Interpretable machine learning for empirical asset pricing

Micha Ianniello
Wirtschaftsinformatik (M.Sc.) SPO 2019
Informationsdienste und elektronische Märkte
Karlsruher Institut für Technologie (KIT)
micha.ianniello@student.kit.edu

Abstract

Machine learning techniques have shown superior predictive power in the area of empirical asset pricing over classical statistical techniques such as linear regression (Gu et al., 2020, 2021). In high-stakes decision making as in finance, it is crucial to understand the inner-workings of ML models, to foster trust and transparency (Gilpin et al., 2018; Linardatos et al., 2020). This seminar paper studies the application of Interpretable Machine Learning (IML) techniques to empirical asset pricing. More specifically, the goal is to investigate whether IML can successfully reveal complex relationships between various predictors and stock returns. The experiments show that IML methods reveal non-trivial patterns in asset pricing data which partially coincide with findings from the literature. The study finds that individual and pairwise profitability-related variables are most important for predicting stock returns.

1 Introduction

It is widely believed that there is an accuracy-interpretability trade-off in Machine Learning (ML). This could be a reason why so-called black-box models, such as deep neural networks or gradient boosted trees are widely preferred over intrinsically interpretable models across domains. But in so-called high stakes decisions, decision makers want to know and might even be required to understand what an ML models' predictions are based on. Rudin (2019) even argues that a lack of interpretability can have serious ramifications.

The author states that while the accuracy-interpretability tradeoff is often assumed to be true, a sophisticated black-box model is not always necessary for having top predictive performance. Often this was not true, especially with structured data which can be well represented by a set of naturally meaningful features. A recent study by Zschech et al. (2022) which compares state-of-the art ML models against a set of intrinsically interpretable ML models on twelve different data sets provides proof that there is no strict accuracy-interpretability trade-off in ML. Maybe, it is time for the ML community to shift its mindset towards accuracy and interpretability. This paper aims to demonstrate the use case of interpretable ML for asset pricing.

There are several reasons why the interpretability of ML models is important. First, regulations like the European General Data Protection Regulation (GDPR) which entails laws about algorithmic decision-making and a "right to explanation" might require interpretations if an automated decision would affect the user in a severe way (Goodman & Flaxman, 2017). Second, interpretability is a precondition for a wider acceptance of ML and can further the deployment of ML systems (Gilpin et al., 2018). Lastly, models that lack interpretability are hard to trust which is especially important in high-stakes decisions. Linardatos et al. (2020) name "sectors, such as healthcare or self-driving cars, where also moral and fairness issues have naturally arisen" as crucial for interpretable

ML. Also, the finance domain can be considered a high-stakes scenario since investment decisions that are made based on ML models might have severe financial consequences for investors and managers. Therefore, this paper studies the application of interpretable ML techniques in Finance, more specifically in the area of empirical asset pricing.

The remainder of this work is structured as follows. In Section 2, a literature review is conducted, discussing the current state of research and related work in interpretable ML and empirical asset pricing. Section 3 introduces the interpretable ML techniques that are used in data analysis. The data set used for the experiments is presented in Section 4 before elaborating on the empirical results of the study in section 5. The paper is finally concluded by a discussion in section 6.

2 Literature review

The application of Machine Learning to the area of Finance and asset pricing more specific has been studied by many researchers (e.g. Freyberger et al. (2020), Gu et al. (2020) and Gu et al. (2021)). However, not many have studied yet the value of using interpretable machine learning (IML) techniques in Finance applications. This section aims to give an overview of the aforementioned fields of research.

2.1 Interpretable Machine Learning

There exist different terms in the realm of interpretability concerning ML which are sometimes used interchangeably, namely interpretability, explainability, intelligibility, and transparency. Since researchers use these terms differently, this raises the necessity to understand what exactly is meant whenever such a term is used. For example, Rudin (2019) and Zschech et al. (2022) use the terms explainability versus interpretability for what Murdoch et al. (2019) call post-hoc interpretations versus model-based interpretations. For the sake of simplicity, I will refer to these concepts by using the terms post-hoc interpretability and intrinsic interpretability following Linardatos et al. (2020).

Post-hoc interpretability techniques refer to procedures of creating insights into a black-box model based on the outputs it created. The issue is that for new instances this model could still create unexpected outputs (Rudin, 2019; Zschech et al., 2022). This shows the need for instrinsic interpretability which is about creating models that are interpretable in the first place. Yang et al. (2020) state that interpretability can be achieved by introducing model constraints such as additivity, sparsity, orthogonality, and smoothness.

Murdoch et al. (2019) define interpretability as the process of extracting information from data. Moreover, it can be viewed as the description of the "internals of a system in a way that is understandable to humans" (Gilpin et al., 2018). Finally, the goal of interpretable ML were to facilitate decision-making as it helps build trust in ML systems.

As the interpretability of ML models increasingly receives attention the research area called explainable AI (XAI) focuses on methods "that produce explainable models that, when combined with effective explanation techniques, enable end users to understand, appropriately trust, and effectively manage the emerging generation of AI systems" (Gunning & Aha, 2019). Other researchers prefer to use the term interpretable ML, e.g., Murdoch et al. (2019). Even though the terms refer to different concepts, they are commonly confused in practice possibly because they have similar goals. This paper, however, studies the application of interpretable ML (IML).

Interpretations can be differentiated according to different criteria which are, unfortunately, not used consistently by researchers. Linardatos et al. (2020) provide a good taxonomy for IML techniques (see Figure 1). The first criteria considers the level of the interpretations. Local interpretations explain single predictions while global interpreta-

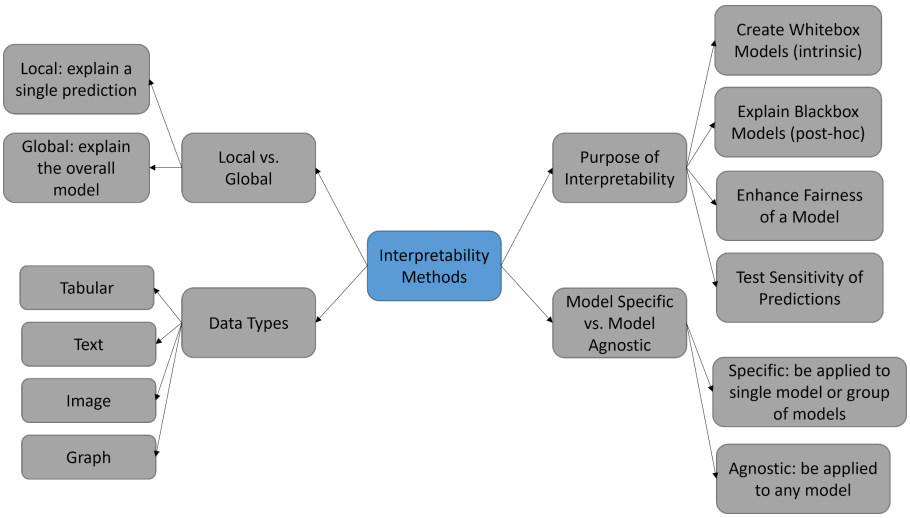

Figure 1: Figure 1: Taxonomy of interpretable ML methods adopted by Linardatos et al. (2020)

tions are able to explain the overall model behavior. Some techniques provide only local interpretations while others can provide both, local and global interpretations. Second, interpretation techniques can be differentiated by the supported data types. Some techniques can handle any data type, while others merely support a subset of data types. The third criteria is the purpose of interpretability. Is it to create a so-called white-box model, which is intrinsically interpretable? Does the technique aim to explain a black-box model after training and prediction (post-hoc)? Last, interpretability methods can be differentiated by whether they can be only applied to a specific set of models (model specific), e.g. tree-based models, or neural networks or whether they are model agnostic, i.e. they can be applied to any model.

