CFR Working Paper No. 23-01

# Deep Parametric Portfolio Policies

F. Simon • S. Weibels • T. Zimmermann

centre for financial research
cologne

# Deep Parametric Portfolio Policies*

Frederik Simon† Sebastian Weibels‡ Tom Zimmermann§

First Version: July 2022; This Version: February 2023

**Abstract**

We directly optimize portfolio weights as a function of firm characteristics 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 show that network-based policies better capture the non-linear relationship between investor utility and firm characteristics. Improvements can be traced to both variable interactions and non-linearity in functional form. Both the linear and the network-based approach 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

*We thank Bryan Kelly, Victor DeMiguel, Christian Fieberg, Alexander Klos, Simon Rottke, Fabricius Somogyi (discussant), Bastidon Cécile (discussant) and participants at the Research in Behavioral Finance Conference (RBFC), the Cardiff Fintech Conference, the 2022 New Zealand Finance Meeting (NZFM) as well as the Paris Financial Management Conference (PFMC) for helpful comments and suggestions.

†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 (PPP) (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* (DPPP) is well-suited to accommodate flexible non-linear and interactive relationships between portfolio weights and stock characteristics, to integrate different utility functions, to 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 can be attributed to the fact that the relationship between firm characteristics and investor utility is non-linear. Second, in our benchmark model, past return-based stock characteristics turn out to be more relevant to the portfolio policy than accounting-based characteristics. However, in line with extant literature (DeMiguel et al., 2020; Jensen et al., 2022), the relevance of return-based characteristics decreases when we model transaction costs explicitly in the objective function. 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 also 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, we show that both non-linearity in variables (i.e. variable interactions) and non-linearity in functional form account for the 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. More specifically, we allow portfolio weights to be of one of the arguably most flexible forms - a neural network. This represents a significant conceptual deviation from linear parametric portfolio policies 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; Freyberger et al., 2020; Gu et al., 2020). It is conceivable that such flexibility will also help to model the relation between portfolio weights and firm characteristics. Second, this 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.

Building on Brandt et al. (2009), we begin 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 or 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 imply higher turnover, such as e.g. short-term reversal.

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 relative to a benchmark portfolio 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 constant relative risk aversion or 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 in parameters and 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 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. 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. The use of machine learning algorithms to estimate coefficients of structural models (in our case portfolio weights) as flexible functions has also been proposed recently by Farrell et al. (2021).

Second, we extend the literature on one-step portfolio optimization. Specifically, we extend the parametric portfolio approach by Brandt et al. (2009). While Brandt et al. (2009) argue 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. Alternative, (machine learning-based) one-step portfolio optimization approaches include Cong et al. (2021), Butler and Kwon (2021), Uysal et al. (2021), Chevalier et al. (2022) and Jensen et al. (2022). Each of these differs from ours in one or more aspects. First and foremost, in contrast to any of these we generalize the approach of Brandt et al. (2009) and explicitly analyze differences between a linear and non-linear specification. In addition, Cong et al. (2021) use a general reinforcement learning - based approach and sort stocks into portfolios to maximize Sharpe ratios while our feed-forward network directly optimizes continuous portfolio weights for various investor utility functions. Butler and Kwon (2021) show that it is possible to integrate regression-based return predictions into the portfolio optimization by means of a two-layer neural network, one layer resembling the return prediction and one layer resembling the weight optimization. However, their results are restricted to a mean-variance setting, while our approach is flexibly applicable to any type of investor preference. Moreover, our empirical analysis is about modeling portfolios of stocks based on stock characteristics, whereas they empirically assess their models on simulated data and commodity future markets. Chevalier et al. (2022) derive optimal in-sample weights based on investor preferences and subsequently predict these weights conditional on covariates. This is conceptually different from our approach, primarily because we do not require the preprocessing step of computing the optimal in-sample weights. Jensen et al. (2022) take a different angle. They aim to specifically tackle the issue of integrating transaction costs into mean-variance portfolio optimization with machine learning. They outline different approaches to do so, inter alia, a ML-based one-step approach. However, rather than extending the approach by Brandt et al. (2009) as we do, they derive a closed form solution to the issue and implement it empirically by using random feature regressions. Moreover, their focus in terms of interpreting the empirical relations lays on comparing different approaches to achieve the aforementioned underlying aim of integrating transaction costs, as well as the importance of features in this setting. We, in contrast, also shed light onto how non-linearities contribute to portfolio optimization.

Finally, we relate to the literature that examines which firm characteristics are jointly significant in explaining expected returns (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), DeMiguel et al. (2020) and Jensen et al. (2022) being important exceptions) or weight constraints, whereas we show how our model allows us to 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 $\theta$ is the coefficient vector to be estimated.

The parameter vector $\theta$ remains constant across assets $i$ and periods $t$, i.e. it maximizes the conditional expected utility at every period $t$. This necessarily implies that $\theta$ also maximizes the unconditional expected utility. Hence, one can estimate $\theta$ by maximizing the unconditional expected utility via the return distribution's sample analogues:

$$\max_{\theta} \frac{1}{T} \sum_{t=1}^{T} u\left(r_{p,t+1}(\theta)\right) = \frac{1}{T} \sum_{t=1}^{T} u\left(\sum_{i=1}^{N_t} f(x_{i,t}; \theta) r_{i,t+1}\right). \tag{3}$$

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

$$w_{i,t} = b_{i,t} + \frac{1}{N_t} g(x_{i,t}; \theta), \tag{4}$$

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]

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

$$w_{i,t} = b_{i,t} + \frac{1}{N_t}\theta^T \hat{x}_{i,t}. \tag{5}$$

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 (4) to take arguably one of the most flexible forms - a feed-forward neural network. As discussed in the introduction, this represents a significant conceptual devivation from the literature in at least 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, in line with the recent literature that finds that such flexiblity can be important to predict expected return (Moritz and Zimmermann, 2016; Freyberger et al., 2020; Gu et al., 2020). Here, our approach explores whether such flexibility also helps to model the relationship between *portfolio weights* and firm characteristics. 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).

Why might non-linear modeling of portfolio weights be important? Consider an investor who trades off mean return against return volatility.The investor uses standard one-dimensional portfolio sorting techniques as pictured in Figure C.1 in Appendix C. Decile portfolios formed on short-term reversal or sales-to-price display monotonically increasing mean return.[2] At the same time, the standard deviations of decile portfolios are non-linear in deciles, in particular

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

[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).

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 potentially 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 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 by maximizing the unconditional sample analogue of a utility function as given in equation (3). For example, in our base case, the loss function $\mathcal{L}$ that we aim to minimize with respect to $\theta$ 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 $\gamma$ is the absolute risk aversion parameter. Note that minimizing Equation (6) is equivalent to maximizing mean-variance utility.

Second, our loss function requires the portfolio return per period $t$, so that we need to aggregate our outputs cross-sectionally in each period. 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] In line with equation (4), 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 achieve 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 issues in regards to the optimization process. Selecting the best network structure is a formidable task and not our main objective.[4] Instead, we rely on the results of Gu et al. (2020) and use their most successful model as our benchmark model. We explore robustness of our findings to changes in both network complexity and network structure in Appendix B.

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 to the equally weighted benchmark because of empirical evidence that it outperforms other benchmarks like the value weighted benchmark for longer periods (DeMiguel et al., 2009).

Lastly, we control for unreasonable results and overfitting in terms of portfolio weights by

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

ex-ante imposing an upper bound on an individual stock's absolute portfolio weight of $|3\%|$, i.e.

$$|w_{i,t}| \leq 0.03. \tag{7}$$

In doing so, we ensure that the model performance does not rely too heavily on particular stocks. We employ a range of different additional regularization techniques that are standard in the deep learning literature. We give an outline of these techniques and a more detailed description of the structure of the model including its hyperparameters in Appendix A.

### 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.[5]

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

[5] 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.

contains a minimum of 1,213, a maximum of 1,855 and an average of 1,422 firms. Table D.1 in the appendix lists the included predictors by original paper. The three columns in the table describe the update frequency of each predictor, the predictor category and the economic category, both taken from Chen and Zimmermann (2022). As part of our robustness check, we exploit that information in Appendix B to construct non-fully connected networks.

### 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 out-of-sample periods corresponding to each expanding window, in our case 25 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. Nevertheless, in our application, understanding the estimated relation between input (firm characteristics) and output (estimated portfolio weights) is essential in order to shed light on the relation between firm characteristics and utility. Moreover, such an understanding allows us to compare our results to the existing literature. We provide three ways of interpreting the models and of identifying the most important predictors among the plethora of variables that enter our models.

First, we evaluate the extent to which non-linearity in variables (i.e. variable interactions) and non-linearity in parameters (i.e. functional form) contribute to the estimated deep parametric portfolio policy. Put differently, we assess the extent to which different forms of non-linearity play a role when optimizing portfolios conditional on firm characteristics. To do so, we estimate a linear surrogate model in which we regress the out-of-sample weight predictions on all firm characteristics. This allows us to assess the extent to which a simple linear model is capable of ex-post explaining the predicted weights. In a next step, we estimate a second surrogate model, this time including all possible two-way interactions, i.e. allowing for non-linearity in variables. To prevent excessive overfitting, we also add a lasso penalty term. This allows us to assess to which extent non-linearity in variables play a role in regards to predicting weights. We attribute the remaining unexplained portion of predicted deep parametric portfolio weights to the effect of non-linearity in functional form. Furthermore, we analyze the portfolio characteristics of the ex-post fitted surrogate models during the out of sample periods. Inter alia, this enables us to assess to which extent non-linearity with respect to weight predictions translates into utility differences.

Second, we calculate variable importance in the model as 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 losses 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.

Third, we evaluate the sensitivity of the model output 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 across the cross-section. 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 yield reasonable results. To circumvent this problem, we apply our own algorithm: when assessing the sensitivity with respect to variable $k$, we set the values of the remaining variables to zero, i.e. their median. This means that effectively, we reduce our input data to the variable of interest. We then predict out-of-sample portfolio weights based on the estimated model and the manipulated data. Subsequently, we plot the weights as a function of input variable $k$. We interpret the behavior of predicted weights conditional on values of $k$ as the sensitivity of weights (i.e. its partial dependence) with respect to $k$.

# 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.[6] Analogous to Brandt et al. (2009) we provide results as follows: We report (1) the utility that a respective portfolio strategy generates, (2) distributional characteristics of the portfolio weights, (3) properties of the portfolio returns and (4) the strategies' alphas against a Fama-French six-factor model.

The first row of Table 1 reports the realized utility across out-of-sample periods for a mean-variance investor with absolute risk aversion of five. The equally weighted and value weighted portfolio yield a utility of 0.0024 and 0.0029, respectively. The standard parametric portfolio policy substantially outperforms the simple portfolios, yielding a utility of 0.0267. However, the deep parametric portfolio policy yields a utility of 0.0469, almost twice as large as the utility derived from the linear parametric portfolio policy. The difference in utilities is significant at the 0.1% level.[7] This suggests that taking into account predictor interactions and non-linear relationships substantially improves 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

[6] To ensure 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. A more detailed description is given in Appendix A.

[7] We follow DeMiguel et al. (2022) and construct one-sided p-values from 10,000 bootstrap samples using the stationary bootstrap method of Politis and Romano (1994) with an average block size of five and the procedure of Ledoit and Wolf (2008). This method is also used when assessing the statistical significance of utility and Sharpe ratio differences between the deep and the linear parametric portfolio policy hereafter.

portfolio policy being almost nine times as large as in the case of the equally weighted and value weighted portfolio, respectively. However, due to the weight constraint shown in Equation (7) these positions remain below 3%. 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 portfolio before rebalancing at time $t$, that is,

$$w^{+}_{i,t-1} = w_{i,t-1} * (1 + r_{i,t}). \tag{8}$$

As Ang et al. (2011) show, average gross leverage of hedge fund companies amounts to 120% in the period after 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 in Section 4.2 by including a penalty term for turnover and a constraint for leverage in our objective function.

