Deep Parametric Portfolio Policies

Frederik Simon* Sebastian Weibels† Tom Zimmermann‡

September 6, 2022

Preliminary and Incomplete

Abstract

We directly optimize portfolio weights via deep neural networks by generalizing the parametric portfolio policy framework. Our results show that network-based portfolio policies result in an increase of investor utility of between 30 and 100 percent over a comparable linear portfolio policy, depending on whether portfolio restrictions on individual stock weights, short-selling or transaction costs are imposed, and depending on an investor's utility function. We provide extensive model interpretation and trace improvements over linear policies to the relevance of predictor interactions. Both approaches agree on the same dominant predictors, namely past return-based firm characteristics.

JEL classification: G11, G12, C58, C45
Keywords: Portfolio Choice, Machine Learning, Expected Utility

*University of Cologne, Department of Business Administration and Corporate Finance, simon.frederik@wiso.uni-koeln.de

†University of Cologne, Department of Business Administration and Bank Management, weibels@wiso.uni-koeln.de

‡University of Cologne, Institute for Econometrics and Statistics and Center for Financial Research, tom.zimmermann@uni-koeln.de

1 Introduction

Consider the formidable problem of an investor who wants to choose an optimal asset allocation within her equity portfolio. The literature provides her with a few options: She can opt for a traditional Markowitz approach (Markowitz, 1952) that requires estimation of expected returns, variances and covariances with the number of moments to estimate escalating quickly. At the other end of the spectrum, she might estimate a low-dimensional parametric portfolio policy (Brandt et al., 2009) but the linear model might not provide sufficient flexibility. She can also consult a large literature that relates characteristics to expected returns but even studies that consider a multitude of firm-level characteristics (e.g. Gu et al. (2020)) only investigate expected returns and do not speak to risk as perceived by different investors' objective functions.

We provide a general solution to the portfolio optimization challenge. In short, we combine the parametric portfolio policy approach that is well-suited to estimate portfolio weights for any utility function with the flexibility of feed-forward networks from the machine learning literature. The resulting approach that we label *Deep Parametric Portfolio Policy* is well-suited to integrate different utility functions, to flexibly deal with leverage or portfolio weight constraints, and to incorporate transaction costs.

Our results are fourfold. First, our model improves significantly over a standard linear parametric portfolio policy. Utility gains range from around 30% to 100% depending on model specification and the incorporation of constraints. Such gains are not restricted to only particular time periods and come with only modest increases in turnover. Second, past return-based stock characteristics turn out to be more relevant to the portfolio policy than accounting-based characteristics. This is despite return-based characteristics typically incurring higher transaction costs. Even when we model transaction costs explicitly in the objective function, return-based characteristics remain among the most relevant ones. Third, utility gains arise for a variety of investors' utility functions that we consider. While our benchmark investor is a classical mean-variance optimizer, our setup easily accommodates other utility functions. We investigate deep parametric portfolio policies for the case of constant relative risk aversion and for loss aversion, and we find substantial utility gains in all cases. Fourth, predictor interactions account for the majority of differences between the estimated weights of the linear parametric portfolio policy and our model.

In essence, our model can be interpreted as a generalization of the linear parametric portfolio policy approach in that we allow firm characteristics to be non-linearly related to portfolio weights. More specifically, we allow portfolio weights to be of one of the arguably most flexible forms - a neural network. This represents a significant conceptual change in two ways: First, by replacing the linear specification by a neural network, we allow the relation between firm characteristics and weights to be non-linear and we allow for potential interactions of firm characteristics. The literature on using machine learning methods to predict future returns shows that such flexibility is relevant to model the relation between firm characteristics and future returns, and can lead to substantial improvements over less flexible specifications (Moritz and Zimmermann, 2016; Gu et al., 2020). It is conceivable that such flexibility will also help to model the relation between portfolio weights and firm characteristics. Second, flexibility comes at the cost of having to estimate a model with a high-dimensional parameter vector. As such, it deviates from the original motivation of the parametric portfolio policy literature that aimed to reduce portfolio optimization to a low-dimensional problem in which only a small number of coefficients need to be estimated. Our benchmark model has around 5,700 parameters compared to the three parameters that need to be estimated in the application of Brandt et al. (2009). Nevertheless, Kelly et al. (2022) argue that model complexity is a virtue for return prediction, and our approach can be viewed as an exploration of that point in the context of parametric portfolio policies.

We start with a benchmark case of a largely unrestricted portfolio policy. In the benchmark case, an investor who optimizes mean-variance utility can take long and short positions with the only restriction that absolute individual stock positions cannot exceed three percent of the overall portfolio. Other aspects of the optimization remain unrestricted, in particular, the investor does not take into account transaction costs and short-selling constraints.

In the benchmark case, a network-based portfolio policy can increase investor utility by about 100% relative to a linear portfolio policy but also incurs higher turnover. Both portfolio policies take comparably large positions in individual stocks but the network-based policy has turnover that is almost twice as large. We find that the difference in turnover can be traced to the network-based policy putting larger weight on past-return based characteristics that have higher turnover.

We then investigate network-based portfolio policies in more realistic contexts that restrict investors in various ways. In particular, we explore results for the case in which an investor cannot take any short positions and for the case in which transaction costs and leverage are part of the optimization problem. In both cases, we find that network-based policies yield higher utility than a linear portfolio policy, with increases between 30% and 40%. For constrained portfolio policies the importance of past-return based characteristics decreases while still being among the most important characteristics. This matches the results of DeMiguel et al. (2020) who find that more characteristics matter under transaction costs.

Moving beyond our benchmark mean-variance investor, we explore different investor preferences: First, we show that utility gains occur for mean-variance utility optimizers with different degrees of risk aversion. We find larger utility gains for less risk averse investors and lower gains for more risk averse investors, consistent with our finding that estimated portfolio policies for more risk averse investors take less extreme positions and hold more diversified portfolios. Second, we also find that utility gains are not restricted to mean-variance utility investors. We find similar results when we consider an investor with constant relative risk aversion or with loss aversion.

Overall, our contribution can be summarized as providing a general solution to the parametric portfolio policy problem that combines recent advances in combining structural economic problems and machine learning methods (Farrell et al. (2021); Kelly et al. (2022)). Our setup seamlessly incorporates non-linearities and interactions across firm characteristics. We also demonstrate how constraints on leverage or portfolio weights can be easily added via customization of the statistical loss function. Lastly, realistic estimates of transaction costs can be taken into account as an additional constraint on the optimization problem.

1.1 Related Literature

Our work relates to three different strands of the literature. First, we build on a number of studies that have investigated the portfolio optimization process via parametric portfolio policies following the seminal study by Brandt et al. (2009). While Brandt et al. (2009) argued that it may be worthwhile to consider non-linear functions and interactions in weight modeling, subsequent papers that have implemented and extended parametric portfolio policies parameterize portfolio weights as a linear function of firm characteristics (e.g. Hjalmarsson and Manchev (2012), Ammann et al. (2016)). DeMiguel et al. (2020) incorporate transaction costs, a larger set of firm characteristics, and statistical regularization but also stay within the linear framework. Our deep parametric portfolio policy replaces the linear model with a feed-forward neural network that accounts for both non-linearity and possible interactions of firm characteristics. In addition, we use a larger set of firm characteristics than previous studies and explore different regularization techniques for both the linear and deep parametric portfolio policies.

Second, we add to a growing literature that explores the potential of machine learning algorithms in finance (e.g. Heaton et al. (2017), Gu et al. (2020), Bianchi et al. (2020), Kelly et al. (2022)). Studies in this literature typically consider a prediction task (e.g. predicting stock returns), and optimize a standard statistical loss function such as the mean squared error (or a related distance metric) between the actual and predicted values. Predicted values are used to construct portfolio weights (e.g. Moritz and Zimmermann (2016), Gu et al. (2020)). In contrast, we optimize a utility function instead of a common loss function and model portfolio weights directly as a function of firm characteristics. This is similar to Cong et al. (2021) who directly optimize Sharpe ratios using reinforcement learning. However, their model estimates scores for each stock and creates long-short portfolios based on the highest and lowest scoring stocks, whereas our deep parametric portfolio policy models stock weights directly as a function of firm characteristics. Also somewhat related Uysal et al. (2021) employ different neural networks to find optimal risk parity strategies among seven exchange-traded funds. The use of machine learning algorithms to estimate coefficients of structural models (in our case, portfolio weights) as flexible functions has also been recently proposed by Farrell et al. (2021).

Finally, we relate to the literature that examines which firm characteristics are jointly significant in explaining expected returns (see, Fama and French (2008), Green et al. (2017), Freyberger et al. (2020)). While all of these studies focus on cross-sectional regression models with extensions, Gu et al. (2020) find that neural networks perform best in predicting mean returns for a large number of firm characteristics. Our portfolio approach using neural networks considers all moments of the return distribution beyond the expected return if they are relevant to an investor's utility function. Most of this literature ignores various real world constraints such as transaction costs (with Novy-Marx and Velikov (2016) and DeMiguel et al. (2020) being important exceptions) or weight constraints, whereas we show how our model can seamlessly integrate transaction costs or other constraints.

2 Model

2.1 Expected Utility Framework and Parametric Portfolio Policies

The starting point of our framework is the parametric portfolio policy model in Brandt et al. (2009). Consider a universe of N_t stocks that an investor can invest in at each month $t \in T$. Each stock i is associated with a vector of firm characteristics $x_{i,t}$ and a return $r_{i,t+1}$ from date t to $t+1$. An investor's objective is to maximize the conditional expected utility of future portfolio returns $r_{p,t+1}$:

$$\max_{\{w_{i,t}\}_{i=1}^{N_t}} E_t\left[u(r_{p,t+1})\right] = E_t\left[u\left(\sum_{i=1}^{N_t} w_{i,t} r_{i,t+1}\right)\right], \tag{1}$$

where $w_{i,t}$ is the weight of stock i in the portfolio at date t and $u(\cdot)$ denotes the respective utility function.

Instead of directly deriving the weights $w_{i,t}$ (as e.g. following the traditional Markowitz approach), we follow Brandt et al. (2009) and parameterize the weights as a function of firm characteristics $x_{i,t}$, i.e.

$$w_{i,t} = f(x_{i,t};\theta), \tag{2}$$

where θ is the coefficient vector to be estimated. The idea behind parametric portfolio policies is that one may exploit firm characteristics in order to tilt some benchmark portfolio towards stocks that increase an investor's utility, so that $f(\cdot)$ can be expressed as

$$w_{i,t} = b_{i,t} + \frac{1}{N_t} g(x_{i,t};\theta), \tag{3}$$

where $b_{i,t}$ denotes benchmark portfolio weights such as the equally weighted or value weighted portfolio and $\hat{x}_{i,t}$ denotes the characteristics of stock i, standardized cross-sectionally to have zero mean and unit standard deviation in each cross section t.[1]

[1] The $1/N_t$ term is a normalization that allows the portfolio weight function to be applied to a time-varying number

Brandt et al. (2009) and the subsequent literature (e.g. DeMiguel et al. (2020)) restrict firm characteristics to affect the portfolio in a linear, additive manner, such that

$$w_{i,t} = b_{i,t} + \frac{1}{N_t}\theta^T \hat{x}_{i,t}. \tag{4}$$

In essence, our model can be interpreted as a generalization of the linear parametric portfolio policy approach, as we allow $\hat{x}_{i,t}$ to enter the model flexibly and non-linearly. More specifically, we allow $g(\cdot)$ in equation (3) to take arguably one of the most flexible forms - a feed-forward neural network (Hornik et al., 1989). This represents a significant conceptual change in two respects: First, by replacing the linear specification with a neural network, we allow the relationship between firm characteristics and weights to be non-linear, and we account for potential interactions of firm characteristics. The literature on the use of machine learning methods to predict future returns shows that such flexibility is relevant for modeling the relationship between firm characteristics and future returns and can lead to significant improvements over less flexible specifications (Moritz and Zimmermann, 2016; Freyberger et al., 2020; Gu et al., 2020). It is conceivable that such flexibility also helps to model the relationship between portfolio weights and firm characteristics. In this context, our model helps to understand the relationship between firm characteristics and higher moments of portfolio return distributions beyond expected returns. Second, this flexibility comes at the cost of having to estimate a model with a high-dimensional parameter vector. Thus, it departs from the original motivation of the parametric portfolio policy literature, which aimed to reduce portfolio optimization to a low-dimensional problem where only a small number of coefficients need to be estimated. Our benchmark model has about 5,700 parameters compared to the three parameters that need to be estimated when using Brandt et al. (2009). Nevertheless, (Kelly et al., 2022) argue that model complexity is a virtue for return prediction, and our approach can be viewed as an exploration of this point in the context of parametric portfolio policies.

Why might non-linear modeling of portfolio weights be important? Consider an investor who trades off mean return against return volatility as the mean-variance investor(i.e. mean-variance utility). The investor uses standard one-dimensional portfolio sorting techniques as pictured in

of stocks. Without this normalization, an increase in the number of stocks with an otherwise unchanged cross-sectional distribution of characteristics leads to more radical allocations, although the investment opportunities are basically unchanged.

Figure B.1 in Appendix B. Decile portfolios formed on short-term reversal or sales-to-price display monotonically increasing mean return.[2] At the same time, the standard deviations decile portfolios are non-linear in deciles, in particular top and bottom decile portfolios display high standard deviation. This leads to the extreme portfolios having comparatively low Sharpe ratios relative to decile portfolios in the middle of the distribution. A mean-variance (long-only) investor would therefore be indifferent between investing in any portfolio in the upper half of the short-term reversal distribution, and she would prefer to invest in portfolios in the middle of the sales-to-price distribution rather than investing in the extreme portfolios. It is exactly these kinds of relationships that a non-linear portfolio policy can capture. On top of modeling such non-linearities, our models below also allow for interactions between different signal variables that cannot be represented by one-dimensional portfolio sorts either.

2.2 Network architecture

We implement and compare a range of so-called feed-forward networks, a popular network structure that is prominently used in prediction contexts such as image recognition but has also recently been applied to stock return prediction. Conceptually, our feed-forward networks are structured to estimate optimal portfolio weights and as such differ from networks used in pure prediction contexts in two important ways.

First, the objective of our estimation is to maximize expected utility. Standard use of predictive modeling (with or without networks) tries to minimize some distance metric (e.g. mean squared error) between e.g. observed stock returns and predicted stock returns. For example, Gu et al. (2020) use neural networks to predict stock returns using a penalized mean squared error as the statistical loss function.

In contrast, we follow Brandt et al. (2009) and directly estimate portfolio weights. More specifically, we predict portfolio weights, conditional on maximizing a given utility function

$$\max_{\theta} \frac{1}{T} \sum_{t=1}^{T} u\left(r_{p,t+1}(\theta)\right) = \frac{1}{T} \sum_{t=1}^{T} u\left(\sum_{i=1}^{N_t} f(x_{i,t};\theta) r_{i,t+1}\right). \tag{5}$$

[2]We picked these two variables for illustrative purposes as these variables are the most important return- and fundamental-based variables in Gu et al. (2020).

For example, in our base case, the loss function $\mathcal{L}$ that we aim to minimize with respect to θ is the negative standard mean-variance utility:

$$\mathcal{L}(\theta) = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{\gamma}{2}\left(r_{p,t+1}(\theta) - \frac{1}{T}\sum_{t=1}^{T} r_{p,t+1}(\theta)\right)^{2} - r_{p,t+1}(\theta)\right), \tag{6}$$

where γ is the absolute risk aversion parameter. Note that minimizing Equation (6) is equivalent to maximizing mean-variance utility.

Second, our loss function requires statistical moments of a portfolio per period. In order to derive e.g. the portfolio return at t, we need to aggregate our outputs cross-sectionally. To do so, we maintain the three-dimensional structure of our data, i.e. we do not treat it as two-dimensional as e.g. Gu et al. (2020) do. Conceptually, our models can be depicted as shown in Figure 1.

[FIGURE 1 ABOUT HERE]

In Figure 1, the input data on the left form a cube (or 3D tensor) with dimensions time t, stocks i and input variables k. Input data are fed into networks with different numbers of hidden layers.[3] The output of the neural network is then normalized by $1/N_t$ and added to the benchmark portfolio b. The output of the model O is a two-dimensional matrix with dimensions $t \times i$ of portfolio weights for each stock and time period.

Constructing a neural network requires many design choices, including the depth (number of layers) and width (units per layer) of the model, respectively. Recent literature suggests that deeper networks can achiever higher accuracy with less width than wider models (Eldan and Shamir, 2016). However, for smaller data sets a large number of parameters can lead to overfitting and/or vanishing or exploding gradients.[4] Selecting the best network structure is a formidable task and not our main objective.[5] We rely on the results of Gu et al. (2020) and use their most

[3]Following Feng et al. (2018) and Bianchi et al. (2020) we only count the number of hidden layers while excluding the output layer in the remainder of this paper.

[4]When training a deep neural network with gradient-based learning and backpropagation, we find the partial derivatives by traversing the network from the last layer to the first layer. Using the chain rule, the deeper layers of the network undergo continuous matrix multiplications to compute their derivatives. If the derivatives are large, the gradient increases exponentially as we move down the model until it eventually explodes. If, on the other hand, the derivatives are small, then the gradient decreases exponentially as we progress through the model until it finally disappears.

[5]In practice, the task is often approximated by comparing a few different structures and selecting the one with the best performance.

successful model as our benchmark model. Thus, our benchmark model consists of an input layer, three hidden layers and an output layer. We apply the geometric pyramid rule (Masters (1993)), i.e. the first hidden layer consists of 32 nodes, the second hidden layer consists of 16 nodes and the third hidden layer consists of eight nodes. As a robustness check, we consider different network architectures in Appendix A.

