FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION OF HIGHER EDUCATION

NATIONAL RESEARCH UNIVERSITY

HIGHER SCHOOL OF ECONOMICS

Faculty of Economic Sciences

Lebedev Andrey Alexandrovich

«Forecasting Real Estate Prices with US Market Data Using Machine Learning Methods»

Thesis
Field of Study 38.04.08 Finance and Credit

Educational programme "Strategic Corporate Finance"

Reviewer
Head of Digital, BCS
Silaev Ilya Alexandrovich

Supervisor
Associate Professor.
Candidate of Sciences in Physics and Mathematics
Kurochkin Sergey Vladimirovich

Moscow 2024

Contents

Introduction

The emergence of modern financial crises has highlighted the need for more accurate forecasting models for risk management teams, to give them greater flexibility in responding to impending shocks. One potential way to achieve more accurate forecasts is to use more frequent data for forecasting purposes. This approach has been shown to be effective even in the case of less liquid assets, such as real estate. For example, it can be used to inform decisions about the optimal time to enter into a property transaction. A number of non-governmental organisations have created weekly indices for specific countries, such as the US - the second largest residential property market at USD 94.4 trillion in April 2024 with expected growth rate at 4.51% YoY (Statista, 2024) - based on the aggregation of market listings and transactions. Another approach is to use machine learning models, which have proven to be a reliable forecasting tool over the past 20 years, not only for static data but also for time series of different frequencies.

The main objective of this study is to determine the feasibility of employing datasets containing weekly residential real estate price indices to predict their values using machine learning techniques on a large (in terms of market size) statistical areas of the USA for the 1, 3 and 6 months in advance (4, 13 and 26 weeks accordingly). Furthermore, this research will assess the impact of macroeconomic variables on the behaviour of the market, which will be evaluated through the use of Shapley values.

Three tasks were set out to achieve this goal. The first was to ascertain which data could be used for the prediction of the real estate market in the US, both as an independent variable and as a target variable. This was based on the previous papers in this field and on the available sources of data and its frequency that can significantly affect the final quality of the models, which will serve as a data-driven part of the literature review. The first chapter of the paper is devoted to the resolution of this issue.

The second chapter of the paper is dedicated to the task of defining which machine learning methods can be applied for the forecasting of weekly time series and which methods can be used to prepare data for modelling based on previous research in finance, economics and computer science. These methods will be described in high-level detail and compared in order to identify the most suitable ones, which will also serve as a technical part of the literature review.

The final task, which will be discussed in the third chapter, is the estimation of the models on the available data, the comparison of the performance on the different prediction horizons and the understanding of how predictions depend on the macro data. Conclusions for the realisation of hypotheses will also be presented here.

The principal implications of the paper are as follows:

1. For scientific researchers, it can be demonstrated that weekly data may be employed for the prediction of the markets with lower levels of liquidity, such as those pertaining to real estate.
2. For analytics in the real estate sphere, the results demonstrate that weekly regional indices can be highly beneficial and worthy of calculation when real estate aggregators are available for specific countries.
3. For businesses, the findings provide significant assistance to businesses in the development of company risk management procedures and in the optimisation of decision-making processes for real estate deals.

It is crucial to highlight that the methodologies presented in this paper are only applicable to datasets with a sufficiently large time series, as this is essential for the results to be both sustainable and of high quality. For datasets that are smaller in terms of both frequency and time series length, it is recommended to opt for more traditional time series analysis techniques, as machine learning methods rely heavily on the volume of data available.

I. Dataset and Hypotheses

I.1. Real Estate Index and Macroeconomic Variables

Contemporary studies on real estate market modeling encompass a broad spectrum of approaches, utilizing a diverse array of tools to analyze data ranging from national-level to individual property specifics. Yet, we can split these tools based on groups of data sources suitable for either classification or/and regression tasks, based on the quality and complexity of the data:

1. Information about past property sales, including sale prices, property features (such as size, location, number of bedrooms/bathrooms, amenities), and sale dates that can be obtained from local private or governmental real estate aggregators (Kuşan et al., 2010; Park & Bae, 2015; Walthert & Sigrist, 2019; Ma et al., 2018; Chiu et al., 2021).
2. Information about location features such as neighbourhood, proximity to amenities (e.g., schools, parks, shopping centres), transportation infrastructure (e.g., highways, public transit), and local economic conditions from geoinformation systems (Park & Bae, 2015; Dimopoulos et al., 2018; Lee et al., 2022; Mubarak et al., 2022).
3. Information about the large districts or the whole regions that can be obtained from regional private or governmental real estate aggregators (Case & Shiller, 1990, 1993; Schindler, 2011; Alfaro-Navarro et al., 2020; Li & Chu, 2017; Chou et al., 2022).
4. Economic indicators such as interest rates, inflation rates, unemployment rates, GDP growth, and consumer confidence can influence real estate markets from private or governmental statistical aggregators (Park & Bae, 2015; Meharie et al., 2021).
5. Data on market trends, such as inventory levels, days on market, housing supply and demand dynamics, and sentiment indicators (e.g., consumer sentiment surveys, real estate market reports) that are sourced from private or governmental real estate aggregators (Wang et al., 2014; Hausler et al., 2018; Ma et al., 2018).
6. Demographic data such as population growth, household income levels, age distribution, and migration patterns are also considered in this analysis. Such data is sourced from statistical aggregators, including private or governmental sources (Reed (2016).

7. Information on seasonal and cyclical patterns is considered, with prices and sales activity varying throughout the year and over longer economic cycles, which are obtained from private or governmental statistical aggregators or economic researchers, such as Lee, et al. (2022).

The first two types of data provide the most accurate results when the objective is to obtain precise pricing in a limited geographical area with comprehensive data coverage. However, in the majority of cases, this data is not publicly available for reasons related to legal requirements or the fact that the data is owned by a company that charges a high price for it. This leads to the necessity of reducing the scale of the data and utilising aggregated data from the remaining types, which range from district-level to regional-level data, depending on the qualifications and budget of the aggregators. This paper focuses on the third, fourth, and partially on the seventh types.

The third type (regional data) is provided by the Haus Services, Inc. The Common Haus Price Index[1] (CHPI) represents the 100 largest statistical areas[2] in terms of market size. In comparison to the Case-Shiller index (CS), which covers only 20 of the largest cities in the country (Case & Shiller, 1990), data for each of the statistical areas is publicly available on a weekly basis. The complete list of areas with available dates and the number of entries used in this paper is presented in Table AI. The weekly data was selected over the monthly or quarterly data due to two reasons. Firstly, the growing liquidity of the real estate markets (van Dijk et al., 2020) necessitated the use of weekly data. Secondly, the need to test models on crisis data was identified, as the behaviour of the market changes during such periods (Salzman & Zwinkels, 2017). This section also incorporates seasonality, which plays an important role in certain states, such as Florida.

Regarding the fourth type, the Federal Reserve Bank of Saint Louis and Yahoo Finance provided economic and financial indicators, which could be divided into two categories. The first category comprised indicators that represented the overall state of the economy, including consumer and producer price indices, inflation, and so forth. The second category consisted of indicators related to real estate prices, such as construction materials indices and 30-year mortgage rates. All of the variables used in this research are listed in Table 1. It should be noted

[1] The index includes prices for a three-bedroom, two-bathroom, 1,500-square-foot (140 square meters) home built in 1977 on a quarter-acre (1012 square meters) lot, as documented by Haus. The original data for index construction was provided by the National Association of Realtors.

[2] A statistical area is a geographic region defined by the U.S. Office of Management and Budget for various demographic, economic, and social analyses. It is defined based on criteria such as population size, economic ties, and geographic contiguity. It may be a part of several states if located on the border.

that the primary data frequency for this paper is weekly. In order to fit the data into the dataset, part of the data had to be interpolated linearly. Additionally, we decided to examine whether our model could be applied to the forecasting of its main ancestor, the Case Shiller index, which also had to be interpolated. The model's ability to assess its suitability for data outside the specified range (2004 to 2012) and its generalisability to other similar indices without additional modifications represents a significant impact on the model evaluation.

Table 1 “Original variables. their sources and frequencies"

Name	**Name in the Models**	**Source**	**Ticker in Source**	**Frequency**
Common House Price Index	CHPI	Haus	CHPI	Weekly for each statistical area
Case-Shiller Index	CS	FRED	SPCS20RSA	Monthly
Federal Funds Effective Rate	KeyRate	FRED	DFF	Daily
Consumer Price Index	CPI	FRED	CPIAUCSL	Daily
Volatility Index by CBOE	VIX	FRED	VIXCLS	Daily
Producer Price Index	PPI	FRED	PCU44414441	Monthly
30-year Mortgage Rate	MortgageRate30	FRED	MORTGAGE30US	Weekly
Electricity Price Index	Electricity	FRED	CUSR0000SEHF01	Monthly
Water Price Index	Water	FRED	CUSR0000SEHG	Monthly
Plywood Price Index	Plywood	FRED	WPU083	Monthly
Steel Price Index	Steel	FRED	WPU101	Monthly
Glass Price Index	Glass	FRED	PCU3272132721	Monthly
Concrete Price Index	Concrete	FRED	PCU32733273	Monthly
Unemployment Rate	Unemployment	FRED	UNRATE	Monthly
10-Year Treasuries Yield	Yield10Y	FRED	DGS10	Daily
Dow Jones Index	DJI	Yahoo Finance	^DJI	Daily
S&P500	S&P500	Yahoo Finance	^GSPC	Daily

A correlation analysis was conducted on both original and generated variables in order to identify which were the most useful in the modelling process. Figure I represents the linear correlations in the main variables and shows that the most useful may be CPI, water, glass, concrete and financial market indices, while mortgage rates, which are directly connected to the task, have a very low correlation. This is probably due to the fact that these time series belong to different spaces. It is also important to note that, despite the high level of multicollinearity observed in some of the variables, both were included in the model. This is because modern, sophisticated machine learning algorithms are capable of working with such data and can separate the effects of these variables (Drobnič et al., 2020).

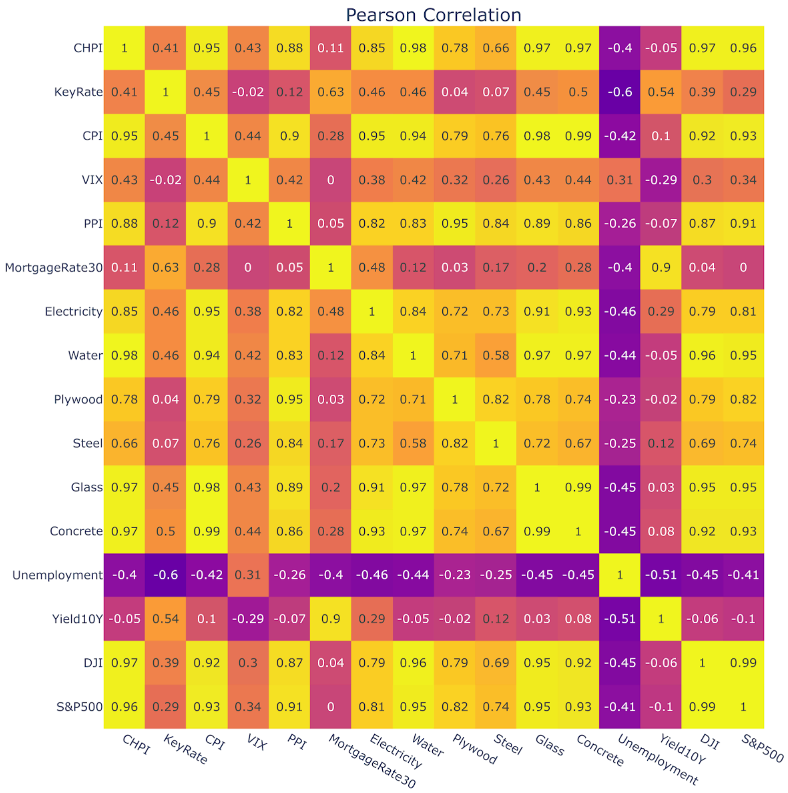

Figure 1 “Pearson (Linear) Correlation Matrix”

Figure II presents the results of the nonlinear correlation analysis, which largely corroborate the conclusions reached from the linear correlations. However, in most cases, strong correlations (as measured by absolute values) exhibited even stronger correlations, while weaker correlations demonstrated fluctuations in both directions, including instances where the correlation could have shifted from positive to negative or vice versa. Of particular interest is the notable change in the correlation between the target index and the steel price index, which declined from 0.66 to 0.32.

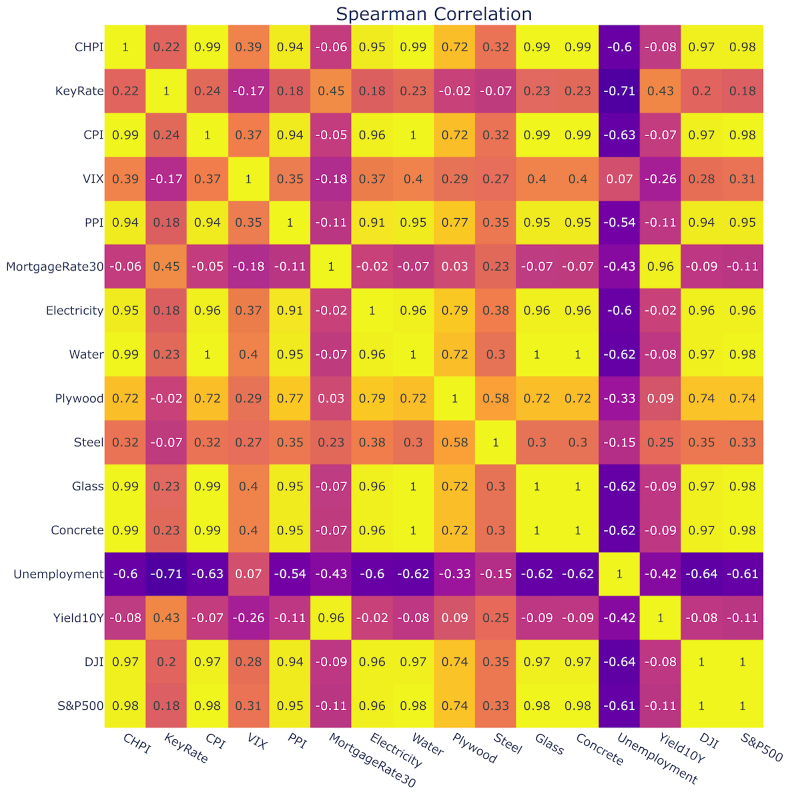

Figure 2 “Spearman (Nonlinear) Correlation Matrix”

In order to comprehend the fundamental characteristics of the time series employed in this study, it was deemed necessary to conduct a stationarity test utilising the Augmented Dickey-Fuller test (ADF) and to calculate the Hurst and largest Lyapunov exponents. The results of these calculations are presented in Table AII. The ADF test determines whether the time series exhibits a unit root or is stationary. As anticipated, the majority of the original time series were found to be non-stationary, whereas the logarithmic returns exhibited a p-value below 0.05, indicating that these time series were stationary. The Hurst exponent allows us to ascertain the degree of stationarity and noise present in a time series. A value below 0.5 indicates anti-persistence, 0.5 represents white noise, and values above 0.5 indicate an increasing level of persistence, which ultimately leads to the conclusion that the data is non-stationary after 1.0. Previously, they have been demonstrated to be a valuable tool for analysing financial data (Dmitriev et al., 2022). In our data, the Hurst exponent aligns with the outcomes of the ADF test. The least common of the three is the largest Lyapunov exponent (Rosenstein et al., 1993), yet it remains a valuable tool for discerning whether a high level of chaos exists in a time series, which could potentially lead to a decline in the performance of variables in models if the exponent exceeds 0.05. In our case, all of the time series remain below this threshold, indicating that they can be utilised in modelling.

