Universiteit Leiden
The Netherlands

ICT in Business

Explainable machine learning for macroeconomic forecasting of the housing market

Christie Bavelaar

Supervisors:
Marc Hilbert, Joost Visser

External Supervisor:
Mike Langen (CPB)

MASTER THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

8/4/2024

Abstract

Macroeconomic forecasts for housing prices are used to gain insights into the future state of the Dutch economy. The insights from these forecasts are used to inform policy decisions. Machine learning has been introduced as a new technique to increase the performance of these forecasting models. Machine learning models have the major disadvantage of being black-box. When it is unclear how or why the model makes certain predictions a forecaster cannot use it to reason about the future state of the economy. This study implements different multi-horizon machine learning models to predict housing prices. Shapley values are used as an explainability technique to make the predictions from this model more transparent. The different machine learning models are able to outperform a traditional forecasting model. Shapley values are useful to explain how the input results in specific forecasting predictions. They are also able to provide some insights into the economic patterns that the machine learning model has internalized.

Contents

1 Introduction 5

1.1 Problem Statement . . . 5

1.2 Research Question and Hypotheses . . . 5

1.3 Thesis Outline . . . 6

2 Business Case 6

2.1 Advantages of Machine Learning . . . 7

2.2 Risks of Machine Learning . . . 8

2.3 Explainable Machine Learning . . . 9

2.4 Requirements for the model . . . 9

3 Related Work 10

3.1 Traditional Economic Forecasting . . . 10

3.2 Machine Learning in Macroeconomic Forecasting . . . 11

3.3 Explainability in Macroeconomic Forecasting . . . 12

4 Forecasting with Machine Learning 12

4.1 Theoretical Framework . . . 12

4.1.1 Problem Definition . . . 13

4.1.2 Feature Engineering . . . 13

4.1.3 Feature Selection . . . 14

4.1.4 Machine Learning Models . . . 15

4.1.5 Hyperparameter Optimisation . . . 16

4.1.6 Performance Evaluation . . . 17

4.2 Methodology . . . 18

4.2.1 Data . . . 18

4.2.2 Model Training . . . 20

4.3 Results . . . 23

4.3.1 Out-of-sample analysis . . . 23

4.3.2 In-sample analysis . . . 24

4.4 Conclusion . . . 26

5 Explainable Machine Learning 26

5.1 Theoretical Framework . . . 27

5.1.1 Dimensions of Interpretability . . . 27

5.1.2 Characterisation of Methods . . . 30

5.1.3 Shapley Values . . . 31

5.1.4 Alternative Explanability Methods . . . 32

5.1.5 Assessment of Explanations . . . 33

5.2 Methodology . . . 34

5.2.1 SHapley Additive exPlanations (SHAP) . . . 34

5.2.2 Visualisation . . . 35

5.3 Results . . . 39

5.3.1 Local Explanation . . . 40

5.3.2 Global Explanation . . . 42

6 Discussion 43

6.1 Connection to Related Work . . . 44

6.2 Comparison to the baseline . . . 44

6.3 Interpretation of Results . . . 44

6.4 Connection to the Business Case . . . 46

6.5 Alternative Explainability Methods . . . 47

6.6 Limitations . . . 47

7 Conclusion 48

7.1 Recommendations for Future Work . . . 48

1 Introduction

Economic forecasts serve as crucial inputs for the decision-making processes of policymakers. These forecasts have a significant impact on the well-being of the citizens within a country. Consequently, the public scrutinizes these predictions extensively. It underscores the importance of both accurate and transparent forecasts of housing prices.

1.1 Problem Statement

A considerable portion of machine learning models are considered "black-box" models. In the case of an interpretable model, also called a white box model, it is understandable to a human observer how the input is transformed to create an output prediction [Loyola-González, 2019]. Black-box models create a more complex relationship between input and output that is no longer interpretable. Black-box models in machine learning offer several advantages that contribute to their widespread use. One key advantage lies in their ability to model and capture intricate relationships within complex datasets. These models can uncover patterns and correlations that may not be apparent through traditional analytical approaches. By leveraging advanced algorithms and techniques, black-box models can achieve high predictive accuracy [Loyola-González, 2019]. This high-performance potential makes them valuable tools for various domains, including economics.

However, when applied to economic forecasts, the black-box nature of these models presents a significant challenge. It becomes impossible for a human forecaster to trace predictions back to input data or assess the validity of patterns learned by the model. The model might inadvertently learn arbitrary data patterns or rely on unethical assumptions, without any means of detection.

Recognizing the need for transparency and interpretability in machine learning, researchers have devoted significant efforts to the field of explainable machine learning. Where interpretability is a passive feature that a model type can possess or not, explainability is used to refer to an active characteristic. In explainability, an effort is made to clarify the internal functions of an otherwise uninterpretable model. The primary goal of the explainabile machine learning research field is to develop techniques that enhance understanding of black-box models and provide meaningful explanations for their predictions.

1.2 Research Question and Hypotheses

In recent years, economists and data scientists have started to experiment with machine learning to improve their economic forecasts. Until now these explanations have relied on black-box models without built-in explainability mechanisms. While these models have demonstrated promising results [Goulet Coulombe et al., 2022], their lack of transparency raises concerns regarding their reliability and potential biases [Delfos et al., 2022]. As a result, there is a need for research to bridge the gap between machine learning and traditional economic forecasts by incorporating explainable machine learning techniques. This would allow economists and data scientists to leverage the power of machine learning while ensuring transparency of their forecasts.

This study aims to address a gap in the research by applying explainable machine learning to housing price forecasts. The thesis aims to answer the following research question (RQ):

RQ. How can machine learning techniques be employed for housing market prediction in an explainable manner?

To guide the research we address the following three hypotheses. In the first hypothesis (H), the focus is on constructing a black-box machine learning model capable of predicting housing prices with reasonable performance

H1: A black-box machine learning model can be used to accurately predict housing prices.

We would like to introduce an explainability method to this black-box model without hurting its performance. Post-hoc methods apply explanations after prediction without influencing the black-box model.

H2: A post-hoc explainability method can increase the interpretability of a black box model.

It is unlikely that a single technique or explanation can provide a comprehensive insight into a black box model. Instead, we aim to explore the possibility of constructing a diverse set of explanations and visualizations.

H3: It is possible to construct a combination of explanations and visualizations to achieve a satisfactory understanding of a black box model and its output.

1.3 Thesis Outline

This work is done in collaboration with the Netherlands Bureau for Economic Policy Analysis (CPB). Chapter 2 the relevance of this research to the organization is explained. Next, Chapter 3 looks at previous research that is similar to the current work. The technical work is divided into two parts. First, Chapter 4 details all work related to black-box machine learning. Then, in 5 explainability is introduced to open up this black box. The results from both parts are discussed in Chapter 6. Finally, Chapter 7 provides a summary and recommendations for future research.

2 Business Case

The Dutch Bureau for Economic Policy Analysis (CPB) operates as an independent research institute within the Ministry of Economic Affairs and Climate Policy. Despite being part of the ministry, CPB maintains full autonomy in its work. The organization conducts research both proactively and per requests from the government.

CPB has three primary goals: conducting research on the Dutch economy and socio-economic policies, translating scientific insights into daily policy practices, and providing official forecasts for the Dutch and global economies. Innovation within CPB is driven by the aim to serve the public good, contribute to the broader field of economic science, and enhance internal efficiency.

Before we consider the possible advantages and disadvantages of Machine Learning (ML) is good to first establish the definitions. ML methods provide a different approach to traditional economic modeling. In traditional economic modeling, the model is chosen based on economic theory. ML methods use a data-driven approach where an algorithm considers many different possible model structures and selects among them to maximize a performance criterion. There is some overlap between the methods. Some methods are used in both econometric sciences and computer sciences. For instance, linear regression, which is referred to as Ordinary Least Squares regression in the economic literature. Some attempts at applying ML in this domain have been done [Goulet Coulombe et al., 2022], [Milunovich, 2020], [Dubovik et al., 2022]. The contents of studies with similar applications of ML and their results will be discussed further in Chapter 3.

2.1 Advantages of Machine Learning

Even though these studies apply ML techniques, the suitability and applicability of ML are not widely understood among economists [Desai, 2023]. The incorporation of ML aligns with CPB's commitment to innovation, enabling the organization to stay at the forefront of economic analysis. By embracing ML, CPB can contribute to these relatively new techniques in economic science.

ML techniques have the potential to improve forecasting accuracy by leveraging non-linearity's [Goulet Coulombe et al., 2022]. ML takes a data-first approach so complex data structures can be uncovered without being specified in advance [Mullainathan and Spiess, 2017]. While some ML models assume feature independence, they generally handle multicollinearity better than traditional methods [Chan et al., 2022]. These desirable properties make ML models function well in environments where the relationships between variables are complex and non-linear. The economy consists of many different actors and to reason about it economists rely on highly simplified versions of the economy to reason about it [Granger, 2012]. It is exactly the sort of field where ML could provide additional methods to capture complexity.

Aside from potential performance benefits. The ML techniques could also provide more structure to the modeling process. Traditional approaches rely heavily on the forecaster's expertise. All design decisions are made by the forecaster as it is infeasible to construct all possible combinations of models to see which works the best. Machine learning techniques can test many different models very quickly and in a structured way, increasing the likelihood of identifying optimal solutions and improving reproducibility [Athey, 2018].

Within economics, there is a clear distinction between micro and macroeconomics. Microeconomics focuses on the behavior of individuals, and macroeconomics on the behavior in aggregate. The two seem closely linked in theory, but in practice, the relationship between the two fields is inconsistent [Weintraub, 1979]. While both could benefit from data-driven methods, the implementation for macroeconomics poses unique challenges due to the larger scale and fewer available models. Nevertheless, ML has been successfully applied to forecast various macroeconomic indicators such as GDP and unemployment rate [Richardson et al., 2018], [Scheer, 2022]. CPB has taken an interest in modeling the housing prices on a national level [Sinninghe Damste and Euwals, 2023].

The CPB employs a diverse array of models for their economic forecasts. The large structural models are designed to provide a long-term estimate that is in line with economic theory. Complementing these are the "zijlicht" models, designed to predict a specific macroeconomic variable with maximum accuracy. These smaller models serve as additional information for the human forecaster. There is no existing ”zijlicht” model for the housing prices. The housing market team at CPB specifically requested the development of such a model to address this gap. The extent to which ML can improve forecasting accuracy will be explored in Chapter 4.

Though this research focuses on house price forecasting, the insights gained from explainable ML applications hold the potential for enhancing forecasts of other macroeconomic indicators.

2.2 Risks of Machine Learning

Despite the promising advantages, the incorporation of ML at CPB introduces certain risks that warrant careful consideration.

The utilization of an ML model shares many of the inherent limitations associated with any forecasting model. Models, by nature, are simplifications of reality, leading to predictions that will never be perfectly accurate. Moreover, there exists a reciprocal relationship between forecasts and reality, where the act of forecasting can itself influence outcomes [Renfro, 2005] [Granger, 2012]. For example, if CPB predicts an increase in child poverty, the government may implement measures to mitigate it, resulting in an outcome different from the forecast. This phenomenon can also manifest as a self-fulfilling prophecy; if CPB predicts a surge in housing prices, buyers may rush to purchase homes before prices escalate further, thereby driving up demand and causing prices to rise. This complicates the evaluation of forecasting performance.

There are risks inherent to traditional and ML models, which ML models are more susceptible to due to their reliance on historical data. While both ML and traditional models utilize historical data, ML models lean more heavily on it as they lack a foundation of economic theory.

Then some risks apply to traditional models that ML models are more exposed to. There are specific drawbacks to using historical data to construct forecasting models. Both ML models and traditional models use historical data, but ML models are more reliant on it because they do not rely on economic theory as well. ML models are trained on historical economic data, which introduces inherent risk. The dynamic nature of the world means that past conditions may not perfectly align with future scenarios. This challenge is amplified in the housing market, where government regulations wield significant influence over market dynamics. Capturing the nuanced effects of past regulations poses a formidable task for ML models.

In addition, models trained on historical data are not expected to do well in crises. A crisis often arises from unforeseen changes in the broader environment, such as the COVID-19 pandemic or the war in Ukraine. These unprecedented changes are inherently difficult to predict, as they represent unique events in history. Even if an ML model were able to fully internalize each historic crisis and their consequences, a new unexpected event could still occur. Even though the accuracy of prediction models is lower during uncertain economic times, the models can still be useful [Overvest et al., 2024].

The use of complex non-linear models in general poses a risk to transparency and accountability of the results. If the inner workings of the models are not well understood, it becomes challenging to explain their predictions, hindering the ability of CPB to justify its analyses to the public.

Even when all patterns learned by the model can be accurately described, this does not guarantee that any causal relationships can be referred from it. An economist may trade the goal of accurate prediction for the unbiased estimate of a causal variable [Athey, 2018]. The only focus of an ML model is on prediction performance, not on causality. This shows in the flexibility of a model that may achieve the same prediction results with very different configurations[Athey, 2018] [Mullainathan and Spiess, 2017]. So a model may perform very well but may be unstable in the way it produces its output. This instability unfortunately also lowers users' trust in the model.

Although the list of challenges is extensive, it does not necessarily imply that risks outweigh opportunities. The opportunities are sufficient reasons to further develop the ML capabilities within CPB while being conscious of the mentioned challenges.

2.3 Explainable Machine Learning

Transparency has already been mentioned as a challenge to ML implementation. This challenge is the most prominent to CPB and could potentially be addressed with explainable machine learning. As a respected public institute, CPB holds the responsibility of adopting ML thoughtfully. The implementation of explainable machine learning fosters collaboration among researchers with diverse expertise, aligning seamlessly with CPB's commitment to maintaining high scientific standards. Additionally, the ability to articulate the reasoning behind predictions empowers CPB researchers to scrutinize and validate model outputs effectively. By embracing explainable machine learning, CPB positions itself to leverage the predictive power of ML while preserving transparency, and emphasizing the continued importance of human expertise in economic analysis. This strategic approach not only bolsters the reliability of CPB's forecasts but also reinforces its reputation as a responsible and innovative institution.

2.4 Requirements for the model

Transparency is paramount for the public to scrutinize CPB's insights and trust its findings. While black-box machine learning models may achieve high performance, their lack of transparency diminishes trust. Therefore, explainability techniques in machine learning are essential.

CPB's research holds scientific significance, particularly in economic science. Innovation in research techniques, including the application of ML, contributes to the advancement of economic science. ML techniques introduce the ability use larger data sets and model more complex relationships [Athey, 2018]. Experimentation and exploration of these possibilities in macroeconomic research can bring new insights and challenges to the forefront. Scientific advancement in economic research necessitates not only high-performing ML models but also explainable ones. Economic theory remains crucial, and explainable models provide insights into the complex patterns learned by the model, contributing to a deeper understanding of economic phenomena.

Researchers at CPB are primarily economists, doing scientific research in various subdomains of the Dutch economy. Employees are versatile, often working on multiple projects simultaneously. This means that research methods should apply to a wide range of research questions and problems. Researchers at CPB hold themselves to a high scientific standard. It is important for them that the methods they use have a strong mathematical foundation and are supported by scientific literature.

The explainable machine learning model resulting from this research will function as a so-called ”zijlicht”. During the projection process, a ”zijlicht” model should be able to provide the forecaster with more information than just the estimation for future house price levels. It should also provide information on what economic indicators have led to this prediction and why its prediction is different from the structural model. From this, we have distilled the following requirements for our explainable machine learning model. An explainable machine learning ”zijlicht” should:

- Be supported by scientific literature.
- Provide accurate projections for housing price levels
- Offer an explanation that allows users with economic expertise to critically evaluate the projections.
- Facilitate economic reasoning, ensuring that the model's interpretations align with economic principles and reasoning

3 Related Work

Forecasting housing prices is not a new objective in economic research, neither is the use of machine learning (ML) for forecasting purposes. This chapter explores previous works that share similarities with this thesis.

3.1 Traditional Economic Forecasting

Econometric forecasting is a large area of research. The scope of this discussion is limited to simple techniques. In macroeconomic forecasting, the objective is to formulate an equation that aligns with economic theory while offering predictive capabilities. Traditional economic forecasting models, also called structural models, commonly incorporate autoregressive (AR) terms to integrate historical data into their predictions [Hamilton, 1994]. Simple AR models can be extended in various ways to reflect economic theory. These statistical approaches typically require prior knowledge about the data distribution for constructing predictive models. Employing such parametric methods demands a deep understanding of mathematical concepts and substantial technical expertise for establishing the model's parameters [Parmezan et al., 2019].

One of the possible model types in econometric forecasting is the Error Correction Model (ECM). The classic formulation involves establishing a long-term relationship between a dependent variable, its lagged values, and other independent variables [Malpezzi, 1999]. In this use case that would be a relationship between housing prices in the next quarter that is dependent on historical housing prices and some other economic indicators. The ECM is a restricted linear regression model. The restriction is that the outcome is guided back towards a desired value [Alogoskoufis and Smith, 1991]. The coefficients of these models are optimized using the Ordinary Least Squares method, known in ML as linear regression.

The housing-price-model is the structural model that is currently used by the Centraal Planbureau (CPB) model to predict housing prices. The model focuses on the relationship between income, interest rates, and housing prices [Sinninghe Damste and Euwals, 2023]. It incorporates an error correction mechanism, addressing short-term shocks and ensuring the long-term equilibrium between household income and housing cost. The change in housing prices for the year t, ΔP_t, is calculated based on the change in the previous year ΔP_{t-1}. Other variables included in the equation are the percentage change in real household income ΔI_t, the change in the real cost of use ΔFGK_t, a term to incorporate the long-term financing cost $dLTW_{t-1}$ and finally a seasonal component S_t. The coefficients β are determined with an Ordinary Least Squares regression, also known as linear regression.

$$\Delta P_t = \beta_1 \Delta P_{t-1} + \beta_2 \Delta I_t + \beta_3 \Delta FGK_t + \beta_4 dLTW_{t-1} + \beta_5 S_t$$

This model draws data from CBS, DNB, HDN and internal forecasts [Sinninghe Damste and Euwals, 2023]. CPB has expressed the intent to exclusively rely on open-source data from CBS for the current project. Consequently, our model will be trained on a different dataset than the benchmark.