As discussed in Section 2.1, the network's output needs to be normalized and can be interpreted as the deviation from a benchmark portfolio. In our application, the benchmark portfolio is the equally weighted portfolio in all models. A common alternative would be a value weighted benchmark portfolio where weights are determined by a stock's market capitalization. We stick with the equal weighted benchmark because there is empirical evidence (DeMiguel et al. (2009)) that it outperforms other benchmarks like the value weighted benchmark for longer periods.

At each node of the network, a linear transformation of the preceding outputs is fed into an activation function. We choose to use the leaky rectified linear unit (leaky ReLU) activation function at every node.

$$R(z) = \begin{cases} z & \text{if } z > 0 \\ \alpha z & \text{otherwise} \end{cases}, \tag{7}$$

where z denotes the input and α denotes some small non-zero constant, in our case 0.01.[6]

Moreover, we shift the activation function at every node in every hidden layer by adding a constant (commonly referred to as bias in the machine learning literature).

Our benchmark network is estimated by minimizing the loss function (utility function) in Equation (6). To do so, we apply the commonly used ADAM stochastic gradient descent optimization technique developed by Kingma and Ba (2014).

We further control for unreasonable results and overfitting in terms of portfolio weights by ex-ante imposing an upper bound on an individual stock's absolute portfolio weight of $|3\%|$, i.e.

$$|w_{i,t}| \leq 0.03. \tag{8}$$

[6]ReLU is the most popular activation function because it is cheap to compute, converges fast and is sparsely activated. The disadvantage of transforming all negative values to zero is a problem called "dying ReLU". A ReLU neuron is "dead" if it is stuck in the negative range and always outputs zero. Since the slope of ReLU in the negative range is also zero, it is unlikely that a neuron will recover once it goes negative. Such neurons play no role in discriminating inputs and are essentially useless. Over time, a large part of the network may do nothing. Leaky ReLU fixes this problem because it has small slope for negative values instead of a flat slope.

To control for the non-linearity and heavy parametrization of the model, we employ further regularization techniques to prevent overfitting: first, we add lasso (l_1) penalization of the parameters. Second, we employ early stopping on a validation data set.[7] Third, we use a dropout layer (Srivastava et al., 2014).[8] We employ dropout only in the beginning before the first hidden layer. The combination together with l_1-regularization and early stopping tremendously helps to reduce overfitting and complexity. Fourth, we adopt an ensemble approach in training our neural network (Hansen and Salamon, 1990). In particular, we initialize five neural networks with different random seeds and construct predictions by averaging the predictions from all networks. This reduces the variance across predictions since different seeds produce different predictions due to the stochastic nature of the optimization process.

Finally, we adopt our own version of a batch normalization algorithm (Ioffe and Szegedy, 2015).[9] Brandt et al. (2009) standardize characteristics cross-sectionally to have zero mean and unit standard deviation across all stocks at date t. Therefore, the predictions are deviations from the benchmark portfolio. However, applying the activation function destroys this distribution. In our model each "observation" can be interpreted as a complete cross-section (e.g. a batch size of 12 refers to 12 complete cross-sections of data). However, the model of Brandt et al. (2009) needs normalization on a cross-sectional level instead of a batch level. Thus, we employ our own version of cross-sectional normalization after applying the activation function in each hidden layer, such that the output of each node in the hidden layer is standardized cross-sectionally to have zero mean and unit standard deviation across all stocks at date t. Hence, the output of each node in each hidden layer can also be interpreted as a deviation from the benchmark portfolio.

[7]Early stopping refers to a very general regularization technique. At each new iteration, predictions are estimated for the validation sample, and the loss (utility) is constructed. The optimization is terminated when the validation sample loss starts to increase by some small specified number (tolerance) over a specified number of iterations (patience). Typically, the termination occurs before the loss is minimized in the training sample. Early stopping is a popular regularization tool because it reduces the computational cost.

[8]The basic idea of dropout is to randomly remove units (and their connections) from the neural network during training. This prevents the units from becoming too similar. During training, samples are taken from an exponential number of different "thinned" networks. At test time, it is easy to approximate the effect of averaging the predictions of all these thinned networks by simply using a single, unthinned network with smaller weights.

[9]In general, training deep neural networks is complicated by the fact that the distribution of inputs to each layer changes during training as the parameters of the previous layers change. This phenomenon is referred to as internal covariate shift and can be remedied by normalizing the layer inputs. The strength of this method is that normalization is part of the model architecture and is performed for each training mini-batch. Batch normalization allows much higher learning rates to be used and less care to be taken in initialization.

2.3 Data

We use the Open Source Asset Pricing dataset of Chen and Zimmermann (2022). The dataset contains monthly US stock-level data on 205 cross-sectional stock return predictors, covering the period from January 1925 to December 2020.

We focus on the period from January 1971 to December 2020, since comprehensive accounting data is only sparsely available in the years prior to that. In addition, we also only keep common stocks, i.e. stocks with share codes 10 and 11, and stocks that are traded on the NYSE (exchange code equal to 1) to ensure that results are not driven by small stocks. We match the data with monthly stock return data from the Center for Research in Security Prices (CRSP). We drop any observation with missing return, size and/or a return of less than -100%. We include continuous firm characteristics from Chen and Zimmermann (2022)'s categories *Price, Trading, Accounting* and *Analyst*, respectively.[10]

Finally, we follow Gu et al. (2020) and replace missing values with the cross-sectional median at each month for each stock, respectively. Additionally, similar to Gu et al. (2020) we rank all stock characteristics cross-sectionally. As in Brandt et al. (2009) and DeMiguel et al. (2020), each predictor is then standardized to have a cross-sectional mean of zero and standard deviation of one. Note that each predictor is signed so that a larger value implies a higher expected return.

Our final dataset contains 157 predictors for a total of 5,154 firms. Each month, the dataset contains a minimum of 1,213, a maximum of 1,855 and an average of 1,422 firms. Table C.1 in the appendix lists the included predictors by original paper. The last two columns in the table describe the predictor category, taken from Chen and Zimmermann (2022), and the update frequency of each predictor.[11] As part of our robustness check, we exploit that information in Appendix A to construct non-fully connected networks.

[10] All characteristics are calculated at a monthly frequency. For variables that are updated at a lower frequency, the monthly value is simply the last observed value. We assume the standard lag of six months for annual accounting data availability and a lag of one quarter for quarterly accounting data availability. For IBES, we assume that earnings estimates are available by the end date of the statistical period. For other data, we follow the respective original research in regards to availability.

[11] Many accounting-based variables are categorized as updating monthly although the accounting information only changes at an annual frequency. This is because those predictors are scaled by market capitalization in the original papers and market capitalization can change monthly based on price and shares outstanding.

2.4 Out-of-sample testing strategy

Following Brandt et al. (2009) and Gu et al. (2020), we use an "expanding window" strategy to generate out-of-sample results. More specifically, we split our data into a training sample used to estimate the model, a validation sample used to tune the hyperparameters of the model and a test sample used to evaluate the out-of-sample performance of the model.

We initially train the model on the first 20 years of the dataset, validate it on the following five years and evaluate its out of-sample-performance on the year following the validation window. We then recursively increase the training sample by one year. Each time the training sample is increased, we refit the entire model while holding the size of the validation and test window fixed. The result is a sequence of performance evaluation measures corresponding to each expanding window, in our case 30 in total. Note that this approach ensures that the temporal ordering of the data is maintained. The testing strategy is depicted graphically in Figure 2.

[FIGURE 2 ABOUT HERE]

2.5 Model interpretation

Machine learning models are notoriously difficult to interpret and neural networks are no exception. In our application, understanding the estimated relation between input (firm characteristics) and output (estimated portfolio weights) is essential though to shed light on the relation between firm characteristics and future returns, and to compare our results to the previous literature. We provide three ways of interpreting the models and identifying the most important predictors among the plethora of variables that enter our models.

First, we evaluate how the deep parametric portfolio policy exploits non-linearities and interactions. To do so we take our portfolio weights obtained from the estimated neural net model, and estimate a linear surrogate model. The linear surrogate model regresses estimated weights on firm characteristics. While the linear surrogate model perfectly explains the linear parametric portfolio policy, the R^2 of the deep parametric portfolio policy gives an indication of how well a linear model without higher-order terms and interactions can explain the estimated weights. In a next step, we estimate a penalized regression model of the estimated weights on firm characteristics as well as all possible two-way interactions. This allows us to infer the degree to which interactions and non-linearities play a role.

Second, we evaluate the variable importance by the decrease in model performance when a particular variable is missing from the model. That is, for every out-of-sample period we set all values of a variable to zero while holding the remaining variables fixed. We then calculate the utility loss as compared to the original model in every out-of-sample period and take the average across all models. For the sake of comparability, we scale the average utility loss across all variables for each model so that they add up to one. As a result, we are able to rank the variables according to the average utility loss that occurs if they are excluded from the model.

In addition, we provide another measure of variable importance based on the aforementioned linear surrogate model. Since the model contains standardized firm characteristics as predictors, a coefficient's magnitude can directly be interpreted as a measure of the variable's relevance. This method is particularly useful to identify the most important firm characteristics interactions because the linear surrogate model seamlessly controls for firm characteristics' main effects.[12]

Third, we attempt to evaluate the sensitivity of the model to each variable. Typically, partial dependence plots provide an assessment of the variables of interest over a range of values. At each value of the variable, the model is evaluated while the remaining variables remain unchanged, and the results are then averaged. However, since the sum of all weights in each cross-section is equal to one and thus the mean weight prediction is always the same, applying this method to parametric portfolio policies does not produce reasonable results. To circumvent this problem, we evaluate the sensitivity of the model before the final normalization step. We follow Gu et al. (2020) so that we evaluate individual predictors over their support [-2,2] and keep all other predictors fixed at their median value of zero. For interaction effects, we vary pairs of predictors simultaneously across their support [-2,2] while holding all other variables fixed at their median value of zero.

Finally, we also evaluate the sensitivity of the different moments of the portfolio return distribution to each characteristic. To do this, we follow the same approach as for the partial dependence on the weights, with the main difference being that we vary one characteristic over

[12]Specifically, we derive the most important two-way interactions through a penalized regression model that includes the 500 most important two-way interaction terms as well as the main variables and calculate the respective standardized coefficients.

its support and fix all other variables at their original value instead of at their median value. We do this because the higher moments depend on the covariance, coskewness, etc. of the variables.

3 Results

3.1 Benchmark portfolios

Table 1 presents the comparison between different portfolios based on their utility, weights and return characteristics. We compare a simple equally weighted and a value weighted portfolio with the parametric portfolio policy of Brandt et al. (2009) and our own deep parametric portfolio policy.[13] Analogous to Brandt et al. (2009) we provide results as follows: We report (1) the utility that a respective portfolio strategy yields, (2) distributional characteristics of the portfolio weights, (3) properties of the portfolio returns and (4) the alphas of a Fama-French six-factor model.

The first row of Table 1 reports the utility for a mean-variance investor with absolute risk aversion of five. The equally weighted and value weighted portfolio yield utilities of 0.0024 and 0.0029, respectively. The standard parametric portfolio policy greatly outperforms the simple portfolios, yielding a utility of 0.0267. However, the deep parametric portfolio policy results in a utility of 0.0469, almost twice as large as the utility from the linear parametric portfolio policy, suggesting that taking into account predictor interactions and non-linear relationships can significantly improve an investor's utility.

The next set of rows gives insight into the distribution of the respective portfolio weights. The active portfolios take comparably large positions, with the average absolute weight of the deep portfolio policy being almost nine times as large as in case of the equally weighted and value weighted portfolio, respectively. However, due to the weight constraint shown in Equation (8) these positions are not extreme. Although the average absolute weight is larger in the deep model as compared to the linear model, the maximum (1.7% versus 2.1%) and minimum weights (1.8% versus 2.2%) are smaller. Comparing the actively managed portfolios, we find that both have similar levels of leverage, with the deep parametric policy being slightly higher (387% versus 315%), yet producing almost twice as much turnover (770% versus 394%), where $w^{+}_{i,t-1}$ is the

[13] To increase comparability between the linear and deep parametric portfolio policy we differ slightly from Brandt et al. (2009) in that the linear model includes l_1-regularization and early stopping, similar to the deep model.

portfolio before rebalancing at time t, that is,

$$w_{i,t-1}^{+} = w_{i,t-1} * (1 + r_{i,t}). \tag{9}$$

As Ang et al. (2011) show, average gross leverage of hedge fund companies amounts to 120% in the period since the financial crisis 2007-2008. This indicates that both the linear and the deep portfolio policies are rather unrealistic in the benchmark case. We address this later on by including a penalty term for turnover and a constraint for leverage in our objective function in Section 4.2.

The monthly mean returns of 4.7% and 7% in the linear and deep policy case are much higher than the mean returns of around 1.1% in the equally weighted and value weighted portfolio cases due to their highly levered nature. Note that our deep model yields a 2.3 percentage point increase as compared to the linear policy, while its variance increases only modestly by 0.7 percentage points, thereby leading to a Sharpe ratio that is around 40% higher. In fact, both models significantly outperform the market porfolios with more than twice as large Sharpe ratios. In terms of skewness and kurtosis the deep portfolio policy stands out as compared to the other portfolios. In particular, the portfolio exhibits a positive skewness (1.05) and high kurtosis (6.51). However, the third and fourth moments are of no interest for an investor with mean-variance preference.

The bottom set of rows reports the alphas and its standard errors with respect to a six-factor model (that appends a momentum factor to the Fama-French five-factor model). The market portfolio alphas are both not significantly different from zero. The linear policy alpha is 3.2%. The deep policy alpha is even higher, amounting to 5.6%. Both alphas are highly statistically significant. These large unexplained returns can partially be attributed to the highly levered nature of the active portfolios, as we show in the following sections.

[TABLE 1 ABOUT HERE]

These results are robust to changing the network architecture as we show in Appendix A. More specifically, we confirm our results for different levels of model complexity and non-fully connected networks.

3.2 Surrogate model, variable importance and partial dependence

Surrogate model

As a first test to assess the relevance of non-linear relationships between weights and predictor variables, we examine how well a linear model with all predictors can explain the estimated weights of our models. Figure 3 shows the adjusted R^2 of an annual linear regression of predicted weights on firm characteristics. As expected, the linear model perfectly explains the linear portfolio policy with an adjusted R^2 of one in each year. In the case of deep portfolio policy, the linear model only explains between about 60% and 73% of the variation in the weights, suggesting that a substantial amount of the variation in the estimated weights can be attributed to non-linearity and interaction terms. Moreover, the variation that can be explained by linear effects appears to vary substantially from year to year in the deep model. With the addition of all possible two-way interactions, the surrogate model explains about 97% of the variation in DPPP weights in each year. Thus, most of the variation in DPPP weights can be explained by a linear model with interaction terms. Interestingly, this implies that the non-linear activation function is not that important to the model.

[FIGURE 3 ABOUT HERE]

Variable importance

Next, we turn to the aforementioned measure of variable importance. Figure 4 compares the most important variables in the linear and deep parametric portfolio policies according to our variable importance measure discussed in Section 2.5. For both models, we find that the majority of the most important predictors relate to past returns. Short-term reversal is the most important variable in both models, mirroring findings in Moritz and Zimmermann (2016) and Gu et al. (2020). The deep parametric portfolio policy is even more tilted towards such variables. In particular, out of the twenty most important variables in the linear parametric portfolio case, eleven are price-related, seven are accounting-related and two are analyst-related. In the deep parametric portfolio case, fourteen of the twenty most important variables are price-related, five are accounting-related and one is analyst related. As past-return based variables typically imply higher turnover, this is consistent with the higher turnover of the resulting portfolio policy. Our alternative measure of variable importance, shown in Figure B.2 in the Appendix, shows similar results.

[FIGURE 4 ABOUT HERE]

We also illustrate the main interaction effects for the deep parametric portfolio policy when controlling for the main effects of each predictor. Figure 3 illustrates that two-way interactions among firm characteristics play an important role in the deep model. Figure 5 shows the main two-way interactions from the surrogate model.

[FIGURE 5 ABOUT HERE]

Partial dependence

As for the partial dependencies, we find that the sensitivity of the respective portfolio weights to the predictors is different for the linear and the deep approach, respectively. To illustrate, we examine the sensitivity of the portfolio weights to three fundamental variables, namely the book-to-market ratio (BM), liquid assets (cash), and quarterly return on assets (roaq), as well as an analyst variable, namely earnings forecast revisions per share (AnalystRevision), and four past return-based variables, namely 12-month momentum (Mom12m), short-term reversal (STreversal), seasonal momentum (MomSeason), and intermediate momentum (IntMom). Recall that each predictor is signed, so a larger value implies a higher expected return.

Figure 6 shows the marginal effects of characteristics on the mean portfolio weights of the respective characteristic according to our partial dependence measure discussed in Section 2.5. As implied by the linearity of the approach, the variables are linearly related to expected portfolio returns in the case of the standard parametric portfolio policy. In contrast, we can see slight non-linearities for the deep model. However, the marginal relation is close to linear for the fundamental variables for both models. The most significant curvature can be observed for the intermediate momentum variable and short-term reversal variable, which is the most important variable in both models. Lastly, Figure B.3 in the Appendix displays the partial dependence of the deep parametric portfolio policy for different time periods. Figure B.3 better highlights non-linearities because results are not averaged across models.

[FIGURE 6 ABOUT HERE]

In addition, we show marginal interaction effects as described above for the deep model. We use the four past return-based variables and show partial dependence on portfolio weights grouped on five evenly distributed values of the book-to-market (BM) ratio. The results are shown in Figure 7. As expected, the marginal relationship changes between different values of the book-to-market ratio. Most of the return-based variables exhibit a negative correlation with the book-to-market ratio. As a result, the largest slope is likely to be seen in the interaction between a low book-to-market ratio and high values for the return-based variables, especially for 12-month momentum and short-term reversal.

