Borup, Daniel; Goulet Coulombe, Philippe; Rapach, David E.; Montes Schütte, Erik Christian; Schwenk-Nebbe, Sander

Working Paper

The anatomy of out-of-sample forecasting accuracy

Working Paper, No. 2022-16

Provided in Cooperation with:
Federal Reserve Bank of Atlanta

Suggested Citation: Borup, Daniel; Goulet Coulombe, Philippe; Rapach, David E.; Montes Schütte, Erik Christian; Schwenk-Nebbe, Sander (2022) : The anatomy of out-of-sample forecasting accuracy, Working Paper, No. 2022-16, Federal Reserve Bank of Atlanta, Atlanta, GA, https://doi.org/10.29338/wp2022-16

This Version is available at:
https://hdl.handle.net/10419/270459

FEDERAL RESERVE BANK *of* ATLANTA **WORKING PAPER SERIES**

The Anatomy of Out-of-Sample Forecasting Accuracy

Daniel Borup, Philippe Goulet Coulombe, David E. Rapach,
Erik Christian Montes Schütte, and Sander Schwenk-Nebbe

Working Paper 2022-16
November 2022

Abstract: We develop metrics based on Shapley values for interpreting time-series forecasting models, including "black-box" models from machine learning. Our metrics are model agnostic, so that they are applicable to any model (linear or nonlinear, parametric or nonparametric). Two of the metrics, iShapley-VI and oShapley-VI, measure the importance of individual predictors in fitted models for explaining the in-sample and out-of-sample predicted target values, respectively. The third metric is the performance-based Shapley value (PBSV), our main methodological contribution. PBSV measures the contributions of individual predictors in fitted models to the out-of-sample loss and thereby anatomizes out-of-sample forecasting accuracy. In an empirical application forecasting US inflation, we find important discrepancies between individual predictor relevance according to the in-sample iShapley-VI and out-of-sample PBSV. We use simulations to analyze potential sources of the discrepancies, including overfitting, structural breaks, and evolving predictor volatilities.

JEL classification: C22, C45, C53, E37, G17

Key words: variable importance, out-of-sample performance, Shapley value, loss function, machine learning, inflation

https://doi.org/10.29338/wp2022-16

The authors thank seminar and conference participants at the European Commission Joint Research Center: Online Seminar, 2022 International Symposium on Forecasting, and Workshop on Advances in Alternative Data and Machine Learning for Macroeconomics and Finance, as well as Daniele Bianchi, Giulio Caperna, Todd Clark, Marco Colagrossi, Claudia Foroni (Workshop on Advances in Alternative Data and Machine Learning discussant), Nikolay Gospodinov, Andreas Joseph, Juri Marcucci, Michael McCracken, Marcelo Medeiros, Stig Møller, and Mirco Rubin, for insightful comments. The authors created the Python package anatomy to compute the metrics for interpreting fitted prediction models developed in this paper. The views expressed here are those of the authors and not necessarily those of the Federal Reserve Bank of Atlanta or the Federal Reserve System. Any remaining errors are the authors' responsibility.

Daniel Borup is with Aarhus University and CREATES. Philippe Goulet Coulombe is with the Université du Québec à Montréal. David E. Rapach is with the Federal Reserve Bank of Atlanta. Erik Christian Montes Schütte is with Aarhus University, CREATES, and DFI. Sander Schwenk-Nebbe is with Aarhus University. Please address questions regarding content to David Rapach, Research Department, Federal Reserve Bank of Atlanta, 1000 Peachtree Street NE, Atlanta, GA 30309, dave.rapach@gmail.com.

1. Introduction

The use of large datasets (i.e., "big data") and machine learning for out-of-sample time-series forecasting in macroeconomics and finance is burgeoning. Indeed, there is growing evidence that the combination of large datasets and machine learning significantly improves out-of-sample performance. Macroeconomic applications include forecasting inflation, output and employment growth, the unemployment rate, unemployment insurance initial claims, and recessions,[1] while applications in finance often investigate stock return prediction.[2] Large datasets allow researchers to draw on a wealth of information, thereby increasing the capacity of prediction models to incorporate relevant signals. Machine learning offers a variety of tools for guarding against overfitting, which is vital for improving out-of-sample performance in the presence of a large number of predictors.[3] Some classes of machine-learning models—such as random forests (Breiman 2001) and neural networks—accommodate general forms of nonlinearities in predictive relations, further increasing the scope for improving out-of-sample performance when nonlinerities are an important attribute of the data-generating process (DGP).[4]

While researchers are certainly concerned with improving out-of-sample forecasting accuracy, they are also keenly interested in *interpreting* fitted prediction models. For example, especially with a large number of predictors, it is important to identify which predictors

[1]See, for example, Li and Chen (2014), Exterkate et al. (2016), Medeiros and Mendes (2016), Döpke, Fritsche, and Pierdzioch (2017), Kim and Swanson (2018), Smeekes and Wijler (2018), Medeiros et al. (2021), Vrontos, Galakis, and Vrontos (2021), Yousuf and Ng (2021), Borup and Schütte (2022), Goulet Coulombe (2022), Goulet Coulombe et al. (2022), Borup, Rapach, and Schütte (forthcoming), and Hauzenberger, Huber, and Klieber (forthcoming).

[2]See, for example, Chinco, Clark-Joseph, and Ye (2019), Rapach et al. (2019), Freyberger, Neuhierl, and Weber (2020), Gu, Kelly, and Xiu (2020), Dong et al. (2022), Han et al. (2022), Avramov, Cheng, and Metzker (forthcoming), and Chen, Pelger, and Zhu (forthcoming).

[3]Stock and Watson (2002a,b) spurred a literature that uses large datasets for macroeconomic forecasting based on principal component regression (e.g., Stock and Watson 1999b; Bernanke and Boivin 2003; Banerjee and Marcellino 2006). Studies that use large datasets and principal component regression to predict stock returns include Ludvigson and Ng (2007), Neely et al. (2014), Çakmaklı and van Dijk (2016), and Dong et al. (2022).

[4]Earlier studies that investigate nonlinear approaches to macroeconomic modeling and forecasting include Lee, White, and Granger (1993), Kuan and White (1994), Swanson and White (1997), Stock and Watson (1999a), Trapletti, Leisch, and Hornik (2000), Moshiri and Cameron (2000), Nakamura (2005), Medeiros, Teräsvirta, and Rech (2006), and Marcellino (2008); see Teräsvirta (2006) for a survey of the earlier literature.

are the most important for determining the forecasts generated by fitted models. It is also valuable to know how the predictors contribute to out-of-sample forecasting accuracy. Such knowledge helps users of forecasting models to wrap their minds around the models, so that they are not simply black boxes that opaquely transform predictors into forecasts. By identifying the most relevant predictors in fitted models that perform well out of sample, researchers gain insight into empirically important economic mechanisms that can help guide the assessment and development of theoretical models. In a similar vein, researchers involved in policy need to be able to interpret forecasting models to provide more accessible advice to policymakers.

An array of tools has been developed for interpreting fitted prediction models. Many of the methods are model agnostic, so that they can be applied to any model. One set of tools analyzes how individual predictors relate to the predictions generated by fitted models. Such methods include partial dependence (PD) plots (Friedman 2001), Shapley values (Shapley 1953; Štrumbelj and Kononenko 2010, 2014; Lundberg and Lee 2017), individual conditional expectation (ICE) plots (Goldstein et al. 2015), locally interpretable model-agnostic explanations (LIME, Ribeiro, Singh, and Guestrin 2016), and accumulated local effects (ALE, Apley and Zhu 2020). A related set of tools measures *variable importance*, namely, how important individual predictors are in accounting for the predictions produced by fitted models. Variable-importance metrics include those based on PD plots (Greenwell, Boehmke, and McCarthy 2018), permutations (Fisher, Rudin, and Dominici 2019), and Shapley values (Lundberg and Lee 2017; Casalicchio, Molnar, and Bischl 2018).

Tools for interpreting fitted forecasting models are typically applied in a manner that is appropriate for cross-sectional data. Specifically, a researcher divides the total sample of observations into training and test samples. The researcher then fits a prediction model using data from the training sample and uses the fitted model to generate predictions for the observations in the test sample. To interpret the model that generates the forecasts, the researcher computes, for example, the variable importance for each predictor based on the

fitted model and training data used to estimate the model. This conventional approach is eminently reasonable, especially in a cross-sectional context.[5] However, it is not necessarily appropriate in a *time-series* setting. In such a setting, a researcher typically re-estimates the prediction model each period using an expanding or rolling window of data, as they generate a sequence of out-of-sample forecasts. Thus, instead of a single model, there is a sequence of estimated models to interpret. The importance of the predictors in explaining the sequence of out-of-sample forecasts is likely to be of interest. Moreover, because researchers are concerned with out-of-sample performance, they will be interested in understanding how the individual predictors contribute to out-of-sample forecasting accuracy.

In this paper, we propose metrics for interpreting time-series forecasting models. The metrics are all based on Shapley values. Using insights from coalitional game theory, Shapley values fairly allocate contributions among predictors and have attractive properties for analyzing predictor relevance (as discussed in Section 2). The first metric is iShapley-VI$_p$, an in-sample variable-importance measure for predictor p. This is an aggregate measure of an individual predictor's importance across the entire set of fitted models that generate the sequence of out-of-sample time-series forecasts. The next metric is oShapley-VI$_p$, which measures the importance of predictor p for the sequence of out-of-sample forecasts. The final metric is the *performance-based Shapley value* (PBSV$_p$), our main methodological contribution. The iShapley-VI$_p$ and oShapley-VI$_p$ metrics are indifferent to the distance to the realized target value; in contrast, PBSV$_p$ measures the contribution of predictor p to the out-of-sample *loss* for the forecast evaluation period (although it can also be computed for any subsample of the forecast evaluation period), thereby taking into account the realized target value. In essence, PBSV$_p$ allows us to anatomize out-of-sample forecasting accuracy. PBSV$_p$ applies to any loss function, including the popular mean squared error (MSE), root mean squared error (RMSE), and mean absolute error (MAE) criteria. All of our metrics are model agnostic, so that they can be applied to any forecasting model (linear or nonlin-

[5]For example, this approach is used on numerous occasions for the applications in the insightful textbook by Molnar (2022).

ear, parametric or nonparametric). In summary, our metrics provide an informative set of tools for interpreting time-series forecasting models. To facilitate their implementation, we develop computationally efficient algorithms for computing oShapley-VI$_p$ and PBSV$_p$.

We illustrate the use of iShapley-VI$_p$, oShapley-VI$_p$, and PBSV$_p$ for analyzing a variety of fitted models in an empirical application forecasting US inflation. Inflation forecasting is the subject of a sizable literature and an important topic in many settings, including for central banks when crafting monetary policy.[6] A spate of recent studies finds that nonlinear machine-learning models, including random forests and neural networks, significantly improve inflation forecasts (e.g., Medeiros et al. 2021; Goulet Coulombe 2022; Goulet Coulombe et al. 2022; Hauzenberger, Huber, and Klieber forthcoming). We generate inflation forecasts using a set of more than 130 predictors—primarily from the FRED-MD database (McCracken and Ng 2016)—and a variety of forecasting strategies—including principal component regression, elastic net (ENet, Zou and Hastie 2005) estimation of a large-scale linear model, random forests, and neural networks—as well as ensemble forecasts based on the different strategies. Although the purpose of the application is not necessarily to generate the "best" inflation forecasts, the different forecasting strategies often outperform a relevant benchmark forecast by significant margins, so that the out-of-sample gains are substantial.

Applying our metrics to the fitted prediction models, we make two primary findings. First, there is considerable overlap between the importance of the individual predictors based on iShapley-VI$_p$ and oShapley-VI$_p$. This is perhaps not surprising, as the fitted models used in determining the importance of individual predictors for the in-sample and out-of-sample predicted target values are the same. Second, there are often substantive discrepancies between the relevance of individual predictors according to the in-sample iShapley-VI$_p$ and out-of-sample PBSV$_p$. In a number of cases, predictors that are among the most important according to iShapley-VI$_p$ contribute adversely to out-of-sample forecasting accuracy according to PBSV$_p$.[7] Differences in iShapley-VI$_p$ and PBSV$_p$ can arise due to overfitting and/or,

[6] See Faust and Wright (2013) for a survey of the inflation forecasting literature.

[7] Section A.3 of the Online Appendix reports results for another empirical application forecasting the US

in a time-series context, changes in the DGP. The discrepancies between iShapley-VI$_p$ and PBSV$_p$ in our empirical application serve as a warning: the in-sample importance of individual predictors in determining the predicted target values does not necessarily align with the predictors' roles in determining out-of-sample forecasting accuracy, even when a forecasting strategy performs well.

Finally, we conduct simulations to glean insight into potential causes of the discrepancies between iShapley-VI$_p$ and PBSV$_p$, including overfitting, structural breaks in slope coefficients, and evolving predictor volatilities. Based on sample sizes and DGPs that accord with the empirical application, we find that overfitting, structural breaks, and evolving predictor volatilities all provide plausible explanations for differences in iShapley-VI$_p$ and PBSV$_p$. The DGPs can account for the differences in rankings of predictor relevance based on iShapley-VI$_p$ and PBSV$_p$ that we find in the data. Of course, because we do not know the actual DGP that generates inflation, we cannot definitively determine the causes of the discrepancies. Nevertheless, the simulations shed light on plausible explanations for the discrepancies between iShapley-VI$_p$ and PBSV$_p$ in the data.

The rest of the paper is organized as follows. Section 2 describes the iShapley-VI$_p$, oShapley-VI$_p$, and PBSV$_p$ metrics for analyzing predictor relevance in a time-series context. Section 3 presents our empirical application, while Section 4 reports simulation results for analyzing potential reasons for differences in predictor relevance based on iShapley-VI$_p$ and PBSV$_p$. Section 5 concludes. We created the `Python` package `anatomy` to implement the algorithms for computing oShapley-VI$_p$ and PBSV$_p$.

2. Methodology

This section describes our methodology for measuring the relevance of individual predictors in time-series forecasting models. We begin with a discussion of Shapley values (Shapley

equity premium; see Koijen and Van Nieuwerburgh (2011) and Rapach and Zhou (2013, 2022) for surveys of the voluminous literature on equity premium predictability.

1953), as they form the foundation for our approach. We then define in-sample and out-of-sample variable-importance measures based on Shapley values. Finally, we propose the PBSV_p measure for analyzing the contributions of predictors to out-of-sample forecasting accuracy.

We use the following notation in our time-series context. We index individual predictors by p and collect the predictors in the index set $S = \{1, \ldots, P\}$. The period-t P-dimensional vector of predictor observations is denoted by $\boldsymbol{x}_t = [\ x_{1,t} \quad \cdots \quad x_{P,t}\ ]'$. The prediction model is given by

$$y_{t+1:t+h} = f(\boldsymbol{x}_t) + \varepsilon_{t+1:t+h}, \tag{1}$$

where $y_{t+1:t+h} = (1/h)\sum_{k=1}^{h} y_{t+k}$ is the target, h is the forecast horizon, f is the conditional mean (i.e., prediction) function, and $\varepsilon_{t+1:t+h}$ is a zero-mean disturbance term.[8] We denote the fitted prediction model by $\hat{f}$, while $W_i = \{t_{i,\text{start}}, \ldots, t_{i,\text{end}}\}$ denotes the set of observations used to train the model based on window W_i. The fitted prediction model evaluated at $\boldsymbol{x}_t$ and trained using W_i for horizon h is denoted by $\hat{f}(\boldsymbol{x}_t\,; W_i, h)$.