Some well-known post-hoc IML techniques are Shapley additive explanations (SHAP) (Lundberg & Lee, 2017) which is based on Shapley values which attribute a contribution value to every feature to the target variable and Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016) which builds a simpler surrogate model to explain local predictions. Linardatos et al. (2020) provide a comprehensive overview of available XAI techniques. As this paper focuses on using intrinsic IML techniques, some of them are described in more detail in section 3.

2.2 Machine Learning in Finance

In recent years, ML techniques have been increasingly addressed by Finance researchers. Goodell et al. (2021) conducted an analysis of 283 scientific articles in the field of Artificial Intelligence (AI) and ML in Finance and show that the number of publications in this field has been monotonically increasing between the years 2014 and 2020. By applying a co-citation and bibliometric coupling analysis on the studied articles, the authors identify three clusters of topics that apply ML and AI in Finance, namely "(1) portfolio construction, valuation, and investor behavior; (2) financial fraud and distress; and (3) sentiment inference, forecasting, and planning".

The application of ML in asset pricing, which belongs to the first of these clusters has been studied by various researchers to date. Gu et al. (2020) study the use of different ML techniques in empirical asset pricing. They find that ML can enhance the understanding of asset prices. Another findings is that neural networks as well as regression trees performed best, which the authors attribute to their capability to model nonlinear interactions missed

by other simpler models. Interestingly, this study finds to some extent a superiority of shallow neural networks over deep ones which is attributed to the low signal-to-noise-ratio as well as a relatively small database in asset pricing.

Another study by Gu et al. (2021) uses autoencoder neural networks to explicitly model the risk-return tradeoff present in asset pricing. Autoencoders are a special type of neural network where the outputs try to mimic the inputs by learning a lower dimensional representation of the data in the hidden layers. It is a type of unsupervised learning. This approach enables dimension reduction as well as an estimation of nonlinear conditional exposures. The study was able to show the predominance of autoencoders over alternative asset pricing factor models including Fama French factor models (Fama & French, 1993), principal components analysis (PCA), and instrumented PCA (Gu et al., 2020) measured by predictive-R^2 and annualized value-weighted Sharpe ratio.

Some Finance researchers have adopted IML techniques. One of them is by Duan et al. (2021) which uses a generalized linear model with factorization, which they call Factorization Asset Pricing Model (FAPM). The study is based on the same data set as Gu et al. (2020) and aims to predict stock returns by focussing on variable interactions. They find that FAPM performed best on the used data set even compared to more sophisticated neural network-based and tree-based models.

Another study by Jaeger et al. (2021) compares two allocation methods to be applied to a multi-asset future portfolio of 17 markets. The authors use gradient boosted trees to regress "the Calmar ratio spread between the two strategies against features of the bootstrapped datasets." The SHAP method is used to interpret the results of the model which is why this study falls in the category of post-hoc interpretability. By inspecting the Shapley values of the features the authors inspect feature importances as well as individual quantitative contributions of each variable to the model output.

3 Interpretable Machine Learning methods

Some ML techniques are traditionally perceived as creating interpretable models due to their simple structure. For example, the simple linear regression model learns one coefficient β_j per variable j which enables one to derive the impact of each variable on the model output. This high degree of interpretability comes at the cost of complexity and performance since linear regression assumes linear relationships between variables only. Another to some degree interpretable technique is a decision tree since its decision leaves can be interpreted as a sequence of if-then rules.

One promising group of algorithms which achieve both model accuracy and intrinsic interpretability are generalized additive models (GAMs), first introduced by Hastie and Tibshirani (1987) where a variety of extensions exist. A GAM is represented by a function g that sums up so-called shape functions f_j plus an intercept term β_0 which are learned respectively. Thus, a GAM is of the form

$$g(E[y]) = \beta_0 + \sum_j f_j(x_j) \tag{1}$$

Since the shape function, f_j can map arbitrary values of x to y, GAMs can learn nonlinear patterns. The functions f_j were originally learned using a so-called local scoring procedure which results in smooth estimates for the f_j. For further details on how this procedure works, it is recommended to read the work of Hastie and Tibshirani (1987). More recent approaches use bagged and boosted trees (Lou et al., 2012; Lou et al., 2013; Nori et al., 2019) as well as neural networks (Agarwal et al., 2021; Yang et al., 2020, 2021). Agarwal et al. (2021) introduced a GAM that uses neural networks to learn shape functions. This approach is called the neural additive model (NAM), where a number of extensions exist (see e.g. section 3.2).

As the originally introduced GAM only modeled univariate effects of variables, models with no such restrictions such as random forests and gradient-boosted trees show higher performance than GAMs (Lou et al., 2012). To mitigate this shortcoming, an approach to include multivariate variable effects in GAMs was introduced by Lou et al. (2013). This results in the so-called GA^{2}M which has the form

$$g(E[y]) = \beta_0 + \sum_j f_j(x_j) + \sum_{i,j} f_{ij}(x_i, x_j) \tag{2}$$

where f_{ij} are bivariate interaction terms.

3.1 Explainable Boosting Machine

The Explainable Boosting Machine (EBM) introduced by Nori et al. (2019) is an implementation of the GA^{2}M. Their work includes the release of a python-package called *interpret*. Being a GA^{2}M, EBM learns a function $g(E[y])$ which is the sum of the learned shape functions over the variables and variable interactions plus a bias term β_0. Moreover, EBMs are able to automatically detect pairwise interactions which results in a model as in equation (2). Shape functions in EBM are learned using two ML techniques, namely bagging and gradient boosting. In bagging, which stands for bootstrap aggregation, an ensemble of m learners is trained on m subsets of the population D (so-called bootstrap samples) and aggregated to make a final prediction. In EBM, these learners are shallow decision trees with limited size. Lou et al. (2012) found that the best performing tree-based GAMs had between two and four leaves per tree. According to the authors, in gradient boosting shape functions are learned iteratively. The procedure for a regression task is depicted in algorithm 1. In the first step, the shape function f_j is initialized to zero (line 1). Next, it is looped over M iterations (line 2) and over the set of features n (line 3). Consecutively, the residuals R for each observation are calculated as the difference of the observation x_{ij}, y_i minus the cumulative shape function outputs. Then, a shape function S is learned based on the residuals R (line 5). In the last step, the shape function f_j is increased by adding S.

