Journal Pre-proof

Industry Return Prediction via Interpretable Deep Learning

Lazaros Zografopoulos , Maria Chiara Iannino ,
Ioannis Psaradellis , Georgios Sermpinis

PII: S0377-2217(24)00687-8
DOI: https://doi.org/10.1016/j.ejor.2024.08.032
Reference: EOR 19162

To appear in: *European Journal of Operational Research*

Received date: 20 June 2023
Accepted date: 30 August 2024

Please cite this article as: Lazaros Zografopoulos , Maria Chiara Iannino , Ioannis Psaradellis , Georgios Sermpinis , Industry Return Prediction via Interpretable Deep Learning, *European Journal of Operational Research* (2024), doi: https://doi.org/10.1016/j.ejor.2024.08.032

Highlights for: Industry Return Prediction via Interpretable Deep Learning

- We apply LassoNet to forecast and trade U.S. industry portfolio returns.
- The model combines a regularization mechanism with a neural network architecture.
- Our findings reveal that the LassoNet outperforms various benchmarks.
- Valuation ratios are the most crucial covariates behind LassoNet performance.
- A trading application translates the forecasts to profitable trades.

Industry Return Prediction via Interpretable Deep Learning

Lazaros Zografopoulos[†], Maria Chiara Iannino[*], Ioannis Psaradellis[†], Georgios Sermpinis[§±]

Abstract

We apply an interpretable machine learning model, the LassoNet, to forecast and trade U.S. industry portfolio returns. The model combines a regularization mechanism with a neural network architecture. A cooperative game-theoretic algorithm is also applied to interpret our findings. The latter hierarchizes the covariates based on their contribution to the overall model performance. Our findings reveal that the LassoNet outperforms various linear and nonlinear benchmarks concerning out-of-sample forecasting accuracy and provides economically meaningful and profitable predictions. Valuation ratios are the most crucial covariates, followed by individual and cross-industry lagged returns. The constructed industry ETF portfolios attain positive Sharpe ratios and positive and statistically significant alphas, surviving even transaction costs.

Keywords: Finance; Forecasting; Machine Learning; Deep Learning; Feature Importance

[†] School of Economics and Finance, University of St Andrews, lz44@st-andrews.ac.uk
[*] School of Economics and Finance, University of St Andrews, mci@st-andrews.ac.uk
[†] School of Economics and Finance, University of St Andrews, ioannis.psaradellis@st-andrews.ac.uk
[§] Adam Smith Business School, University of Glasgow, Georgios.Sermpinis@glasgow.ac.uk
[±] Corresponding Author

1. Introduction

There is a significant strand of literature showing that traditional asset pricing factor models do not perform well, or at least as well as initially advertised, in explaining the cross-section of stock returns or stock return predictability (Ferson and Harvey, 1991; Ferson and Korajczyk, 1995; Ferson and Harvey, 1999; Avramov, 2004, Lewellen, et al., 2010). The return predictability they evinced could result from asset pricing misspecifications. Scientific remedies suggest constructing efficient aggregate portfolios, such as industry portfolios. However, even these solutions still need to improve the explanatory power of linear asset pricing models (see Lewellen et al., 2010). For that reason, recent studies investigate stock return predictability via machine learning (ML) techniques better suited to uncover nonlinear patterns and cross-sectional relationships among firm and fund characteristics (see among others, Krauss et al., 2017; Fischer and Krauss, 2018; Gu et al., 2020; DeMiguel et al., 2023). Those techniques have the advantage of addressing issues arising from many irrelevant or highly correlated predictors while minimizing the risk of overfitting, contrary to the widely used linear methods.

However, there are only a few insights into how these methods capture precisely the relationships between predictors and forecasts in the context of asset pricing. ML interpretability is an essential tool in empirical asset pricing applications as we can understand which input variables affect the output the most and better identify the problem (see Brigo et al., 2021); likewise, a human who can consistently predict the result of a model and explain the logic behind the result (see, Kim et al., 2016). Existing literature using linear asset pricing models has shown that profitability (see Fama and French, 2015 and Ball et al., 2016), liquidity (see Pastor and Stambaugh, 2003) and industry interdependencies (see Rapach et al., 2015) are some of the most significant factors in determining expected returns of stocks and industry portfolios of stocks. Thus, it is worth revising those empirical findings under the prism of a nonlinear investigation.

Moreover, sparse literature investigates the aggregate stock return predictability at the industry portfolio level. Most of the studies mainly focus on applying linear methodologies, such as the Ordinary Least Squares (OLS) and Lasso to industry returns forecasting, and they often rely on a relatively small number of predictors (see Cohen and Frazzini, 2008; Menzly and Ozbas, 2010; Rapach et al., 2015, 2019). Besides the aggregate market return predictability, the increasing popularity of industry exchange-traded funds (ETFs) and efficient capital allocation across industries make industry-sorted portfolios' predictability an interesting topic for academics and practitioners.

This study uses a deep learning framework that uniquely unveils the underlying structure between forecasts and predictors to predict industry portfolio returns. Specifically, we apply the LassoNet method of Lemhadri et al. (2021) to forecast U.S. industry portfolio returns by extracting information from a large-scale dataset of multiple predictors. This is achieved by jointly selecting a subset from a large class of input variables and minimizing the objective loss function of a neural network in a mathematically elegant way. Lasso and Neural Network specifications have received great attention from numerous studies over the past years, which highlight the high performance of those methods in financial applications (see Huang and Shi, 2022; Chen et al., 2023, DeMiguel et al., 2023). LassoNet can capture data nonlinearities via a deep learning mechanism and perform feature selection due to its Lasso-type component simultaneously, unlike standard neural network architectures.[1] As a result, the set of the selected covariates drives the model's forecasts endogenously, thus providing more accurate predictions.

However, LassoNet is only partially interpretable because it performs feature selection rather than quantifying the importance of the selected features on the model's performance. To assess how much each feature contributes to LassoNet's performance overall and so to uncover the factors leading our forecasts precisely, we use post-LassoNet, the SAGE method of Covert et al. (2020), which is an additive global importance interpretability method relying on Shapley values. We input the features selected by LassoNet to SAGE to shed more light on the importance of each selected covariate on industry portfolio predictability. The advantage of SAGE lies in using a cooperative game theoretic framework to identify the predictors that contribute the most to reducing the out-of-sample (OOS) error or increasing the models' predictability. SAGE achieves this by considering all possible interactions and individual components (e.g., specific lags) across the dataset of predictors in optimizing the model's performance. This granularity is crucial for model refinement and feature selection. For example, identifying which lags of a variable are most influential can streamline the models to focus on the most relevant features, potentially improving model efficiency and performance. SAGE features contrast with other commonly used interpretability methods, such as the partial derivatives or the regular SHAP method, which deal with the issue only by identifying the variables causing the most significant variation in the model output or how much each feature contributes to a single prediction.[2] This is the first time LassoNet and SAGE methods have been applied to a financial study, either separately or jointly. The above two-step approach is employed in forecasting and trading 10 U.S. industry portfolio returns over the 2010 – 2019 period based on a large

[1] We are also aware of other interpretable deep learning approaches, see for example the Temporal Fusion Transformer (TFT) of Lim et al. (2021). However, the properties of these methods do not suit our problem. TFT specifically, has limited predictability in short-term horizons, mainly accommodates static covariates, and is more computationally demanding than LassoNet. Our purpose is to perform one-month ahead forecasts based on a large set of time-variant covariates as an investor would have done in practice.

[2] Other methods, well suited for causal relationships in time series data include deep learning-based Granger causality applications. However, those can indicate only whether a variable helps forecast another.

universe of 88 predictors, including financial ratios and past returns for each industry, *other-industry* past returns and macroeconomic variables. In this way, we provide a detailed exercise of LassoNet and SAGE on financial time series forecasting. We compare our results with those obtained by linear (i.e., linear regression, Group Lasso, Elastic-Net) and several nonlinear ML models (i.e., XGBoost and two neural network specifications) as proposed by the previous literature. Finally, we assess the economic significance of LassoNet forecasts by forming long-short portfolios of corresponding industry ETFs based on the highest-lowest forecasted returns of industry portfolios. In this way, we also evaluate LassoNet's ability to allocate capital to aggregate portfolios.

Our findings indicate that LassoNet outperforms in the OOS, all benchmark methodologies employed across the ten industries examined in predicting industry portfolios' returns. For instance, LassoNet achieves the smallest forecasting error compared to other powerful ML models, such as the XGBoost and standard neural networks. Those findings are justified by pairwise statistical significance tests for predictability, such as those of Diebold Mariano (1995) and Giacomini and White (2006), as well as tests for statistical inference of multiple benchmark forecasts at the same time and while accounting for *alpha-level inflation* such as those of Hansen (2005) and Hansen et al. (2011). The SAGE method reveals the importance of valuation ratios, individual and cross-industry lagged industry returns to our predictions and so their high explanatory power on industry portfolios' returns. Thus, we provide new evidence in the linear asset pricing literature, which reports that profitability (see Fama and French, 2015 and Ball et al., 2016) and liquidity (see Pastor and Stambaugh, 2003) are some of the most significant factors of the expected returns of stocks and portfolios of stocks. Regarding the economic significance of the proposed model, the most profitable long-short portfolio of ETFs in terms of mean return is the one constructed on LassoNet's industry predictions and it generates a post-transaction costs Sharpe ratio of 2.13 and statistically significant four and five-factor alphas of 21.1% and 20.9% per annum, respectively. The same portfolio outperforms portfolios constructed with the forecasts of the competing ML specifications and several benchmarks (e.g., the CRSP and S&P 500 indices, equally-weighted portfolio of industry ETFs) across a battery of performance measures.

The remainder of this paper is structured as follows. Section 2 provides a literature review. Section 3 presents our methodology and modelling framework. Section 4 describes our dataset and experimental design. Section 5 covers the main empirical results. Section 6 reports a subperiod analysis as a robustness check. Finally, section 6 concludes.