[FIGURE 7 ABOUT HERE]

Portfolio moments

Next, we report the sensitivity of the predictors to the distribution of portfolio returns. Figure 8 shows the marginal effects of characteristics on expected portfolio returns. Again, the variables are linearly related to expected portfolio returns in the case of the standard parametric portfolio policy. In contrast, in the case of the deep portfolio policy, the variables all follow a slightly concave pattern. As already shown in Figure 4, the variables based on past returns (Mom12m, STReversal, MomSeason, IntMom) are the most important and therefore have steeper slopes, showing that the models are more sensitive to these variables.

[FIGURE 8 ABOUT HERE]

Figure 9 illustrates the sensitivity of portfolio volatility to firm characteristics. We first note that volatility is higher for the deep model, which is consistent with our main results. Moreover, the volatility of the portfolio has the highest sensitivity to STreversal among all the variables studied. Interestingly, the portfolio reaches its minimum volatility for values close to zero in the PPP case for each variable. In the DPPP case, the minimum portfolio volatility lies at values of -2 for each variable, with STreversal being the only exception. This suggests that covariance among input variables plays a larger role in the DPPP case.

[FIGURE 9 ABOUT HERE]

4 Extensions

4.1 Long only

A large majority of equity portfolios face restrictions on short selling. In Table 2, we show the results from estimating long-only portfolios. We follow Brandt et al. (2009) such that we truncate the portfolios weights at zero (while still keeping keeping the cap of 3% per stock). We rebalance the portfolio weights as follows:

$$w^*_{i,t} = \frac{max[0, w_{i,t}]}{\sum_{j=1}^{N_t} max[0, w_{i,t}]}. \tag{10}$$

This term is added at the end of the optimization process to make sure the portfolio weights sum to one.

Again, the deep portfolio policy yields the highest utility. However, the utility decrease compared to the unconstrained case is also the highest for the deep portfolio policy. Still, the utility of the deep portfolio policy is around four times higher than the utility of the market portfolios and around 40% higher than the utility of the linear portfolio policy.

As in the unconstrained case, the estimated models produce higher maximum portfolio weights, however, the deep portfolio policy yields a maximum portfolio weight of 1.64% whereas the linear model only yields a maximum portfolio weight of 0.42%. Turnover follows a similar pattern, i.e. both active portfolios result in a much higher turnover than the market portfolios and the deep portfolio policy produces a higher turnover than the linear portfolio policy (125% versus 72%). Different from the table for the unconstrained model, this table reports the fraction of weights that are equal to zero. Interestingly, the deep portfolio policy does not include 11% of stocks, while the linear portfolio policy does not include 32% of the available stocks. Since the deep portfolio policy invests in more stocks but has a higher individual maximum weight, this indicates that many weights are possibly very low.

The deep portfolio policy yields higher expected returns than the linear portfolio policy, with a moderate increase in volatility resulting in a Sharpe ratio that is around 20% higher than the Sharpe ratio of the linear portfolio policy. Interestingly, the third and fourth moments of all portfolio policies are similar and not heavily skewed or tailed. Lastly, the alphas of the Fama-French model are a lot smaller, while still being highly significant in both the linear and the deep portfolio policy case. The estimated portfolios are much more realistic without the ability to take extreme short positions. Nonetheless, the deep portfolio policy is still able to outperform all other portfolios.

The comparison between the unconstrained (Table 1) and the long-only case (Table 2) also yields interesting insights. First, the unconstrained portfolio can consist of both positive and negative weights, while the constrained portfolio can only consist of positive weights. Second, the unconstrained portfolio benefits from using the short positions as leverage to increase exposure to the long positions. Consistent with those arguments, the linear portfolio policy has a similar fraction of short positions and stocks not held in the two models. Also, the maximum weight of the linear portfolio policy decreases by around 80% in the long-only case as compared to the unconstrained case. Interestingly, both findings do not apply to the deep portfolio policy. The fraction of short positions is a lot higher than the fraction of stocks not held in the long-only deep portfolio policy. Moreover, the maximum weight is similar in the unconstrained and constrained case. This can be attributed to the non-linearity of the deep model and interactions.

[TABLE 2 ABOUT HERE]

The R^2 that a surrogate model with first order and all possible two-way interactions achieves in explaining the portfolio weights, lies between around 0.88 and 0.96. In contrast, the benchmark model achieves a more consistent R^2 of about 0.98. The long-only model achieves the lowest R^2 of all models, which might be due to the strong penalty function that cannot be modeled by the linear surrogate model. Figure 10 depicts this graphically.

[FIGURE 10 ABOUT HERE]

In terms of variable importance, the picture is also similar to the unconstrained models. Figure 11 shows the variable importance of the 50 most important firm characteristics, ranked by average importance across all models. These include the two benchmark models, the linear and deep long-only models, and the linear and deep constraint models from Section 4.2. Each column corresponds to a single model, and the color gradations within each column indicate the most important (black) to least important (white) firm characteristics. The third and fourth columns correspond to the long-only models and show that the importance of the variables is similar to the benchmark models. In both the unconstrained and the long only models, characteristics based on past returns are at the top, with short-term reversal being the most important variable in three of the four models. In the linear long model the industry return of big firms (IndRetBig) exhibits the highest importance. Moreover, the importance in terms of values is similar between the benchmark and the long-only models. To conclude, these results show that the long-only investor also relies heavily on past return-based characteristics.

[FIGURE 11 ABOUT HERE]

4.2 Transaction costs and leverage

As mentioned in the previous sections, the results of the unconstrained linear and the deep portfolio policy yield unfeasible portfolios with high leverage and turnover. To show that the deep portfolio policy also outperforms the regular portfolio policy in a more realistic setting, we include a penalty term for transactions costs similar to DeMiguel et al. (2020) and include an additional constraint for maximum leverage.

In our estimation, we use estimated transaction costs from Chen and Velikov (2021).[14] Thus, analogously, we define transaction costs $\kappa_{i,t}$ as the effective half bid-ask spread. We follow DeMiguel et al. (2020) in constructing the penalty term added to the policy optimization as

$$TC = E_t[\sum_{i=1}^{N_t} |\kappa_{i,t}(w_{i,t} - w_{i,t-1}^{+})|], \tag{11}$$

where $w_{i,t-1}^{+}$ is the portfolio before rebalancing as in Equation (9).

The leverage constraint is constructed analogously to our weight constraint in Equation (8). Ang et al. (2011) show that the average gross leverage of hedge fund companies amounts to 120% in the period since the financial crisis 2007-2008. We use a slightly more conservative number of a maximum leverage of 100%. The penalty is constructed such that the gross leverage cannot exceed 100% in a single period in model training. This constraint is formulated for every period t as

$$\sum_{i=1}^{N_t} w_i I(w_i < 0) \geq -1 \tag{12}$$

for each period, where $I(w_i < 0)$ is a vector where an element is one if the corresponding portfolio weight is smaller than zero and zero otherwise.

Table 3 shows the results for the penalized and constrained optimization process. We see that the constraints lead to a decrease in utility for the deep and linear policy. The utility decrease is greater for the deep policy. Both estimated portfolios still outperform the market portfolios. Interestingly, the constraints lead to the deep portfolio policy being much closer to the linear one. This indicates that the deep model exploits the short-selling ability and characteristics with high turnover to create high weights in good performing stocks with less diversification. Nevertheless, the deep model still has a higher turnover than the linear model (168% versus 97%). Therefore, the model can still generate more mean-variance utility despite higher turnover than the linear model. Overall, in both models, the maximum and minimum positions are less extreme than in the unconstrained case and thus more realistic compared to the unconstrained case.

Furthermore, the return and the variance decrease in both active models. However, the linear portfolio policy only suffers a small decrease in Sharpe ratio, while the deep portfolio policy's

[14]We thank the authors for making an updated version of the data available.

Sharpe ratio decreases by around a third. The third and fourth moment are similar across all portfolios. The alphas of the estimated models are much smaller, but still highly significant.

[TABLE 3 ABOUT HERE]

Regressing the portfolio weights onto a model of first-order as well as all possible two-way interaction terms yields a yearly R^2 of around 0.98 to 0.99. This results in the highest R^2 of all models. Comparing this to the benchmark as well as the long-only case shows, that terms of an order above two as well as non-linearities further lose their influence on the portfolio weights when introducing transaction costs as well as leverage constraints, i.e. drawing a more realistic picture. Figure 10 depicts this graphically.

Comparing the variable importance of the firm characteristics with the previous models, we find that this set of constraints leads to a very different picture in terms of variable importance. Figure 11 shows the importance of the variables for the constrained models in columns five and six. The figure illustrates that the importance of characteristics based on past returns is much lower compared to the previous four models. Overall, short-term reversal loses its place as the most important variable in the linear model, and we can see that the loadings are much more balanced across variables similar to the results of DeMiguel et al. (2020). In the deep model, short-term reversal is still the most important variable, but it becomes evident that its importance is lower than in the previous models. As in the linear model, the loadings become more balanced in the deep model when introducing transaction costs and leverage constraints. This is also underlined by lower (higher) maximum (minimum) portfolio weights compared to the previous models. The mean absolute portfolio weights are also much smaller than for the benchmark portfolios. This shows that the constraints lead to a more diversified portfolio, which is reflected in a more balanced importance of firm characteristics.

5 Different investor utility functions

5.1 Different risk aversion parameters

Different investors may exhibit different levels of risk aversion. Our benchmark model uses an absolute risk aversion coefficient of five. Table 4 shows how our model performs for different degrees of risk aversion in the mean-variance case. In order to meaningfully interpret the differences in utility, we do not report utility itself, but rather the difference in utility relative to a constant benchmark, i.e. an equally weighted portfolio. Other than that, we report the same result metrics as before.

The results indicate that investors with a risk aversion of five experience the biggest gains in utility relative to the equally weighted portfolio benchmark. In our case, utility increases seem to follow a concave trend that peaks for γ equal to five. In general, we observe that utility gains over an equally weighted portfolio decrease with higher risk aversion, which is due to the fact, that the portfolio of the highly risk averse investor is more diversified and therefore, closer to the equally weighted portfolio. This shows, that the equally weighted portfolio performs better for investors with higher risk aversion.

We further observe a negative correlation between risk aversion and absolute portfolio weights as well as leverage and turnover. This aligns with the intuition of more risk averse investors not focusing on single high return characteristics, but rather on diversifying their portfolio with a more balanced weight distribution. This in turn results in portfolios that display lower expected returns, but also lower volatility for more risk averse investors. Moreover, all portfolios seem to have a similar Sharpe ratios. The third and fourth moment of the portfolio return distributions tend to be less extreme the higher the risk aversion, indicating that the portfolio distributions of the investors with higher risk aversion is more normally distributed. Intuitively, as the portfolios become more realistic, the alpha of the Fama-French regression decreases.

[TABLE 4 ABOUT HERE]

5.2 CRRA and loss aversion

Analogously to varying risk aversion for a mean-variance investor, we can account for different investor types by changing the utility function in our optimization process in Equation (1). In particular, we explore linear and deep portfolio policies for an investor with constant relative risk aversion utility defined as

$$u(r_{p,t+1}) = \frac{(1 + r_{p,t+1})^{1-\gamma}}{1-\gamma}, \tag{13}$$

where γ is the relative risk aversion of the investor, and for a loss-averse investor according to Tversky and Kahneman (1992) with utility defined as

$$u(r_{p,t+1}) = \begin{cases} -l(\overline{W} - (1 + r_{p,t+1}))^b & \text{if } (1 + r_{p,t+1}) < \overline{W} \\ ((1 + r_{p,t+1}) - \overline{W})^b & \text{otherwise} \end{cases}, \tag{14}$$

where $\overline{W}$ is a reference wealth level determined in the editing stage, the parameter l measures the investor's loss aversion and the parameter b captures the degree of risk seeking over losses and risk aversion over gains.

Table 5 reports the results for the linear and deep portfolio policy for an investor with constant relative risk aversion of five and an investor with a subjective wealth level $\overline{W}$ equal to one, loss aversion of 2.5 and parameter value b equal to one which corresponds to pure loss aversion. Strikingly, for both preferences the deep portfolio policy achieves higher utility than the linear portfolio policy.

The results for the CRRA preferences are similar to those for mean-variance preferences with similar risk aversion, except that the third and fourth moment of the deep policy are not as extreme. The differences in the higher moments can be attributed to the investor's preference for higher order moments, which differentiate the CRRA investor from an equally risk-averse mean-variance investor. In our data, however, the effect of the higher order moments is not strong enough to heaviliy change the portfolio weight distribution and the resulting portfolio returns.

By far the most interesting part of the loss averse investor's preference is the fact that she actually cares about the mean to standard deviation ratio of returns, which measures the size of the tail of the portfolio return distribution, rather than the mean to variance ratio, which is important to a mean-variance investor. This is also reflected in the results in Figure 5. Both portfolios show a high variance compared to the mean-variance and CRRA investor, however, they also display a higher skewness. This high positive skewness illustrates a highly right tailed distribution. Interestingly, the Sharpe ratios between the linear and deep portfolio policy are very close, while the deep model results in more than twice the utility than the linear model. The deep portfolio policy produces a high variance paired with a high skewness and kurtosis. Thus, the portfolio return distribution is heavily tailed to the right with no particularly high losses. The weight distribution of the portfolios is still very similar to the other utility models, while the deep portfolio policy produces slightly higher leverage and turnover.

[TABLE 5 ABOUT HERE]

6 Conclusion

Building on the parametric portfolio policy of Brandt et al. (2009), we show that feed-forward neural networks can be used to optimize portfolios based on a large number of firm characteristics for different investor preferences. We develop a flexible framework that can be used to implement neural networks for portfolio choice problems to optimize different utility functions with flexible constraints. Our results show that neural networks can be used to optimize portfolio weighting directly based on firm characteristics. Further, we show that traditional distance loss functions can be replaced by context-specific utility functions. Our empirical results indicate that neural networks perform significantly better than linear models in regards to portfolio allocation, suggesting that firm characteristics are non-linear and interactive. Moreover, we can use our framework to shed light on the relationship between firm characteristics and higher moments of the portfolio return distributions. Finally, we find that the models for a mean-variance investor with an absolute risk aversion of five are consistent when it comes to a set of dominant predictors, the strongest of which are associated with price trends, including short-term reversals and momentum. Moreover, we find that neural networks are able to better exploit these predictors.

The results show that overall, neural networks are successful in portfolio choice problems. They help to solve the challenge of identifying reliable economic factors for asset pricing by allowing factors to affect not only the expected return, but to also affect higher moments of the return distribution non-linearly and in an interactive manner. Finally, our flexible approach helps practitioners to create reasonable portfolio allocations based on firm characteristics and preferences, highlighting the growing role of machine learning and non-linear models in finance.

References

Ammann, M., G. Coqueret, and J.-P. Schade (2016). Characteristics-based portfolio choice with leverage constraints. *Journal of Banking & Finance 70*, 23–37.

Ang, A., S. Gorovyy, and G. B. van Inwegen (2011). Hedge fund leverage. *Journal of Financial Economics 102*(1), 102–126.

Bianchi, D., M. Büchner, and A. Tamoni (2020). Bond Risk Premiums with Machine Learning. *The Review of Financial Studies 34*(2), 1046–1089.

Brandt, M. W., P. Santa-Clara, and R. Valkanov (2009). Parametric Portfolio Policies: Exploiting Characteristics in the Cross-Section of Equity Returns. *The Review of Financial Studies 22*(9), 3411–3447.

Chen, A. Y. and M. Velikov (2021). Zeroing in on the Expected Returns of Anomalies. Working Paper.

Chen, A. Y. and T. Zimmermann (2022). Open source cross-sectional asset pricing. *Critical Finance Review 27*(2), 207–264.

Cong, L., K. Tang, J. Wang, and Y. Zhan (2021). Alphaportfolio: Direct construction through deep reinforcement learning and interpretable ai. *Available at SSRN 3554486*.

DeMiguel, V., L. Garlappi, and R. Uppal (2009). Optimal Versus Naive Diversification: How Inefficient is the 1/N Portfolio Strategy? *The Review of Financial Studies 22*(5), 1915–1953.

DeMiguel, V., A. Martín-Utrera, F. J. Nogales, and R. Uppal (2020). A Transaction-Cost Perspective on the Multitude of Firm Characteristics. *The Review of Financial Studies 33*(5), 2180–2222.

Eldan, R. and O. Shamir (2016). The power of depth for feedforward neural networks. In V. Feldman, A. Rakhlin, and O. Shamir (Eds.), *29th Annual Conference on Learning Theory*, Volume 49 of *Proceedings of Machine Learning Research*, pp. 907–940.

Fama, E. F. and K. R. French (2008). Dissecting anomalies. *The Journal of Finance 63*(4), 1653–1678.

Farrell, M. H., T. Liang, and S. Misra (2021). Deep learning for individual heterogeneity: An automatic inference framework. arXiv:2010.14694.

Feng, G., J. He, and N. G. Polson (2018). Deep Learning for Predicting Asset Returns. arXiv:1804.09314.

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

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

Green, J., J. R. M. Hand, and X. F. Zhang (2017, 03). The Characteristics that Provide Independent Information about Average U.S. Monthly Stock Returns. *The Review of Financial Studies 30*(12), 4389–4436.

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

Hansen, L. and P. Salamon (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence 12*(10), 993–1001.

Heaton, J. B., N. G. Polson, and J. H. Witte (2017). Deep learning for finance: deep portfolios. *Applied Stochastic Models in Business and Industry 33*(1), 3–12.