2.1. Shapley Values

Shapley values draw on coalitional game theory to utilize the analogy between the predictors (or features) in a model and players in a cooperative game earning payoffs, where the payoff corresponds to an individual predictor's contribution to the model's prediction. In a time-series setting, the goal is to explain the prediction $\hat{f}(\boldsymbol{x}_t\,; W_i, h)$ in terms of the marginal contribution of each predictor $x_{p,t}$ for $p \in S$, given the presence of all of the other predictors $(S \setminus \{p\})$. Viewed through the lens of coalitional game theory, Shapley values provide a means for fairly allocating the contributions among predictors.

Adapting Štrumbelj and Kononenko (2010, 2014) to our time-series context, the Shapley

[8] It is straightforward to extend the notation to allow for the conditional mean function in Equation (1) to include additional lags of $\boldsymbol{x}_t$.

value for predictor p and instance $\boldsymbol{x}_t$ for a model trained using window W_i for horizon h is given by

$$\phi_p(\boldsymbol{x}_t\,; W_i, h) = \sum_{Q \subseteq S \setminus \{p\}} \frac{|Q|!(P - |Q| - 1)!}{P!} \big[\xi_{Q \cup \{p\}}(\boldsymbol{x}_t\,; W_i, h) - \xi_Q(\boldsymbol{x}_t\,; W_i, h)\big] \tag{2}$$

for $p \in S$ and $t \in W_i$, where Q is a subset of predictors (i.e., a coalition), $Q \subseteq S \setminus \{p\}$ is the set of all possible coalitions of $P - 1$ predictors in S that exclude predictor p, $|Q|$ is the cardinality of Q, $|Q|!(P - |Q| - 1)!/P!$ is a combinatorial weight, and

$$\xi_Q(\boldsymbol{x}_t\,; W_i, h) = \mathbb{E}\Big[\hat{f} \,\Big|\, X_{j,t} = x_{j,t}\, \forall\, j \in Q\,; W_i, h\Big]. \tag{3}$$

The expression $\xi_Q(\boldsymbol{x}_t\,; W_i, h)$ in Equation (3) is the prediction of the fitted model conditional on the predictors in coalition Q, so that $\xi_{Q \cup \{p\}}(\boldsymbol{x}_t\,; W_i, h) - \xi_Q(\boldsymbol{x}_t\,; W_i, h)$ in Equation (2) measures the change in the prediction, conditional on the predictors in coalition Q, when the predictor p is included in the conditioning information set. The difference $\xi_{Q \cup \{p\}}(\boldsymbol{x}_t\,; W_i, h) - \xi_Q(\boldsymbol{x}_t\,; W_i, h)$ is computed for all possible coalitions of $P - 1$ predictors that exclude predictor p, with each quantity receiving the weight $|Q|!(P - |Q| - 1)!/P!$ in the summation in Equation (2) (the weights sum to one). In essence, the Shapley value uses coalitions to control for the other predictors when measuring the contribution of predictor p to the prediction corresponding to instance $\boldsymbol{x}_t$.

The Shapley value in Equation (2) has a number of attractive properties, including the following:

- Efficiency: $\sum_{p \in S} \phi_p(\boldsymbol{x}_t\,; W_i, h) = \hat{f}(\boldsymbol{x}_t\,; W_i, h) - \mathbb{E}\Big[\hat{f}\,; W_i, h\Big]$;
- Missingness: $\forall\, R \subseteq S \setminus \{p\} : \xi_{R \cup \{p\}}(\boldsymbol{x}_t\,; W_i, h) = \xi_R(\boldsymbol{x}_t\,; W_i, h) \Rightarrow \phi_p(\boldsymbol{x}_t\,; W_i, h) = 0$;
- Symmetry: $\forall\, R \subseteq S \setminus \{p, q\} : \xi_{R \cup \{p\}}(\boldsymbol{x}_t\,; W_i, h) = \xi_{R \cup \{q\}}(\boldsymbol{x}_t\,; W_i, h) \Rightarrow \phi_p(\boldsymbol{x}_t\,; W_i, h) = \phi_q(\boldsymbol{x}_t\,; W_i, h)$;

- Linearity: for any real numbers c_1 and c_2 and models $\hat{f}(\boldsymbol{x}_t\,;W_i,h)$ and $\hat{f}'(\boldsymbol{x}_t\,;W_i,h)$, $\phi_p\Big(c_1\Big[\hat{f}(\boldsymbol{x}_t\,;W_i,h)+c_2\hat{f}'(\boldsymbol{x}_t\,;W_i,h)\Big]\Big)=c_1\phi_p\Big(\hat{f}(\boldsymbol{x}_t\,;W_i,h)\Big)+c_1c_2\phi_p\Big(\hat{f}'(\boldsymbol{x}_t\,;W_i,h)\Big)$.

Efficiency, also known as local accuracy, says that we can exactly decompose the prediction corresponding to instance $\boldsymbol{x}_t$ (in terms of deviation from the average prediction) into the sum of the Shapley values for the individual predictors for that instance. Missingness and symmetry are intuitively appealing properties, while linearity is useful for computing Shapley values for ensembles of prediction models.

It is practically infeasible to compute the exact Shapley value in Equation (2) for even a moderate number of predictors, as the prediction function has to be evaluated for all possible coalitions both with and without predictor p. Building on the sampling-based approach of Castro, Gómez, and Tejada (2009), Štrumbelj and Kononenko (2014) develop an algorithm for estimating $\phi_p(\boldsymbol{x}_t\,;W_i,h)$. We use a refined version of their algorithm. We first express Equation (2) in the equivalent form:

$$\phi_p(\boldsymbol{x}_t\,;W_i,h)=\frac{1}{P!}\sum_{\mathcal{O}\,\in\,\pi(P)}\Big[\xi_{\mathrm{Pre}_p(\mathcal{O})\,\cup\,\{p\}}(\boldsymbol{x}_t\,;W_i,h)-\xi_{\mathrm{Pre}_p(\mathcal{O})}(\boldsymbol{x}_t\,;W_i,h)\Big] \tag{4}$$

for $p\in S$ and $t\in W_i$, where $\mathcal{O}$ is an ordered permutation for the predictor indices in S, $\pi(P)$ is the set of all ordered permutations for S, and $\mathrm{Pre}_p(\mathcal{O})$ is the set of indices that precede p in $\mathcal{O}$.

The algorithm is based on making a random draw with replacement for an ordered permutation from $\pi(P)$, which we denote by $\mathcal{O}_m$. Using $\mathcal{O}_m$, we compute the following measure:

$$\begin{aligned}\theta_{p,m}(\boldsymbol{x}_t\,;W_i,h)=\frac{1}{|W_i|}\sum_{s\,\in\,W_i}\Big[&\hat{f}(\boldsymbol{x}_{j,t}:j\in\mathrm{Pre}_p(\mathcal{O}_m)\cup\{p\},\,\boldsymbol{x}_{k,s}:k\in\mathrm{Post}_p(\mathcal{O}_m)\,;W_i,h)-\\&\hat{f}(\boldsymbol{x}_{j,t}:j\in\mathrm{Pre}_p(\mathcal{O}_m),\,\boldsymbol{x}_{k,s}:k\in\mathrm{Post}_p(\mathcal{O}_m)\cup\{p\}\,;W_i,h)\Big]\end{aligned} \tag{5}$$

for $p\in S$ and $t\in W_i$, where $\mathrm{Post}_p(\mathcal{O})$ is the set of indices that follow p in $\mathcal{O}$. Equation (5) approximates the effect of removing predictors that are not in the coalition by replacing them

with background data from the training sample (Štrumbelj and Kononenko 2014; Lundberg and Lee 2017).[9] The estimate of $\phi_p(\boldsymbol{x}_t\,; W_i, h)$ in Equation (4) is then given by

$$\hat{\phi}_p(\boldsymbol{x}_t\,; W_i, h) = \frac{1}{2M}\sum_{m=1}^{2M}\theta_{p,m}(\boldsymbol{x}_t\,; W_i, h) \tag{6}$$

for $p \in S$ and $t \in W_i$, where M is the number of random draws. To increase computational efficiency, we follow Castro, Gómez, and Tejada (2009) and compute Shapley values for each predictor $p \in S$ for a randomly drawn ordered permutation from $\pi(P)$. In addition, we implement antithetic sampling as a variance-reduction technique by computing $\theta_{p,m}(\boldsymbol{x}_t\,; W_i, h)$ in Equation (5) for the original order of a randomly drawn ordered permutation, as well as when the order is reversed (Mitchell et al. 2022).[10] Equation (6) retains the properties in Section 2.1, including efficiency, so that

$$\sum_{p\,\in\,S}\hat{\phi}_p(\boldsymbol{x}_t\,; W_i, h) = \hat{f}(\boldsymbol{x}_t\,; W_i, h) - \underbrace{\bar{\hat{f}}(W_i, h)}_{\hat{\phi}_\emptyset(W_i,h)} \tag{7}$$

for $t \in W_i$, where $\bar{\hat{f}}(W_i, h) = (1/|W_i|)\sum_{t\,\in\,W_i}\hat{f}(\boldsymbol{x}_t\,; W_i, h)$ is the average in-sample prediction for the model trained using sample W_i, which corresponds to the unconditional forecast (i.e., the forecast based on the empty coalition set, denoted by $\hat{\phi}_\emptyset(W_i, h)$).

Suppose that the prediction model is linear in the predictors: $f(\boldsymbol{x}_t) = \alpha + \sum_{p=1}^{P}\beta_p x_{p,t}$; the fitted prediction model is given by $\hat{f}(\boldsymbol{x}_t) = \hat{\alpha} + \sum_{p=1}^{P}\hat{\beta}_p x_{p,t}$, where $\hat{\alpha}, \hat{\beta}_1, \ldots, \hat{\beta}_P$ are estimates of $\alpha, \beta_1, \ldots, \beta_P$, respectively. In this case, the Shapley value in Equation (4) is

[9] "Background data" refer to the data that are used to integrate out the predictors that are not in the coalition when estimating the conditional expectation in Equation (3). Equation (5) effectively samples from the empirical marginal distribution based on the training sample for the predictors not in the coalition, which implicitly assumes that the predictors not in the coalition are distributed independently of those in the coalition. Because this assumption is not likely to hold in practice, Lundberg and Lee (2017) propose sampling from the empirical conditional distribution for the predictors not in the coalition. However, using insights from Pearl (2009), Janzing, Minorics, and Blöbaum (2020) argue that, to fairly allocate the contributions across the individual predictors, it is more appropriate to use the empirical marginal distribution, as in Equation (5).

[10] The algorithm in Equations (5) and (6) can be implemented with these enhancements in the `Permutation Explainer` in the `SHAP` package in `Python` when `max_samples` is set to the number of instances in the training sample.

given by

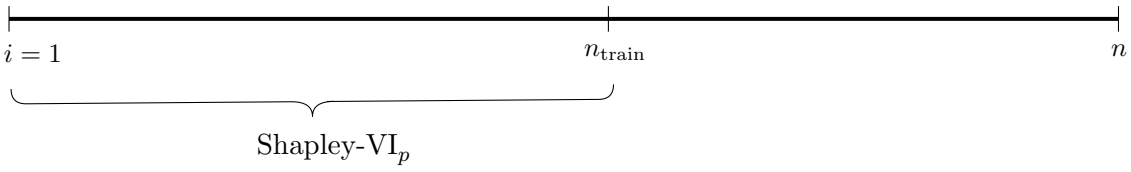
$$\hat{\phi}_p(\boldsymbol{x}_t\,; W_i, h) = \hat{\beta}_p(x_{p,t} - \bar{x}_p) \tag{8}$$

for $p \in S$ and $t \in W_i$, where $\bar{x}_p$ is the sample mean of $x_{p,t}$ for the training sample. Because there are no interactions for a linear model, it is straightforward to compute the Shapley values via Equation (8).[11]

The Shapley value $\hat{\phi}_p(\boldsymbol{x}_t\,; W_i, h)$ provides a local measure of the contribution of predictor p to the prediction corresponding to instance $\boldsymbol{x}_t$ in the training sample. A global measure of the importance of predictor p can be computed by taking the average of the absolute values of the Shapley values for predictor p across the training sample observations:

$$\text{Shapley-VI}_p(W_i, h) = \frac{1}{|W_i|} \sum_{t \in W_i} \left| \hat{\phi}_p(\boldsymbol{x}_t\,; W_i, h) \right| \tag{9}$$

for $p \in S$. The variable-importance measure in Equation (9) is a popular metric for assessing predictor importance in machine-learning applications (e.g., Molnar 2022, Chapter 9.6). Equation (9) is based on a single training sample. Tools for interpreting fitted models are typically applied in this manner, which is appropriate for cross-sectional data (or time-series data if a researcher only estimates the prediction model once). The following diagram illustrates the conventional case for cross-sectional data indexed by $i = 1, \ldots, n$, where the first n_{train} observations comprise the training sample.



In a time-series context, however, researchers often re-estimate the model on a regular basis

[11] Similarly, suppose that the fitted prediction model is a general additive model (GAM): $f(\boldsymbol{x}_t) = \sum_{p=1}^{P} f_p(x_{p,t})$; the fitted prediction model is given by $\hat{f}(\boldsymbol{x}_t) = \sum_{p=1}^{P} \hat{f}_p(x_{p,t})$. For a GAM, there are no interactions between the predictors, but each predictor can affect the target in a nonlinear fashion. In the case of a linear model or GAM, because there are no interactions, we can conveniently compute the Shapley value by running the algorithm with only one iteration.

over time as additional data become available, so that there are multiple training samples. In Section 2.2, we develop variable-importance metrics that are more suited to this practice.

2.2. In-Sample and Out-of-Sample Shapley Values

When forecasting time-series variables in macroeconomics and finance, it is common to regularly retrain the prediction model using data available at the time of forecast formation. For example, if we are forecasting a monthly variable at horizon h, we re-estimate the prediction model each month as additional data become available, which is typically done using either an expanding or rolling window, where the estimation sample becomes longer (remains the same size) for the former (latter). Suppose that there are $t = 1, \ldots, T$ total observations available. The initial in-sample period ends in $t = T_{\text{in}}$, while the remaining $T - T_{\text{in}} = D$ observations constitute the out-of-sample period.

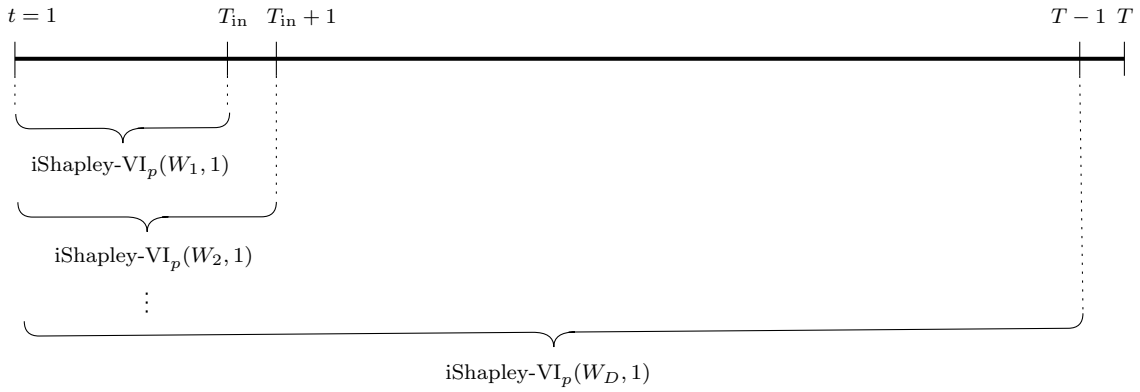
Mimicking the situation of a forecaster in real time, we proceed as follows. We first use data from $t = 1$ through $t = T_{\text{in}}$ to fit the prediction model and generate an out-of-sample forecast of $y_{T_{\text{in}}+1:T_{\text{in}}+h}$. After accounting for the forecast horizon and lag in Equation (1), there are $T_{\text{in}} - (h-1) - 1$ usable observations for training the prediction model for the first out-of-sample forecast. For an expanding (rolling) window, we then use data from $t = 1$ ($t = 2$) through $T_{\text{in}} + 1$ to fit the prediction model and generate a forecast of $y_{T_{\text{in}}+2:T_{\text{in}}+h+1}$. Continuing in this manner, we generate a sequence of $D - (h-1)$ out-of-sample forecasts, where, for the final forecast, we use data from the first period (period $T - D - (h-1)$) through $T - h$ for an expanding (rolling) window to fit the prediction model and generate a forecast of $y_{T-(h-1):T}$. Note that we only use data available at the time of forecast formation to train the model, so that there is no look-ahead bias in the out-of-sample forecasts.