2. Literature Review

Our work is linked to the emerging operations research literature of ML methods applications on forecasting equity returns, either as single stocks or industry portfolios of stocks (see among others, Rapach et al., 2015; Krauss et al., 2017; Fischer and Krauss, 2018; Rapach et al., 2019; Gu et al., 2020; Bianchi and McAlinn, 2021). Krauss et al., (2017) use DNNs, gradient-boosted-trees, and random forests to predict the probability of each of the S&P 500 constituents' daily returns to outperform the market cross-sectionally for 1992 – 2015. The authors construct long-short portfolios based on those predictions while they use different lagged returns of each stock as covariates. The method generating the highest return is an equally weighted ML approach ensemble. In a follow-up paper, Fischer and Krauss (2018) assess the predictive power of a state-of-the-art long short-term memory neural network (LSTM) on classifying the same universe of stocks based on the cross-sectional median. The LSTM method outperforms all benchmark models (i.e., random forest, neural network, and logistic regression) in statistical and economic terms. Gu et al. (2020) perform a large-scale comparative analysis of the most popular ML algorithms (e.g., penalized regressions, principal component analysis, regression trees, neural networks) in predicting 30 thousand U.S. stocks from March 1957 to December 2016. The study uses 94 stock-level characteristics as predictors in the forecasting experiment of stock returns in a panel data format. It concludes that these models offer an improved description of the expected return behaviour relative to traditional forecast methods. DNNs were the best specification in forecasting and trading tasks, generating the highest R-squared and Sharpe ratios, respectively.

Concerning the industry portfolios' return predictability via ML, Rapach et al. (2015) examine the predictability of the adaptive Lasso model of Zou (2006) for shrinkage and optimal variable selection on 30 industry portfolios while using as predictors monthly lagged returns of different industries for the period 1960 – 2014. Their findings report significant evidence of cross-industry returns predictability, with at least four lagged industry returns being significant predictors. These results are also verified by principal component and partial least squares methods by extracting the latent factors of industry returns. In a similar setting and for the same dataset, Rapach et al. (2019) use a Lasso model for dimensionality reduction of lagged industry returns predictors while applying an OLS post-Lasso regression to estimate predictor coefficients and so better forecast industry returns accurately. They also employ a multiple-hypothesis testing framework to assess the statistical significance of the selected predictors. The findings align with those of Rapach et al. (2015) while being statistically significant. The authors also test the economic significance of the above predictability by constructing industry spread portfolios, which generate higher performance than naïve benchmarks. Recently, Bianchi and McAlinn (2021) propose an ensemble of linear predictive regressions for industry portfolio returns, considering the correlation structure of 75 covariates, especially when highly correlated. Even though their proposed method does not belong to the class of nonlinear methods, they compare its performance with that of conventional ML techniques. They conclude that financial ratios provide valuable information for forecasting stock returns at the industry and aggregate market levels.

In addition, we relate our study to the recent literature attempting to interpret the ML *black box* properties in financial applications. Seminal studies implementing different techniques that assign importance measures to the individual financial covariates include those of Gu et al. (2020) and Kim et al. (2020). Gu et al. (2020) mainly use two standard techniques for feature importance. The first one evaluates the reduction in the R-squared of the predictive regression by setting each covariate to zero while keeping the rest of the predictors unmodified. The second approach assesses the sensitivity of the forecasting model fit to changes in a covariate by measuring the sum squared partial derivatives of the model to each predictor. Such an approach is assumed conventional in ML literature (see also Dimopoulos et al., 1995). Recently, Kim et al. (2020) use a DNN to forecast the profitability of retail investors in spread trading while considering different feature groups related to investors (e.g., past performance, preferences in markets and channels, demographics, etc). They employ the information-fusion-based sensitivity analysis (IFBSA) of Delen et al. (2007) as their primary feature importance method to obtain the most informative predictors. IFBSA tests the marginal impact of a predictor on the error of a model without a specific covariate concerning the model, which includes that covariate. At the same time, the procedure is repeated for all covariates.

Our study extends the existing literature by applying an interpretable Lasso-based deep learning method in forecasting and trading industry portfolio returns. We demonstrate our approach's forecasting accuracy and economic significance compared to other commonly used ML methods and reveal the covariates that drive this performance. Compared to the existing literature, we create and use a large-scale set of 88 predictors, consisting of 10 distinct categories of financial ratios, past returns and macroeconomic variables, by bringing together and expanding predictors used from the past literature for industry return predictability (e.g., Rapach et al. 2019; Bianchi and McAlinn, 2021). Thus, we increase the universe of the covariates under study compared to the existing literature and offer new insights into their significance in the predictability and profitability of industry returns. Overall, our study can be of great interest to researchers and policymakers to efficiently predict financial market movements and make informed decisions about optimal trading execution and capital allocation.

3. Methodology

In this section, we discuss the proposed deep learning framework. In the first subsection, we start with a detailed description of the LassoNet model and its main advantages. In the following subsection, we discuss the model’s hyper-parameterization. The last subsection provides a discussion of the game-theoretic framework of Shapley Additive Global importancE (SAGE) (Covert et al., 2020), which is used to assign importance scores to the selected covariates. To adequately assess LassoNet's predictive ability, we also compare its forecasting performance against several benchmark models, which are presented in the Online Appendix A. The benchmark model set includes linear regression, Group Lasso, Elastic-Net, two Neural Network models, a simple MLP and Lasso combined with an MLP (Lasso-MLP), and XGBoost. The hyperparameters for all models are tuned in-sample (IS). Following the common practice, we employ the early stopping and sensitivity analysis procedure for the neural networks (see Krauss et al., 2017; Fischer and Krauss, 2018; Gu et al., 2020) and cross-validation for all other machine learning models. Using this extensive benchmark model set enables us to compare the LassoNet with models that perform covariate selection (group lasso regression, elastic net regression, Lasso-MLP) as well as models that use the full covariate set for their forecasts (linear regression, MLP, XGBoost). The description of the benchmark models, as well as their optimization hyperparameters, are reported in Online Appendix A.

3.1 LassoNet

We define an asset's excess return as the sum of its conditional expected return and the prediction error component. The conditional expected return of an industry portfolio i at time t+1 can be represented as a function of covariates that maximizes the OOS prediction of the realized return, $r_{i,t+1}$, in a nonlinear setting (see also Gu et al., 2020):

$$r_{i,t+1} = \mathbb{E}\left(r_{i,t+1}\right) + e_{i,t+1} = g^{\star}\left(\mathbb{X}_{i,t}\right) + e_{i,t+1}$$

where the conditional expected return $g^{\star}(\bullet)$ term represents a nonlinear flexible function that a machine learning model parameterizes, $\mathbb{X}_{i,t}$ is a D-dimensional vector of covariates and $e_{i,t+1}$ is the error term. In our case, we use a balanced panel dataset $\left\{\left(\mathbb{X}_{i,t}, r_{i,t+1}\right)\right\}_{1\leq i\leq n}$ spanning across the covariate set for the ten industries and the period examined in our study. We denote with $r_{i,t+1}$ the industry returns, the target variable in our forecasting task. To construct the mapping $g^{\star}: \mathbb{X}_{i,t} \mapsto r_{i,t+1}$ for each industry, the LassoNet method extends the traditional linear regularized regression models by simply adding a nonlinear component (see Lemhadri et al., 2021). In essence, the added term is the nonlinear transformation of the input variables as they propagate forward through the layers of a neural network with activation functions. Mathematically, the LassoNet for each industry is formulated as follows:

$$g^{\star} \equiv g_{\theta,W}: \mathbb{X}_{i,t} \mapsto \theta^{T}\mathbb{X}_{i,t} + H_{W}(\mathbb{X}_{i,t})$$

where $g^{\star}$ is a class of residual feed-forward neural networks of arbitrary width and depth.[3] The network is parameterized by weights $\{(\theta, W)\}$, where θ denotes the vector of weights in the residual layer (i.e., skip-connection), and W denotes the vector of weights in the hidden layer of a fully connected feed-forward network H_w. Hence, $\theta^T \mathbb{X}_{i,t}$ corresponds to the linear component, and $H_W(\mathbb{X}_{i,t})$ corresponds to the nonlinear component of the neural network architecture. Following Lemhadri et al. (2021), the objective function for each industry's prediction takes the form of:

$$\min_{\theta, W} L(\theta, W) + \lambda \|\theta\|_1$$

$$subject\ to \left\| W_p^{(1)} \right\|_\infty \leq \mathrm{M} \left| \theta_p \right|, \qquad p = 1, 2, \ldots, D$$

where $L(\theta, W)$ is the loss function, λ denotes the feature sparsity penalty parameter, $W_p^{(1)}$ indicates the weights for covariate p in the first hidden layer, and M is a hierarchy coefficient.[4] The key features of LassoNet are introducing a penalty term in the loss function, which enforces covariate selection, and the so-called *hierarchy coefficient* M, which controls the relative strength of linear and nonlinear components of the model. The residual and the first hidden layer are jointly optimized.

The main innovation of the model lies in the constraint $\left\| W_p^{(1)} \right\|_\infty \leq \mathrm{M} \left| \theta_p \right|$, which indicates that a covariate p is not involved in the feed-forward network (i.e. $W_p^{(1)}$=0) if the residual layer weight is zero (i.e., $\theta_p = 0$). In other words, the constraint budgets the level of participation of a covariate p in the nonlinear operations of the model (i.e., first and subsequent layers) based on its relative importance, which is achieved by tying every covariate to the single coefficient, θ, of the linear component (i.e., skip-connection). In this way, the linear component is used to guide feature sparsity in the nonlinear component (i.e., feed-forward neural network), and both components are fitted simultaneously to capture nonlinear patterns in the dataset via the neural network. Moreover, a closer inspection of the objective function reveals that the LassoNet nests the linear Lasso and the standard feed-forward neural network in cases where the *hierarchy coefficient,* M, takes the extreme values of zero and infinity, respectively.