Algorithm 1 Gradient Boosting for Regression

1: $f_j \leftarrow 0$
2: **for** $m = 1$ to M **do**
3: **for** $j = 1$ to n **do**
4: $R \leftarrow \{x_{ij}, y_i - \sum_k f_k\}_1^N$
5: Learn shape function $S : x_j \rightarrow y$ using R as training data set
6: $f_j \leftarrow f_j + S$
7: **end for**
8: **end for**

The resulting shape functions f_j are step functions that have one constant value f_j per bin/interval. Note, the number of bins in EBMs is a predefined hyperparameter. At prediction time the shape functions are used as lookup tables and the resulting value per variable are summed up for the final prediction. Nori et al. (2019) highlight that the simple operations used for predicting make EBMs very quick at predicting, while being slower in training compared to other state-of-the-art tree-based ML models. EBMs provides both instrinsic interpretability and state-of-the-art performance across data sets as shown by the authors. Intrinsic interpretability is ensured by the fact that each feature's contribution to the model output is learned by a function f_j and can be visualized globally. The python-package *interpret* also provides an explanation dashboard to visualize global and local interpretations as well as feature importance learned by EBMs.

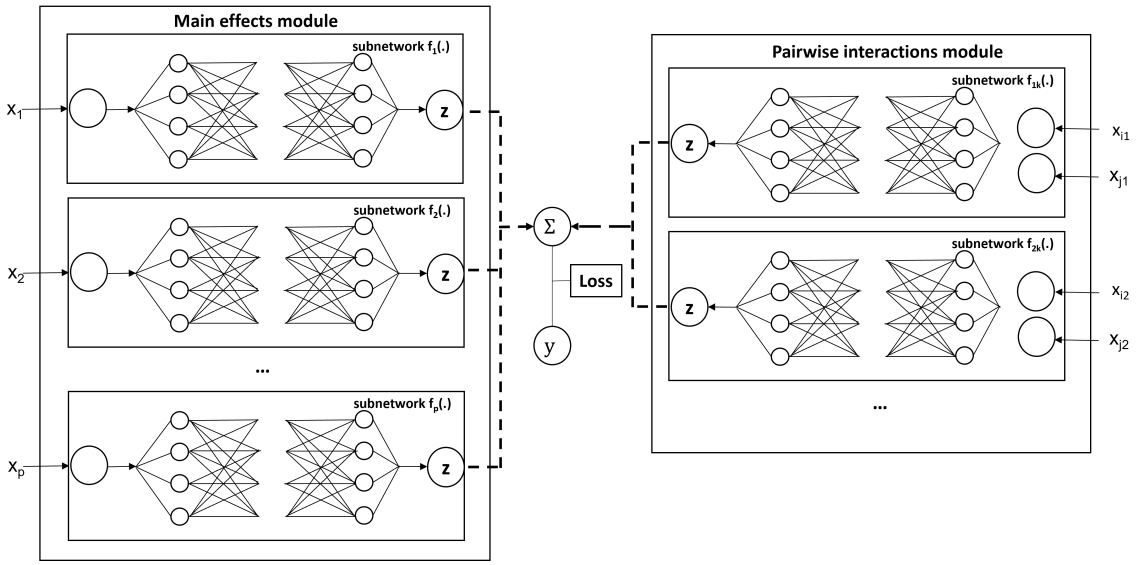

Figure 2: GAMI-Net architecture

3.2 GAMI-Net

GAMI-Net is a neural additive model (NAM) with structured interactions (short: GAMI-Net) introduced by Yang et al. (2021). Conceptually, it follows the idea of NAM which is learning a shape function f_j for each variable j using a neural network. In contrast to NAM, GAMI-Net automatically extracts interaction terms which makes it a GA^{2}M. The model is described as

$$g(E[y|x]) = \beta_0 + \sum_{j \in S_1} f_j(x_j) + \sum_{(j,k) \in S_2} f_{jk}(x_j, x_k). \tag{3}$$

This is very similar to equation (2), except that j and k are from sets S_1 and S_2 respectively. These result from the sparsity constraint of GAMI-Net. Note, that for simplicity the authors focussed on pairwise interaction terms only, while this would be extensible to higher-order interactions as well. Moreover, the authors assume each main effect and pairwise interaction to have zero means. The GAMI-Net architecture consists of two modules, namely the main effects module and the pairwise interaction module as depicted in figure 2. The modules are trained sequentially in three stages: the main effect stage, the pairwise interaction stage, and finetuning. For details of the training procedure please refer the paper by Yang et al. (2021).

GAMI-Net training is subject to three constraints. The sparsity, as well as heredity constraint, serve to improve model interpretability while the marginal clarity constraint ensures the identifiability of main effects and their child pairwise interaction terms. GAMI-Net can be interpreted by inspecting importance ratios and most importantly by plotting the learned shape functions (see section 5.)

The sparsity constraint refers to the selection of main effects as well as pairwise interactions based on an importance measure. After the first two training stages, effects with low importance are disregarded and the remaining effects are kept within S_1 ranked by importance. The importance threshold for effect pruning can be predefined or dynamically determined by GAMI-Net. The importance measure used in GAMI-Net is the importance ratio which is based on sample variance. A more detailed description of this measure is in section 5.1.

The heredity constraint states that for an interaction term jk to be included in the

set of interaction terms S_2, at least one of the parent terms j or k have to be in the set of main effects S_1. This is called weak heredity:

$$\forall (j,k) \in S_2 : j \in S_1 \vee k \in S_1. \tag{4}$$

The Marginal clarity constraint aims to create models that are identifiable, so that their main effects are not absorbed by their child interactions and vice versa. This mitigates unstable training and the creation of unclear model representations. Marginal clarity is achieved by introducing a penalty for non-orthogonality of main effects and pairwise interactions which is to be minimized. Thus the resulting optimization problem is

$$\min_{\theta} \mathcal{L}_\lambda(\theta) = \ell(\theta) + \lambda \sum_{j \in S_1} \sum_{(j,k) \in S_2} \Omega(f_j, f_{jk}), \tag{5}$$

$$s.t. \int f_j(x_j) dF(x_j) = 0, \forall j \in S_1$$

$$\int f_{jk}(x_j, x_k) dF(x_j, x_k) = 0, \forall (j,k) \in S_2,$$

where

$$\Omega(f_j, f_{jk}) = |\frac{1}{n} \sum f_j(x_j) f_{jk}(x_j, x_k)|,$$

i.e. Ω computes the degree of non-orthogonality of effects. The closer Ω is to zero the more clearly separable the main effect f_j is from its child interaction terms f_{jk}.