There are important differences between traditional forecasting methods and ML. Unlike traditional modeling approaches, ML adopts a data-first approach, making fewer assumptions about the data distribution. The success of ML lies in its capacity to generalize complex patterns not specified in advance [Mullainathan and Spiess, 2017]. Traditional models are built based on economic theory which offers significant advantages. They are more suited to causal reasoning, contrasting with ML models that prioritize performance optimization without necessarily considering causal relationships [Athey, 2018]. This emphasis on causation allows traditional models to offer deeper insights into the underlying mechanisms driving economic phenomena. A model with more structure can also be more resistant to noisy data during training. However, there is a clear trade-off here between resistance to noise and an ability to pick up on new patterns. Another disadvantage becomes apparent when real-world dynamics deviate significantly from assumed linear relationships, leading to diminished performance. Finally, these models heavily rely on the domain expertise of the forecaster [Parmezan et al., 2019].

3.2 Machine Learning in Macroeconomic Forecasting

The application of different ML models in time series forecasting has been explored in various studies [Ahmed et al., 2010] [Corani et al., 2021]. Unfortunately, the application of ML application in the economic domain is still limited. This section provides an overview of studies using ML methods in macroeconomic forecasting. Milunovich [2020] compared traditional and ML techniques for forecasting Australia's real house price index. Interestingly, the results showcased that ML models outperformed their traditional counterparts [Milunovich, 2020]. Some other studies support similar findings. Stamer [2021] applied an ML method to predict trade flows and found that this method improves upon a traditional baseline. This study has limited comparability to the current work as it uses only one ML method and applies this to monthly data. Monthly data allows for many more data points while spanning the same time frame, so these data sets are typically bigger. More relevant is the work by Richardson et al. [2018]. They used ML methods for nowcasting New Zealand's GDP. Typically GDP metrics are published with a delay. Nowcasting is used to estimate the current value of an economic indicator based on historic values. In this study, ML models were also found to outperform traditional models. Additionally, [Richardson et al., 2018] found that forecast combinations provided significant value. However, the literature presents a mixed perspective on the effectiveness of ML methods. In experiments conducted by [Dubovik et al., 2022], ML methods were used to predict trade flow on a monthly frequency. The results show that ML models could not surpass the traditional Bayesian VAR. Interestingly, Dubovik et al. [2022] experimented with combining outcomes from both types of models through averaging, resulting in improved performance over the traditional Bayesian VAR [Dubovik et al., 2022].

Taking a slightly different approach, Chong et al. [2017] applied a deep neural network (DNN) to the residuals of a traditional model. This unique strategy demonstrated that the DNN could enhance predictions by adding valuable information to the residuals [Chong et al., 2017].

These studies collectively underscore the potential of ML in economic forecasting while highlighting variations in performance across different models and scenarios.

3.3 Explainability in Macroeconomic Forecasting

The studies discussed earlier primarily employed non-linear ML methods. These models are considered black-box due to their complex internal structures. Notably, these models showcased promising results in terms of forecasting accuracy. However, an important aspect that has not been extensively addressed in the literature is the explainability of these models. Their lack of transparency raises concerns regarding their reliability and potential biases. Explainability remains a crucial element in macroeconomic forecasting, offering transparency and insights into the rationale behind predictions. Traditional modeling methods have an inherent advantage in this regard, as the economic reasoning is embedded within the model during its construction. The lack of explicit mention of explainability methods in the context of ML models for macroeconomic forecasting highlights a gap in current research. While these advanced models offer impressive predictive capabilities, their adoption into practical forecasting frameworks may be hindered without effective strategies for interpreting and validating their predictions.

4 Forecasting with Machine Learning

This chapter focuses on the first technical part of the research and addresses the first hypothesis: ”A black-box machine learning model can be used to accurately predict housing prices.” First, the choices made during the construction of the model are motivated in Section 4.1. The exact methodology that takes the data from raw input to predictions is described in Section 4.2 and then finally the results of the model are shown in Section 4.3.

4.1 Theoretical Framework

Solving a machine learning problem extends beyond merely applying a model; it involves the orchestration of a complete machine learning pipeline that transforms raw input into meaningful predictions. This process contains numerous design decisions. The following section explores the different steps of the machine learning pipeline and lays the theoretical foundations to support the design decisions that have been made.

4.1.1 Problem Definition

Various subcategories of problems are defined within the ML field. The types of models that are available depend on the type of problem at hand. One fundamental distinction is between supervised and unsupervised techniques. In unsupervised learning, the variable of interest is not provided, and the goal may involve tasks such as clustering similar examples or estimating data distribution within the input space. On the other hand, supervised learning deals with situations where the variable of interest is available [Bishop, 2006]. While any problem could be framed as an unsupervised problem, our objective is better suited to supervised techniques. We have access to historical data on housing prices, which serves as the target or dependent variable during the training phase. Lagged information on housing prices and other macroeconomic indicators can serve as input data or independent variables. Both the dependent and independent variables are continuous, marking this as a supervised regression problem.

Predicting housing prices introduces an additional layer of complexity as it involves time series analysis. A time series is a set of discrete observations over time. For economic predictions, these observations are usually made on a yearly, quarterly, or monthly basis. In our case, measurements for both the dependent and independent variables are available for every quarter. The goal of time series analysis is to find a pattern in the data and predict future values of the time series [Parmezan et al., 2019]. This type of analysis is distinct from regular supervised regression because data points are dependent on time. This presents complications for some conventional ML techniques when those assume independent samples. The following subsections address this challenge whenever necessary.

A final intriguing aspect of this ML problem lies in the number of outputs. When making economic projections we are expected to predict housing prices for eight quarters into the future. That makes this task a multi-horizon forecasting problem. One approach to deal with this is to use a single model that uses its own predictions as input for the next quarter. This recursive approach carries the risk of propagating any errors made in earlier predictions [Ben Taieb et al., 2012]. Alternatively, a single model can be designed to produce multiple outputs, but this approach significantly limits the choice of models. Our chosen strategy is to train eight distinct machine learning models, each dedicated to forecasting for a specific horizon. This approach involves more training time, but with a small data set it remains feasible.

4.1.2 Feature Engineering

The first step in an ML pipeline is feature engineering. It is the process of transforming raw input into usable features. The right features make the learning process easier for a model, making it a crucial step in the pipeline [Zheng and Casari, 2018]. This section explains the different transformations applied to the input data to construct new features.

Section 3.1 mentions the use of autoregressive (AR) terms in economic forecasting. These terms, also described as lags introduce are values of a feature in a previous time period. The introduction of lags on the target or other explanatory variables provides the model with additional information. A model could use many lagged features, each one adding additional historical information. The introduction of this feature does reduce the number of available samples in a data set. So it is not always beneficial to add more lags. Section 4.1.3 introduces techniques to limit the number of variables used in the model.

In addition to adding new variables, transformations on input data can enhance prediction accuracy [Johnson, 2019]. The housing market prices have a rising trend [Sinninghe Damste and Euwals, 2023]. Predicting future values beyond the historical range seen in training is a challenge for ML algorithms. It is therefore easier to predict the mutation of the target variable instead of the level. This is also the custom in traditional models for prediction of housing prices [Luth, 2023], [Malpezzi, 1999], [Boelhouwer et al., 2001], [Sinninghe Damste and Euwals, 2023]. Explanatory variables can also have rising trends, so using the change in this variable as a feature can make prediction easier as well. Taking the mutation of a variable has the additional benefit of reducing dependency between data points.

Another transformation is to correct variables for inflation. The general price level is naturally correlated with housing prices as well as other economic indicators. Correcting for inflation allows us to focus on the housing market specifically, making the prediction task simpler.

Lastly, the application of moving averages emerges in literature as a valuable transformation. The procedure helps to smooth input data and mitigate the impact of outliers [Goulet Coulombe et al., 2021]. Consequently, we incorporate this transformation into our process.

In summary, the different feature engineering procedures we have discussed increase the number of features that are in consideration for our model. The following section motivates the need for feature selection and outlines a procedure for it.

4.1.3 Feature Selection

In macroeconomic forecasting, the amount of data points available is usually limited. As the number of features grows, the amount of data required to find patterns in these dimensions increases exponentially [Trunk, 1979]. After applying the feature engineering procedures we discussed in the previous section, the number of features to consider is too large compared to the number of data points. It becomes necessary to apply dimensionality reduction.

Dimensionality reduction techniques can be divided into two approaches. Feature extraction projects the original data into a lower-dimensional space, creating new features. This method allows for great information retention, but the new features are usually uninterpretable. Given our emphasis on model interpretability, feature selection would be the more suitable approach. Feature selection takes a subset of features to use in the model. It is an approach that results in information loss, but this is a downside we accept in favor of interpretability.

The challenges in the implementation of feature selection are twofold. Firstly, most methods require specifying the number of selected features. Having too many features can introduce noise, redundancy, and irrelevant information. Select too few features and risk excluding relevant information. The optimal number of features will change for every ML task [Li et al., 2018]. Considering all possible features would grow the search space exponentially [Vergara and Estévez, 2014]. The selection method we present is only one of the possible techniques to apply feature selection.

Mutual information maximization (MIM) efficiently assesses feature importance based on its correlation with the target, providing a swift form of feature selection [Li et al., 2018]. However, MIM evaluates feature scores individually. This disregards any correlation between features that make them redundant. The method's limitation lies in assuming feature independence. In practice, features should correlate with the target without being highly correlated with each other. Section 4.2.1 covers an implementation of MIM where this problem is addressed.

4.1.4 Machine Learning Models

The diverse nature of data and varying complexities of real-world problems have resulted in the development of many different types of machine learning models. Each type of model has its strengths, limitations, and suitability for different tasks. The following types of models can all be used for time series prediction.

Linear Models The output of linear models is defined as a linear combination of input features [Bishop, 2006]. The model is also known as Ordinary Least Squares Regression (OLS) in the economic literature. While its simplicity and interpretability are advantageous, Linear Regression models are not able to capture nonlinear relationships.

Nearest Neighbors These models work based on the assumption that the most similar historic data points can serve as a meaningful prediction for the future. K-Nearest Neighbors (KNN) offers predictions by averaging the values of the k number of data points in the training data that are the closest to the new instance [Bishop, 2006]. The use of distance metrics makes the algorithm sensitive to feature scaling and outliers [Johnson, 2019], this is something to take into account during feature engineering.

Support Vector Machines The Support Vector Machine (SVM) is another distance-based model. Models of this type can handle classification and regression tasks [Müller et al., 1997]. It maps data points into a higher-dimensional space, seeking the optimal hyperplane for class separation or continuous value prediction [Cortes and Vapnik, 1995]. While SVM inherently models linear decision boundaries, applying kernels allows for more intricate boundary formulations [G. Brereton and R. Lloyd, 2010].

Tree-based Approaches Tree-based models are algorithms that make decisions based on hierarchical structures [Breiman, 2001]. A regular decision tree creates a structure of decision points that can be followed to reach a prediction. A Random Forest creates many different trees and takes their average prediction. These trees should be randomized and diverse so that they can compensate each others mistakes [Breiman, 2001]. Other examples of tree-based approaches are the CART Regression tree, Gradient Boosting Machine, and eXtreme Gradient Boosting.

Stochastic Models Stochastic models use probability theory and statistical distributions to make predictions [Rasmussen et al., 2004]. These models are designed to capture uncertainty in the data. The Gaussian Process is a stochastic model that assumes a joined Gaussian distribution for its input variables. Instead of taking the input variables as they are, the input is sampled from the distribution to capture a level of randomness [Rasmussen et al., 2004].

Neural Networks Neural networks are a type of machine learning model inspired by the functioning of the human brain [Hopfield, 1982]. The model consists of interconnected nodes (neurons) organized in layers. The individual nodes can turn on or off, based on their input. With enough nodes and layers neural networks are capable of learning complex patterns and relationships [Abiodun et al., 2018]. The training procedure of a neural network relies on an extensive amount of training data, which usually is not available in macroeconomic forecasting. Given the constraints of available data, our focus will be directed towards exploring other methods.

The literature does not agree on which type of model is the most effective to use for macroeconomic forecasts. For instance, Milunovich [2020] found the SVM to outperform other models in most scenarios, whereas [Ahmed et al., 2010] found this model to be outperformed by both KNN and CART regression trees. This lack of consensus shows that the most effective model differs depending on the use case. Consequently, our approach involves testing a diverse set of ML models.

4.1.5 Hyperparameter Optimisation

Each model we discussed in the previous subsection contains hyperparameters that determine the structure of the model. For instance, in the KNN algorithm, the number of neighbors k to consider is a hyperparameter. The optimal values for hyperparameters differ from one prediction problem to the next. Therefore, tuning the hyperparameters is essential to find an optimal or near-optimal configuration for each model.

Hyperparameter tuning can be achieved through various methods. In manual search, the hyperparameters are chosen using the researcher's expertise. This is a quick procedure, but difficult to reproduce and the effectiveness is heavily dependent on the expertise of the researcher. Grid search involves defining a finite set of possible values for each hyperparameter, with the search algorithm exploring every combination. In contrast, random search tests random configurations until a predefined budget is exhausted. Random search tends to outperform grid search when certain hyperparameters have a disproportionate impact on model performance [Bergstra and Bengio, 2012]. Random search is still very inefficient when compared to more complex methods such as genetic algorithms or Baysian optimisation Hutter et al. [2019]. However, our primary focus is not on finding the optimal hyperparameters. Instead, our goal is to build a well-performing machine learning model within a constrained time frame. Random search is implemented within the CARET package, making it the preferred method for this pipeline.

In tuning the hyperparameters of a model there is the risk of overfitting. Overfitting occurs when a model fits the training data too closely, making it challenging to generalize to new data. To guard against overfitting performance is measured on previously unseen data and hyperparameter tuning is done using different data again. The data that is set aside for testing and tuning can no longer be used in the training process. Cross-validation helps mitigate the loss of information. In cross-validation, the dataset is split into non-overlapping training and validation sets, with the model trained on the former and evaluated on the latter. This process is repeated multiple times to obtain an average performance score [Berrar, 2018].

A fundamental assumption of cross-validation is that data points are independent and identically distributed (i.i.d) [Arlot and Celisse, 2010]. Unfortunately, this assumption is violated by time series data. Burman et al. [1994] proposes a blocked method of cross-validation where data is divided into time-based blocks. Data points that are dependent on points in another block are removed. This eliminates the dependency between the test and training set but also reduces the number of samples available for training. In their experiments with cross-validation on time series data Bergmeir and Benítez [2012] did not find any practical consequences of the violation of the independence assumption. Goulet Coulombe et al. [2022] also recommend the use of standard k-fold cross-validation. Given the limited number of data points at our disposal, we have opted for standard k-fold cross-validation.

4.1.6 Performance Evaluation

After the model has been trained, its performance needs to be evaluated. Various metrics are available to measure model performance. In this work, we will use the Mean Absolute Percentage Error (MAPE) and the Root Mean Squared Error (RMSE). The MAPE provides a relative measure of error that is intuitive to interpret. The MAPE calculates the absolute difference between prediction y_i and actual value $\hat{y}_i$. This is divided by the actual value $\hat{y}$ to get the error as a percentage of the actual value. The average across all N samples gives the MAPE.

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{\hat{y}_i}$$

The Mean Absolute Percentage Error (MAPE) has notable limitations, with one of the most significant being its sensitivity to the scale of the actual values. Specifically, an error on a large actual value is considered smaller than the same absolute error on a smaller actual value Makridakis [1993]. Given the upward trend observed in housing prices, this characteristic of MAPE could potentially pose a problem when evaluating forecasting accuracy.

The Root Mean Squared Error is one of the most popular error measures to use [Botchkarev, 2019]. The difference between the prediction y_i and actual value $\hat{y}_i$ is squared. The root of the sum of squared errors across all N samples is then taken. This has as advantage that the error is expressed on the same scale as the target variable. The formula is defined as follows.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{N}}$$

The RMSE gives a larger weight to predictions with a larger absolute error. The metric is not without controversy. Some researchers argue that its sensitivity to outliers makes the RMSE an unreliable error metric [Armstrong and Collopy, 1992] [Willmott and Matsuura, 2005]. Others still view it as a useful method [Chai and Draxler, 2014] [Shcherbakov et al., 2013].

Every error metric has its limitations, and theoretically, there is no universally ”best” metric [Makridakis, 1993]. To offer a comprehensive assessment of model performance, we report on two different types of error metrics. However, during hyperparameter optimization, we can only optimize for one metric. In this case, we prioritize the Root Mean Square Error (RMSE).

4.2 Methodology

As demonstrated in Section 4.1, the ML pipeline involves several distinct steps. This section details how the available data is processed and used to train ML models. By providing a comprehensive overview of these procedures, this section aims to offer clarity on the methodology employed in preparing the data and developing the predictive models.

4.2.1 Data

Data Sources The data utilized in this project is provided by CPB. The input for the pipeline is derived from multiple sources. All original series are taken from CBS but some have been cleaned and modified by the housing market team to fit preceding projects. The first is the data set that is used as input for this project is the one used to train the existing housing-price-model [Sinninghe Damste and Euwals, 2023]. Additional features that were considered for the housing-price-model are included as a second manually selected data set. Finally, CPB's NLdata, a recently developed database, contains over 1100 time series related to the Dutch economy from various sources like CBS and the ECB. Variables from this database are taken directly from their original source. The NLdata is a very large database in which most variables are not related to the housing market. To control the number of variables that we consider for the model, we do not consider feature transformations for the variables from NLdata.

Feature Engineering Given the large amount of features in the NLdata database, feature engineering is only applied to the manually selected data sets. The features that result from applying these transformations are considered in the feature selection procedure. Here, the following transformations are performed:

- **lags** Incorporate historical data points as new features.
- **Moving average** Calculate the average of preceding data points, with the number of considered data points determined by a predefined window size.
- **Correction for inflation** Transform a feature by dividing it by the Consumer Price Index (CPI).
- **Mutation** Take the ratio between the preceding feature value and the current feature value.

The objective of the ML model is to predict the housing market price level, sourced from the CPB dataset. Information about the target variable is summarized in Table 1. To ease prediction for the ML model we apply several transformations and use the mutation in real housing prices as a target.