The estimation of Lassonet includes a standard backpropagation process, which is initially applied to all model parameters, and a proximal operator is applied on the input layer's set of weights (i.e., $\{\theta, W^{(1)}\}$). Specifically, we use gradient descent with Adam optimizer to update the LassoNet set of

[3] The exact LassoNet architecture is different for each industry after hyperparameter optimization since we train the model separately for each industry.

[4] Penalization of the weights is only required for the neurons' first hidden layer because of the feed-forward architecture of the network, while the λ penalty term in the objective function operates in the same way as in the standard linear Lasso model.

weights as a first step. A *hierarchical proximal operator* is applied exclusively to the skip-connection weights, θ, and the neural network weights connecting the covariates to the first hidden layer, $W^{(1)}$, as a second step.[5] To fit our dataset optimally, we examine deep neural network architectures with multiple hidden layers to ensure that Lassonet can better capture all possible nonlinearities in our financial dataset. We also implemented the hyperbolic tangent ($\tanh(\bullet)$) as the activation function and the Mean Squared Error (MSE) as the loss function for our LassoNet implementation.[6] The exact hyperparameter tuning is described in the following section.

3.2 LassoNet's Hyperparameters

In this section, we present the LassoNet's hyperparameters optimization process. We optimize LassoNet by using early stopping and sensitivity analysis in the I.S. To conduct early stopping, we use a validation dataset, which is constructed by splitting the I.S. into training and validation sub-samples. We use the first 18 years (i.e., 1985 – 2002) as the training dataset and the last seven years (i.e., 2003 – 2009) as the validation dataset (see also Granger, 1993; Dunis et al., 2011; Gu et al., 2020). Sensitivity analysis in conjunction with early stopping results in the optimal combination of hyperparameters, achieving the lowest mean squared error in the validation sub-sample. We keep this network as the optimal LassoNet architecture for generating the OOS forecasts. More specifically, we optimize the number of hidden layers, hidden neurons, and hyperparameters M and λ

with early stopping and sensitivity analysis.

Regarding LassoNet's nonlinear component specification, we follow Gu et al. (2020) and decide on the optimal LassoNet specification from a fixed candidate model set. We avoid shallow neural network architectures with a single hidden layer in the candidate model set because of possible nonlinearities in our dataset (i.e., a vast set of covariates). It has also been shown that deeper architectures with multiple hidden layers outperform shallower ones with a single hidden layer due to the higher-order nonlinear interactions between the covariates (see Mhaskar et al., 2016). However, given that industry portfolios' monthly data frequency limits the number of samples available for the model's estimation, we do not explore architectures with more than three hidden layers to avoid model overfitting. Similar to other studies, we explore architectures with a higher number of neurons in the first layer(s) followed by a layer(s) with a smaller number of neurons (see Gu et al., 2020). Based on the above reasoning, we explore two LassoNet architectures with two hidden layers (i.e., (16 4), (16, 8)) and two architectures with three hidden Layers (i.e., (16, 8, 4), (16, 16, 4)). To determine the number of

[5] The detailed optimization steps can also be found in Lemhadri et al. (2021).

[6] This is $L(\theta_i, W_i) = \frac{1}{n}\sum_{\tau=t+1}^{t+n}(r_{i\tau} - \hat{r}_{i\tau})^2$

neurons for the first hidden layer, we follow again the relevant literature (see Gu et al., 2020; Filippou et al., 2022) and a common rule of thumb (i.e., the number of neurons equals the square root of the number of covariates).

For the *hierarchy coefficient* M, we investigate four values (i.e., 0.005, 0.05, 0.5, 5) and choose the optimal one via e early stopping and sensitivity analysis procedures. To obtain the optimal value of $the\ \lambda$ hyperparameter, we follow Lemhadri et al. (2021) and adopt a heuristic mechanism based on a sequence of λ values. The initial value of λ equals the generated MSE of a conventional MLP estimated with the selected covariates on the validation data. The exact MLP architecture is defined above (e.g., 16 4). The initial value of λ is used to run the LassoNet algorithm as described in Section 3.1. Then, the estimation of the following λ hyperparameters is given based on a *regularization path multiplier* of 1.05. For instance, the algorithm is re-estimated every time the λ is increased based on that multiplier. The procedure continues until the λ reaches a value that imposes a regularization powerful enough that the LassoNet selects no covariate. Table 1 presents the different hyperparameter configurations of the LassoNet model. We examine all possible combinations of 4 different hidden layer architectures and four different M values for 16 candidate configurations.

Table 1. Hyperparameter search space for the LassoNet model

This table reports the hyperparameter search space for the LassoNet model. We use the validation dataset to choose the set that generates the lowest mean squared error metric.

Architectures	**[(16, 4), (16, 8), (16, 8, 4), (16, 16, 4)]**
M parameter	[0.005, 0.05,0.5, 5]
Regularization Path multiplier	[1.05]

For LassoNet's iterative estimation algorithm, we keep the number of training iterations, known as epochs, at 200. We also use a batch size of 72 observations. To retain the temporal ordering of the data, we enforce that batch construction follows the time sequence of the observations and that the data are not shuffled before feeding them to LassoNet's training algorithm. Finally, we use 50 model training iterations for the early stopping mechanism to avoid model overfitting. The final optimized LassoNet provides transparency regarding which covariates drive its forecasts. Moreover, it provides an input variable selection mechanism to handle high-dimensional asset pricing datasets and regularize a large pool of covariates. However, even for the case of the LassoNet, there still needs to be one solved issue of determining the importance of each covariate.

3.3 Estimating Covariates' Importance

For many years, the financial literature criticised standard ML approaches for their black-box properties, or in other words, not providing the importance of financial and economic factors on target forecasts as economic and asset pricing theory would expect. We apply an extra step on top of LassoNet's feature selection property to identify the covariates that drive our model's forecasts. As described above, the optimized LassoNet selects specific covariates through its penalized regression component in nonlinear environments. However, it does not perform feature importance in ranking covariates based on the extent to which the model depends on each of them overall. For that reason, we employ a cooperative game-theoretic method based on Shapley values to measure the importance of each selected covariate. The SAGE method of Covert et al. (2020) provides the Shapley values, which quantify the ranking in the importance of each covariate or group of covariates. We notate the Shapley values estimated by the SAGE as *SAGE* values. The significant contribution of SAGE, contrary to the commonly used SHapley Additive exPlanations (SHAP) method of Lundberg and Lee (2017), is that it represents a global interpretability method which provides feature importance by considering the behaviour of the model across the whole dataset and not how much each feature contributes to an individual prediction (i.e., every month).[7] It also fundamentally differs from other interpretability methods (e.g., partial derivatives and the deep-learning-based Granger approaches), which only measure importance in a way that finds which covariates cause the most considerable variation in the model output. SAGE helps to understand not just whether a variable is important but also how different aspects, including individual lags in the time series of that variable, contribute to the performance (i.e., which features or sequences of features most influenced a model's performance). Overall, affecting only the model output does not necessarily indicate a covariate as informative, so it is more insightful to identify which covariates drive the forecasts and, at the same time, consider whether these covariates enhance the model's predictive accuracy.

The SAGE method assigns credit to the covariates based on their contribution to lowering the LassoNet’s loss OOS (i.e., $L\left(\mathbb{E}\left[g^{\star}\left(\mathbb{X}_{i,t}\right)\right], r_{i,t+1}\right)$). This contrasts with the SHAP method, which only assigns each feature a value representing whether it pushes the prediction higher or lower. Additionally, we acknowledge that covariates contribute different information when inputted into a model together with other covariates versus being in isolation (see Covert et al., 2020). To properly account for variable interaction effects and synergies, we consider all possible subsets of our selected covariate set and then measure the degree of increase of the model's error without a specific

[7] We use the SAGE Python library to estimate SAGE values based on the papers Covert et al. (2020) and Covert and Lee (2021).

covariate. This process is repeated by focusing on a different covariate each time. Implementing this method involves constructing a cooperative game v_f that represents the model's overall performance and is defined as follows:

$$v_f(S) = -\mathbb{E}\left[L\left(\mathbb{E}\left[g^\star(\mathbb{X}_{i,t})\middle|\mathbb{X}_{i,t}^S\right], r_{i,t+1}\right)\right]$$

where $g^\star$ represents the optimized LassoNet, S indexes a subset of the total number of covariates (i.e., $S \leq D$), v_f quantitatively represents the model's performance given the subset $\mathbb{X}_{i,t}^S$ of covariates. The minus sign in front of the loss indicates that a lower loss increases the value of $v_f(S)$.

According to the above framework, a specific Shapley value, $\varphi_p(v_f)$, is attributed to each covariate p to quantify the contribution to lowering the model's prediction error. Only the covariates with positive values $\varphi_p(v_f) > 0$ are essential for the forecast and instrumental in increasing the model's statistical accuracy and improving its forecasting performance. The estimated Shapley values represent importance scores for the corresponding covariates when they are estimated for cooperative games in the form of $v_f(S)$. Finally, we identify covariate importance on an aggregated category level. To arrive at this estimation, we sum the SAGE values for all covariates that were selected by the LassoNet and belong to the same category of variables.

4. Data and Experimental Design

We forecast 10 U.S. industry portfolio returns as given by Kenneth French's website for the period 1985 to 2019 in a rolling-window format, using January 2010 to December 2019 as the OOS. The forecasting horizon is one month ahead. The ten industry sectors we examine in this study are: Consumer Durables (DURBL), Energy (ENRGY), High-Technology (HITEC), Health (HLTH), Manufacturing (MANUF), Consumer Nondurables (NODUR), Shops (SHOPS), Telecommunications (TELCM), Utilities (UTILS), and the other remaining industry sectors merged (OTHER)[8]. Accordingly, the industry returns correspond to the value-weighted average of their constituent stocks.