The following limitations of the data should be noted:

1. The CHPI has only been available since the end of 2010.
2. Part of the time series had to be interpolated (linearly) from monthly to weekly to fit the model.
3. There is no publicly available data of the same volume specifically for each of the statistical areas. Consequently, we had to use data from the national level, which can be representative of most, but not all, of the regions.
4. During the first half of 2010th, several artefacts were observed in the CHPI indices for some of the statistical areas, including significant drops or jumps in the data (more than 15% WoW) that had to be filtered out to avoid giving the model false patterns.
5. Only 42 of the 50 states are included in the sample.
6. Several states account for a large proportion of the sample: The number of statistical areas included in the sample from the following states is as follows: Florida (11), California (9), Ohio (7), Pennsylvania, Texas and New York (6). This may be regarded as a limitation, as these states are the main drivers of real estate development in the country. However, in some cases, it may have a negative impact on the quality of the predictions for less powerful states.
7. Due to the much lower seasonality of CS its predictions might start to suffer during the transfer learning process.

I.2. Feature Generation and Scaling

In order to incorporate both static and dynamic variables, we generate them from the original variables (both independent and target) for the 4, 13 and 26 week horizons in the form of logarithmic returns for these periods in the past. In this case, the nominal values act as anchors for the price decision, while the logarithmic returns add inertia from past data, thereby increasing the ability of the models to track trends without the need to add any kind of dummy variable.

During the feature generation process, we also create final targets: future values of the indices for 4, 13 and 26 weeks. These are only used separately in the models in order to avoid the introduction of real future information into the training process. It should be noted that we initially had the idea of chain prediction (using model predictions to predict values for even more dates). However, this was ultimately rejected in order to minimise the potential for error in the final model.

Yet, when dealing with widely dispersed prices for real estate and construction resources, it becomes imperative to downscale them, simplifying computations for the model and potentially enhancing its overall quality (Ferreira et al., 2019), since in this case, the remaining regions with high prices will be unable to influence the model to a significant extent due to the presence of large errors. While this will still be a problem for the final models, it will be to a much lesser extent than in the case where scaling is not applied. For the final target variables, we employ the use of natural logarithms in order to circumvent the limitations imposed by price buckets, while simultaneously achieving scaling for the largest price categories. Limiting the target is a crucial aspect of machine learning models based on decision trees, as the basic implementation is unable to predict values outside the specified range. With regard to the final independent variables, we have opted for standardisation. This is a preprocessing technique employed to standardise the features by subtracting the mean and scaling them to variance. In this instance, standardisation was selected over normalisation, as the latter still maintains a wide distribution of the data, but has been scaled to the [0, 1] segment (Jo, 2019).

The full high-level logic of the described transformations applied to the dataset to improve its quality is presented in Figure 3.

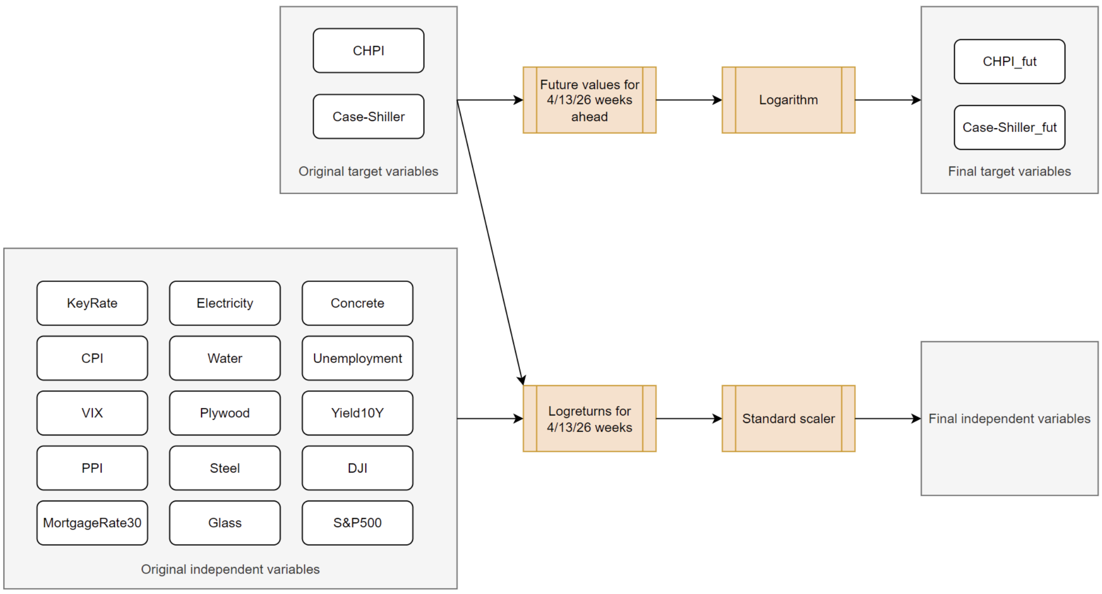

Figure 3 “Feature generation for target and independent variables”

I.3. Research Hypotheses

The data collection and feature generation process yielded a set of seven hypotheses, as follows:

1. Past prices would have the most determining effect on the model predictions, given the low degree of chaos and relatively high degree of persistence, as evidenced by the columns "Lyapunov Largest Exponent" and "Hurst Exponent" in Table AII.
2. Building material indices would be the second most valuable variables.
3. The predictive power of the independent variables would increase with the horizon, as the autoregressive effect of the target variable would decrease over time and be replaced by less powerful predictors.
4. The predictive power of logarithmic returns would be higher compared to static variables for the smaller horizons, as a result of the same autoregressive effect.
5. Statistical areas with lower prices would have better prediction errors than those with higher prices (mainly because there are only a few high-price areas in the sample).
6. The error of the models would increase significantly between the 4 and 13 week forecasts, but the difference between the 13 and 26 week forecasts would be less dramatic. This is because the decreasing autoregressive effect would be partly compensated by the other variables.
7. Stacking the gradient boosting models would allow us to outperform the OLS benchmark both on the original data and on the Case-Shiller index. This is because of the ability to capture more patterns in the data.

II. Methods

II.1. Prediction Models and Shapley Values

Regarding modern modelling techniques, there is a number of machine learning instruments that are employed in scientific research and other fields to predict high-frequency time series:

- Gradient boosting models (the most prominent implementations being LightGBM, XGBoost, and CatBoost) involve sequentially training a series of small models, which are in most cases decision trees. Each subsequent model corrects the errors made by the previous one. The iterative process has the objective of minimising the overall prediction error by combining the predictions of multiple weak learners in a weighted manner (Bentéjac et al., 2020). The method can be applied to any kind of data, including both time series and static data. In this paper, decision trees and random forests are not presented as separate solutions in order to avoid the repetition of methodology. However, they still can be employed in the simpler models for real estate price prediction (Kuşan et al., 2010; Dimopoulos et al., 2018).
- Recurrent neural networks (one of the most wide spread implementations – Long Short-Term Memory) are neural networks that capture information about past inputs. This enables RNNs to effectively model dependencies and patterns in sequences, making them particularly well-suited for tasks such as language modelling, time series prediction, and sequence generation (Sutskever et al., 2014; Salehinejad et al., 2018). However, the effectiveness of RNNs is heavily dependent on the quality of the data and shows best performance with residential data (Lee et al., 2022).
- Deep neural networks - they can process non-sequential data while still being useful for time series forecasting even for real estate markets (Li & Chu, 2017). The concept of deep neural networks involves leveraging multiple layers of interconnected neurons to learn hierarchical representations of data. Each layer in the network extracts increasingly abstract features from the input data, with deeper layers capturing more complex patterns (Canziani et al., 2017). However, it also requires high-quality data to prevent the creation of false logical connections.
- Support vector machines (SVM), which are predominantly employed for classification, can also be utilised in regressive tasks pertaining to real estate price forecasting (Wang et al., 2014; Chou et al., 2022). They are comparatively less reliant on the

quality of data in comparison with neural networks, yet still necessitate a meticulously prepared dataset to generate a viable prediction for time series.

Classification and clustering algorithms are not directly applicable to the solution of the problem under investigation in this paper (Mohd et al., 2020). However, they can assist in the segmentation of the sample, enabling the creation of separate models for significantly different groups (Baldominos et al., 2018; Ma et al., 2018). The use of these techniques is not employed in this paper, as preliminary testing of models with group labels indicated that they yielded inferior results on the validation and test samples by 5-10% on average. However, it should be noted that there may be a grouping method or combination of methods that could potentially yield superior performance, which may be identified during the continuation of this paper's research.

Due to the limitations of the available data, which represents the regional level and includes only national macroeconomic variables, it is not possible to use neural networks and SVM to predict future values. This is because the target variables are regionally dependent and, in fact, only part of the information from the higher levels can be effectively used on the lower ones. However, RNN and other similar models continue to search for deep connections, which can be erroneous. The time-spatial characteristics of the dataset present a significant challenge for RNN and its derivatives. One solution is to employ a multiheaded model (Loh et al., 2022), which, however, cannot be directly applied in this case due to the inability of the model to work with the new data that was not previously used in training. In contrast, gradient boosting, with a well-chosen set of hyperparameters[3] (Probst et al., 2019; Yang & Shami, 2020) that prevent severe overfitting[4] (Ying, 2019), seeks to identify higher-level connections. These connections are less susceptible to overfitting, but nevertheless present a challenge, as will be demonstrated in the results section.

The Root-Mean-Squared Error (RMSE) as calculated in (1) was used as the scoring metric for all models utilized in this study. The other option was the Mean Absolute Error (MAE), which is shown in (2), but it wasn’t selected as main for two distinct rationales:

1. Real estate prices vary significantly across different statistical areas, which could result in increased errors within the model.

[3] In machine learning, hyperparameters are parameters that are set prior to the training process and control aspects of the learning process itself, rather than being learned from the data. These parameters can influence the behaviour and performance of the model, but are not directly learned from the training data. Hyperparameter instances can be represented by the learning rate within gradient descent optimization algorithms and the quantity of hidden layers within a neural network, the depth of a decision tree, and the regularisation strength in linear models.

[4] Overfitting in machine learning occurs when a model learns to capture noise and random fluctuations in the training data rather than the underlying pattern or relationship. This results in a model that performs well on the training data but fails to generalise to unseen data or new examples.

2. Secondly, the RMSE is more punitive towards models with greater error, thereby encouraging the construction of more accurate predictions (Hodson, 2022).
3. The root mean square error (RMSE) is optimal for the models with normally distributed errors (Chai & Draxler, 2014). This allows the construction of more robust models that can be used for more predictable transfer learning, given that the errors are less variable.

$$RMSE(y, \hat{y}) = \sqrt{\frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} (y_i - \hat{y}_i)^2} \quad (1)$$

$$MAE(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} y_i - \hat{y}_i \quad (2)$$

In order to additionally evaluate percentage errors, we are utilising the Mean Absolute Percentage Error (MAPE - Formula 3) metric, which allows us to understand the size of the errors relative to the original data.

$$MAPE(y, \hat{y}) = \frac{100\%}{n_{samples}} \sum_{i=0}^{n_{samples}-1} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (3)$$

Gradient boosting implementations that are used in this paper:

1. LightGBM, created by Microsoft, employs a gradient-based one-side sampling method, which reduces the number of data instances used for building each tree. Additionally, it utilises a leaf-wise growth strategy for building trees, whereby the tree node-wise is grown instead of level-wise (Ke et al., 2017).
2. XGBoost is an open-source (publicly managed) model that minimises the error of the loss function approximated using Taylor series. It incorporates L1[5] and L2[6] regularisation terms into its objective function to control model complexity and prevent overfitting (Ng, 2004). Furthermore, it employs a technique called tree pruning[7] (Zhu & Gupta, 2017) to regulate the depth of individual trees during training (Chen & Guestrin, 2016).

[5] L1 (Lasso) regularisation introduces a penalty term to the loss function proportional to the absolute values of the model's coefficients.
[6] L2 (Ridge) regularisation introduces a penalty term to the loss function, which is proportional to the squared magnitudes of the model's coefficients.
[7] The pruning of trees is a process whereby certain parts of the tree are removed that are deemed unnecessary or redundant. This prevents the trees from growing too deep and helps to reduce overfitting while maintaining predictive accuracy.

3. CatBoost, created by Yandex, employs a range of optimisation techniques to handle numerical features in an efficient manner. These include ordered boosting, which sorts numerical features during training in order to reduce memory usage and improve training speed. It also utilises symmetric trees and incorporates built-in handling of categorical variables (Ostroumova Prokhorenkova et al., 2017).

The hyperparameter optimisation process utilises Optuna (Akiba et al., 2019), which serves to unify the training process and enhance the quality of the validation and test results. Optuna's architecture enables researchers to dynamically shape the search space. Its define-by-run API alleviates the necessity to meticulously predetermine every aspect of the optimisation strategy in advance. Optuna conceptualises hyperparameter optimisation as a task to minimise or maximise an objective function, which accepts a set of hyperparameters and yields a validation score. Following its publication, this instrument continued to be one of the most useful and effective instruments for the choice of hyperparameters.

The algorithm of the optimization in Optuna operates as follows:

1. A set of optimized hyperparameters is defined
2. At each iteration search, a space for hyperparameters is constructed (including the results from the previous iterations) and evaluated on the model
3. The algorithm continues optimization search and pruning until the target of the iteration (min or max[8]) is reached
4. After the end of the optimization process the best set of the hyperparameters is returned.