Hjalmarsson, E. and P. Manchev (2012). Characteristic-based mean-variance portfolio choice. *Journal of Banking & Finance 36*(5), 1392–1401.

Hornik, K., M. Stinchcombe, and H. White (1989). Multilayer feedforward networks are universal approximators. *Neural Networks 2*(5), 359–366.

Ioffe, S. and C. Szegedy (2015, 07–09 Jul). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *37*, 448–456.

Kelly, B. T., S. Malamud, and K. Zhou (2022). The virtue of complexity in machine learning portfolios. Swiss Finance Institute Research Paper Series 21-90.

Kingma, D. P. and J. Ba (2014). Adam: A method for stochastic optimization.

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance 7*(1), 77–91.

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

Moritz, B. and T. Zimmermann (2016). Tree-based conditional portfolio sorts: The relation between past and future stock returns. *Available at SSRN 2740751*.

Novy-Marx, R. and M. Velikov (2016). A taxonomy of anomalies and their trading costs. *The Review of Financial Studies 29*(1), 104–147.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research 15*(56), 1929–1958.

Tversky, A. and D. Kahneman (1992). Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty 5*(4), 297–323.

Uysal, A. S., X. Li, and J. M. Mulvey (2021). End-to-end risk budgeting portfolio optimization with neural networks. *10.48550/ARXIV.2107.04636*.

Figures

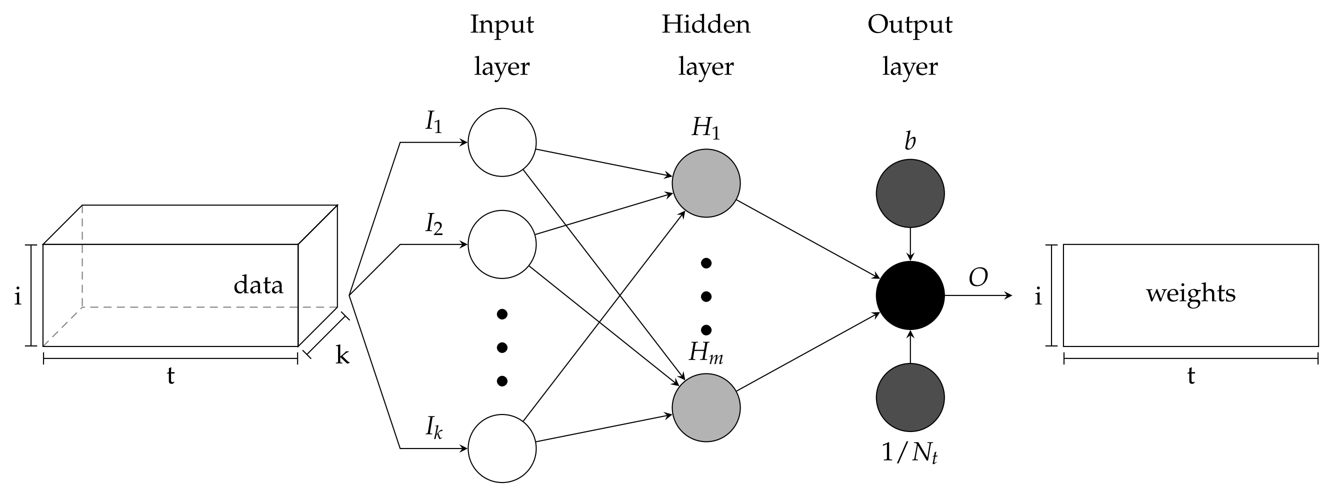


Figure 1: Neural Network Structure
This figure presents the core structure of our neural networks. White circles denote the input layer, grey circles denote the hidden layer and black circles denote the output layer. The data cube on the left depicts the structure of our data, i.e. we have k variables across i cross-sections in t periods. The rectangle on the right depicts our output, i.e. weights across i cross-sections in t periods. The output of the neural network is normalized by $1/N_t$ and added to the benchmark portfolio b. The final output is labeled O.

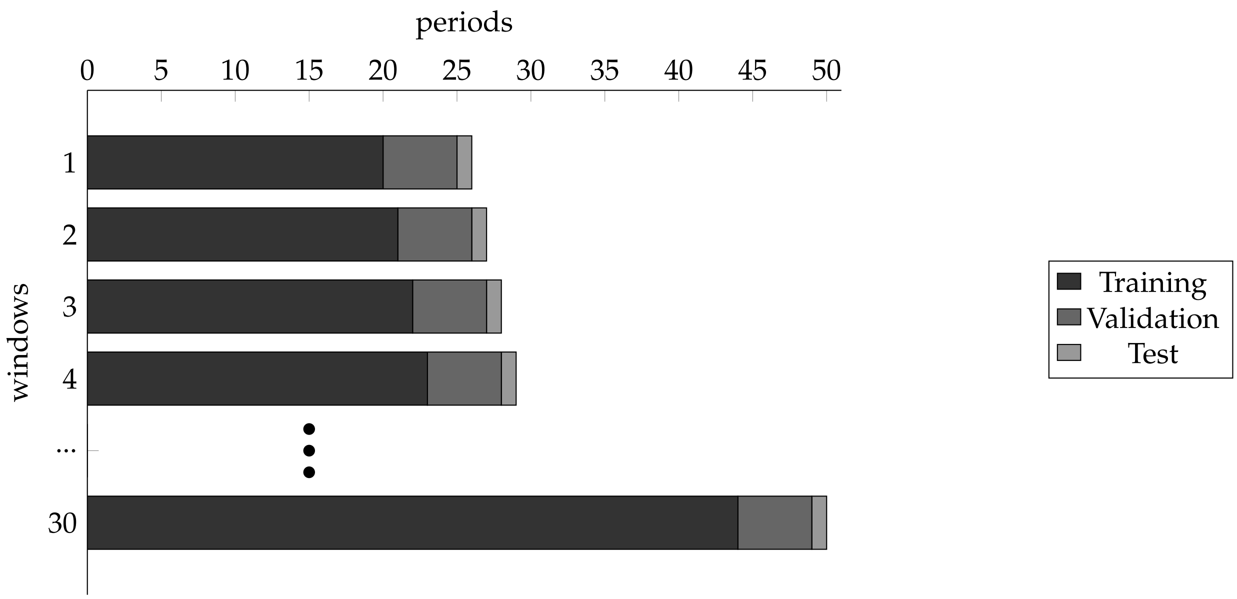


Figure 2: Out-Of-Sample Testing Strategy
This figure presents our out-of-sample testing strategy. We recursively increase our training window, presented by the black portion of each bar, while holding validation and test window constant, presented by the grey portions of each bar.

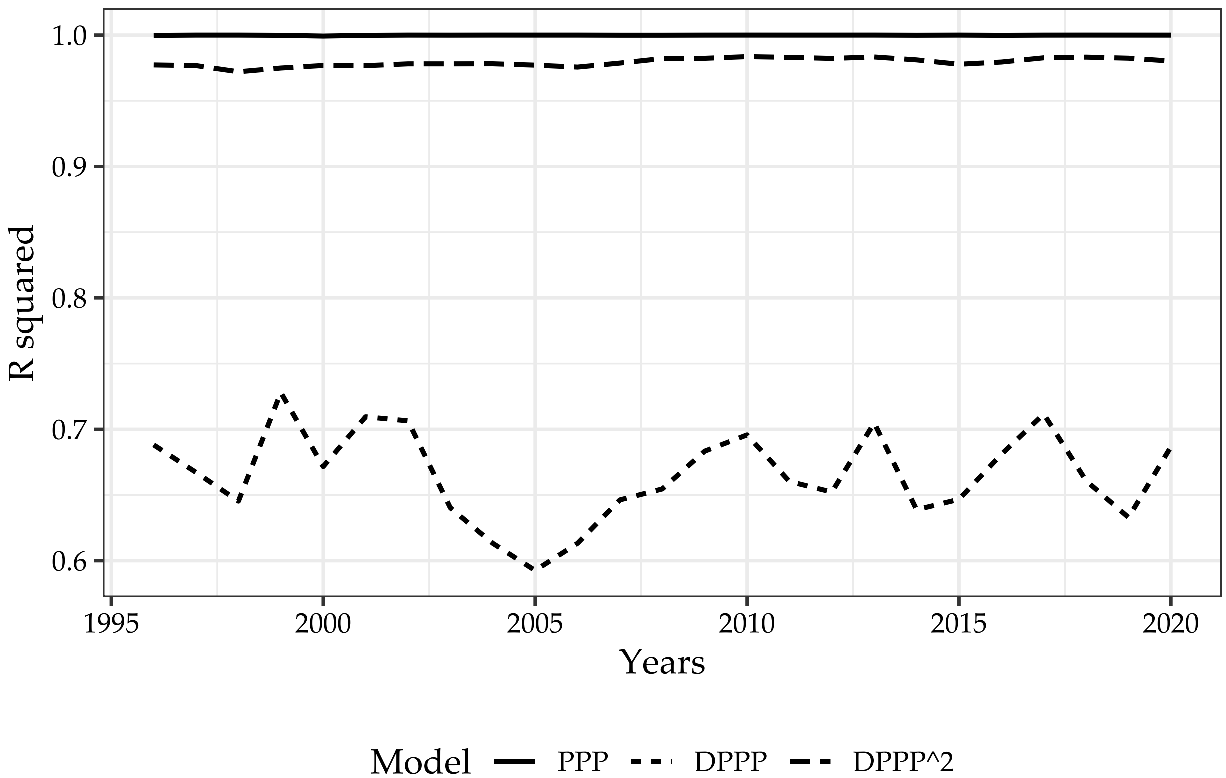
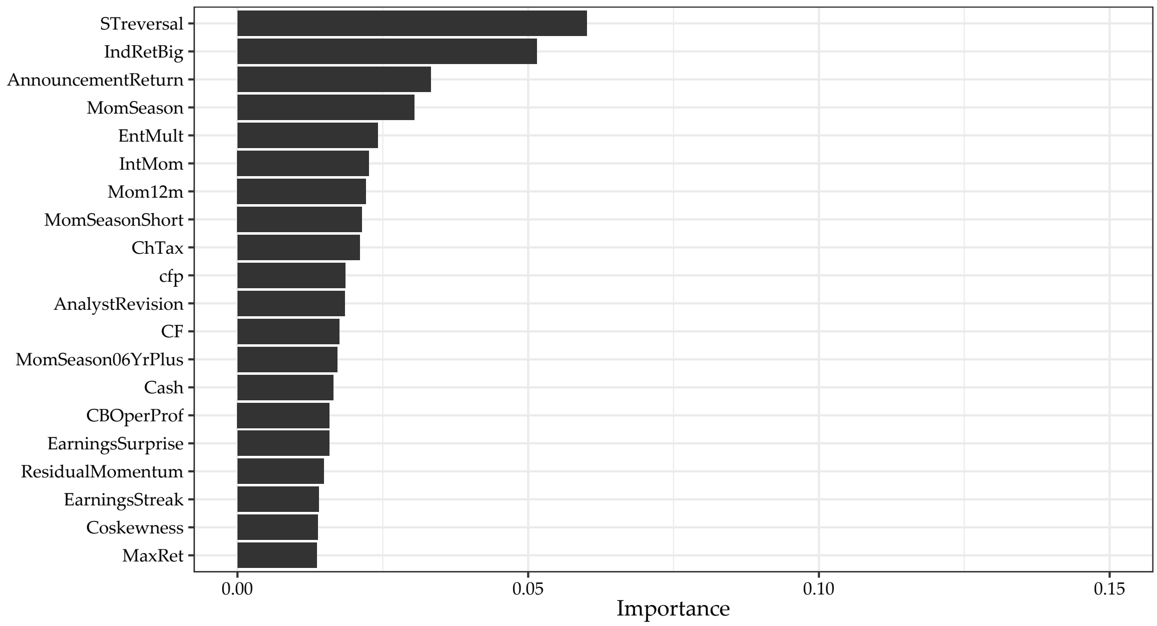
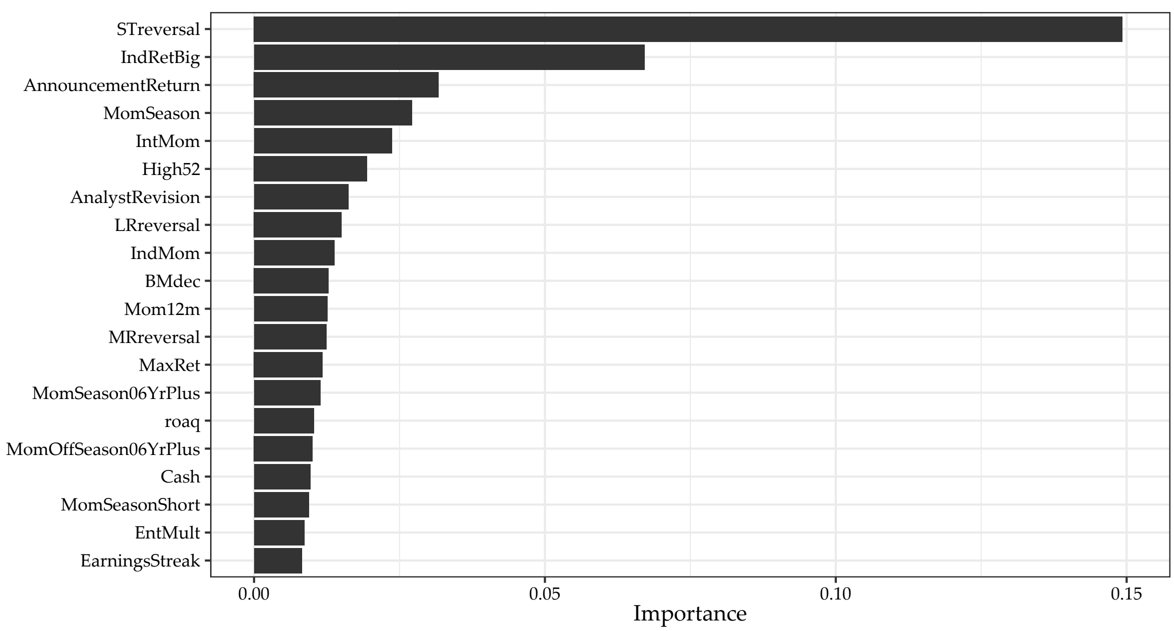


Figure 3: Surrogate R^2
This figure depicts the R^2 of the surrogate models in the benchmark case. More specifically, the "PPP"-line depicts the R^2 of a linear surrogate model in case of the PPP, the "DPPP"-line depicts the R^2 of a linear surrogate model in case of the DPPP and the "DPPP^2"-line depicts the R^2 of a surrogate model including first order effects and all possible two-way interactions.



(a) PPP



(b) DPPP

Figure 4: Variable importance for PPP and DPPP
Variable importance for the 20 most influential variables in the linear and deep parametric portfolio policy. Variable importance is an average over all training samples and normalized to sum to one.

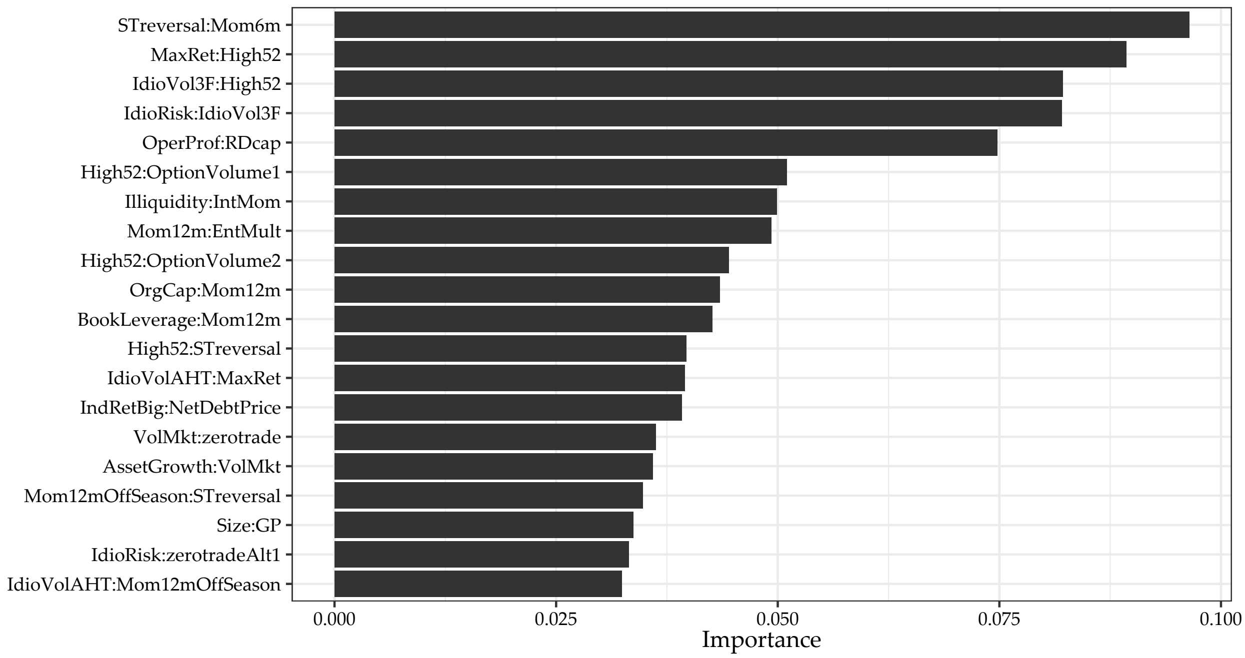


Figure 5: Variable importance of two-way interactions for DPPP
Variable importance for the 20 most influential two-way interactions in the surrogate model for the deep parametric portfolio policy. Variable importance is an average over all test samples and normalized to sum to one.