For our prediction task, we construct a set consisting of 88 covariates. To construct our dataset, we use the Compustat database from the Wharton Research Data Services (WRDS) platform and precisely 63 industry financial ratios, which belong to capitalization, efficiency, financial soundness, solvency, liquidity, profitability, and valuation categories. Capitalization ratios measure the debt component of a firm's total capital structure; efficiency ratios capture the effectiveness of the firm's

[8]The industry definitions are available on Kenneth French's website: https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/Data_Library/det_10_ind_port.html.

usage of assets and liability; financial soundness and solvency ratios capture the firm's ability to meet long-term obligations; liquidity ratios measure a firm's ability to meet its short-term obligations; profitability ratios measure the ability of a firm to generate profit; valuation ratios estimate the attractiveness of a firm's stock. To aggregate financial ratios at the industry level, we take the median value from the companies belonging to the specific industry. The covariates set is on a monthly frequency. In the case of only quarterly or annual data for some ratios, the most recently available observation item is carried forward to fill each month. All observations are lagged by two months to avoid look-ahead bias issues and ensure that the information was publicly announced at a given timestamp in our dataset. Following the findings of Rapach et al. (2015), which provide evidence of industry interdependencies and cross-industry return predictability, we also decided to include other industries' lagged returns to extend our dataset of covariates further. Thus, for each U.S. industry portfolio, we also include up to 12-month excess lagged returns and the 1-month value-weighted lagged returns of all other nine industries as extra covariates.[9] Our dataset also includes four macroeconomic variables downloaded from the FRED database, namely the Chicago Fed National Financial Conditions Index (NFCI), Chicago Fed National Activity Index (CFNAI), Chicago Fed National Activity Index: Production and Income (PANDI), and the Consumer Price Index (CPI). The NFCI index captures U.S. financial conditions in money, debt, and equity markets and the traditional and *shadow* banking systems. The PANDI index provides information regarding the national economy's expansion with respect to its historical trend rate of growth. The CFNAI index captures overall economic activity and the related inflationary pressure, while the CPI is used as an inflation index. In the trading application, we use ETFs prices from the CRSP Stocks and Mutual Funds datasets. We select all traded index-funds identified by share code 73, with style stated as *Equity*, *Domestic* and *Sectorial*. Then, we keep the ETFs which include in their name the industry classification closest to the Fama-French industries and with the longest time series.[10]

Regarding the experimental design, our full sample ranges from January 1985 until December 2019. We use 2010 – 2019 as the OOS period and the previous 25 years of monthly data (i.e., 300 months/ data points) as I.S. in a rolling-window structure. Our partition corresponds to the 70%-30% split

[9] We use the 3-Month Treasury Bill to calculate excess returns. The data were downloaded from the FRED database: https://fred.stlouisfed.org/

[10] We present all the employed covariates for our predictive task, their corresponding categories, and the ETF details used for our trading simulation in Online Appendix B. We have also trained LassoNet with a broader set of 98 covariates, including more macroeconomic and volatility predictors than those followed by the relevant literature. More specifically, we added in the covariates set the 3-Month Treasury Bill, the Implied Volatility Index (VIX), the dividend yield of the S&P Global index, the momentum factor, three commodities (silver, gold, crude oil), and three exchange rates (EUR/USD, GPB/USD, YEN/USD). We find that the LassoNet's performance remains the same by including additional covariates, so they did not comprise additional information beyond the initial set of 88 covariates. The relevant results are available upon request.

commonly employed in the related literature (see also Harvey and Liu, 2015). As both LassoNet and the ML benchmarks are computationally demanding, we refit all models yearly and use IS data to produce forecasts for the following year (i.e., 12 months) (in line also with the related literature i.e. Gu et al., 2020; Chen et al., 2023; DeMiguel et al., 2023). Also, aligned with the fact that some of the covariates have available only quarterly or annual data, as described above, is rational to refit ML models on a common basis for all signals of our covariates. Figure 1 presents the separation of the full sample in IS and OOS periods.

Figure 1. In-sample and out-of-sample partition of monthly observations

In-sample	Out-of-sample
1985/01-2009/12	2010/01-2019/12

We also generate forecasts and evaluate LassoNet's predictive ability and covariates importance on four OOS subperiods (i.e., 2000-2006, 2007-2009, 2010-2014, 2015-2019) for robustness purposes. This way, we examine how the model's performance varies across time and business cycles. The relevant results of the subperiod analysis are available in the Online Appendix C and confirm the superiority of LassoNet in predicting industry portfolios across the four different subperiods.

5. Empirical Results

5.1 Forecasting Accuracy

We conduct a forecasting evaluation of the LassoNet and benchmark models' forecasts by computing the root mean squared error (RMSE) and the mean absolute error (MAE) metrics. Table 2 presents the OOS error metrics across the ten industries from 2010 to 2019. The results indicate that the LassoNet consistently outperforms all the other benchmark models across most industries by generating the lowest prediction error metrics under the RSME and MAE criteria. Consequently, the optimization process in LassoNet is superior to simply optimizing a standard regularization linear model or feed-forward neural networks. The outperformance stems from LassoNet's algorithm, which jointly optimizes linear and nonlinear components, enabling LassoNet to retain the advantages of both components without retaining any limitations. We show that covariate selection enhances a model's forecasting ability OOS, which can provide additional reasoning on why LassoNet outperforms a standard neural network trained on the complete set of covariates.

Table 2. OOS statistical performance for the LassoNet and the employed benchmark models.

The table reports the OOS statistical performance over the 2010 – 2019 period. For each industry, we compare the performance of the LassoNet model against the employed benchmarks (i.e., OLS-Regression, Group-Lasso, Elastic-Net, MLP-NN, Lasso-ANN-MLP, and XGBOOST). We report the root mean squared error (RMSE) and the mean absolute Error (MAE) as error metrics. In Panel B, we also present the range for the RMSE and MAE metrics, which are calculated by the difference between each error metric's highest and lowest values. The lowest values are reported in bold.

Panel *A*: *Loss Criteria*	**LASSONET**	**OLS**	**GROUP LASSO**	**ELASTIC NET**	**MLP**	**LASSO-MLP**	**XGBOOST**		**LASSONET**	**OLS**	**GROUP LASSO**	**ELASTIC NET**	**MLP**	**LASSO - MLP**	**XGBOOST**
	DURBL								**ENRGY**						
RMSE	**0.0618**	0.0821	0.0630	0.0640	0.2369	0.0658	0.0674		**0.0534**	0.1060	0.0808	0.0879	0.2300	0.1581	0.0589
MAE	**0.0467**	0.0630	0.0508	0.0527	0.1610	0.0489	0.0500		**0.0421**	0.0816	0.0643	0.0692	0.1692	0.1252	0.0474
	HLTH								**NODUR**						
RMSE	**0.0356**	0.1240	0.0415	0.0409	0.1692	0.0780	0.0439		**0.0328**	0.0640	0.0424	0.0406	0.1327	0.1115	0.0361
MAE	**0.0281**	0.0869	0.0348	0.0341	0.1257	0.0576	0.0330		**0.0259**	0.0482	0.0330	0.0316	0.1100	0.0884	0.0287
	TELCM								**UTILS**						
RMSE	**0.0353**	0.1280	0.0420	0.0390	0.1392	0.1502	0.0410		0.0364	0.0841	0.0512	0.0587	0.1996	0.1643	**0.0331**
MAE	**0.0287**	0.0778	0.0320	0.0312	0.1159	0.1102	0.0337		0.0287	0.0681	0.0404	0.0463	0.1308	0.1360	**0.0263**
	MANUF								**HITEC**						
RMSE	**0.0348**	0.0700	0.0410	0.0438	0.1538	0.1291	0.0502		**0.0341**	0.1785	0.0762	0.0786	0.1782	0.1051	0.0467
MAE	**0.0269**	0.0557	0.0325	0.0336	0.1223	0.1030	0.0379		**0.0269**	0.1421	0.0618	0.0631	0.1465	0.1376	0.0370
	SHOPS								**OTHER**						
RMSE	**0.0334**	0.0935	0.0566	0.0529	0.2673	0.2569	0.0400		**0.0384**	0.0849	0.0480	0.0511	0.1819	0.1726	0.0464
MAE	**0.0269**	0.0713	0.0422	0.0401	0.1756	0.1637	0.0312		**0.0277**	0.0609	0.0381	0.0400	0.1270	0.1303	0.0359

Panel B: *Error Metric Range*	**LASSONET**	**OLS**	**GROUP LASSO**	**ELASTIC NET**	**MLP**	**LASSO - MLP**	**XGBOOST**
RMSE Range	**0.0290**	0.1710	0.0398	0.0489	0.1346	0.1911	0.0343
MAE Range	**0.0208**	0.0938	0.0323	0.0380	0.0656	0.1148	0.0237

The XGBoost, Group Lasso and Elastic Net are the second, third and fourth-best models, with the remaining models (OLS, MLP and Lasso-MLP) showing far worse performance. For instance, XGBoost generates better forecasting performance than LassoNet for *utilities* industry portfolios.

Table 2 also presents the spread between the minimum and maximum values of the error metrics across all industries. A wider spread denotes that a specific model performs well for certain industries and poorly for others, while less variation for the error metrics represents consistent performance across industries, which is desirable. We report such an error metric range in panel B of the same table for that purpose. The range is calculated by the difference between the highest and the lowest values of each model's generated error (i.e., RMSE, MAE) across all industries. Again, LassoNet presents the lowest dispersion across error metrics' minimum and maximum values. Therefore, we can infer that the LassoNet has the most consistent performance across the ten industries, with its forecasting ability not being industry or data-dependent.

To examine the statistical significance of the performance of the LassoNet, we first perform the Diebold and Mariano (1995) (D.M.) test and the Giacomini and White (2006) (G.W.) test based on the forecasts' squared error loss functions.[11] We use these tests to compare the OOS performance of the LassoNet against each one-off implemented benchmark. The D.M. tests the null that two forecasts have equal predictive ability based on the difference of the loss functions of two forecasts against the alternative that the loss differential is different from zero. A negative and significant *t*-statistic rejects the null hypothesis, and it reports the superiority of LassoNet against the benchmark (i.e., lower loss).