The Shapley-based variable importance in Equation (9) corresponds to a prediction model trained once using the observations in W_i. To accommodate the sequence of $D - (h-1)$ time-series forecasts for models regularly retrained with an expanding or rolling window, we denote the set of training samples by $W = \{W_1, \ldots, W_{D-(h-1)}\}$. In this context, we define

the *in-sample Shapley-based variable importance* as

$$\text{iShapley-VI}_p(W, h) = \frac{1}{|W|} \sum_{i \in W} \text{Shapley-VI}_p(W_i, h) \tag{10}$$

for $p \in S$, which is the average of the variable-importance measures for predictor p across all of the training samples that are used to generate the sequence of time-series forecasts. To help make the temporal dimension of Equation (10) clear, the following diagram shows how $\text{iShapley-VI}_p(W, h)$ is computed in terms of the time-series observations for an expanding window and $h = 1$.



$$\text{iShapley-VI}_p(W, 1) = \frac{1}{D} \sum_{i=1}^{D} \text{iShapley-VI}_p(W_i, 1)$$

We are also interested in measuring variable importance for the sequence of out-of-sample forecasts. We begin by defining the Shapley value for the fitted model and vector of predictors used to generate an out-of-sample forecast, which corresponds to an out-of-sample version of Equation (4):

$$\phi_p^{\text{out}}\left(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h\right) = \frac{1}{P!} \sum_{\mathcal{O} \in \pi(P)} \left[\xi_{\text{Pre}_p(\mathcal{O}) \cup \{p\}}\left(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h\right) - \xi_{\text{Pre}_p(\mathcal{O})}\left(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h\right)\right] \tag{11}$$

for $p \in S$ and $i = 1, \ldots, D-(h-1)$, where $\boldsymbol{x}_{T_{\text{in}}+(i-1)}$ is the vector of predictors plugged into the fitted prediction model trained with W_i that is used to generate the ith out-of-sample forecast given by $\hat{y}_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} = \hat{f}\left(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h\right)$. To estimate Equation (11), we use

a suitably modified version of the algorithm in Section 2.1. For a random draw of an ordered permutation $\mathcal{O}_m$, we modify Equation (5) to

$$\theta^{\text{out}}_{p,m}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big) = \frac{1}{|W_i|}\sum_{s\in W_i}\Big[\hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j\in \text{Pre}_p(\mathcal{O}_m)\cup\{p\},\, \boldsymbol{x}_{k,s} : k\in \text{Post}_p(\mathcal{O}_m)\,;W_i,h\big) - \hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j\in \text{Pre}_p(\mathcal{O}_m),\, \boldsymbol{x}_{k,s} : k\in \text{Post}_p(\mathcal{O}_m)\cup\{p\}\,;W_i,h\big)\Big], \tag{12}$$

while Equation (6) becomes

$$\hat{\phi}^{\text{out}}_p\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big) = \frac{1}{2M}\sum_{m=1}^{2M}\theta^{\text{out}}_{p,m}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big) \tag{13}$$

for $p\in S$ and $i=1,\ldots,D-(h-1)$. Equation (12) continues to approximate the effect of removing predictors that are not in the coalition by replacing them with background data from W_i, as this is the sample that is used to train the prediction model that generates the out-of-sample forecast; in this sense, we remain *true to the model* that we use for forecasting.[12]

The $\hat{\phi}^{\text{out}}_p\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big)$ estimate in Equation (13) continues to be characterized by efficiency, so that we can decompose the out-of-sample forecast corresponding to $\boldsymbol{x}_{T_{\text{in}}+(i-1)}$ as follows:

$$\sum_{p\in S}\hat{\phi}_p\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big) = \hat{f}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big) - \hat{\phi}_{\emptyset}(W_i,h). \tag{14}$$

For a model that is linear in the predictors, the Shapley value in Equation (11) is given by

$$\hat{\phi}^{\text{out}}_p\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big) = \hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big) \tag{15}$$

for $p\in S$ and $i=1,\ldots,D-(h-1)$, where $\hat{\beta}_p$ and $\bar{x}_p$ are again the estimate of β_p and

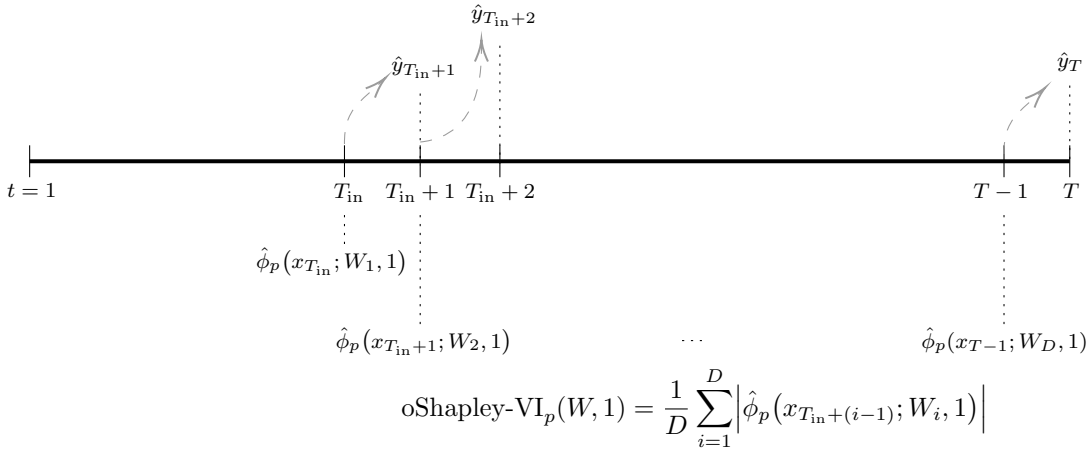
[12] "True to the model" means that we use parameter estimates from the fitted prediction model and background data from the training sample that is used to fit the prediction model. In other words, we retain the basic elements of the fitted model when estimating the Shapley value in Equation (13).

sample mean of $x_{p,t}$, respectively, based on the training sample.

Taking the absolute value of $\hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,; W_i, h\big)$ in Equation (13) produces a Shapley-based variable-importance measure for predictor p and a particular out-of-sample forecast. To compute the variable importance for p for the entire sequence of out-of-sample forecasts, we proceed analogously to the in-sample Shapley-based variable importance in Equation (10) and define the *out-of-sample Shapley-based variable importance* by taking the average of the absolute values of Equation (13) across the out-of-sample forecasts:

$$\text{oShapley-VI}_p(W, h) = \frac{1}{|W|} \sum_{i \in W} \left| \hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,; W_i, h\big) \right| \tag{16}$$

for $p \in S$. The following diagram depicts how the time-series observations are incorporated into Equation (16) for an expanding window and $h = 1$.



2.3. Performance-Based Shapley Values

Out-of-sample forecasts are typically assessed using a loss function. Accordingly, we propose PBSV_p to decompose the loss over the out-of-sample period into the components attributable to the individual predictors $p \in S$.

The key insight for computing PBSV_p is to wrap a loss function around the predictions

in Equation (12). We denote a generic loss function by

$$L\big(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \underbrace{\hat{y}_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}}_{\hat{f}\left(\boldsymbol{x}_{T_{\text{in}}+(i-1)};W_i,h\right)}\big). \tag{17}$$

To incorporate the loss function, we further modify the algorithm. For a random draw of an ordered permutation $\mathcal{O}_m$, we adjust Equation (12) as follows:

$$\begin{aligned}
&\theta^{\text{out}}_{p,m}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h,L\big) = \\
&L\left(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \frac{1}{|W_i|}\sum_{s\,\in\,W_i}\hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j\in \text{Pre}_p(\mathcal{O}_m)\cup\{p\},\, \boldsymbol{x}_{k,s} : k\in\text{Post}_p(\mathcal{O}_m)\,;W_i,h\big)\right) - \\
&L\left(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \frac{1}{|W_i|}\sum_{s\,\in\,W_i}\hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j\in \text{Pre}_p(\mathcal{O}_m),\, \boldsymbol{x}_{k,s} : k\in\text{Post}_p(\mathcal{O}_m)\cup\{p\}\,;W_i,h\big)\right)
\end{aligned} \tag{18}$$

for $p \in S$ and $i = 1,\ldots,D-(h-1)$. Equation (13) becomes

$$\hat{\phi}^{\text{out}}_p\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h,L\big) = \frac{1}{2M}\sum_{m=1}^{2M}\theta^{\text{out}}_{p,m}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h,L\big) \tag{19}$$

for $p \in S$ and $i = 1,\ldots,D-(h-1)$. The local PBSV_p in Equation (19) measures the contribution of predictor p to the loss incurred by the ith out-of-sample forecast. Like Equation (12), Equation (18) approximates the effect of removing predictors that are not in the coalition by replacing them with background data from the training sample W_i, so that we continue to remain true to the model that generates the out-of-sample forecast. Based on the logic of Shapley values, the local PBSV_p in Equation (19) fairly allocates the loss among the predictors for the ith out-of-sample forecast. Equation (19) is characterized by efficiency:

$$\sum_{p\,\in\,S}\hat{\phi}^{\text{out}}_p\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h,L\big) = L\Big(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \hat{f}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,;W_i,h\big)\Big) - \hat{\phi}^{\text{out}}_{\emptyset}(W_i,h,L) \tag{20}$$

for $i = 1,\ldots,D-(h-1)$, where $\hat{\phi}^{\text{out}}_{\emptyset}(W_i,h,L)$ corresponds to the loss for the prediction

conditional on the empty coalition set.

Because the loss function can be nonlinear, for a prediction model that is linear in the predictors, we do not have a simple expression analogous to Equation (8) or Equation (15) for the local PBSV_p. Nevertheless, in the special case of a linear model, we can derive an analytical expression for the local PBSV_p for a specific loss function. For example, consider the squared error loss for the ith out-of-sample forecast:

$$L\big(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \hat{y}_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}\big) = \big(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - \hat{y}_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}\big)^2. \tag{21}$$

For a linear model and Equation (21), the local PBSV_p can be expressed as

$$\begin{aligned}\hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h, \text{SE}\big) = \underbrace{\hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big)}_{\hat{\phi}_p^{\text{out}}\left(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h\right)} \Big[\big(\hat{y}_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}\big) \\ - \Big(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - \hat{\phi}_{\emptyset}(W_i, h)\Big)\Big],\end{aligned} \tag{22}$$

where $\hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h\big) = \hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big)$ is from Equation (15). We can view $\hat{\phi}_{\emptyset}(W_i, h)$ in Equation (22) as a naïve forecast that ignores the information in the predictors and simply uses the sample mean of the target for the training sample as the prediction. For the squared error loss, the local PBSV_p measures the contribution of predictor p to the squared error for the forecast that incorporates the information in the predictors relative to the squared error for the naïve forecast that ignores the information. In the special case of a linear model, Equation (22) says that $\hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h, \text{SE}\big)$ is proportional to the error for the forecast based on the set of predictors—after adjusting for the naïve forecast error—where the factor of proportionality is given by $\hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big)$ (i.e., the Shapley value for predictor p and instance $\boldsymbol{x}_{T_{\text{in}}+(i-1)}$ for a linear model). Furthermore, the sign of $\hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h, \text{SE}\big)$ in Equation (22) depends on the signs of the term in brackets and $\hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big)$.

To gain some intuition for Equation (22), suppose that the linear model forecast is perfect

$(\hat{y}_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} = y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)})$. In addition, assume that the realized target value is greater than the naïve forecast ($y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} > \hat{\phi}_{\emptyset}(W_i, h)$), so that the term in brackets in Equation (22) is negative. If $\hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big) > 0$, then $\hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,; W_i, h, \text{SE}\big) < 0$. In this case, predictor p contributes to the forecast being higher than the naïve forecast—since $\hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big) > 0$—which is in line with the realized target value being greater than the naïve forecast; accordingly, the local PBSV_p in Equation (22) deems that predictor p contributes to lowering the squared error vis-à-vis the naïve forecast.[13]

We are primarily interested in the performance of the entire sequence of out-of-sample forecasts, so that we also define a global PBSV_p. To obtain the global PBSV_p, we again modify the algorithm. Specifically, we expand Equation (18) to reflect the average loss for the out-of-sample period:

$$\theta_{p,m}^{\text{out}}(W, h, L) =$$
$$\frac{1}{|W|}\sum_{i\,\in\, W} L\left(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \frac{1}{|W_i|}\sum_{s\,\in\, W_i} \hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j \in \text{Pre}_p(\mathcal{O}_m) \cup \{p\},\, \boldsymbol{x}_{k,s} : k \in \text{Post}_p(\mathcal{O}_m)\,; W_i, h\big)\right) -$$
$$\frac{1}{|W|}\sum_{i\,\in\, W} L\left(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \frac{1}{|W_i|}\sum_{s\,\in\, W_i} \hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j \in \text{Pre}_p(\mathcal{O}_m),\, \boldsymbol{x}_{k,s} : k \in \text{Post}_p(\mathcal{O}_m) \cup \{p\}\,; W_i, h\big)\right) \tag{23}$$

for $p \in S$. To remain true to the model, Equation (23) continues to approximate the effect of removing predictors that are not in the coalition by replacing them with background data from the training sample. Equation (19) is now given by

$$\hat{\phi}_p^{\text{out}}(W, h, L) = \frac{1}{2M}\sum_{m=1}^{2M} \theta_{p,m}^{\text{out}}(W, h, L) \tag{24}$$

for $p \in S$. The global PBSV_p in Equation (24) allows us to decompose the average loss for a

[13] Conversely, if $\hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big) < 0$, then $\hat{\phi}_p^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,; W_i, h, \text{SE}\big) > 0$. In this case, although the linear model forecast is perfect, the local PBSV_p deems that predictor p increases the squared error vis-à-vis the naïve forecast, as p contributes to the forecast being below the naïve forecast, while the realized target value is above the naïve forecast. A perfect forecast, $y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} > \hat{\phi}_{\emptyset}(W_i, h)$, and $\hat{\beta}_p\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big) < 0$ imply that there are one or more other predictors $q \neq p$ for which $\hat{\beta}_q\big(x_{p,T_{\text{in}}+(i-1)} - \bar{x}_p\big) > 0$ and $\hat{\phi}_q^{\text{out}}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,; W_i, h, \text{SE}\big) < 0$, as the other predictors contribute to the forecast being higher than the naïve forecast, ultimately producing the perfect forecast.

sequence of out-of-sample forecasts into the contributions of each of the P predictors. In this way, we anatomize out-of-sample performance by fairly assessing how the individual predictors contribute to out-of-sample forecasting accuracy. Equation (24) is again characterized by efficiency:

$$\sum_{p \in S} \hat{\phi}_p^{\text{out}}(W, h, L) = \frac{1}{|W|} \sum_{i \in W} L\Big(y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)}, \hat{f}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}; W_i, h\big)\Big) - \hat{\phi}_{\emptyset}^{\text{out}}(W, h, L), \quad (25)$$

where $\hat{\phi}_{\emptyset}^{\text{out}}(W, h, L)$ corresponds to the average loss for the sequence of forecasts based on the empty coalition set. Section A.1 of the Online Appendix provides details for our algorithm for computing PBSV_p. The algorithm can be used to compute PBSV_p for any fitted prediction model (as well as ensembles of prediction models) and any loss function.[14]

Our PBSV_p bears some resemblance to the Shapley feature importance (SFIMP) metric in Casalicchio, Molnar, and Bischl (2018), in that both measures are computed using a loss function for the out-of-sample observations. However, there are important differences between PBSV_p and SFIMP. SFIMP assumes that the prediction model is estimated only once, which is more appropriate for cross-sectional data, while PBSV_p is explicitly designed for time-series data when the out-of-sample forecasts are generated by a sequence of fitted models based on an expanding or rolling window. Furthermore, there are substantive differences in the algorithms used to compute PBSV_p and SFIMP (beyond the fact that the former is based on a sequence of fitted models, while the latter is not). For example, SFIMP uses background data from the test sample to control for predictors not in the coalition when computing Shapley values; in contrast, Equation (23) always uses background data from the training sample, so that we remain true to the fitted models that generate the out-of-sample forecasts.[15] In summary, PBSV_p provides a means for fairly allocating the out-of-sample loss for a sequence of time-series forecasts across the individual predictors, thereby shedding

[14] In addition to the entire out-of-sample period, PBSV_p in Equation (24) can also be computed for any subsample of interest for the forecast evaluation period (e.g., the Great Recession or COVID-19 pandemic).