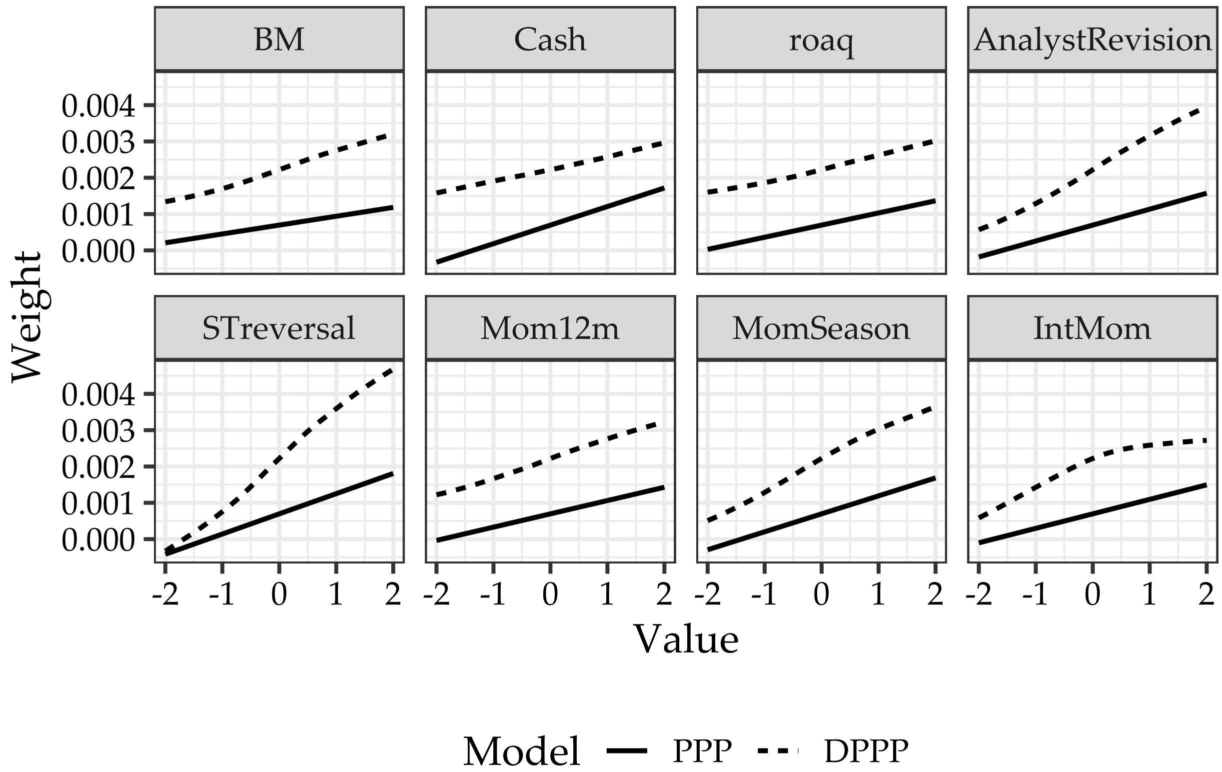


Figure 6: Marginal association between portfolio weights and characteristics
The panels show the sensitivity of mean portfolio weights (vertical axis) to the individual characteristics, holding all other covariates fixed at their original values.

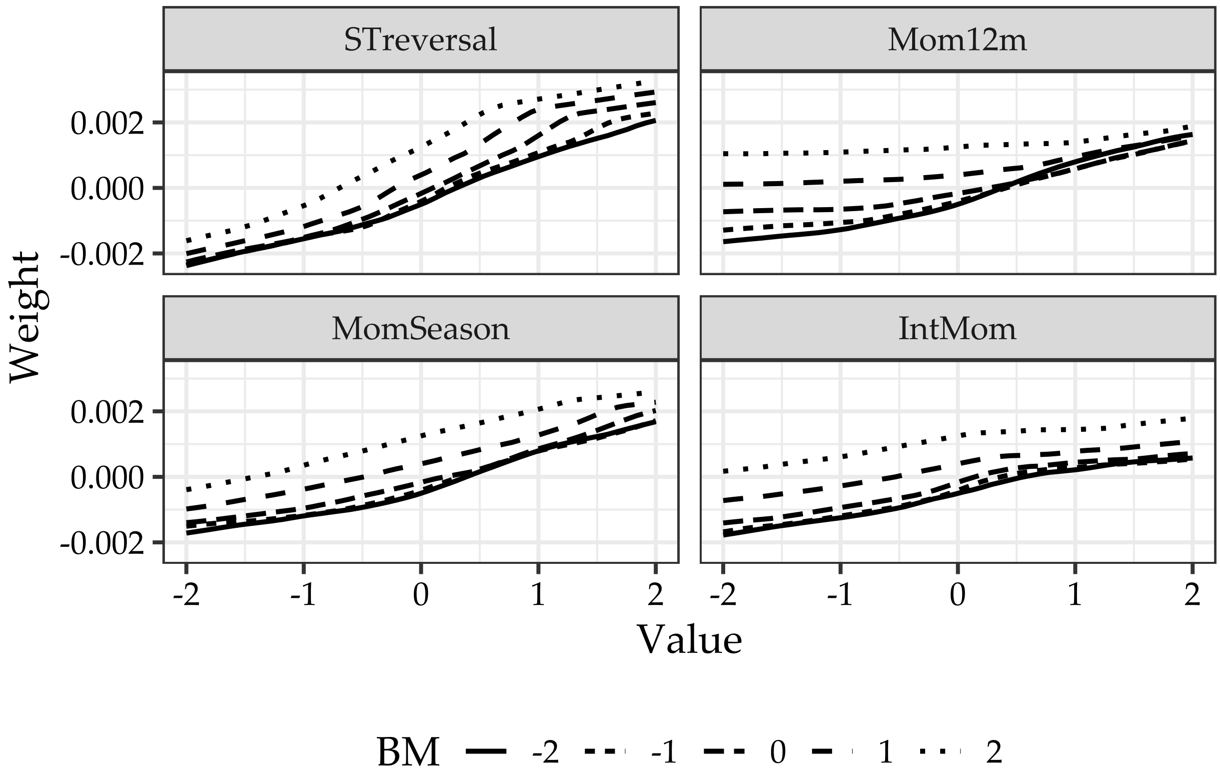


Figure 7: Marginal association between portfolio weights and interaction of past return based characteristics with BM
The panels show the sensitivity of mean portfolio weights (vertical axis) to four past return based characteristics grouped on the five values of book-to-market ratio (BM), holding all other covariates fixed at their original values.

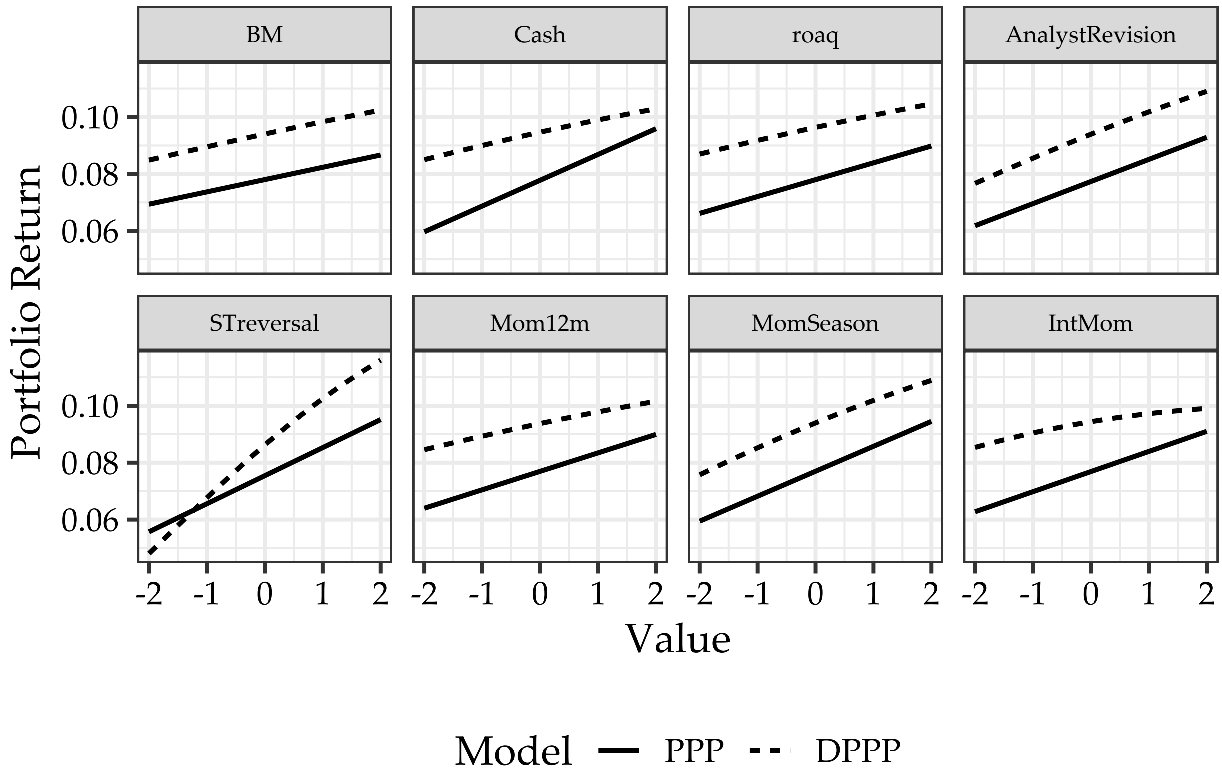


Figure 8: Marginal association between expected portfolio returns and characteristics
The panels show the sensitivity of expected portfolio returns (vertical axis) to the individual characteristics, holding all other covariates fixed at their original values.

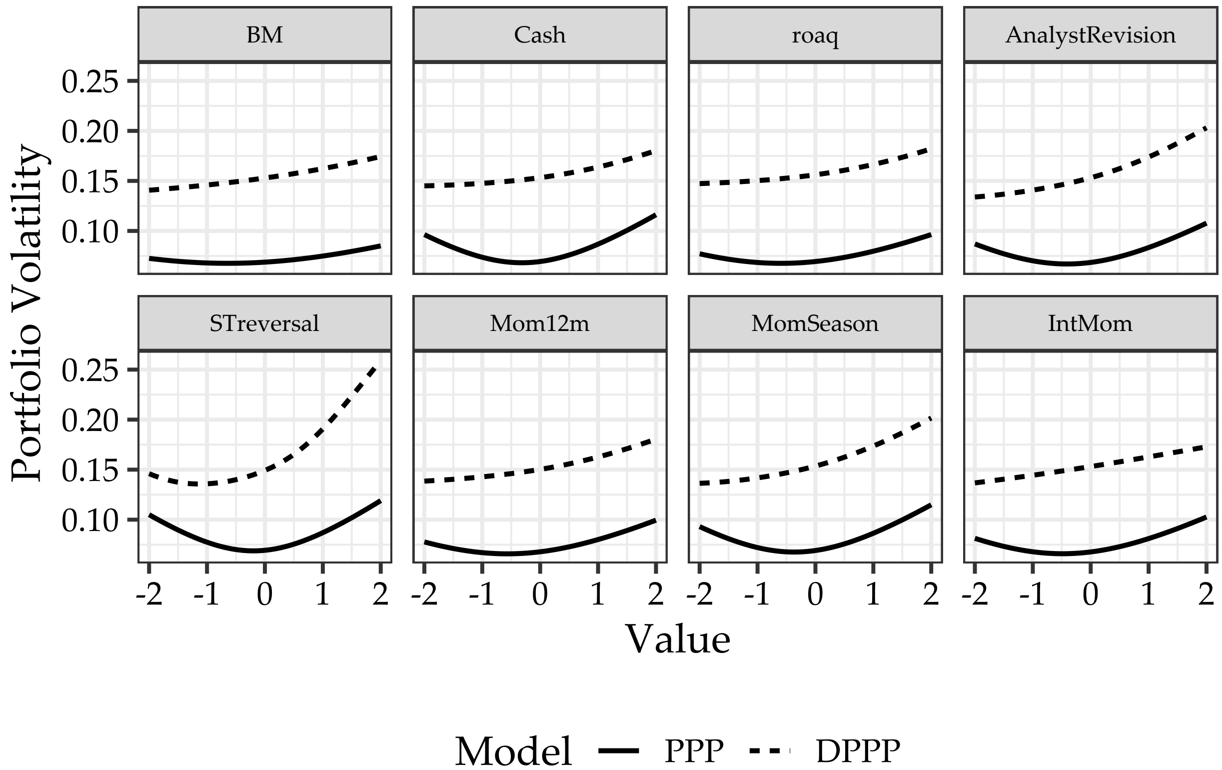


Figure 9: Marginal association between the portfolios' volatility and characteristics
The panels show the sensitivity of the portfolios' volatility (vertical axis) to the individual characteristics, holding all other covariates fixed at their original values.

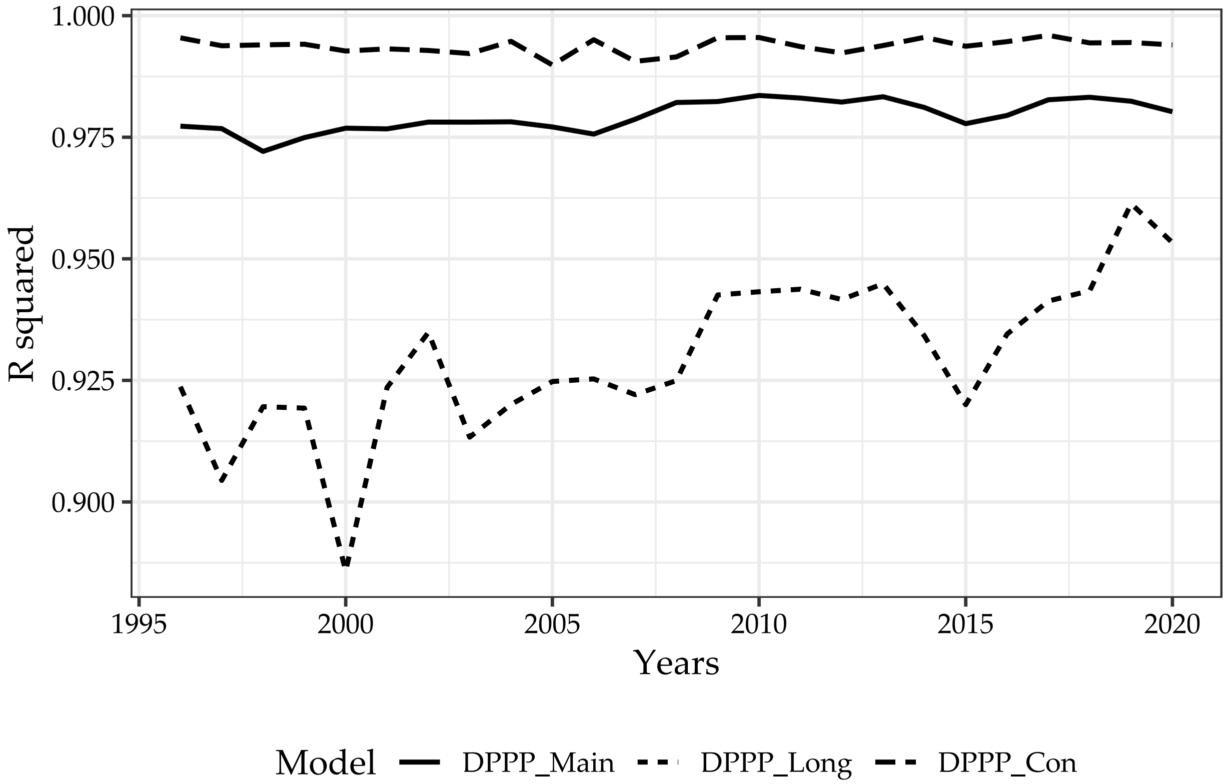


Figure 10: Surrogate R^2 across models
This figure depicts the R^2 of the surrogate model including first order and all possible two-way interactions in the benchmark, as well as the long-only and transaction cost/leverage constraint case. More specifically, the "DPPP_Main"-line depicts the R^2 in the benchmark case, the "DPPP_Long"-line depicts the R^2 in the long-only case and the "DPPP_Con"-line depicts the R^2 in the transaction cost/leverage constraint case.