Table 3 reports the generated *t*-statistics from the D.M. test and their corresponding *p*-values in parenthesis for all industries under study. D.M. test results show that the LassoNet has superior predictive ability against most benchmarks and across industries (i.e., corresponding *p*-values are below the significance thresholds, and *t*-statistics are negative).

Table 3. Diebold Mariano test results for the LassoNet against benchmark models.

The table displays the *t*-statistics and *p*-values of the D.M. (1995) test for LassoNet against each benchmark pairwise across industries for the 2010-2019 OOS period. The null hypothesis tests that LassoNet and the benchmark forecast have equal predictive ability. Bold *p*-values and *t*-statistics indicate that we reject the null hypothesis of the two forecasts' equivalence and show the superiority of LassoNet against the benchmark.

	D.M. test: *t-statistic* (*p*-value)									
	DURBL	**ENRGY**	**HITEC**	**HLTH**	**MANUF**	**NODUR**	**OTHER**	**SHOPS**	**TELCM**	**UTILS**

[11] We also implement Giacomini and Rossi's (2010) (GR) fluctuation test, which measures the relative local forecasting performance of the LassoNet model compared to one benchmark over time given changing conditions. The relevant results show the outperformance of LassoNet against the benchmarks and are presented in Online Appendix D

OLS	**-2.84 (0.003)**	**-5.06 (0.000)**	**-7.17 (0.000)**	**-4.09 (0.000)**	**-6.03 (0.000)**	**-5.00 (0.000)**	**-4.56 (0.000)**	**-5.92 (0.000)**	**-3.90 (0.000)**	**-6.40 (0.000)**
GROUP LASSO	-0.38 (0.752)	**-4.00 (0.000)**	**-6.25 (0.000)**	**-2.89 (0.002)**	**-1.76 (0.089)**	**-3.00 (0.002)**	**-2.16 (0.024)**	**-3.71 (0.000)**	**-2.49 (0.012)**	**-3.54 (0.000)**
ELASTIC NET	-0.64 (0.583)	**-4.41 (0.000)**	**-6.32 (0.000)**	**-2.59 (0.009)**	**-2.49 (0.011)**	**-2.57 (0.012)**	**-2.57 (0.010)**	**-3.54 (0.000)**	**-2.44 (0.014)**	**-4.46 (0.000)**
MLP	**-5.03 (0.000)**	**-6.02 (0.000)**	**-8.27 (0.000)**	**-5.40 (0.000)**	**-6.78 (0.000)**	**-8.92 (0.000)**	**-4.75 (0.000)**	**-4.92 (0.000)**	**-8.42 (0.000)**	**-4.28 (0.000)**
LASSO-MLP	**-2.13 (0.037)**	**-6.74 (0.000)**	**-7.04 (0.000)**	**-5.00 (0.000)**	**-8.00 (0.000)**	**-7.04 (0.000)**	**-6.68 (0.000)**	**-3.82 (0.000)**	**-5.66 (0.000)**	**-8.70 (0.000)**
XGBOOST	-1.37 (0.157)	**-2.20 (0.026)**	**-3.91 (0.000)**	**-3.03 (0.000)**	**-3.51 (0.000)**	**-2.28 (0.022)**	**-1.92 (0.081)**	**-2.37 (0.019)**	**-2.81 (0.004)**	1.46 (0.147)

There are only two industries, namely *durables* and *utilities*, in which LassoNet shows equivalent performance with the XGBoost, Group Lasso and Elastic Net and XGBoost models, respectively. For the remaining industries, LassoNet generates significantly better performance.

The G.W. conditional predictive ability test assesses the null of equal predictive ability between two models (i.e., pairwise comparison) when the forecasting model (i.e., LassoNet) may be miss-specified. G.W. test complements D.M. in terms of reflecting the effect of estimation uncertainty and permitting a unified treatment of nested and non-nested models. The latter feature is essential for our experiment as LassoNet includes both Lasso and neural network components. Also, the G.W. test uses available information to predict which forecast will be more accurate for a specific future date (i.e., one month ahead in our case), conditional on given information. This property improves D.M. test which evaluates which forecast was more accurate, on average, in the past. The alternative hypothesis of G.W. suggests which forecast performs better by producing a lower average loss than the competing model.

Table 4 presents the *p*-values generated by the G.W. test by performing a pairwise comparison of LassoNet against each benchmark for all industries examined.[12] A *p*-value rejecting the null indicates that LassoNet performs better than the benchmark. Again, our findings show that LassoNet is a better forecaster than all benchmarks for most industry portfolios. Only in the case of DURBL industry, we observe that LassoNet does not generate more accurate predictions.

[12] For the lack of space, we present only the *p*-values of the pairwise comparison of LassoNet with each one of the benchmarks. The corresponding *p*-values of benchmarks' pairwise comparisons are available upon request.

The LassoNet's predictive performance is further assessed using the unconditional *Superior Predictive Ability* (SPA) procedure of Hansen (2005) and the *Model Confidence Set* (MCS) procedure of Hansen et al. (2011) with a 10% test size. Those tests simultaneously perform an OOS statistical inference of many forecasts while controlling for data snooping (i.e., *alpha-level inflation* problem).

Table 4. Giacomini and White (2006) test results for the LassoNet against benchmark models

The table reports the Giacomini and White (2006) test *p*-values for the LassoNet against each benchmark across all industries over the 2010 – 2019 OOS period. The null hypothesis is the equal predictive ability between two models when the forecasting model (i.e., LassoNet) may be misspecified. Significant *p*-values indicate the superiority of LassoNet against the benchmark. *, **, *** denote significance at the 10%, 5% and 1% level, respectively.

	G.W. test: *p*-value									
	DURBL	**ENRGY**	**HITEC**	**HLTH**	**MANUF**	**NODUR**	**OTHER**	**SHOPS**	**TELCM**	**UTILS**
OLS	0.024**	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***
GROUP LASSO	0.756	0.000***	0.000***	0.012**	0.056*	0.014**	0.056*	0.016**	0.045**	0.002***
ELASTIC NET	0.829	0.000***	0.000***	0.041**	0.016**	0.041**	0.032**	0.003***	0.056*	0.000***
MLP	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***
LASSO-MLP	0.106	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***	0.000***
XGBOOST	0.161	0.023**	0.002***	0.009***	0.004***	0.081*	0.065*	0.033**	0.029**	0.086*

The SPA test assesses the relative forecasting performance of LassoNet, as the point of the reference model, against the complete set of benchmark forecasts. The null hypothesis of the SPA test is that the LassoNet forecast is not inferior to the best alternative model's forecast based on a given loss function. Table 5 reports that the SPA test generated *p*-values under the RMSE and MAE criteria. The SPA test results show that the corresponding *p*-values of LassoNet against the benchmarks are high enough (i.e., fail to reject the null hypothesis) to conclude that LassoNet is not inferior to the benchmarks under both the RMSE and MAE criteria and across all industry portfolios.

Table 5. SPA test results for the LassoNet and the employed benchmark models.

The table displays the *p*-values for the SPA test of Hansen (2005) over the 2010 – 2019 OOS period. The test provides insights regarding the relative performance of LassoNet against the employed benchmarks (OLS-Regression, Group Lasso, Elastic-Net, MLP, Lasso-MLP, and XGBOOST). Bold *p*-values indicate a failure to reject the null hypothesis that LassoNet is not inferior to the benchmarks.

		2010-2019									
		DURBL	**ENRGY**	**HITEC**	**HLTH**	**MANUF**	**NODUR**	**OTHER**	**SHOPS**	**TELCM**	**UTILS**
P-											

	MSE	**0.405**	**0.871**	**0.950**	**0.894**	**0.824**	**0.999**	**0.774**	**0.859**	**0.882**	**0.348**
	MAE	**0.427**	**0.815**	**0.990**	**0.998**	**0.838**	**0.990**	**0.716**	**0.991**	**0.895**	**0.361**

Implementing and making robust conclusions about predictive ability with the SPA test is challenging when there is no natural model as a reference point or when more than one model is considered. The MCS procedure of Hansen et al. (2011) addresses such an issue by bypassing point-of-reference models. The MCS aims to find a superior set of models indistinguishable from the best, including the best model. It consists of a sequence of tests which permits the construction of a set of *superior* models, where the null hypothesis of equal predictive ability is not rejected at a certain confidence level. The test requires two procedures: an equivalence test, determining whether models are equal according to their loss and an elimination rule, which dictates which model to eliminate if the equivalence test reveals that two models are not equivalent (i.e., there is one with a larger loss). The output of the MCS is a model set containing the true set of *best* models with a probability weakly larger than $1 - a$, where a is the significance level. Also, if only one best model exists, the test will find it asymptotically. As a rule of thumb, a low (high) *p*-value is associated with a model that is unlikely (likely) to belong to the set of the *best* models. Therefore, *p*-values that exceed the nominal significance levels advocate that the tested model belongs to the confidence set of *best* models (Psaradellis and Sermpinis, 2016; Grønborg et al., 2021). However, The MCS *p*-value is not a statement about the probability that a model is the best.

Table 6 presents the relevant results based on a 10% significance level. In particular, the table reports the MCS *p*-values for each model. We present the *p*-values of the models belonging to the confidence set in bold. We observe that LassoNet always belongs to the confidence set of *best* models, while for most of the cases, it is the model with the lowest loss (i.e., *p*-value = 1), except for the case of the *utilities* industry. Interestingly, LassoNet is the only model in the true set for half of the industries examined (i.e., ENERGY, HITECH, HLTH, SHOPS and TELCM). We can conclude that no models are indistinguishable from the best (LassoNet).

Table 6. MCS test results for the LassoNet and the employed benchmark models.

The table reports *p*-values for Hansen et al.'s (2011) MCS procedure over the 2010 -2019 OOS period and each industry at a 10% confidence level. A sequence of significance tests is performed to find model forecasts that are not inferior to others. *P*-values that exceed the nominal significance levels (i.e., 1%, 5%, and 10%) show that the model belongs to the MCS. The models belonging to the confidence set at the 10% significance level are reported in bold.