4 Data description

The data used for the study corresponds to the data used by Freyberger et al. (2020). The authors study the impact of different stock- and balance-sheet-related variables on cross-sectional stock return prediction. The stock data originates from the Center for Research in Security Prices (CRSP) monthly stock file. The stocks included in the data set are from firms listed on NYSE, Amex, and Nasdaq. Additionally, the data set includes balance sheet data from the Standard and Poor's Compustat database. The data includes observations from between July 1962 to May 2014. This includes approximately 1.6 million observations before and 1.3 million observations after data cleaning. The authors group the variables in the data set into six groups, namely past return based variables, investment-related variables, profitability-related variables, intangibles, value-related variables, and trading frictions. An overview and a description of some of the important variables is given in table 2. For an explanation of the variables in detail please refer to the internet appendix of Freyberger et al. (2020).

The boxplot in figure 2 shows the distribution of the target variable which is the *return*. The box represents the inter-quartile range, i.e. the area with the highest density. The figure points out that there are many extreme values that are way beyond the inter-quartile range. Actually, most of the independent variables have this kind of long tail distribution. In the analysis, extreme outliers of returns by the definition of Ince and Porter (2006) were replaced with their median to ensure more balanced training data. Figure 2 also shows that the bulk of data points is around 0. Refer to table 1 for further summary statistics of returns. Note, that returns are provided in monthly frequency and that the target is predicting *return*s one month ahead. In the analysis, *return* is also used as a predictor to predict the next month's *return*, i.e. the target variable.

5 Empirical results

This study applied the IML methods described in section 3 on the data set described in section 4. Various experiments were conducted while this section summarizes the best

Statistic	Mean	SD	Min	Median	Max
Value	0.00	0.17	-0.98	0.00	24.00

Table 1: Summary statistics for variable 'returns'

Group	Abbreviation	Variable Name	Freyberger	EBM	GAMI-Net
Past returns	r_{2-1}	Return 1 month before prediction	x	x	
	r_{6-2}	Return from 6 to 2 months before prediction	x	x	
	r_{12-2}	Return from 12 to 2 months before prediction	x		
	r_{12-7}	Return from 12 to 7 months before prediction	x		
	r_{36-13}	Return from 36 to 13 months before prediction	x		
Investment	investment	Year on year growth rate in Total Assets	x		
	ΔCEQ	% change in book value of Equity			x
	$\Delta PI2A$	Change in PP&E and inventory over lagged AT		x	
	$\Delta Shrout$	% change in shares outstanding	x	x	x
	NOA	Net-operating assets over lagged AT	x		
Profitability	ATO	Sales to lagged net operating assets	x	x	x
	$\Delta(\Delta GM - \Delta Sales)$	$\Delta(\%changeingrossmarginand\%changeinsales)$		x	x
	EPS	Earnings per share		x	x
	IPM	Pre-tax Profit Margin		x	x
	PM	Profit Margin		x	x
	PM_{adj}	Adjusted Profit Margin	x	x	x
	Prof	Profitability		x	
	RNA	Return to net operating assets			
	ROA	Return-on-Assets		x	
	ROC	Return-on-Cash	x	x	x
	ROE	Return-on-Equity		x	x
Intangibles	OA	Operating accruals		x	x
Value	BEME	Book value of equity to market value of equity	x		
	C2D	Cash flow to total liabilities		x	x
	ΔSO	Log change in the split adjusted shares outstanding	x		
	Free CF	Free cash flow to book value of Equity		x	x
	LDP	Trailing 12-months dividends to price	x		
	NOP	Net payouts to Size		x	
	O2P	Operating payouts to market cap		x	x
	S2P	Sales to price	x		
Trading frictions	DTO	De-trended Turnover - market Turnover		x	x
	Idio vol	Idiosyncratic volatility			x
	LME	Size as price times shares outstanding	x		
	Lturnover	Last month's volume to shares outstanding	x		
	Rel to high price	Price to 52 week high price	x		
	Ret max	Maximum daily return	x		
	SUV	Standard unexplained volume	x	x	
	Total vol	Standard deviation of daily returns	x		x

Table 2: Comparison of important variables by group (column 1) and method (columns 4 to 6). Note, that Freyberger refers to variables selected by Freyberger et al. (2020) and EBM and GAMI-Net show the variables with an importance ratio (IR) above one per cent.

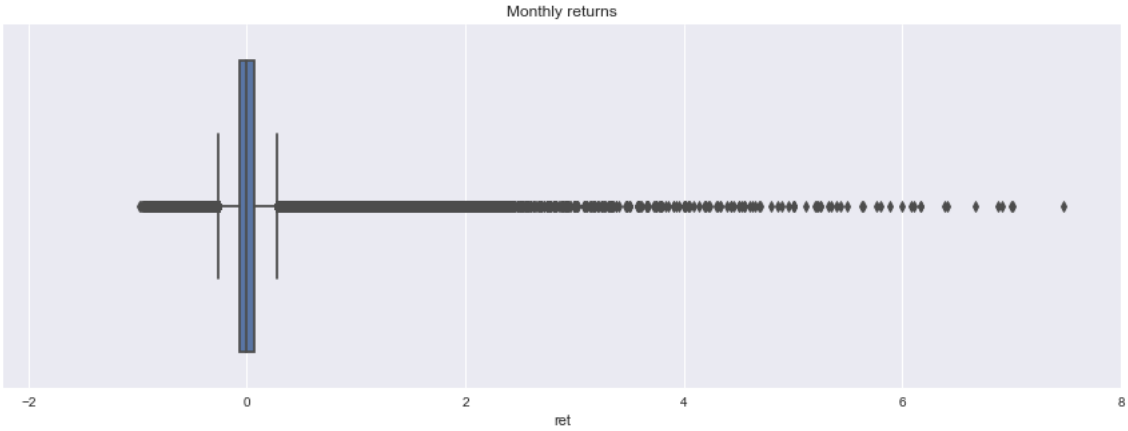

Figure 3: Distribution of the target variable *return*. The blue box is the inter-quartile range and the whiskers are located at 1.5 times the inter-quartile range. The points beyond the whiskers are heuristically considered outliers.

results obtained. This section aims to demonstrate the interpretability of the mentioned methods as well as compare the results to existing literature. First, a comparison of variable importance is made before demonstrating the individual and pairwise effects learned by the methods. Lastly, the out-of-time model performance of the applied methods is discussed.