[15] PBSV_p has a different focus from the "Shapley regressions" proposed by Joseph (2021). Shapley regressions relate the realized target values to Shapley values for the out-of-sample observations in a linear regression framework.

light on the anatomy of out-of-sample forecasting accuracy.

As an example of computing PBSV_p for a specific loss function, consider the MSE criterion:

$$\text{MSE} = \frac{1}{|W|} \sum_{i \in W} \Big[y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - \hat{f}\big(\boldsymbol{x}_{T_{\text{in}}+(i-1)}\,; W_i, h\big) \Big]^2. \tag{26}$$

To obtain the global PBSV_p for the MSE using the algorithm, we use the following version of Equation (23):

$$\begin{aligned}
&\theta^{\text{out}}_{p,m}(W, h, \text{MSE}) = \\
&\frac{1}{|W|} \sum_{i \in W} \left[y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - \frac{1}{|W_i|} \sum_{s \in W_i} \hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j \in \text{Pre}_p(\mathcal{O}_m) \cup \{p\},\, \boldsymbol{x}_{k,s} : k \in \text{Post}_p(\mathcal{O}_m)\,; W_i, h\big) \right]^2 - \\
&\frac{1}{|W|} \sum_{i \in W} \left[y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - \frac{1}{|W_i|} \sum_{s \in W_i} \hat{f}\big(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j \in \text{Pre}_p(\mathcal{O}_m),\, \boldsymbol{x}_{k,s} : k \in \text{Post}_p(\mathcal{O}_m) \cup \{p\}\,; W_i, h\big) \right]^2
\end{aligned} \tag{27}$$

for $p \in S$. Equation (24) is then given by

$$\hat{\phi}^{\text{out}}_p(W, h, \text{MSE}) = \frac{1}{2M} \sum_{m=1}^{2M} \theta^{\text{out}}_{p,m}(W, h, \text{MSE}) \tag{28}$$

for $p \in S$. According to the efficiency property,

$$\sum_{p \in S} \hat{\phi}^{\text{out}}_p(W, h, \text{MSE}) = \text{MSE} - \hat{\phi}^{\text{out}}_{\emptyset}(W, h, \text{MSE}). \tag{29}$$

Because it is expressed in the same units as the target, the RMSE (i.e., the square root of the MSE) is often reported. For the RMSE, we straightforwardly modify Equation (27)

as follows:

$$\theta^{\text{out}}_{p,m}(W,h,\text{RMSE}) = \left\{\frac{1}{|W|}\sum_{i\in W}\left[y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - \frac{1}{|W_i|}\sum_{s\in W_i}\hat{f}\left(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j\in \text{Pre}_p(\mathcal{O}_m)\cup\{p\},\, \boldsymbol{x}_{k,s} : k\in \text{Post}_p(\mathcal{O}_m)\,;W_i,h\right)\right]^2\right\}^{0.5} - \left\{\frac{1}{|W|}\sum_{i\in W}\left[y_{T_{\text{in}}+i:T_{\text{in}}+h+(i-1)} - \frac{1}{|W_i|}\sum_{s\in W_i}\hat{f}\left(\boldsymbol{x}_{j,T_{\text{in}}+(i-1)} : j\in \text{Pre}_p(\mathcal{O}_m),\, \boldsymbol{x}_{k,s} : k\in \text{Post}_p(\mathcal{O}_m)\cup\{p\}\,;W_i,h\right)\right]^2\right\}^{0.5}. \tag{30}$$

We have analogous versions of Equations (28) and (29):

$$\hat{\phi}^{\text{out}}_p(W,h,\text{RMSE}) = \frac{1}{2M}\sum_{m=1}^{2M}\theta^{\text{out}}_{p,m}(W,h,\text{RMSE}) \tag{31}$$

and

$$\sum_{p\in S}\hat{\phi}^{\text{out}}_p(W,h,\text{RMSE}) = \text{RMSE} - \hat{\phi}^{\text{out}}_{\emptyset}(W,h,\text{RMSE}), \tag{32}$$

respectively.[16]

3. Forecasting Inflation

In this section, we use the time-series metrics developed in Section 2 to analyze out-of-sample forecasts of US inflation. Inflation forecasting is an important topic in macroeconomics, including for policymakers. There is recent evidence that traditional inflation benchmark forecasts can be outperformed by the use of big data in conjunction with machine-learning methods, and the outperformance is largely attributable to nonlinearities (e.g., Medeiros et al. 2021; Goulet Coulombe 2022; Goulet Coulombe et al. 2022; Hauzenberger, Huber, and Klieber forthcoming). In addition to nonlinearities, forecasting inflation is an interesting

[16] We use $M = 500$ for the algorithms when computing iShapley-VI$_p$, oShapley-VI$_p$, and PBSV$_p$ for the empirical application in Section 3.

case study because there is evidence of structural breaks in inflation processes for numerous countries, including the United States (e.g., Stock and Watson 1996, 2003; O'Reilly and Whelan 2005; Bataa et al. 2013, 2014).

We begin with the following general prediction model for inflation:

$$\pi_{t+1:t+h} = f\left(\boldsymbol{\pi}^{\text{AR}}_{t-L:t}, \boldsymbol{w}_t, \boldsymbol{w}^{\text{MA}(q)}_t\right) + \varepsilon_{t+1:t+h}, \tag{33}$$

where $\pi_{t+1:t+h} = (1/h)\sum_{k=1}^{h}\pi_{t+k}$, $\pi_t = \log(\text{CPI}_t) - \log(\text{CPI}_{t-1})$, CPI_t is the month-t US consumer price index (CPI), $\boldsymbol{\pi}^{\text{AR}}_{t-L:t} = [\ \pi_t \ \cdots \ \pi_{t-L}\ ]'$ captures the autoregressive (AR) component in inflation, $\boldsymbol{w}_t$ is a vector of predictors, and $\boldsymbol{w}^{\text{MA}(q)}_t = (1/q)\sum_{k=1}^{q}\boldsymbol{w}_{t-(k-1)}$ is a vector of MAs of order q for the predictors in $\boldsymbol{w}_t$. The inclusion of MAs of the predictors is motivated by Goulet Coulombe et al. (2021), who find that MAs of predictors provide substantive out-of-sample gains for forecasting macroeconomic variables. We set $q = 3$, which allows predictors up to a quarter in the past to affect the prediction. In terms of the AR component, we set $L = 11$, corresponding to twelve lags of inflation in Equation (33). Based on Equation (33), the forecast of $\pi_{t+1:t+h}$ is given by

$$\hat{\pi}_{t+1:t+h} = \hat{f}\left(\boldsymbol{\pi}^{\text{AR}}_{t-L:t}, \boldsymbol{w}_t, \boldsymbol{w}^{\text{MA}(q)}_t\right), \tag{34}$$

where $\hat{f}$ is the fitted prediction model based on data through t.

A natural starting point for generating an inflation forecast based on multiple predictors is a linear predictive regression model:

$$\pi_{t+1:t+h} = \underbrace{\alpha + \boldsymbol{x}_t'\boldsymbol{\beta}}_{f(\boldsymbol{x}_t)} + \varepsilon_{t+1:t+h}, \tag{35}$$

where $\boldsymbol{x}_t = [\ \boldsymbol{\pi}^{\text{AR}}_{t-L:t}{}' \ \ \boldsymbol{w}_t' \ \ \boldsymbol{w}^{\text{MA}(q)\prime}_t\ ]'$, α is the intercept, and $\boldsymbol{\beta} = [\ \beta_1 \ \cdots \ \beta_P\ ]'$ is a P-dimensional vector of slope coefficients. It is straightforward to estimate Equation (35) via

ordinary least squares (OLS), leading to the forecast:

$$\hat{\pi}^{\text{OLS}}_{t+1:t+h} = \hat{\alpha}^{\text{OLS}} + \boldsymbol{x}_t' \hat{\boldsymbol{\beta}}^{\text{OLS}}, \tag{36}$$

where $\hat{\alpha}^{\text{OLS}}$ and $\hat{\boldsymbol{\beta}}^{\text{OLS}}$ are the OLS estimates of α and $\boldsymbol{\beta}$, respectively, in Equation (35) based on data through t. Although straightforward to compute, the forecast in Equation (36) tends to perform poorly in practice. By construction, OLS maximizes the fit of the model over the training sample, which can result in in-sample overfitting and thus poor out-of-sample performance. Because inflation contains a sizable unpredictable component, the signal-to-noise ratio is relatively small, so that the forecast in Equation (36) is likely to perform poorly, especially when P is large and the predictors are correlated.

3.1. Forecasting Strategies

We consider four strategies for improving inflation forecasts. The first two are based on linear specifications, while the last two allow for general nonlinearities in the prediction model. All of the approaches employ a rolling estimation window. We also consider ensemble forecasts that combine forecasts based on the different strategies.

3.1.1. Principal Component Regression

Our first forecasting approach is principal component regression (PCR). Beginning with Stock and Watson (2002a,b), there is an ample literature that uses PCR to forecast macroeconomic variables, including inflation. Let $\boldsymbol{z}_t = [\ z_{1,t} \ \cdots \ z_{C,t}\ ]'$ denote the vector containing the first C principal components corresponding to $\boldsymbol{x}_t$, where $C \ll P$. The PCR specification can be expressed as

$$\pi_{t+1:t+h} = \alpha_z + \boldsymbol{z}_t' \boldsymbol{\beta}_z + \varepsilon_{t+1:t+h}, \tag{37}$$

where $\boldsymbol{\beta}_z = [\ \beta_{z,1} \quad \cdots \quad \beta_{z,C}\ ]'$ is a C-dimensional vector of slope coefficients. The forecast corresponding to Equation (37) is given by

$$\hat{\pi}^{\text{PCR}}_{t+1:t+h} = \hat{\alpha}^{\text{OLS}}_z + \hat{\boldsymbol{z}}'_t \hat{\boldsymbol{\beta}}^{\text{OLS}}_z, \tag{38}$$

where $\hat{\alpha}^{\text{OLS}}_z$ and $\hat{\boldsymbol{\beta}}^{\text{OLS}}_z$ are the OLS estimates of α_z and $\boldsymbol{\beta}_z$, respectively, in Equation (37), and $\hat{\boldsymbol{z}}_t$ is the C-dimensional vector of the first C principal components computed from $\boldsymbol{x}_t$, all of which are based on data through t. Because the principal components are linear combinations of the underlying predictors in $\boldsymbol{x}_t$, the PCR forecast is linear in the predictors.

Intuitively, we extract a limited set of principal components from $\boldsymbol{x}_t$ to estimate the key latent variables that underlie the comovements among the entire set of predictors, thereby filtering much of the noise in the individual predictors to produce a more reliable signal; the principal components then serve as predictors in a low-dimensional predictive regression with uncorrelated explanatory variables.[17] We select L in $\boldsymbol{\pi}^{\text{AR}}_{t-L:t}$ and C by choosing the combination that maximizes the adjusted R^2 for the training sample (allowing for maximum values of eleven and ten for L and C, respectively).[18]

3.1.2. Elastic Net

The second approach uses the ENet (Zou and Hastie 2005) to estimate Equation (35). The ENet is a refinement of the least absolute shrinkage and selection operator (LASSO, Tibshirani 1996), a popular machine-learning device for implementing shrinkage. The LASSO and ENet employ penalized regression to shrink the estimated slope coefficients toward zero to guard against overfitting, and there is evidence that penalized regression helps to improve inflation forecasting (e.g., Li and Chen 2014; Medeiros and Mendes 2016; Smeekes and Wijler 2018). The LASSO relies on the ℓ_1 norm in its penalty term, so that it can shrink slope

[17] The principal components are uncorrelated by construction. Following convention, we standardize the predictors (using data through t) before computing the principal components.

[18] It is also possible to estimate an AR-augmented PCR regression, where the AR terms are estimated separately. Doing so results in slightly worse out-of-sample performance.

coefficients to exactly zero, thereby performing variable selection. A potential drawback to the LASSO is that it tends to arbitrarily select a single predictor from a group of highly correlated predictors. The ENet mitigates this tendency by including both ℓ_1 and ℓ_2 components in its penalty term; the latter component is from ridge regression (Hoerl and Kennard 1970).

The objective function for ENet estimation of Equation (35) can be expressed as

$$\underset{\alpha,\boldsymbol{\beta}}{\arg\min}\ \frac{1}{2[t-(h-1)-1]}\left\{\sum_{s=1}^{t-(h-1)-1}\left[\pi_{s+1:s+h}-(\alpha+\boldsymbol{x}_s'\boldsymbol{\beta})\right]^2\right\}+\lambda P_\delta(\boldsymbol{\beta}), \tag{39}$$

where

$$P_\delta(\boldsymbol{\beta})=0.5(1-\delta)\|\boldsymbol{\beta}\|_2^2+\delta\|\boldsymbol{\beta}\|_1; \tag{40}$$

$\lambda \geq 0$ is a hyperparameter that governs the degree of shrinkage; $\|\cdot\|_1$ and $\|\cdot\|_2$ are the ℓ_1 and ℓ_2 norms, respectively; and $0 \leq \delta \leq 1$ is a hyperparameter for blending the ℓ_1 and ℓ_2 components in the penalty term.[19] We follow the recommendation of Hastie and Qian (2016) and set $\delta = 0.5$, which they point out results in a stronger tendency to select highly correlated predictors as a group. To tune λ, we use a walk-forward cross-validation procedure that is designed for a time-series context. The ENet forecast based on Equation (35) is given by

$$\hat{\pi}_{t+1:t+h}^{\text{ENet}}=\hat{\alpha}^{\text{ENet}}+\boldsymbol{x}_t'\hat{\boldsymbol{\beta}}^{\text{ENet}}, \tag{41}$$

where $\hat{\alpha}^{\text{ENet}}$ and $\hat{\boldsymbol{\beta}}^{\text{ENet}}$ are the ENet estimates of α and $\boldsymbol{\beta}$, respectively, in Equation (35) based on data through t.