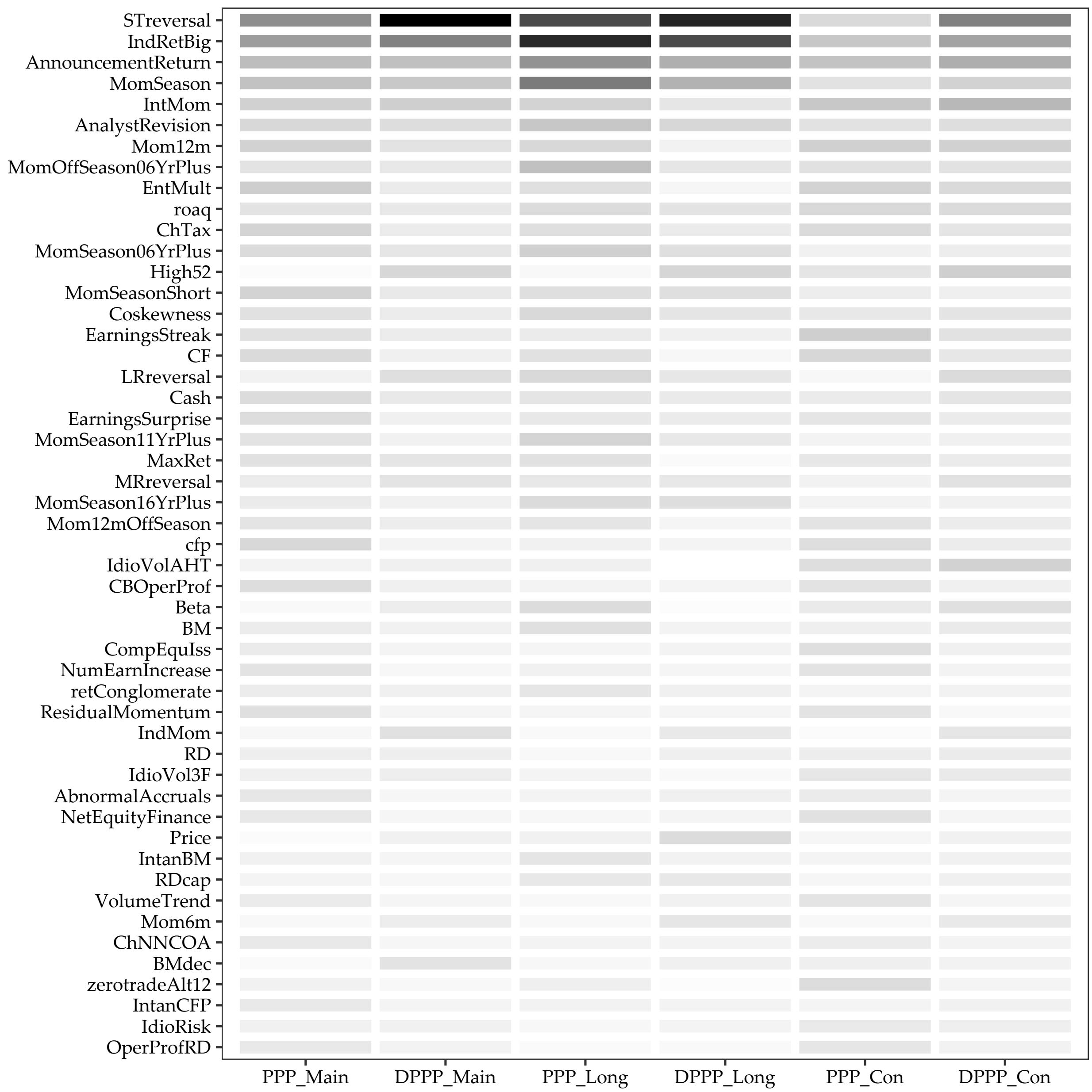


Figure 11: Variable importance across models
Ranking of the 50 most important stock characteristics in terms of overall model contribution. Characteristics are ranked by average importance across all models, with the most influential characteristics at the top and the least influential characteristics at the bottom. Columns correspond to individual models, with "Main" representing unconstrained models, "Long" representing long-only models, and "Con" representing models with constrained leverage and transaction costs. The color gradations within each column indicate the most important (black) to the least important (white) stock characteristics.

Tables

Table 1: Deep and linear portfolio policy

	EW	VW	PPP	DPPP
Utility	0.0024	0.0029	0.0267	0.0469
$\|w_i\| * 100$	0.0694	0.0694	0.5060	0.6057
$max\ w_i * 100$	0.0704	0.1113	2.0748	1.7260
$min\ w_i * 100$	0.0704	0.0410	-2.2097	-1.8370
$\sum w_i I(w_i < 0)$	0.0000	0.0000	-3.1475	-3.8665
$\sum w_i I(w_i < 0)/N_t$	0.0000	0.0000	0.4334	0.4411
$\sum \|w_{i,t} - w^{+}_{i,t-1}\|$	0.0931	0.0779	3.9370	7.6984
Mean	0.0110	0.0105	0.0468	0.0701
StdDev	0.0587	0.0552	0.0897	0.0965
Skew	-0.3716	-0.5039	-0.1451	1.0537
Kurt	3.6591	3.3455	1.8391	6.5084
SR	0.1865	0.1908	0.5216	0.7266
$FF5 + Mom\ \alpha$	-0.0002	-0.0003	0.0323	0.0559
$StdErr(\alpha)$	0.0007	0.0006	0.0040	0.0051

This table shows out-of-sample estimates of the deep and linear portfolio policies with 157 firm characteristics as specified in Equation 1 and optimized for a mean-variance investor with absolute risk aversion of five. The regular portfolio policy is a linear model for Equation 3, while the deep model is a feed-forward neural network with three hidden layers and 32, 16, and eight nodes, respectively. We use data from the Open Source Asset Pricing Dataset from January 1971 to December 2020. The columns labeled "EW", "VW", "PPP" and "DPPP" show the statistics of the equal-weighted portfolio, value-weighted portfolio, parametric portfolio policy, and deep parametric portfolio policy, respectively. The first row shows the utility of the investor. The second set of rows shows statistics on portfolio weights averaged over time. These statistics include the average absolute portfolio weight, the average maximum and minimum portfolio weights, the average sum of negative weights in the portfolio, the average proportion of negative weights in the portfolio, and the turnover in the portfolio. The third set of rows shows the first four moments of the final portfolio return distributions as well as the Sharpe ratios. The bottom panel shows the alphas and their standard errors with respect to the Fama-French five-factor model extended to include the momentum factor.

Table 2: Long-only deep and linear portfolio policy

	EW	VW	PPP	DPPP
Utility	0.0024	0.0029	0.0084	0.0116
$\lvert w_i \rvert * 100$	0.0694	0.0694	0.0694	0.0694
$max\ w_i * 100$	0.0704	0.1113	0.4155	1.6420
$min\ w_i * 100$	0.0704	0.0410	0.0000	0.0000
$\sum w_i I(w_i < 0)$	0.0000	0.0000	0.0000	0.0000
$\sum w_i I(w_i = 0)/N_t$	0.0000	0.0000	0.3173	0.1148
$\sum \lvert w_{i,t} - w^{+}_{i,t-1} \rvert$	0.0931	0.0779	0.7222	1.2519
Mean	0.0110	0.0105	0.0153	0.0198
StdDev	0.0587	0.0552	0.0526	0.0573
Skew	-0.3716	-0.5039	-0.5551	-0.4191
Kurt	3.6591	3.3455	3.5843	4.0876
SR	0.1865	0.1908	0.2900	0.3447
$FF5 + Mom\ \alpha$	-0.0002	-0.0003	0.0048	0.0090
$StdErr(\alpha)$	0.0007	0.0006	0.0008	0.0011

This table shows out-of-sample estimates of the deep and linear portfolio policies with long-only weights in Equation 10 with 157 firm characteristics as specified in Equation 1 and optimized for a mean-variance investor with absolute risk aversion of five. The regular portfolio policy is a linear model for Equation 3, while the deep model is a feed-forward neural network with three hidden layers and 32, 16, and eight nodes, respectively. We use data from the Open Source Asset Pricing Dataset from January 1971 to December 2020. The columns labeled "EW", "VW", "PPP" and "DPPP" show the statistics of the equal-weighted portfolio, value-weighted portfolio, parametric portfolio policy, and deep parametric portfolio policy, respectively. The first row shows the utility of the investor. The second set of rows shows statistics on portfolio weights averaged over time. These statistics include the average absolute portfolio weight, the average maximum and minimum portfolio weights, the average sum of negative weights in the portfolio, the average proportion of negative weights in the portfolio, and the turnover in the portfolio. The third set of rows shows the first four moments of the final portfolio return distributions as well as the Sharpe ratios. The bottom panel shows the alphas and their standard errors with respect to the Fama-French five-factor model extended to include the momentum factor.

Table 3: Constrained and Penalized deep and linear portfolio policy

	EW	VW	PPP	DPPP
Utility	0.0021	0.0028	0.0139	0.0169
$\|w_i\| * 100$	0.0694	0.0694	0.1749	0.1819
$max\ w_i * 100$	0.0704	0.1113	0.6827	0.7866
$min\ w_i * 100$	0.0704	0.0410	-0.6817	-0.9814
$\sum w_i I(w_i < 0)$	0.0000	0.0000	-0.7607	-0.8113
$\sum w_i I(w_i < 0) / N_t$	0.0000	0.0000	0.3417	0.3181
$\sum \|w_{i,t} - w^{+}_{i,t-1}\|$	0.0931	0.0779	0.9699	1.6756
Mean	0.0110	0.0105	0.0202	0.0254
StdDev	0.0587	0.0552	0.0424	0.0469
Skew	-0.3716	-0.5039	-0.8764	-0.7162
Kurt	3.6591	3.3455	2.5245	2.7795
SR	0.1865	0.1908	0.4766	0.5412
$FF5 + Mom\ \alpha$	-0.0002	-0.0003	0.0093	0.0142
$StdErr(\alpha)$	0.0007	0.0006	0.0013	0.0017

This table shows out-of-sample estimates of the deep and linear portfolio policies with the transaction costs penalty in Equation 11 and leverage constraint in Equation 12 with 157 firm characteristics as specified in Equation 1 and optimized for a mean-variance investor with absolute risk aversion of five. The regular portfolio policy is a linear model for Equation 3, while the deep model is a feed-forward neural network with three hidden layers and 32, 16, and eight nodes, respectively. We use data from the Open Source Asset Pricing Dataset from January 1971 to December 2020. The columns labeled "EW", "VW", "PPP" and "DPPP" show the statistics of the equal-weighted portfolio, value-weighted portfolio, parametric portfolio policy, and deep parametric portfolio policy, respectively. The first row shows the utility of the investor. The second set of rows shows statistics on portfolio weights averaged over time. These statistics include the average absolute portfolio weight, the average maximum and minimum portfolio weights, the average sum of negative weights in the portfolio, the average proportion of negative weights in the portfolio, and the turnover in the portfolio. The third set of rows shows the first four moments of the final portfolio return distributions as well as the Sharpe ratios. The bottom panel shows the alphas and their standard errors with respect to the Fama-French five-factor model extended to include the momentum factor.

Table 4: Deep portfolio policy for mean-variance investors with different degrees of risk aversion

	$\gamma = 2$	$\gamma = 5$	$\gamma = 10$	$\gamma = 20$
% Utility Increase	780.4002	1885.2435	565.3362	122.6475
$\lvert w_i \rvert * 100$	0.6749	0.6057	0.5295	0.3847
$max\ w_i * 100$	1.8125	1.7260	1.6331	1.2971
$min\ w_i * 100$	-1.8523	-1.8370	-1.8039	-1.3872
$\sum w_i I(w_i < 0)$	-4.3656	-3.8665	-3.3171	-2.2737
$\sum w_i I(w_i < 0)/N_t$	0.4451	0.4411	0.4344	0.4171
$\sum \lvert w_{i,t} - w_{i,t-1}^{+} \rvert$	8.5704	7.6984	6.7283	4.8273
Mean	0.0786	0.0701	0.0628	0.0482
StdDev	0.1115	0.0965	0.0824	0.0656
Skew	1.3035	1.0537	0.3598	0.5061
Kurt	8.2253	6.5084	0.9416	1.3940
SR	0.7046	0.7266	0.7621	0.7345
$FF5 + Mom\ \alpha$	0.0626	0.0559	0.0492	0.0368
$StdErr(\alpha)$	0.0058	0.0051	0.0043	0.0033

This table shows out-of-sample estimates of the deep portfolio policies with 157 firm characteristics as specified in Equation 1 and optimized for a mean-variance investor with absolute risk aversion of two, five, ten and 20, respectively. The deep model is a feed-forward neural network with three hidden layers and 32, 16, and eight nodes, respectively. We use data from the Open Source Asset Pricing Dataset from January 1971 to December 2020. The columns labeled "$\gamma = 2$", "$\gamma = 5$", "$\gamma = 10$" and "$\gamma = 20$" show the statistics of the deep parametric portfolio policy with risk aversion of two, five, ten and 20, respectively. The first row shows the difference in utility relative to an equally weighted portfolio. The second set of rows shows statistics on portfolio weights averaged over time. These statistics include the average absolute portfolio weight, the average maximum and minimum portfolio weights, the average sum of negative weights in the portfolio, the average proportion of negative weights in the portfolio, and the turnover in the portfolio. The third set of rows shows the first four moments of the final portfolio return distributions as well as the Sharpe ratios. The bottom panel shows the alphas and their standard errors with respect to the Fama-French five-factor model extended to include the momentum factor.

Table 5: Deep portfolio policy with different investor preferences

	CRRA		LA	
	PPP	DPPP	PPP	DPPP
Utility	-0.2253	-0.2063	0.0266	0.0574
$\lvert w_i \rvert * 100$	0.4972	0.6127	0.5034	0.6468
$max\ w_i * 100$	2.0363	1.7452	2.0743	1.7618
$min\ w_i * 100$	-2.1712	-1.8709	-2.1577	-1.7841
$\sum w_i I(w_i < 0)$	-3.0841	-3.9171	-3.1290	-4.1627
$\sum w_i I(w_i < 0) / N_t$	0.4351	0.4430	0.4307	0.4490
$\sum \lvert w_{i,t} - w^+_{i,t-1} \rvert$	3.7816	7.8053	3.7464	8.3677
Mean	0.0473	0.0711	0.0473	0.0783
StdDev	0.0890	0.0982	0.0871	0.1359
Skew	-0.1004	0.8169	0.0996	3.5153
Kurt	1.3766	4.9609	0.8451	33.2542
SR	0.5309	0.7246	0.5424	0.5763
$FF5 + Mom\ \alpha$	0.0324	0.0570	0.0338	0.0624
$StdErr(\alpha)$	0.0040	0.0052	0.0040	0.0067

This table shows out-of-sample estimates of the deep and linear portfolio policies with 157 firm characteristics as specified in Equation 1 and optimized for an investor with constant relative risk aversion preference (CRRA) with relative risk aversion of five and a loss averse (LA) investor with loss aversion of 2.5, subjective wealth level of one and degree of risk seeking of one, respectively. The regular portfolio policy is a linear model for Equation 3, while the deep model is a feed-forward neural network with three hidden layers and 32, 16, and eight nodes, respectively. We use data from the Open Source Asset Pricing Dataset from January 1971 to December 2020. The columns labeled "PPP" and "DPPP" show the statistics of the parametric portfolio policy, and deep parametric portfolio policy, respectively. The first row shows the utility of the investor. The second set of rows shows statistics on portfolio weights averaged over time. These statistics include the average absolute portfolio weight, the average maximum and minimum portfolio weights, the average sum of negative weights in the portfolio, the average proportion of negative weights in the portfolio, and the turnover in the portfolio. The third set of rows shows the first four moments of the final portfolio return distributions as well as the Sharpe ratios. The bottom panel shows the alphas and their standard errors with respect to the Fama-French five-factor model extended to include the momentum factor.

Description of appendices

- Appendix A: Changing the Network Architecture
- Appendix B: Supplementary figures
- Appendix C: Supplementary tables

Appendix A Changing the Network Architecture

A.1 Model complexity

Our benchmark model is a relatively shallow neural net with only three hidden layers. It is conceivable that a more complex model can achieve even higher utility gains over a linear model. For example, Goodfellow et al. (2016) observe that neural nets with more hidden layers tend to outperform neural nets with fewer hidden layers but more nodes per layer. Kelly et al. (2022) report evidence in support of complex models in the context of forecasting aggregate stock market returns.

We extend our benchmark model to include between two and five hidden layers. All models start with 32 nodes in the first hidden layer and then halve the number of nodes in each subsequent layer. The number of parameters across models therefore varies between 5,600 and 5,768. Additionally, we add different possible learning rates to our hyperparameter tuning and increase the number of epochs and patience for early stopping, to account for the different complexities of the models and to ensure that more complex models also reach their respective potential.

Table C.2 shows results. The second model is our original benchmark model that we added for comparison.[15] The remaining columns contain results based on networks with two, four or five hidden layers. Overall, utility increases with more complex models but the increases are relatively modest. Interestingly, more complex models lead to lower turnover. Expected returns are roughly unchanged (or slightly lower for more complex models), whereas the return's standard deviation decreases. This suggests that improvements in mean-variance utility for more complex models are driven by decreases in variance rather than by increases in mean returns.

[TABLE C.2 ABOUT HERE]

A.2 Non-fully connected networks

Theoretically, there is a large range of different options to how one may adjust the network structure. In this section, we explore one structural change. Following Bianchi et al. (2020), we

[15] Note that the utility slightly differs from our benchmark in Section 3.1. This is due to the aforementioned fact that we add different possible learning rates as well as increase the number of epochs and patience for early stopping. We do so not only for the model variations, but also for our benchmark to ensure consistency across models.

split our input according to its characteristics and feed the resulting input groups separately into the model. This is illustrated in Figure B.4.