5.1 Variable importance

Global variable importance further referred to as *importance*, is an interpretability concept also common in post-hoc methods such as SHAP and LIME. In SHAP e.g., importance is a measure of the average contribution of an individual variable to the model output (Lundberg & Lee, 2017). In general, importances help improve interpretability by highlighting how strong of an impact individual variables or interactions have on the prediction. In EBM, importance measures the global average contribution of a variable to the prediction, i.e.

$$Importance_{EBM}(j) = \frac{1}{N} \sum_{n \in N} |f_j(x_j^{(n)}|), \tag{6}$$

where $x_j^{(n)} \in X^{JxN)}$ represents one out of N samples in the data set taken from the j-th variable.

In contrast, GAMI-Net importance is based on sample variance, which is measured by the average sum of squares of the shape function, i.e.

$$Var(f_j) = \frac{1}{N-1} \sum_{n \in N} f_j^2(x_j^{(n)}) \tag{7}$$

for individual effects and

$$Var(f_{jk}) = \frac{1}{N-1} \sum_{n \in N} f_{jk}^2(x_j^{(n)}, x_k^{(n)}) \tag{8}$$

for pairwise effects, where f_j is an individual shape function and f_{jk} are the pairwise shape functions of effects j, k, and N is the sample size. Additionally, an importance

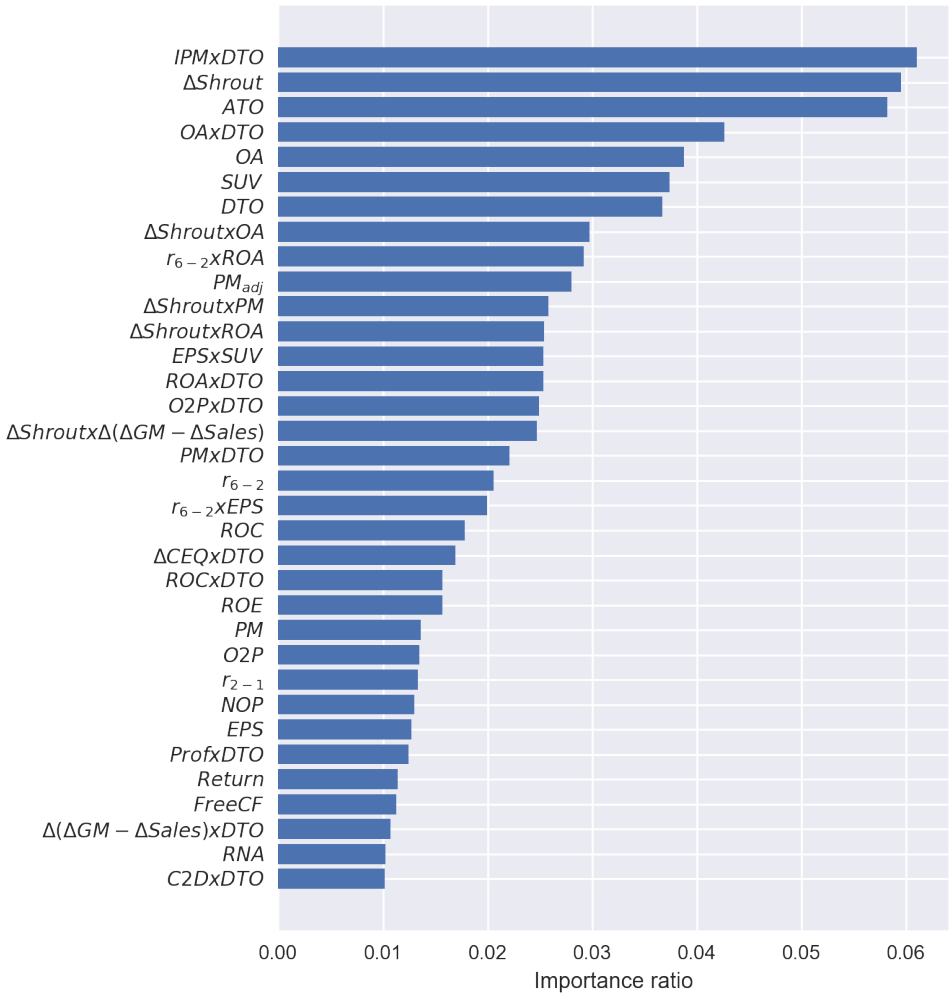

Figure 4: EBM variable importance.

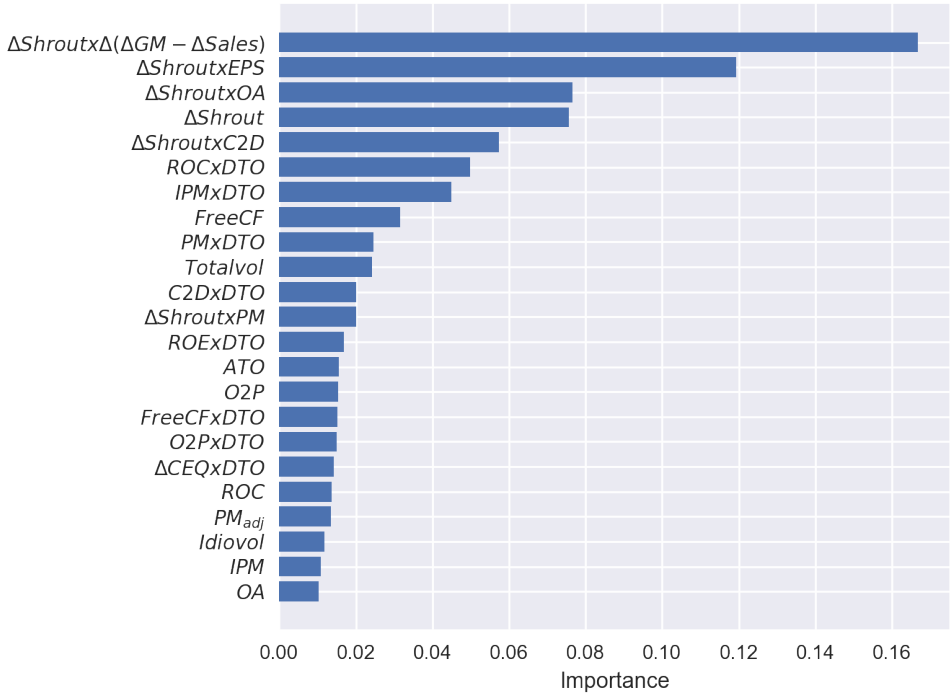

Figure 5: GAMI-Net variable importances.

ratio (IR) is calculated as the relative variance of an effect to the overall variance, defined by

$$IR(j) = Var(f_j)/T \tag{9}$$

for individual effects and

$$IR(j) = Var(f_{jk})/T \tag{10}$$

for pairwise effects where T is the total sample variance of all shape functions $f_j \in \mathrm{S}_1$ and $f_{jk} \in \mathrm{S}_2$;

$$T = \sum_{j \in S_1} D(f_j) + \sum_{j,k \in S_2} D(f_{jk}).$$

In training stages one and two, effects are ranked according to IR. Consecutively, the top-s_1 individual effects and the top-s_2 pairwise effects are selected resulting in feature sparsity (see section 2.2). The higher IR of an effect, the more likely it is to being selected. Note, that in GAMI-Net, IR sums up to one over all effects $j \in \mathrm{S}_1$ and $jk \in \mathrm{S}_2$, while in EBM importance does not. Moreover, EBM importance is greater than zero for every variable and pairwise interaction term while GAMI-Net enforces some variable effects and the corresponding IR to zero. To make the importance measures comparable, IR is also calculated for the EBM shape functions according to equations 9 and 10.