Name	prijsindex_best_koopwon
Description	Nominal price level of sold houses.
Datatype	Continuous
Available time period	1995Q2 - 2023Q3
Sampling rate	Quarterly
Number of values	114

Table 1: Information on the target variable before feature engineering has been applied.

As mentioned in Section 4.1.1 we have opted to train a separate model for each forecasting horizon. The target of this model then becomes the change in housing prices from the last known year. For example, a submodel predicting $t + 2$ predicts the change in housing prices between 2023Q1 and 2023Q3. The next submodel for $t + 3$ predicts the change in housing prices between 2023Q1 and 2023Q4.

Feature Selection For feature selection, a variation on mutual information maximization is employed. Initially, the mutual information between the features and the target variable is computed. The top N features are then selected based on their scores. Since this process solely considers individual features without accounting for interactions, an additional step is taken. The correlation matrix of the top N features is calculated and features that have a correlation higher than 0.8 with another selected feature are eliminated. This iterative process continues N features are obtained.

While this approach does not guarantee the optimal set of features, it aims to minimize redundant information and assess whether features provide information about the target. Still, a better-performing feature set could exist. This approach avoids the need to train models for feature selection. Macroeconomic datasets are usually small, with few samples, making the training process quicker. Yet there are many possible features to use, making it infeasible to test each possible combination of features.

Following this first feature selection process, the final features are determined through expert input. Experts used their judgment to identify complementary variables, leading to the removal of some and the inclusion of others. The choice of the number of features to select remains arbitrary. Due to the limited length of the dataset, the number of features has to be limited. The final feature set is summarized in Table 2. A sample of the data set that is used for training is shown in Table 3.

Feature Scaling Standardizing data to a common scale across all features is necessary for distance-based learners like KNN and can also aid other models by promoting faster convergence. Given our limited number of data points, simplifying the process for the models is essential. The data X is standardized so we compute the difference between the feature value x and its mean μ_x. This is divided by the standard deviation σ_x

$$x' = \frac{x - \mu_x}{\sigma_x}$$

The mean and standard deviation are calculated based on the training set. Data in the test set should not influence the procedures performed on the training data, this prevents information leakage.

Name	Description
ink_nom_niv	Nominal income level
Prijs_re_mut	Real price mutation
Prijs_re_mut_avg8	Moving average of real price mutation, window size is 8 quarters
Consumenten_vertrouwen_mut_avg4	Moving average of change in consumer trust, window size is 4 quarters
BrutoInvesteringen	Gross investments in residential properties.
BBP	Gross Domestic Product (GDP)
AantalVerkochteWoningen	Number of houses sold

Table 2: States the name of each feature that is used in the ML models and provides a short description.

ink_re _niv	prijs_re _mut	prijs_re _mut_avg8	r_nom _niv_avg4	consumenten _vertrouwen _avg4	Bruto Investeringen	BBP	Aantal Verkochte Woningen
38915	1,01	1,01	7,59	10,50	4264	73658	37054
39268	1,03	1,02	7,41	11,67	4476	75571	40498
39628	1,00	1,01	7,22	11,50	4460	76210	46282
38500	1,02	1,02	6,85	8,00	3761	76308	35669
38948	1,03	1,02	6,64	6,75	4850	77197	43810
39293	1,03	1,02	6,46	6,25	5126	78420	46582

Table 3: Shows the first rows of the training set. The target y is the change in real house price for one quarter ahead. The column *periode* is used as an identifier.

4.2.2 Model Training

Train, test and validation sets To protect a model from overfitting the performance should be evaluated on different data than what the model is trained on. The hyperparameters are tuned on different data again. To get an accurate and unbiased measure of the performance, the test set should be as large as possible. At the same time, the performance of a model also depends on having enough data to train, presenting a trade-off. Any data that you use for training, can no longer be used for testing and visa versa. Given the small number of samples we have at our disposal, the division of training and test set requires careful consideration.

In our use case, the model will be deployed to make projections from the final input date forward. As explained in Section 4.2.1, a different target is created for each forecasting horizon. Eight datasets are created, each corresponding to a specific horizon. To simulate the model in production, it makes sense to designate the final eight data points as the test set. However, judging performance solely on this single projection of eight quarters would be unreliable. The chosen time period for the test set could be arbitrarily easy or difficult to predict, leading to biased performance evaluation.

When the target reflects the price change further into the future there is less data available. So the data set with targets for predicting one quarter ahead has less data points than the one with a target eight quarters ahead. This phenomenon is visualized in Figure 1. The test set is used for demonstration purposes and not to report on performance so the loss in data points is absorbed in the test set. This way, the training set remains as large as possible.

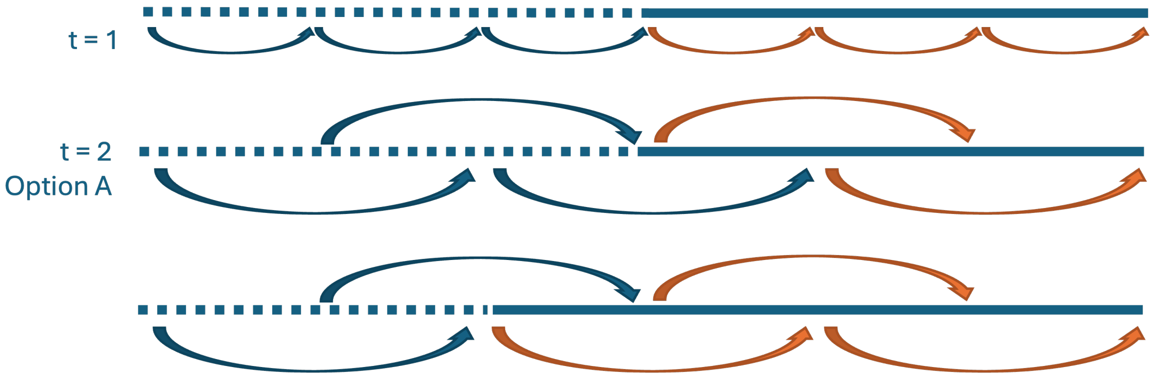
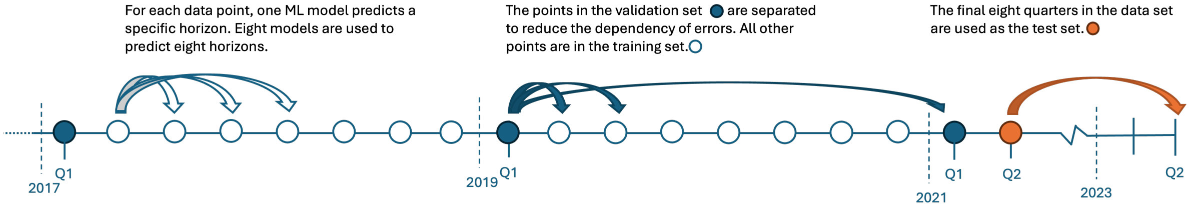


Figure 1: Shows the amount of samples that fit into the dataset. Each arrow represent a sample of change in housing price from $t = 0$ to $t = 1$ or $t = 2$. The dotted line represents the training set and the solid line the test set. A data set with $t = 2$ as a target can fit one less data point. The test set can be decreased in size (option A) or the training set (Option B). We have chosen to implement option A.

A separate validation set is introduced to allow for an unbiased performance valuation, not to be confused with the one used for hyperparameter optimization. This set should have diverse points spread over time, preventing the selection of overly easy or challenging forecasting periods. For that, the errors on the predictions should be independent. Unlike cross-validation, there's no need to rotate the split between the training and validation sets. Therefore, we've implemented a different procedure, outlined as follows. As our projection is for eight quarters, the final eight data points are reserved for the test set. The remaining data is divided into blocks of eight data points. Any data points at the end of the data set that do not fit into a block are in the training set. A prediction of housing price change over eight quarters creates a dependency between the prediction errors made on this sample and the 7 samples preceding it. For this reason, the final sample in each block is used for the validation set while the others are used for training. That way, there is no dependency between the errors on the different samples in the validation set. The division between the training, validation, and test set is also explained in Figure 2.

The decision to use non-overlapping prediction periods in the validation set limits the number of points in the validation set. With an eight-quarter horizon, only a limited number of intervals can fit within a dataset spanning from 1995 to 2023. Consequently, the validation set comprises thirteen points, which may not be sufficient to ensure a robust performance estimate. Nevertheless, it is the maximum amount we could use given the total number of samples available. The number of data points in each set is summarized in Table 4.

Model selection Section 4.1.4 covers several broad categories of ML models. In practice, there exist numerous variations within these categories. For ease of implementation, the CARET package was utilized. This package consolidates numerous implementation packages from various makers into a unified interface. The package offers access to 238 unique models. To cover the broad categories



For each data point, one ML model predicts a specific horizon. Eight models are used to predict eight horizons.

The points in the validation set ● are separated to reduce the dependency of errors. All other points are in the training set.○

The final eight quarters in the data set are used as the test set.●

Figure 2: Shows how the training set, validation set, and test set are split. Each dot is a point in the data set. Based on this point the change in real housing prices is predicted for eight quarters into the future. To predict eight quarters ahead, eight ML models are used, indicated with the arrows. The data points from 2021Q2 till 2023Q2 are used for the test set. When predicting one quarter ahead, it is possible to fit more data points within this time frame than when predicting eight quarters ahead. The remaining data is divided into blocks of eight data points. The first point of each block is in the validation set. Any remaining data points that are not able to fit in a block are put in the training data set.

Horizon	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Training set	92	92	92	92	92	92	92	92
Validation set	13	13	13	13	13	13	13	13
Test set	8	7	6	5	4	3	2	1
Total	113	112	111	110	109	108	107	106

Table 4: Shows the number of data points in each data set. When predicting the price change further into the future, the number of available data points becomes smaller and the size of the test set decreases.

discussed in the theoretical framework, one model of each category is implemented. The small data set makes it difficult to effectively tune a large number of hyperparameters. So in cases where multiple comparable implementations are available, the number of hyperparameters is used to select a model. The implemented models are summarized in Table 4.2.2.

Model	**Category**	**Identifier**
Linear Regression	Linear Model	lm
k-Nearest Neighbors	Nearest Neighbors	knn
Gaussian Process with Polynomial Kernel	Stochastic Model	gaussrPoly
Support Vector Machines with Radial Basis Function Kernel	Support Vector Machine	svmRadial
Random Forest	Tree-based Approache	rf

Table 5: Gives an overview of the ML models used for forecasting. The category refers to the type of ML models described in section 4.1.4. The identifier is the value used to call the method in the CARET package.

Hyperparameter Optimisation To optimize the hyperparameters of every model, we employ standard 10-fold cross-validation and random search, as previously motivated in Section 4.1.5. The CARET library provides a function to generate random tuning parameter combinations. The specific hyperparameters tuned for each model are detailed in Table 4.2.2.

Identifier	**Hyperparameter**	**Description**
lm	intercept	Mean value of the output when all input variables are equal to zero
knn	k	Number of neighbors to consider
gaussrPoly	degree scale	The degree of the polynomial Scale is used to normalize data
svmRadial	sigma C	Kernel width, determines which points are considered similar Regularization parameter
rf	mtry	At each split in a decision tree a random number of features is sampled as a candidate feature to split on. The number of features that is sampled is the mtry hyperparameter.

Table 6: Describes the hyperparameters that are tuned for each model. Which hyperparameters are tuned is defined by their CARET implementation.

Evaluation When evaluating model performance the MAPE and RMSE are reported. The choice for these metrics has been explained in Section 4.1.6. A proper valuation of ML models benefits from a comparison to a traditional economic model. The housing-price model introduced in Section 3.1 is implemented as a baseline. There is one important issue with this baseline. The model uses projected interest rates in the predictions. Historic projections are not available so the model uses actual interest rates for training and prediction. This could make the baseline perform unrealistically well. That should be taken into account when comparing performances.

4.3 Results

This section presents an overview of the results. In forecasting a distinction is made between in-sample predictions and out-of-sample predictions. In-sample predictions are made using data used for model training, while out-of-sample predictions use unseen data. Both the validation and test sets in this project are out-of-sample. In-sample analysis assesses how well the model fits the training data, while out-of-sample analysis predicts model performance in new situations [Tashman, 2000]. Comparing the model's performance for in-sample and out-of-sample predictions offers insights into the amount of overfitting experienced by the model.

4.3.1 Out-of-sample analysis

The performance of the models on the validation set is shown in Figure 3. Figure 4 illustrates how the error on the validation set increases as the horizon becomes larger. This aligns with expectations, as predicting housing prices further into the future becomes more challenging. The baseline model follows a different path. Unlike the other models, the baseline model utilizes its prediction for the preceding quarter as input to forecast the subsequent one. While this approach offers an advantage if the initial prediction is accurate, errors may propagate in subsequent forecasts.

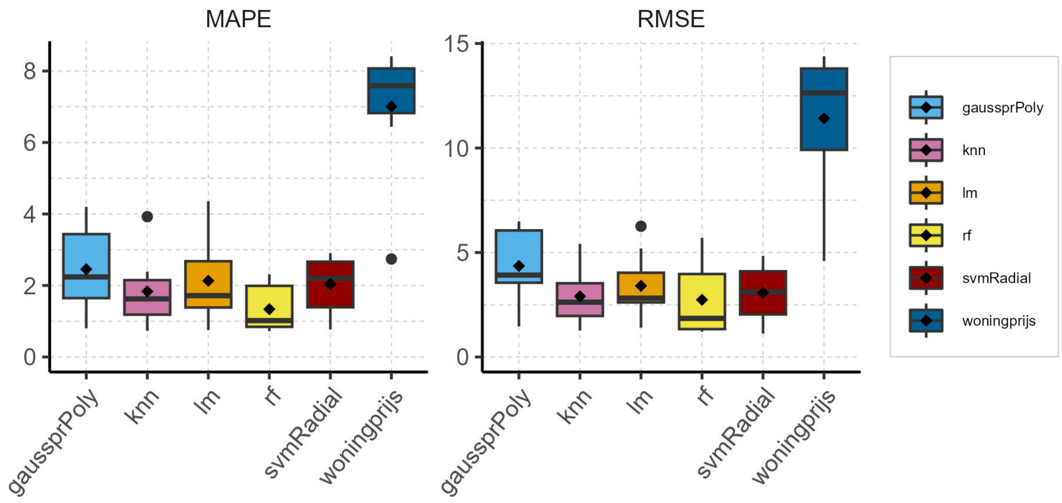


Figure 3: RMSE and MAPE on out-of-sample validation sets. The ML models all outperform the traditional housing price model (woningprijs). The best performing ML model is the random forest (RF).

The performance of these models is also described in table 7. The choice of the best-performing model depends on the preferred error metric and the prediction horizon. It becomes clear that the ML methods perform better than the traditional economic model (woningprijs). Out of the ML models, Random Forest (RF) outperforms the others.

Figure 5 shows the predictions made on the validation set by the RF. These are predictions for the first quarter of each year, looking eight quarters into the future. The RF predictions are close to the actual house prices and can even anticipate turning points. Only the drop in housing prices in 2022 is not picked up on.

Figure 6 displays the out-of-sample predictions on the test set, showing a single projection from 2021Q2 to 2023Q2. There is a clear difference between the predictions of the ML models and the structural housing-price-model (woningprijs). The latter provides a much more conservative forecast which turns out to be more accurate.

4.3.2 In-sample analysis

Figure 7 shows the performance on the training set. The RF, KNN and SVM stand out from the other models with lower errors. The RMSE and MAPE on the training set are much lower than on the validation set. This indicates that the models are overfitting.

Figure 8 illustrate how the Linear Regression (LR) and RF models fit to the actual housing price levels on the training set. Comparing these figures reveals that the RF model fits more closely to

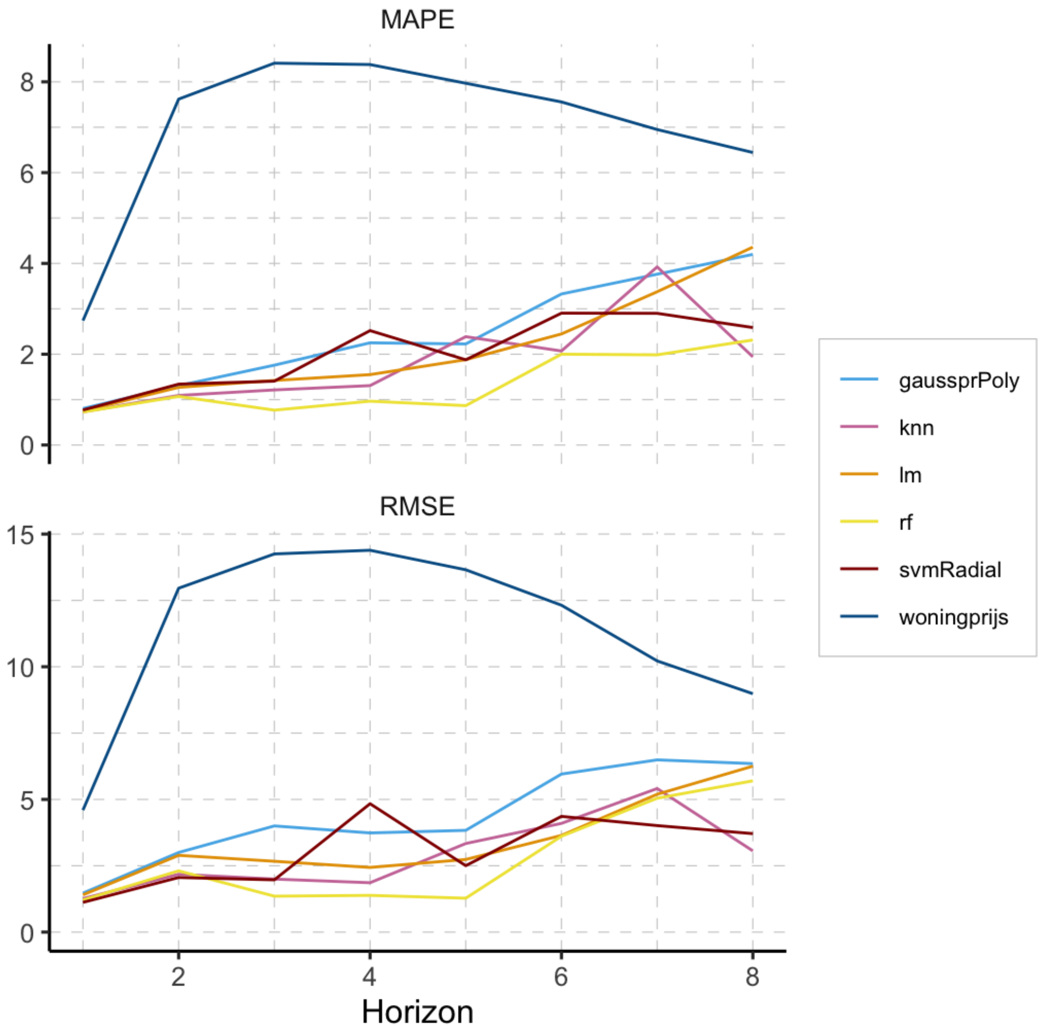


Figure 4: RMSE and MAPE on out-of-sample validation sets for one till eight quarters ahead prediction. As the models forecast further into the future, the errors generally increase, with the exception of the baseline (woningprijs). The baseline utilizes a single model that takes its previous period's prediction as input instead of employing separate models for each prediction horizon.

the training data than the LR.