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 standard deviation increases only modestly by 0.7 percentage points, thereby leading to a Sharpe ratio that is around 40% higher. The difference in Sharpe ratios is statistically significant at the 1% level. In fact, both models substantially 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 to 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 B. 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**

Surrogate modeling allows us to disentangle the contributions of non-linearity in variables and non-linearity in functional form with respect to the predictions as well as the utility gains of the deep parametric portfolio policy as compared to the linear parametric portfolio policy. As one would expect, Figure 3 shows that a simple linear surrogate model perfectly explains the out-of-sample weight predictions retrieved from the linear parametric portfolio policy. However, a simple linear model only explains 60-70% of the variation in out-of-sample weights predicted by the deep parametric portfolio policy. An extended surrogate model that allows for non-linearity in variables explains between 80-88% of the variation in out-of-sample weights. Based on these numbers, one can infer that up to ∼70% of the underlying characteristic-weight relationship is of linear nature, ∼10-20% can be captured by interactions, and the remaining ∼10-20% can be captured by a non-linear functional form.

[FIGURE 3 ABOUT HERE]

In Table 2 we further analyze the portfolios generated by the the respective surrogate models. As implicitly indicated by the aforementioned $R^2$'s, the surrogate model for the parametric portfolio policy yields a portfolio that is equivalent to the original model. Hence, its portfolio characteristics, especially its utility, are equivalent to the original model. In the deep parametric portfolio policy case, three observations stand out: First, the simple linear surrogate model yields a utility that is nearly 20% lower than that of the original deep parametric portfolio policy. Second, the linear surrogate model extended by two-way interactions yields a utility that is slightly higher than the utility based on the simple linear surrogate model. Thus, the above mentioned observations in terms of $R^2$ roughly translate into differences in utility. However, this difference is not as large as the difference in terms of $R^2$. Third, note that the utility of a linear surrogate model which is fitted to the estimated portfolio weights of the deep parametric portfolio policy (DPPP Pred.) is much larger than the utility of the linear portfolio policy that is directly fitted to the data (the original PPP model). This can seem puzzling at first sight as the PPP model should be able to generate the same weights. The reason is that surrogate models are fitted on the out-of-sample weight predictions. Hence, the predictions of the linear surrogate model yield in-sample utility, while the original models show out-of-sample utility.

[TABLE 2 ABOUT HERE]

**Variable importance**

Next, we turn to variable importance measured as discussed in section 2.5. Figure 4 compares the most important variables in the linear and deep parametric portfolio policies. 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.

[FIGURE 4 ABOUT HERE]

Figure C.2 in the appendix attempts to group variables into categories (such as "earnings-related", or "risk-related"). We do again find that the most important categories contain past-return based variables. The other relevant variable categories are "delayed processing" (anomalies that are based on delayed processing of information, e.g. industry momentum) and earnings-related variables.[8]

**Partial dependence**

Figure 5 depicts the marginal association between portfolio weights and input variables. We examine the sensitivity with respect 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 that a larger value implies a higher expected return. As implied by the linearity of the approach, the variables are linearly related to predicted weights in the case of the standard linear parametric portfolio policy. In contrast, the deep parametric portfolio policy weights are non-linearly related to the variables. More specifically, these relationships all appear to be convex. Interestingly, the convex shape appears to be quite similar for every variable: a steep increase in weight prediction occurs in the sixth or seventh decile, respectively. Moreover, the weight predictions generally appear to roughly follow the trend in mean returns across deciles. The difference in marginal sensitivities between the linear and the deep parametric portfolio policy illustrates that the latter is picking up non-linear relationships that the former is not able to pick up by construction.

[FIGURE 5 ABOUT HERE]

# 4 Extensions of the benchmark model

## 4.1 Long only

A large majority of equity portfolios face restrictions on short selling. We incorporate short-sale constraints as in Brandt et al. (2009), i.e. we truncate portfolios weights at zero (and still keep the

[8] Table D.1 in the appendix shows the category of each anomaly variable, based on Jensen et al. (2021) and extended by us for variables that are not considered in their study.

cap of 3% per stock). In particular, to make sure that portfolio weights still sum up to one, we add the following portfolio rebalancing term to the end of our optimization process:

$$w^*_{i,t} = \frac{max[0, w_{i,t}]}{\sum_{j=1}^{N_t} max[0, w_{i,t}]}. \tag{9}$$

Table 3, shows results from estimating long-only portfolios. Again, the deep parametric portfolio policy yields the highest utility, although utility is markedly lower than in the unconstrained case. Still, the utility of the deep parametric portfolio policy is around four times higher than the utility of the market portfolios and around 40% higher than the utility of the linear parametric portfolio policy. The difference between the utility of the deep and the linear parametric portfolio policy is statistically significant at the 0.1% level.

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 unconstrained benchmark results in Table 1, here we report the fraction of weights that are equal to zero. Interestingly, on average the deep portfolio policy does not include 11% of stocks, while the linear portfolio policy does not include 32% of the available stocks. Thus, the deep portfolio policy invests in more stocks but also has a higher individual maximum weight (1.64% vs 0.42%), indicating 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 19% higher than the Sharpe ratio of the linear portfolio policy. This difference is statistically significant at the 0.1% level. Interestingly, the third and fourth moments of all portfolio policies are similar and the portfolio return distributions are 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. Without the ability to take (potentially extreme) short positions, the estimated parametric portfolios appear to be much more realistic. Nonetheless, the deep portfolio policy still outperforms the other portfolios in terms of realized out-of-sample utility.

The comparison between the unconstrained (Table 1) and the long-only case (Table 3) also yields interesting insights. First, the unconstrained portfolio benefits from using the short positions as leverage to increase exposure to the long positions. Consistent with this observation, the linear portfolio policy has a similar fraction of short positions and stocks not held in the two models. Second, 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.

Variable importance rankings are similar to the unconstrained models. Figure 6 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-only 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 6 ABOUT HERE]

## 4.2 Transaction costs and leverage

The results of the unconstrained linear and the deep portfolio policy yield unfeasible portfolios with high leverage and turnover. To investigate whether 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).[9] Thus,

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

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

$$TC = E_t[\sum_{i=1}^{N_t} |\kappa_{i,t}(w_{i,t} - w^+_{i,t-1})|], \tag{10}$$

where $w^+_{i,t-1}$ is the portfolio before rebalancing as in Equation (8).

The leverage constraint is constructed analogously to our weight constraint in Equation (7). Ang et al. (2011) show that the average gross leverage of hedge fund companies amounts to 120% in the period after 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

$$\sum_{i=1}^{N_t} w_i I(w_i < 0) \geq -1 \tag{11}$$

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 4 shows the results for the 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 portfolio 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 more extensively than the linear model. More specifically, the deep model predicts high weights in good performing stocks at the cost of less diversification. Still, the deep parametric portfolio policy delivers a utility gain over a linear policy of about 20%, statistically significant at the 1% level. Thus, despite turnover still being higher than in the linear approach (168% versus 97%), the deep model still yields a higher realized mean-variance utility. 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, mean return and 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

Sharpe ratio decreases by around a third. Nonetheless, the difference between Sharpe ratios is still significant at the 5% level. The third and fourth moment are similar across all portfolios. The alphas of the estimated models are much smaller, but still highly significant.

[TABLE 4 ABOUT HERE]

Comparing the variable importance of included firm characteristics with the previous models, we find that this set of constraints leads to a very different picture. Figure 6 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. This is an intuitive result, since trading conditional on short-term reversal implies turnover by definition. Hence, when penalizing turnover via transaction costs, short-term reversal will necessarily lose some of its importance to a certain degree. Further, in line with the results of DeMiguel et al. (2020), we observe that variable importance is much more balanced across variables in general. In the deep model, short-term reversal is still the most important variable, but it becomes evident that its relative importance is lower than in the previous models. Again, this intuitively follows from the aforementioned mechanism. As in the linear model, variable importance becomes 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. In our benchmark model we assume an absolute risk aversion coefficient of five. Table 5 shows how our model performs for different degrees of absolute 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 show that investors with an absolute risk aversion of five experience the largest utility gains relative to the equally weighted portfolio benchmark. In general, we observe that the utility gains decrease relative to an equally weighted portfolio 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. Further, this shows that the risk aversion parameter can also be used as a regularization parameter, since increasing risk aversion leads to decreasing variance in the predicted weights, which reduces overfitting. Consequently, the investor with a risk aversion of two achieves lower utility gains relative to the benchmark portfolio than the investor with a risk aversion of five due to overfitting.

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 higher the risk aversion, the more the respective portfolio return distribution tends towards a normal distribution. Intuitively, with increasing risk aversion the alphas of the factor model regressions decrease.

[TABLE 5 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

$$u(r_{p,t+1}) = \frac{(1+r_{p,t+1})^{1-\gamma}}{1-\gamma}, \tag{12}$$

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

$$u(r_{p,t+1}) = \begin{cases} -l(\overline{W} - (1+r_{p,t+1}))^b & \text{if } (1+r_{p,t+1}) < \overline{W} \\ ((1+r_{p,t+1}) - \overline{W})^b & \text{otherwise} \end{cases}, \tag{13}$$

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 6 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. Interestingly, 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 over higher order moments, which differentiate the CRRA investor from an equally risk-averse mean-variance investor. In our data, however, the effect of higher order moments is not strong enough to heavily 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 cares about the size of the tail of the portfolio return distribution, rather than the mean to variance ratio, which is relevant to a mean-variance investor. This is also reflected in the results in Figure 6. Both portfolios show a high variance compared to the mean-variance and CRRA investor, however, they also display higher skewness. This high positive skewness illustrates a highly right tailed distribution. As the p-value indicates, the Sharpe ratios do not seem to differ significantly, while the deep model results in more than twice the utility than the linear model. The deep portfolio policy yields 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 yields slightly higher leverage and turnover.

[TABLE 6 ABOUT HERE]

We further investigate how utility is accumulated across out-of-sample periods. More specifically, for each utility function, we plot the cumulative out-of-sample period per period utility of the equally weighted portfolio, the linear parametric portfolio and the deep parametric portfolio. Irrespective of the investor's utility function, the deep parametric portfolio consistently outperforms the linear parametric portfolio policy and an equally weighted portfolio in utility terms.

[FIGURE 7 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. More specifically, we show that neural networks can be used to optimize portfolio weighting based on firm characteristics in a one-step optimization framework. Furthermore, we show how traditional distance loss functions can be replaced by context-specific utility functions in neural networks.

Our empirical results indicate that neural networks perform significantly better than linear models in regards to portfolio allocation, suggesting that firm characteristics are non-linearly related to optimal portfolio weights. Consistent with this hypothesis, we show that linear surrogate models are not able to fully explain the deep parametric portfolio weight predictions, even when accounting for two-way interactions. We further shed light on the non-linear relationship between characteristics and predicted weights by depicting the sensitivity of predicted weights with respect to the input. Again, we show a clearly non-linear effect. Gaining more insights into the models, we find that return-based stock characteristics resemble the most important group of predictors. However, consistent with DeMiguel et al. (2020), variable importance is more evenly distributed and puts less weight on past returns when constraints and transaction costs are taken into account in the optimization process.

Exploring variations in the degree of an investor's risk aversion or their utility function, we find that (absolute) portfolio weights are typically lower when risk aversion is higher, consistent with more risk-averse investors aiming for a more balanced portfolio weight distribution.

Overall, the results show that neural networks are successful in solving portfolio choice problems. Specifically, this is due to neural networks allowing predictor variables to relate to moments of the expected return distribution non-linearly, both in terms of variable interactions and in terms of functional form. Highlighting the growing role of machine learning and non-linear models in finance, our approach thus resembles a comparably simple and flexible neural network based model that enables practitioners and researchers alike to create reasonable portfolio allocations based on firm characteristics and preferences.

## 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.

Butler, A. and R. H. Kwon (2021). Integrating prediction in mean-variance portfolio optimization. Working Paper.