Figures 4 and 5 show the variables ranked most important by EBM and GAMI-Net. As a heuristic approach, only the effects identified with an IR above one percent are presented. Due to the missing variable selection mechanism in EBM, importance differs in magnitude and in relation. Since GAMI-Net uses the sparsity constraint, it identifies a few individual and pairwise effects as very important and considers other effects as trivial pushing IR to zero or close to zero. In total, 63 individual and pairwise effects were selected by GAMI-Net. In EBM, 34 individual and pairwise effects have importance above one percent while with GAMI-Net only 23 individual and pairwise effects exceed this threshold.

Figure 6: Training loss (left) and validation loss (right) trajectory of GAMI-Net training.

Table 2 additionally provides an overview of the heuristically most important variables identified by applied methods (columns 5 and 6) in comparison to the work by Freyberger et al. (2020) (column 4). It can be seen that while all methods agree on only four important variables, namely $\Delta Shrout$, ATO, PM_adj and ROC, there is more agreement between EBM and Freyberger (seven variables in common) as well as between EBM and GAMI-Net which have 14 variables in common. GAMI-Net and Freyberger only agree on five variables according to the heuristic chosen. The strongest agreement between the methods can be seen in the group *Profitability*. One the one hand, the variable selection shows partial disagreement of IML methods with existing literature, on the other hand, the IML methods yield new insights by identifying important pairwise interactions. The validity and generalizability of these effects still have to be verified. One interesting finding is that including pairwise interaction terms significantly improves model performance of GAMI-Net as can be seen in Figure 6. This hints to the fact that some pairwise interactions entail information, otherwise not present in individual variables. The analysis of pairwise interactions could be a promising field of study for researchers and finance practitioners though.

5.2 Individual and pairwise effects

Another GAM property for global interpretabilty is provided by visualizing the learned shape functions to inspect relationships between one or two variables and the model output. A common way of visualizing these functions is by using 1D-line plots and heatmaps for individual and pairwise effects resepectively (as in Caruana et al. (2015), Lou et al. (2013), and Zschech et al. (2022).

As mentioned in section 3, EBMs learn shape functions as a sequence of constant values for a predefined number of bins respectively value intervals. This results in step-like functions. GAMI-Net computes continuous shape functions which can be piecewise linear or more smooth depending on the activation function used. The authors claim that GAMI-Net creates smoother shape functions than EBM that are more robust to noise in the data. When looking at figures 6 and 7, one can notice that GAMI-Net indeed learns smoother functions. This can be observed e.g. for $\Delta Shrout$, ATO and PM_{adj}.

For the pairwise interaction plots, it sticks out that GAMI-Net shape functions are much smoother. This effect is enhanced by the fact that the predefined number of bins used for learning functions f_{jk} in EBM is only 32. The larger the number of bins in EBM the more smoothly appearing shape functions result. While most individual shape functions f_j behave similarly in EBM and GAMI-Net, the pairwise interaction functions f_jk barely show similarities between the two methods. This might be due to the before mentioned hyperparameter or to instabilities in model training. The reliability of pairwise interaction functions identified by GAM models is a potential avenue for future research.