In addition, this Figure sheds light on why all ML models predict growing housing prices on the test set. Most historical data points have shown rising housing prices. Consequently, an overfitted model maximizes its performance on the training data by predicting rising housing prices. However, in this case, it comes at the detriment of performance on the test set, where the realizations differ from the trend.

RMSE										
Model	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Mean**	**Sd**
RF	1,212	2,309	**1,354**	**1,385**	**1,278**	**3,613**	5,045	5,701	**2,74**	1,82
SVM	**1,120**	**2,058**	1,969	4,838	3,503	4,361	**4,014**	**3,711**	3,07	**1,33**
GP	1,467	3,000	4,001	3,737	3,835	5,956	6,491	6,349	4,35	1,77
LR	1,404	2,893	2,669	2,438	2,737	3,644	5,194	6,254	3,40	1,58
KNN	1,263	2,180	1,998	1,861	3,338	4,098	5,410	3,061	2,90	1,37
Baseline	4,597	12,960	14,251	14,390	13,654	12,319	10,220	8,983	11,40	2,36
MAPE										
Model	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**Mean**	**Sd**
RF	**0,726**	**1,070**	**0,768**	**0,965**	**0,867**	**2,000**	**1,986**	**2,124**	**1,34**	**0,65**
SVM	0,771	1,341	1,341	2,520	1,877	2,904	2,901	2,587	2,04	0,81
GP	0,803	1,308	1,308	2,250	2,226	3,327	3,762	4,199	2,45	1,20
LR	0,752	1,270	1,422	1,551	2,386	2,445	3,756	4,359	2,13	1,20
KNN	0,731	1,092	1,213	1,301	1,880	2,071	3,923	1,949	1,83	1,01
Baseline	2,720	7,619	8,411	8,381	7,969	7,557	6,949	6,443	7,01	1,85

Table 7: MAPE and RMSE on validation set for all models predicting 1 till 8 quarters ahead. When looking at the MAPE, the random forest (RF) scores best. Considering the RMSE, the Support Vector Machine (SVM) is the best performing on the first two prediction horizons. Overall the RF seems to be the best performing ML model.

4.4 Conclusion

In conclusion, our analysis has shown that machine learning models can effectively predict housing price levels, as demonstrated by their strong performance on the validation set. However, the Gaussian Process and Linear Regression models have proven to be less effective compared to other models in this context. There is a large contrast in performance on the training set and validation set. This is an indication that the models are overfitting on the training data. The errors in the training data are then minimized at the expense of performance on the validation set. Overfitting is a common issue across all models, particularly pronounced in the Random Forest and Support Vector Machine. Furthermore, on the test set the models struggle to identify the turning point in the time series. To gain deeper insights into the workings of the black-box models, the next chapter involves applying a post-hoc explainability method.

5 Explainable Machine Learning

This chapter addresses the second and third hypotheses concerning explainability. Similar to the preceding chapter, it follows a structured approach by first examining relevant literature, followed by a detailed description of the implementation, and concluding with the presentation of results.

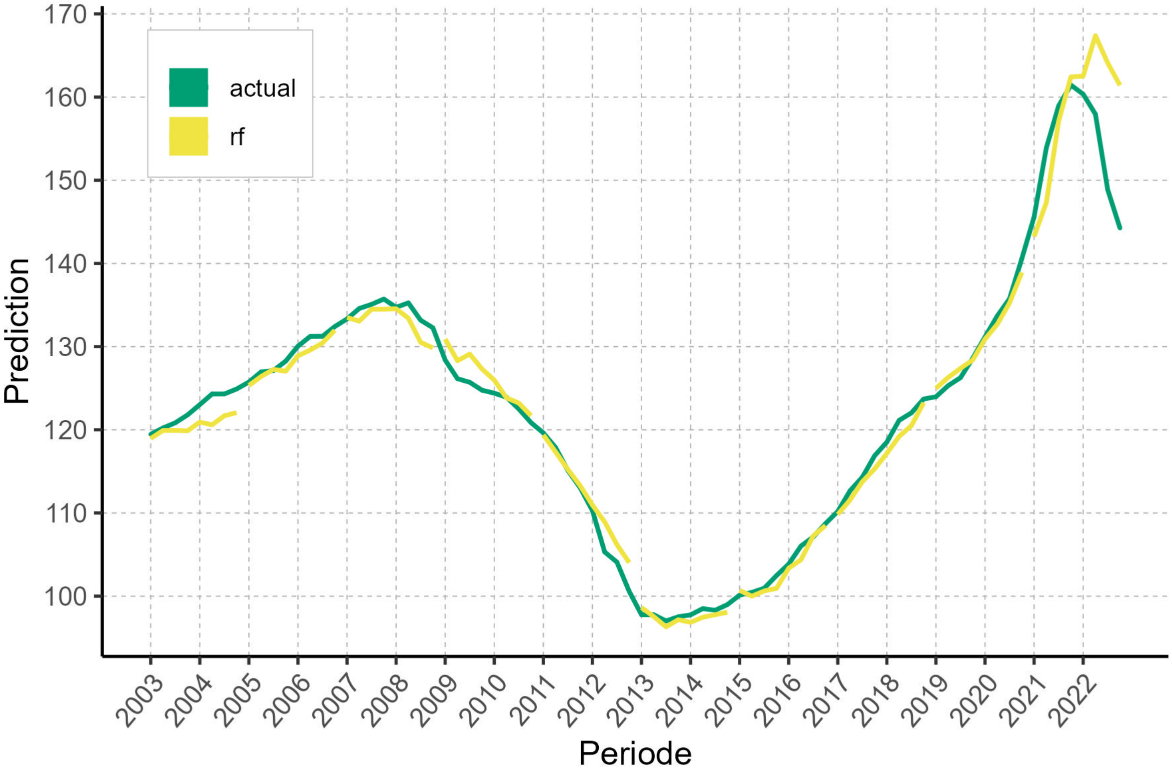


Figure 5: Predictions on validation set by Random Forest (RF) models. The actual values are the real historic housing prices. Each yellow line shows a forecast of eight quarters starting in the first quarter of that year.

5.1 Theoretical Framework

Explainable machine learning is a much less clearly defined field of research than conventional machine learning. This section aims to establish precise definitions within the scope of this research, outline the distinctive features of explainability methods, and motivate implementation decisions.

5.1.1 Dimensions of Interpretability

Existing literature presents diverse ways of characterizing and structuring the multitude of explainability methods [Arrieta et al., 2019], [Burkart and Huber, 2021], [Mohseni et al., 2020], [Guidotti et al., 2018]. This section engages in a discussion centered around three dimensions of interpretability. The aim is to offer a nuanced understanding of the landscape of explainability methods and their applicability in the context of macroeconomic forecasting.

Post-hoc explainability Traditional forecasting models and most linear machine learning models are interpretable in the sense that it is simple for a human observer to understand the link between input and output. With more complex machine learning models often referred to as 'black-box' models, the interpretability diminishes. Despite their lack of inherent interpretability, techniques

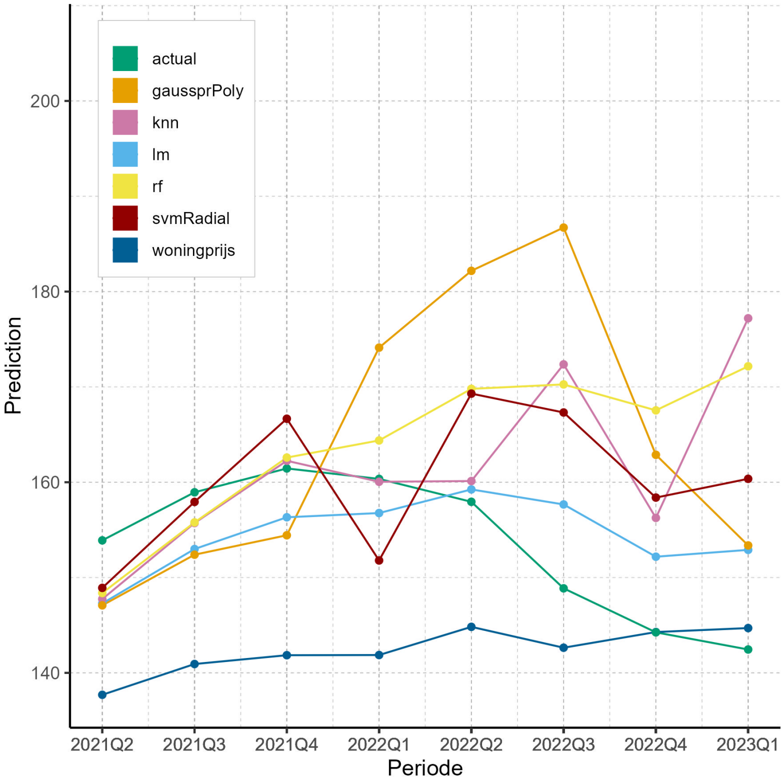


Figure 6: Predictions on the test set. Here, the predictions made by for example the Random Forest are eight predictions made by eight different models, all trained on a different forecasting horizon. The traditional housing price model (woningprijs) provides a lower estimation than the ML models. The ML models overshoot the tipping point in real housing prices.

exist to explain predictions from these black-box models. These explanations can be formed during the training phase of the model or added after predictions are made [Barredo Arrieta et al., 2020] [Mohseni et al., 2020]. When explanations are added after the model has made predictions, they are termed post-hoc explanations. This is in contrast to ante-hoc explanations where interpretability is built into the model during its construction [Burkart and Huber, 2021]. The trade-off between explainability and performance is a critical consideration in this context. Post-hoc explainability methods often provide approximations of the model's inner workings, introducing fidelity issues. Despite this limitation, post-hoc methods offer the advantage of not restricting the ML model, allowing it to achieve maximum performance.

Local versus global explanations In explainable machine learning, a distinction is made between local and global interpretability. Global interpretability refers to understanding the overall logic of a model, and comprehending how the input transforms into the output. On the other hand, local interpretability refers to the rationale behind a specific prediction, providing insights into why that particular outcome occurred [Guidotti et al., 2018] [Burkart and Huber, 2021].

In our use case of economic projections at CPB, economists are required to justify their choices and specific housing price projections. Throughout the projection process, a "zijlicht" model

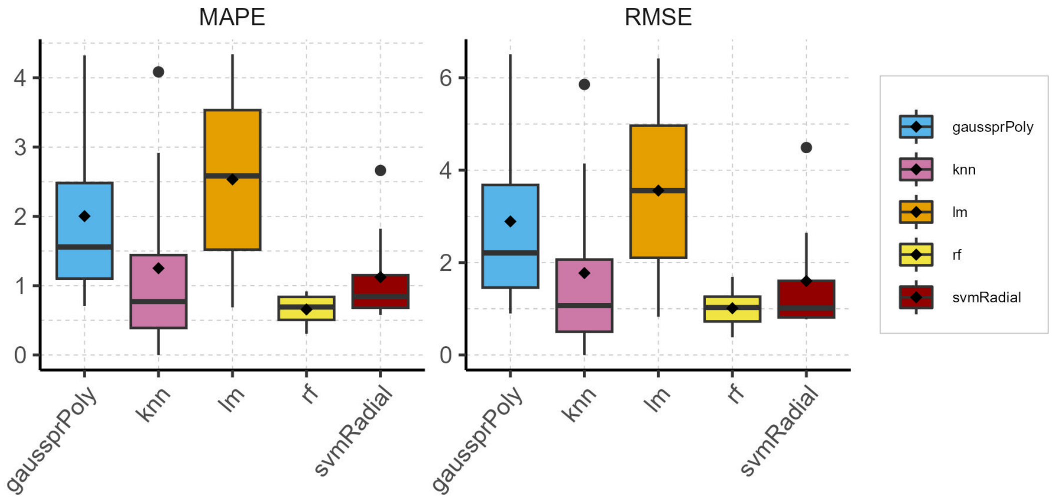


Figure 7: RMSE and MAPE on training sets. The K-neirest neighbor (KNN), random forest (RF) and support vector machine (SVM) have the lowest errors. For these models the error on the training set is much lower than on the validation set, a sign that they are overfitting.

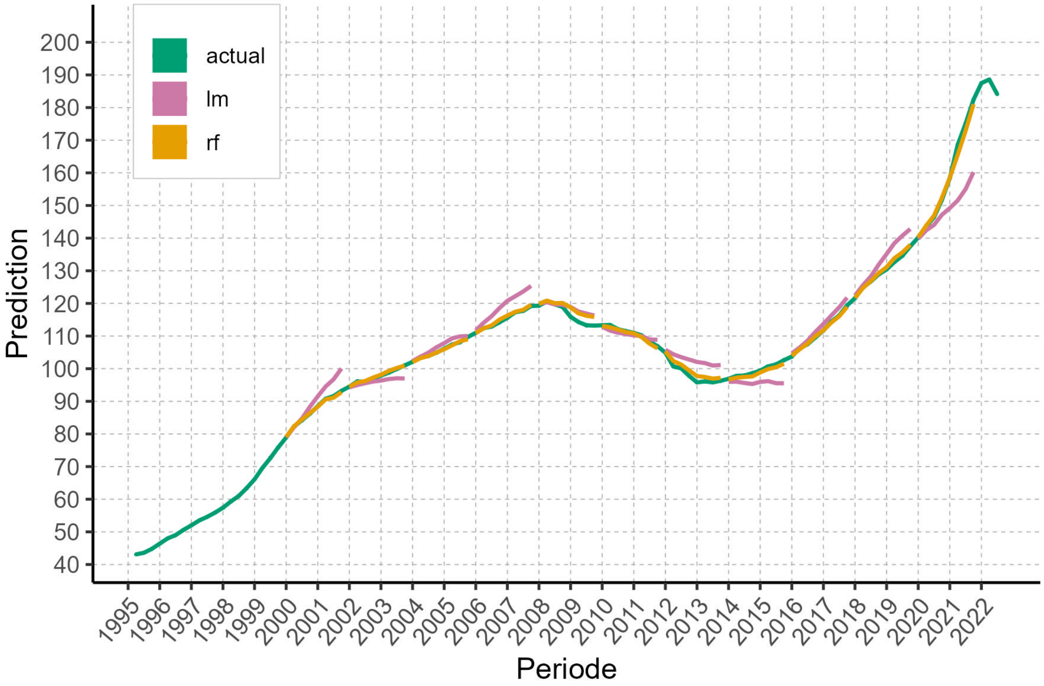


Figure 8: Predictions on the training set. Here, the predictions made by the Random Forest (RF) and Linear Regression (LM) models comprise eight forecasts generated by eight distinct submodels, each trained on a different forecasting horizon. The Random Forest closely follows the actual housing price levels. In contrast, the Linear Regression model predominantly extrapolates trends from preceding housing price data, resulting in a less precise fit.

is employed to guide expert judgment, influencing input variables and outcomes of traditional models. In this scenario, the economist needs to understand why a prediction deviates from their expectations. This emphasizes the importance of local interpretability. Notably, the requirements of economic forecasters diverge from those of economic researchers. While researchers focus on discerning overarching global patterns, forecasters prioritize understanding the nuances of individual projections.

While local interpretability is crucial for decision-making during projections, the interpretability of the model overall remains important. Economists are interested in comprehending the economic patterns that the model has learned. If the model follows economic principles this increases trust in its forecasts. Therefore, both local and global explanations are desirable, but the emphasis remains on local interpretability.

Nature of user expertise The prior knowledge of the user, in this case, the economist plays a crucial role in the explanation process [Guidotti et al., 2018]. It is essential to consider the audience when constructing an explanation. The forecasters are experts in their field, possessing extensive domain knowledge and strong mathematical skills. However, they have limited knowledge of machine learning. [Mohseni et al., 2020] characterizes such users as data experts and emphasizes the role of visualization and interaction techniques to support them. Burkart and Huber [2021] refer to these users as domain experts. They highlight the domain experts' goal of understanding the systems and factors used by the system to incorporate their knowledge. Trust is identified as a crucial requirement for system deployment in the eyes of domain experts.

5.1.2 Characterisation of Methods

When creating an explainability technique, researchers have to deal with the approximation dilemma. This dilemma entails that the explanation should preserve the essential features of the complex model, while also meeting the requirements of the audience [Barredo Arrieta et al., 2020]. This results in a large toolkit of techniques that can be put into various subcategories. Some post-hoc techniques include explanation by visualization, explanation by simplification, and explanation by feature importance [Barredo Arrieta et al., 2020].

- **Visual explanation** techniques are another type of explanation that mainly focuses on form. These explanations are usually only created as model-specific techniques. This is because it is more difficult to create a visualisation without using any information on the internal structure of a model [Barredo Arrieta et al., 2020]. Still, in this work we aim to use only model-agnostic techniques.

- **Explanation by example** approach involves the extraction of examples that provide some insight into the relationships that the black-box model has generalized [Barredo Arrieta et al., 2020].