[Figure B.4 ABOUT HERE]

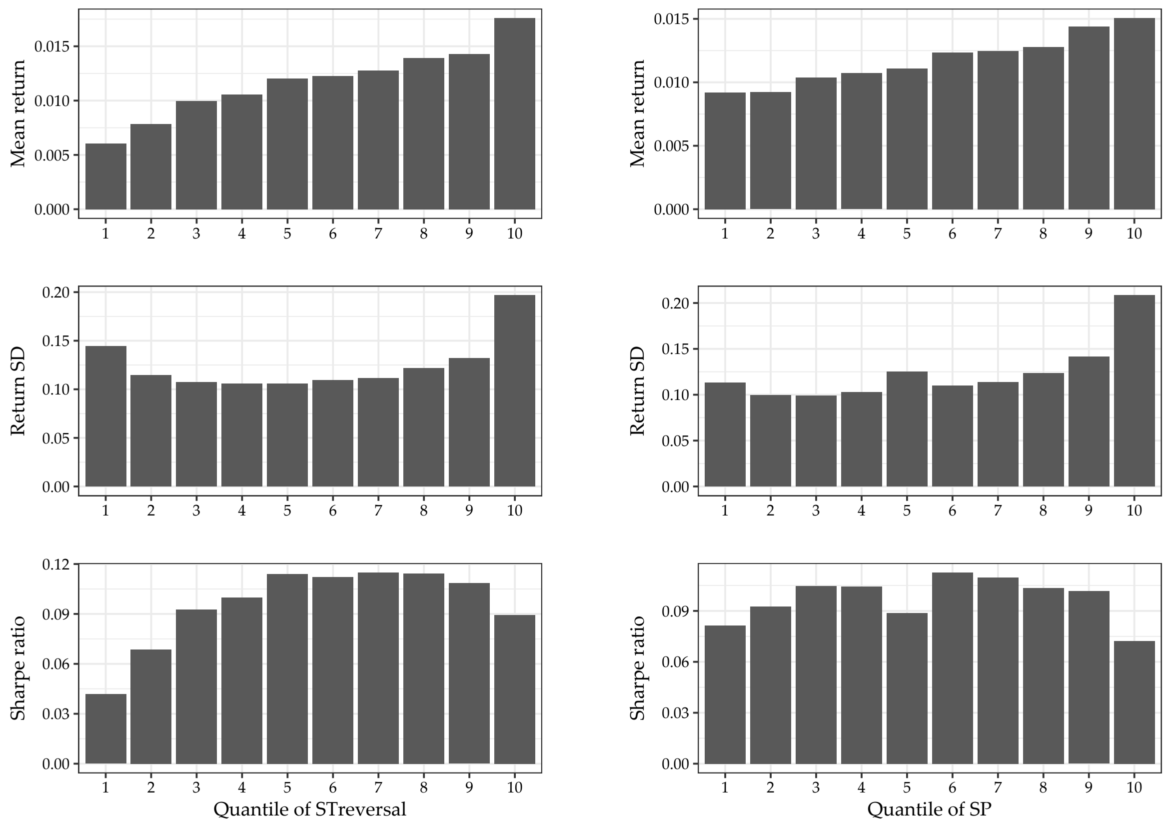
More specifically, we split our data according to its update frequency and its data category, respectively. For update frequency we divide our data into monthly, yearly and quarterly characteristics. For data category we divide our data into Accounting, Price, Trading and Analyst characteristics. The update frequency and data category of each predictor is shown in Table C.1 in the appendix.

We interact only characteristics with the same frequency (category) in the first hidden layer which can be interpreted as a dimension reduction for each frequency (category). After that we proceed with the ordinary network architecture in the second and third hidden layer. These are just two different network structure variations out of the plethora of different possibilities.

Table C.3 shows the results for the benchmark linear and deep portfolio policy followed by the two different architectures for the deep portfolio policy. The results indicate that the different architectures produces similar but slightly better results. Both new models produce slightly higher leverage and turnover than the base deep portfolio policy. Moreover, the new models yield higher Sharpe ratios by reducing the variance of the portfolio return distributions. The highest deviation can be observed in the third and fourth moment of the return distribution, where both new models show less extreme skewness and kurtosis which results in more realistic return distributions.

[TABLE C.3 ABOUT HERE]

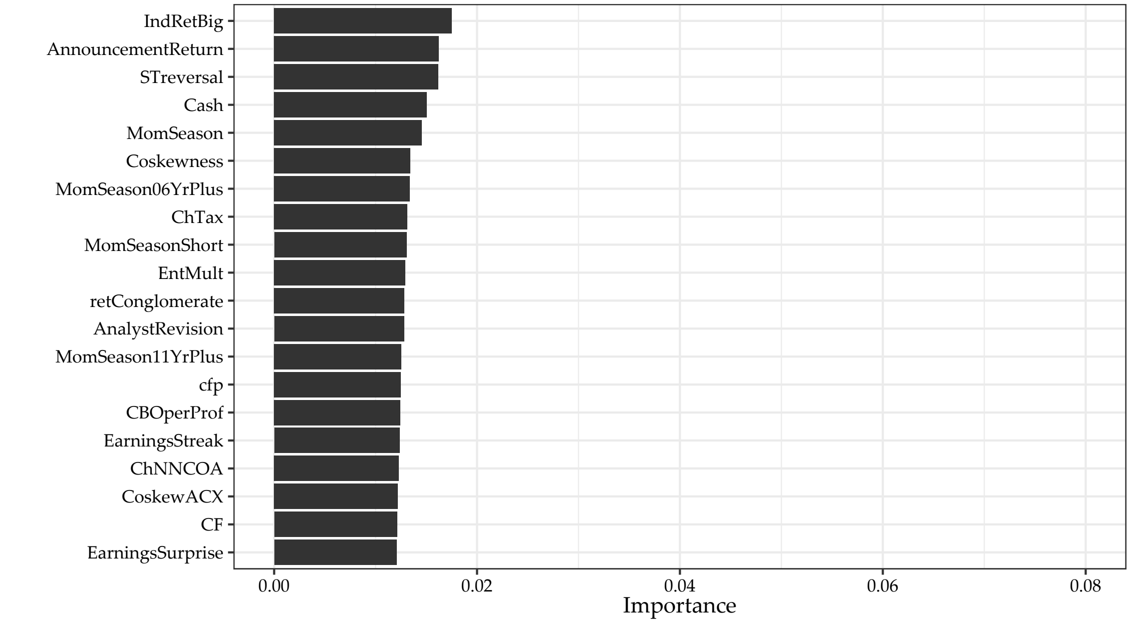
Appendix B Supplementary Figures



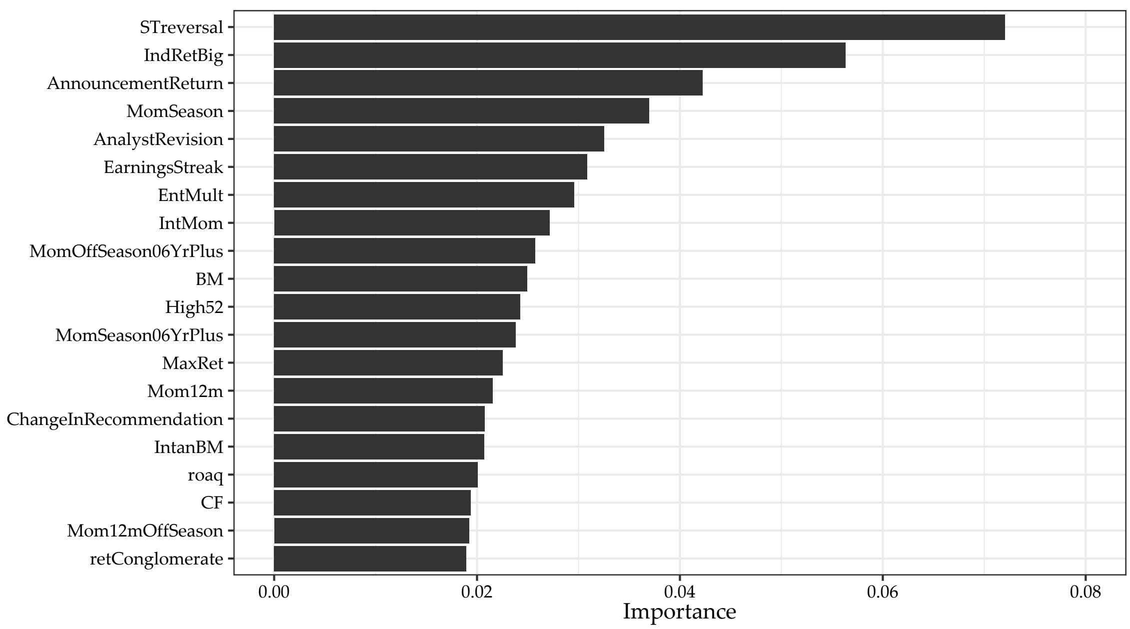
(a) Short-Term Reversal

(b) Sales-To-Price

Figure B.1: Mean returns, standard deviations and Sharpe ratios of one-dimensional portfolio sorts
Mean returns, standard deviations and Sharpe ratios of decile portfolios sorted on short-term reversal (left panel) and sales-to-price ratio (right panel).



(a) PPP



(b) DPPP

Figure B.2: Regression coefficients from a linear surrogate model for PPP and DPPP
Variable importance is measured as the coefficients of a linear surrogate model of the estimated weights on the input variables. The plot shows the variable importance for the 20 most influential variables in the linear and deep parametric portfolio policies. Variable importance is an average over all training samples and normalized so that the sum equals one.

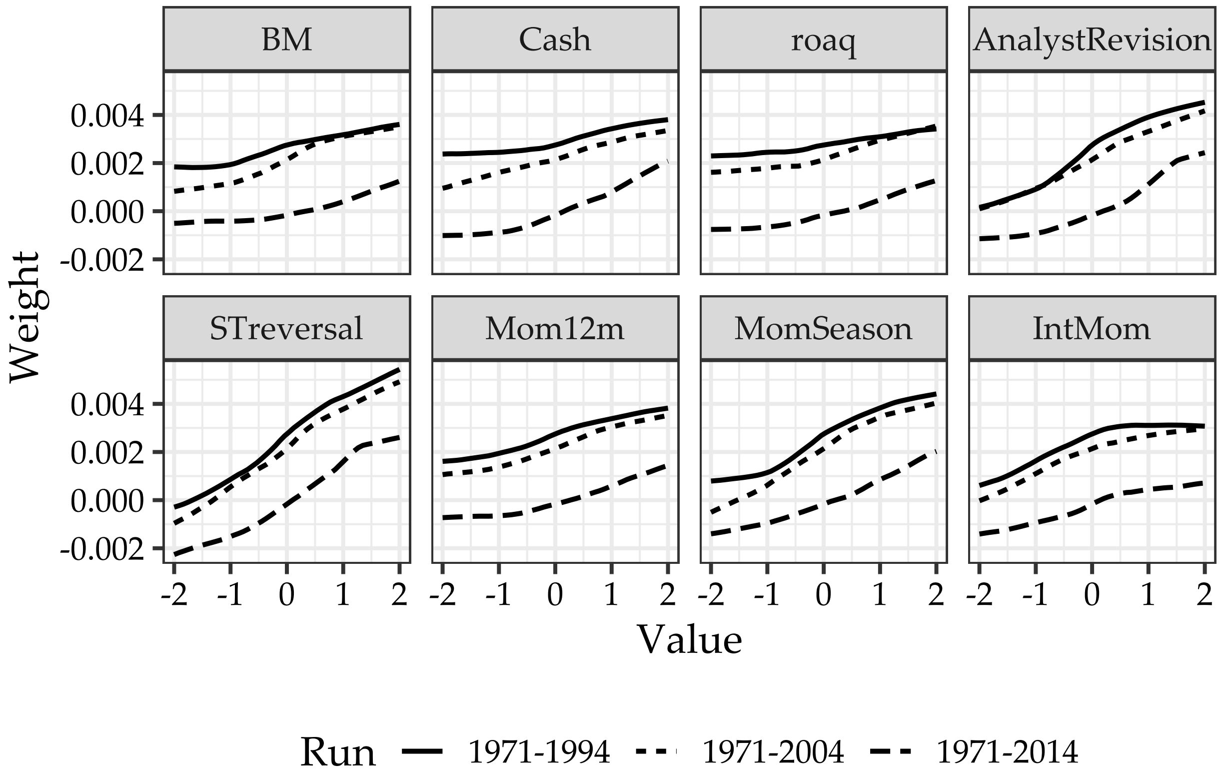


Figure B.3: Marginal association between portfolio weights and characteristics for different time periods
The panels show the sensitivity of mean portfolio weights (vertical axis) to the individual characteristics in different time periods, holding all other covariates fixed at their original values.

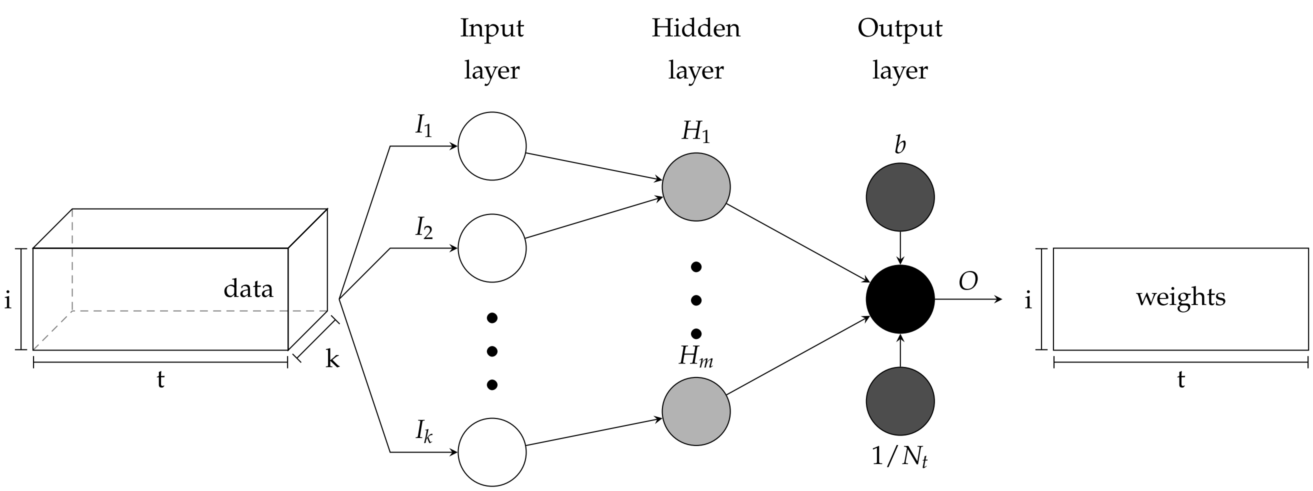


Figure B.4: Non-Fully Connected Neural Network Structure
This figure presents the structure of our non-fully connected networks. White circles denote the input layer, grey circles denote the hidden layer and black circles denote the output layer. The data cube on the left depicts the structure of our data, i.e. we have k variables across i cross-sections in t periods. The rectangle on the right depicts our output, i.e. weights across i cross-sections in t periods. The output of the neural network is normalized by $1/N_t$ and added to the benchmark portfolio b. The final output is labeled O.

Appendix C Supplementary Tables

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
ChInvIA	Change in capital inv (ind adj)	Abarbanell and Bushee	1998, AR	Accounting	yearly
GrSaleToGrInv	Sales growth over inventory growth	Abarbanell and Bushee	1998, AR	Accounting	yearly
GrSaleToGrOverhead	Sales growth over overhead growth	Abarbanell and Bushee	1998, AR	Accounting	yearly
IdioVolAHT	Idiosyncratic risk (AHT)	Ali, Hwang, and Trombley	2003, JFE	Price	monthly
EarningsConsistency	Earnings consistency	Alwathainani	2009, BAR	Accounting	yearly
Illiquidity	Amihud's illiquidity	Amihud	2002, JFM	Trading	monthly
BidAskSpread	Bid-ask spread	Amihud and Mendelsohn	1986, JFE	Trading	monthly
grcapx	Change in capex (two years)	Anderson and Garcia-Feijoo	2006, JF	Accounting	yearly
grcapx3y	Change in capex (three years)	Anderson and Garcia-Feijoo	2006, JF	Accounting	yearly
betaVIX	Systematic volatility	Ang et al.	2006, JF	Price	monthly
IdioRisk	Idiosyncratic risk	Ang et al.	2006, JF	Price	monthly
IdioVol3F	Idiosyncratic risk (3 factor)	Ang et al.	2006, JF	Price	monthly
CoskewACX	Coskewness using daily returns	Ang, Chen and Xing	2006, RFS	Price	monthly
Mom6mJunk	Junk Stock Momentum	Avramov et al	2007, JF	Price	monthly
OrderBacklogChg	Change in order backlog	Baik and Ahn	2007, Other	Accounting	yearly
roaq	Return on assets (qtrly)	Balakrishnan, Bartov and Faurel	2010, JAE	Accounting	quarterly
MaxRet	Maximum return over month	Bali, Cakici, and Whitelaw	2010, JF	Price	monthly
ReturnSkew	Return skewness	Bali, Engle and Murray	2015, Book	Price	monthly
ReturnSkew3F	Idiosyncratic skewness (3F model)	Bali, Engle and Murray	2015, Book	Price	monthly
CBOperProf	Cash-based operating profitability	Ball et al.	2016, JFE	Accounting	yearly
OperProfRD	Operating profitability R&D adjusted	Ball et al.	2016, JFE	Accounting	yearly
Size	Size	Banz	1981, JFE	Price	monthly

Continued on next page

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
SP	Sales-to-price	Barbee, Mukherji and Raines	1996, FAJ	Accounting	yearly
EP	Earnings-to-Price Ratio	Basu	1977, JF	Price	monthly
InvGrowth	Inventory Growth	Belo and Lin	2012, RFS	Accounting	yearly
BrandInvest	Brand capital investment	Belo, Lin and Vitorino	2014, RED	Accounting	yearly
Leverage	Market leverage	Bhandari	1988, JFE	Price	monthly
ResidualMomentum	Momentum based on FF3 residuals	Blitz, Huij and Martens	2011, JEmpFin	Price	monthly
Price	Price	Blume and Husic	1972, JF	Price	monthly
NetPayoutYield	Net Payout Yield	Boudoukh et al.	2007, JF	Price	monthly
PayoutYield	Payout Yield	Boudoukh et al.	2007, JF	Price	monthly
NetDebtFinance	Net debt financing	Bradshaw, Richardson, Sloan	2006, JAE	Accounting	yearly
NetEquityFinance	Net equity financing	Bradshaw, Richardson, Sloan	2006, JAE	Accounting	yearly
XFIN	Net external financing	Bradshaw, Richardson, Sloan	2006, JAE	Accounting	yearly
DolVol	Past trading volume	Brennan, Chordia, Subra	1998, JFE	Trading	monthly
FEPS	Analyst earnings per share	Cen, Wei, and Zhang	2006, WP	Analyst	monthly
AnnouncementReturn	Earnings announcement return	Chan, Jegadeesh and Lakonishok	1996, JF	Price	monthly
REV6	Earnings forecast revisions	Chan, Jegadeesh and Lakonishok	1996, JF	Analyst	monthly
AdExp	Advertising Expense	Chan, Lakonishok and Sougiannis	2001, JF	Accounting	monthly
RD	R&D over market cap	Chan, Lakonishok and Sougiannis	2001, JF	Accounting	monthly
CashProd	Cash Productivity	Chandrashekar and Rao	2009, WP	Accounting	yearly
std_turn	Share turnover volatility	Chordia, Subra, Anshuman	2001, JFE	Trading	monthly
VolSD	Volume Variance	Chordia, Subra, Anshuman	2001, JFE	Trading	monthly
retConglomerate	Conglomerate return	Cohen and Lou	2012, JFE	Price	monthly