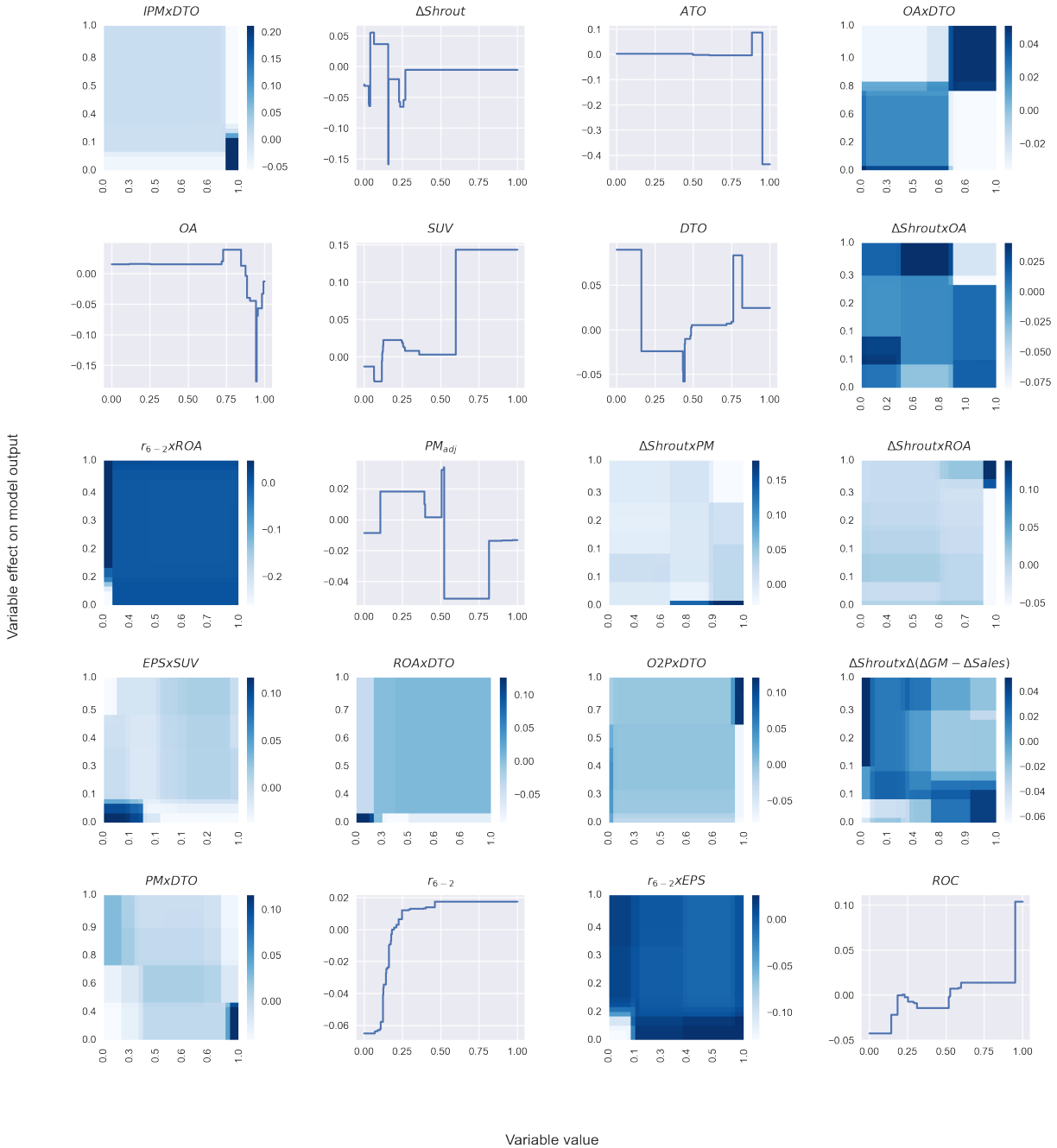

Figure 7: Shape functions learned by EBM. Individual effects are drawn as a line charts, while pairwise effects are shown as heatmaps with the effect values on x- and y-axis and the strength of the effect indicated by hue. Darker blue indicates stronger contribution to the prediction. For the sake of space only the twenty effects ranked most important are shown. Note, that for EBM shape function values are normalized since the python package does not yet support rescaling to the original scale.

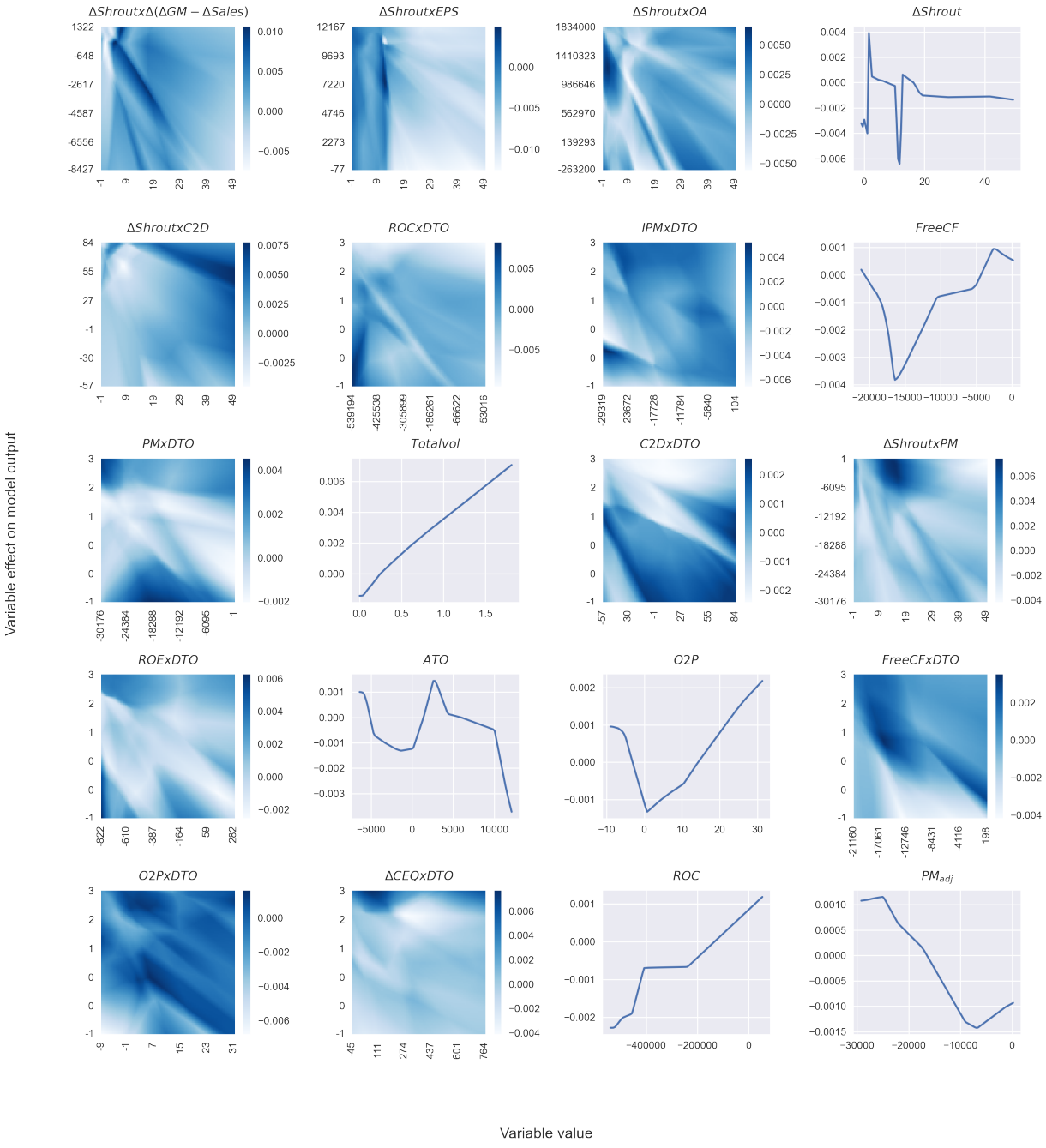

Figure 8: Shape functions learned by GAMI-Net. For the sake of space only the twenty effects ranked most important are shown.

Metric	EBM	GAMI-Net
MAE	**0.0096**	0.0106
RMSE	0.0158	**0.0150**
Training time (min)	**10**	140

Table 3: Out-of-time performance of EBM and GAMI-Net. Lower score means better for MAE and RMSE. Note, that training time reported depends on the hardware used as well as the hyperparameters set.

5.3 Out-of-time performance

The models from section 3 are evaluated on a holdout data set. Out-of-time performance can be measured by different metrics, such as root mean square error (RMSE) and mean absolute error (MAE). Both metrics indicate the average error a model makes, where the former penalizes higher errors more. RMSE is defined as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{i,true} - y_{i,pred})} \tag{11}$$

and MAE as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|(y_{i,true} - y_{i,pred})|. \tag{12}$$

Table 3 reports the out-of-time performance for both EBM and GAMI-Net. When it comes to MAE, EBM scored slightly better, which means its prediction error is smaller on average. Measured by RMSE though, GAMI-Net scores better than EBM. This indicates that GAMI-Net makes less big errors on average which are penalized by RMSE. These findings generally coincide with Yang et al. (2020) and Zschech et al. (2022) who find that the performance differences between EBM and GAMI-Net are marginal and that depending on the task and data set EBM or GAMI-Net may perform better. Concerning computation time, the experiments showed that EBM can be computed in a fraction of the time than GAMI-Net. However, this is strongly dependent on the hardware used, the hyperparameters set, e.g. number of epochs/iterations and also the size of the data set. If time is not a crucial factor GAMI-Net can be considered for achieving smoother shape functions.