- **Explanation by feature importance** techniques explain by quantifying how the sensitive a model is to the different input features [Barredo Arrieta et al., 2020]. One of the techniques is the calculation of Shapley values.

Model-agnostic explainability Model-agnostic methods are explainability methods that can be applied to any type of model as opposed to model-specific methods that only work for a specific model class [Burkart and Huber, 2021]. These models treat every model as a black-box, relying on changing the inputs and observing the resulting changes in output. These model-agnostic methods are therefore always post-hoc methods. They are not able to exploit specific properties of any model. They can still convey useful information to end users and provide means to compare different types of models [Štrumbelj and Kononenko, 2014] [Lipton, 2017]. As we have previously seen when constructing a machine learning pipeline, there is not yet a single model that stands out to be the best performing on macroeconomic forecasting. The literature does not agree on the best model, and during our analysis, not one stood out. So it is best to determine the best model on a case-by-case basis and apply a model-agnostic method to provide a similar explanation for each model.

5.1.3 Shapley Values

Shapley values are a concept taken from game theory, the study where two or more players are involved in a strategy to achieve a desired outcome. The contribution of a player is measured by taking the difference in output between games with and without this player. The average of this difference over each possible subset of players is the Shapely value of the player [Shapley, 1953]. The analogy here is that the features are players, working together to achieve an accurate prediction. In the context of feature importance, the features are the players in a predictive model, collaborating to enhance prediction accuracy.

Shapely values adhere to the concept of fair payout as it adheres to four specific conditions [Molnar, 2020].

1. **Additivity:** The sum of Shapley values for a group of features should equal the sum of the Shapley values of each feature within that group.
2. **Dummy:** Any player who contributes nothing to a coalition should receive a value of zero when part of that coalition.
3. **Symmetry:** Features that contribute equally to the prediction should be assigned the same Shapley value.
4. **Efficiency:** The sum of all Shapely values equals the difference between the prediction and the average value of the target.

One key misinterpretation is that Shapley values are often interpreted as the change in prediction if one feature were removed from the model. In reality, they represent the average marginal contribution of a feature value across all possible feature subsets. These feature subsets are not limited to the specific set of features used by the model.

Calculating the Shapley value for a particular feature involves making two predictions for each possible feature subset: one with the feature intact and another with its value replaced by a reasonable random value. The difference between these predictions, averaged over all possible subsets, gives the Shapley value. Unfortunately, this computation becomes infeasible for most real-world examples due to the exponential growth in the number of subsets with increasing feature count. To address this challenge, one can sample subsets of features in the space, though this leaves only an estimation of the actual Shapley values.

Accurate computation of Shapley values also requires a deep understanding of the data to replace feature values with reasonable alternatives. The replacement of feature values can also pose a problem when features are highly correlated. Random values from their marginal distribution may not align with the overall feature relationships, making the resulting sample unrealistic. [Nohara et al., 2022] propose feature packing where the highly correlated feature are packed into one combined feature, without retraining the model. When two features are highly correlated the importance of the combined feature is higher than the sum of each individual importance. This approach allows us to consider correlated features but does not provide insights into their individual contribution.

Despite these challenges, Shapley values offer advantages. They allow for contrasting examples, facilitating outcome explanations by comparing them to alternative scenarios. Additionally, they are grounded in strong mathematical theory, making them a robust choice for an explainability technique.

In their study, Jesus et al. [2021] set out to assess three post-hoc explainability methods. They employed a similar approach to that recommended by Doshi-Velez and Kim [2017] and Hoffman et al. [2019], all be it with a shorter questionnaire. Their findings revealed that the incorporation of SHAP explanations led to enhanced user decision-making accuracy compared to relying solely on raw data and machine learning scores [Jesus et al., 2021]. In this use case, it is an additional benefit for the Shapely values to originate from game theory. It is easier for economists at CPB to adopt an explainable machine learning method when it has its foundation in familiar concepts.

5.1.4 Alternative Explanability Methods

Shapley values represent just one approach to explainability. This section explores alternative methods, outlining their strengths and weaknesses, and justifies the decision to adopt Shapley values.

The permutation feature importance can be defined as the amount by which the error on a prediction increases after changing the value of a feature [Molnar, 2020]. The intuition behind this approach is that changing an important feature has a large impact on the prediction of the model and therefore also on the error. This also includes the effect of the new feature value involving other features. So both the direct and indirect effects are measured in one number [Molnar, 2020]. It is an advantage that dependency between features is taken into account. This does mean that the forecaster should be careful with interpretation. Like with Shapley values, a forecaster may be tempted to use a permutation feature importance as a linear regression coefficient. This approach would only be correct if there is no interaction between the features, as is the case in a linear regression model. It is also not possible to calculate a permutation feature importance for a test set without a true outcome [Molnar, 2020]. So the method is not useful for explaining future forecasts.

Accumulated local effects (ALE) plots are developed as an alternative to Partial Dependence plots by Apley and Zhu [2019]. ALE plots are more suitable when working with dependent features [Apley and Zhu, 2019]. To show the importance of a feature, the difference between predictions is measured when this feature experiences small changes. ALE plots work even when features are correlated but it is not possible to see how large or small the correlation between features is [Molnar, 2020]. Another clear disadvantage of ALE plots remains that they only provide global explanations.

Individual Conditional Expectation (ICE) plots can provide a local explanation for a single feature. For one feature, the ICE plots one line for every sample in the data set. This line shows what the predicted output would be if the feature were a different value [Goldstein et al., 2014]. It is a major downside that the ICE curve can only display a single feature. Another downside to the curve is that in practice not all feature values are realistic, yet those may still be shown in the curve [Molnar, 2020].

LIME (Local Interpretable Model-agnostic Explanations) constructs a new dataset for each data point by changing feature values. It trains a simple interpretable model on this data set and observes the corresponding predictions from the black-box model [Ribeiro et al., 2016]. While the learned model should provide a good local approximation of the black box predictions, this does not translate to a global explanation [Molnar, 2020]. LIME provides human-friendly explanations, particularly useful if decision trees are preferred for understanding. LIME does not satisfy the same mathematical properties as Shapley values [Lundberg and Lee, 2017], but the built-in fidelity measure offers insights into its reliability. A final risk of using LIME, which is present in all perpetuation-based methods, is that small changes in the input data can have large effects on explanations. Here, LIME appears to be particularly unstable [Alvarez-Melis and Jaakkola, 2018].

Considering these other methods Shapley values provide the most flexibility of explanations on both a local and global level while maintaining a strong mathematical foundation. This is why it has been selected as the preferred method at the start of this project. Chapter 6 will comment on whether the intended benefits were realized and consider if alternative methods might not suffer from the same drawbacks.

5.1.5 Assessment of Explanations

Assessment of models constitutes a fundamental aspect of machine learning. Most metrics measure how close the predicted values are to the actual values. Some measure the time it takes to train a model and perform predictions. What exactly constitutes an explanation and how it ought to be valuated is a philosophical discussion without concluding theory [Barredo Arrieta et al., 2020] [Ras et al., 2022]. Unfortunately there is no definitive way to judge the value of an explanation. What working definitions are used in this research are therefore merely one the many possible definitions that could be used. When looking at the value of an interpretable model we look at the following three characteristics taken from [Guidotti et al., 2018].

1. **Accuracy:** Accuracy refers to the extend to which the model accurately predicts samples that it did not trained on. This is the most important characteristic in the assesment of a black-box machine learning model as well. The fact that the model is interpretable, does not remove the need for competitie predictions.

2. **Fidelity:** The fidelity captures how good the explanation approximates the prediction of the black-box model. If the explanation is very clear, but not closely connected to the model outcome, it is still useless.

3. **Interpretability:** This term is also refered to as comprehensibility. It referst to the extent to which the model and or its predictions are understandable by humans.

Let's delve deeper into the concept of interpretability. Interpretability can be viewed as an indication of the level of trust that a user places in an explanation. According to the work of [Ras et al., 2022], trust in an explanation is achieved when the rationale possesses the following properties:

- Easily interpretable
- Relatable to the user
- Connects the decision with contextual information
- Reflects the intermediate thinking

However, a trade-off exists between these properties. An explanation rich in contextual information may be too dense for easy interpretation, while a simple explanation may lack the details needed to reflect intermediate thinking. This properties provide a structure to assessment of an explanation, but methods for valuation are lacking.

5.2 Methodology

This thesis focuses on Shapley values as an explainability method. The previous section introduced the theory behind the Shapley values, the following section discusses the implementation.

5.2.1 SHapley Additive exPlanations (SHAP)

SHapley Additive exPlanations (SHAP) is a method introduced by Lundberg and Lee [2017]. SHAP takes a very specific approach to compute the Shapley values. The linear model g is used to explain the prediction of an instance.

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i$$

$z'_i \in \{0,1\}^M$ is a vector representing a subset of M features. For each feature, the vector denotes whether they are present in the subset i or not. $\phi_j \in R$ is the Shapely value for feature j.

SHAP computes the Shapley values, meaning that it inherits all the advantages and disadvantages of these Shapley values. The resulting Shapley values also satisfy the properties of efficiency, symmetry, dummy, and additivity. In addition to these properties, the paper by Lundberg and Lee [2017] introduces three other desirable properties.

1. **Local Accuracy** The output of the original model f for a specific input x should match the local model g predicting for a feature set x' that the maximum number of features in the model.

$$f(x) = g(x') = \phi_0 + \sum_{i=1}^{M} \phi_i x'_i$$

This property is interchangeable with the property of efficiency.

2. **Missingness** This property ensures that a missing feature is attributed a value of zero.
3. **Consistency** If a model changes in a way that a feature's marginal contribution changes, the Shapley value should change with it. The same applies to the situation where a marginal contribution remains constant. In that case the Shapley value should stay constant as well. From this property, the symmetry, dummy, and additivity properties can be derived.

The theoretical Shapley values are infeasible to compute in real-world applications so different approximations have been implemented. Kernel SHAP provides an efficient approximation but a downside is that features are assumed to be independent [Aas et al., 2021]. Aas et al. [2021] extended this method so that this assumption is no longer necessary. This is the method that has been implemented in the R-Package that is used in this work.

Despite the Kernel SHAP method's reduced computation time, it may still face challenges with speed, especially in the context of large datasets with numerous instances [Molnar, 2020]. During our work with the R package, we observed that practical limits in computation time were encountered around ten to twelve variables. Fortunately, this drawback has minimal impact on the current use case, as macroeconomic forecasting typically involves small data sets. In Chapter 4, the ML models were intentionally restricted to using eight features due to the challenge of finding meaningful patterns with the limited available data points. In this scenario, the limitation of data availability proves to be more restrictive than the computation time of Shapley values.

Shapley values can be combined to create global explanations. The local explanations serve as building blocks of global interpretations, so consistency between the two is maintained. In this work, Shapley values are computed for both the test set and the training set. Shapley values computed for predictions on the test set, offer insights into the derivation of these predictions, constituting local explanations. Additionally, computing Shapley values on the training data itself provides valuable insights into the global patterns learned by the models.

5.2.2 Visualisation

The Shapley values themselves are mere numerical values, insufficient on their own to offer comprehensive explanations. They are also easy to misinterpret as coefficients in a linear regression. A Shapley value is the marginal contribution of a feature across all possible subsets of features, which is different from the coefficients. Still, the two are easily mixed up. How Shapley values are represented significantly influences their interpretation. The following subsection describes the visualizations that serve as local and global explanations. Leveraging the SHAPR package, various visualization options are explored, and custom visualizations are developed to suit the specific use case. Collaborative efforts with the housing market team at CPB have led to the creation of these custom visualizations. Through meetings and presentations, different versions of the visualizations were showcased, and valuable feedback was collected.

Local Explanation As mentioned earlier, Shapley values offer a local explanation, explaining individual predictions. The following plots aim to explain to the user how the individual predictions are built up.

1. **Bar plot:** The most simple way that a Shapley value can be displayed is through a bar plot. The bar plot is native to the SHAPR package and shows the magnitude of each Shapley value. An example of a bar plot can be found in Figure 9.
2. **Waterfall plot:** The waterfall plot reorganizes the bars in the bar plot bars to fall one after the other. This shows the user how the Shapley values together add up to the black-box prediction. Features also are sorted by the magnitude of their Shapley values values. Figure 10 shows a waterfall plot.
3. **Stacked bar plot:** The previous two plots explain a single prediction. In the housing market, predictions are never made for a single quarter. CPB would like to forecast eight quarters ahead and also compare the predictions for different forecasting horizons. The stacked bar plot, developed in collaboration with CPB, is a bar graph displaying the contribution of each Shapley value to the final prediction across all eight points of the projection. While providing less detail than the waterfall plot, this graph enables comparisons of Shapley values across different predictions. An example of a stacked bar plot is shown in Figure 11.

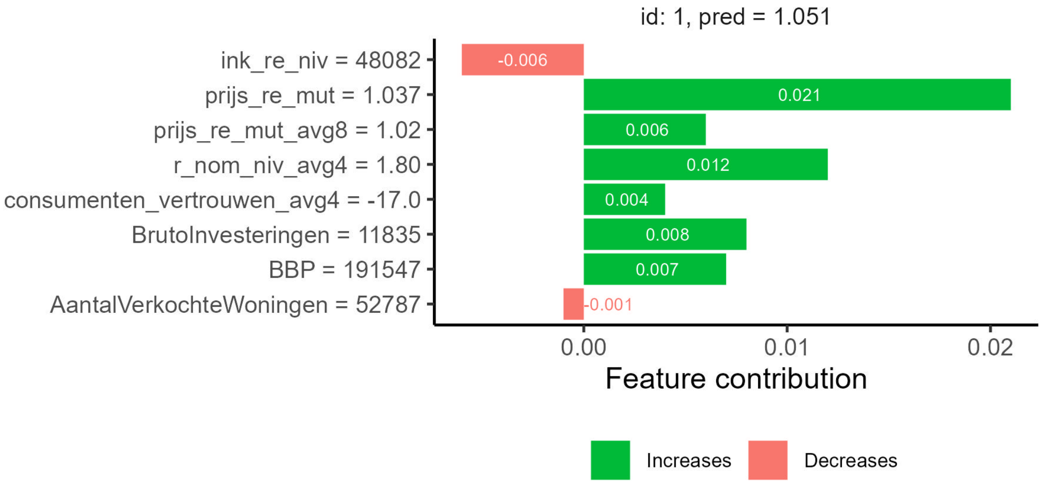


Figure 9: Bar graph shows magnitude of Shapley values. Prediction on the test set, sample from 2021Q2 predicting two quarters ahead. Predictions made using Random Forest model.

Global Explanation Shapley values offer not only local explanations for individual predictions but can also be aggregated to provide global explanations. The first method that is explored involves a measure of feature importance. While it is a misinterpretation to view an individual Shapley value as feature importance, combining them can yield a feature importance metric. The intuition behind this feature importance measure is as follows. If a feature has a large absolute Shapley value, it contributes to the prediction to a large extend and should be considered important [Molnar,

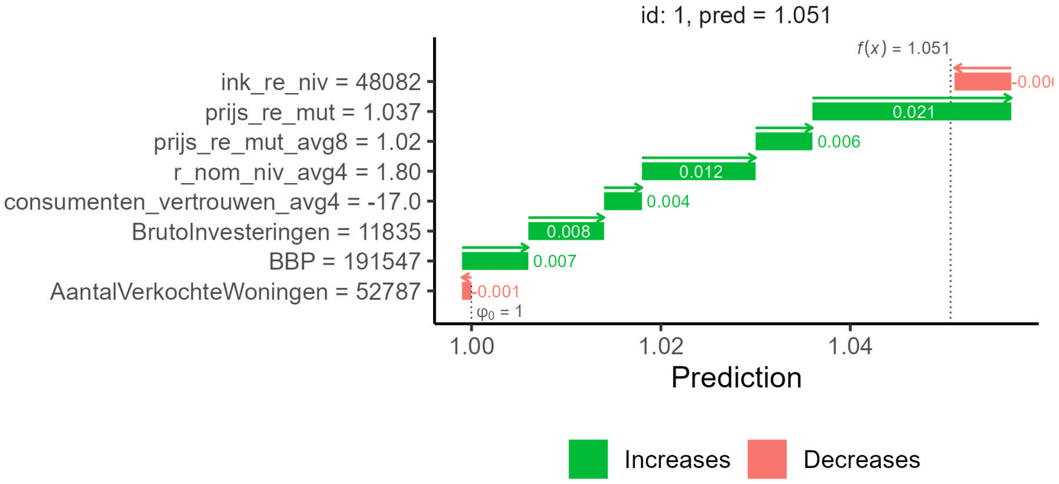


Figure 10: Waterfall graph shows magnitude of Shapley values. Prediction on the test set, sample from 2021Q2 predicting two quarter ahead. Predictions made using Random Forest model. The colored bars show the magnitude and direction of feature contribution for each feature.

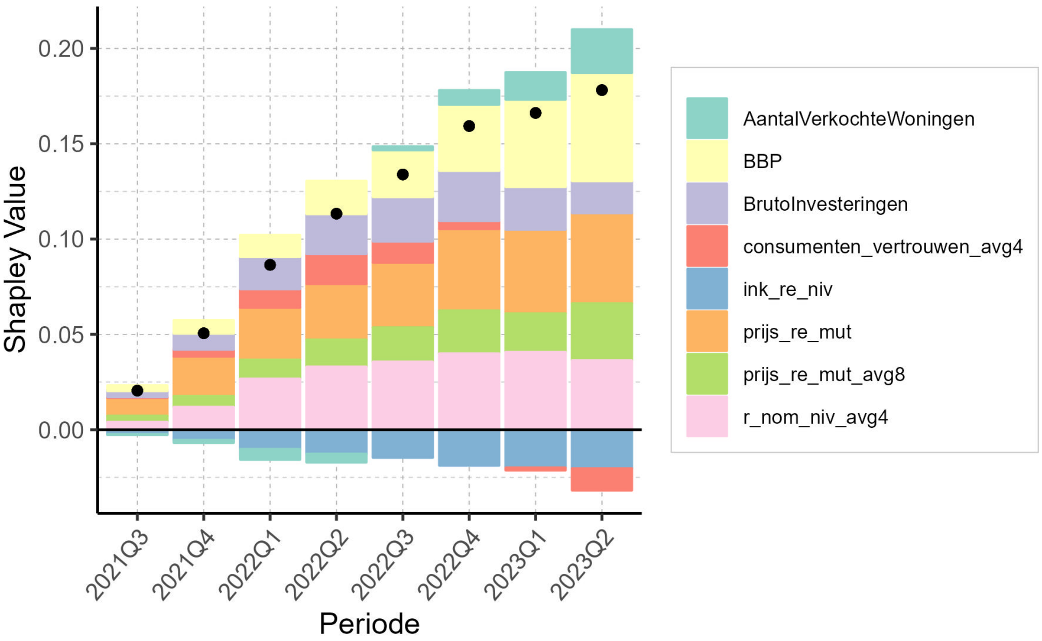


Figure 11: Stacked bar graph show a forecast of eight quarters with 2021Q2 being the last known period. The colored bars show the magnitude and direction of feature contribution. The black dots represent the resulting predicted growth in real housing price.