	MCS test: *p*-value									
	DURBL	**ENRGY**	**HITEC**	**HLTH**	**MANUF**	**NODUR**	**OTHER**	**SHOPS**	**TELCM**	**UTILS**

OLS	0.019	0.000	0.000	0.016	0.008	0.029	**0.142**	0.020	0.003	0.008
GROUP LASSO	**0.814**	0.079	0.000	0.025	**0.108**	**0.193**	**0.142**	0.060	0.056	**0.127**
ELASTICNET	**0.814**	0.079	0.000	0.016	0.098	**0.193**	**0.142**	0.060	0.056	0.014
MLP	0.019	0.000	0.000	0.014	0.000	0.000	0.093	0.020	0.000	0.008
LASSO-MLP	**0.487**	0.000	0.000	0.016	0.000	0.000	0.004	0.020	0.003	0.000
XGBOOST	**0.487**	0.079	0.000	0.016	0.098	**0.193**	**0.142**	0.060	0.056	**1.000**
LASSONET	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**0.485**

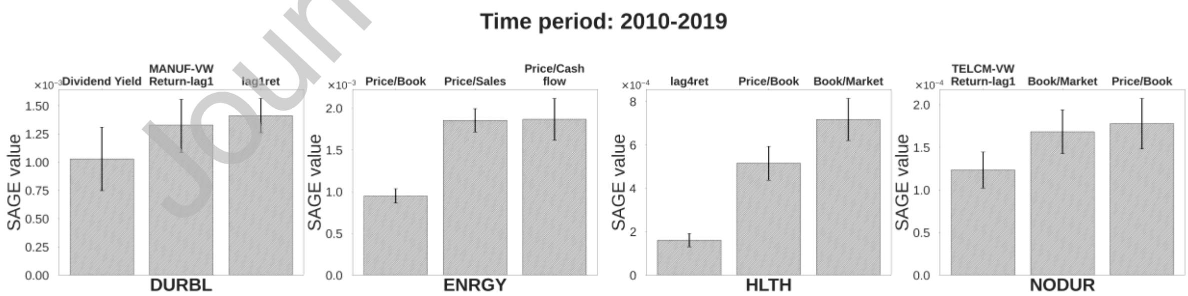
5.2 Covariates' Importance

5.1.1 SAGE Value Estimation

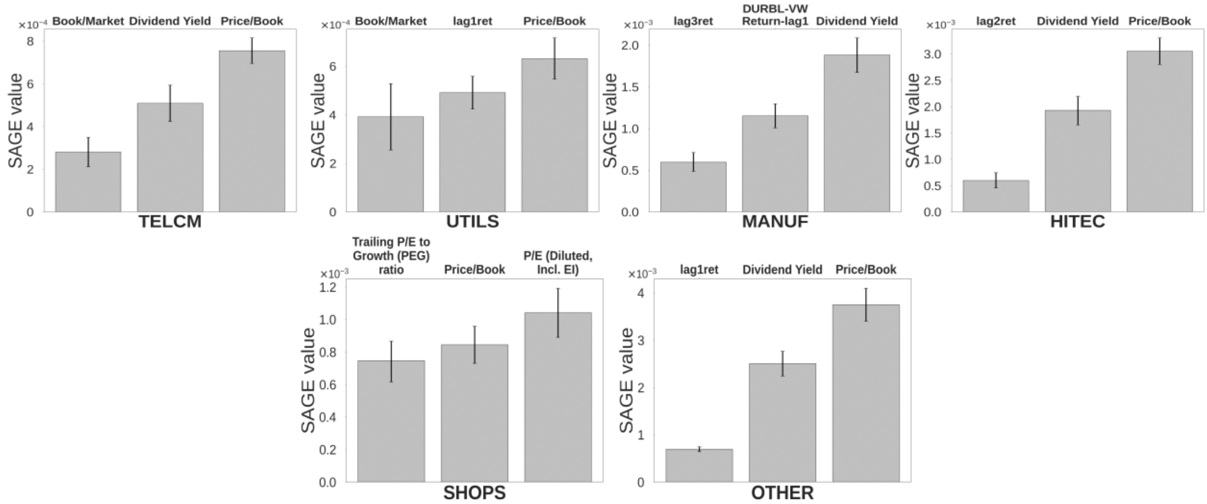
After establishing that the LassoNet outperforms all other benchmark models, it is crucial to investigate the covariates driving its forecasts. Figure 2 presents the three covariates with the highest SAGE values OOS for every industry separately, along with 95% confidence intervals around the mean SAGE value of each covariate across the rolling windows.[13]

Figure 2. SAGE values bar plots.

The figure displays the three covariates with the highest SAGE values for every industry across the 2010 – 2019 OOS period. We restrict our results to the three covariates with the highest SAGE values to investigate the most significant variables (and the categories they belong to). The bar graphs also include 95% confidence intervals around the mean SAGE value of each covariate.



[13] We obtain the OOS SAGE values per covariate by aggregating their values for each rolling window estimation while we calculated the 95% confidence intervals around the mean of SAGE values across all rolling windows.

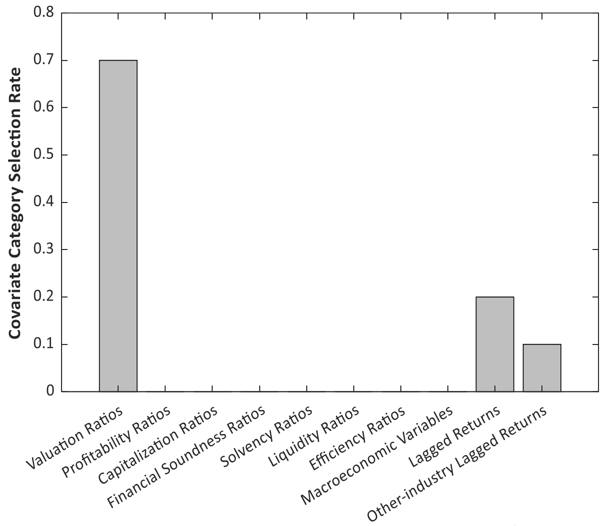


We arrive at three significant conclusions. First, the valuation ratios, followed by the individual industry's lagged returns and the cross-industry one-month lagged returns, are the most pivotal categories for the model's predictive performance. Second, this result is generally consistent across the different industries. Third, we witness greater variability and less consistency regarding the participation of other covariate categories in the three highest SAGE value positions. The above results are consistent with a body of literature that examines the predictive relationship between valuation ratios and asset returns. Notable examples include the work of Keim and Stambaugh (1986), Fama and French (1988), and Campbell and Shiller (1988). More recent studies outlining that valuation ratios can effectively predict stock returns include the work of Campbell and Yogo (2006) and Campbell and Thompson (2008). Additionally, our framework reveals significant interdependencies and gradual information diffusion across the industries, given that individual and cross-industry lagged returns are the second and third most crucial covariate categories, respectively. In this direction, Rapach et al. (2015) note that links between industries can be established not just with customer-supplier relationships but even more broadly via technology spillovers and production chain interactions.

We also calculate the selection rate of each variable's category based on the highest SAGE values generated across the ten industries examined and rolling windows. To define a category's selection rate, we consider the appearance of its corresponding covariates within the top three highest SAGE values OOS across the ten industries and window estimations. Then, for each category, we calculate the fraction of the covariates with the highest SAGE values across all industries. Figure 3 presents the estimated selection rates for the different covariates' categories.

Figure 3. Selection rates for the covariates' categories.

The figure displays the selection rate that each category's covariates appear within the three highest positions regarding their corresponding SAGE values across the 2010 – 2019 OOS period and ten industries.



Consistent with the findings of Figure 2, we evidence that the valuation ratios have the highest aggregate presence, as indicated by a selection rate of up to 70%. In second and third place come each industry's lagged returns and *other-industry* lagged returns with selection rates of around 20% and 10%, respectively. While these rates are lower than the valuation ratios, they still report the importance of each industry's lagged and cross-industry lagged returns.

5.2.1 Statistical Significance of SAGE Values

According to the SAGE value of a specific variable, we can measure its positive contribution to the overall LassoNet performance. By summing all variables belonging to a specific category across industries and forecast windows, we can obtain an aggregate measure of a category's overall positive contribution to the model's performance. We employ pairwise hypothesis tests between the covariates' categories to evaluate any differences in positive contribution OOS statistically. We effectively create a set of 10 category aggregate SAGE values. Finally, we employ pairwise two-tailed *t*-tests between the aggregate SAGE of covariate categories. In Table 7, we report the *t*-statistics and their corresponding *p*-values of the cross-category hypothesis tests and the corresponding *p*-values controlling for *p-value correction* under the Hommel (1988) criterion, which are reported in brackets. The pairwise tests compare the mean of the column covariate against the mean of every other covariate category presented in each row. Hence, a positive *t*-statistic with a low *p*-value indicates that the column category has a statistically significant higher positive contribution than the row category.

The findings presented in Table 7 quantitatively validate the results of the categories' selection rates, as presented in Figure 2. For example, they report that the mean of the valuation ratios against the mean of every other covariate category is positive and statistically significant at the 1% level for almost all cases. In addition, the *t*-statistics and their *p*-values also reveal a favourable and statistically significant outcome for the mean positive contribution of the lagged returns category compared to most of the covariate categories at 1%, except to those *of other-industry* lagged returns and valuation ratios. The results for the *other-industry* lagged returns are similar to the lagged returns category and are at a statistical significance level of 1%. Also, macroeconomic variables categories and financial soundness show positive significance against specific covariates such as solvency, capitalization, liquidity and efficiency ratios.