Contemporary machine learning models are not sufficiently robust when employed in isolation. Consequently, they are reliant on the quality of the interpretation to be effective in both academic and business contexts. To address this issue, we are utilising Shapley values. In explainable machine learning, Shapley values are calculated to evaluate the impact of input features on the output of a machine learning model on a per-instance basis (Benedek Rózemberczki et al., 2022). With a specific data point under consideration, the objective is to deconstruct the model's prediction and apportion Shapley values to each feature of that instance. These Shapley values represent explanatory attributions to the input features, and in the event of missing input feature values, they are substituted with a reference value, such as the mean derived from numerous instances (which is not the case in this research due to the preliminary data filtering). The implementation utilized in this paper is SHAP (Lundberg & Lee, 2017),

[8] In the case of models with RMSE, the objective is to minimise the error. In contrast, the majority of classification metrics employ maximisation.

which derives Shapley values through linear approximation.

II.2. Final Model Architecture and Prediction Quality Enhancement

In order to enhance the final quality of the model, several techniques are employed. The initial approach is stacking, a highly effective strategy for addressing classification and regression problems in machine learning. This involves utilising the predictions generated by models at a preceding level as input features for models at subsequent levels (Pavlyshenko, 2018). In this paper, we are utilising only one level of stacking, as there are three models that produce predictions from a single space, which can be efficiently stacked without overfitting and may be employed in construction price forecasting (Alfaro-Navarro et al., 2020; Meharie et al., 2021). The stacking process is facilitated by the use of OLS (with constant), which parameters are additionally filtered based on their p-value. In this specific paper, if the model parameter has a p-value higher than 0.05 in the stacking process, it is excluded from the stacking model and the stacking model is recalculated to identify new variables that can be excluded. This process is employed to ensure that only high-quality predictions are included in the final model. The model's architecture is depicted in Figure 4, which illustrates the three stages of the modelling process: data selection and hyperparameters optimisation, base gradient boosting models, and a stacking model.

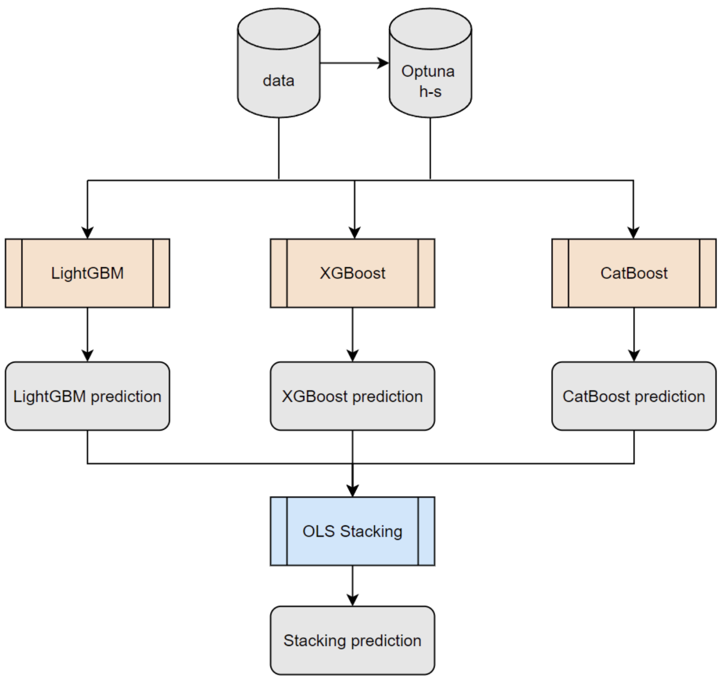

Figure 4 “Architecture of the stacking model”

In the initial stage, data filtration, as described in Chapter 2, is conducted, and gradient boosting hyperparameters (learning rate, number of iterations, early stopping, size and number of leaves, regularisations, etc.[9]) are calculated for all three base models on the training data. This is then validated on the validation data, with the test data not being used at this stage.

During the second stage, models undergo training using optimal hyperparameters, and predictions are generated for both the validation and test sets to evaluate the risk of overfitting in the base models. This evaluation involves comparing the error rates observed on the validation and test sets.

In the third stage, models are assembled following the methodology described earlier. The OLS model is trained on the validation set, creating predictions that are then evaluated on the test set. Here we obtain a finalized model prepared for application to other indices with comparable traits, such as the S&P Case-Shiller index.

An idea of using models on datasets other than those for which they were trained is called transfer learning (Zhuang et al., 2021). During its implementation we may encounter a number of challenges:

1. Differing dimensionality of the data sets. For instance, a model may have been trained on data from multiple regions, and then applied to a high-level index. This can be mitigated by some form of averaging or smoothing.
2. Differing frequency of the data. For example, a model may have been trained on weekly data, while the new data is monthly data, interpolation can be employed to address this issue to a certain extent.
3. Different time periods, such as when the left time boundary of the available data is earlier in comparison to the original data, it is not straightforward how to ascertain the impact of this discrepancy on the accuracy of the model. This is because the market behaviour patterns evolve over time, and therefore, the error growth cannot be easily quantified.

The second instrument used to enhance the quality of the prediction is the application of empirical mode decomposition (EMD) (Rios & Mello, 2016). This technique decomposes a timer series (signal) into a finite set of intrinsic mode functions (IMFs), which represent oscillatory modes of different frequencies present in the original signal. The EMD process comprises four stages:

1. Identify the local extrema of the signal. These extrema can be conceptualised as the "envelopes" that encapsulate the oscillatory components of different scales in

[9] The random state (seed) for the random number generator is not included in this list because it is fixed at 2024 for the entirety of the research project. This is done to avoid confusion among readers. However, it should be noted that this also affects the final results and can be used as one of the tuned hyperparameters in the model.

the signal.

2. A mean envelope is created by interpolating between the maxima and minima at each point in the signal. This mean envelope serves as a reference for extracting oscillatory modes.
3. The IMF is then obtained, which is the difference between the original signal and the mean envelope. This IMF represents the finest scale oscillatory mode present in the signal. If the IMF is monotonic, the subsequent step is to be taken; otherwise, the first three steps must be repeated.
4. The IMFs are to be separated into deterministic and stochastic components based on the mutual information criteria. The higher the criteria, the fewer IMFs are included in the deterministic component.

In this paper, EMD is employed to address the initial issue connected to the transfer learning (differing dimensionality). This approach enables the removal of noise, which was introduced by the fact that the original model was trained on 100 separate statistical areas. Consequently, there may be instances where erroneous patterns were learned for specific regions that were not included in the transfer dataset or did not have a significant impact on the Case-Shiller index, given that it encompasses only the 20 largest cities in the United States.

In order to ascertain the utility of the final stacking model, we also estimated a benchmark OLS for all time horizons and both the original HAUS data and the Case-Shiller index. This benchmark will demonstrate the extent to which machine learning models perform and the rate at which their error deteriorates over time in comparison with this benchmark.

III. Results and Conclusions

Following the modelling process, we have obtained a significant set of results, which are presented in Table 2. This table not only displays the outcomes of individual gradient boosting models and their integration with CHPI and CS data, but also a benchmark OLS on the same datasets. This was conducted to ascertain the efficiency of these models not only in the context of regional real-estate price prediction. For all time horizons, LightGBM and XGBoost demonstrated comparable performance, exhibiting higher levels of performance than the benchmark. However, CatBoost did not meet the required quality standards, even in the absence of severe overfitting. This issue led to the exclusion of CatBoost from the final model during the feature selection process for stacking, as outlined in subsection II.2. The primary reason for this was that this implementation was originally designed for classification tasks, rather than regression ones. Nevertheless, the final stacked model exhibited an error rate that was 4-5% lower than that of the separate gradient boosting models. In all three cases, both models that were retained in the final model received at least 29% impact on the final prediction. Furthermore, the quality of the stacked model exhibited a slower decline in performance with an increase in the prediction horizon than that of the OLS benchmark. The OLS benchmark began at a level of 15% worse at 4 weeks of prediction and ended at 35% worse at 26 weeks. The coefficients, standard errors and p-values of the stacked models can be seen in the table BI.

A comparable picture emerges when considering the test on the CS data. Here, the benchmark demonstrated a high level of performance only at four weeks after transferring from the original dataset. In contrast, performance on the other horizons decreased significantly. Furthermore, the stacked model for 13 and 26 weeks outperformed the benchmark even without the implementation of smoothing, which was introduced due to the considerable seasonality observed in the CHPI for certain statistical areas, as previously discussed in subsection II.2. The improvement in accuracy resulting from the smoothing process became increasingly pronounced as the prediction horizon increased, as the separate models transitioned from an averaging approach to a more cyclical one. The issue of different seasonalities represents a significant challenge for transfer learning between datasets with different aggregation approaches and requires further investigation beyond the smoothing instruments, especially for the training datasets. Additional seasonality variables, which could be considered the most obvious instrument for this case, may not be as effective, but they can be employed in future studies with machine learning models. Nevertheless, the results of this study demonstrate the continued utility of these techniques.

Table 2 "RMSE and MAPE (%) for models on the HAUS and Case-Shiller data"

Models	4 weeks prediction		13 weeks prediction		26 weeks prediction	
	Validation RMSE / MAPE (%)	**Test RMSE / MAPE (%)**	**Validation RMSE / MAPE (%)**	**Test RMSE / MAPE (%)**	**Validation RMSE / MAPE (%)**	**Test RMSE / MAPE (%)**
LightGBM	5 697.991 / 0.097	5 815.907 / 0.098	8 416.799 / 0.156	8 479.791 / 0.153	10 771.676 / 0.193	10 495.897 / 0.189
XGBoost	5 544.043 / 0.093	5 574.208 / 0.095	7 803.822 / 0.147	7 801.324 / 0.146	10 271.295 / 0.197	10 720.080 / 0.199
CatBoost	8 270.735 / 0.136	8 763.188 / 0.138	10 782.573 / 0.185	11 283.988 / 0.187	13 566.801 / 0.240	13 183.759 / 0.236
Stacked model	**5 265.810 / 0.091**	**5 274.117 / 0.093**	**7 633.267 / 0.145**	**7 517.742 / 0.143**	**9 968.613 / 0.187**	**10 041.655 / 0.185**
Benchmark (OLS)	6 176.287 / 1.241	6 407.752 / 1.273	11 630.265 / 2.425	12 375.693 / 2.448	15 375.112 / 3.355	16 404.234 / 3.417
Stacked on Case-Shiller	2 242.050 / 0.856		7 182.020 / 2.607		8 952.761 / 3.447	
Smoothed stacked on Case-Shiller	**2 009.940 / 0.760**		**6 044.992 / 2.065**		**6 947.070 / 2.723**	
Benchmark (OLS) on Case-Shiller	1 820.272 / 0.711		8 275.517 / 3.331		27 636.263 / 11.767	

The cyclicality issue is evident in Figure 5, which depicts the period of stable market growth from 2014 to 2020. During this period, CHPI prices exhibited a notable increase during the summer months, while CS prices exhibited a markedly more consistent trend. However, the model exhibited a tendency to respond slowly during periods of economic crisis. It should be noted that the threshold for the separation between stochastic and deterministic components for the smoothed prediction was chosen automatically based on the optimal level of error for each of the time horizons. This is the reason why it can be observed that for all of the cases, different smoothing leads to the prediction artifacts that are not replicated for other time series.

A 13-week forecast from Figure 6 reveals that the predictions deviate from the true values during the 2008 crisis and 2022 slowdown. Concurrently, an increase in cyclicality overflow between datasets is observed, as previously described. However, the main trend is consistently followed, and the model can be used for predicting the CS index. With regard to the 26-week prediction (Figure 7), it is evident that the majority of the errors in the predictions are attributable to the overall lagging of the model on the turning points. This phenomenon has been quantified in Table BII, which presents a detailed analysis of stacked models across different time periods (before 2010-01-01, between 2010-01-01 and 2018-01-01, and after 2018-01-01).

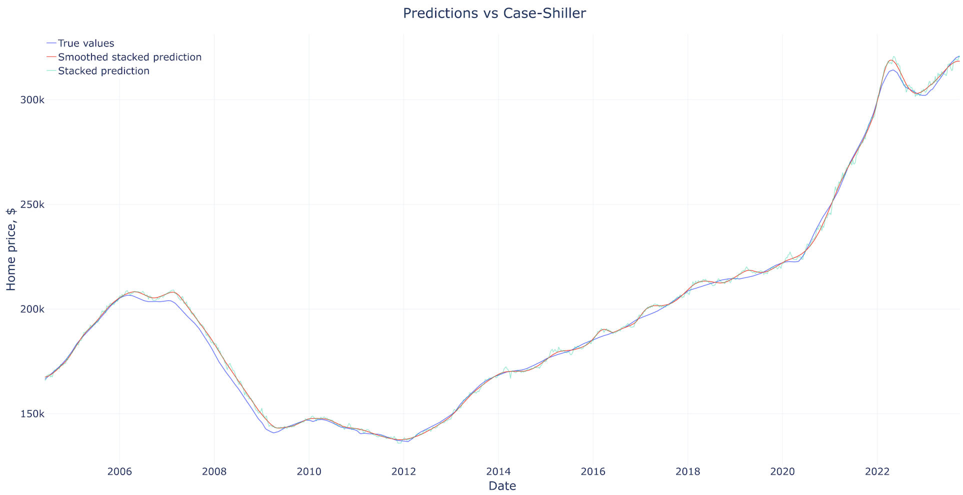

Figure 5 “Predictions vs Case Shiller for 4 weeks”

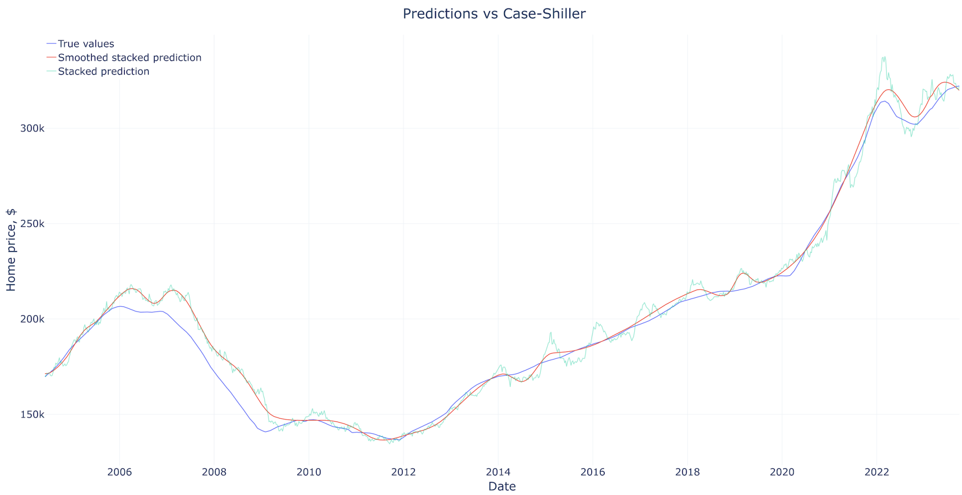

Figure 6 “Predictions vs Case Shiller for 13 weeks”

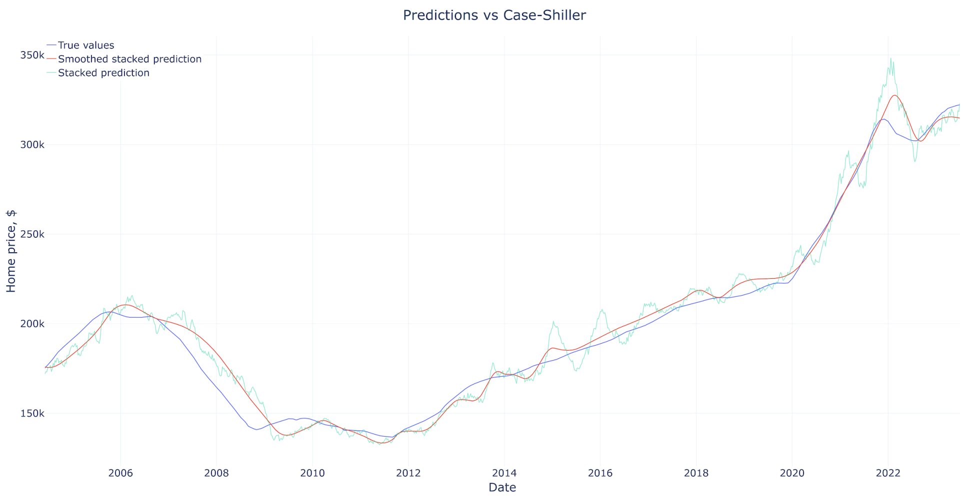

Figure 7 “Predictions vs Case Shiller for 26 weeks”

However, upon returning to the original data, it becomes evident that the primary limitation of the models is their inability to accurately predict patterns that result in the occurrence of significant errors, which can be observed in Figures B7-B18 and Tables BIII-BVIII. The figures illustrate the distributions of predictions against true values and their errors. They demonstrate that all of the gradient boosting approaches are capable of working even with the long tails of distributions of the target data when tuned correctly. Tables divide the same data into 10 buckets for each sample and horizon. They demonstrate that the models perform well with prices below 700-800k in statistical areas, while for the remaining buckets, we observe an increase in errors, which, in percentage terms, is still comparable with the lower buckets.