2020]. To find a global measure of importance the absolute Shapley values (ϕ) are averaged across every sample i. The feature importance for a feature j becomes:

$$I_j = \frac{1}{n} \sum_{i=1}^{n} |\phi_{ij}|$$

The analysis can be conducted for each forecasting horizon individually or collectively as one multi-horizon model. In the latter case, the feature importance is averaged across different forecasting horizons. Figure 12 illustrates both scenarios.

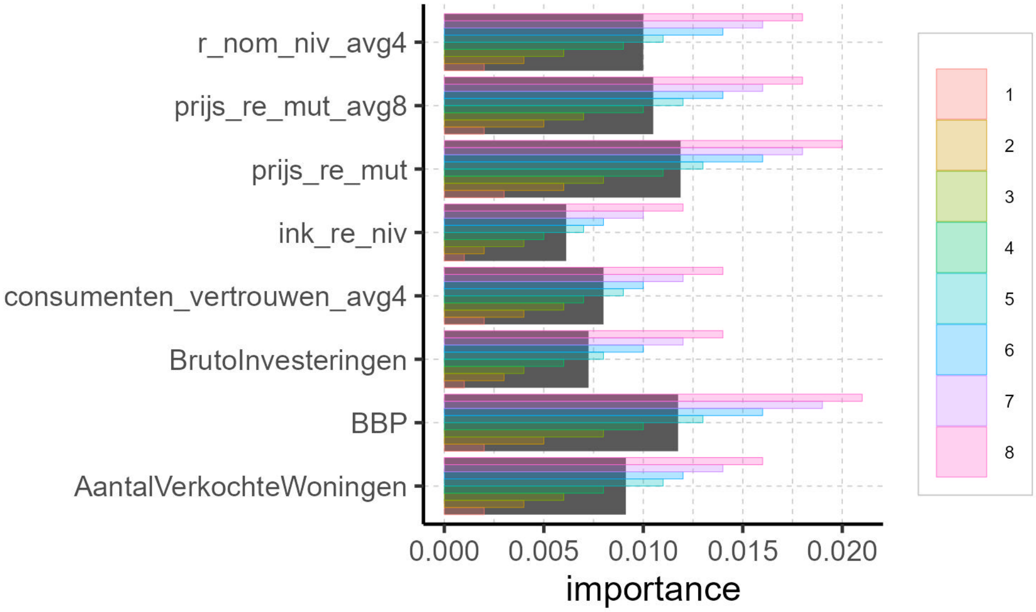


Figure 12: Plot the global feature importance of the random forest model by taking the average absolute Shapely value of a feature. The Shapley values are calculated on the training set. Y-axis shows the features, x-axis the feature importance. The different colors show the feature importance's for different forecasting horizon. The gray bars shows the average across horizons.

The feature importance measure only describes the size of the contribution, not the direction of the effect. The beeswarm plot also aims to connect the Shapley value to the input feature value. Each point on the plot corresponds to a Shapley value for a specific feature and instance. On the y-axis every feature used in the model is plotted. The x-axis shows the magnitude of the Shapley value. The color coding indicates the feature value, ranging from low to high. The beeswarm plot could give a forecaster information about the connection between the feature value and Shapley value. It also shows whether the Shapley values for a feature are generally low or high, compared to other features. This plot is also natively supported by the SHAPR package. An example of a beeswarm plot is shown in Figure 13

For a comprehensive examination of the relationship between feature value and Shapley value, analyzing the distribution of Shapley values can be highly informative. SHAPR provides a clear global interpretation plot where the x-axis represents the feature value, the y-axis shows the Shapley values. An example of this plot is displayed in Figure 14. This visualization also incorporates a histogram to depict the distribution of the feature data. Such a plot proves valuable when exploring

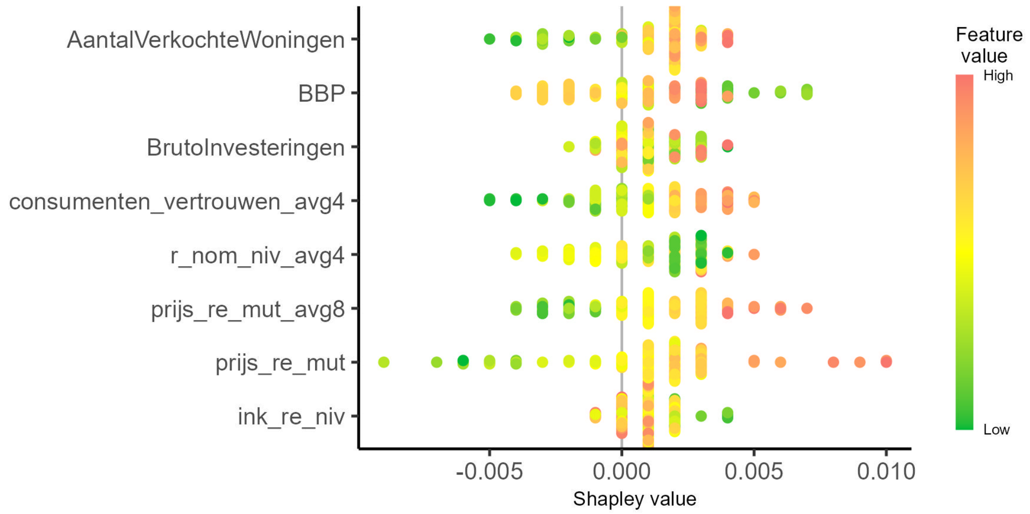


Figure 13: Shows the distribution of Shapely values for the training set by plotting the Shapley value of each feature and instance. Y-axis shows the features, x-axis the shapely value. The color shows whether the feature value of that instance was relatively low or high. Predictions are made on the training set, predicting one quarter ahead using a Random Forest.

potential non-linear relationships between the model's input and output, akin to the VEC plot proposed by [Cortez and Embrechts, 2011].

Figure 14 focuses solely on the Shapley values for one prediction horizon. However, since the multi-horizon model will ultimately be employed as a single model, it is advantageous to examine the relationships between input and output for all horizons in one figure. In Figure 15, the x-axis represents the feature value, and the y-axis represents the Shapley value. This time, all Shapley values for instances in the training set are plotted, with different forecasting horizons displayed in distinct colors.

5.3 Results

In Section 5.1.5, the dimensions on which to evaluate the explainable machine learning model were discussed.The results concerning the first dimension, accuracy, have been outlined in Section 4.3. Addressing the fidelity aspect of the explanation involves examining the mathematical foundation of the method. The estimated Shapley values provide local fidelity, offering a mathematical basis to assume their accuracy in representing the individual predictions. However, any global conclusions drawn from them lack this mathematical foundation, thus global fidelity is not guaranteed. The remaining discussion in this section will focus on the interpretability of the explanation plots that were introduced in Section 5.2.2. An example of each plot will be discussed as well as how it relates to the properties of interpretability that we outlined in Section 5.1.5.

In Chapter 4, a ML pipeline with multiple models was constructed. The explanation methods provided are independent of the choice of model. For simplicity, only the explanations provided on the Random Forest model will be showcased. The choice of this model is arbitrary and does not

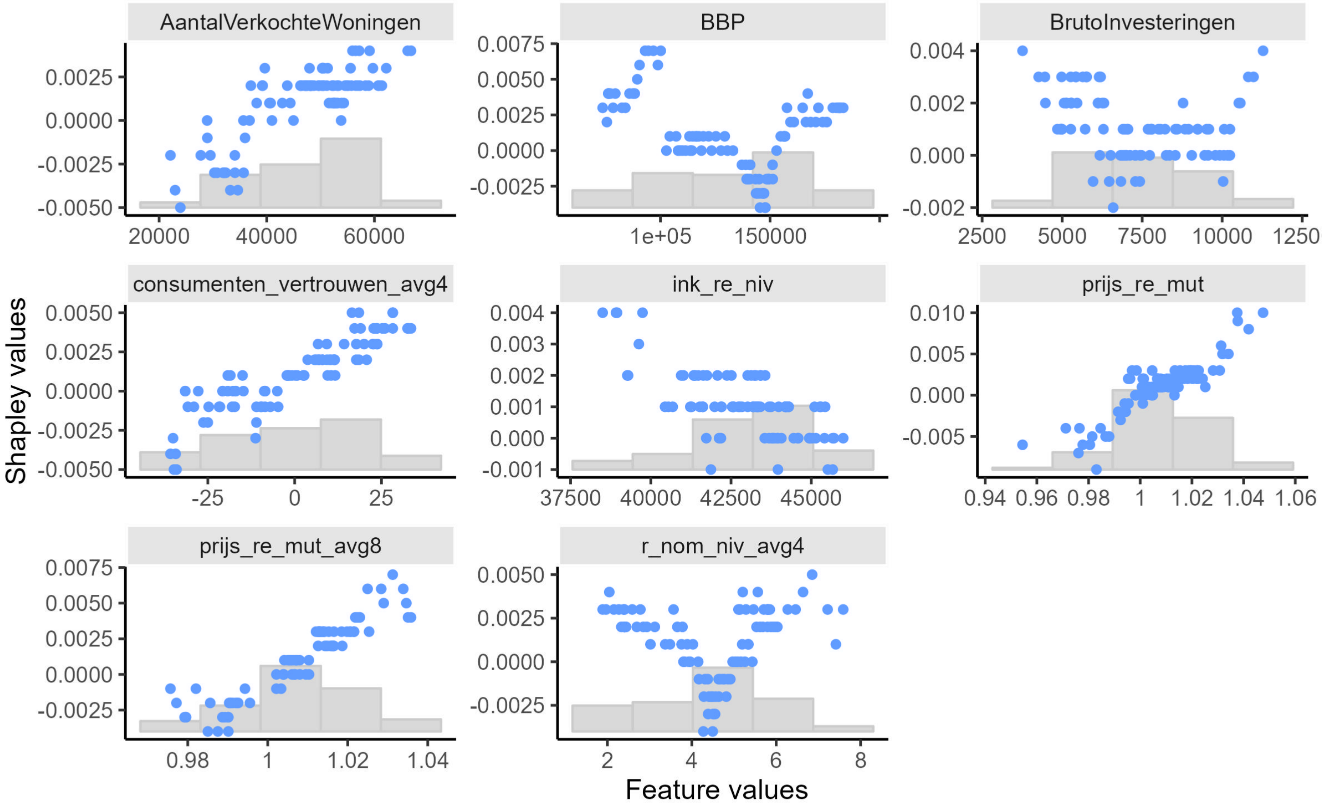


Figure 14: Scatterplot to show the distribution of shapely values by plotting the Shapley values for each feature in a separate subplot. Y-axis shows the Shapley value, x-axis the feature value. The gray histogram shows the distribution of the feature data. Predictions are made on the training set, predicting one quarter ahead using a Random Forest.

alter the results of the explainability method.

During the sessions with the housing market team it became apparent that the following questions should be answered with a collection of explanations plots.

- Which variables contributed to this prediction and to what extend?
- What is the economic reasoning that can explain this relationship?
- How is it possible that models with similar or equal input reach different predictions?

5.3.1 Local Explanation

The local explanations serve to provide information about the predictions on the test set. The bar graph in Figure 9 offers ease of interpretation and provides context to the user by displaying both the Shapley values and their corresponding feature values. This is still a very limited context because it is not possible to compare a prediction for one quarter to the preceding or following quarters. In our use case, a single projection always consists of 8 predictions. A prediction is not

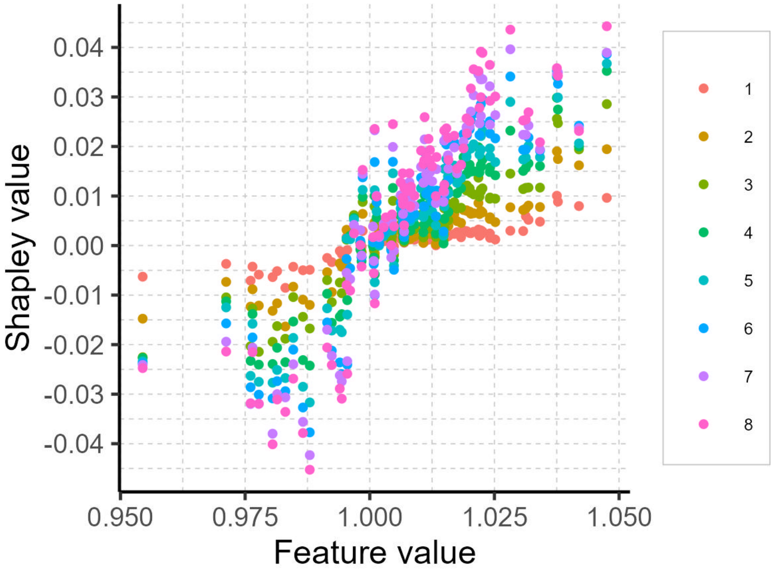


Figure 15: Scatterplot to show the distribution of Shapely values by plotting the Shapley values of the feature *prijs_re_mut*. Shapley values are calculated using the training set and the random forest model. Y-axis shows the Shapley value, x-axis the feature value. The colors represent the different forecasting horizons. Figure shows how the magnitude of Shapley values increases when the models predict further into the future.

only relevant in isolation but with respect to the other predictions as well. This comparison is not facilitated with the bar and waterfall graphs. An advantage that the bar graph does have is the straightforward comparison of the magnitudes of feature contributions. This is done by presenting the actual Shapley values in labels and aligning the bars side by side.

The waterfall plot in Figure 10 is also relatively easy to interpret, as it visually depicts how Shapley values accumulate to form the final prediction. The sequential arrangement aids in understanding the additive impact of features. This additive arrangement is relatable to a forecaster who is used to working with linear equations. Just like the bar graph, the waterfall graph is limited in the context it can provide to the user.

In the previous two visualizations, it is not possible to compare different predictions. The stacked bar graph in figure 11 provides this much-needed additional context. This visualization puts the predictions back in their time order which is more relatable to an economist. This reflects the intermediate thinking of the economist as it allows the user to reason about how the contribution of features can change over time. A disadvantage of this model is that it does not provide contextual information on the feature values that served as input for the predictions. It is also more difficult to see what the exact Shapley values are. When discussing this visualization the economists were more inclined to view the predictions as coming from a single model and perceived a connection between these predictions. While this aligns with their perspective, it doesn't accurately represent the technological implementation. In the ML pipeline, the predictions are made by different ML models of the same type that each make predictions for a specific forecasting horizon. The predictions made one quarter ahead do not influence the predictions for two quarters ahead. While this may not be particularly relevant to individual predictions, it could pose challenges when local explanations are used to reason about the model as a whole. This situation embodies a trade-off between simplicity and staying true to technical implementation. Overall this graph is viewed as the most simple and facilitates economic reasoning. This economic reasoning is supported further by global explanations.

5.3.2 Global Explanation

Figure 12 attempts to present the feature importance for the separate horizon models and their average across eight models. The resulting visualization illustrates that the magnitude of the Shapley values is consistently the largest in the model for horizon eight. This indicates that the model for a further horizon reacts more to the input variables than the model with a shorter horizon. A comparison across different horizons becomes difficult to interpret. An advantage of the graph is that it is relatable to the user. The concept of feature importance resembles the coefficients of an Ordinary Least Squares (OLS) regression. While both coefficients and average absolute Shapley values measure feature importance, the average absolute Shapley value lacks the same direct connection to future predictions by the model. The visualization also provides no information on the distribution of feature values or the dependency between different variables, so the offered context is limited. The overall feature importance in gray does not show much variation. In conclusion, this visualization is not able to show economic patterns or intermediate thinking of the user and is not considered very useful.

The beeswarm graph in Figure 13 attempts to convey a lot of information. It illustrates the distribution and frequency of Shapley values for each feature, along with the distribution of feature values. The interpretation is relatively straightforward when the relationship between the feature value and Shapley value is linear, as seen with the feature *prijs_re_mut_avg8*. However, it becomes considerably more challenging to interpret when this relationship is non-linear, as exemplified by the feature *BBP*. The graph does not specifically relate to any aspect of traditional forecasting so it is not very relatable nor does it reflect intermediate thinking. While it provides some context about feature values, it lacks insights into dependencies between features.

In contrast to Figure 13, Figure 14 is perceived as much more straightforward to interpret. It offers insights into the patterns learned by the ML model and easily relates to economic theory. For instance, it provides observations such as ”When the housing price increased in the previous period, this contributes to an increase in housing prices in the next period.” This connection to economic theory aligns with intermediate thinking. It is important to recognize that while Shapley values reveal associations between features and predictions, these relationships are not causal. Take for instance the interest rate. When the interest rate is low, this contributes to a higher prediction for housing prices. This aligns with the economic theory that when interest rates fall, households have a larger budget so housing prices rise [Vries and Boelhouwer, 2008]. At the same time, when housing prices are high central banks may increase interest rates in an effort to dampen the positive price spiral. This dynamic can explain the V-shaped pattern observed in the Shapley values for interest rates. Consequently, while Shapley values may capture aspects of causal relationships, they may also reflect non-causal associations. Therefore, linking Shapley values to intermediate thinking does not inherently imply a facilitation of causal reasoning. While the histogram in the background offers some context about the data used to train the model, information about the dependency between features is not yet provided. Moreover, users can assess the model's consistency by observing the scatter's spread and direction, as these indicate the strength of the relationship. If the Shapley values exhibit consistency, it enhances trust in the model's predictions. This visualization still has some drawbacks. The different y-axes make it easy to observe patterns for a single feature but complicate the comparison of the magnitude of contributions from different features. Additionally, the visualization is designed for a single submodel, in this case, predicting one quarter in the future. While similarities in patterns may exist across different forecasting horizons, assumptions cannot be made without examining each visualization for every horizon.