[19] The ENet objective function in Equation (39) reduces to that for OLS when $\lambda = 0$. If $\delta = 1$ ($\delta = 0$), then Equation (39) corresponds to the LASSO (ridge) objective function.

3.1.3. Random Forest

Our next strategy employs the random forest (RF) estimator of Breiman (2001). RFs are nonlinear machine-learning techniques that have a strong track record in macroeconomic forecasting (e.g., Medeiros et al. 2021; Borup and Schütte 2022; Goulet Coulombe et al. 2022). RFs build on regression trees, machine-learning devices for incorporating nonlinearities in a flexibe manner via multi-way interactions and higher-order effects of the predictors. A regression tree is constructed by sequentially splitting the predictor space into regions, with the final set of regions referred to as *terminal nodes* or *leaves*. The prediction is the average value of the target in a given leaf. We can express the forecast corresponding to a regression tree with U leaves as

$$\hat{\pi}^{\text{RT}}_{t+1:t+h} = \sum_{u=1}^{U} \bar{\pi}_u \mathbf{1}_u(\boldsymbol{x}_t\,;\boldsymbol{\eta}_u), \tag{42}$$

where the indicator function $\mathbf{1}_u(\boldsymbol{x}_t\,;\boldsymbol{\eta}_u) = 1$ if $\boldsymbol{x}_t \in R_u(\boldsymbol{\eta}_u)$ for the uth region denoted by R_u (which is determined by the parameter vector $\boldsymbol{\eta}_u$) and 0 otherwise, and $\bar{\pi}_u$ is the average value of the target observations in R_u for the training sample based on data through t.

A large (or *deep*) regression tree is typically able to capture complex nonlinear relations in the data. However, in light of the bias-variance trade-off, it is susceptible to overfitting due to the high variance of the tree. The RF methodology reduces the variance by averaging forecasts over many regression trees, where each tree is constructed based on a bootstrap sample of the original data using a randomly selected subset of the predictors for each split. By using a randomly selected subset of the predictors, we decorrelate the trees to further reduce the variance. Indexing the bootstrap samples by b, the RF forecast is given by

$$\hat{\pi}^{\text{RF}}_{t+1:t+h} = \frac{1}{B}\sum_{b=1}^{B}\left[\sum_{u=1}^{U} \bar{\pi}_u^{(b)} \mathbf{1}_u^{(b)}(\boldsymbol{x}_t\,;\boldsymbol{\eta}_u)\right], \tag{43}$$

where B is the number of bootstrap samples, and $\bar{\pi}_u^{(b)}$ and $\mathbf{1}_u^{(b)}(\boldsymbol{x}_t\,;\boldsymbol{\eta}_u)$ are the counterparts

to $\bar{\pi}_u$ and $\mathbf{1}_u(\boldsymbol{x}_t\,;\boldsymbol{\eta}_u)$, respectively, in Equation (42) for the bth bootstrap sample. We set $B = 500$ and let each tree grow fully deep. The proportion of predictors randomly selected for each split is tuned via a walk-forward cross-validation procedure.

3.1.4. Neural Network

The final strategy that we consider for forecasting inflation employs feedforward neural networks (NNs). NNs are machine-learning devices that permit nonlinearities and have proven useful for forecasting macroeconomic variables (e.g., Borup, Rapach, and Schütte forthcoming; Hauzenberger, Huber, and Klieber forthcoming). An NN contains multiple layers. The first is the input layer, which is comprised of the set of predictors, followed by $L \geq 1$ hidden layers. Each hidden layer l contains P_l neurons, where each neuron takes signals from the neurons in the previous layer to generate a subsequent signal:

$$h_m^{(l)} = g\left(\omega_{m,0}^{(l)} + \sum_{j=1}^{P_{l-1}} \omega_{m,j}^{(l)} h_j^{(l-1)}\right) \tag{44}$$

for $m = 1, \ldots, P_l$ and $l = 1, \ldots, L$, where $h_m^{(l)}$ is the signal corresponding to the mth neuron in the lth hidden layer;[20] $\omega_{m,0}^{(l)}, \omega_{m,1}^{(l)}, \ldots, \omega_{m,P_{l-1}}^{(l)}$ are weights; and $g(\cdot)$ is a nonlinear activation function. The output layer is the final layer. It takes the signals from the last hidden layer and converts them into a prediction:

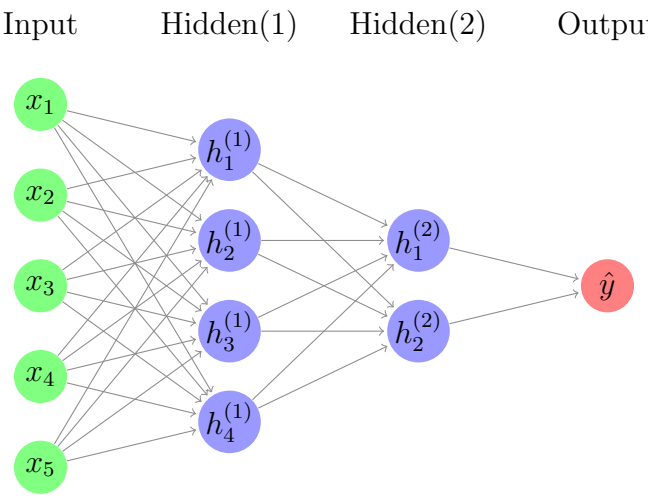
$$\hat{\pi}_{t+1:t+h}^{\text{NN}} = \omega_0^{(L+1)} + \sum_{j=1}^{P_L} \omega_j^{(L+1)} h_j^{(L)}. \tag{45}$$

For the activation function, we use the popular rectified linear unit (ReLU) function: $g(x) = \max\{x, 0\}$.

To illustrate the basic structure of a feedforward NN, the following diagram shows a network consisting of five inputs and two hidden layers with four and two neurons, respectively,

[20] For the first hidden layer, $h_j^{(0)} = x_{j,t}$ for $j = 1, \ldots, P$.

where $\hat{y}$ generically denotes the forecast of the target:



The interactions in the network and activation function permit complex nonlinearities as the inputs feed through to the hidden layers and finally to the output layer. Theoretically, a single hidden layer is sufficient for approximating any smooth function (Cybenko 1989; Funahashi 1989; Hornik, Stinchcombe, and White 1989; Hornik 1991; Barron 1994). However, there are potential advantages to including multiple hidden layers in NNs (Goodfellow, Bengio, and Courville 2016; Rolnick and Tegmark 2018), so that NNs with multiple hidden layers are commonly used.

Determining the NN architecture—that is, the number of hidden layers and number of neurons in each layer—for a given application largely remains an empirical matter, and we cannot know that the optimal architecture has been selected (Goodfellow, Bengio, and Courville 2016). Accordingly, we choose an equal-weighted ensemble of two different NN architectures: a shallow NN with one hidden layer and a deep NN with three hidden layers.[21] We follow a conventional geometric pyramid rule (Masters 1993) in setting the number of neurons in the hidden layers, so that the shallow NN has $\lceil\sqrt{P}\rceil$ neurons in its hidden payer, while the deep neural network has $\lceil P^{3/4}\rceil$, $\lceil P^{2/4}\rceil$, and $\lceil P^{1/4}\rceil$ in its first, second, and third hidden layers, respectively.

Fitting an NN necessitates estimating the weights, which is typically done using a stochas-

[21] An NN with one or two (three or more) hidden layers is typically called a *shallow* (*deep*) NN.

tic gradient descent (SGD) algorithm. We fit the NNs by minimizing the training sample MSE using the Adam SGD algorithm (Kingma and Ba 2015). To reduce the influence of the random number generator in the initialization of the weights when estimating the NNs, we fit each model 200 times with a different seed each time and use the median of the predictions.[22]

3.1.5. Ensembles

We also consider ensembles of prediction models, which are popular in the machine-learning literature. An ensemble forecast can be straightforwardly computed as a simple average of the forecasts generated by the models in the ensemble.[23] We construct three ensembles:

- Ensemble-linear: average of the linear PCR and ENet forecasts in Sections 3.1.1 and 3.1.2, respectively;
- Ensemble-nonlinear: average of the nonlinear RF and NN forecasts in Sections 3.1.3 and 3.1.4, respectively;
- Ensemble-all: average of all of the individual forecasts in Sections 3.1.1 to 3.1.4.

3.2. Data

We measure inflation based on the US CPI. CPI data are from the FRED database at the Federal Reserve Bank of St. Louis (ticker `CPIAUCSL`). The predictors are from three data sources. First, we use a set of 121 predictors from the FRED-MD database (McCracken and Ng 2016), which is employed by a number of recent macroeconomic forecasting studies (e.g., Kotchoni, Leroux, and Stevanovic 2019; Medeiros et al. 2021; Borup and Schütte 2022;

[22] Although the Adam SGD algorithm is a powerful optimizer, it is our experience that NNs at times get stuck near local minima. Using the median of 200 fitted NNs substantially reduces the influence of local minima in computing the prediction. We fit the NNs using the `scikit-learn` package in `Python`. We implement a degree of regularization by augmenting the objective function with an ℓ_2 penalty term; we set the hyperparameter for the ℓ_2 penalty term to 0.0001 in the `MPLregressor` function. The batch size and number of epochs are set to 32 and 1,000, respectively.

[23] The algorithm for computing the PBSV$_p$ in Equation (24) straightforwardly accommodates ensemble forecasts, including those that use data-driven methods to select the combining weights (e.g., Gospodinov and Maasoumi 2021).

Goulet Coulombe et al. 2022; Hauzenberger, Huber, and Klieber forthcoming). Second, we include seven predictors from the Institute for Supply Management: Manufacturing Inventories Index, Manufacturing Production Index, Manufacturing New Orders Index, Manufacturing Employment Index, Manufacturing Prices Index, Manufacturing Supplier Deliveries Index, and PMI Composite Index. Finally, we consider three predictors from the University of Michigan Survey of Consumers: Index of Consumer Sentiment, Index of Consumer Expectations, and Index of Current Economic Conditions.

Section A.2 of the Online Appendix provides a complete list of the inflation predictors. The sample period covers 1960:01 to 2022:01. We specify 1960:01 to 1999:12 as the initial in-sample period and compute out-of-sample forecasts for 2000:01 to 2022:01. Because we use a rolling estimation window, as we move through time, we include an additional monthly observation at the end of the training sample and drop the first monthly observation from the previous training sample.

3.3. Results

An AR model of order k serves as the benchmark, where we determine k recursively using the Bayesian information criterion (BIC, Schwarz 1978), considering a maximum value of twelve (so that the general prediction model in Equation (33) nests the benchmark). Like the models in Sections 3.1.1 to 3.1.4, we estimate the AR benchmark model via a rolling window. The AR model is a standard benchmark in the macroeconomic forecasting literature, including for inflation (e.g., Kotchoni, Leroux, and Stevanovic 2019; Medeiros et al. 2021), and is designed to account for the evident persistence in inflation.

We evaluate the inflation forecasts using the RMSE criterion. Table 1 reports results for the accuracy of the inflation forecasts for horizons of one, three, six, and twelve months. The table provides the RMSE for the AR benchmark forecast, as well as the RMSE ratio for each competing forecast vis-à-vis the AR benchmark. We use the Diebold and Mariano (1995) and West (1996) statistic to test the null hypothesis that the MSE (in population) for the

Table 1: Out-of-Sample Forecasting Results for Inflation

The table reports the root mean squared error (RMSE) for an autoregressive (AR) benchmark forecast and RSME ratio for the competing forecast in the first column vis-à-vis the AR benchmark forecast for inflation for 2000:01 to 2022:01 and the forecast horizon (h) in the column heading. The forecasts are generated using a rolling estimation window; the initial window spans 1960:01 to 1999:12. Brackets report the p-value for the Diebold and Mariano (1995) and West (1996) statistic for testing the null hypothesis that the benchmark forecast MSE is less than or equal to the competing forecast MSE against the (one-sided, upper tail) alternative hypothesis that the benchmark forecast MSE is greater than the competing forecast MSE.

(1) Forecast	(2) $h = 1$	(3) $h = 3$	(4) $h = 6$	(5) $h = 12$
AR RMSE	0.29%	0.26%	0.21%	0.17%
Principal component regression	1.098	0.980	0.977	0.961
	[0.930]	[0.311]	[0.352]	[0.284]
Elastic net	0.958	0.951	0.979	1.003
	[0.021]	[0.060]	[0.354]	[0.529]
Random forest	1.010	0.971	0.980	0.738
	[0.631]	[0.274]	[0.100]	[0.010]
Neural network	0.973	0.894	0.855	0.730
	[0.184]	[0.022]	[0.055]	[0.008]
Ensemble-linear	1.000	0.947	0.958	0.939
	[0.505]	[0.052]	[0.231]	[0.118]
Ensemble-nonlinear	0.973	0.908	0.858	0.720
	[0.160]	[0.032]	[0.046]	[0.006]
Ensemble-all	0.965	0.909	0.878	0.763
	[0.077]	[0.016]	[0.048]	[0.006]

AR benchmark forecast is less than or equal to that for the competing forecast against the (one-sided, upper-tail) alternative that the AR forecast MSE is greater than the competing forecast MSE.[24]

The RMSE for the AR benchmark forecast decreases monotonically with the horizon from 0.29% ($h = 1$) to 0.17% ($h = 12$) in Table 1. For the one-month horizon, the RMSE ratios

[24]We use a robust standard error (Newey and West 1987) to compute the Diebold and Mariano (1995) and West (1996) statistic, which accounts for the autocorrelation induced by overlapping observations when $h > 1$.

in the second column are less than one for four of the seven competing forecasts, so that the competing forecasts produce a lower RMSE than the AR benchmark for the majority of cases. Specifically, the linear ENet and nonlinear NN forecasts, as well as the Ensemble-nonlinear and Ensemble-all forecasts, outperform the benchmark. The improvement in MSE is statistically significant at the 5% (10%) level for the linear ENet (Ensemble-all) forecast. The ENet forecast is the most accurate for the one-month horizon, lowering the RMSE by 4.2% compared to the AR benchmark, followed by Ensemble-all, with an RMSE reduction of 3.5%.

The accuracy of the competing forecasts relative to the AR benchmark typically improves as the horizon increases. All of the competing forecasts outperform the AR benchmark for the three- and six-month horizons in the third and fourth columns, respectively, of Table 1, and the majority of the decreases in MSE vis-à-vis the AR benchmark are significant at the 5% or 10% level. For the three-month horizon, the NN forecast is the most accurate, with a reduction in RMSE of 10.6% relative to the AR benchmark. The Ensemble-nonlinear and Ensemble-all forecasts also perform well, with RMSE reductions of 9.2% and 9.1%, respectively. For the six-month horizon, the NN forecast lowers the RMSE by 14.5% relative to the AR benchmark, again making it the most accurate forecast. The Ensemble-nonlinear and Ensemble-all forecasts again perform well, reducing the RMSE by 14.2% and 12.2%, respectively.

For the twelve-month horizon in the last column of Table 1, six of the seven competing forecasts outperform the AR benchmark in terms of RMSE (the exception is the linear ENet, with an RMSE ratio slightly above one). The improvements in MSE vis-à-vis the AR benchmark are significant at the 1% level for the RF, NN, Ensemble-nonlinear, and Ensemble-all forecasts. The nonlinear RF and NN forecasts produce sizable RMSE reductions of 26.2% and 27%, respectively, while the Ensemble-nonlinear forecast performs the best, reducing the RMSE by 28%. The Ensemble-all approach also performs well, delivering an RMSE re-

duction of 23.7%.[25] Consistent with the recent literature, Table 1 indicates that big data in conjunction with machine learning is an effective strategy for forecasting inflation and that nonlinear machine-learning devices are especially useful for forecasting inflation at longer horizons.