Controlling for *p-value correction* under the Hommel (1988) criterion, the corresponding *p*-values in brackets reveal a similar picture mainly for valuation ratios, lagged and *other-industry* lagged returns and financial soundness ratios, which retain their positive and statistical significance against the rest of the categories. The above findings can help quantitative fund managers experiment beyond conventional statistical models and effectively guide decisions concerning asset allocations while attaining model transparency via the SAGE. Policymakers can also benefit from such an interpretable learning framework when designing economic policies by forecasting the movements of industry sectors and identifying the most critical covariates governing the underlying price dynamics.

Table 7. SAGE values pairwise hypothesis tests between the covariates' categories

We sum the SAGE values of all covariates belonging to the same category for each of the ten industries over the 2010 – 2019 OOS period and the forecasting windows examined. For all covariate categories, we create a set of aggregate SAGE values. We then employ pairwise *t*-tests between the covariates' categories to evaluate differences in the positive contribution statistically. We present the *t-statistics* and the corresponding *p*-values for the hypothesis tests in parenthesis. We additionally report the corresponding *p*-values under the Hommel (1988) criterion for *p*-value correction in brackets. The *t*-statistics and *p*-values of the column category with a statistically significant higher positive contribution than row one are presented in bold. *, **, *** denote significance at the 10%, 5% and 1% level, respectively.

	Valuation Ratios	Lagged Returns	Other-industry Lagged Returns	Macroeconomic Variables	Financial Soundness Ratios	Solvency Ratios	Profitability Ratios	Capitalization Ratios	Liquidity Ratios	Efficiency Ratios
Valuation Ratios	-	-	-	-	-	-	-	-	-	-
Lagged Returns	**1.833 (0.086*) [0.086*]**	-	-	-	-	-	-	-	-	-
Other-industry Lagged Returns	**3.118 (0.009***) [0.019**]**	1.549 (0.143) [0.143]	-	-	-	-	-	-	-	-
Macroeconomic Variables	**4.679 (0.001***) [0.003***]**	**3.873 (0.003***) [0.006***]**	**3.454 (0.005***) [0.005***]**	-	-	-	-	-	-	-
Financial Soundness Ratios	**4.718 (0.001***) [0.003***]**	**3.944 (0.003***) [0.006***]**	**3.611 (0.005***) [0.005***]**	0.096 (0.925) [0.925]	-	-	-	-	-	-
Solvency Rations	**5.092 (0.001***) [0.002***]**	**4.534 (0.001***) [0.004***]**	**4.642 (0.001***) [0.005***]**	**2.718 (0.023**) [0.090*]**	**3.906 (0.003***) [0.009***]**	-	-	-	-	-
Profitability Ratios	**4.859 (0.001***) [0.003***]**	**4.162 (0.002***) [0.005***]**	**3.974 (0.003***) [0.005***]**	0.973 (0.346) [0.691]	1.130 (0.273) [0.273]	**-2.326 (0.042**) [0.167]**	-	-	-	-
Capitalization Ratios	**5.082 (0.001***) [0.002***]**	**4.517 (0.001***) [0.004***]**	**4.613 (0.001***) [0.005***]**	**2.631 (0.026**) [0.102]**	**3.749 (0.003***) [0.010**]**	-0.269 (0.791) [0.791]	**2.192 (0.051*) [0.109]**	-	-	-
Liquidity Ratios	**5.066 (0.001***)**	**4.491 (0.001***)**	**4.564 (0.001***)**	**2.483 (0.032**)**	**3.460**	-0.544 (0.594)	**1.963 (0.073*)**	-0.308 (0.762)	-	-

		[0.002*]**	**[0.004***]**	**[0.005***]**	**[0.10**2]	**(0.005***) [0.014**]**	[0.791]	**[0.143]**	[0.762]		
	Efficiency Ratios	**5.027 (0.001***) [0.002***]**	**4.431 (0.002***) [0.005***]**	**4.463 (0.002***) [0.005***]**	**2.211 (0.051*) [0.152]**	**3.076 (0.009***) [0.019**]**	-1.483 (0.158) [0.475]	1.571 (0.143) [0.143]	-1.184 (0.253) [0.505]	-0.752 (0.462) [0.462]	-

5.3 Trading application

To adequately assess the economic significance of the LassoNet's predictive ability, we apply a trading simulation to use the OOS industry returns' forecasts to form spread portfolios on their corresponding industry ETFs. We construct monthly spread portfolios of industry ETFs based on the best and worst-performing industries according to their one-month ahead forecasts. For example, we sort the industries each month based on the corresponding LassoNet's return predictions. Then, we form long positions on ETFs for the industries with the highest forecasted returns and short positions for those with the lowest returns. We consider three variations of long-short portfolios. First, the *Max1-Min1* which considers the spread of only the top and bottom forecasted industry returns. Second, the *Max2-Min2* is the return of being long (short) on the two top (bottom) industry portfolios with the highest (lowest) predicted returns and likewise, *Max3-Min3* is the spread of the top and bottom three portfolios, respectively. We consider the effective half spreads of the ETFs as transaction costs.[14] [15]

Table 8 presents a battery of performance metrics for each portfolio and several benchmarks. We report the annualized mean return, volatility, annualized Sharpe ratio, maximum drawdown and the annualized alphas of 4-factor (Carhart, 1997) and 5-factor (Fama & French, 2015) models. We also regress the portfolios' returns against the four-factor (Carhart, 1997) and five-factor models (Fama & French, 2015), and explore the presence of positive and statistically significant alphas. We choose to compare the performance of ETF portfolios based on LassoNet forecasts against the corresponding portfolios constructed based on Group Lasso, Elastic Net and XGBoost, which are the following best performers after LassoNet in terms of statistical accuracy. We also use a buy-and-hold strategy on the value-weighted returns of the CRSP index and the S&P 500 index, equally weighted portfolios of the industry portfolio returns, equally weighted portfolios of the selected industry ETFs, and a strategy that shorts the ETFs' returns in the current month to trade their spread on the following as our benchmarks. The choice of the CRSP index as the most representative benchmark aligns with earlier literature, given the diversification properties of the market return, which is free from anomalies specific to individual industries (see Moskowitz and Grinblatt, 1999; Dong et al., 2022).

[14] We present the corresponding results with the realized spreads as transaction costs in Online Appendix E, and we find that the overall picture remains the same. We also present the performance of a trading strategy investing directly in industry portfolios.

[15] We also perform a similar trading exercise on the forecasts of the maximum number of industry portfolios (i.e., 49) available on Kenneth French's website. Since not enough ETFs are available to track the expanded number of industry portfolios (i.e., 49 industries) we implement our strategy directly on industry portfolios. The findings show that LassoNet yields the highest performance and are available in Online Appendix F.

Table 8. Performance of industry ETF portfolios based on OOS forecasts.

The table demonstrates performance metrics for trading strategies of industry ETFs based on LassoNet's, Group Lasso, Elastic Net and XGBoost forecasts and those of benchmark strategies over the 2010 – 2019 OOS period. The *Max1-Min1*, *Max2-Min2*, and *Max3-Min3* industry ETFs spread portfolios are constructed based on the highest and lowest-performing industries according to their corresponding forecasts while considering the effective half spread of each ETF as transaction cost. We report the annualized mean return and Sharpe ratio, maximum drawdown and annualized alphas. The reported alphas are obtained using the 4-factor (Carhart, 1997) and 5-factor (Fama & French, 2015) models. *, **, *** denote significance at the 10%, 5% and 1% level, respectively.

	Portfolios				**Benchmark Strategies**				
***Panel A :* Max1-Min1**	**LASSONET**	**GROUP LASSO**	**ELASTIC NET**	**XGBOOST**	**VW-CRSP**	**SP500**	**EW-Ind. Port**	**EW-Ind. ETFs**	**1 MAH**
Ann. Mean Return (%)	20.93	19.13	20.09	5.39	12.32	11.46	9.24	11.22	--7.94
Volatility (%)	14.80	15.12	14.70	12.54	12.72	12.46	12.15	12.15	16.55
Ann. Sharpe ratio	1.38	1.23	1.33	0.39	0.93	0.88	0.71	0.88	-0.51
Max Drawdown (%)	20.58	19.71	16.49	33.77	18.52	17.04	17.69	17.94	109.91
Ann. 4-factor alpha (%)	20.62***	18.08***	19.72***	2.61	-	-	8.31***	9.78**	-10.40*
Ann. 5-factor alpha (%)	19.19***	15.96***	16.98***	1.43	-	-	8.39***	9.20**	-11.17**
Panel B: Max2-Min2	**LASSONET**	**GROUP LASSO**	**ELASTIC NET**	**XGBOOST**	**VW-CRSP**	**SP500**	**EW-Ind. Port**	**EW-Ind. ETFs**	**1 MAH**
Ann. Mean Return (%)	22.23	17.70	17.06	3.31	12.32	11.46	9.24	11.22	-2.25
Volatility (%)	10.22	10.41	10.31	9.99	12.72	12.46	12.15	12.15	12.82
Ann. Sharpe ratio	2.13	1.65	1.61	0.28	0.93	0.88	0.71	0.88	-0.22
Max Drawdown (%)	6.88	8.08	8.99	18.21	18.52	17.04	17.69	17.94	54.31
Ann. 4-factor alpha (%)	21.11***	16.27***	17.51***	1.91	-	-	8.31***	9.78**	-3.79
Ann. 5-factor alpha (%)	20.90***	14.80***	15.68***	1.31	-	-	8.39***	9.20**	-4.63
Panel C: Max3-Min3	**LASSONET**	**GROUP LASSO**	**ELASTIC NET**	**XGBOOST**	**VW-CRSP**	**SP500**	**EW-Ind. Port**	**EW-Ind. ETFs**	**1 MAH**
Ann. Mean Return (%)	14.03	13.46	12.44	1.42	12.32	11.46	9.24	11.22	-3.97
Volatility (%)	6.19	8.38	8.07	6.52	12.72	12.46	12.15	12.15	10.37
Ann. Sharpe ratio	2.18	1.56	1.47	0.14	0.93	0.88	0.71	0.88	-0.43
Max Drawdown (%)	4.56	6.59	10.28	24.61	18.52	17.04	17.69	17.94	65.28
Ann. 4-factor alpha (%)	13.55***	12.79***	12.01***	-0.68	-	-	8.31***	9.78**	-4.78
Ann. 5-factor alpha (%)	13.01***	11.64***	10.86***	-1.01	-	-	8.39***	9.20**	-5.47*

The ETF portfolios based on LassoNet forecasts generate the highest returns, Sharpe ratios, and positive and statistically significant alphas compared to the rest of the ML models and benchmarks. In particular, the *Max2-Min2* portfolio achieves the highest performance, generating a Sharpe ratio of 2.13, followed by the *Max1-Min1* portfolio. The same portfolio yields a statistically significant annualized alpha of 22.23%. Similarly, the maximum drawdown of the LassoNet constructed portfolios is the lowest most of the time. These findings demonstrate that LassoNet's forecasts can effectively generate positive ETF returns not captured by seminal factor models while minimizing the downside risk.