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.

Chevalier, G., G. Coqueret, and T. Raffinot (2022). Supervised portfolios. *Quantitative Finance 22*(12), 2275–2295.

Cong, L., K. Tang, J. Wang, and Y. Zhan (2021). Alphaportfolio: Direct construction through deep reinforcement learning and interpretable ai. Working Paper.

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. Martin-Utrera, and R. Uppal (2022). A multifactor perspective on volatility-managed portfolios. Working Paper.

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. Working Paper.

Feng, G., J. He, and N. G. Polson (2018). Deep Learning for Predicting Asset Returns. Working Paper.

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.

Ioffe, S. and C. Szegedy (2015, 07–09 Jul). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Proceedings of the 32nd International Conference on Machine Learning 37*, 448–456.

Jensen, T. I., B. T. Kelly, S. Malamud, and L. H. Pedersen (2022). Machine learning and the implementable efficient frontier. Swiss Finance Institute Research Paper No. 22-63.

Jensen, T. I., B. T. Kelly, and L. H. Pedersen (2021). Is there a replication crisis in finance? Technical report, National Bureau of Economic Research.

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. Working Paper.

Ledoit, O. and M. Wolf (2008). Robust performance hypothesis testing with the sharpe ratio. *Journal of Empirical Finance 15*(5), 850–859.

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

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

Moritz, B. and T. Zimmermann (2016). Tree-based conditional portfolio sorts: The relation between past and future stock returns. Working Paper.

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

Politis, D. N. and J. P. Romano (1994). The stationary bootstrap. *Journal of the American Statistical Association 89*(428), 1303–1313.

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. Working Paper.

# Figures

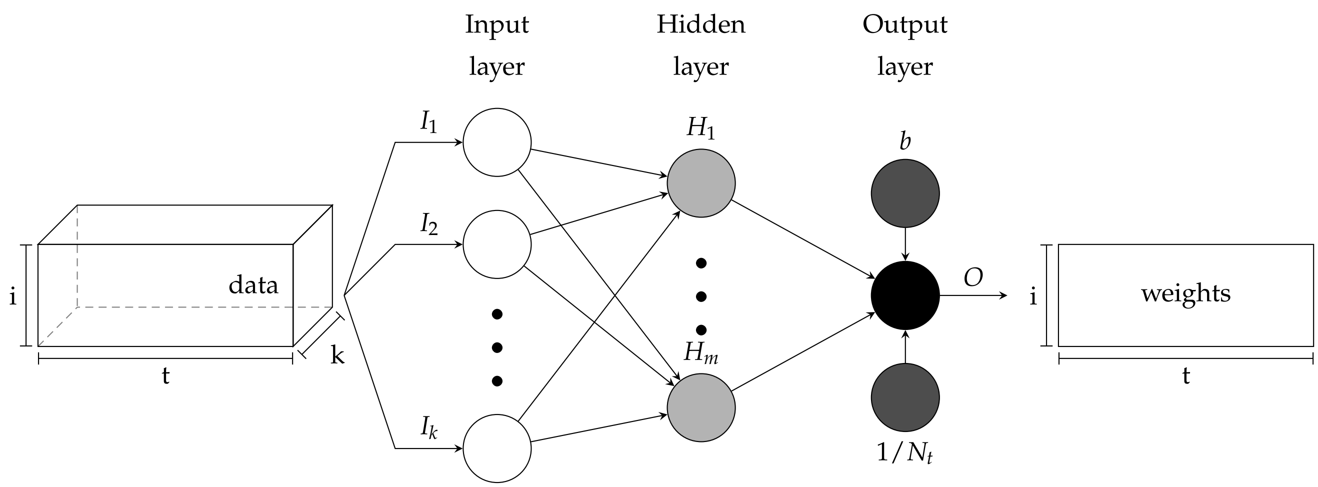



**Figure 1: Neural Network Structure**
This figure presents the core structure of our neural networks. White circles denote the input layer, grey circles denote the hidden layer and black circles denote the output layer. The data cube on the left depicts the structure of our data, i.e. we have $k$ variables 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 $O$.

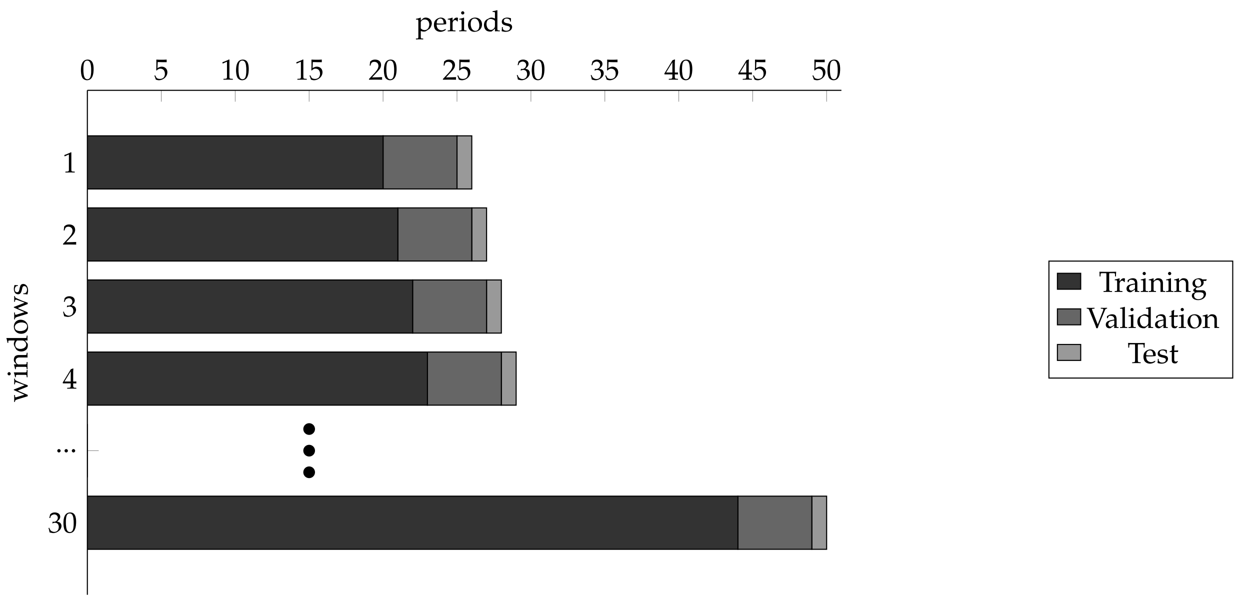



**Figure 2: Out-Of-Sample Testing Strategy**
This figure presents our out-of-sample testing strategy. We recursively increase our training window, presented by the black portion of each bar, while holding validation and test window constant, presented by the grey portions of each bar.

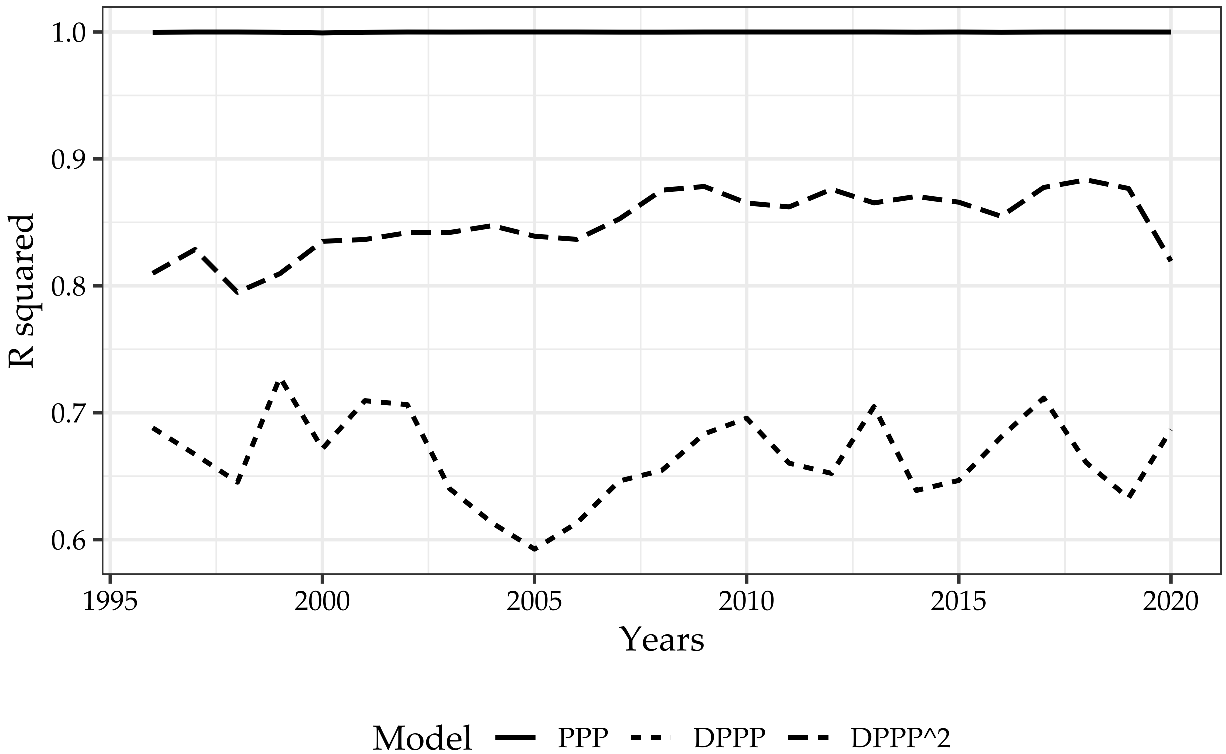
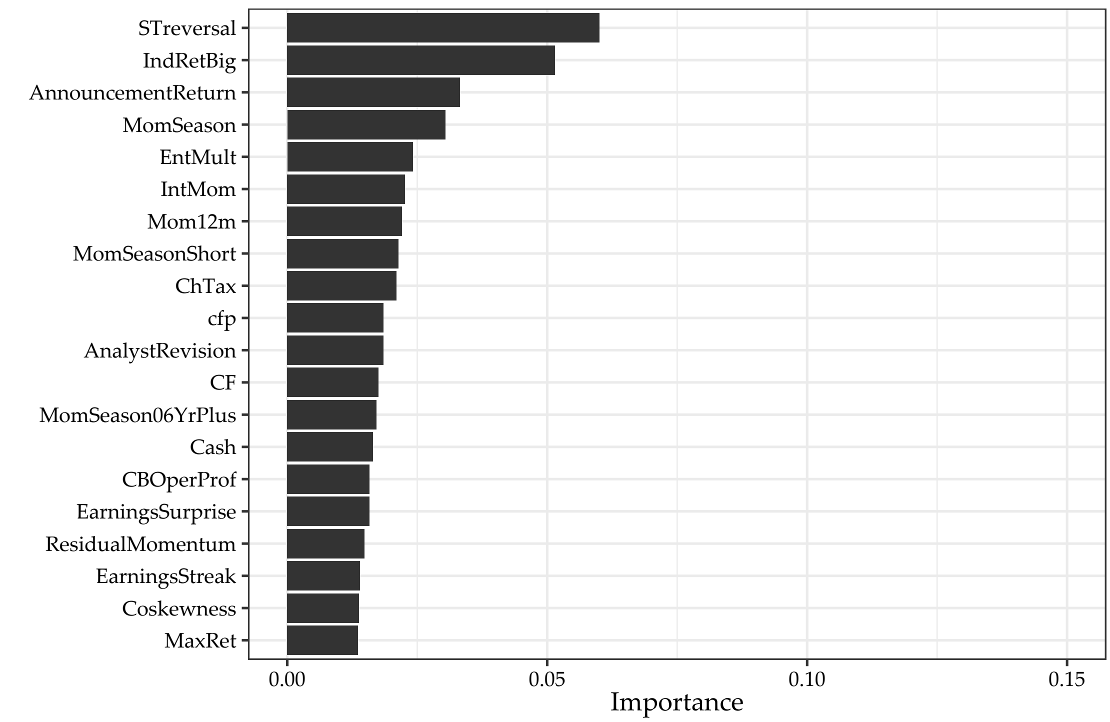
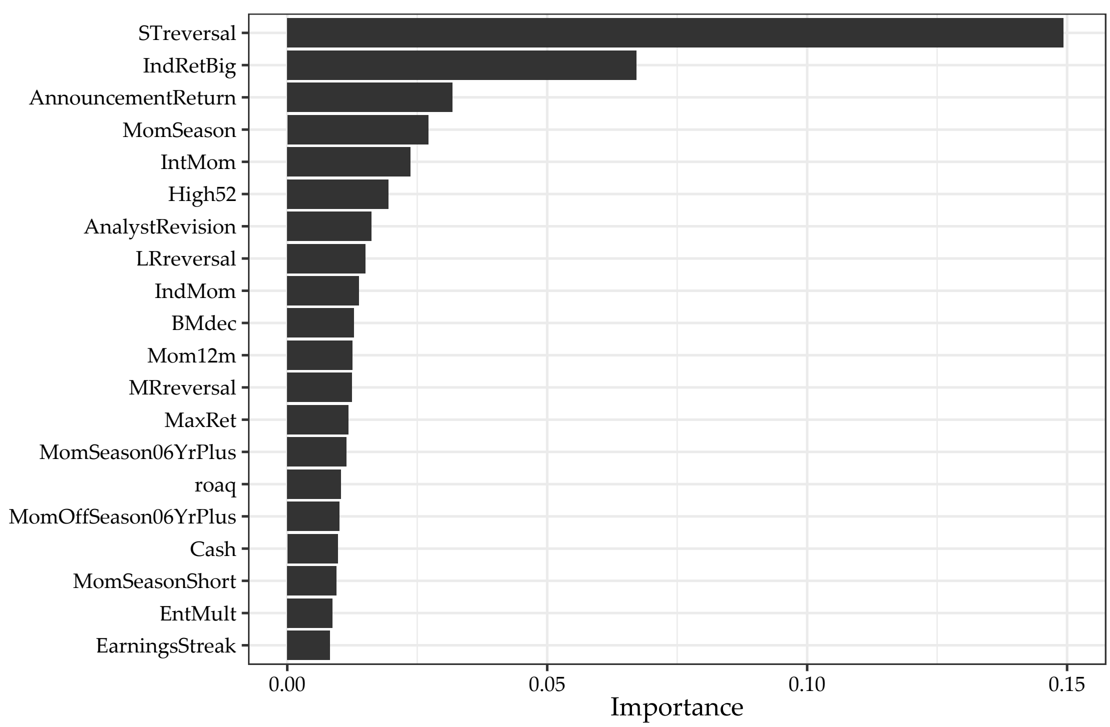