Figure 1 shows the iShapley-VI$_p$ and oShapley-VI$_p$ measures in Equations (10) and (16), respectively, as well as the global PBSV$_p$ in Equation (31) based on the RMSE, for the NN forecast. Using the methodology in Section 2.3, the PBSV$_p$ measures in Figure 1 decompose the RMSEs for the NN forecasts in Table 1 into components attributable to the individual predictors. The different panels in Figure 1 display results for the different horizons. The predictors on the horizontal axis in each panel are ordered according to iShapley-VI$_p$. The red bars and black lines correspond to iShapley-VI$_p$ and oShapley-VI$_p$, respectively, while the green bars correspond to $\hat{\phi}_p^{\text{out}}(W, h, \text{RMSE})$ in Equation (31).[26] To conserve space, the horizontal axis contains the 25 most important and five least important predictors in descending order based on iShapley-VI$_p$. The numbers associated with the green bars are rankings for the contributions of the predictors to out-of-sample forecasting accuracy, where predictors with a positive (negative) ranking contribute negatively (positively) to RMSE over the out-of-sample period; for example, a ranking of 1 (-1) signifies the predictor that contributes the most in a positive (negative) sense to out-of-sample forecasting accuracy.[27]

Comparing the red bars with the black lines, there is a reasonably close correspondence between in-sample and out-of-sample variable importance according to iShapley-VI$_p$ and oShapley-VI$_p$, respectively. This is perhaps not surprising, as the in-sample and out-of-sample predicted target values are based on the same fitted models when determining the importance of individual predictors. Comparing the red to the green bars, we also see considerable accord across the in-sample iShapley-VI$_p$ and out-of-sample PBSV$_p$. This is especially

[25] As expected, the OLS forecast in Equation (36) substantially underperforms the AR benchmark at all horizons.

[26] In Figures 1 to 4, we sum the Shapley values for each predictor and its MA(q). We also sum the Shapley values for the twelve lags of inflation.

[27] By a positive (negative) contribution to out-of-sample forecasting accuracy, we mean an improvement (deterioration) in accuracy.

Figure 1. Variable importance and PBSV for NN inflation forecast. The figure shows iShapley-VI (left axis), oShapley-VI (left axis), and PBSV (right axis) measures for the neural network (NN) inflation forecast for the 2000:01 to 2022:01 out-of-sample period. The forecast is generated using a rolling estimation window; the initial window spans 1960:01 to 1999:12. iShapley-VI (oShapley-VI) is the predictor's importance for all of the in-sample predictions over all of the rolling estimation windows (out-of-sample forecasts); PBSV is the predictor's contribution to the RMSE over the out-of-sample period. The predictors on the horizontal axis are the top 25 and bottom five predictors ordered according to their importance based on iShapley-VI. The numbers associated with the green bars are rankings of predictors according to PBSV; a positive (negative) ranking indicates predictors that improve (decrease) out-of-sample forecasting accuracy.

evident for the AR component (`ar`), which is the most relevant predictor according to both iShapley-VI$_p$ and PBSV$_p$ for horizons of one, three, and twelve months; for the six-month horizon, `ar` is the first (second) most relevant predictor based on iShapley-VI$_p$ (PBSV$_p$).

Other predictors that appear relatively important based on both iShapley-VI$_p$ and PBSV$_p$ for the different horizons include various interest rate spreads (`baaffm`, `aaafm`, `t5yffm`), Average Weekly Hours: Manufacturing (`awhman`), CPI: Medical Care (`cpimedsl`), Average Weekly Hours: Goods-Producing (`ces0600000007`), and Manufacturing Prices Index (`man_prices`).

However, there are also major points of discord between the in-sample iShapley-VI$_p$ and out-of-sample PBSV$_p$ in Figure 1. These often involve predictors related to housing. For example, Housing Starts: South (`housts`) is the predictor that contributes the most to increasing the out-of-sample loss for all horizons, while it is always among the 25 most important predictors based on the in-sample iShapley-VI$_p$. Although it appears among the top 25 predictors on an in-sample basis for the one- and three-month horizons, Housing Starts: Northeast (`houstne`) contributes the second most to increasing the out-of-sample loss for these horizons. The Index of Consumer Expectations (`soc_ice`) is the fourth most important predictor for the one-month horizon and second most important predictor for the remaining horizons on an in-sample basis, but it is among the predictors that contribute the most to out-of-sample loss for the four horizons. The Index of Consumer Sentiment (`soc_ics`) and Manufacturing Employment Index (`man_empl`) evince similar discrepancies between their in-sample importance and contributions to out-of-sample forecasting accuracy.

Figure 2 depicts results for the Ensemble-nonlinear forecast. Overall, the results are similar to those for the NN forecast in Figure 1. We see that numerous predictors are important in Figure 2 according to the in-sample iShapley-VI$_p$ and also contribute positively to forecasting accuracy according to the out-of-sample PBSV$_p$. These predictors again include `ar`, `cpimedsl`, various interest rate spreads, `man_prices`, and `awhman`. However, like Figure 1, there are also notable discrepancies between iShapley-VI$_p$ and PBSV$_p$ in Figure 2. The predictor `housts` again appears among the top predictors in terms of in-sample variable importance, while it contributes the most to increasing out-of-sample loss. Discrepancies between iShapley-VI$_p$ and PBSV$_p$ are also evident for Housing Permits: South (`permits`)

Figure 2. Variable importance and PBSV for Ensemble-nonlinear inflation forecast. See the notes for Figure 2 with Ensemble-nonlinear replacing the neural network forecast.

for the one-month horizon; `soc_ice`, `soc_ics`, and `man_empl` for all reported horizons; and `houstne` for the three-month horizon.

For the Ensemble-linear forecast in Figure 3, there are generally fewer discrepancies between iShapley-VI$_p$ and PBSV$_p$. Predictors that rank highly in terms of both their in-sample importance and contributions to improvements in out-of-sample forecasting accuracy across the reported horizons include `ar`, Average Hourly Earnings: Manufacturing (`ces3000000008`), `baaffm`, Personal Consumption Expenditures: Durable Goods

Figure 3. Variable importance and PBSV for Ensemble-linear inflation forecast. See the notes for Figure 2 with Ensemble-linear replacing the neural network forecast.

(`ddurrg3m086sbea`), and CPI: Durables (`cusr0000sad`). There are still some notable discrepancies between iShapley-VI$_p$ and PBSV$_p$, including for `soc_ice` for all reported horizons, Total Reserves of Depository Institutions (`totresns`) for the one-month horizon, Housing Permits: West (`permitw`) for the three-month horizon, and Unfilled Orders for Durable Goods (`amdmuox`) for the six- and twelve-month horizons.

Overall, the results in Figures 1 to 3 indicate the following. While there is often agreement between a predictor's in-sample variable importance and its contribution to improvements

in out-of-sample forecasting accuracy, this is not necessarily the case. We find a number of instances where a variable that is deemed relatively important on an in-sample basis contributes to out-of-sample performance in a manner that decreases forecasting accuracy.[28]

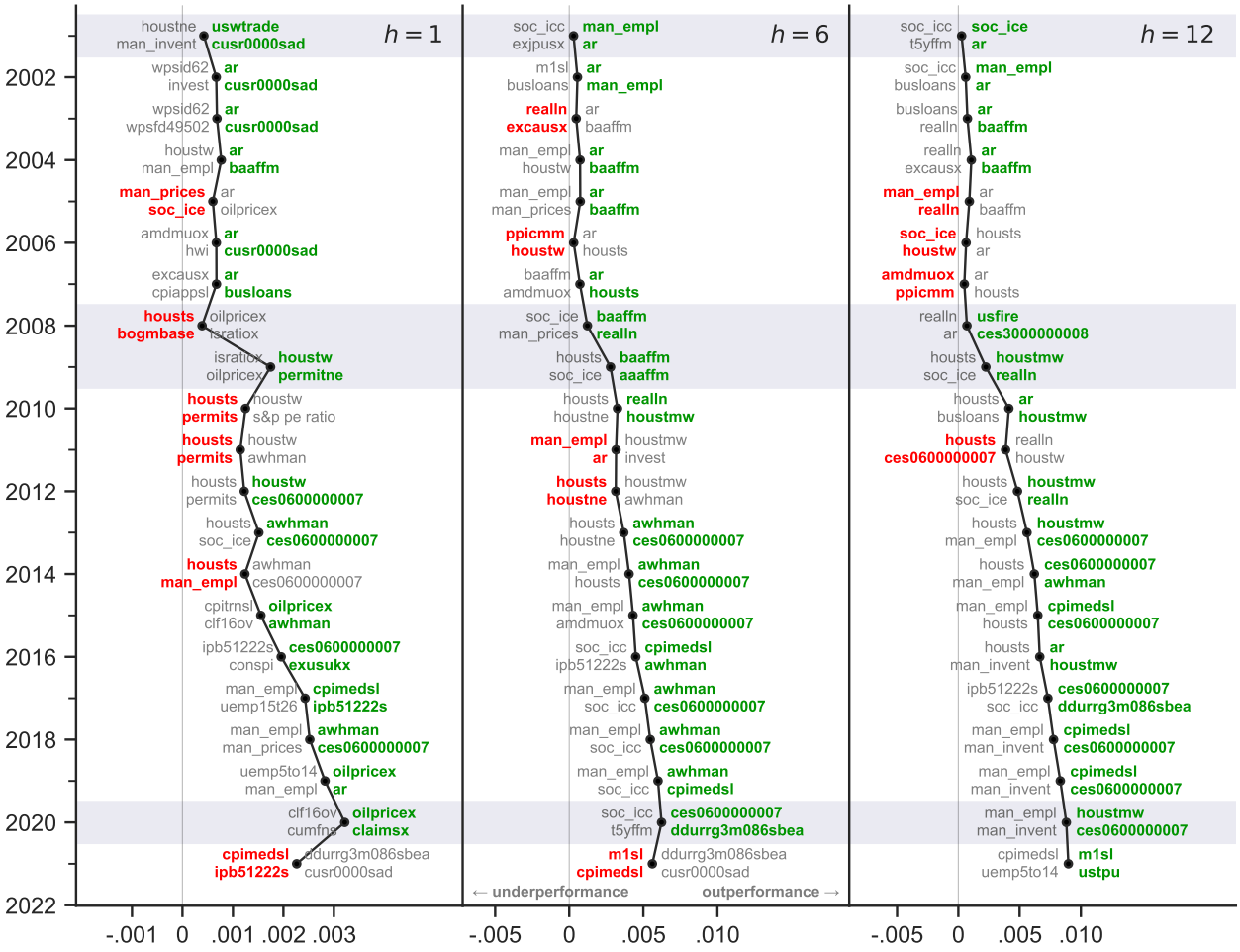


Figure 4. Cumulative difference in squared errors for NN inflation forecasts. The figure shows the cumulative difference in squared errors for the autoregressive (AR) benchmark forecast vis-à-vis the neural network (NN) forecast for the 2000:01 to 2022:01 out-of-sample period. Shifts to the right (left) imply an improvement (deterioration) in forecasting performance relative to the AR benchmark. The figure also shows the two top (bottom) contributors to the improvement (deterioration) in forecasting performance for non-overlapping 24-month subsamples in the out-of-sample period; a green (red) color for the predictor indicates that the 24-month subsample is associated with an improvement (deterioration) in performance. Horizontal gray bars indicate 24-month subsamples that contain an NBER-dated recession.

To get a sense of out-of-sample forecasting performance over time, Figure 4 plots the cumulative difference in squared errors (CDSE, Goyal and Welch 2008) between the benchmark

[28]Section A.3 of the Online Appendix reports results for an empirical application forecasting the US equity premium using 28 predictors from Neely et al. (2014). We also find important discrepancies between iShapley-VI$_p$ and PBSV$_p$ for equity premium forecasts.

and a competing forecast. To conserve space, we focus on the NN forecast, as it generally performs well in Table 1, and report results for horizons of one, six, and twelve months. The CDSE provides a convenient and informative graphical device for ascertaining whether a competing forecast is more accurate than the benchmark for any subsample of the out-of-sample period. In terms of Figure 4, we compare the CDSE at the beginning and end of the interval corresponding to a subsample. If the curve lies more to the right (left) at the end of the interval relative to the beginning, then the NN (AR) forecast is more accurate in terms of MSE for the subsample. We also use our procedure in Section 2.3 to compute PBSV_p measures for the NN forecast for non-overlapping 24-month rolling subsamples. Figure 4 shows the two predictors that contribute the most to positive performance during a subsample, as well as the two that contribute the most to negative performance. Predictors in green (red) to the right (left) of the curve indicate that the NN forecast delivers a lower (higher) MSE than the AR benchmark for the subsample. The variables in green (red) help us to identify the predictors that are most responsible for the outperformance (underperformance) of the NN forecast vis-à-vis the AR benchmark for the subsample. The horizontal gray bars indicate 24-month subsamples that contain an NBER-dated recession.

The CDSE plots in Figure 4 are consistently positively sloped (when viewed from top to bottom), so that the NN forecasts outperform the AR benchmark on a consistent basis over time. The plots are relatively steeply sloped during the Global Financial Crisis and concomitant Great Recession in the late 2000s, so that the information in the set of predictors is especially useful for forecasting inflation during that period.

For the one-month horizon in the left panel of Figure 4, there are three 24-month periods for which the price of oil (`oilpricex`) is the predictor most responsible for the outperformance of the NN forecast relative to the AR benchmark, including during the recent recession corresponding to the advent of COVID-19. This is not surprising, given the important influence of oil prices—and energy prices more generally—on short-run CPI fluctuations. But oil prices are far from the complete story. Among the other predictors, `awhman` and

ces0600000007 are primarily responsible for the outperformance of the NN forecast for a number of periods during the 2010s. Consistent with the top panel of Figure 1, housts is the predictor most responsible for the relatively poor performance of the NN forecast for four of the six 24-month periods when the NN forecast fails to beat the AR benchmark in the left panel of Figure 4.

For the six-month horizon in the middle panel of Figure 4, awhman and ces0600000007 play even more important roles in accounting for the outperformace of the NN forecast vis-à-vis the AR benchmark starting in 2010, as awhman (ces0600000007) is selected among the top two predictors for seven (six) of the 24-month periods. The importance of awhman for forecasting inflation accords with Clark et al. (2022) and Goulet Coulombe (2022), who find that awhman is a leading inflation predictor based on nonparametric Bayesian methods and hemispheric neural networks, respectively. Indeed, Goulet Coulombe (2022) finds that awhman (in combination with other predictors) is particularly pertinent for measuring economic "slack" in a deep learning-based Phillips Curve. Together with our PBSV$_p$ results, this suggests that the average number of hours worked in manufacturing is a more potent forcing variable than unemployment in Phillips curve and NN-based inflation forecasting models.

The right panel of Figure 4 shows results for the twelve-month horizon. In addition to the importance of ces0600000007, Housing Starts: Midwest (houstmw) and cpimedsl emerge as leading predictors in accounting for the outperformance of the NN forecast compared to the AR benchmark. With respect to cpimedsl, it is the leading predictor for three of the 24-month periods starting in 2014. Medical care inflation is considerably more stable than volatile CPI components like food and energy price inflation. In fact, Bils and Klenow (2004) and Bryan and Meyer (2010) rank medical care among the stickiest components of the CPI (in terms of its low frequency of price adjustment), and it is an important component in the Federal Reserve Bank of Atlanta's Sticky-Price CPI. Accordingly, cpimedsl (or some nonlinear transformation of it) likely captures slowly evolving inflation expectations that

can be important for forecasting inflation at longer horizons (Bryan and Meyer 2010).