The Shapley values for XGBoost, which became the primary predictor in the stacked model, are presented in Figures 3-5. For LightGBM and CatBoost, these values can be observed in Figures B1-B6. The horizontal axis of the graphs represents the degree of impact of the impact variables on the prediction. The colours of the variables themselves are used to represent their values. These plots convey three key insights regarding the influence of different variables:

1. The models for all of the shown horizons are autoregressive and depend on the previous values of the time series to a greater extent than other variables.
2. For smaller horizons, logarithmic returns have a greater impact on the final prediction, while on the 26 weeks, static variables show a higher level of importance.
3. The most significant variables in the model, with the exception of the previous state of the system and its dynamics, were concrete, plywood, CPI and PPI. Other macroeconomic factors were less useful in this particular model structure.

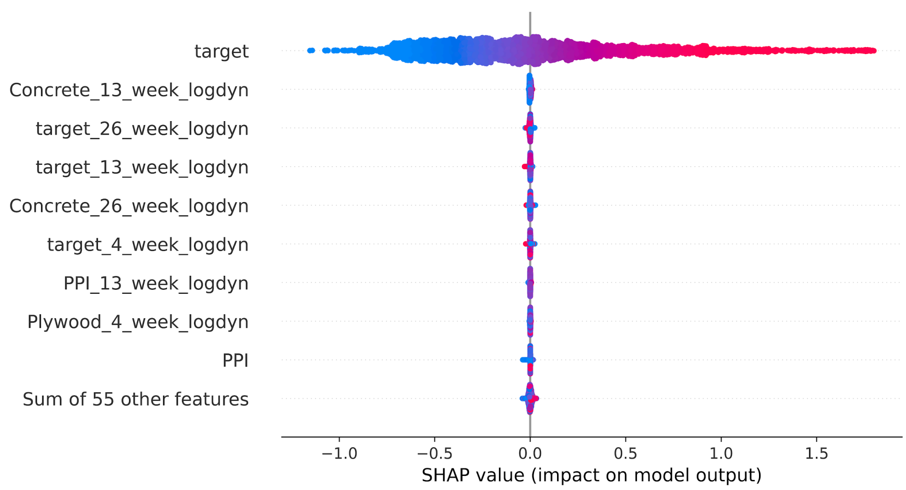

Figure 3 “Shapley values for XGBoost with 4 weeks prediction”

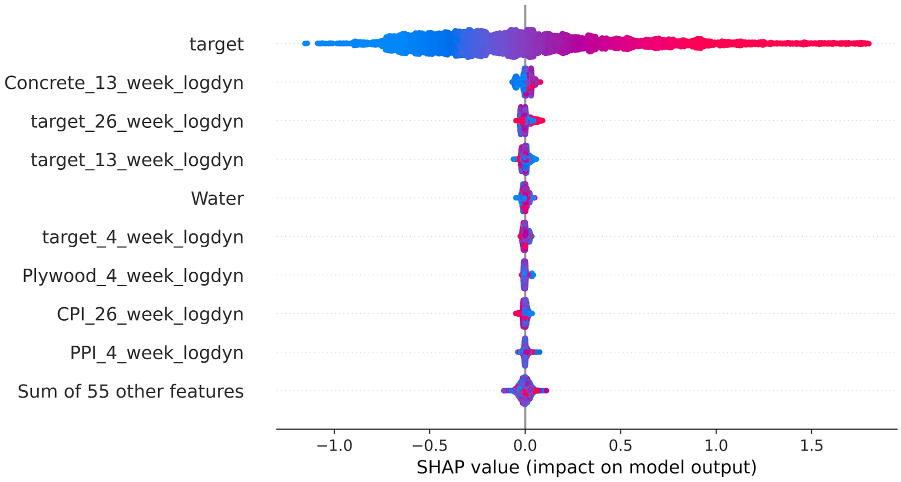

Figure 4 “Shapley values for XGBoost with 13 weeks prediction”

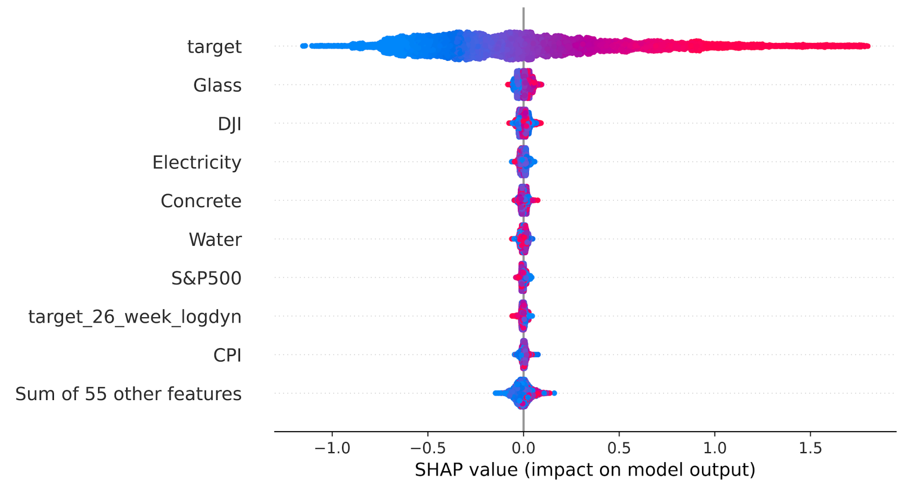

Figure 5 “Shapley values for XGBoost with 26 weeks prediction”

Conclusions for the research hypotheses:

1. Past prices would have the most determining effect on the model predictions - This is correct, as these prices have the defining impact on the predictions and their quality.
2. Building material indices would be the second most descriptive - This is partially correct, as they have shown much larger impact for all of the horizons and models in comparison with pure macroeconomic variables but market indices managed to get relatively high Shapley values.
3. The predictive power of the independent variables would increase with the horizon - This is correct, as the autoregressive part is no longer capable of describing all of the changes in the dataset.

4. The predictive power of logarithmic returns would be higher compared to static variables for the smaller horizons - This is correct, as dynamic variables are losing their power on the longer horizons.
5. Statistical areas with lower prices would have better prediction errors than those with higher prices - This is correct, even after considering that in percentage terms they are actually at the same level of error.
6. The error of the models would increase significantly between the 4 and 13 week forecasts, but the difference between the 13 and 26 week forecasts would be less dramatic - This is correct, as building material indices allow to mitigate part of the decrease in the accuracy.
7. Stacking the gradient boosting models would allow us to outperform the OLS benchmark both on the original data and on the Case-Shiller index. This hypothesis is partially correct, with the exception of the four-week prediction on the Case-Shiller data.

In conclusion, we have successfully created a stacked model of three gradient boosting models that can predict weekly real estate prices for standard households in 100 of the largest statistical areas in the US. Furthermore, this model can be used for transfer learning on different real estate indices, which adds flexibility for the real implementation in research and business purposes. Nevertheless, there may be ways to enhance the quality of the predictions for future research. One approach could be to incorporate filters that would permit the correction of significant errors. Additionally, data preprocessing techniques could be employed to mitigate the impact of seasonality. Another strategy could involve the segmentation of data through classification or clustering, which could facilitate the identification of more discernible patterns.

Data availability and used generative models

The data and code for this paper are available in the public GitHub repository at the following link: https://github.com/lebedevaale/real_estate_price_prediction.

Data sources:

- The Common Haus Price Index (Haus Services, Inc.) is a privately held company that does not offer a publicly available application programming interface (API). However, data is available for free from the company's website (Haus Services, Inc., 2024).
- All indices and indicators except DJI and S&P500 - FRED offers a publicly available API that requires a key to use, which can be accessed with various Python libraries (Federal Reserve Bank of St. Louis, 2024).
- DJI and S&P500 are two entities that can be accessed via the publicly available API offered by Yahoo Finance. This API can be used with several Python libraries (Yahoo Finance, 2024).

In order to ascertain the grammatical, punctuation and stylistic correctness of this paper, the generative model "DeepL Write" has been employed. It should be noted that the use of this model should not have affected the originality of this paper. It is not the policy of the developers of "DeepL Write" to publicly disclose the version of the software that has been used in this instance: https://www.deepl.com/write

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. https://doi.org/10.1145/3292500.3330701

Alfaro-Navarro, J.-L., Cano, E. L., Alfaro-Cortés, E., García, N., Gámez, M., & Larraz, B. (2020). A Fully Automated Adjustment of Ensemble Methods in Machine Learning for Modeling Complex Real Estate Systems. *Complexity*, *2020*, 1–12. https://doi.org/10.1155/2020/5287263

Baldominos, A., Blanco, I., Moreno, A., Iturrarte, R., Bernárdez, Ó., & Afonso, C. (2018). Identifying Real Estate Opportunities Using Machine Learning. *Applied Sciences*, *8*(11), 2321. https://doi.org/10.3390/app8112321

Benedek Rózemberczki, Watson, L., Bayer, P., Yang, H.-T., Kiss, O., Nilsson, S., & Sarkar, R. (2022). The Shapley Value in Machine Learning. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2202.05594

Bentéjac, C., Csörgő, A., & Martínez-Muñoz, G. (2020). A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, *54*. https://doi.org/10.1007/s10462-020-09896-5

Canziani, A., Paszke, A., & Culurciello, E. (2017, April 14). *An Analysis of Deep Neural Network Models for Practical Applications*. ArXiv.org. https://doi.org/10.48550/arXiv.1605.07678

Case, K. E., & Shiller, R. J. (1990). Forecasting Prices and Excess Returns in the Housing Market. *Real Estate Economics*, *18*(3), 253–273. https://doi.org/10.1111/1540-6229.00521

Case, K. E., Shiller, R. J., & Weiss, A. N. (1993). Index- Based Futures And Options Markets In Real Estate. *The Journal of Portfolio Management*, *19*(2), 83–92. https://doi.org/10.3905/jpm.1993.409441

Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, *7*(3), 1247–1250. https://doi.org/10.5194/gmd-7-1247-2014

Chen, T., & Guestrin, C. (2016). XGBoost: a Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*, 785–794. https://doi.org/10.1145/2939672.2939785

Chiu, S.-M., Chen, Y.-C., & Lee, C. (2021). Estate price prediction system based on temporal and spatial features and lightweight deep learning model. *Applied Intelligence*, *52*(1), 808–834. https://doi.org/10.1007/s10489-021-02472-6

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*. https://doi.org/10.1007/s10901-022-09937-1

Dimopoulos, T., Tyralis, H., Bakas, N. P., & Hadjimitsis, D. (2018). Accuracy measurement of Random Forests and Linear Regression for mass appraisal models that estimate the prices of residential apartments in Nicosia, Cyprus. *Advances in Geosciences*, *45*, 377–382. https://doi.org/10.5194/adgeo-45-377-2018

Dmitriev, A., Lebedev, A., Kornilov, V., & Dmitriev, V. (2022). Multifractal Early Warning Signals about Sudden Changes in the Stock Exchange States. *Complexity*, *2022*, 1–10. https://doi.org/10.1155/2022/8177307

Drobnič, F., Kos, A., & Pustišek, M. (2020). On the Interpretability of Machine Learning Models and Experimental Feature Selection in Case of Multicollinear Data. *Electronics*, *9*(5), 761. https://doi.org/10.3390/electronics9050761

Federal Reserve Bank of St. Louis. (2024). *FRED Economic Data*. Stlouisfed.org; Federal Reserve Bank of St. Louis. https://fred.stlouisfed.org/

Ferreira, P., C. Le, D., & Zincir-Heywood, N. (2019, October 1). *Exploring Feature Normalization and Temporal Information for Machine Learning Based Insider Threat Detection*. IEEE Xplore. https://doi.org/10.23919/CNSM46954.2019.9012708

Haus Services, Inc. (2024, March 13). *The weekly Common Haus Price Index*. Haus.com. https://haus.com/resources/the-common-haus-price-index

Hausler, J., Ruscheinsky, J., & Lang, M. (2018). News-based sentiment analysis in real estate: a machine learning approach. *Journal of Property Research*, *35*(4), 344–371. https://doi.org/10.1080/09599916.2018.1551923

Hodson, T. O. (2022). Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geoscientific Model Development*, *15*(14), 5481–5487. https://doi.org/10.5194/gmd-15-5481-2022

Jo, J.-M. (2019). Effectiveness of Normalization Pre-Processing of Big Data to the Machine Learning Performance. *The Journal of the Korea Institute of Electronic Communication Sciences*, *14*(3), 547–552. https://doi.org/10.13067/JKIECS.2019.14.3.547

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. *Advances in Neural Information Processing Systems*, *30*. https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html

Kuşan, H., Aytekin, O., & Özdemir, İ. (2010). The use of fuzzy logic in predicting house selling price. *Expert Systems with Applications*, *37*(3), 1808–1813. https://doi.org/10.1016/j.eswa.2009.07.031

Lee, J., Kim, H., & Shim, G. (2022). Prediction Model of Real Estate ROI with the LSTM Model based on AI and Bigdata. *International Journal of Advanced Smart Convergence*, *11*(1), 19–27. https://doi.org/10.7236/IJASC.2022.11.1.19

Li, L., & Chu, K.-H. (2017, May 1). *Prediction of real estate price variation based on economic parameters*. IEEE Xplore. https://doi.org/10.1109/ICASI.2017.7988353

Loh, S. B., Debaditya, R., & Basura, F. (2022). Long-term Action Forecasting Using Multi-headed Attention-based Variational Recurrent Neural Networks. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. https://doi.org/10.1109/cvprw56347.2022.00270

Lundberg, S., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Neural Information Processing Systems*, *30*, 4768–4777.