6. Conclusion

We apply an interpretable ML framework, the LassoNet, to forecast U.S. industry portfolio returns over the 2010 – 2019 period based on a data-rich environment of 88 predictors. We compare the performance of LassoNet with that of a battery of linear (i.e., linear regression, Group Lasso, Elastic-Net) and nonlinear (i.e., XGBoost, neural networks) methods. We quantify the critical determinants of our forecasts by applying the SAGE on features selected by LassoNet. Finally, we evaluate the economic significance of industry portfolio returns forecasts in a capital allocation strategy.

We find that LassoNet can capture nonlinear patterns and interactions among our predictors to forecast better industry portfolio returns than linear and nonlinear ML approaches. LassoNet reports significantly smaller forecasting errors across most industries examined than other ML models, such as the XGBoost and neural networks, as shown by a batter of statistical tests such as the D.M., GR, SPA and MCS tests. Such a performance lies in the specific characteristic of LassoNet, which is to perform feature selection and deep learning for forecasting purposes simultaneously. Additionally, we demonstrate the economic significance of LassoNet forecasts by constructing profitable spread portfolios. All LassoNet-constructed portfolios generate high Sharpe ratios and positive and statistically significant multifactor alphas. Hence, we expand previous studies reporting the ability of Lasso methods to accurately predict industry portfolios' returns by successfully applying a Lasso-based deep learning method for nonlinear environments. We also note that valuation ratios and individual and cross-industry lagged industry returns are critical determinants for industry portfolio forecasts by generating the highest SAGE values. Such evidence complements the relevant findings of studies using linear asset pricing for return predictability, which mainly reveal profitability and liquidity ratios as essential factors of stock returns. Our study goes forward on the application of interpretable deep-learning machine learning approaches, and their merits in investment problems.

References

Avramov, D. (2004). Stock Return Predictability and Asset Pricing Models. *Review of Financial Studies,* 17 (3), 699–738.

Ball, R., Gerakos, J., Linnainmaa, J. T., and Nikolaev, V. (2016). Accruals, cash flows, and operating profitability in the cross-section of stock returns. *Journal of Financial Economics*, 121(1), 28-45.

Bianchi, D., & McAlinn, K. (2021). Divide and Conquer: Financial Ratios and Industry Returns Predictability. *SSRN Electronic Journal*.

Brigo, D., Huang, X., Pallavicini, A., & De Oscariz-Borde, H.,S. (2021). Interpretability in deep learning for finance: a case study for the Heston model. Working paper, SSRN.

Campbell, J. Y., & Shiller, R. J. (1988). Stock prices, earnings, and expected dividends. the Journal of Finance, 43(3), 661-676.

Campbell, J. Y., & Yogo, M. (2006). Efficient tests of stock return predictability. Journal of Financial Economics, 81(1), 27–60.

Campbell, J. Y., & Thompson, S. B. (2008). Predicting excess stock returns out of sample: Can anything beat the historical average? The Review of Financial Studies, 21(4), 1509–1531.

Carhart, M. M. (1997). On Persistence in Mutual Fund Performance. *The Journal of Finance*, *52*(1), 57–82.

Chen, L., Pelger, M., Zhu, J., (2023). Deep Learning in Asset Pricing. Management Science, articles in advance.

Cohen, L. & Frazzini, A. (2008). Economic links and predictable returns. *Journal of Finance,* 63, 1977–2011.

Covert, I., Lundberg, S. M., & Lee, S. I. (2020). Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, *33*, 17212-17223.

Covert, I. C., & Lee, S. I. (2021). Improving KernelSHAP: Practical Shapley Value Estimation via Linear Regression. arXiv preprint arXiv: http://arxiv.org/abs/2012.01536

Delen, D. , Sharda, R. , & Kumar, P. (2007). Movie forecast guru: A web-based dss for hollywood managers. *Decision Support Systems*, 43 (4), 1151–1170.

DeMiguel, V., Gil-Bazo, J., J. Nogales, F., & Santos, A.A.P. (2023). Machine learning and fund characteristics help to select mutual funds with positive alpha. *Journal of Financial Economics*

150 (3).

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

Dimopoulos, Y., P. Bourret, & S. Lek. (1995). Use of some sensitivity criteria for choosing networks with good generalization ability. *Neural Processing Letters* (2), 1–4.

Dong, X., & Li, Y.,. Rapach, D., E., & Guofu Zhou (2022). "Anomalies and the Expected Market Return," *Journal of Finance*, 77 (1), 639-681.

Dunis, C. L., Laws, J., & Sermpinis, G. (2011). Higher order and recurrent neural architectures for trading the EUR/USD exchange rate. *Quantitative Finance*, *11*(4), 615–629.

Fama, E. F., & French, K. R. (1988). Dividend yields and expected stock returns. Journal of Financial Economics, 22(1), 3–25.

Fama, E. F., & French, K. R. (2015). A five-factor asset pricing model. *Journal of Financial Economics*, *116*(1), 1–22.

Ferson, W. E. & Harvey, C. R. (1991). The Variation of Economic Risk Premiums. *Journal of Political Economy,* 99 (2), 385–415.

Ferson, W. E. & Korajczyk, R. A. (1995). Do arbitrage pricing models explain the predictability of stock returns? *Journal of Business*, pages 309-349.

Ferson, W. E. & Harvey, C. R. (1999). Conditioning Variables and the Cross Section of Stock Returns. *Journal of Finance*, 54 (4), 1325–1360.

Filippou, I., Rapach, D., Taylor, M. P., & Zhou, G. (2022). Out-of-Sample Exchange Rate Prediction: A Machine Learning Perspective. *Available at SSRN*.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, *270*(2), 654–669.

Giacomini, R., & White, H. (2006). Tests of conditional predictive ability. *Econometrica, 74*(6), 1545-1578

Granger, C. W. (1993). Strategies for modelling nonlinear time-series relationships. *Economic Record*, *69*(3), 233–238.

Grønborg, S.,N., Lunde, A., Timmermann, A., & Wermers, R. (2021). Picking funds with confidence. Journal of Financial Economics, 139 (1) 1-28.

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

Hansen, P. R. (2005). A Test for Superior Predictive Ability. *Journal of Business & Economic Statistics*, *23*(4), 365–380.

Hansen, P. R., Lunde, A., & Nason, J. M. (2011). The model confidence set. *Econometrica*, *79*(2), 453-497.

Harvey, C., R. & Liu, Y. (2015). Backtesting. *SSRN Working Paper*

Hommel, G. (1988). A stagewise rejective multiple test procedure based on a modified Bonferroni test. Biometrika, 75(2), 383-386.

Huang, J., Z., & Shi Z. (2022). Machine-Learning-Based Return Predictors and the Spanning Controversy in Macro-Finance. Management Science 69, 1780-1804.

Keim, D. B., & Stambaugh, R. F. (1986). Predicting returns in the stock and bond markets. Journal of Financial Economics, 17(2), 357-390.

Kim, A., Yang, Y., Lessmann, S., Ma, T., Sung, M., C., & Johnson J., E., V. (2020). Can deep learning predict risky retail investors? A case study in financial risk behaviour forecasting. *European Journal of Operational Research, 283 (1), 217-234.*

Kim, B., Khanna, R., & Koyejo, O. O. (2016). Examples are not enough, learn to criticize! criticism for interpretability. *In Advances in neural information processing systems*, pages 2280–2288.

Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, *259*(2), 689–702.

Lemhadri, I., Ruan, F., Abraham, L., & Tibshirani, R. (2021). Lassonet: A neural network with feature sparsity. *Journal of Machine Learning Research*, *22*, 1–29.

Lewellen, J., Nagel, S., & Shanken, J. (2010). A skeptical appraisal of asset pricing tests. Journal of Financial Economics, 96 (2), 175-194.

Lundberg, S.M., Lee, S.-I., (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 1–10.

Menzly, L. & O. Ozbas, O. (2010). Market segmentation and the cross-predictability of returns. *Journal of Finance,* 65, 1555–1580.

Mhaskar, H., Liao, Q., & Poggio, T. (2016). Learning Functions: When Is Deep Better Than Shallow.

arXiv preprint arXiv: http://arxiv.org/abs/1603.00988.

Moskowitz and Grinblatt (1999). "Do industries explain momentum?" Journal of Finance, 54(4) 1249-1290.

Pastor, L. & Stambaugh, R. F. (2003). Liquidity risk and expected stock returns. *Journal of Political Economy*, 111(3), 642–685.

Psaradellis, I., & Sermpinis, G. (2016). Modelling and trading the U.S. implied volatility indices. Evidence from the VIX, VXN and VXD indices. *International Journal of Forecasting*, *32*(4), 1268-1283.

Rapach, D. E., Strauss, J. K., Tu, J., & Zhou, G. (2015). Industry Interdependencies and Cross-Industry Return Predictability. *Research Collection Lee Kong Chian School Of Business*.

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

Zou, H. (2006). The adaptive LASSO and its oracle properties. *Journal of the American Statistical Association 101, 1418–1429*.