4. Simulations

Section 3 provides evidence that the out-of-sample PBSV_p in Equation (24) can rank predictors quite differently from the in-sample iShapley-VI_p in Equation (10) (as well as the out-of-sample oShapley-VI_p in Equation (16)). To understand potential reasons for such differences, we use Monte Carlo simulations to highlight situations where there can be a wedge between the rankings of predictors according to iShapley-VI_p and PBSV_p. We evaluate differences in rankings using the mean squared deviation in rankings (MSDR):

$$\text{MSDR} = \frac{1}{P}\sum_{p=1}^{P}\Big[\text{ranking}\Big(\hat{\phi}_p^{\text{out}}(W,h,L)\Big) - \text{ranking}\big(\text{iShapley-VI}_p(W,h)\big)\Big]^2, \tag{46}$$

where $\hat{\phi}_p^{\text{out}}(W,h,L)$ is the global PBSV_p in Equation (24). Of course, we do not know the true DGP for the empirical applications in Section 3, so that we cannot know the actual reasons for the differences in rankings. The simulations are intended to shed light on the relevance of plausible explanations for the divergence in rankings. We consider three potential explanations: overfitting, structural breaks in slope coefficients, and evolving predictor volatilities.

We incorporate nonlinearities in the simulations via a modification of the oft-used "Friedman DGP" (e.g., Friedman, Grosse, and Stuetzle 1983; Friedman 1991):

$$\pi_{t+1} = \sum_{i=1}^{28}\tilde{z}_{i,t} + \varepsilon_{t+1}, \tag{47}$$

$$\tilde{z}_{i,t} = \beta_{1,i}z_{1,i,t} + \beta_{2,i}\sin(\pi z_{1,i,t}z_{2,i,t}) + \beta_{3,i}(z_{3,i,t}-0.5)^2 + \beta_{4,i}z_{4,i,t} + \beta_{5,i}z_{5,i,t} \tag{48}$$

for $i = 1,\ldots,28$. We collect the complete set of predictors in the vector $\boldsymbol{z}_t = [\ \boldsymbol{z}_{1,t}' \ \cdots \ \boldsymbol{z}_{28,t}'\ ]'$, where $\boldsymbol{z}_{i,t} = [\ z_{1,i,t} \ \cdots \ z_{5,i,t}\ ]'$ for $i = 1,\ldots,28$. In essence, we sum many low-dimensional

Friedman DGPs to construct a large-scale DGP with $P = 28 \times 5 = 140$ predictors. Collecting the slope coefficients in $\boldsymbol{\beta} = [\ \boldsymbol{\beta}_1' \quad \cdots \quad \boldsymbol{\beta}_{28}'\ ]'$, where $\boldsymbol{\beta}_i = [\ \beta_{1,i} \quad \cdots \quad \beta_{5,i}\ ]'$ for $i = 1, \ldots, 28$, for each iteration, we draw the each slope coefficient independently from the uniform distribution $\mathcal{U}(-P/2, P/2)$.[29] For the predictors, we assume that $\boldsymbol{z}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_z)$, where $\boldsymbol{\Sigma}_z$ is based on the predictor data in Section 3. For the disturbance term in Equation (47), we assume that $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$. To ensure that the R^2 statistic is at the desired level for the DGP, we set

$$\sigma^2 = \text{var}\left(\sum_{i=1}^{28} \tilde{z}_{i,t}\right) \Big/ \underbrace{\left[-R^2/\left(R^2 - 1\right)\right]}_{\text{STN}}, \tag{49}$$

where STN is the signal-to-noise ratio. In addition, we center and scale the right-hand-side of Equation (47) so that the generated data for π_t have the same mean and variance as actual inflation over the out-of-sample period in Section 3. The sizes of the initial in-sample and out-of-sample periods match those for the application in Section 3. For each iteration of generated time-series observations (and using 1,000 iterations), as in Section 3, we compute the linear PCR and ENet forecasts in Sections 3.1.1 and 3.1.2, respectively, and nonlinear RF and NN forecasts in Sections 3.1.3 and 3.1.4, respectively, using a rolling estimation window, as well as the ensemble forecasts in Section 3.1.5. For the various forecasts, we compute predictor rankings based on iShapley-VI$_p$ and $\hat{\phi}_p^{\text{out}}(W, h, \text{RMSE})$ and the standardized MSDR.[30] Finally, we take the average of simulated standardized MSDRs across the iterations.

To investigate the potential role of overfitting, the top panel of Figure 5 shows the relation between the standardized MSDR and signal strength as measured by the R^2 statistic. Results are reported for the nonlinear RF and NN forecasts, as well as the Ensemble-linear and Ensemble-nonlinear forecasts. The horizontal dashed line at 50% corresponds to a baseline case in which the rankings are random, so that there is no link between iShapley-VI$_p$ and

[29] We introduce some sparsity by setting the seven slope coefficients with the smallest magnitudes to zero.

[30] We standardize the MSDR by dividing the MSDR in Equation (46) by the maximum MSDR between two rankings (i.e., the MSDR for rankings $\{1, 2, \ldots, P-1, P\}$ and $\{P, P-1, \ldots, 2, 1\}$), which is given by $\left(P^2 - 1\right)/3$.

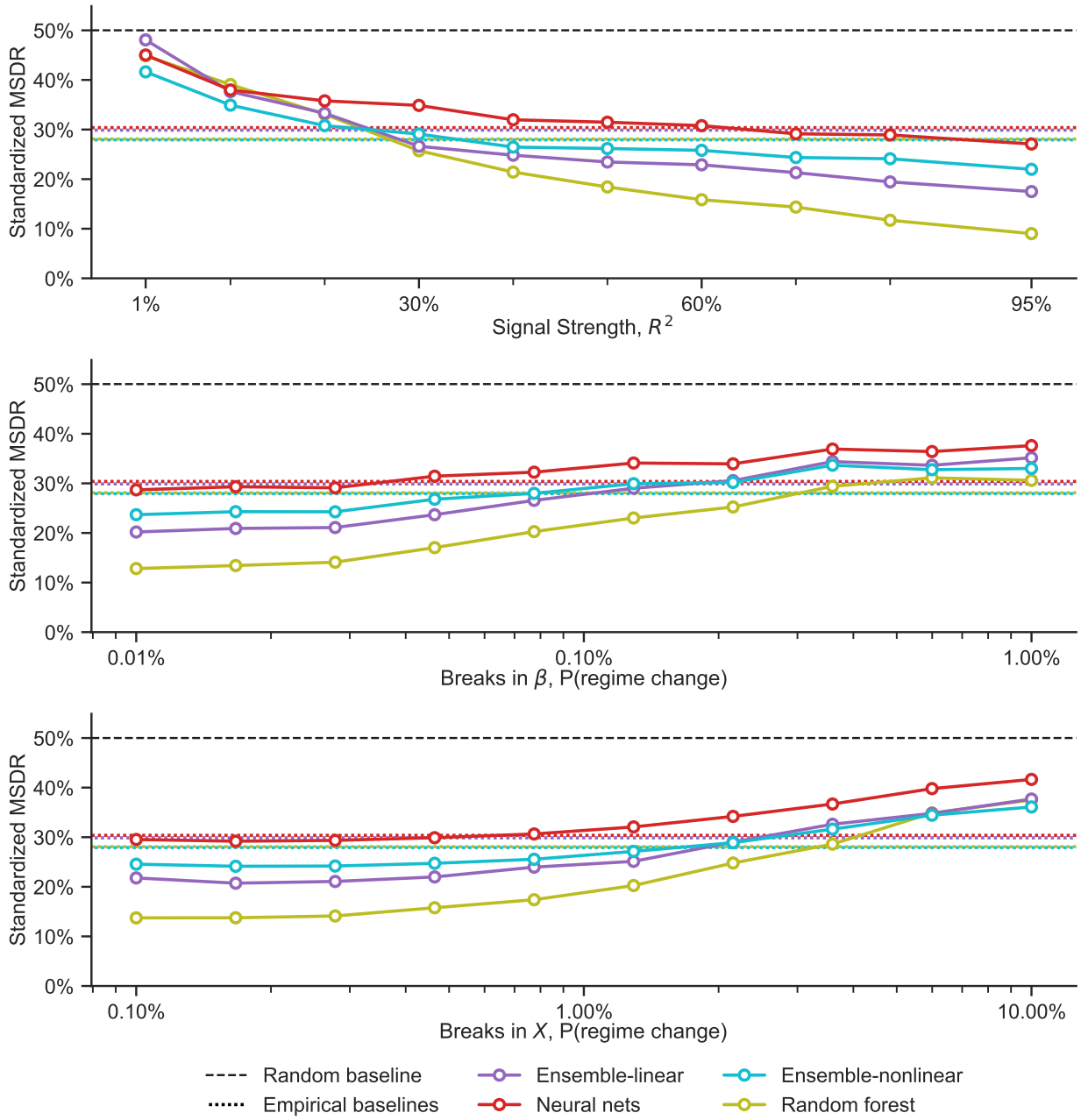


Figure 5. Standardized MSDR for DGPs based on the empirical application. The figure shows the standardized mean squared deviation in rankings (MSDR) for data-generating processes (DGPs) based on the empirical application in Section 3.

PBSV$_p$.[31] The figure also includes horizontal dotted lines corresponding to standardized MSDRs based on the actual data for $h = 1$ for the application in Section 3. The MSDR curves in the top panel of Figure 5 are negatively sloped (and monotonically so), so that, as

[31] The expected MSDR for two permutations is $(P^2 - 1)/6$, so that the standardized MSDR for the random baseline is $[(P^2 - 1)/6]/[(P^2 - 1)/3] = 0.5$.

expected, differences in predictor rankings based on iShapley-VI$_p$ and PBSV$_p$ decrease as the signal strength increases. This is consistent with the intuitive notion that overfitting creates greater discrepancies between iShapley-VI$_p$ and PBSV$_p$ as the signal strength weakens. The horizontal dotted lines intersect the corresponding MSDR curves at about 20%, 25%, 35%, and 65% for the Ensemble-linear, RF, Ensemble-nonlinear, and NN forecasts, respectively; in other words, these values for the R^2 statistics in the DGP are consistent with the differences in predictor rankings in Section 3.

We next consider structural breaks in slope coefficients as a source of differences in predictor rankings. Structural breaks are a natural candidate for explaining divergences in iShapley-VI$_p$ and PBSV$_p$, as they create changes in the DGP over time. We introduce structural breaks using the same DGP, except that we fix the R^2 statistic at 75% and generate the slope coefficients as follows. For each iteration, we draw three vectors of slope coefficients via the uniform distribution. Each period, with some probability, the DGP can change from the current vector of slope coefficients to one of the other two vectors.[32]

The middle panel of Figure 5 depicts how the standardized MSDR varies with the probability of a structural break in the slope coefficients. The MSDR curves are nearly uniformly positively sloped, so that differences in predictor rankings increase as structural breaks in the slope coefficients occur more frequently, indicating that structural breaks provide a plausible explanation for discrepancies between predictor rankings based on iShapley-VI$_p$ and PBSV$_p$. According to the intersections of the horizontal dotted lines and MSDR curves, the simulations suggest that structural break probabilities of approximately 0.03%, 0.08%, 0.2%, and 0.3% for the NN, Ensemble-nonlinear, Ensemble-linear, and RF forecasts, respectively, are in line with the results for the application in Section 3.

We consider evolving predictor volatilities as a final potential reason for differences in predictor rankings. Similarly to structural breaks in slope coefficients, evolving volatilities are a natural candidate for explaining differences in rankings, since they constitute changes

[32]When a change occurs, the DGP moves to one of the other two vectors of slope coefficients with equal probability.

in the DGP over time, such as a predictor taking on more extreme values during a crisis. In terms of the DGP, we again generate the slope coefficients via the uniform distribution, hold the slope coefficients constant over time, and again set the R^2 statistic to 75%. To incorporate evolving volatilities, each period, with some probability, the magnitude of one of the predictors increases by a factor of ten. The predictor experiencing the increase in its magnitude is selected randomly, and the increase in magnitude lasts for twelve periods.

The bottom panel of Figure 5 examines the effects of evolving predictor volatilities on the differences in predictor rankings. The MSDR curves are nearly uniformly positively sloped, so that, as anticipated, differences in predictor rankings become greater as the probability of an increase in the magnitude of a predictor increases. The intersections of the horizontal dotted lines with the MSDR curves suggest that magnitude change probabilities of around 0.5%, 2%, 2.5%, and 3.5% for the NN, Ensemble-nonlinear, Ensemble-linear, and RF forecasts, respectively, are consistent with the results in Section 3.

In summary, Figure 5 indicates that a weaker signal strength, greater probability of a structural break in slope coefficients, and greater probability of an increase in the magnitude of a predictor lead to larger differences in predictor rankings based on iShapley-VI$_p$ and PBSV$_p$. These patters appear quite plausible and provide insight into potential reasons for the discrepancies between predictor rankings for the application in Section 3.

5. Conclusion

As large datasets and machine learning become more popular in macroeconomics and finance, researchers are increasingly concerned with interpreting forecasting models fitted with time-series data. While the literature provides a variety of informative tools for interpreting fitted prediction models, existing tools are typically more appropriate for models estimated with cross-sectional data. In this paper, we develop metrics based on Shapley values for interpreting time-series forecasting models. The metrics recognize that forecasting models

are re-estimated on a regular basis as additional data become available over time. The iShapley-VI$_p$ and oShapley-VI$_p$ metrics measure the importance of a predictor for explaining the in- and out-of-sample predicted target values, respectively. Our primary methodological contribution is the PBSV$_p$, which measures the contribution of a predictor to the out-of-sample loss. By computing PBSV$_p$ for the set of predictors that are used to compute a sequence of time-series forecasts, we anatomize the model's out-of-sample forecasting accuracy. Our metrics are flexible: they are model agnostic, so that they can be applied to any prediction model (as well as ensembles of models), and PBSV$_p$ can be applied to any loss function.

We use our metrics to interpret fitted models that employ large datasets and machine learning to forecast US inflation. In line with the recent literature, we find that large datasets in conjunction with machine learning generate significant out-of-sample gains for forecasting inflation. When it comes to model interpretation, the iShapley-VI$_p$ and oShapley-VI$_p$ metrics generally paint the same picture in terms of the importance of individual predictors for the in- and out-of-sample predicted target values produced by the fitted models. In contrast, we detect a number of substantial differences in the rankings of predictors according to the in-sample iShapley-VI$_p$ and out-of-sample PBSV$_p$. This finding makes an important cautionary point: when researchers interpret time-series forecasting models, predictors that are important for determining a model's predicted values are not necessarily those that are primarily responsible for the model's out-of-sample forecasting accuracy. Via simulations, we explore potential causes of the discrepancies between iShapley-VI$_p$ and PBSV$_p$, including overfitting relating to the signal-to-noise ratio, structural breaks in slope coefficients, and evolving predictor volatilities. The simulations suggest that all three factors provide plausible explanations for the differences between iShapley-VI$_p$ and PBSV$_p$ that we find in the data.

References

Apley, D. W. and J. Zhu (2020). Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 82:4, 1059–1086.

Avramov, D., S. Cheng, and L. Metzker (forthcoming). Machine Learning Versus Economic Restrictions: Evidence from Stock Return Predictability. *Management Science*.

Banerjee, A. and M. Marcellino (2006). Are There Any Reliable Leading Indicators for US Inflation and GDP Growth? *International Journal of Forecasting* 22:1, 137–151.