Ma, Y., Zhang, Z., Ihler, A., & Pan, B. (2018). Estimating Warehouse Rental Price using Machine Learning Techniques. *International Journal of Computers Communications & Control*, *13*(2), 235–250. https://doi.org/10.15837/ijccc.2018.2.3034

Meharie, M. G., Mengesha, W. J., Gariy, Z. A., & Mutuku, R. N. N. (2021). Application of stacking ensemble machine learning algorithm in predicting the cost of highway construction projects. *Engineering, Construction and Architectural Management*, *ahead-of-print*(ahead-of-print). https://doi.org/10.1108/ecam-02-2020-0128

Mohd, T., Nur Izzah Jamil, Noraini Johari, Abdullah, L., & Suraya Masrom. (2020). An Overview of Real Estate Modelling Techniques for House Price Prediction. *Springer, Singapore.*, 321–338. https://doi.org/10.1007/978-981-15-3859-9_28

Mubarak, M., Tahir, A., Waqar, F., Haneef, I., McArdle, G., Bertolotto, M., & Saeed, M. T. (2022). A Map-Based Recommendation System and House Price Prediction Model for Real Estate. *ISPRS International Journal of Geo-Information*, *11*(3), 178. https://doi.org/10.3390/ijgi11030178

Ng, A. Y. (2004). Feature selection, L1 vs. L2 regularization, and rotational invariance. *Twenty-First International Conference on Machine Learning - ICML '04*. https://doi.org/10.1145/1015330.1015435

Ostroumova Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2017). CatBoost: unbiased boosting with categorical features. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1706.09516

Park, B., & Bae, J. K. (2015). Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data. *Expert Systems with Applications*, *42*(6), 2928–2934. https://doi.org/10.1016/j.eswa.2014.11.040

Pavlyshenko, B. (2018, August 1). *Using Stacking Approaches for Machine Learning Models*. IEEE Xplore. https://doi.org/10.1109/DSMP.2018.8478522

Probst, P., Boulesteix, A.-L., & Bischl, B. (2019). Tunability: Importance of Hyperparameters of Machine Learning Algorithms. *Journal of Machine Learning Research*, *20*(53), 1–32. https://www.jmlr.org/papers/v20/18-444.html

Reed, R. (2016). The relationship between house prices and demographic variables. *International Journal of Housing Markets and Analysis*, *9*(4), 520–537. https://doi.org/10.1108/ijhma-02-2016-0013

Rios, R. A., & Mello, R. F. de. (2016). Applying Empirical Mode Decomposition and mutual information to separate stochastic and deterministic influences embedded in signals. *Signal Processing*, *118*, 159–176. https://doi.org/10.1016/j.sigpro.2015.07.003

Rosenstein, M. T., Collins, J. J., & De Luca, C. J. (1993). A practical method for calculating largest Lyapunov exponents from small data sets. *Physica D: Nonlinear Phenomena*, *65*(1-2), 117–134. https://doi.org/10.1016/0167-2789(93)90009-p

Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2018, February 22). *Recent Advances in Recurrent Neural Networks*. ArXiv.org. https://doi.org/10.48550/arXiv.1801.01078

Salzman, D., & Zwinkels, R. C. J. (2017). Behavioral Real Estate. *Journal of Real Estate Literature*, *25*(1), 77–106. https://doi.org/10.1080/10835547.2017.12090455

Schindler, F. (2011). Predictability and Persistence of the Price Movements of the S&P/Case-Shiller House Price Indices. *The Journal of Real Estate Finance and Economics*, *46*(1), 44–90. https://doi.org/10.1007/s11146-011-9316-1

Statista. (2024, April). *Real Estate - United States | Statista Market Forecast*. Statista. https://www.statista.com/outlook/fmo/real-estate/united-states#value

Sutskever, I., Vinyals, O., & Le, Q. V. (2014, December 14). *Sequence to Sequence Learning with Neural Networks*. ArXiv.org. https://doi.org/10.48550/arXiv.1409.3215

van Dijk, D. W., Geltner, D. M., & van de Minne, A. M. (2020). The Dynamics of Liquidity in Commercial Property Markets: Revisiting Supply and Demand Indexes in Real Estate. *The Journal of Real Estate Finance and Economics*. https://doi.org/10.1007/s11146-020-09782-5

Walthert, L., & Sigrist, F. (2019). Deep Learning for Real Estate Price Prediction. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3393434

Wang, X., Wen, J., Zhang, Y., & Wang, Y. (2014). Real estate price forecasting based on SVM optimized by PSO. *Optik*, *125*(3), 1439–1443. https://doi.org/10.1016/j.ijleo.2013.09.017

Winter, E. (2002). Chapter 53 The shapley value. *Handbook of Game Theory with Economic Applications*, 2025–2054. https://doi.org/10.1016/s1574-0005(02)03016-3

Yahoo Finance. (2024). *Yahoo Finance - Business Finance, Stock Market, Quotes, News*. Yahoo Finance. https://finance.yahoo.com/

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, October 14). *Machine Learning, Deep Learning, and Hedonic Methods for Real Estate Price Prediction*. ArXiv.org. https://doi.org/10.48550/arXiv.2110.07151

Ying, X. (2019). An Overview of Overfitting and its Solutions. *Journal of Physics: Conference Series*, *1168*(2), 022022. https://doi.org/10.1088/1742-6596/1168/2/022022

Yu, H., & Wu, J. (n.d.). *Real Estate Price Prediction with Regression and Classification CS 229 Autumn 2016 Project Final Report*. https://cs229.stanford.edu/proj2016/report/WuYu_HousingPrice_report.pdf

Zhu, M. H., & Gupta, S. (2017). To prune, or not to prune: exploring the efficacy of pruning for model compression. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1710.01878

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., & He, Q. (2021). A Comprehensive Survey on Transfer Learning. *Proceedings of the IEEE*, *109*(1), 43–76. https://doi.org/10.1109/jproc.2020.3004555

Appendix A

Table AI “Statistical areas that were used in modelling”

Code	City	State	Minimal date	Maximum date	Number of entries
10420	Akron	OH	2011-01-07	2023-03-24	638
10580	Albany-Schenectady-Troy	NY	2010-03-12	2023-03-24	681
10740	Albuquerque	NM	2010-01-22	2023-03-24	688
10900	Allentown-Bethlehem-Easton	PA-NJ	2010-01-22	2023-03-24	688
11244	Anaheim-Santa Ana-Irvine	CA	2010-01-22	2023-03-24	688
12060	Atlanta-Sandy Springs-Roswell	GA	2010-06-18	2023-03-24	585
12420	Austin-Round Rock	TX	2010-01-22	2023-03-24	688
12580	Baltimore-Columbia-Towson	MD	2010-01-22	2023-03-24	688
12940	Baton Rouge	LA	2010-09-17	2023-03-24	654
13820	Birmingham-Hoover	AL	2012-02-10	2023-03-24	581
14454	Boston	MA	2010-01-22	2023-03-24	556
14860	Bridgeport-Stamford-Norwalk	CT	2010-01-22	2023-03-24	688
15380	Buffalo-Cheektowaga-Niagara Falls	NY	2010-07-09	2023-03-24	664
15764	Cambridge-Newton-Framingham	MA	2010-01-22	2023-03-24	557
15804	Camden	NJ	2010-01-22	2023-03-24	688
15980	Cape Coral-Fort Myers	FL	2010-01-22	2023-03-24	688
16700	Charleston-North Charleston	SC	2010-01-22	2023-03-24	531
16740	Charlotte-Concord-Gastonia	NC-SC	2010-01-22	2023-03-24	688
16974	Chicago-Naperville-Arlington Heights	IL	2010-01-22	2023-03-24	688
17140	Cincinnati	OH-KY-IN	2010-01-22	2023-03-24	688
17460	Cleveland-Elyria	OH	2010-01-22	2023-03-24	688
17820	Colorado Springs	CO	2010-01-22	2023-03-24	607
17900	Columbia	SC	2010-01-29	2023-03-24	536
18140	Columbus	OH	2010-01-22	2023-03-24	688
19124	Dallas-Plano-Irving	TX	2010-01-22	2023-03-24	688
19380	Dayton	OH	2010-07-09	2023-03-24	664
19660	Deltona-Daytona Beach-Ormond Beach	FL	2010-07-23	2023-03-24	662
19740	Denver-Aurora-Lakewood	CO	2010-01-22	2023-03-24	688
19804	Detroit-Dearborn-Livonia	MI	2010-01-22	2023-03-24	688
21340	El Paso	TX	2010-07-09	2023-03-24	664
22744	Fort Lauderdale-Pompano Beach-Deerfield Beach	FL	2010-01-22	2023-03-24	688
23104	Fort Worth-Arlington	TX	2010-01-22	2023-03-24	688
23844	Gary	IN	2010-01-22	2023-03-24	688
24340	Grand Rapids-Wyoming	MI	2010-02-05	2023-03-24	666
24660	Greensboro-High Point	NC	2010-04-23	2023-03-24	665
24860	Greenville-Anderson-Mauldin	SC	2010-01-29	2023-03-24	687

25540	Hartford-West Hartford-East Hartford	CT	2010-01-22	2023-03-24	688
26420	Houston-The Woodlands-Sugar Land	TX	2010-01-22	2023-03-24	688
26900	Indianapolis-Carmel-Anderson	IN	2010-01-22	2023-03-24	688
27260	Jacksonville	FL	2010-01-22	2023-03-24	627
28140	Kansas City	MO-KS	2010-01-22	2023-03-24	688
28940	Knoxville	TN	2010-01-22	2023-03-24	688
29404	Lake County-Kenosha County	IL-WI	2010-01-22	2023-03-24	688
29460	Lakeland-Winter Haven	FL	2010-01-22	2023-03-24	684
29820	Las Vegas-Henderson-Paradise	NV	2010-01-22	2023-03-24	688
30780	Little Rock-North Little Rock-Conway	AR	2010-08-06	2023-03-24	660
31084	Los Angeles-Long Beach-Glendale	CA	2010-01-22	2023-03-24	688
31140	Louisville/Jefferson County	KY-IN	2010-01-22	2023-03-24	688
32820	Memphis	TN-MS-AR	2010-01-22	2023-03-24	688
33124	Miami-Miami Beach-Kendall	FL	2010-01-22	2023-03-24	688
33340	Milwaukee-Waukesha-West Allis	WI	2010-01-22	2023-03-24	688
33460	Minneapolis-St. Paul-Bloomington	MN-WI	2010-01-22	2023-03-24	688
33874	Montgomery County-Bucks County-Chester County	PA	2010-01-22	2023-03-24	688
34980	Nashville-Davidson--Murfreesboro--Franklin	TN	2010-01-22	2023-03-24	688
35004	Nassau County-Suffolk County	NY	2010-01-22	2023-03-24	688
35084	Newark	NJ-PA	2010-01-22	2023-03-24	688
35300	New Haven-Milford	CT	2010-01-22	2023-03-24	688
35380	New Orleans-Metairie	LA	2010-01-22	2023-03-24	688
35614	New York-Jersey City-White Plains	NY-NJ	2010-01-22	2023-03-24	688
35840	North Port-Sarasota-Bradenton	FL	2010-01-22	2023-03-24	688
36084	Oakland-Hayward-Berkeley	CA	2010-01-22	2023-03-24	688
36420	Oklahoma City	OK	2010-01-22	2023-03-24	688
36540	Omaha-Council Bluffs	NE-IA	2010-07-16	2023-03-24	663
36740	Orlando-Kissimmee-Sanford	FL	2010-01-22	2023-03-24	687
37100	Oxnard-Thousand Oaks-Ventura	CA	2010-01-22	2023-03-24	631
37340	Palm Bay-Melbourne-Titusville	FL	2010-07-23	2023-03-24	662
37964	Philadelphia	PA	2010-01-22	2023-03-24	688
38060	Phoenix-Mesa-Scottsdale	AZ	2010-01-22	2023-03-24	688
38300	Pittsburgh	PA	2010-01-22	2023-03-24	676
38900	Portland-Vancouver-Hillsboro	OR-WA	2010-01-22	2023-03-24	688
39300	Providence-Warwick	RI-MA	2010-01-22	2023-03-24	688
39580	Raleigh	NC	2010-01-22	2023-03-24	688
40060	Richmond	VA	2010-01-22	2023-03-24	688
40140	Riverside-San Bernardino-Ontario	CA	2010-01-22	2023-03-24	688
40380	Rochester	NY	2010-01-29	2023-03-24	670
40900	Sacramento--Roseville--Arden-Arcade	CA	2010-01-22	2023-03-24	688

41180	St. Louis	MO-IL	2010-01-22	2023-03-24	670
41620	Salt Lake City	UT	2010-01-22	2023-03-24	688
41700	San Antonio-New Braunfels	TX	2014-04-04	2023-03-24	467
41740	San Diego-Carlsbad	CA	2010-01-22	2023-03-24	688
41884	San Francisco-Redwood City-South San Francisco	CA	2010-01-22	2023-03-24	688
41940	San Jose-Sunnyvale-Santa Clara	CA	2010-01-22	2023-03-24	688
42644	Seattle-Bellevue-Everett	WA	2010-01-22	2023-03-24	688
43524	Silver Spring-Frederick-Rockville	MD	2010-01-22	2023-03-24	688
45060	Syracuse	NY	2010-04-23	2023-03-24	662
45104	Tacoma-Lakewood	WA	2010-01-22	2023-03-24	688
45300	Tampa-St. Petersburg-Clearwater	FL	2010-01-22	2023-03-24	688
45780	Toledo	OH	2010-04-30	2023-03-24	674
46060	Tucson	AZ	2010-01-22	2023-03-24	603
46140	Tulsa	OK	2010-02-05	2023-03-24	653
46520	Urban Honolulu	HI	2010-01-22	2023-03-24	685
47260	Virginia Beach-Norfolk-Newport News	VA-NC	2010-01-22	2023-03-24	688
47664	Warren-Troy-Farmington Hills	MI	2010-01-22	2023-03-24	688
47894	Washington-Arlington-Alexandria	DC-VA-MD-WV	2010-01-22	2023-03-24	687
48424	West Palm Beach-Boca Raton-Delray Beach	FL	2010-01-22	2023-03-24	688
48620	Wichita	KS	2010-03-26	2023-03-24	679
48864	Wilmington	DE-MD-NJ	2010-01-22	2023-03-24	688
49180	Winston-Salem	NC	2010-03-12	2023-03-24	670
49340	Worcester	MA-CT	2010-04-09	2023-03-24	669
49660	Youngstown-Warren-Boardman	OH-PA	2010-09-17	2023-03-24	654

