rents in the San Francisco Bay Area* (arXiv:2011.14924). arXiv. https://doi.org/10.48550/arXiv.2011.14924

Xu, L., & Li, Z. (2021). A New Appraisal Model of Second-Hand Housing Prices in China's First-Tier Cities Based on Machine Learning Algorithms. *Computational Economics*, *57*(2), 617–637. https://doi.org/10.1007/s10614-020-09973-5

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295–316. https://doi.org/10.1016/j.neucom.2020.07.061

Yazdani, M. (2021). *Machine Learning, Deep Learning, and Hedonic Methods for Real Estate Price Prediction* (arXiv:2110.07151). arXiv. https://doi.org/10.48550/arXiv.2110.07151

Yilmazer, S., & Kocaman, S. (2020). A mass appraisal assessment study using machine learning based on multiple regression and random forest. *Land Use Policy*, *99*, 104889. https://doi.org/10.1016/j.landusepol.2020.104889

Yoshida, T., & Seya, H. (2021). *Spatial prediction of apartment rent using regression-based and machine learning-based approaches with a large dataset* (arXiv:2107.12539). arXiv. https://doi.org/10.48550/arXiv.2107.12539

Zhan, C., Liu, Y., Wu, Z., Zhao, M., & Chow, T. W. S. (2023). A hybrid machine learning framework for forecasting house price. *Expert Systems with Applications*, *233*, 120981. https://doi.org/10.1016/j.eswa.2023.120981

Zhou, X., Tong, W., & Li, D. (2019). Modeling Housing Rent in the Atlanta Metropolitan Area Using Textual Information and Deep Learning. *ISPRS*

International Journal of Geo-Information, *8*(8), Article 8. https://doi.org/10.3390/ijgi8080349

Zulkifley, N., Rahman, S., Nor Hasbiah, U., & Ibrahim, I. (2020). House Price Prediction using a Machine Learning Model: A Survey of Literature. *International Journal of Modern Education and Computer Science*, *12*, 46–54. https://doi.org/10.5815/ijmecs.2020.06.04