**Figure 3: Surrogate** $R^2$
This figure depicts the $R^2$ of the surrogate models in the benchmark case. More specifically, the "PPP"-line depicts the $R^2$ of a linear surrogate model in case of the PPP, the "DPPP"-line depicts the $R^2$ of a linear surrogate model in case of the DPPP and the "DPPP$^2$"-line depicts the $R^2$ of a $l^1$-regularized linear 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.

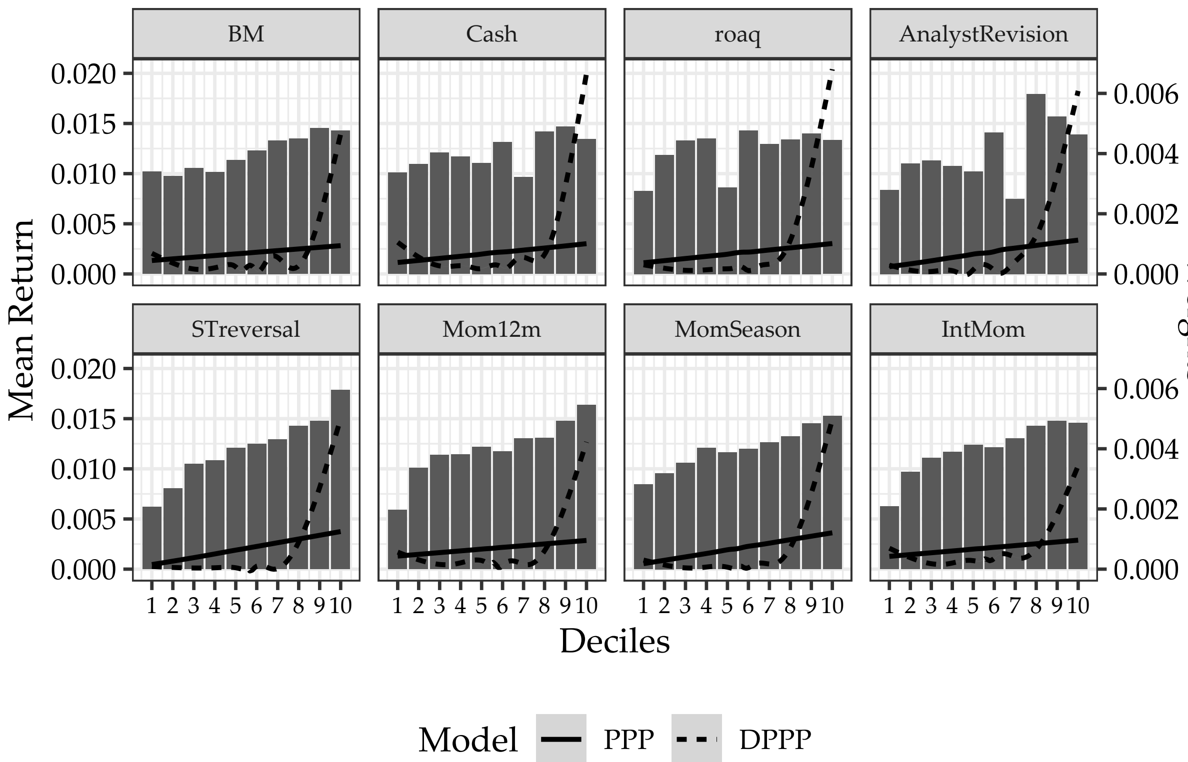



**Figure 5: Marginal association between portfolio weights and characteristics**
This figure shows the sensitivity of predicted weights (right vertical axis) with respect to values of the respective variable (horizontal axis). The aforementioned relationship is depicted by curves, smoothed via spline-regressions. The figure also includes bars, depicting the mean return (left vertical axis), per variable decile (horizontal axis).


**Figure 6: Variable importance across models**
We rank the top 50 stock characteristics in terms of its importance across all models. The higher a stock characteristic within the figure, the higher its average importance across all models. Columns correspond to individual models, with columns ending with "_Main" representing unconstrained models, columns ending with "_Long" representing long-only models, and columns ending with "_Con" representing models with constrained leverage and transaction costs. The color gradations within each column indicate importance, i.e. the darker the gradation, the more important the stock characteristic.

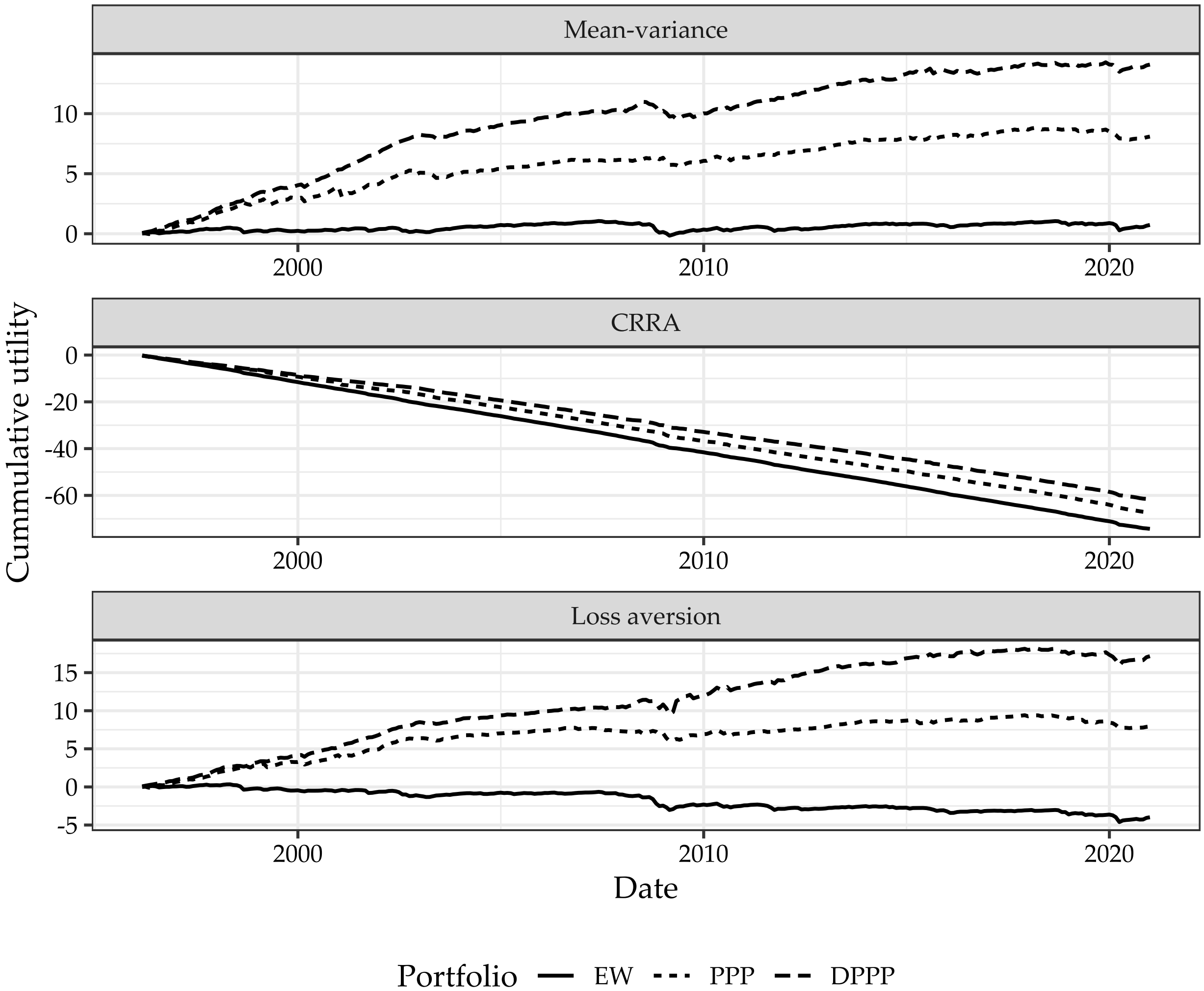



**Figure 7: Cumulative utility for different utility functions**
This figure includes plots of the cumulative utility for each of the utility functions considered. More specifically, for each utility function, we plot the cumulative utility across out-of-sample periods for the equally weighted portfolio (EW), the parametric portfolio policy (PPP) and the deep parametric portfolio policy (DPPP).

