Structural Deep Learning in Conditional Asset Pricing[*]

Jianqing Fan[†] Zheng Tracy Ke[‡] Yuan Liao[§]
Andreas Neuhierl [¶]

Abstract

We develop new structural nonparametric methods for estimating conditional asset pricing models using deep neural networks. Our method is guided by economic theory and employs time-varying conditional information on alphas and betas carried by firm-specific characteristics. Contrary to many applications of neural networks in economics, we open the "black box" of machine learning predictions by incorporating finance theory into the learning, and provide an economic interpretation of the successful predictions obtained from neural networks, by decomposing the neural predictors as risk-related and mispricing components. Our estimation method starts with period-by-period cross-sectional deep learning, followed by local PCAs to capture time-varying features such as latent factors of the model. We formally establish the asymptotic theory of the structural deep-learning estimators, which apply to both in-sample fit and out-of-sample predictions. We also illustrate the "double-descent-risk" phenomena associated with over-parametrized predictions, which justifies the use of over-fitting machine learning methods.

Key words: factor pricing model, neural network, double descent, alphas, characteristics, risk premium

[*] We thank Torben Andersen, Svetlana Bryzgalova, Carter Davis, Victor DeMiguel, Andrew Patton, Robert Jarrow, Robert Korajczyk, Oliver Linton, Viktor Todorov, Guofu Zhou, and seminar and conference participants at Northwestern University, IDC Herzlyia, University of Washington, University of Virginia, Rutgers University, Syracuse University, London Business School, Cornell University, AI & Big Data in Financial Research Forum, RCEA Big Data & Machine Learning Conference, SFS Calvacade, and Griffin Applied Economics Incubator Conference on Machine Learning Across Disciplines: New Theoretical Developments for valuable comments.

[†] Department of ORFE, Princeton University. jqfan@princeton.edu.

[‡] Department of Statistics, Harvard University. zke@fas.harvard.edu

[§] Department of Economics, Rutgers University. yuan.liao@rutgers.edu

[¶] Olin Business School, Washington University in St. Louis, andreas.neuhierl@wustl.edu

1 Introduction

In this paper, we develop new nonparametric methods to obtain economic interpretations of asset return predictions obtained from deep neural networks. Our analysis is guided by financial economic theory. We use mild economic structure of asset pricing models and develop econometric theories for interpreting each of the components for predicting asset returns. Deep learning methods have proven to be among the most successful approaches for high dimensional and unstructured prediction problems with little "curse of dimensionality" in implementation. They have been shown to adapt automatically low-dimensional structures when unknown functions are compositions of low-dimensional ones such as additive or bivariate interaction models (Kohler and Langer, 2021). Despite of its popularity in the analysis of financial market data, however, deep learning has often been criticized as a *black box*, i.e. we see the inputs and the outputs, but we do not know enough about the structure of the underlying problem. Neither do we know the source of the predictive power (mispricing vs. risk premium), nor it is clear to us whether there are nontrivial portions of noises that contain little predictive power inside the black box.

We take the success of deep learning methods as given, i.e. we do not aim to produce better predictions by modifying the components of deep learning architecture in asset pricing. Our primary contribution is to open the black box by joining rigorous asymptotic theory with financial economic theory. Thereby we obtain economic understanding for *why* deep learning models have been shown to produce successful prediction in financial economics and *how* to improve their prediction powers. More concretely, our framework admits a structural decomposition of the predictions obtained from deep neural networks into compensation for risk and possible mispricing. In addition, we also characterize the temporal evolution of these components. In order to obtain these results, we only need to impose mild economic restrictions on the data generating process - in particular we merely assume that returns follow a factor model and that some of the observed variables are informative about factor loadings or mispricing.

Our approach features dynamics for the risk related components as well as the mispricing component. We use the rich information in a large set of time-varying characteristics to estimate mispricing and factor exposures more efficiently. The dynamics can be driven by two sources. First, we allow the time-variation in characteristics to map directly into the time-variation of alphas and betas. Second, we allow the functions that map characteristics into alphas and betas to be varying over time. As the characteristic information is purely cross-sectional, i.e. only the cross-sectional ranking rather than the raw value matters, featuring a time-varying mapping from characteristics to alphas and betas is crucial to capture important empirical facts such as the alpha decay (McLean and Pontiff, 2016). The alpha decay, i.e. the tendency that abnormal returns tend to diminish over time, will arise naturally in a setting in which investors learn about predictors and take advantage of arbitrage opportunities to (eventually) eliminate them. In addition, the mapping from characteristics to betas and mispring has been assumed to be time-invariant in the recent literature, with only characteristics being dynamics. In contrast, we allow time-varying nonparametric mappings, and argue that the alpha decay in our empirical study also demonstrates the importance of incorporating time-varying mappings.