Table AII “Stationarity and Measure of Chaos for Original Variables”

Variable	**ADF statistics**	**ADF p-value**	**Largest Lyapunov Exponent**	**Hurst Exponent**
CHPI	-0.386	0.912	0.002	0.892
KeyRate	-0.819	0.814	0.003	0.946
CPI	2.961	1.000	0.004	0.910
VIX	-3.346	0.013	-0.009	0.773
PPI	-0.370	0.915	0.014	0.860
MortgageRate30	-0.531	0.886	0.011	0.914
Electricity	0.704	0.990	0.008	0.880
Water	2.178	0.999	0.000	0.910
Plywood	-1.458	0.554	0.022	0.901

Steel	-2.224	0.198	0.009	0.903
Glass	2.377	0.999	0.003	0.920
Concrete	2.901	1.000	0.002	0.906
Unemployment	-2.648	0.083	-0.012	0.849
Yield10Y	-1.465	0.551	-0.000	0.935
DJI	-0.903	0.787	0.002	0.860
S&P500	-0.911	0.784	0.004	0.886
KeyRate_4_week_logdyn	-4.881	0.000	0.025	0.590
CPI_4_week_logdyn	-2.828	0.054	0.008	0.779
VIX_4_week_logdyn	-7.290	0.000	0.018	0.438
PPI_4_week_logdyn	-7.377	0.000	0.024	0.737
MortgageRate30_4_week_logdyn	-4.188	0.001	0.029	0.692
Electricity_4_week_logdyn	-2.457	0.126	0.028	0.689
Water_4_week_logdyn	-4.135	0.001	-0.014	0.704
Plywood_4_week_logdyn	-6.202	0.000	0.019	0.806
Steel_4_week_logdyn	-2.853	0.051	0.016	0.805
Glass_4_week_logdyn	-3.636	0.005	0.018	0.698
Concrete_4_week_logdyn	-2.091	0.248	0.000	0.743
Unemployment_4_week_logdyn	-5.462	0.000	0.030	0.671
Yield10Y_4_week_logdyn	-4.947	0.000	0.031	0.665
DJI_4_week_logdyn	-5.896	0.000	0.030	0.623
S&P500_4_week_logdyn	-5.714	0.000	0.039	0.637
KeyRate_13_week_logdyn	-3.835	0.003	0.027	0.921
CPI_13_week_logdyn	-2.723	0.070	0.035	0.844
VIX_13_week_logdyn	-6.230	0.000	0.015	0.819
PPI_13_week_logdyn	-4.946	0.000	0.019	0.844
MortgageRate30_13_week_logdyn	-2.973	0.038	0.017	0.882
Electricity_13_week_logdyn	-2.459	0.126	0.020	0.843
Water_13_week_logdyn	-3.476	0.009	-0.006	0.823
Plywood_13_week_logdyn	-5.612	0.000	0.019	0.869
Steel_13_week_logdyn	-3.356	0.013	0.014	0.902
Glass_13_week_logdyn	-3.594	0.006	0.018	0.886
Concrete_13_week_logdyn	-2.819	0.056	0.014	0.909
Unemployment_13_week_logdyn	-5.555	0.000	0.014	0.789
Yield10Y_13_week_logdyn	-3.323	0.014	0.011	0.879
DJI_13_week_logdyn	-5.015	0.000	0.021	0.856
S&P500_13_week_logdyn	-4.650	0.000	0.018	0.806
KeyRate_26_week_logdyn	-3.030	0.032	0.012	0.964
CPI_26_week_logdyn	-2.345	0.158	0.032	0.877
VIX_26_week_logdyn	-4.909	0.000	0.013	0.827
PPI_26_week_logdyn	-4.274	0.000	0.025	0.853
MortgageRate30_26_week_logdyn	-3.772	0.003	0.007	0.937
Electricity_26_week_logdyn	-3.194	0.020	0.013	0.855

Water_26_week_logdyn	-3.890	0.002	0.020	0.868
Plywood_26_week_logdyn	-3.567	0.006	0.013	0.909
Steel_26_week_logdyn	-2.778	0.062	0.001	0.908
Glass_26_week_logdyn	-3.746	0.004	0.003	0.908
Concrete_26_week_logdyn	-2.542	0.106	0.018	0.909
Unemployment_26_week_logdyn	-4.929	0.000	0.005	0.850
Yield10Y_26_week_logdyn	-3.789	0.003	0.008	0.930
DJI_26_week_logdyn	-4.086	0.001	0.010	0.871
S&P500_26_week_logdyn	-3.995	0.001	0.006	0.872

Appendix B

Table BI "The coefficients for the stacked models"

Lag	Separate model	Coefficient	Standard error	P-value
4 weeks	LightGBM	0.3427	0.027	0.000
	XGBoost	0.6574	0.027	0.000
	CatBoost	-	-	> 0.05
13 weeks	LightGBM	0.2971	0.025	0.000
	XGBoost	0.7029	0.025	0.000
	CatBoost	-	-	> 0.05
26 weeks	LightGBM	0.5355	0.026	0.000
	XGBoost	0.4644	0.026	0.000
	CatBoost	-	-	> 0.05

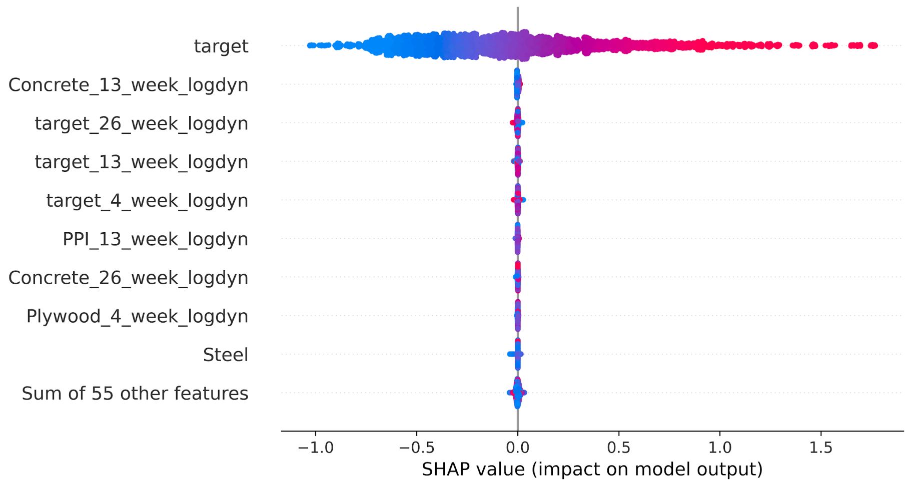

Figure B1 "Shapley values for LightGBM with 4 weeks prediction"

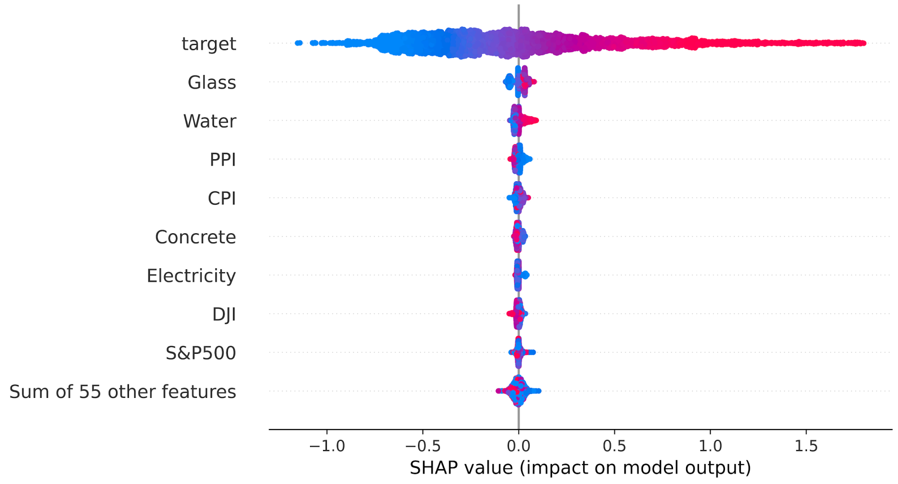

Figure B2 “Shapley values for CatBoost with 4 weeks prediction”

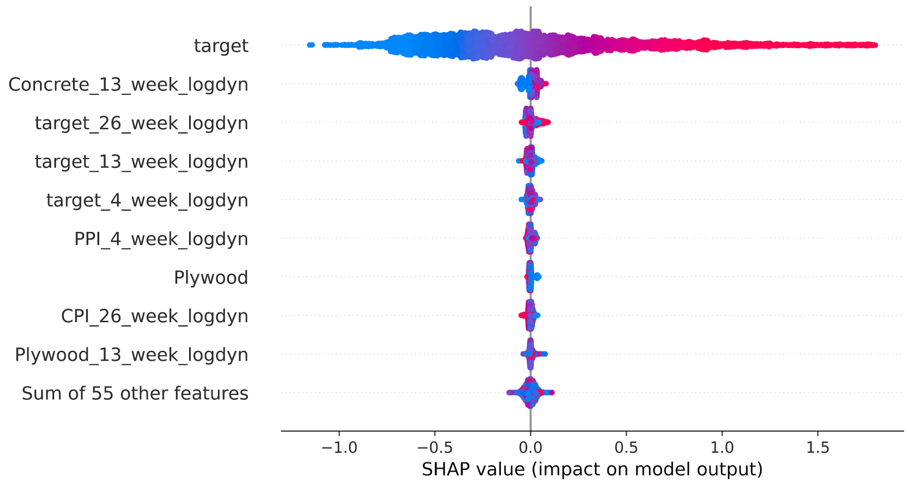

Figure B3 “Shapley values for LightGBM with 13 weeks prediction”

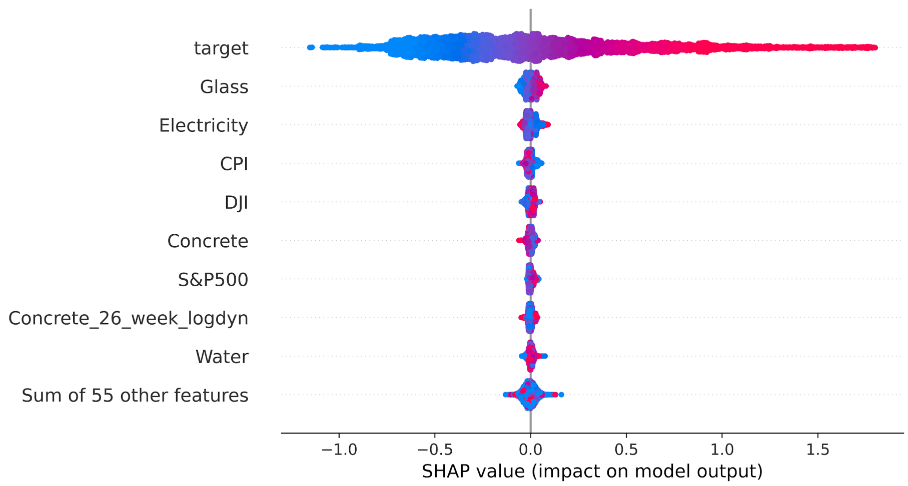

Figure B4 “Shapley values for CatBoost with 13 weeks prediction”

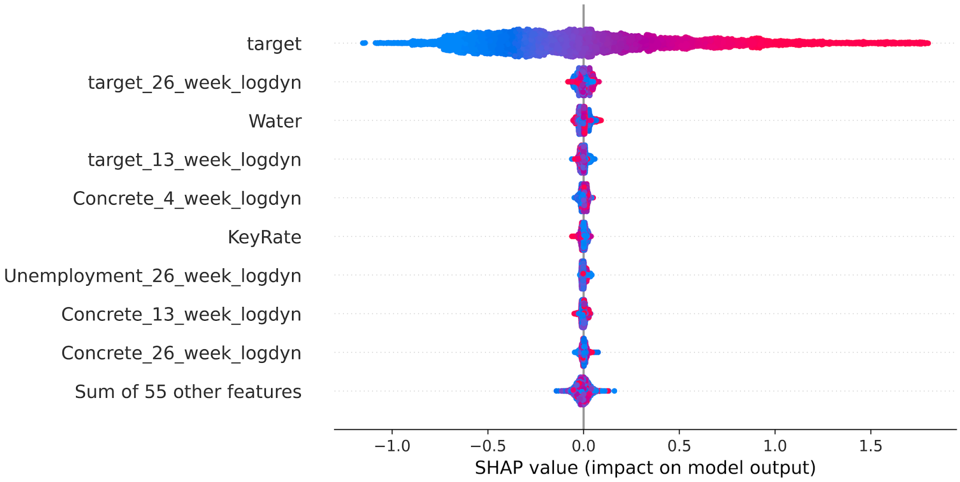

Figure B5 “Shapley values for LightGBM with 26 weeks prediction”

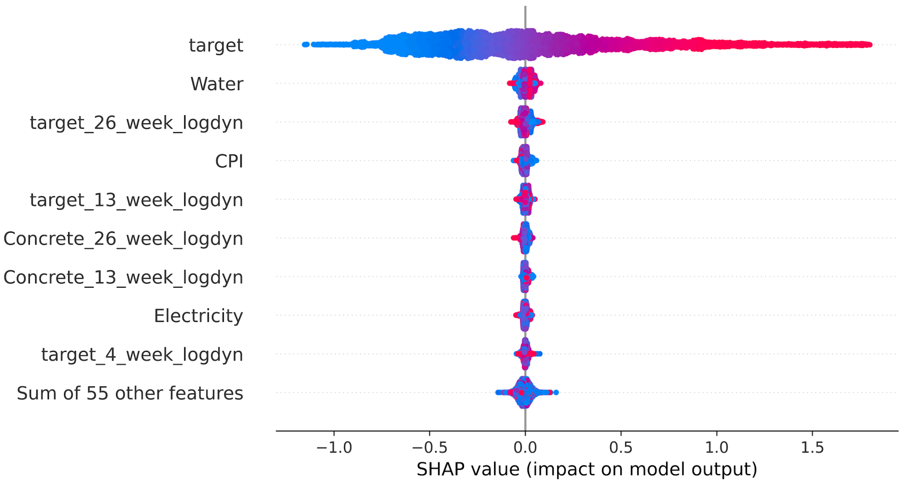

Figure B6 “Shapley values for CatBoost with 26 weeks prediction”