# Tables

**Table 1: Deep and linear portfolio policy**

| | EW | VW | PPP | DPPP |
|---|---|---|---|---|
| Utility | 0.0024 | 0.0029 | 0.0267 | 0.0469 |
| p-value$(U_{DPPP} - U_{PPP})$ | | | | 0.0004 |
| | | | | |
| $\sum \lvert w_i \rvert / N_t * 100$ | 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 |
| $\sum I(w_i < 0)/N_t$ | 0.0000 | 0.0000 | 0.4334 | 0.4411 |
| $\sum \lvert w_{i,t} - w^{+}_{i,t-1} \rvert$ | 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 | 0.6461 | 0.6609 | 1.8070 | 2.5170 |
| p-value$(SR_{DPPP} - SR_{PPP})$ | | | | 0.0066 |
| | | | | |
| $FF5 + Mom\ \alpha$ | -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 optimized for a mean-variance investor with absolute risk aversion of five conditional on 157 firm characteristics. The regular portfolio policy is a linear specification of Equation (4), 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 equally-weighted portfolio, value-weighted portfolio, parametric portfolio policy, and deep parametric portfolio policy, respectively. The first rows show the utility of the investor as well as the bootstrapped one-sided p-value for the difference in utility between the DPPP and the PPP. 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 annualized Sharpe ratios and the bootstrapped one-sided p-value for the difference in Sharpe ratios between the DPPP and the PPP. 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: Surrogate models**

| | PPP | PPP Pred. | DPPP | DPPP Pred. | DPPP^2 Pred. |
|---|---|---|---|---|---|
| $R^2$ | | 0.9999 | | 0.6607 | 0.8552 |
| | | | | | |
| Utility | 0.0267 | 0.0266 | 0.0469 | 0.0382 | 0.0398 |
| | | | | | |
| $\sum \lvert w_i \rvert / N_t * 100$ | 0.5060 | 0.5059 | 0.6057 | 0.4674 | 0.4619 |
| $max\ w_i * 100$ | 2.0748 | 2.0744 | 1.7260 | 2.0096 | 1.9105 |
| $min\ w_i * 100$ | -2.2097 | -2.2095 | -1.8370 | -2.0663 | -2.2662 |
| $\sum w_i I(w_i < 0)$ | -3.1475 | -3.1474 | -3.8665 | -2.8697 | -2.8300 |
| $\sum I(w_i < 0)/N_t$ | 0.4334 | 0.4335 | 0.4411 | 0.4412 | 0.4290 |
| $\sum \lvert w_{i,t} - w^{+}_{i,t-1} \rvert$ | 3.9370 | 3.9365 | 7.6984 | 6.2002 | 6.1322 |
| | | | | | |
| Mean | 0.0451 | 0.0450 | 0.0684 | 0.0537 | 0.0563 |
| StdDev | 0.0895 | 0.0896 | 0.0963 | 0.0831 | 0.0854 |
| Skew | -0.1636 | -0.2002 | 1.0761 | 0.2137 | 0.3930 |
| Kurt | 1.9043 | 2.0386 | 6.6301 | 2.6537 | 2.8152 |
| SR | 1.7449 | 1.7403 | 2.4591 | 2.2388 | 2.2840 |

This table shows out-of-sample estimates of the deep and linear portfolio policies with 157 firm characteristics and optimized for a mean-variance investor with absolute risk aversion of five as well as the estimates of the linear surrogate models for the PPP and the DPPP, respectively. The regular portfolio policy is a linear specification of Equation (4), while the deep model is a feed-forward neural network with three hidden layers and 32, 16, and eight nodes, respectively. The columns labeled "PPP" and "DPPP" show the statistics of the originally estimated portfolio policies, while the columns labeled "PPP Pred." and "DPPP Pred." show the statistics of the linear surrogate model. Finally, the column labeled "DPPP^2 Pred." shows the statistics of a lasso surrogate model that includes the predictors and all possible two-way interactions. The first row shows the average $R^2$ of regressing out-of-sample weight predictions on the respective surrogate model. The second row shows the utility of the investor. The third 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 fourth set of rows shows the first four moments of the final portfolio return distributions as well as the annualized Sharpe ratios.

**Table 3: Long-only deep and linear portfolio policy**

| | EW | VW | PPP | DPPP |
|---|---|---|---|---|
| Utility | 0.0024 | 0.0029 | 0.0084 | 0.0116 |
| p-value$(U_{DPPP} - U_{PPP})$ | | | | 0.0001 |
| | | | | |
| $\sum \lvert w_i \rvert / N_t * 100$ | 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 |
| $\sum I(w_i < 0)/N_t$ | 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 | 0.6461 | 0.6609 | 1.0045 | 1.1941 |
| p-value$(SR_{DPPP} - SR_{PPP})$ | | | | 0.0002 |
| | | | | |
| $FF5 + Mom\ \alpha$ | -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 including a long-only constraint optimized for a mean-variance investor with absolute risk aversion of five conditional on 157 firm characteristics. The regular portfolio policy is a linear specification of Equation (4), 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 equally-weighted portfolio, value-weighted portfolio, parametric portfolio policy, and deep parametric portfolio policy, respectively. The first rows show the utility of the investor as well as the bootstrapped one-sided p-value for the difference in utility between the DPPP and the PPP. 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 annualized Sharpe ratios and the bootstrapped one-sided p-value for the difference in Sharpe ratios between the DPPP and the PPP. 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: Constrained and penalized deep and linear portfolio policy**

| | EW | VW | PPP | DPPP |
|---|---|---|---|---|
| Utility | 0.0021 | 0.0028 | 0.0139 | 0.0169 |
| p-value$(U_{DPPP} - U_{PPP})$ | | | | 0.0015 |
| | | | | |
| $\sum \lvert w_i \rvert / N_t * 100$ | 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 |
| $\sum I(w_i < 0) / N_t$ | 0.0000 | 0.0000 | 0.3417 | 0.3181 |
| $\sum \lvert w_{i,t} - w_{i,t-1}^{+} \rvert$ | 0.0931 | 0.0779 | 0.9699 | 1.6756 |
| | | | | |
| Mean | 0.0107 | 0.0104 | 0.0184 | 0.0223 |
| StdDev | 0.0584 | 0.0549 | 0.0421 | 0.0465 |
| Skew | -0.3711 | -0.5055 | -0.9085 | -0.7414 |
| Kurt | 3.6640 | 3.3435 | 2.6099 | 2.8353 |
| SR | 0.6345 | 0.6533 | 1.5110 | 1.6624 |
| p-value$(SR_{DPPP} - SR_{PPP})$ | | | | 0.0451 |
| | | | | |
| $FF5 + Mom\ \alpha$ | -0.0003 | -0.0004 | 0.0076 | 0.0113 |
| $StdErr(\alpha)$ | 0.0007 | 0.0006 | 0.0013 | 0.0017 |

This table shows out-of-sample estimates of the deep and linear portfolio policies including a transaction cost penalty and a leverage constraint, optimized for a mean-variance investor with absolute risk aversion of five conditional on 157 firm characteristics. The regular portfolio policy is a linear specification of Equation (4), 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 equally-weighted portfolio, value-weighted portfolio, parametric portfolio policy, and deep parametric portfolio policy, respectively. The first rows show the utility of the investor as well as the bootstrapped one-sided p-value for the difference in utility between the DPPP and the PPP. 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 (net of transaction costs) as well as the annualized Sharpe ratios and the bootstrapped one-sided p-value for the difference in Sharpe ratios between the DPPP and the PPP. 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 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 |
| | | | | |
| $\sum \lvert w_i \rvert / N_t * 100$ | 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 | 2.4408 | 2.5170 | 2.6402 | 2.5443 |
| | | | | |
| $FF5 + Mom\ \alpha$ | 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, optimized for a mean-variance investor with absolute risk aversion of two, five, ten and 20, respectively, conditional on 157 characteristics. 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 annualized 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 6: Deep portfolio policy with different investor preferences**

| | CRRA | | LA | |
|---|---|---|---|---|
| | PPP | DPPP | PPP | DPPP |
| Utility | -0.2253 | -0.2063 | 0.0266 | 0.0574 |
| p-value$(U_{DPPP} - U_{PPP})$ | | 0.0003 | | 0.0004 |
| $\sum \lvert w_i \rvert / N_t * 100$ | 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 |
| $\sum I(w_i < 0)/N_t$ | 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 | 1.8391 | 2.5101 | 1.8789 | 1.9963 |
| p-value$(SR_{DPPP} - SR_{PPP})$ | | 0.0075 | | 0.4227 |
| $FF5 + Mom\ \alpha$ | 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, 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, conditional on 157 firm characteristics. The regular portfolio policy is a linear specification of Equation (4), 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 rows show the utility of the investor as well as the bootstrapped one-sided p-value for the difference in utility between the DPPP and the PPP. 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 annualized Sharpe ratios and the bootstrapped one-sided p-value for the difference in Sharpe ratios between the DPPP and the PPP. 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: Neural Network Configuration
- Appendix B: Robustness
- Appendix C: Supplementary figures
- Appendix D: Supplementary tables

# Appendix A Neural Network Configuration

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. We consider different network architectures in Appendix B.

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.

$$R(z) = \begin{cases} z & \text{if } z > 0 \\ \alpha z & \text{otherwise} \end{cases}, \tag{14}$$

where $z$ denotes the input and $\alpha$ denotes some small non-zero constant, in our case 0.01. 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. Moreover, we shift the activation function at every node in every hidden layer by adding a constant. This is commonly referred to as bias in the machine learning literature.

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

To control for the non-linearity and heavy parametrization of the model, we employ different regularization techniques to prevent overfitting: first, as mentioned above, we impose a constraint on an individual stock's absolute portfolio weight of $|3\%|$.

Second, we add a lasso ($l_1$) penalty term to the loss function to be minimized. Adding the penalty implies a potential shrinkage of coefficients towards 0. This in turn reduces the variance of the prediction, i.e. preventing the model to be overfitted.

Third, we employ early stopping on the validation data. 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.

Fourth, we implement a dropout layer before the first hidden layer (Srivastava et al., 2014). 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. The combination of a dropout layer, $l_1$-regularization and early stopping tremendously helps to reduce overfitting and model complexity.

Fifth, 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). 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. Brandt et al. (2009) standardize characteristics cross-sectionally to have zero mean and unit standard deviation across all stocks at date $t$. Hence, the model predictions represent deviations from the benchmark portfolio. However, applying the aforementioned activation function destroys this structure. 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) requires 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.

We provide a summary of the relevant hyperparameters in Table D.2.

[TABLE D.2 ABOUT HERE]

# Appendix B Robustness Checks

## B.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 D.3 shows the results. The second model is our original benchmark model that we added for comparison.[10] The remaining columns contain results based on networks with two, four or five hidden layers. We observe that reducing the number of hidden layers to two slightly reduces the utility. This reduction in utility is significant at the 10%-level. In contrast, increasing the number of hidden layers to four or five, respectively, does not yield statistically significant differences in utility. We thus conclude that in general, reasonable complexity adjustments in terms of the number of hidden layers do not lead to significantly different outcomes.

[TABLE D.3 ABOUT HERE]

## B.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

[10] 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 C.3.