We also contribute to the econometric literature by rigorously deriving the rate of convergence for structural deep learning estimation of (predicted) returns, alphas and compensation for risk. A novel theoretical result is that out-of-sample rates of convergence are also derived for predicted alphas and risk-related return components using deep neural networks. To our best knowledge, this is the first time such results are established in the asset pricing literature, and even in the econometric literature as most existing results concentrate on in-sample convergence. The derived rates of convergence depend on three key ingredients, (1) the approximation error for the unknown functions using deep neural networks (DNN), (2) the complexity of the neural network in which the model is being trained, and (3) the degree of time-series dynamics of nonparametric functions that measures the transition from in-sample to out-of-sample periods. Our theory shows that the predictive error naturally pins down these three sources of learning errors. In particular, while the first two components are common in the deep learning theory though only in the nonstructural nonparametric setting, the third component also appears naturally when we apply time localization such as kernel smoothing for the time-varying principal components analysis (PCA).

We apply our structural decomposition to the standard CRSP/Compustat panel. For the in-sample decomposition, we find that about 90% of the explained variation can be attributed to risk. Within this 90%, the bulk of the explanatory power is driven by the factor realization ($\approx 95\%$) and roughly 5% by the long-term risk premium. Up to 1% of the explained in-sample return can be attributed to mispricing.

Meanwhile, our analysis provides new insights for the out-of-sample prediction of returns in the cross-section: the predictive success is driven almost exclusively by the risk premium component and the mispricing part. In addition, the predictability is nearly entirely attributed to the risk premium for large firms, and is attributed to both risk premium (about 76%) and mispricing (about 24%) for small firms. Meanwhile, the factor realization is essentially not predictable as the factor returns are themselves excess returns, which are known to have very low persistence in the time series. Due to the presence of *old* factor realizations, the out-of-sample R^2 is negative for the standard plugin forecast using DNN estimates in all scenarios under study. This reveals that we can obtain greater predictive accuracy by focusing only on the risk premium component and the mispricing component rather than the prevailing practice of plugging new data into the estimated model. The former is possible due to our novel econometric methods that allow us to consistently estimate each component in the structural conditional asset pricing model. In contrast, the standard "plugin" approach will lead to a suboptimal prediction which is "noised up" and "misguided" by the past factor realization.

Finally, the neural networks that we empirically implement contain multilayers with minimum degrees of regularizations, which may encounter the overfitting issues. To this extent, we also numerically document an interesting phenomenon known as "double descent" for machine learning predictions. That is, in contrast to the traditional statistical wisdom on the bias-variance tradeoff, the prediction risk starts decreasing as the number of trained parameters exceeds the sample size and continues growing. Using simulated data, we illustrate this phenomenon in one of the best known economic predictive models as in Stock and Watson (2002). Recently, a similar "virtual of complexity" is also studied in the asset pricing context by Kelly et al. (2021), who document that Sharpe ratios of machine learning portfolios may increase for overparametrized models.

It is important to point out that while our analysis accomplishes sensible improvements on the out-of-sample forecast, it should be regarded as a means to provide a structural decomposition of the machine learning forecast. Our main goal is to reach an economically meaningful interpretation of source of predictability for asset returns.

Related Literature

Deep learning models have achieved remarkable success in science and engineering. The overall literature is too vast to be summarized here. Therefore, we refer to Fan et al. (2021) for a broader overview and only mention the most closely connected papers. Theoretically, deep learning has been shown to be able to approximate a broad class of highly nonlinear functions, see, e.g. Mhaskar et al. (2016); Rolnick and Tegmark (2017); Lin et al. (2017); Shen et al. (2021). Statistically, Bauer and Kohler (2019), Schmidt-Hieber (2020), Kohler and Langer (2021) and Fan et al. (2022) demonstrate the ability of deep neural networks for circumventing the curse of dimensionality arising from high-dimensional predictors in nonparametric regression with automatic adaptation to low-dimensional structures. Farrell et al. (2021) showed that rates of convergence for deep neural nets are sufficiently fast to establish valid second-step inference after first-step estimation with deep learning such as treatment effect evaluation. Since the pioneering contributions of Bansal and Viswanathan (1993) and Chen and Ludvigson (2009), machine learning methods have recently been applied in asset pricing frequently and have shown great promise. Freyberger et al. (2020), Gu et al. (2020), Bianchi et al. (2021) and Chen et al. (2020); Guijarro-Ordonez et al. (2021) show that equity and bond return predictions are improved significantly via applications of neural networks relative to linear (or other parametric) models. Gu et al. (2020) conducted extensive comparative studies to illustrate the gain of using these methods.