The final plot in Figure 15 does aim to display the distribution of Shapley values for the different forecasting horizons. This figure shows one interesting insight. The magnitude of Shapley values increases with the horizon, causing the curve to tilt. This phenomenon suggests that models predicting further into the future anticipate larger changes in housing prices. The underlying intuition is that the housing market exhibits less responsiveness in the short term. This comparison across horizons is interesting but becomes difficult to read when the relationship between Shapley values and feature values is less strong in Figure 16. On the whole, the visualisation is experienced as much less useful than Figure 14.

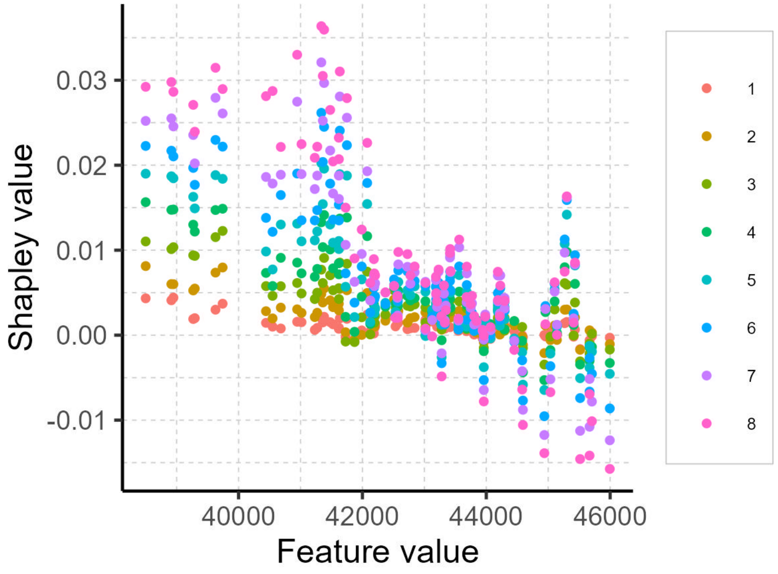


Figure 16: Scatterplot to show the distribution of Shapely values by plotting the Shapley values of the feature *ink_re_niv*. Shapley values are calculated using the training set and the random forest model. Y-axis shows the Shapley value, x-axis the feature value. The colors represent the different forecasting horizons.

6 Discussion

This chapter structures and interprets the results of the previous chapters. The results are critically evaluated in the context of the literature and the business case. Finally, the chapter reflects on the potential for improvement.

6.1 Connection to Related Work

In this study, the Random Forest (RF) model emerged as the highest-performing model. The studies discussed in Chapter 3 lack detailed information about their machine learning (ML) pipeline and their design decisions, but broadly similar ML models were employed in our work. The good performance of the RF is not reflected in the literature. In the work of [Milunovich, 2020], it did not even make it into the top five forecasters. Richardson et al. [2018] did not test the RF so there is no comparison to that work. Milunovich [2020] and Richardson et al. [2018] identified the SVM as the best-performing ML model. In our study, the SVM also performs very well. Goulet Coulombe et al. [2022] found that non-linearity was the ML property that increased performance the most. In this use case, the non-linear models also outperformed the linear ones. Overall, the performance of the RF is surprising but the other results align well with findings in earlier research.

6.2 Comparison to the baseline

The outcomes from the ML models exhibit a remarkable advancement over the baseline, prompting suspicion. Figure 17 shows the predictions of the baseline housing price model, random forest, and linear regression models on the validation set. The substantial discrepancy in performance between the baseline and ML models cannot solely be attributed to non-linearity. Although the linear regression model underperforms relative to its non-linear counterparts, the disparity is less pronounced compared to the baseline. Notably, the ML models leverage twice as many variables, potentially contributing to their enhanced performance. Moreover, the baseline model's conservative predictions diverge from the observed upward trend in housing prices. This conservative stance is likely influenced by the model's error correction mechanism, dampening predicted growth and resulting in lower housing price estimates from 2015 onwards. Particularly large prediction errors in 2010, 2013, and 2021 also contribute to the stark performance contrast.

6.3 Interpretation of Results

Chapter 1 introduced three hypotheses for this research. This section relates the work to these hypotheses to determine whether they can be accepted.

H1: A black-box machine learning model can be used to accurately predict housing prices.

Chapter 4 presented an ML pipeline resulting in multiple black-box models with varying performance. The Random Forest emerges as the most effective with the Support Vector Machine and K-Nearest-Neighbor models following close behind. The ML models all outperform the baseline on the validation set, but not on the test set. This can be explained by the unique characteristics of the test set. The data set aligns with a turning point in housing prices. The price levels have mostly risen steeply in the training data, but in the test set the prices stagnate and fall slightly. This is to the advantage of the baseline which has more conservative estimations due to the error correction term in the model. Still, as explained in Section 3.1, the performance of the housing-price-model is an upper bound. Based on overall performance and consistency the RF is chosen as the preferred model. Overall it can be concluded that the black-box ML models can compete and outperform traditional forecasting methods and the first hypothesis is accepted.

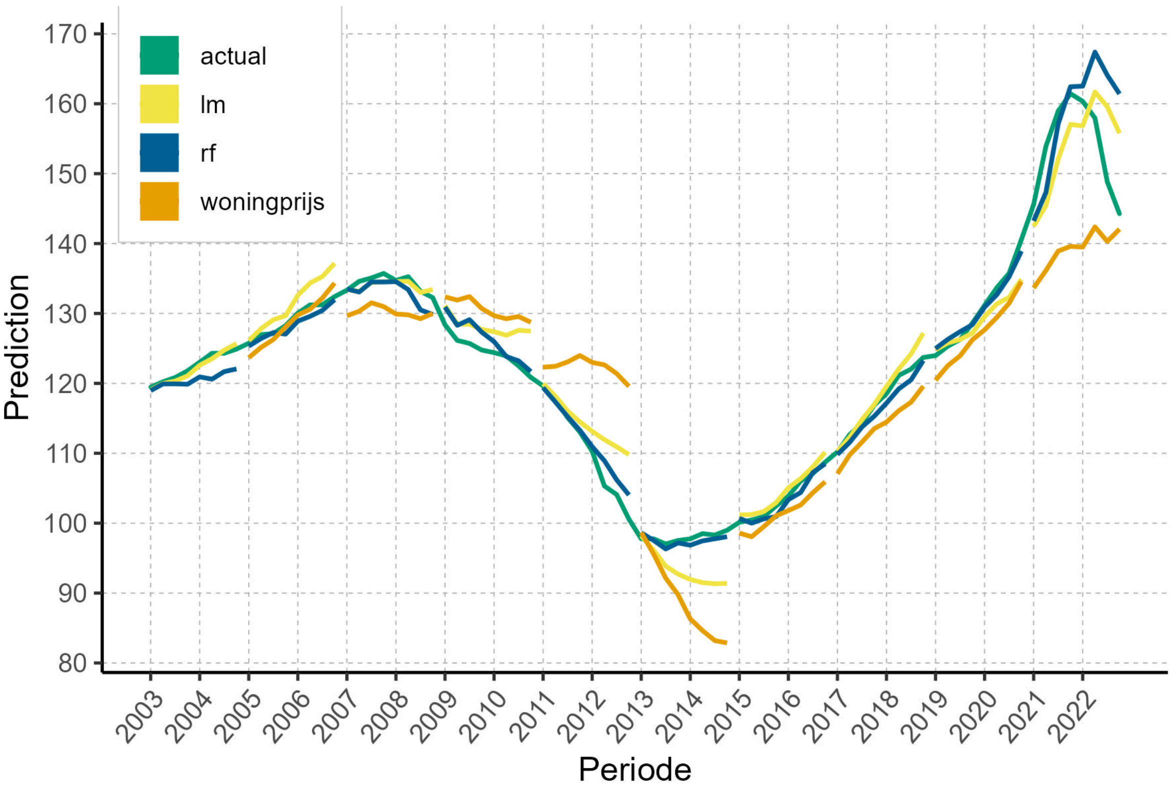


Figure 17: Predictions on validation set by Random Forest (RF), Linear Regression (LR) and baseline (woningprijs) models. Prediction points are made for eight quarters ahead starting at time periods in the validation set. The predictions are compared to the actual house prices. The machine learning models fit closer to the real house prices, especially in the turning points in 2013 and 2021.

H2: A post-hoc explainability method can increase the interpretability of a black box model.

To assess whether interpretability can be enhanced, we first needed to define this concept and establish a structure for reasoning about it. Explainability is inherently a subjective concept, lacking well-defined metrics. In the absence of a numeric metric, we have derived a framework from the literature to evaluate interpretability. Accuracy, fidelity, and interpretability are chosen as characteristics of a good interpretable model.

The accuracy of the black box models is demonstrated in Chapter 4 and is deemed satisfactory. The selected method of Shapley values has a strong mathematical foundation that provides local fidelity. The local explanations from Shapley values can be combined to form global explanations. While these global explanations are consistent with the local ones, the local fidelity does not translate to global fidelity. Overall, the explainability method satisfies fidelity, but there is room for improvement.

Interpretability is another subjective characteristic so to ease the assessment we discuss how the explanation relates to the following five properties. An explanation ought to be easily interpretable, relatable to the user, connect the decision with contextual information, and reflect intermediate thinking. Shapley values, in and of themselves, are only numerical values and do not inherently satisfy these properties. Visualizations that combine multiple Shapley values and incorporate contextual information can fulfill these properties to varying extents. In discussions with forecasters, Figure 11 was found to be the most useful for reasoning about the entire forecasting projection of eight quarters, and the visualization worked well to facilitate economic discussion. This leads to the acceptance of the second hypothesis.

H3: It is possible to construct a combination of explanations and visualizations to achieve a satisfactory understanding of a black box model and its output.

The third and final hypothesis recognizes that no single visualization or explanation is likely to provide complete insights into the black box model or its output. There is already a distinction between local explanations and global explanations, both relevant for macroeconomic forecasting. None of the visualizations introduced in Chapter 5 can offer all relevant information. We found Figure 10 and Figure 11 to be the most useful local explanations. The waterfall graph and stacked bar graph complement each other, with the waterfall graph offering more details and context, and the stacked bar graph allowing for the comparison of predictions within an eight-quarter projection. Together, they provide a comprehensive local explanation.

For global explanations, only the distribution scatter plot in Figure 14 was found useful by the forecasting team. It was simple to interpret, provided context, and facilitated economic reasoning. However, this visualization has limitations as it only shows the Shapley values for one forecasting horizon, requiring the forecaster to view eight different plots to interpret the model. Attempts to combine Shapley values for models of different forecasting horizons into a single figure, as seen in Figure 12 and 14, negatively impacted interpretability. Another limitation of the global explanations is that none have been able to address the dependency between features. Shapley values consider these dependencies, but none of the introduced visualizations capitalize on this aspect to show these dependencies. Visualizations inherently face limitations in representing more than two or three dimensions. With eight or nine features, it becomes impossible to visualize all dependencies in a single graph. For economists to gain explicit insight into the workings of the model, addressing these dependencies is crucial. Consequently, there is not yet a satisfactory understanding of the black box model, leading to the rejection of the third hypothesis.

6.4 Connection to the Business Case

Chapter 2 described the motivations of the CPB and the resulting requirements for the ”zijlicht” model. All decisions made in the construction of the black box model were informed by literature to ensure a strong scientific foundation. The black box models have demonstrated the ability to provide accurate housing price level projections along with explanations. In collaboration with the housing market team, these explanations could be used to evaluate both the projections and the model itself. This shows that the ”zijlicht” model meets the first three requirements effectively.

The final requirement is only partly satisfied. The black-box models do not inherently include economic principles. The explanation of results is based on patterns identified by the model. The economic reasoning behind this pattern has to be formulated by the forecaster. This introduces a level of subjectivity, where a creative economist can argue a relationship they see fit. Furthermore, the forecaster has limited options to modify the relationship between input and output established by the model. If the valuation of a feature does not align with economic principles the only option is to remove the feature from the model entirely. While this eliminates a faulty relationship and may enhance the model, it is an imprecise tool and diminishes trust in the model overall.

6.5 Alternative Explainability Methods

In Section 5.1.4, various alternative explainability methods were explored. Ultimately, Shapley values were selected due to their robust mathematical foundation and their versatility in providing diverse types of explanations. Throughout this study, we leveraged this versatility to generate both local and global explanations. Similar outcomes could potentially be achieved by combining alternative methods, such as using ALE plots for global explanations and ICE plots for local explanations. However, these approaches may not seamlessly integrate due to differing computation methods. The primary gap in the explanation provided by Shapley values lies in the absence of a global explanation illustrating feature dependencies. None of the proposed alternatives can effectively address this gap. To summarize, while Shapley values have proven to be a valuable choice, it remains important to recognize the existence of alternative methods and their potential value.

6.6 Limitations

The limitations of the explainable ”zijlicht” model have already been mentioned in the previous subsections, but it is worth repeating them here. The black box model is designed to maximize performance, implying that it does not inherently adhere to economic principles. To force this behavior into the model, the only influence a forecaster has is on the features used for prediction.

In constructing the ML pipeline, various design choices were made. These decisions, while supported by existing literature, were usually not the most technically advanced techniques. For instance, the use of random search in hyperparameter optimization. The literature shows that this method outperforms other simple methods like manual search or grid search. Still, it may be surpassed by more sophisticated alternatives like genetic algorithms. The techniques used in this are chosen to balance simplicity and effectiveness. This could leave some potential for further performance improvements.

The black-box models are also limited by the availability of data. This is however a problem that is universal in macroeconomic forecasting so it should not be seen as a limitation of the study but a rather constraint of the field where explainable machine learning is applied.

Finally to turn back to the explainability part of the research. While the explanations offer substantial information, global explanations are still limited. Only explanations for each submodel are interpretable. This reflects the implementation, but not how the model will be used in practice. Notably, there is also no explanation of the dependency between feature values. It’s important to approach the interpretation of Shapley values with caution, as they describe the model rather than economic reality and it is not possible to infer causal relationships. Furthermore, The valuation of the explanation is subjective and varies highly for different individuals. The perception of the housing market team at CPB may be somewhat generalizable to other macroeconomic forecasters, but likely not to other domains.

7 Conclusion

The thesis explores the application of explainable machine learning for macroeconomic forecasting, focusing on the question: "How can machine learning techniques be employed in housing market prediction in an explainable manner?" Explainable machine learning is applied to a novel domain. While some experiments with ML models have been conducted, there is still limited available research on the application of machine learning for macroeconomic forecasting. To make ML models useful in practice, the untransparent nature of these models has to be addressed. This work describes a structured approach to explain machine learning for the use case of macroeconomic forecasting.

A multi-horizon machine learning approach is constructed to predict housing price levels for eight quarters into the future. For each forecasting horizon, a separate ML model is trained. Multiple types of models are tested, out of those the Random Forest is found to be the most useful. The ML models can outperform a traditional economic forecasting model. It shows how ML can be an effective tool in housing price prediction. Shapley values are used to open up these black-box models. Shapley values quantify the marginal contribution of a feature, considering all possible feature sub-combinations. This explains feature contribution with a strong mathematical foundation and also takes into account dependencies among features. While exact computation of Shapley values is not computationally feasible, reliable estimations exist. The Shapley values explain the prediction outcomes for a single observation. This is very suitable to the case of macroeconomic forecasting because a forecaster has to motivate their projected housing price for each quarter. So the Shapley values provide a suitable explanation, but they are easy to misinterpret. To prevent misinterpretation Shapley values should be displayed in an accessible way that facilitates accurate interpretation and economic reasoning. Chapter 5 proposes various visualization methods, enhancing the interpretability of Shapley values and providing valuable context for forecasters. In the end, it is possible to provide the forecaster with clear model explanations that provide additional context and facilitate economic reasoning. This opens up the possibility for ML to be used in macroeconomic forecasting.

7.1 Recommendations for Future Work

Acknowledging the limitations outlined in Section 6.6, future work could aim to address these. There is still some room for improvement in prediction performance. In addition, different explainability methods could be explored to improve global explainability. Finally, we can go a step further than just explaining the black-box models. The usability of ML models for forecasters could be improved by aligning ML-learned patterns with economic principles. The realm of interactive machine learning holds promise for granting forecasters greater control over ML algorithms.

References

Kjersti Aas, Martin Jullum, and Anders Løland. Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. *Artificial Intelligence*, 298: 103502, September 2021. ISSN 00043702. doi: 10.1016/j.artint.2021.103502. URL https://linkinghub.elsevier.com/retrieve/pii/S0004370221000539.

Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Nachaat AbdElatif Mohamed, and Humaira Arshad. State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11):e00938, November 2018. ISSN 2405-8440. doi: 10.1016/j.heliyon.2018.e00938. URL https://www.sciencedirect.com/science/article/pii/S2405844018332067.

Nesreen K. Ahmed, Amir F. Atiya, Neamat El Gayar, and Hisham El-Shishiny. An Empirical Comparison of Machine Learning Models for Time Series Forecasting. *Econometric Reviews*, 29 (5-6):594–621, August 2010. ISSN 0747-4938, 1532-4168. doi: 10.1080/07474938.2010.481556. URL http://www.tandfonline.com/doi/abs/10.1080/07474938.2010.481556.

George Alogoskoufis and Ron Smith. On Error Correction Models: Specification, Interpretation, Estimation. *Journal of Economic Surveys*, 5:97–128, February 1991. doi: 10.1111/j.1467-6419.1991.tb00128.x.

David Alvarez-Melis and Tommi S. Jaakkola. On the Robustness of Interpretability Methods, June 2018. URL http://arxiv.org/abs/1806.08049. arXiv:1806.08049 [cs, stat].

Daniel W. Apley and Jingyu Zhu. Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models, August 2019. URL http://arxiv.org/abs/1612.08468. arXiv:1612.08468 [stat].