[Figure C.3 ABOUT HERE]

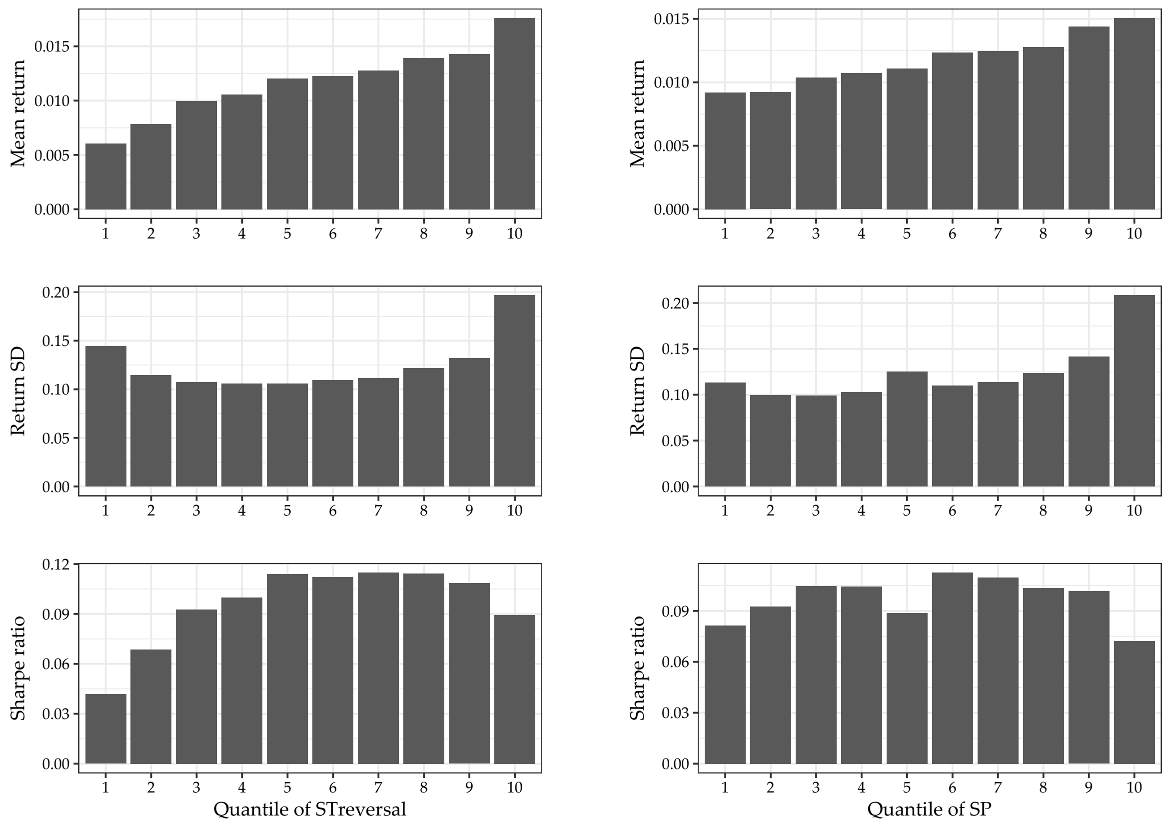
More specifically, we split our data according to its update frequency and its data category, respectively. For update frequency we divide our data into monthly, yearly and quarterly characteristics. For data category we divide our data into Accounting, Price, Trading and Analyst characteristics. The update frequency and data category of each predictor is shown in Table D.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 D.4 shows the results for the benchmark linear and deep portfolio policy followed by the two variations in network architectures for the deep portfolio policy. The results indicate that changes in realized utility are not large. In fact, splitting according the predictor category does not yield significant gains or losses in terms of utility. Splitting according to the frequency of predictors does lead to a small increase of utility, which is significant at the 10%-level. 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 largest differences can be observed for 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 D.4 ABOUT HERE]

# Appendix C Supplementary Figures




**(a)** Short-Term Reversal

**(b)** Sales-To-Price

**Figure C.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).

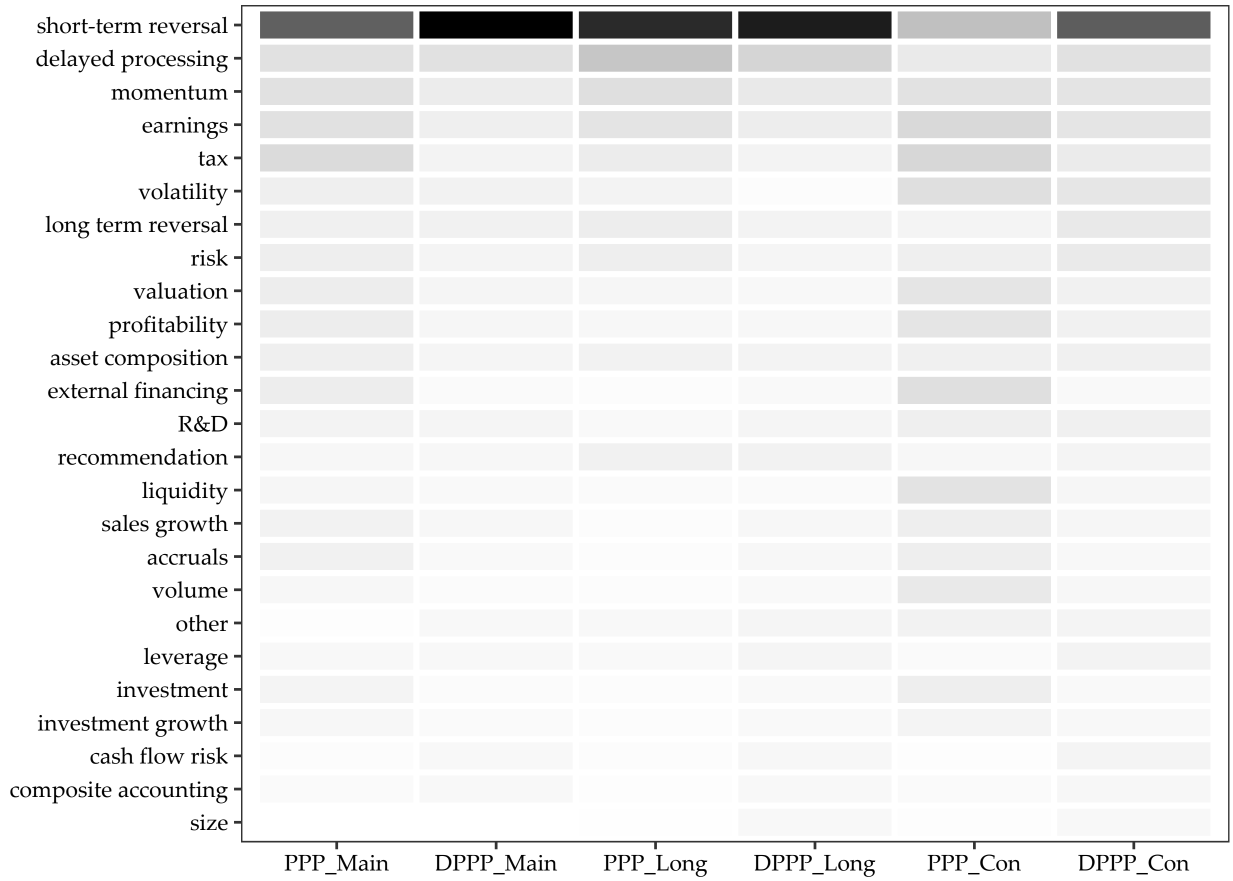



**Figure C.2: Variable importance per cluster**
We group the variables into clusters according to the economic category specified in the Open Source Asset Pricing data set by Chen and Zimmermann (2022). Clusters are then ranked by average characteristic importance within the respective cluster. The higher a cluster within the figure, the higher its average importance across all models. Columns correspond to individual models, with columns ending with "_Main" representing unconstrained models, columns ending with "_Long" representing long-only models, and columns ending with "_Con" representing models with constrained leverage and transaction costs. The color gradations within each column indicate importance, i.e. the darker the gradation, the more important the cluster.

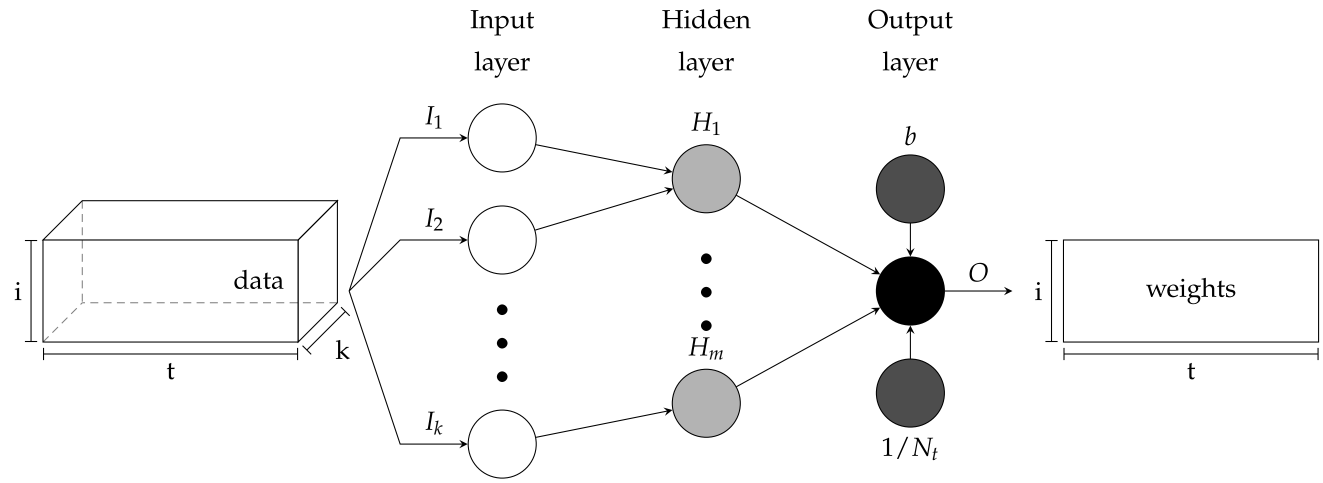



**Figure C.3: Non-Fully Connected Neural Network Structure**
This figure presents the structure of our non-fully connected networks. White circles denote the input layer, grey circles denote the hidden layer and black circles denote the output layer. The data cube on the left depicts the structure of 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 D Supplementary Tables

| Acronym | Long Description | Author(s) | Year, Journal | Frequency | Cat.Data | Cat.Economic |
|---|---|---|---|---|---|---|
| ChInvIA | Change in capital inv (ind adj) | Abarbanell and Bushee | 1998, AR | yearly | Accounting | investment growth |
| GrSaleToGrInv | Sales growth over inventory growth | Abarbanell and Bushee | 1998, AR | yearly | Accounting | sales growth |
| GrSaleToGrOverhead | Sales growth over overhead growth | Abarbanell and Bushee | 1998, AR | yearly | Accounting | sales growth |
| IdioVolAHT | Idiosyncratic risk (AHT) | Ali, Hwang, and Trombley | 2003, JFE | monthly | Price | volatility |
| EarningsConsistency | Earnings consistency | Alwathainani | 2009, BAR | yearly | Accounting | earnings |
| Illiquidity | Amihud's illiquidity | Amihud | 2002, JFM | monthly | Trading | liquidity |
| BidAskSpread | Bid-ask spread | Amihud and Mendelsohn | 1986, JFE | monthly | Trading | liquidity |
| grcapx | Change in capex (two years) | Anderson and Garcia-Feijoo | 2006, JF | yearly | Accounting | investment growth |
| grcapx3y | Change in capex (three years) | Anderson and Garcia-Feijoo | 2006, JF | yearly | Accounting | investment growth |
| betaVIX | Systematic volatility | Ang et al. | 2006, JF | monthly | Price | volatility |
| IdioRisk | Idiosyncratic risk | Ang et al. | 2006, JF | monthly | Price | volatility |
| IdioVol3F | Idiosyncratic risk (3 factor) | Ang et al. | 2006, JF | monthly | Price | volatility |
| CoskewACX | Coskewness using daily returns | Ang, Chen and Xing | 2006, RFS | monthly | Price | risk |
| Mom6mJunk | Junk Stock Momentum | Avramov et al | 2007, JF | monthly | Price | momentum |
| OrderBacklogChg | Change in order backlog | Baik and Ahn | 2007, Other | yearly | Accounting | accruals |
| roaq | Return on assets (qtrly) | Balakrishnan, Bartov and Faurel | 2010, JAE | quarterly | Accounting | profitability |
| MaxRet | Maximum return over month | Bali, Cakici, and Whitelaw | 2010, JF | monthly | Price | volatility |
| ReturnSkew | Return skewness | Bali, Engle and Murray | 2015, Book | monthly | Price | risk |
| ReturnSkew3F | Idiosyncratic skewness (3F model) | Bali, Engle and Murray | 2015, Book | monthly | Price | risk |
| CBOperProf | Cash-based operating profitability | Ball et al. | 2016, JFE | yearly | Accounting | profitability |
| OperProfRD | Operating profitability R&D adjusted | Ball et al. | 2016, JFE | yearly | Accounting | profitability |
| Size | Size | Banz | 1981, JFE | monthly | Price | size |
| SP | Sales-to-price | Barbee, Mukherji and Raines | 1996, FAJ | yearly | Accounting | valuation |
| EP | Earnings-to-Price Ratio | Basu | 1977, JF | monthly | Price | valuation |