Our paper also contributes to the large literature and (conditional) factor models in asset pricing. Important early contribution to this literature where made by Chen et al. (1986), Connor and Korajczyk (1986) and Fama and French (1992). More recently, Connor et al. (2012); Fan et al. (2016); Giglio and Xiu (2021); Giglio et al. (2021); Kim et al. (2021) studied the unconditional model in the presence of latent factors. Meanwhile, conditional linear factor models have been popularly used to capture time-varying effects of financial variables and firm-specific characteristics (Shanken, 1990; Ferson and Harvey, 1999; Lettau and Ludvigson, 2001; Ghysels, 1998; Ang and Kristensen, 2012; Gagliardini et al., 2016). To account for dynamic factor betas, Kelly et al. (2019, 2020); Chen et al. (2021); Gu et al. (2019) studied models in which dynamic characteristics are mapped to alphas and betas and the mapping is assumed to be time-invariant. Their findings are consistent with Bakalli et al. (2021) which show that standard factor models can be improved significantly by considering the additional information about risk contained in characteristics. We refer to Gagliardini et al. (2020) for an excellent survey for econometric methodologies for large-dimensional conditional factor models. Parallel to the literature on factor model in asset pricing, a sizeable literature on latent factor modeling has developed in econometrics. Important contributions were made (among many others) by Bai and Ng (2002); Forni et al. (2000); Stock and Watson (2002); Onatski (2012).

2 The Model

2.1 The conditional factor pricing model

We consider the following time-varying factor model with intercepts:

$$y_{it} = \alpha_{i,t-1} + \boldsymbol{\beta}'_{i,t-1}\boldsymbol{\lambda}_{t-1} + \boldsymbol{\beta}'_{i,t-1}(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t) + u_{it}, \quad i \leq N, t \leq T, \tag{2.1}$$

where y_{it} is the excess return of asset i at time t; $\mathbf{f}_t$ is a $K \times 1$ vector of latent factors; $\alpha_{i,t-1}$ and $\boldsymbol{\beta}_{i,t-1}$ respectively denote the (possibly) time-varying alpha and beta of the factor model; $\boldsymbol{\lambda}_{t-1}$ is the vector of factor risk premia which also allow for nontradable factors; u_{it} is the idiosyncratic component. $\alpha_{i,t-1}$ allows the possibility of mispricing for asset i and thus a test of the factor pricing model. Formal assumptions identifiability, and for establishing properties of estimators are discussed in Section 4.

We consider the scenario where alphas and betas can be (partially) explained by a set of individual-specific characteristics. Let $\mathbf{x}_{i,t-1}$ be a d-dimensional vector of observed characteristics associated with stock i. We model

$$\begin{aligned} \alpha_{i,t-1} &= g_{\alpha,t}(\mathbf{x}_{i,t-1}) + \gamma_{\alpha,i,t-1}, \quad \mathbb{E}(\gamma_{\alpha,i,t-1}|\mathbf{x}_{i,t-1}, \mathbf{f}_t) = 0, \\ \boldsymbol{\beta}_{i,t-1} &= g_{\beta,t}(\mathbf{x}_{i,t-1}) + \boldsymbol{\gamma}_{\beta,i,t-1}, \quad \mathbb{E}(\boldsymbol{\gamma}_{\beta,i,t-1}|\mathbf{x}_{i,t-1}, \mathbf{f}_t) = 0. \end{aligned} \tag{2.2}$$

Here $g_{\alpha,t}(\cdot)$ and $g_{\beta,t}(\cdot)$ are time-varying nonparametric functions of characteristics; $\gamma_{\alpha,it}$ and $\boldsymbol{\gamma}_{\beta,it}$ respectively represent the source of alphas and betas that cannot be explained by the characteristics (Fan et al., 2016). This model extends the arbitrage model of Kim et al. (2021) and Li and Linton (2020) to conditional models where the characteristic effects $g_{\alpha,t}(\cdot)$ and $g_{\beta,t}(\cdot)$ should be not only possibly nonlinear but also dynamic. An important feature of this model is that, $\mathbf{x}_{i,t}$ and $\boldsymbol{\gamma}_{it} := (\gamma_{\alpha,it}, \boldsymbol{\gamma}_{\beta,it})$ may vary in different frequencies. So both $\alpha_{i,t}$ and $\boldsymbol{\beta}_{i,t}$ may vary rapidly over time due to the high-frequency change of $\gamma_{\alpha,it}$ and $\boldsymbol{\gamma}_{\beta,it}$.