6 Discussion

This paper aims to study the applicability of IML methods to finance, more specifically to empirical asset pricing. The methods EBM and GAMI-Net were applied to the asset pricing data set introduced by Freyberger et al. (2020). The experiments conducted showed several strengths and weaknesses of the methods applied.

EBM has shown to be very fast in training even with large data sets like the one studied. Additionally, it is able to learn shape functions that can be visualized to understand the effect of individual and pairwise variable terms. Thus, it can reveal complex relationships like the ones demonstrated in section 5 and help understand what drives stock returns. EBM's predictive performance is comparable to other state-of-the-art methods, like neural networks and XGBoost Zschech et al. (2022) and to the performance of GAMI-Net. Despite the strengths mentioned, EBM has a few shortcomings. The number of interactions it is supposed to find has to be predetermined, which is difficult even for domain experts due to the large number of possible interactions. Additionally, EBM learns shape functions that are piecewise constant. This can result in sudden jumps. In EBM every variable is considered, since it does not entail variable selection.

GAMI-Net provides help to some of EBM's difficulties, namely its interpretability constraints facilitate variable sparsity, marginal clarity and (weak) heredity, which help identify important individual and pairwise effects. Since GAMI-Net learns shape functions by subnetworks it learns smoother shape functions which makes it more robust to noisy data. However, GAMI-Net has many trainable parameters depending on the subnet architectures and the number of variables in a data set. Thus, for high-dimensional and data sets with many observations, training time of GAMI-Net significantly exceeds EBM.

7 Conclusion

ML methods have the potential to reveal complex patterns in data. Thus, ML models have been adopted by finance researchers as discussed in section 1. IML methods, however, have not been studied much yet. This paper shows that IML methods can indeed benefit asset pricing research by uncovering non-trivial individual and pairwise effects on stock returns. The strengths of the methods studied lies in their interpretability and the inclusion of pairwise interaction terms. GAMs and some of its current extensions are introduced as state-of-the-art ML models that are both intrinsically interpretable and accurate. Other groups of IML methods exist e.g. transformer-based models like Arik and Pfister (2021) and Lim et al. (2021) that are able to extract important features as well as temporal patterns. These are well suited for time-series analysis and thus, generally applicable to empirical asset pricing.

Current IML methods focus on structured tabular data, while XAI research addresses unstructured data as well. Since most data in finance is structured, it is a suitable application area are for IML. The creation of IML methods for unstructured data could be an interesting research direction though. Future work can also be done on the issue of variable selection since for high-dimensional data, neural-network-based methods result in many trainable parameters and thus high training cost. Some work has been done in this direction e.g. Lemhadri et al. (2021) and Xu et al. (2022) who combine Lasso regularization and neural networks to achieve feature sparsity. However, these approaches do not consider pairwise interaction terms yet.

References

Agarwal, R., Melnick, L., Frosst, N., Zhang, X., Lengerich, B., Caruana, R., & Hinton, G. E. (2021). Neural additive models: Interpretable machine learning with neural nets. *Advances in Neural Information Processing Systems*, *34*, 4699–4711.

Arik, S. Ö., & Pfister, T. (2021). Tabnet: Attentive interpretable tabular learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(8), 6679–6687.

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015). Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1721–1730.

Duan, Z., Gong, Z., & Qi, Q. (2021). Factorization asset pricing. *Available at SSRN 3940074*.

Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of financial economics*, *33*(1), 3–56.

Freyberger, J., Neuhierl, A., & Weber, M. (2020). Dissecting characteristics nonparametrically. *The Review of Financial Studies*, *33*(5), 2326–2377.

Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining explanations: An overview of interpretability of machine learning. *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, 80–89.

Goodell, J. W., Kumar, S., Lim, W. M., & Pattnaik, D. (2021). Artificial intelligence and machine learning in finance: Identifying foundations, themes, and research clus-

ters from bibliometric analysis. *Journal of Behavioral and Experimental Finance*, *32*, 100577.

Goodman, B., & Flaxman, S. (2017). European union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, *38*(3), 50–57.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies*, *33*(5), 2223–2273.

Gu, S., Kelly, B., & Xiu, D. (2021). Autoencoder asset pricing models. *Journal of Econometrics*, *222*(1), 429–450.

Gunning, D., & Aha, D. (2019). Darpa's explainable artificial intelligence (xai) program. *AI magazine*, *40*(2), 44–58.

Hastie, T., & Tibshirani, R. (1987). Generalized additive models: Some applications. *Journal of the American Statistical Association*, *82*(398), 371–386.

Ince, O. S., & Porter, R. B. (2006). Individual equity return data from thomson datastream: Handle with care! *Journal of Financial Research*, *29*(4), 463–479.

Jaeger, M., Krügel, S., Marinelli, D., Papenbrock, J., & Schwendner, P. (2021). Interpretable machine learning for diversified portfolio construction. *The Journal of Financial Data Science*, *3*(3), 31–51.

Lemhadri, I., Ruan, F., & Tibshirani, R. (2021). Lassonet: Neural networks with feature sparsity. *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, *130*, 10–18.

Lim, B., Arık, S. Ö., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748–1764.

Linardatos, P., Papastefanopoulos, V., & Kotsiantis, S. (2020). Explainable ai: A review of machine learning interpretability methods. *Entropy*, *23*(1), 18.

Lou, Y., Caruana, R., & Gehrke, J. (2012). Intelligible models for classification and regression. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 150–158.

Lou, Y., Caruana, R., Gehrke, J., & Hooker, G. (2013). Accurate intelligible models with pairwise interactions. *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 623–631.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*.

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., & Yu, B. (2019). Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, *116*(44), 22071–22080.

Nori, H., Jenkins, S., Koch, P., & Caruana, R. (2019). Interpretml: A unified framework for machine learning interpretability. *CoRR*, *abs/1909.09223*.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, *1*(5), 206–215.

Xu, S., Bu, Z., Chaudhari, P., & Barnett, I. J. (2022). Sparse neural additive model: Interpretable deep learning with feature selection via group sparsity. *ICLR 2022 Workshop on PAIR^2Struct: Privacy, Accountability, Interpretability, Robustness, Reasoning on Structured Data*.

Yang, Z., Zhang, A., & Sudjianto, A. (2020). Enhancing explainability of neural networks through architecture constraints. *IEEE Transactions on Neural Networks and Learning Systems*, *32*(6), 2610–2621.

Yang, Z., Zhang, A., & Sudjianto, A. (2021). Gami-net: An explainable neural network based on generalized additive models with structured interactions. *Pattern Recognition*, *120*, 108192.

Zschech, P., Weinzierl, S., Hambauer, N., Zilker, S., & Kraus, M. (2022). Gam(e) changer or not? an evaluation of interpretable machine learning models based on additive model constraints. *Thirtieth European Conference on Information Systems (ECIS 2022)*.

Eigenständigkeitserklärung

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Karlsruhe, den November 11, 2022

Micha Ianniello