Sylvain Arlot and Alain Celisse. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4(none), January 2010. ISSN 1935-7516. doi: 10.1214/09-SS054. URL http://arxiv.org/abs/0907.4728. arXiv:0907.4728 [math, stat].

J.Scott Armstrong and Fred Collopy. Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting*, 8(1):69–80, June 1992. ISSN 01692070. doi: 10.1016/0169-2070(92)90008-W. URL https://linkinghub.elsevier.com/retrieve/pii/016920709290008W.

Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI, December 2019. URL http://arxiv.org/abs/1910.10045. arXiv:1910.10045 [cs].

Susan Athey. The Impact of Machine Learning on Economics. In *The Economics of Artificial Intelligence: An Agenda*, pages 507–547. University of Chicago Press, January 2018. URL https://www.nber.org/books-and-chapters/economics-artificial-intelligence-agenda/impact-machine-learning-economics.

Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, June 2020. ISSN 1566-2535. doi: 10.1016/j.inffus.2019.12.012. URL https://www.sciencedirect.com/science/article/pii/S1566253519308103.

Souhaib Ben Taieb, Gianluca Bontempi, Amir F. Atiya, and Antti Sorjamaa. A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. *Expert Systems with Applications*, 39(8):7067–7083, June 2012. ISSN 0957-4174. doi: 10.1016/j.eswa.2012.01.039. URL https://www.sciencedirect.com/science/article/pii/S0957417412000528.

Christoph Bergmeir and José M. Benítez. On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191:192–213, May 2012. ISSN 00200255. doi: 10.1016/j.ins.2011.12.028. URL https://linkinghub.elsevier.com/retrieve/pii/S0020025511006773.

James Bergstra and Yoshua Bengio. Random Search for Hyper-Parameter Optimization. 2012.

Daniel Berrar. Cross-Validation. January 2018. ISBN 978-0-12-809633-8. doi: 10.1016/B978-0-12-809633-8.20349-X.

Christopher M. Bishop. *Pattern recognition and machine learning.* Information science and statistics. Springer, New York, 2006. ISBN 978-0-387-31073-2.

P J Boelhouwer, M E A Haffner, P Neuteboom, and P de Vries. Koopprijsontwikkeling en de fiscale behandeling van het ei. . . . 2001.

Alexei Botchkarev. A New Typology Design of Performance Metrics to Measure Errors in Machine Learning Regression Algorithms. *Interdisciplinary Journal of Information, Knowledge, and Management*, 14:045–076, 2019. ISSN 1555-1229, 1555-1237. doi: 10.28945/4184. URL https://www.informingscience.org/Publications/4184.

Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, October 2001. ISSN 1573-0565. doi: 10.1023/A:1010933404324. URL https://doi.org/10.1023/A:1010933404324.

Nadia Burkart and Marco F. Huber. A Survey on the Explainability of Supervised Machine Learning. *Journal of Artificial Intelligence Research*, 70:245–317, January 2021. ISSN 1076-9757. doi: 10.1613/jair.1.12228. URL https://jair.org/index.php/jair/article/view/12228.

Prabir Burman, Edmond Chow, and Deborah Nolan. A Cross-Validatory Method for Dependent Data. *Biometrika*, 81(2):351–358, 1994. ISSN 0006-3444. doi: 10.2307/2336965. URL https://www.jstor.org/stable/2336965. Publisher: [Oxford University Press, Biometrika Trust].

T. Chai and R. R. Draxler. Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3):1247–1250, June 2014. ISSN 1991-959X. doi: 10.5194/gmd-7-1247-2014. URL https://gmd.copernicus.org/articles/7/1247/2014/gmd-7-1247-2014.html. Publisher: Copernicus GmbH.

Jireh Yi-Le Chan, Steven Mun Hong Leow, Khean Thye Bea, Wai Khuen Cheng, Seuk Wai Phoong, Zeng-Wei Hong, and Yen-Lin Chen. Mitigating the Multicollinearity Problem and Its Machine Learning Approach: A Review. *Mathematics*, 10(8):1283, January 2022. ISSN 2227-7390. doi: 10.3390/math10081283. URL https://www.mdpi.com/2227-7390/10/8/1283. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute.

Eunsuk Chong, Chulwoo Han, and Frank Park. Deep Learning Networks for Stock Market Analysis and Prediction: Methodology, Data Representations, and Case Studies. *Expert Systems with Applications*, 83, April 2017. doi: 10.1016/j.eswa.2017.04.030.

Giorgio Corani, Alessio Benavoli, and Marco Zaffalon. Time series forecasting with Gaussian Processes needs priors. volume 12978, pages 103–117. 2021. doi: 10.1007/978-3-030-86514-6_7. URL http://arxiv.org/abs/2009.08102. arXiv:2009.08102 [cs, stat].

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995. ISSN 0885-6125, 1573-0565. doi: 10.1007/BF00994018. URL http://link.springer.com/10.1007/BF00994018.

Paulo Cortez and Mark J. Embrechts. Opening black box Data Mining models using Sensitivity Analysis. In *2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pages 341–348, Paris, France, April 2011. IEEE. ISBN 978-1-4244-9926-7. doi: 10.1109/CIDM.2011.5949423. URL http://ieeexplore.ieee.org/document/5949423/.

Jeroen Delfos, Anneke Zuiderwijk, Sander Van Cranenburgh, and Caspar Chorus. Perceived challenges and opportunities of machine learning applications in governmental organisations: an interview-based exploration in the Netherlands. In *15th International Conference on Theory and Practice of Electronic Governance*, pages 82–89, Guimarães Portugal, October 2022. ACM. ISBN 978-1-4503-9635-6. doi: 10.1145/3560107.3560122. URL https://dl.acm.org/doi/10.1145/3560107.3560122.

Ajit Desai. Machine learning for economics research: when, what and how, October 2023. URL https://www.bankofcanada.ca/2023/10/staff-analytical-note-2023-16/. Number: 2023-16 Publisher: Bank of Canada.

Finale Doshi-Velez and Been Kim. Towards A Rigorous Science of Interpretable Machine Learning, March 2017. URL http://arxiv.org/abs/1702.08608. arXiv:1702.08608 [cs, stat].

Andrei Dubovik, Adam Elbourne, Mark Kattenberg, and Bram Hendriks. Forecasting World Trade Using Big Data and Machine Learning Techniques. 2022.

Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation, March 2014. URL http://arxiv.org/abs/1309.6392. arXiv:1309.6392 [stat].

Philippe Goulet Coulombe, Maxime Leroux, Dalibor Stevanovic, and Stéphane Surprenant. Macroeconomic data transformations matter. *International Journal of Forecasting*, 37(4): 1338–1354, October 2021. ISSN 01692070. doi: 10.1016/j.ijforecast.2021.05.005. URL https://linkinghub.elsevier.com/retrieve/pii/S0169207021000777.

Philippe Goulet Coulombe, Maxime Leroux, Dalibor Stevanovic, and Stéphane Surprenant. How is machine learning useful for macroeconomic forecasting? *Journal of Applied Econometrics*, 37(5):920–964, August 2022. ISSN 0883-7252, 1099-1255. doi: 10.1002/jae.2910. URL https://onlinelibrary.wiley.com/doi/10.1002/jae.2910.

Clive W.J. Granger. The Philosophy of Economic Forecasting. In *Philosophy of Economics*, pages 311–327. Elsevier, 2012. ISBN 978-0-444-51676-3. doi: 10.1016/B978-0-444-51676-3.50012-9. URL https://linkinghub.elsevier.com/retrieve/pii/B9780444516763500129.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys*, 51(5):93:1–93:42, August 2018. ISSN 0360-0300. doi: 10.1145/3236009. URL https://dl.acm.org/doi/10.1145/3236009.

Richard G. Brereton and Gavin R. Lloyd. Support Vector Machines for classification and regression. *Analyst*, 135(2):230–267, 2010. doi: 10.1039/B918972F. URL https://pubs.rsc.org/en/content/articlelanding/2010/an/b918972f. Publisher: Royal Society of Chemistry.

James D. Hamilton. *Time series analysis*. Princeton Univ. Press, Princeton, NJ, 1994. ISBN 978-0-691-04289-3.

Robert R. Hoffman, Shane T. Mueller, Gary Klein, and Jordan Litman. Metrics for Explainable AI: Challenges and Prospects, February 2019. URL http://arxiv.org/abs/1812.04608. arXiv:1812.04608 [cs].

J J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences of the United States of America*, 79(8): 2554–2558, April 1982. ISSN 0027-8424. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC346238/.

Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren, editors. *Automated Machine Learning: Methods, Systems, Challenges*. The Springer Series on Challenges in Machine Learning. Springer International Publishing, Cham, 2019. ISBN 978-3-030-05317-8 978-3-030-05318-5. doi: 10.1007/978-3-030-05318-5. URL http://link.springer.com/10.1007/978-3-030-05318-5.

Sérgio Jesus, Catarina Belém, Vladimir Balayan, João Bento, Pedro Saleiro, Pedro Bizarro, and João Gama. How can I choose an explainer? An Application-grounded Evaluation of Post-hoc Explanations. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 805–815, March 2021. doi: 10.1145/3442188.3445941. URL http://arxiv.org/abs/2101.08758. arXiv:2101.08758 [cs].

Max Kuhn and Kjell Johnson. *Feature Engineering and Selection: A Practical Approach for Predictive Models*. June 2019. URL http://www.feat.engineering/.

Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. Feature Selection: A Data Perspective. *ACM Computing Surveys*, 50(6):1–45, November 2018. ISSN 0360-0300, 1557-7341. doi: 10.1145/3136625. URL https://dl.acm.org/doi/10.1145/3136625.

Zachary C. Lipton. The Mythos of Model Interpretability, March 2017. URL http://arxiv.org/abs/1606.03490. arXiv:1606.03490 [cs, stat].

Octavio Loyola-González. Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses From a Practical Point of View. *IEEE Access*, 7:154096–154113, 2019. ISSN 2169-3536. doi: 10.1109/ACCESS.2019.2949286. URL https://ieeexplore.ieee.org/abstract/document/8882211. Conference Name: IEEE Access.

Scott Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions, November 2017. URL http://arxiv.org/abs/1705.07874. arXiv:1705.07874 [cs, stat].

K M Luth. Analyse Financieringsruimte en huizenprijzen. 2023.

Spyros Makridakis. Accuracy measures: theoretical and practical concerns. *International Journal of Forecasting*, 9(4):527–529, December 1993. ISSN 0169-2070. doi: 10.1016/0169-2070(93)90079-3. URL https://www.sciencedirect.com/science/article/pii/0169207093900793.

Stephen Malpezzi. A Simple Error Correction Model of House Prices. *Journal of Housing Economics*, 8(1):27–62, March 1999. ISSN 1051-1377. doi: 10.1006/jhec.1999.0240. URL https://www.sciencedirect.com/science/article/pii/S1051137799902401.

George Milunovich. Forecasting Australia's real house price index: A comparison of time series and machine learning methods. *Journal of Forecasting*, 39(7):1098–1118, November 2020. ISSN 0277-6693, 1099-131X. doi: 10.1002/for.2678. URL https://onlinelibrary.wiley.com/doi/10.1002/for.2678.

Sina Mohseni, Niloofar Zarei, and Eric D. Ragan. A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems, August 2020. URL http://arxiv.org/abs/1811.11839. arXiv:1811.11839 [cs].

Christoph Molnar. *Interpretable Machine Learning*. 2020. URL https://christophm.github.io/interpretable-ml-book/.

Sendhil Mullainathan and Jann Spiess. Machine Learning: An Applied Econometric Approach. *Journal of Economic Perspectives*, 31(2):87–106, May 2017. ISSN 0895-3309. doi: 10.1257/jep.31.2.87. URL https://www.aeaweb.org/articles?id=10.1257/jep.31.2.87.

K. R. Müller, A. J. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik. Predicting time series with support vector machines. In Wulfram Gerstner, Alain Germond, Martin Hasler, and Jean-Daniel Nicoud, editors, *Artificial Neural Networks — ICANN'97*, Lecture Notes in Computer Science, pages 999–1004, Berlin, Heidelberg, 1997. Springer. ISBN 978-3-540-69620-9. doi: 10.1007/BFb0020283.

Yasunobu Nohara, Koutarou Matsumoto, Hidehisa Soejima, and Naoki Nakashima. Explanation of Machine Learning Models Using Shapley Additive Explanation and Application for Real Data in Hospital. *Computer Methods and Programs in Biomedicine*, 214:106584, February 2022. ISSN 01692607. doi: 10.1016/j.cmpb.2021.106584. URL http://arxiv.org/abs/2112.11071. arXiv:2112.11071 [cs, stat].

Bastiaan Overvest, Jasper de Winter, Loes Verstegen, Mark Kattenberg, and Jonathan Rusch. Macromodellen ook waardevol in crisistijd. February 2024. URL https://esb.nu/macromodellen-ook-waardevol-in-crisistijd/.

Antonio Rafael Sabino Parmezan, Vinicius M.A. Souza, and Gustavo E.A.P.A. Batista. Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model. *Information Sciences*, 484:302–337, May 2019. ISSN 00200255. doi: 10.1016/j.ins.2019.01.076. URL https://linkinghub.elsevier.com/retrieve/pii/S0020025519300945.

Gabrielle Ras, Ning Xie, Marcel van Gerven, and Derek Doran. Explainable Deep Learning: A Field Guide for the Uninitiated. *Journal of Artificial Intelligence Research*, 73:329–396, January 2022. ISSN 1076-9757. doi: 10.1613/jair.1.13200. URL https://www.jair.org/index.php/jair/article/view/13200.

Carl Rasmussen, Olivier Bousquet, Ulrike Luxburg, and Gunnar Rätsch. Gaussian Processes in Machine Learning. *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures, 63-71 (2004)*, 3176, September 2004. ISSN 978-3-540-23122-6. doi: 10.1007/978-3-540-28650-9_4.

Charles G. Renfro. Economic Forecasts. In Kimberly Kempf-Leonard, editor, *Encyclopedia of Social Measurement*, pages 741–750. Elsevier, New York, January 2005. ISBN 978-0-12-369398-3. doi: 10.1016/B0-12-369398-5/00546-6. URL https://www.sciencedirect.com/science/article/pii/B0123693985005466.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier, August 2016. URL http://arxiv.org/abs/1602.04938. arXiv:1602.04938 [cs, stat].

Adam Richardson, Thomas Mulder, and Tugru L Vehbi. Nowcasting New Zealand GDP using machine learning algorithms. *SSRN Electronic Journal*, 2018. ISSN 1556-5068. doi: 10.2139/ssrn.3256578. URL https://www.ssrn.com/abstract=3256578.

Bas Scheer. Addressing Unemployment Rate Forecast Errors in Relation to the Business Cycle. 2022. doi: 10.34932/K7S1-Y237. URL https://www.cpb.nl/en/addressing-unemployment-rate-forecast-errors-in-relation-to-the-business-cycle. Medium: pdf Publisher: [object Object].

L. S. Shapley. 17. A Value for n-Person Games. In Harold William Kuhn and Albert William Tucker, editors, *Contributions to the Theory of Games (AM-28), Volume II*, pages 307–318. Princeton University Press, December 1953. ISBN 978-1-4008-8197-0. doi: 10.1515/9781400881970-018. URL https://www.degruyter.com/document/doi/10.1515/9781400881970-018/html.

Maxim Shcherbakov, Adriaan Brebels, N.L. Shcherbakova, Anton Tyukov, T.A. Janovsky, and V.A. Kamaev. A survey of forecast error measures. *World Applied Sciences Journal*, 24:171–176, January 2013. doi: 10.5829/idosi.wasj.2013.24.itmies.80032.

Jort Sinninghe Damste and Rob Euwals. Modelbeschrijving: het CPB-woningprijsmodel. *Modelbeschrijving: het CPB-woningprijsmodel*, page 12, July 2023. URL `https://www.cpb.nl/modelbeschrijving-het-cpb-woningprijsmodel#:~:text=Het%20model%20zal%20gebruikt%20worden,zoals%20aanpassingen%20van%20de%20hypotheekrenteaftrek`.

Vincent Stamer. Thinking outside the container: a machine learning approach to forecasting trade flows. *Kiel Working Papers*, 2021. URL `https://ideas.repec.org//p/zbw/ifwkwp/2179.html`. Number: 2179 Publisher: Kiel Institute for the World Economy (IfW Kiel).

Leonard J. Tashman. Out-of-sample tests of forecasting accuracy: an analysis and review. *International Journal of Forecasting*, 16(4):437–450, October 2000. ISSN 0169-2070. doi: 10.1016/S0169-2070(00)00065-0. URL `https://www.sciencedirect.com/science/article/pii/S0169207000000650`.

G. V. Trunk. A problem of dimensionality, a simple example. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(3):306–307, 1979. ISSN 1939-3539. doi: 10.1109/TPAMI.1979.4766926. URL `https://ieeexplore.ieee.org/document/4766926`. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.

Jorge R. Vergara and Pablo A. Estévez. A review of feature selection methods based on mutual information. *Neural Computing and Applications*, 24(1):175–186, 2014. ISSN 1433-3058. doi: 10.1007/s00521-013-1368-0. URL `https://doi.org/10.1007/s00521-013-1368-0`.

P. Vries and Peter Boelhouwer. Equilibrium between interest payments and income in the housing market. *Journal of Housing and the Built Environment*, 24:19–29, April 2008. doi: 10.1007/s10901-008-9131-z.

E. Roy Weintraub. *Microfoundations: The Compatibility of Microeconomics and Macroeconomics*. Cambridge University Press, February 1979. ISBN 978-0-521-29445-4.

C. Willmott and K Matsuura. Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. *Climate Research*, 30:79, December 2005. doi: 10.3354/cr030079.

Alice Zheng and Amanda Casari. *Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists*. "O'Reilly Media, Inc.", March 2018. ISBN 978-1-4919-5319-8. Google-Books-ID: sthSDwAAQBAJ.

Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and Information Systems*, 41(3):647–665, 2014. ISSN 0219-3116. doi: 10.1007/s10115-013-0679-x. URL `https://doi.org/10.1007/s10115-013-0679-x`.