Continued on next page

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
RDAbility	R&D ability	Cohen, Diether and Malloy	2013, RFS	Accounting	yearly
AssetGrowth	Asset growth	Cooper, Gulen and Schill	2008, JF	Accounting	yearly
EarningsForecastDisparity	Long-vs-short EPS forecasts	Da and Warachka	2011, JFE	Analyst	monthly
CompEquIss	Composite equity issuance	Daniel and Titman	2006, JF	Accounting	monthly
IntanBM	Intangible return using BM	Daniel and Titman	2006, JF	Accounting	yearly
IntanCFP	Intangible return using CFtoP	Daniel and Titman	2006, JF	Accounting	yearly
IntanEP	Intangible return using EP	Daniel and Titman	2006, JF	Accounting	yearly
IntanSP	Intangible return using Sale2P	Daniel and Titman	2006, JF	Accounting	yearly
ShareIss5Y	Share issuance (5 year)	Daniel and Titman	2006, JF	Accounting	monthly
LRreversal	Long-run reversal	De Bondt and Thaler	1985, JF	Price	monthly
MRreversal	Medium-run reversal	De Bondt and Thaler	1985, JF	Price	monthly
EquityDuration	Equity Duration	Dechow, Sloan and Soliman	2004, RAS	Price	yearly
cfp	Operating Cash flows to price	Desai, Rajgopal, Venkatachalam	2004, AR	Accounting	yearly
ForecastDispersion	EPS Forecast Dispersion	Diether, Malloy and Scherbina	2002, JF	Analyst	monthly
ExclExp	Excluded Expenses	Doyle, Lundholm and Soliman	2003, RAS	Analyst	quarterly
ProbInformedTrading	Probability of Informed Trading	Easley, Hvidkjaer and O'Hara	2002, JF	Trading	yearly
OrgCap	Organizational capital	Eisfeldt and Papanikolaou	2013, JF	Accounting	yearly
sfe	Earnings Forecast to price	Elgers, Lo and Pfeiffer	2001, AR	Analyst	monthly
GrLTNOA	Growth in long term operating assets	Fairfield, Whisenant and Yohn	2003, AR	Accounting	yearly
AM	Total assets to market	Fama and French	1992, JF	Accounting	yearly
BMdec	Book to market using December ME	Fama and French	1992, JPM	Accounting	yearly
BookLeverage	Book leverage (annual)	Fama and French	1992, JF	Accounting	yearly

Continued on next page

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
OperProf	operating profits / book equity	Fama and French	2006, JFE	Accounting	yearly
Beta	CAPM beta	Fama and MacBeth	1973, JPE	Price	monthly
EarningsSurprise	Earnings Surprise	Foster, Olsen and Shevlin	1984, AR	Analyst	quarterly
AnalystValue	Analyst Value	Frankel and Lee	1998, JAE	Analyst	monthly
AOP	Analyst Optimism	Frankel and Lee	1998, JAE	Analyst	monthly
PredictedFE	Predicted Analyst forecast error	Frankel and Lee	1998, JAE	Accounting	monthly
FR	Pension Funding Status	Franzoni and Marin	2006, JF	Accounting	monthly
BetaFP	Frazzini-Pedersen Beta	Frazzini and Pedersen	2014, JFE	Price	monthly
High52	52 week high	George and Hwang	2004, JF	Price	monthly
IndMom	Industry Momentum	Grinblatt and Moskowitz	1999, JFE	Price	monthly
PctAcc	Percent Operating Accruals	Hafzalla, Lundholm, Van Winkle	2011, AR	Accounting	yearly
PctTotAcc	Percent Total Accruals	Hafzalla, Lundholm, Van Winkle	2011, AR	Accounting	yearly
tang	Tangibility	Hahn and Lee	2009, JF	Accounting	yearly
Coskewness	Coskewness	Harvey and Siddique	2000, JF	Price	monthly
RoE	net income / book equity	Haugen and Baker	1996, JFE	Accounting	yearly
VarCF	Cash-flow to price variance	Haugen and Baker	1996, JFE	Accounting	monthly
VolMkt	Volume to market equity	Haugen and Baker	1996, JFE	Trading	monthly
VolumeTrend	Volume Trend	Haugen and Baker	1996, JFE	Trading	monthly
AnalystRevision	EPS forecast revision	Hawkins, Chamberlin, Daniel	1984, FAJ	Analyst	monthly
Mom12mOffSeason	Momentum without the seasonal part	Heston and Sadka	2008, JFE	Price	monthly
MomOffSeason	Off season long-term reversal	Heston and Sadka	2008, JFE	Price	monthly
MomOffSeason06YrPlus	Off season reversal years 6 to 10	Heston and Sadka	2008, JFE	Price	monthly

Continued on next page

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
MomOffSeason11YrPlus	Off season reversal years 11 to 15	Heston and Sadka	2008, JFE	Price	monthly
MomOffSeason16YrPlus	Off season reversal years 16 to 20	Heston and Sadka	2008, JFE	Price	monthly
MomSeason	Return seasonality years 2 to 5	Heston and Sadka	2008, JFE	Price	monthly
MomSeason06YrPlus	Return seasonality years 6 to 10	Heston and Sadka	2008, JFE	Price	monthly
MomSeason11YrPlus	Return seasonality years 11 to 15	Heston and Sadka	2008, JFE	Price	monthly
MomSeason16YrPlus	Return seasonality years 16 to 20	Heston and Sadka	2008, JFE	Price	monthly
MomSeasonShort	Return seasonality last year	Heston and Sadka	2008, JFE	Price	monthly
NOA	Net Operating Assets	Hirshleifer et al.	2004, JAE	Accounting	yearly
dNoa	change in net operating assets	Hirshleifer, Hou, Teoh, Zhang	2004, JAE	Accounting	yearly
EarnSupBig	Earnings surprise of big firms	Hou	2007, RFS	Accounting	quarterly
IndRetBig	Industry return of big firms	Hou	2007, RFS	Price	monthly
PriceDelayRsq	Price delay r square	Hou and Moskowitz	2005, RFS	Price	monthly
PriceDelaySlope	Price delay coeff	Hou and Moskowitz	2005, RFS	Price	monthly
PriceDelayTstat	Price delay SE adjusted	Hou and Moskowitz	2005, RFS	Price	monthly
STreversal	Short term reversal	Jegadeesh	1989, JF	Price	monthly
RevenueSurprise	Revenue Surprise	Jegadeesh and Livnat	2006, JFE	Accounting	quarterly
Mom12m	Momentum (12 month)	Jegadeesh and Titman	1993, JF	Price	monthly
Mom6m	Momentum (6 month)	Jegadeesh and Titman	1993, JF	Price	monthly
ChangeInRecommendation	Change in recommendation	Jegadeesh et al.	2004, JF	Analyst	monthly
OptionVolume1	Option to stock volume	Johnson and So	2012, JFE	Trading	monthly
OptionVolume2	Option volume to average	Johnson and So	2012, JFE	Trading	monthly
BetaTailRisk	Tail risk beta	Kelly and Jiang	2014, RFS	Price	monthly

Continued on next page

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
fgr5yrLag	Long-term EPS forecast	La Porta	1996, JF	Analyst	monthly
CF	Cash flow to market	Lakonishok, Shleifer, Vishny	1994, JF	Accounting	monthly
MeanRankRevGrowth	Revenue Growth Rank	Lakonishok, Shleifer, Vishny	1994, JF	Accounting	yearly
RDS	Real dirty surplus	Landsman et al.	2011, AR	Accounting	yearly
Tax	Taxable income to income	Lev and Nissim	2004, AR	Accounting	yearly
RDcap	R&D capital-to-assets	Li	2011, RFS	Accounting	yearly
zerotrade	Days with zero trades	Liu	2006, JFE	Trading	monthly
zerotradeAlt1	Days with zero trades	Liu	2006, JFE	Trading	monthly
zerotradeAlt12	Days with zero trades	Liu	2006, JFE	Trading	monthly
ChEQ	Growth in book equity	Lockwood and Prombutr	2010, JFR	Accounting	yearly
EarningsStreak	Earnings surprise streak	Loh and Warachka	2012, MS	Accounting	monthly
NumEarnIncrease	Earnings streak length	Loh and Warachka	2012, MS	Accounting	quarterly
GrAdExp	Growth in advertising expenses	Lou	2014, RFS	Accounting	yearly
EntMult	Enterprise Multiple	Loughran and Wellman	2011, JFQA	Accounting	monthly
CompositeDebtIssuance	Composite debt issuance	Lyandres, Sun and Zhang	2008, RFS	Accounting	yearly
InvestPPEInv	change in ppe and inv/assets	Lyandres, Sun and Zhang	2008, RFS	Accounting	yearly
Frontier	Efficient frontier index	Nguyen and Swanson	2009, JFQA	Accounting	yearly
GP	gross profits / total assets	Novy-Marx	2013, JFE	Accounting	yearly
IntMom	Intermediate Momentum	Novy-Marx	2012, JFE	Price	monthly
OPLeverage	Operating leverage	Novy-Marx	2010, ROF	Accounting	yearly
Cash	Cash to assets	Palazzo	2012, JFE	Accounting	quarterly
BetaLiquidityPS	Pastor-Stambaugh liquidity beta	Pastor and Stambaugh	2003, JPE	Price	monthly

Continued on next page

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
BPEBM	Leverage component of BM	Penman, Richardson and Tuna	2007, JAR	Accounting	monthly
EBM	Enterprise component of BM	Penman, Richardson and Tuna	2007, JAR	Accounting	monthly
NetDebtPrice	Net debt to price	Penman, Richardson and Tuna	2007, JAR	Accounting	monthly
PS	Piotroski F-score	Piotroski	2000, AR	Accounting	yearly
ShareIss1Y	Share issuance (1 year)	Pontiff and Woodgate	2008, JF	Accounting	monthly
DelDRC	Deferred Revenue	Prakash and Sinha	2012, CAR	Accounting	yearly
OrderBacklog	Order backlog	Rajgopal, Shevlin, Venkatachalam	2003, RAS	Accounting	yearly
DelCOA	Change in current operating assets	Richardson et al.	2005, JAE	Accounting	yearly
DelCOL	Change in current operating liabilities	Richardson et al.	2005, JAE	Accounting	yearly
DelEqu	Change in equity to assets	Richardson et al.	2005, JAE	Accounting	yearly
DelFINL	Change in financial liabilities	Richardson et al.	2005, JAE	Accounting	yearly
DelLTI	Change in long-term investment	Richardson et al.	2005, JAE	Accounting	yearly
DelNetFin	Change in net financial assets	Richardson et al.	2005, JAE	Accounting	yearly
TotalAccruals	Total accruals	Richardson et al.	2005, JAE	Accounting	yearly
BM	Book to market using most recent ME	Rosenberg, Reid, and Lanstein	1985, JF	Accounting	monthly
Accruals	Accruals	Sloan	1996, AR	Accounting	yearly
ChAssetTurnover	Change in Asset Turnover	Soliman	2008, AR	Accounting	yearly
ChNNCOA	Change in Net Noncurrent Op Assets	Soliman	2008, AR	Accounting	yearly
ChNWC	Change in Net Working Capital	Soliman	2008, AR	Accounting	yearly
ChInv	Inventory Growth	Thomas and Zhang	2002, RAS	Accounting	yearly
ChTax	Change in Taxes	Thomas and Zhang	2011, JAR	Accounting	quarterly
Investment	Investment to revenue	Titman, Wei and Xie	2004, JFQA	Accounting	yearly

Continued on next page

Table C.1: Included characteristics

Acronym	Long Description	Author(s)	Year, Journal	Category	Frequency
realestate	Real estate holdings	Tuzel	2010, RFS	Accounting	yearly
AbnormalAccruals	Abnormal Accruals	Xie	2001, AR	Accounting	yearly
FirmAgeMom	Firm Age - Momentum	Zhang	2004, JF	Price	monthly

The table displays all available characteristics from the data set marked as clear, the author(s), year as well as journal of publication.

Table C.2: Deep portfolio policy with different number of hidden layers

	Layer 2	Layer 3	Layer 4	Layer 5
Utility	0.0510	0.0559	0.0548	0.0567
$\|w_i\| * 100$	1.2532	1.1636	1.1648	0.9059
$max\ w_i * 100$	2.3479	2.1713	2.3620	2.2910
$min\ w_i * 100$	-2.2734	-2.1431	-2.3701	-2.3172
$\sum w_i I(w_i < 0)$	-8.5341	-7.8884	-7.8970	-6.0309
$\sum w_i I(w_i < 0)/N_t$	0.4838	0.4751	0.4704	0.4607
$\sum \|w_{i,t} - w^{+}_{i,t-1}\|$	15.4961	14.1319	14.5830	11.8094
Mean	0.1111	0.1042	0.1159	0.1037
StdDev	0.1553	0.1392	0.1566	0.1374
Skew	0.2862	0.3584	0.6250	0.6940
Kurt	1.6373	1.0855	1.9909	1.4173
SR	0.7155	0.7486	0.7400	0.7550
$FF5 + Mom\ \alpha$	0.0933	0.0863	0.0988	0.0885
$StdErr(\alpha)$	0.0085	0.0076	0.0088	0.0077

This table shows out-of-sample estimates of the deep portfolio policies with different number of hidden layers with 157 firm characteristics as specified in Equation 1 and optimized for a mean-variance investor with absolute risk aversion of five. The deep models are feed-forward neural networks with two (32, 16), three (32, 16, 8), four (32, 16, 8, 4) and five (32, 16, 8, 4, 2) hidden layers (nodes), respectively. We use data from the Open Source Asset Pricing Dataset from January 1971 to December 2020. The columns labeled "Layer 2", "Layer 3", "Layer 4" and "Layer 5" show the statistics of the deep parametric portfolio policy with two, three, four and five hidden layers, respectively. The first row shows the utility of the investor. The second set of rows shows statistics on portfolio weights averaged over time. These statistics include the average absolute portfolio weight, the average maximum and minimum portfolio weights, the average sum of negative weights in the portfolio, the average proportion of negative weights in the portfolio, and the turnover in the portfolio. The third set of rows shows the first four moments of the final portfolio return distributions as well as the Sharpe ratios. The bottom panel shows the alphas and their standard errors with respect to the Fama-French five-factor model extended to include the momentum factor.

Table C.3: Deep portfolio policy with different network architectures

	PPP	DPPP	Frequency	Category
Utility	0.0267	0.0469	0.0499	0.0473
$\lvert w_i \rvert * 100$	0.5060	0.6057	0.6360	0.6355
$max\ w_i * 100$	2.0748	1.7260	1.7891	1.7235
$min\ w_i * 100$	-2.2097	-1.8370	-1.8926	-1.8203
$\sum w_i I(w_i < 0)$	-3.1475	-3.8665	-4.0847	-4.0813
$\sum w_i I(w_i < 0) / N_t$	0.4334	0.4411	0.4471	0.4478
$\sum \lvert w_{i,t} - w_{i,t-1}^{+} \rvert$	3.9370	7.6984	8.3253	8.2927
Mean	0.0468	0.0701	0.0699	0.0670
StdDev	0.0897	0.0965	0.0898	0.0889
Skew	-0.1451	1.0537	0.2517	0.2430
Kurt	1.8391	6.5084	1.7880	2.2713
SR	0.5216	0.7266	0.7790	0.7535
$FF5 + Mom\ \alpha$	0.0323	0.0559	0.0564	0.0526
$StdErr(\alpha)$	0.0040	0.0051	0.0048	0.0046

This table shows out-of-sample estimates of three deep portfolio policies and one linear portfolio policy with 157 firm characteristics as specified in Equation 1 and optimized for a mean-variance investor with absolute risk aversion of five. The regular portfolio policy is a linear model for Equation 3, while the deep model is a feed-forward neural network with three hidden layers and 32, 16, and eight nodes, respectively. We use data from the Open Source Asset Pricing Dataset from January 1971 to December 2020. The columns labeled "PPP", "DPPP", "Frequency" and "Category" show the statistics of the linear portfolio policy, deep portfolio policy, deep portfolio policy with variables grouped by frequency, and deep portfolio policy with variables grouped by category, respectively. The last to columns refer to different network architectures where the variables are only interacted with variables of their own group in the first hidden layer. The first row shows the utility of the investor. The second set of rows shows statistics on portfolio weights averaged over time. These statistics include the average absolute portfolio weight, the average maximum and minimum portfolio weights, the average sum of negative weights in the portfolio, the average proportion of negative weights in the portfolio, and the turnover in the portfolio. The third set of rows shows the first four moments of the final portfolio return distributions as well as the Sharpe ratios. The bottom panel shows the alphas and their standard errors with respect to the Fama-French five-factor model extended to include the momentum factor.