Continued on next page

| Acronym | Long Description | Author(s) | Year, Journal | Frequency | Cat.Data | Cat.Economic |
|---|---|---|---|---|---|---|
| InvGrowth | Inventory Growth | Belo and Lin | 2012, RFS | yearly | Accounting | profitability |
| BrandInvest | Brand capital investment | Belo, Lin and Vitorino | 2014, RED | yearly | Accounting | investment |
| Leverage | Market leverage | Bhandari | 1988, JFE | monthly | Price | leverage |
| ResidualMomentum | Momentum based on FF3 residuals | Blitz, Huij and Martens | 2011, JEmpFin | monthly | Price | momentum |
| Price | Price | Blume and Husic | 1972, JF | monthly | Price | other |
| NetPayoutYield | Net Payout Yield | Boudoukh et al. | 2007, JF | monthly | Price | valuation |
| PayoutYield | Payout Yield | Boudoukh et al. | 2007, JF | monthly | Price | valuation |
| NetDebtFinance | Net debt financing | Bradshaw, Richardson, Sloan | 2006, JAE | yearly | Accounting | external financing |
| NetEquityFinance | Net equity financing | Bradshaw, Richardson, Sloan | 2006, JAE | yearly | Accounting | external financing |
| XFIN | Net external financing | Bradshaw, Richardson, Sloan | 2006, JAE | yearly | Accounting | external financing |
| DolVol | Past trading volume | Brennan, Chordia, Subra | 1998, JFE | monthly | Trading | volume |
| FEPS | Analyst earnings per share | Cen, Wei, and Zhang | 2006, WP | monthly | Analyst | profitability |
| AnnouncementReturn | Earnings announcement return | Chan, Jegadeesh and Lakonishok | 1996, JF | monthly | Price | earnings |
| REV6 | Earnings forecast revisions | Chan, Jegadeesh and Lakonishok | 1996, JF | monthly | Analyst | earnings |
| AdExp | Advertising Expense | Chan, Lakonishok and Sougiannis | 2001, JF | monthly | Accounting | R&D |
| RD | R&D over market cap | Chan, Lakonishok and Sougiannis | 2001, JF | monthly | Accounting | R&D |
| CashProd | Cash Productivity | Chandrashekar and Rao | 2009, WP | yearly | Accounting | profitability |
| std_turn | Share turnover volatility | Chordia, Subra, Anshuman | 2001, JFE | monthly | Trading | liquidity |
| VolSD | Volume Variance | Chordia, Subra, Anshuman | 2001, JFE | monthly | Trading | liquidity |
| retConglomerate | Conglomerate return | Cohen and Lou | 2012, JFE | monthly | Price | delayed processing |
| RDAbility | R&D ability | Cohen, Diether and Malloy | 2013, RFS | yearly | Accounting | other |
| AssetGrowth | Asset growth | Cooper, Gulen and Schill | 2008, JF | yearly | Accounting | investment |
| EarningsForecastDisparity | Long-vs-short EPS forecasts | Da and Warachka | 2011, JFE | monthly | Analyst | earnings |
| CompEquIss | Composite equity issuance | Daniel and Titman | 2006, JF | monthly | Accounting | external financing |

Continued on next page

| Acronym | Long Description | Author(s) | Year, Journal | Frequency | Cat.Data | Cat.Economic |
|---|---|---|---|---|---|---|
| IntanBM | Intangible return using BM | Daniel and Titman | 2006, JF | yearly | Accounting | long term reversal |
| IntanCFP | Intangible return using CFtoP | Daniel and Titman | 2006, JF | yearly | Accounting | long term reversal |
| IntanEP | Intangible return using EP | Daniel and Titman | 2006, JF | yearly | Accounting | long term reversal |
| IntanSP | Intangible return using Sale2P | Daniel and Titman | 2006, JF | yearly | Accounting | long term reversal |
| ShareIss5Y | Share issuance (5 year) | Daniel and Titman | 2006, JF | monthly | Accounting | external financing |
| LRreversal | Long-run reversal | De Bondt and Thaler | 1985, JF | monthly | Price | long term reversal |
| MRreversal | Medium-run reversal | De Bondt and Thaler | 1985, JF | monthly | Price | long term reversal |
| EquityDuration | Equity Duration | Dechow, Sloan and Soliman | 2004, RAS | yearly | Price | valuation |
| cfp | Operating Cash flows to price | Desai, Rajgopal, Venkatachalam | 2004, AR | yearly | Accounting | valuation |
| ForecastDispersion | EPS Forecast Dispersion | Diether, Malloy and Scherbina | 2002, JF | monthly | Analyst | volatility |
| ExclExp | Excluded Expenses | Doyle, Lundholm and Soliman | 2003, RAS | quarterly | Analyst | composite accounting |
| ProbInformedTrading | Probability of Informed Trading | Easley, Hvidkjaer and O'Hara | 2002, JF | yearly | Trading | liquidity |
| OrgCap | Organizational capital | Eisfeldt and Papanikolaou | 2013, JF | yearly | Accounting | R&D |
| sfe | Earnings Forecast to price | Elgers, Lo and Pfeiffer | 2001, AR | monthly | Analyst | valuation |
| GrLTNOA | Growth in long term operating assets | Fairfield, Whisenant and Yohn | 2003, AR | yearly | Accounting | investment |
| AM | Total assets to market | Fama and French | 1992, JF | yearly | Accounting | valuation |
| BMdec | Book to market using December ME | Fama and French | 1992, JPM | yearly | Accounting | valuation |
| BookLeverage | Book leverage (annual) | Fama and French | 1992, JF | yearly | Accounting | leverage |
| OperProf | operating profits / book equity | Fama and French | 2006, JFE | yearly | Accounting | profitability |
| Beta | CAPM beta | Fama and MacBeth | 1973, JPE | monthly | Price | risk |
| EarningsSurprise | Earnings Surprise | Foster, Olsen and Shevlin | 1984, AR | quarterly | Analyst | earnings |
| AnalystValue | Analyst Value | Frankel and Lee | 1998, JAE | monthly | Analyst | valuation |
| AOP | Analyst Optimism | Frankel and Lee | 1998, JAE | monthly | Analyst | other |
| PredictedFE | Predicted Analyst forecast error | Frankel and Lee | 1998, JAE | monthly | Accounting | earnings |

Continued on next page

| Acronym | Long Description | Author(s) | Year, Journal | Frequency | Cat.Data | Cat.Economic |
|---|---|---|---|---|---|---|
| FR | Pension Funding Status | Franzoni and Marin | 2006, JF | monthly | Accounting | composite accounting |
| BetaFP | Frazzini-Pedersen Beta | Frazzini and Pedersen | 2014, JFE | monthly | Price | other |
| High52 | 52 week high | George and Hwang | 2004, JF | monthly | Price | momentum |
| IndMom | Industry Momentum | Grinblatt and Moskowitz | 1999, JFE | monthly | Price | momentum |
| PctAcc | Percent Operating Accruals | Hafzalla, Lundholm, Van Winkle | 2011, AR | yearly | Accounting | accruals |
| PctTotAcc | Percent Total Accruals | Hafzalla, Lundholm, Van Winkle | 2011, AR | yearly | Accounting | accruals |
| tang | Tangibility | Hahn and Lee | 2009, JF | yearly | Accounting | asset composition |
| Coskewness | Coskewness | Harvey and Siddique | 2000, JF | monthly | Price | risk |
| RoE | net income / book equity | Haugen and Baker | 1996, JFE | yearly | Accounting | profitability |
| VarCF | Cash-flow to price variance | Haugen and Baker | 1996, JFE | monthly | Accounting | cash flow risk |
| VolMkt | Volume to market equity | Haugen and Baker | 1996, JFE | monthly | Trading | volume |
| VolumeTrend | Volume Trend | Haugen and Baker | 1996, JFE | monthly | Trading | volume |
| AnalystRevision | EPS forecast revision | Hawkins, Chamberlin, Daniel | 1984, FAJ | monthly | Analyst | earnings |
| Mom12mOffSeason | Momentum without the seasonal part | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomOffSeason | Off season long-term reversal | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomOffSeason06YrPlus | Off season reversal years 6 to 10 | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomOffSeason11YrPlus | Off season reversal years 11 to 15 | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomOffSeason16YrPlus | Off season reversal years 16 to 20 | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomSeason | Return seasonality years 2 to 5 | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomSeason06YrPlus | Return seasonality years 6 to 10 | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomSeason11YrPlus | Return seasonality years 11 to 15 | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomSeason16YrPlus | Return seasonality years 16 to 20 | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| MomSeasonShort | Return seasonality last year | Heston and Sadka | 2008, JFE | monthly | Price | momentum |
| NOA | Net Operating Assets | Hirshleifer et al. | 2004, JAE | yearly | Accounting | asset composition |

Continued on next page

| Acronym | Long Description | Author(s) | Year, Journal | Frequency | Cat.Data | Cat.Economic |
|---|---|---|---|---|---|---|
| dNoa | change in net operating assets | Hirshleifer, Hou, Teoh, Zhang | 2004, JAE | yearly | Accounting | investment |
| EarnSupBig | Earnings surprise of big firms | Hou | 2007, RFS | quarterly | Accounting | delayed processing |
| IndRetBig | Industry return of big firms | Hou | 2007, RFS | monthly | Price | delayed processing |
| PriceDelayRsq | Price delay r square | Hou and Moskowitz | 2005, RFS | monthly | Price | delayed processing |
| PriceDelaySlope | Price delay coeff | Hou and Moskowitz | 2005, RFS | monthly | Price | delayed processing |
| PriceDelayTstat | Price delay SE adjusted | Hou and Moskowitz | 2005, RFS | monthly | Price | delayed processing |
| STreversal | Short term reversal | Jegadeesh | 1989, JF | monthly | Price | short-term reversal |
| RevenueSurprise | Revenue Surprise | Jegadeesh and Livnat | 2006, JFE | quarterly | Accounting | sales growth |
| Mom12m | Momentum (12 month) | Jegadeesh and Titman | 1993, JF | monthly | Price | momentum |
| Mom6m | Momentum (6 month) | Jegadeesh and Titman | 1993, JF | monthly | Price | momentum |
| ChangeInRecommendation | Change in recommendation | Jegadeesh et al. | 2004, JF | monthly | Analyst | recommendation |
| OptionVolume1 | Option to stock volume | Johnson and So | 2012, JFE | monthly | Trading | volume |
| OptionVolume2 | Option volume to average | Johnson and So | 2012, JFE | monthly | Trading | volume |
| BetaTailRisk | Tail risk beta | Kelly and Jiang | 2014, RFS | monthly | Price | risk |
| fgr5yrLag | Long-term EPS forecast | La Porta | 1996, JF | monthly | Analyst | earnings |
| CF | Cash flow to market | Lakonishok, Shleifer, Vishny | 1994, JF | monthly | Accounting | valuation |
| MeanRankRevGrowth | Revenue Growth Rank | Lakonishok, Shleifer, Vishny | 1994, JF | yearly | Accounting | sales growth |
| RDS | Real dirty surplus | Landsman et al. | 2011, AR | yearly | Accounting | composite accounting |
| Tax | Taxable income to income | Lev and Nissim | 2004, AR | yearly | Accounting | tax |
| RDcap | R&D capital-to-assets | Li | 2011, RFS | yearly | Accounting | asset composition |
| zerotrade | Days with zero trades | Liu | 2006, JFE | monthly | Trading | liquidity |
| zerotradeAlt1 | Days with zero trades | Liu | 2006, JFE | monthly | Trading | liquidity |
| zerotradeAlt12 | Days with zero trades | Liu | 2006, JFE | monthly | Trading | liquidity |
| ChEQ | Growth in book equity | Lockwood and Prombutr | 2010, JFR | yearly | Accounting | investment |