Table BII “RMSE and MAPE (%) for stacked model on Case-Shiller data”

Lag	Period	RMSE ($) / MAPE (%)	RMSE ($) / MAPE (%) with Smoothening
4 weeks	Whole	2 242.050 / 0.856	2 009.940 / 0.76
	Before 2010-01-01	2 975.273 / 1.302	2 856.586 / 1.262
	2010-01-01 - 2018-01-01	1 375.758 / 0.630	1 110.665 / 0.507
	After 2018-01-01	2 378.139 / 0.734	1 981.697 / 0.623
13 weeks	Whole	7 182.020 / 2.607	6 044.992 / 2.065
	Before 2010-01-01	10 380.332 / 4.365	9 974.363 / 4.234
	2010-01-01 - 2018-01-01	3 963.906 / 1.803	2 210.441 / 1.102
	After 2018-01-01	6 834.538 / 2.010	4 382.027 / 1.286
26 weeks	Whole	8 952.761 / 3.447	6 947.070 / 2.723
	Before 2010-01-01	10 850.050 / 5.289	10 319.326 / 4.915
	2010-01-01 - 2018-01-01	6 205.332 / 2.669	3 904.350 / 1.983
	After 2018-01-01	10 120.035 / 2.700	6 077.45 / 1.566

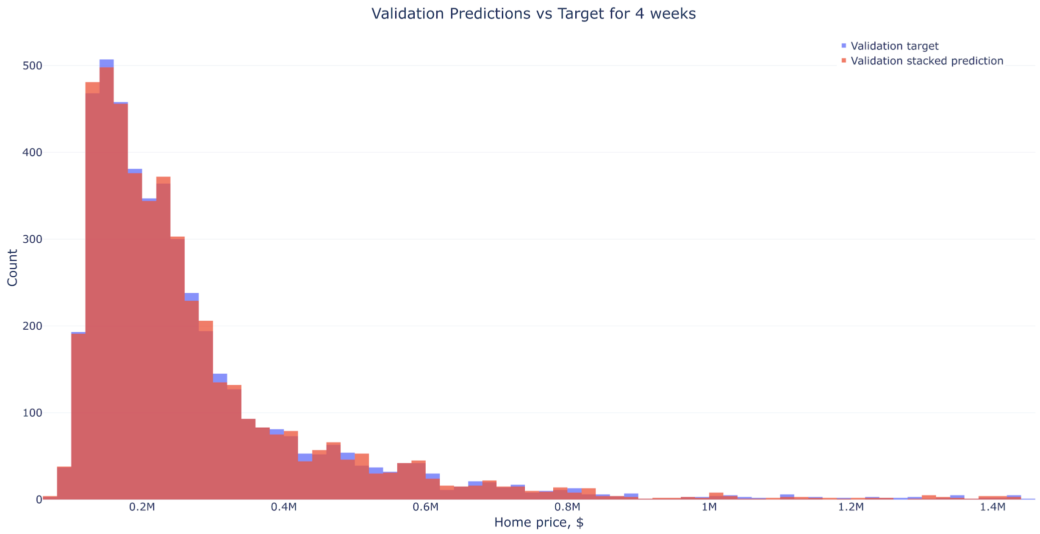

Figure B7 “Validation predictions vs target for 4 weeks prediction”

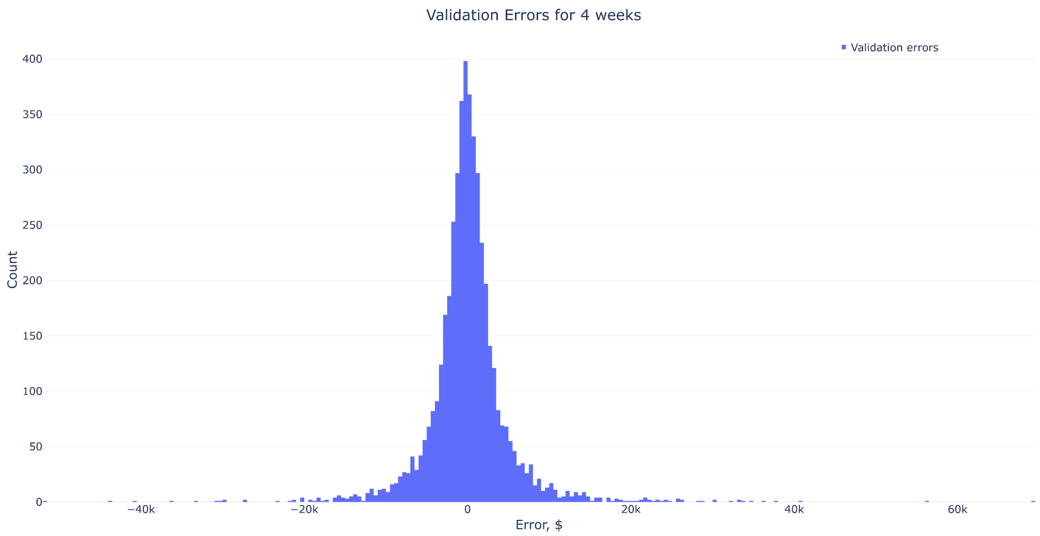

Figure B8 “Validation errors for 4 weeks prediction”

Table BIII “Validation buckets for 4 weeks prediction”

Bucket	**Lower**	**Upper**	**Number**	**RMSE**	**MAE**	**MAPE (%)**
1	74 623.23	211 646.52	2 254	2 608.93	1 809.77	1.138
2	211 646.52	348 669.81	1 548	4 228.82	2 949.84	1.107
3	348 669.81	485 693.09	480	6 198.43	4 236.59	1.026
4	485 693.09	622 716.38	256	8 538.65	6 198.49	1.114
5	622 716.38	759 739.67	104	11 558.97	8 894.96	1.287
6	759 739.67	896 762.95	56	14 087.18	11 270.37	1.370
7	896 762.95	1 033 786.24	18	13 415.71	11 437.12	1.172
8	1 033 786.24	1 170 809.53	16	15 346.56	11 514.28	1.061
9	1 170 809.53	1 307 832.81	13	25 217.84	18 845.96	1.491
10	1 307 832.81	1 444 856.10	18	25 448.50	20 990.76	1.521

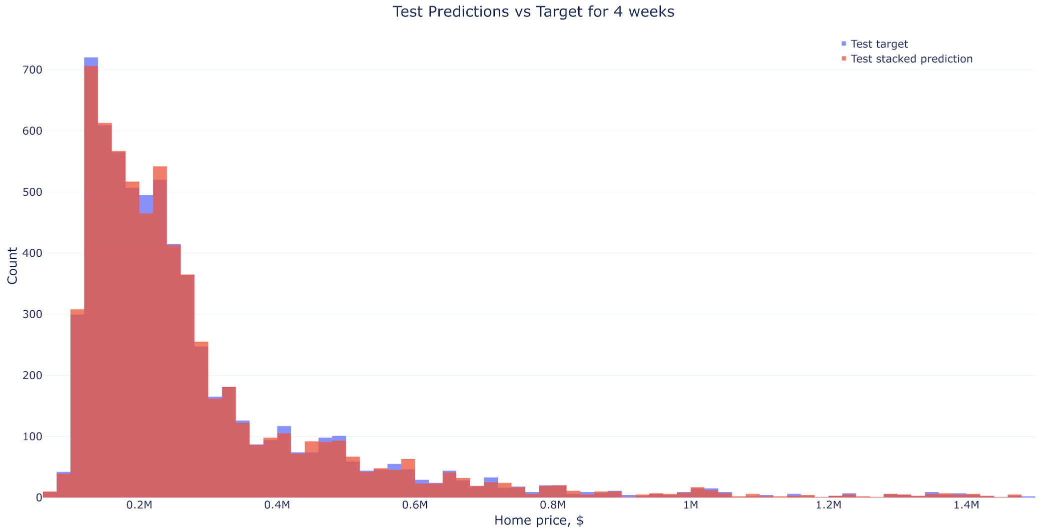

Figure B9 “Test predictions vs target for 4 weeks prediction”

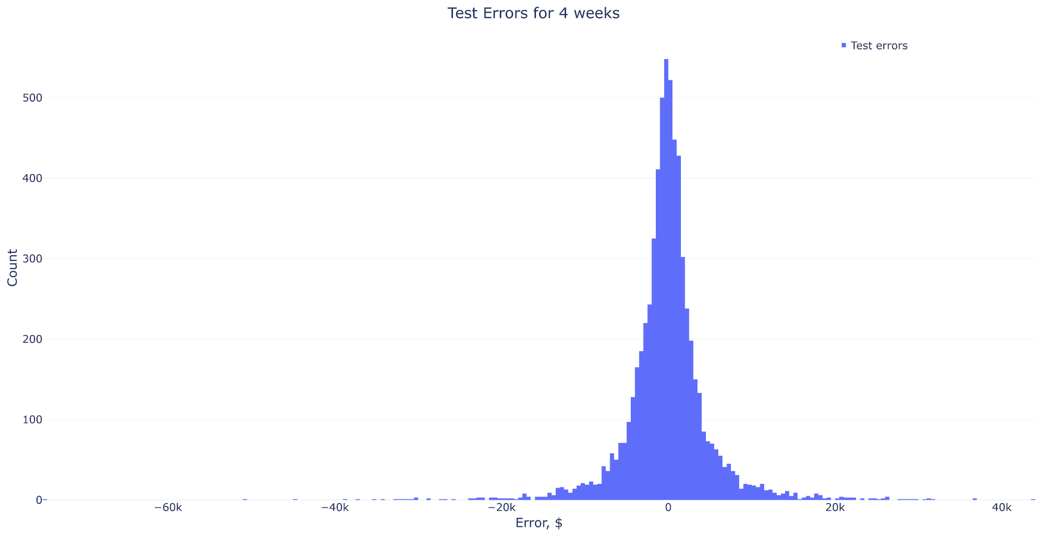

Figure B10 “Test errors for 4 weeks prediction”

Table BIV “Test buckets for 4 weeks prediction”

Bucket	**Lower**	**Upper**	**Number**	**RMSE**	**MAE**	**MAPE (%)**
1	71 697.97	213 548.93	3 075	2 654.98	1 834.34	1.162
2	213 548.93	355 399.90	2 158	4 558.06	3 104.20	1.163
3	355 399.90	497 250.86	666	6 472.75	4 631.05	1.086
4	497 250.86	639 101.82	310	8 271.75	5 991.56	1.062
5	639 101.82	780 952.78	171	10 258.85	7 852.57	1.129
6	780 952.78	922 803.75	77	11 834.03	8 696.62	1.036
7	922 803.75	1 064 654.71	61	17 761.32	12 486.21	1.242
8	1 064 654.71	1 206 505.67	17	14 705.66	11 923.17	1.048
9	1 206 505.67	1 348 356.64	27	16 205.56	11 333.50	0.890
10	1 348 356.64	1 490 207.60	31	17 018.71	13 039.02	0.934

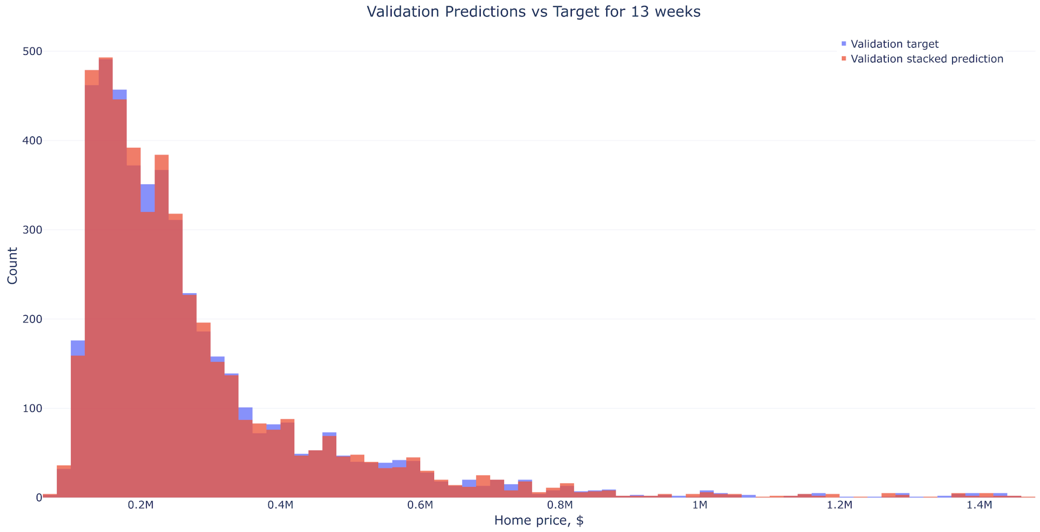

Figure B11 “Validation predictions vs target for 13 weeks prediction”

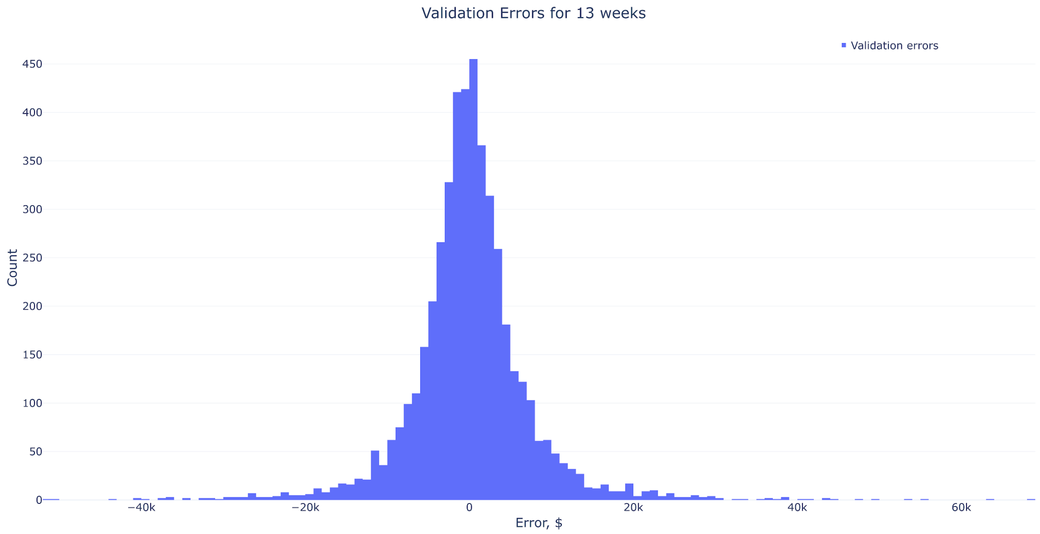

Figure B12 “Validation errors for 13 weeks prediction”

Table BV “Validation buckets for 13 weeks prediction”