Barron, A. R. (1994). Approximation and Estimation Bounds for Artificial Neural Networks. *Machine Learning* 14:1, 115–133.

Bataa, E., D. R. Osborn, M. Sensier, and D. van Dijk (2013). Structural Breaks in the International Dynamics of Inflation. *Review of Economics and Statistics* 95:2, 646–659.

Bataa, E., D. R. Osborn, M. Sensier, and D. van Dijk (2014). Identifying Changes in Mean, Seasonality, Persistence and Volatility for G7 and Euro Area Inflation. *Oxford Bulletin of Economics and Statistics* 76:3, 360–388.

Bernanke, B. S. and J. Boivin (2003). Monetary Policy in a Data-Rich Environment. *Journal of Monetary Economics* 50:3, 525–546.

Bils, M. and P. J. Klenow (2004). Some Evidence on the Importance of Sticky Prices. *Journal of Political Economy* 112:5, 947–985.

Borup, D., D. E. Rapach, and E. C. M. Schütte (forthcoming). Mixed-Frequency Machine Learning: Nowcasting and Backcasting Weekly Initial Claims with Daily Internet Search Volume Data. *International Journal of Forecasting*.

Borup, D. and E. C. M. Schütte (2022). In Search of a Job: Forecasting Employment Growth Using Google Trends. *Journal of Business & Economic Statistics* 40:1, 186–200.

Breiman, L. (2001). Random Forests. *Machine Learning* 45:1, 5–32.

Bryan, M. and B. Meyer (2010). Are Some Prices in the CPI More Forward Looking than Others? We Think So. *Federal Reserve Bank of Cleveland Economic Commentary* 2010:02.

Çakmaklı, C. and D. van Dijk (2016). Getting the Most Out of Macroeconomic Information for Predicting Excess Stock Returns. *International Journal of Forecasting* 32:3, 650–668.

Casalicchio, G., C. Molnar, and B. Bischl (2018). *Visualizing the Feature Importance for Black Box Models*. ECML PKDD Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Dublin.

Castro, J., D. Gómez, and J. Tejada (2009). Polynomial Calculation of the Shapley Value Based on Sampling. *Computer and Operations Research* 36:5, 1726–1730.

Chen, L., M. Pelger, and J. Zhu (forthcoming). Deep Learning in Asset Pricing. *Management Science*.

Chinco, A., A. D. Clark-Joseph, and M. Ye (2019). Sparse Signals in the Cross-Section of Returns. *Journal of Finance* 74:1, 449–492.

Clark, T. E., F. Huber, G. Koop, and M. Marcellino (2022). Forecasting US Inflation Using Bayesian Nonparametric Models. Federal Reserve Bank of Cleveland Working Paper 22-05.

Cybenko, G. (1989). Approximation by Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals, and Systems* 2:4, 303–314.

Diebold, F. X. and R. S. Mariano (1995). Comparing Predictive Accuracy. *Journal of Business and Economic Statistics* 13:3, 253–263.

Dong, X., Y. Li, D. E. Rapach, and G. Zhou (2022). Anomalies and the Expected Market Return. *Journal of Finance* 77:1, 639–681.

Döpke, J., U. Fritsche, and C. Pierdzioch (2017). Predicting Recessions with Boosted Regression Trees. *International Journal of Forecasting* 33:4, 745–759.

Exterkate, P., P. J. F. Groenen, C. Heij, and D. van Dijk (2016). Nonlinear Forecasting with Many Predictors Using Kernel Ridge Regression. *International Journal of Forecasting* 32:3, 736–753.

Faust, J. and J. H. Wright (2013). Forecasting Inflation. In: G. Elliott and A. Timmermann, eds. *Handbook of Economic Forecasting*. Vol. 2A. Amsterdam: Elsevier, pp. 2–56.

Fisher, A., C. Rudin, and F. Dominici (2019). All Models Are Wrong, But Many Are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *Journal of Machine Learning Research* 20:177, 1–81.

Freyberger, J., A. Neuhierl, and M. Weber (2020). Dissecting Characteristics Nonparametrically. *Review of Financial Studies* 33:5, 2326–2377.

Friedman, J. H. (1991). Multivariate Adaptive Regression Splines. *Annals of Statistics* 19:1, 1–67.

Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics* 29:5, 1189–1232.

Friedman, J. H., E. Grosse, and W. Stuetzle (1983). Multidimensional Additive Spline Approximation. *SIAM Journal on Scientific and Statistical Computing* 4:2, 291–301.

Funahashi, K.-I. (1989). On the Approximate Realization of Continuous Mappings by Neural Networks. *Neural Networks* 2:3, 183–192.

Goldstein, A., A. Kapelner, J. Bleich, and E. Pitkin (2015). Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation. *Journal of Computational and Graphical Statistics* 24:1, 44–65.

Goodfellow, I., Y. Bengio, and A. Courville (2016). *Deep Learning*. Boston: MIT Press.

Gospodinov, N. and E. Maasoumi (2021). Generalized Aggregation of Misspecified Models: With an Application to Asset Pricing. *Journal of Econometrics* 222:1B, 451–467.

Goulet Coulombe, P. (2022). A Neural Phillips Curve and a Deep Output Gap. Working Paper arXiv:2202.04146v1.

Goulet Coulombe, P., M. Leroux, D. Stevanovic, and S. Surprenant (2021). Macroeconomic Data Transformations Matter. *International Journal of Forecasting* 37:4, 1338–1354.

Goulet Coulombe, P., M. Leroux, D. Stevanovic, and S. Surprenant (2022). How is Machine Learning Useful for Macroeconomic Forecasting? *Journal of Applied Econometrics* 37:5, 920–964.

Goyal, A. and I. Welch (2008). A Comprehensive Look at the Empirical Performance of Equity Premium Prediction. *Review of Financial Studies* 21:4, 1455–1508.

Greenwell, B. M., B. C. Boehmke, and A. J. McCarthy (2018). A Simple and Effective Model-Based Variable Importance Measure. Working Paper arXiv:1805.04755v1.

Gu, S., B. Kelly, and D. Xiu (2020). Empirical Asset Pricing via Machine Learning. *Review of Financial Studies* 33:5, 2223–2273.

Han, Y., A. He, D. E. Rapach, and G. Zhou (2022). Expected Stock Returns and Firm Characteristics: E-ENet, Assessment, and Implications. Working Paper, Washington University in St. Louis.

Hastie, T. and J. Qian (2016). Glmnet Vignette. Manuscript, Stanford University.

Hauzenberger, N., F. Huber, and K. Klieber (forthcoming). Real-Time Inflation Forecasting Using Non-Linear Dimension Reduction Techniques. *International Journal of Forecasting*.

Hoerl, A. E. and R. W. Kennard (1970). Ridge Regression: Applications to Nonorthogonal Problems. *Technometrics* 12:1, 69–82.

Hornik, K. (1991). Approximation Capabilities of Multilayer Feedforward Networks. *Neural Networks* 4:2, 251–257.

Hornik, K., M. Stinchcombe, and H. White (1989). Multilayer Feedforward Networks Are Universal Approximators. *Neural Networks* 2:5, 359–366.

Janzing, D., L. Minorics, and P. Blöbaum (2020). *Feature Relevance Quantification in Explainable AI: A Causal Problem*. 23rd International Conference on Artificial Intelligence and Statistics. Palermo.

Joseph, A. (2021). Shapley Regressions: A Framework for Statistical Inference on Machine Learning Models. Working Paper arXiv:1903.04209v1.

Kim, H. H. and N. R. Swanson (2018). Mining Big Data Using Parsimonious Factor, Machine Learning, Variable Selection and Shrinkage Methods. *International Journal of Forecasting* 34:2, 339–354.

Kingma, D. P. and J. Ba (2015). *Adam: A Method for Stochastic Optimization*. Third Annual International Conference on Learning Representations. San Diego.

Koijen, R. S. J. and S. Van Nieuwerburgh (2011). Predictability of Returns and Cash Flows. *Annual Review of Financial Economics* 3:1, 467–491.

Kotchoni, R., M. Leroux, and D. Stevanovic (2019). Macroeconomic Forecast Accuracy in a Data-Rich Environment. *Journal of Applied Econometrics* 34:7, 1050–1072.

Kuan, C.-M. and H. White (1994). Artificial Neural Networks: An Econometric Perspective. *Econometric Reviews* 13:1, 1–91.

Lee, T.-H., H. White, and C. W. J. Granger (1993). Testing for Neglected Nonlinearity in Time Series Models: A Comparison of Neural Network Methods and Alternative Tests. *Journal of Econometrics* 56:3, 269–290.

Li, J. and W. Chen (2014). Forecasting Macroeconomic Time Series: LASSO-Based Approaches and Their Forecast Combinations with Dynamic Factor Models. *International Journal of Forecasting* 30:4, 996–1015.

Ludvigson, S. C. and S. Ng (2007). The Empirical Risk-Return Relation: A Factor Analysis Approach. *Journal of Financial Economics* 83:1, 171–222.

Lundberg, S. M. and S.-I. Lee (2017). *A Unified Approach to Interpreting Model Predictions*. Advances in Neural Information Processing Systems. Long Beach.

Marcellino, M. (2008). A Linear Benchmark for Forecasting GDP Growth and Inflation? *Journal of Forecasting* 27:4, 305–340.

Masters, T. (1993). *Practical Neural Network Recipes in C++*. Boston: Academic Press.

McCracken, M. W. and S. Ng (2016). FRED-MD: A Monthly Database for Macroeconomic Research. *Journal of Business & Economic Statistics* 34:4, 574–589.

Medeiros, M. C. and E. F. Mendes (2016). ℓ_1-Regularization of High-Dimensional Time-Series Models with Non-Gaussian and Heteroskedastic Errors. *Journal of Econometrics* 191:1, 255–271.

Medeiros, M. C., T. Teräsvirta, and G. Rech (2006). Building Neural Network Models for Time Series: A Statistical Approach. *Journal of Forecasting* 25:1, 49–75.

Medeiros, M. C., G. F. R. Vasconcelos, Á. Veiga, and E. Zilberman (2021). Forecasting Inflation in a Data-Rich Environment: The Benefits of Machine Learning Methods. *Journal of Business & Economic Statistics* 39:1, 98–119.

Mitchell, R., J. Cooper, E. Frank, and G. Holmes (2022). Sampling Permutations for Shapley Value Estimation. *Journal of Machine Learning Research* 23:43, 1–46.

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

Moshiri, S. and N. Cameron (2000). Neural Network Versus Econometric Models in Forecasting Inflation. *Journal of Forecasting* 19:3, 201–217.

Nakamura, E. (2005). Inflation Forecasting Using a Neural Network. *Economics Letters* 86:3, 373–378.

Neely, C. J., D. E. Rapach, J. Tu, and G. Zhou (2014). Forecasting the Equity Risk Premium: The Role of Technical Indicators. *Management Science* 60:7, 1772–1791.

Newey, W. K. and K. D. West (1987). A Simple, Positive Semi-Definite, Heteroskedasticity and Autocorrelation Consistent Covariance Matrix. *Econometrica* 55:3, 703–708.

O'Reilly, G. and K. Whelan (2005). Has Euro-Area Inflation Persistence Changed Over Time? *Review of Economics and Statistics* 87:4, 709–720.

Pearl, J. (2009). *Causality*. Second Edition. Cambridge: Cambridge University Press.

Rapach, D. E., J. K. Strauss, J. Tu, and G. Zhou (2019). Industry Return Predictability: A Machine Learning Approach. *Journal of Financial Data Science* 1:3, 9–28.

Rapach, D. E. and G. Zhou (2013). Forecasting Stock Returns. In: G. Elliott and A. Timmermann, eds. *Handbook of Economic Forecasting*. Vol. 2A. Amsterdam: Elsevier, pp. 328–383.

Rapach, D. E. and G. Zhou (2022). Asset Pricing: Time-Series Predictability. *Oxford Research Encyclopedia of Economics and Finance*, June 20, 2022.

Ribeiro, M. T., S. Singh, and C. Guestrin (2016). *'Why Should I Trust You?' Explaining the Predictions of Any Classifier*. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco.

Rolnick, D. and M. Tegmark (2018). *The Power of Deeper Networks for Expressing Natural Functions*. Sixth Annual International Conference on Learning Representations. Vancouver.

Schwarz, G. (1978). Estimating the Dimension of a Model. *Annals of Statistics* 6:2, 461–464.

Shapley, L. S. (1953). A Value for n-Person Games. *Contributions to the Theory of Games* 2:28, 307–317.

Smeekes, S. and E. Wijler (2018). Macroeconomic Forecasting Using Penalized Regression Methods. *International Journal of Forecasting* 34:3, 408–430.

Stock, J. H. and M. W. Watson (1996). Evidence on Structural Instability in Macroeconomic Time Series Relations. *Journal of Business & Economic Statistics* 14:1, 11–30.

Stock, J. H. and M. W. Watson (1999a). A Comparison of Linear and Nonlinear Univariate Models for Forecasting Macroeconomic Time Series. In: R. F. Engle and H. White, eds. *Cointegration, Causality and Forecasting: A Festschrift for Clive W. J. Granger*. Oxford: Oxford University Press, pp. 1–44.

Stock, J. H. and M. W. Watson (1999b). Forecasting Inflation. *Journal of Monetary Economics* 44:2, 293–335.

Stock, J. H. and M. W. Watson (2002a). Forecasting Using Principal Components From a Large Number of Predictors. *Journal of the American Statistical Association* 97:460, 1167–1179.

Stock, J. H. and M. W. Watson (2002b). Macroeconomic Forecasting Using Diffusion Indexes. *Journal of Business & Economic Statistics* 20:2, 147–162.

Stock, J. H. and M. W. Watson (2003). Forecasting Output and Inflation: The Role of Asset Prices. *Journal of Economic Literature* 41:3, 788–829.

Štrumbelj, E. and I. Kononenko (2010). An Efficient Explanation of Individual Classifications Using Game Theory. *Journal of Machine Learning Research* 11:1, 1–18.

Štrumbelj, E. and I. Kononenko (2014). Explaining Prediction Models and Individual Predictions with Feature Contributions. *Knowledge and Information Systems* 41:1, 647–665.

Swanson, N. R. and H. White (1997). A Model Selection Approach to Real-Time Macroeconomic Forecasting Using Linear Models and Artificial Neural Networks. *Review of Economics and Statistics* 79:4, 540–550.

Teräsvirta, T. (2006). Forecasting Economic Variables with Nonlinear Models. In: G. Elliott, C. W. J. Granger, and A. Timmermann, eds. *Handbook of Economic Forecasting*. Vol. 1. Amsterdam: Elsevier, pp. 413–457.

Tibshirani, R. (1996). Regression Shrinkage and Selection Via the LASSO. *Journal of the Royal Statistical Society. Series B (Methodological)* 58:1, 267–288.

Trapletti, A., F. Leisch, and K. Hornik (2000). Stationary and Integrated Autoregressive Neural Network Processes. *Neural Computation* 12:10, 2427–2450.

Vrontos, S. D., J. Galakis, and I. D. Vrontos (2021). Modeling and Predicting U.S. Recessions Using Machine Learning Techniques. *International Journal of Forecasting* 37:2, 647–671.

West, K. D. (1996). Asymptotic Inference About Predictive Ability. *Econometrica* 64:5, 1067–1084.

Yousuf, K. and S. Ng (2021). Boosting High Dimensional Predictive Regressions with Time Varying Parameters. *Journal of Econometrics* 224:1, 60–87.

Zou, H. and T. Hastie (2005). Regularization and Variable Selection Via the Elastic Net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 67:2, 301–320.