Continued on next page

| Acronym | Long Description | Author(s) | Year, Journal | Frequency | Cat.Data | Cat.Economic |
|---|---|---|---|---|---|---|
| EarningsStreak | Earnings surprise streak | Loh and Warachka | 2012, MS | monthly | Accounting | earnings |
| NumEarnIncrease | Earnings streak length | Loh and Warachka | 2012, MS | quarterly | Accounting | earnings |
| GrAdExp | Growth in advertising expenses | Lou | 2014, RFS | yearly | Accounting | investment |
| EntMult | Enterprise Multiple | Loughran and Wellman | 2011, JFQA | monthly | Accounting | valuation |
| CompositeDebtIssuance | Composite debt issuance | Lyandres, Sun and Zhang | 2008, RFS | yearly | Accounting | external financing |
| InvestPPEInv | change in ppe and inv/assets | Lyandres, Sun and Zhang | 2008, RFS | yearly | Accounting | investment |
| Frontier | Efficient frontier index | Nguyen and Swanson | 2009, JFQA | yearly | Accounting | valuation |
| GP | gross profits / total assets | Novy-Marx | 2013, JFE | yearly | Accounting | profitability |
| IntMom | Intermediate Momentum | Novy-Marx | 2012, JFE | monthly | Price | momentum |
| OPLeverage | Operating leverage | Novy-Marx | 2010, ROF | yearly | Accounting | other |
| Cash | Cash to assets | Palazzo | 2012, JFE | quarterly | Accounting | asset composition |
| BetaLiquidityPS | Pastor-Stambaugh liquidity beta | Pastor and Stambaugh | 2003, JPE | monthly | Price | liquidity |
| BPEBM | Leverage component of BM | Penman, Richardson and Tuna | 2007, JAR | monthly | Accounting | leverage |
| EBM | Enterprise component of BM | Penman, Richardson and Tuna | 2007, JAR | monthly | Accounting | valuation |
| NetDebtPrice | Net debt to price | Penman, Richardson and Tuna | 2007, JAR | monthly | Accounting | leverage |
| PS | Piotroski F-score | Piotroski | 2000, AR | yearly | Accounting | composite accounting |
| ShareIss1Y | Share issuance (1 year) | Pontiff and Woodgate | 2008, JF | monthly | Accounting | external financing |
| DelDRC | Deferred Revenue | Prakash and Sinha | 2012, CAR | yearly | Accounting | investment |
| OrderBacklog | Order backlog | Rajgopal, Shevlin, Venkatachalam | 2003, RAS | yearly | Accounting | sales growth |
| DelCOA | Change in current operating assets | Richardson et al. | 2005, JAE | yearly | Accounting | investment |
| DelCOL | Change in current operating liabilities | Richardson et al. | 2005, JAE | yearly | Accounting | external financing |
| DelEqu | Change in equity to assets | Richardson et al. | 2005, JAE | yearly | Accounting | investment |
| DelFINL | Change in financial liabilities | Richardson et al. | 2005, JAE | yearly | Accounting | external financing |
| DelLTI | Change in long-term investment | Richardson et al. | 2005, JAE | yearly | Accounting | investment |

Continued on next page

| Acronym | Long Description | Author(s) | Year, Journal | Frequency | Cat.Data | Cat.Economic |
|---|---|---|---|---|---|---|
| DelNetFin | Change in net financial assets | Richardson et al. | 2005, JAE | yearly | Accounting | investment |
| TotalAccruals | Total accruals | Richardson et al. | 2005, JAE | yearly | Accounting | investment |
| BM | Book to market using most recent ME | Rosenberg, Reid, and Lanstein | 1985, JF | monthly | Accounting | valuation |
| Accruals | Accruals | Sloan | 1996, AR | yearly | Accounting | accruals |
| ChAssetTurnover | Change in Asset Turnover | Soliman | 2008, AR | yearly | Accounting | sales growth |
| ChNNCOA | Change in Net Noncurrent Op Assets | Soliman | 2008, AR | yearly | Accounting | investment |
| ChNWC | Change in Net Working Capital | Soliman | 2008, AR | yearly | Accounting | investment |
| ChInv | Inventory Growth | Thomas and Zhang | 2002, RAS | yearly | Accounting | investment |
| ChTax | Change in Taxes | Thomas and Zhang | 2011, JAR | quarterly | Accounting | tax |
| Investment | Investment to revenue | Titman, Wei and Xie | 2004, JFQA | yearly | Accounting | investment |
| realestate | Real estate holdings | Tuzel | 2010, RFS | yearly | Accounting | asset composition |
| AbnormalAccruals | Abnormal Accruals | Xie | 2001, AR | yearly | Accounting | accruals |
| FirmAgeMom | Firm Age - Momentum | Zhang | 2004, JF | monthly | Price | momentum |

**Table D.1:** The table shows all available characteristics used, the author(s), the year and the journal of publication. In addition, this table shows the update frequency, the data category as well as the economic category.

**Table D.2: Hyperparameters**

| | PPP | DPPP |
|---|---|---|
| L1 penalty | $\lambda \in \{0, 10^{-5}, 10^{-3}\}$ | $\lambda \in \{0, 10^{-5}, 10^{-3}\}$ |
| Learning Rate | 0.001 | 0.001 |
| Dropout | 0 | $D \in \{0, 0.2, 0.4\}$ |
| Batch Size | 12 | 12 |
| Epochs | 200 | 200 |
| Patience | 20 | 20 |
| Ensemble | 0 | 5 |
| Leaky ReLU | – | 0.01 |

This table gives the hyperparameters that we tune. The first column shows the hyperparameters for the linear parametric portfolio policy (PPP). The second column shows the hyperparameters for the deep parametric portfolio policy (DPPP).

**Table D.3: 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 |
| p-value($U_{L_i} - U_{L3}$) | 0.0797 | | 0.3460 | 0.3996 |
| | | | | |
| $\sum \lvert w_i \rvert / N_t * 100$ | 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 |
| $\sum I(w_i < 0)/N_t$ | 0.4838 | 0.4751 | 0.4704 | 0.4607 |
| $\sum \lvert w_{i,t} - w_{i,t-1}^{+} \rvert$ | 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 | 2.4785 | 2.5934 | 2.5634 | 2.6152 |
| p-value($SR_{L_i} - SR_{L3}$) | 0.0824 | | 0.3359 | 0.4053 |
| | | | | |
| $FF5 + Mom\ \alpha$ | 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 optimized for a mean-variance investor with absolute risk aversion of five conditional on 157 firm characteristics. 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 rows show the utility of the investor as well as the bootstrapped one-sided p-value for the difference in utility between the model with 3 layers and the other models. 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 annualized Sharpe ratios and the bootstrapped one-sided p-value for the difference in Sharpe ratios between the model with 3 layers and the other models. 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 D.4: Deep portfolio policy with different network architectures**

| | PPP | DPPP | Frequency | Category |
|---|---|---|---|---|
| Utility | 0.0267 | 0.0469 | 0.0499 | 0.0473 |
| p-value($U_{DPPP} - U_{architecture}$) | | | 0.0574 | 0.3910 |
| | | | | |
| $\sum \lvert w_i \rvert / N_t * 100$ | 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 |
| $\sum I(w_i < 0)/N_t$ | 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 | 1.8070 | 2.5170 | 2.6985 | 2.6101 |
| p-value($SR_{DPPP} - SR_{architecture}$) | | | 0.0318 | 0.1581 |
| | | | | |
| $FF5 + Mom\ \alpha$ | 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 the deep and linear portfolio policies optimized for a mean-variance investor with absolute risk aversion of five conditional on 157 firm characteristics. The regular portfolio policy is a linear specification of Equation (4), 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 rows show the utility of the investor as well as the bootstrapped one-sided p-value for the difference in utility between the DPPP and the "Frequency" and "Category" models, respectively. 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 annualized Sharpe ratios and the bootstrapped one-sided p-value for the difference in Sharpe ratios between the DPPP and the "Frequency" and "Category" models, respectively. 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.

cfr working paper series

centre for financial research cologne

cfr working papers are available for download from www.cfr-cologne.de.

2023

| No. | Author(s) | Title |
|---|---|---|
| 23-01 | F. Simon, S. Weibels, T. Zimmermann | Deep Parametric Portfolio Policies |

2022

| No. | Author(s) | Title |
|---|---|---|
| 22-12 | V. Agarwal, A. Cochardt, V. Orlov | Birth Order and Fund Manager's Trading Behavior: Role of Sibling Rivalry |
| 22-11 | G. Cici, S. Gibson, N. Qin, A. Zhang | The Performance of Corporate Bond Mutual Funds and the Allocation of Underpriced New Issues |
| 22-10 | E. Theissen, C. Westheide | One for the Money, Two for the Show? The Number of Designated Market Makers and Liquidity |
| 22-09 | R. Campbell, P. Limbach, J. Reusche | Once Bitten, Twice Shy: Failed Deals and Subsequent M&A Cautiousness |
| 22-08 | M. Gehde-Trapp, L. Klingler | The Effect of Sentiment on Institutional Investors: A Gender Analysis |
| 22-07 | T. Bauckloh, V. Beyer, C. Klein | Does it Pay to Invest in Dirty Industries? – New Insights on the Shunned-Stock Hypothesis |
| 22-06 | J. Balthrop and G. Cici | Conflicting Incentives in the Management of 529 Plans |
| 22-05 | I. T. Ivanov, T. Zimmermann, N. W. Heinrich | Limits of Disclosure Regulation in the Municipal Bond Market |
| 22-04 | M. Ammann, A. Cochardt, S. Straumann, F. Weigert | Back to the Roots: Ancestral Origin and Mutual Fund Manager Portfolio Choice |
| 22-03 | A. Betzer, J. Gider, P. Limbach | Do Financial Advisors Matter for M&A Pre-Announcement Returns? |
| 22-02 | S. Lesmeister, P. Limbach, P.R. Rau, F. Sonnenburg | Indexing and the Performance-Flow Relation of Actively Managed Mutual Funds |
| 22-01 | T. Bauckloh, C. Klein, T. Pioch, F. Schiemann | Under Pressure: The Link between Mandatory Climate Reporting and Firms' Carbon Performance |

2021

| No. | Author(s) | Title |
|---|---|---|
| 21-11 | V. Agarwal, H. Ren, K. Shen, H. Zhao | Redemption in Kind and Mutual Fund Liquidity Management |
| 21-10 | N.C. Brown, W. B. Elliott, R. Wermers, R. M. White | News or noise: Mobile internet technology and stock market activity |
| 21-09 | V. Agarwal, B. Barber, S. Cheng, A. Hameed, A. Yasuda | Private Company Valuations by Mutual Funds |
| 21-08 | T.G. Bali, H. Beckmeyer, M. Moerke, F. Weigert | Option Return Predictability with Machine Learning and Big Data |
| 21-07 | F. Chabi-Yo, M. Huggenberger, F. Weigert | Multivariate Crash Risk |
| 21-06 | V. Agarwal, H. Aslan, L. Huang, H. Ren | Political Uncertainty and Household Stock Market Participation |
| 21-05 | G. Cici, P. Zhang | On the Valuation Skills of Corporate Bond Mutual Funds |
| 21-04 | V. Agarwal, P. Hanouna, R. Moussawi, C.W. Stahel | Do ETFs Increase the Commonality in Liquidity of Underlying Stocks? |
| 21-03 | M. Saglam, T. Tuzun, R. Wermers | Do ETFs Increase Liquidity? |
| 21-02 | C. Andres, D. Bazhutov, D. Cumming, P. Limbach | Does Speculative News Hurt Productivity? Evidence from Takeover Rumors |
| 21-01 | T.G. Bali, F. Weigert | Hedge Funds and the Positive Idiosyncratic Volatility Effect |

This document only covers the most recent CFR working papers. A full list can be found at www.cfr-cologne.de.

**centre for financial research**
cfr/university of cologne
Albertus-Magnus-Platz
D-50923 cologne
fon +49(0)221-470-6995
fax +49(0)221-470-3992
kempf@cfr-cologne.de
www.cfr-cologne.de