Bucket	Lower	Upper	Number	RMSE	MAE	MAPE (%)
1	71 940.71	208 808.34	2134	3 885.38	2 844.15	1.802
2	208 808.34	345 675.97	1635	6 545.16	4 806.00	1.815
3	345 675.97	482 543.60	485	9 404.34	6 999.32	1.715
4	482 543.60	619 411.23	269	12 292.54	8 987.82	1.643
5	619 411.23	756 278.86	118	17 112.45	13 374.09	1.923
6	756 278.86	893 146.48	53	19 522.15	13 986.48	1.686
7	893 146.48	1 030 014.11	21	18 369.57	13 537.09	1.399
8	1 030 014.11	1 166 881.74	17	27 960.95	23 721.49	2.182
9	1 166 881.74	1 303 749.37	11	19 942.16	17 029.94	1.379
10	1 303 749.37	1 440 617.00	20	24 718.41	19 663.45	1.420

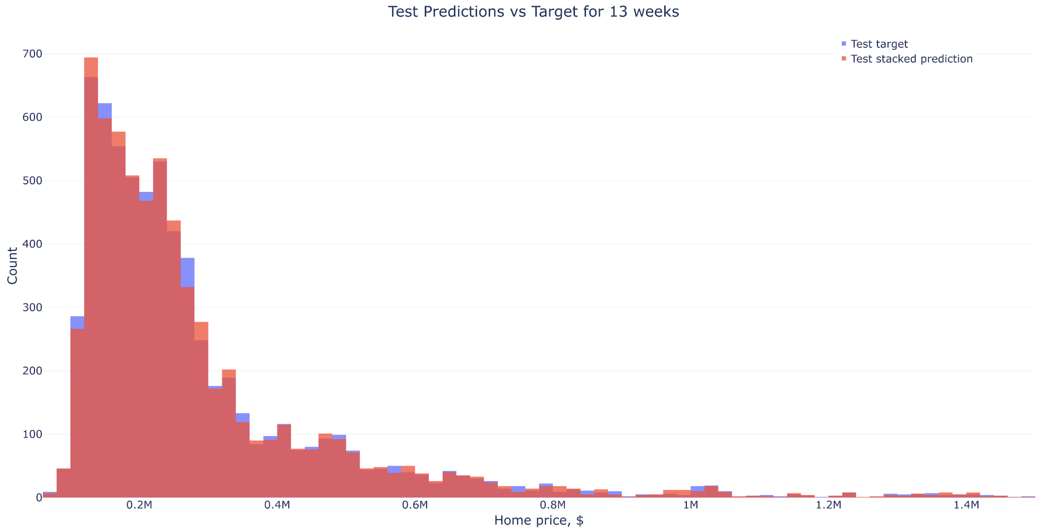

Figure B13 "Test predictions vs target for 13 weeks prediction"

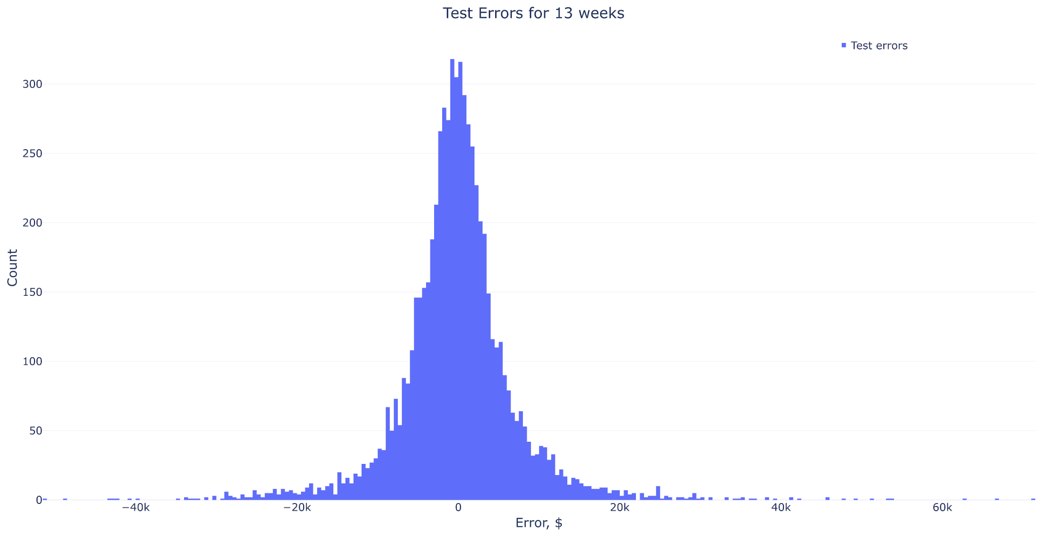

Figure B14 "Test errors for 13 weeks prediction"

Table BVI "Test buckets for 13 weeks prediction"

Bucket	Lower	Upper	Number	RMSE	MAE	MAPE (%)
1	71 808.14	213 231.28	3006	3 923.86	2 856.19	1.803
2	213 231.28	354 654.41	2196	6 644.10	4 893.08	1.831
3	354 654.41	496 077.55	673	9 480.41	7 070.48	1.658
4	496 077.55	637 500.68	320	10 875.83	8 259.94	1.483
5	637 500.68	778 923.82	177	13 461.65	9 937.24	1.424
6	778 923.82	920 346.96	75	19 098.24	14 854.79	1.772
7	920 346.96	1 061 770.09	67	19 535.86	14 656.46	1.450
8	1 061 770.09	1 203 193.23	20	21 294.32	16 519.76	1.455
9	1 203 193.23	1 344 616.36	30	24 917.96	20 457.48	1.581
10	1 344 616.36	1 486 039.50	29	25 157.16	20 588.26	1.460

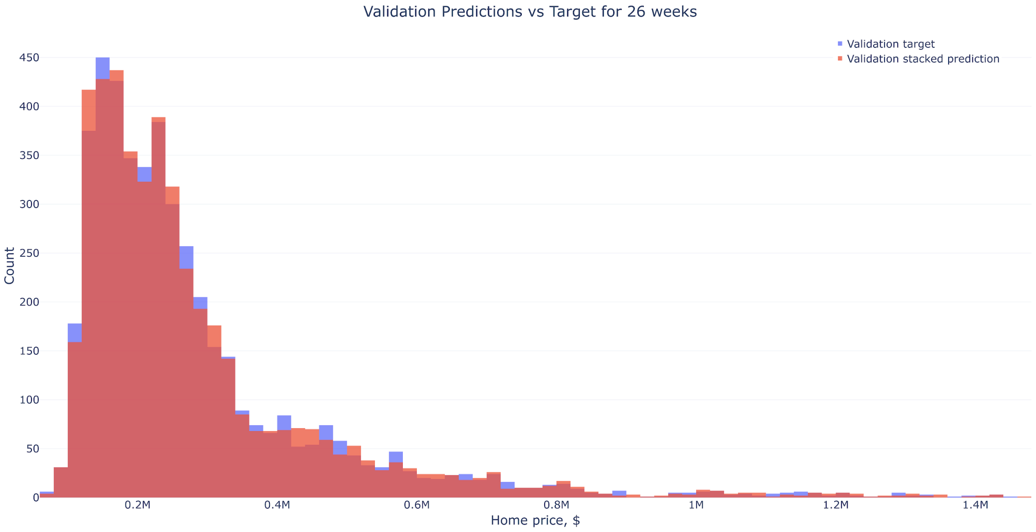

Figure B15 “Validation predictions vs target for 26 weeks prediction”

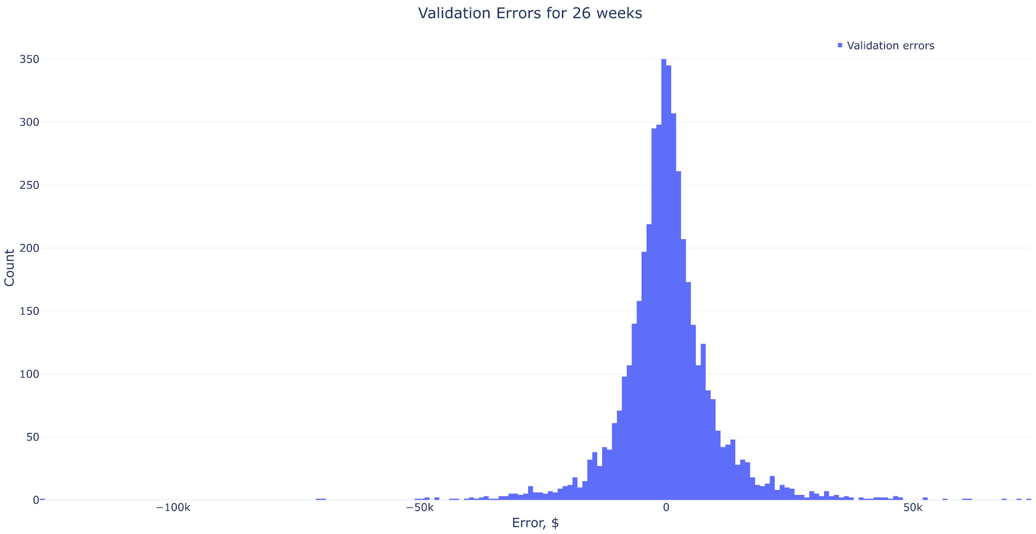

Figure B16 “Validation errors for 26 weeks prediction”

Table BVII “Validation buckets for 26 weeks prediction”

Bucket	Lower	Upper	Number	RMSE	MAE	MAPE (%)
1	71 573.30	210 129.26	1972	4 788.13	3 566.69	2.255
2	210 129.26	348 685.22	1660	9 112.68	6 528.32	2.463
3	348 685.22	487 241.18	480	13 476.37	10 076.24	2.429
4	487 241.18	625 797.14	239	15 302.01	10 927.52	1.987
5	625 797.14	764 353.10	130	18 973.24	13 953.73	2.029
6	764 353.10	902 909.06	62	19 968.76	15 008.92	1.835
7	902 909.06	1 041 465.02	25	19 874.74	16 056.18	1.620
8	1 041 465.02	1 180 020.98	29	37 400.30	26 072.12	2.327
9	1 180 020.98	1 318 576.94	18	19 948.90	16 084.64	1.288
10	1 318 576.94	1 457 132.90	12	19 325.35	15 179.10	1.086

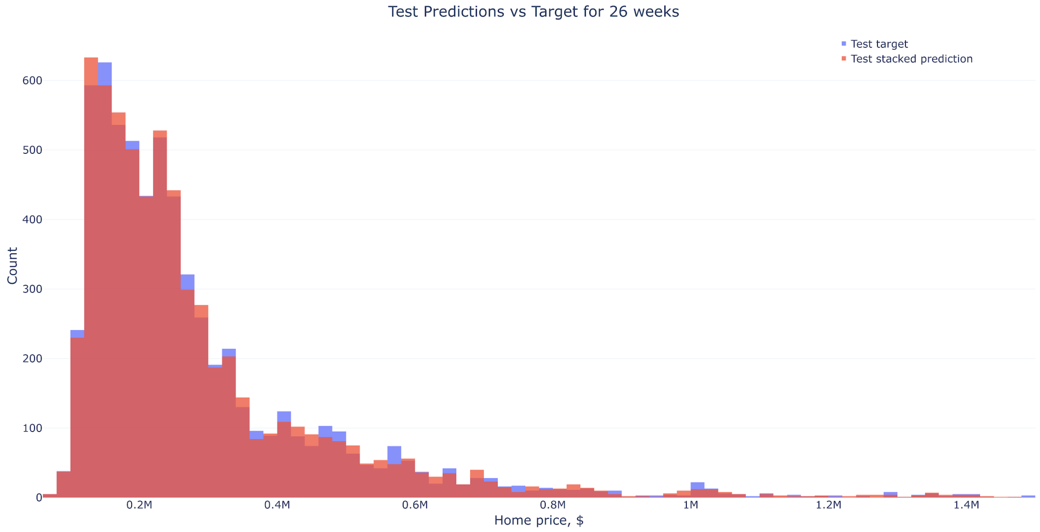

Figure B17 “Test predictions vs target for 26 weeks prediction”

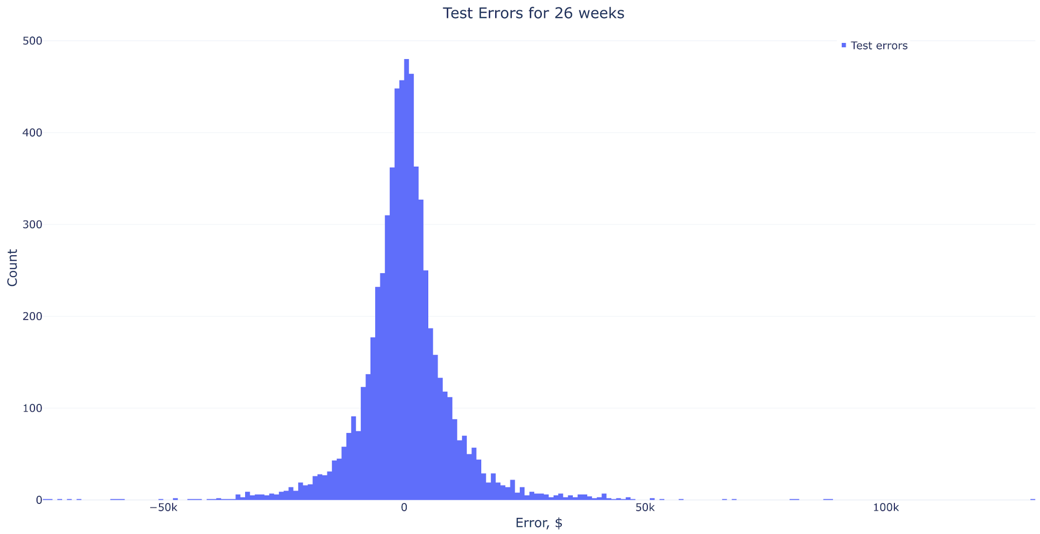

Figure B18 “Test errors for 26 weeks prediction”

Table BVIII “Test buckets for 26 weeks prediction”

Bucket	Lower	Upper	Number	RMSE	MAE	MAPE (%)
1	71 808.14	213 394.63	2833	5 100.47	3 618.12	2.268
2	213 394.63	354 981.11	2182	8 968.12	6 569.51	2.446
3	354 981.11	496 567.60	689	12 333.60	9 370.73	2.205
4	496 567.60	638 154.08	350	16 029.55	11 593.84	2.068
5	638 154.08	779 740.57	163	20 440.70	14 961.85	2.144
6	779 740.57	921 327.06	70	18 519.91	13 882.00	1.658
7	921 327.06	1 062 913.54	56	32 207.30	20 504.45	2.059
8	1 062 913.54	1 204 500.03	21	39 304.83	32 319.29	2.852
9	1 204 500.03	1 346 086.51	21	20 503.36	15 542.85	1.198
10	1 346 086.51	1 487 673.00	19	23 328.11	20 431.10	1.441