Machine learning methods, in particular, deep learning, has been successfully employed to predict asset returns using large amount of conditional information in the characteristics. For ease of interpretation, let us consider a framework of period-by-period prediction. Suppose at period t, researchers obtain a prediction function $\widehat{m}_t(\cdot)$ by applying deep neural networks on the cross-sectional data $\{(y_{i,t}, \mathbf{x}_{i,t-1}) : i = 1, \cdots, N\}$. They then predict $y_{i,t+1}$ by substituting $\mathbf{x}_{i,t}$ to obtain:

$$\widehat{y}_{i,t+1|t} := \widehat{m}_t(\mathbf{x}_{i,t}).$$

The learned function $\widehat{m}_t(\cdot)$ is an estimate of the conditional mean function $m_t^0(\mathbf{x}) = \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1} = \mathbf{x}, \mathbf{f}_t)$ in the cross-sectional regression model:

$$y_{it} = m_t^0(\mathbf{x}_{i,t-1}) + e_{it}, \quad i = 1, \cdots, N, \tag{2.3}$$

where e_{it} is the regression noise.[1] Note that the notation $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1} = \mathbf{x}, \mathbf{f}_t)$ that defines the conditional mean further conditions on $\mathbf{f}_t$, which regards $\mathbf{f}_t - \mathbb{E}\mathbf{f}_t$ in (2.1) as a *fixed parameter rather than a regressor* at period t. Keep in mind that when estimating $m_t^0(\cdot)$ at period t, we run a cross-sectional regression only by regressing on $\mathbf{x}_{i,t-1}$, treating any components arising from $\mathbf{f}_t$ as unknown, yet fixed parameters.

While it has been established that machine learning models as (2.3 are very successful in prediction, little interpretation has been given regarding the source of predictability in these models. In this paper, we aim to open the "black box" of the machine learning prediction model (2.3) and provide an economically insightful interpretation of the successful prediction obtained using machine learning methods.

[1] Latent factors $\mathbf{f}_t$ are given as they are already realized across assets at time t. Pooled regression is also often used in the literature: $\min_m \sum_t \sum_i (y_{it} - m(\mathbf{x}_{i,t-1}))^2$. In conditional pricing models however, the pooled regression does not incorporate the time-varyingness of betas and pricing errors, which would not be consistent when characteristics vary over time even if the functions $g_{\alpha,t}()$ and $g_{\beta,t}()$ do not vary. In conditional models, one may replace the pooled regression by "localized pooling": $\min_m \sum_t \sum_i (y_{it} - m(\mathbf{x}_{i,t-1}))^2 K(\frac{t-s}{Th})$, where $K(\frac{t-s}{Th})$ is a kernel function to incorporate the time-varyingness as we shall explain later in this paper. This localized pooling would give a similar estimator for $g_{\alpha,t}(\mathbf{x}_{i,t-1}) + g_{\beta,t}(\mathbf{x}_{i,t-1})'\boldsymbol{\lambda}_{i-1}$ as in our approach, which removes the factor shocks, and is equivalent to ours in linear models. However, unlike our method, it does not produce estimators for factor realizations, betas, or pricing errors. In addition, it combines time smoothing and DNN in a single step, which does not have the flexibility for choosing bandwidths as our method does.

2.2 Structural machine learning predictions

In this section we present decompositions of both in-sample estimation and out-of-sample predictions. While the decompositions are obtained for a generic machine learning method, in this paper we are primarily interested in the deep neural network estimation, due to its various advantages on multi-dimensional nonparametric function estimation, including representation powers and the arts of scalable implementations in high-dimension.

The conditional mean function $m_t^0(\mathbf{x}) = \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1} = \mathbf{x}, \mathbf{f}_t)$ can be presented as:

$$\begin{aligned} m_t^0(\mathbf{x}) &= g_{\alpha,t}(\mathbf{x}) + g_{\mathrm{riskP},t}(\mathbf{x}) + g_{\mathrm{factor},t}(\mathbf{x}), \\ g_{\mathrm{riskP},t}(\mathbf{x}) &= g_{\beta,t}(\mathbf{x})'\boldsymbol{\lambda}_{t-1}, \\ g_{\mathrm{factor},t}(\mathbf{x}) &= g_{\beta,t}(\mathbf{x})'(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t). \end{aligned} \tag{2.4}$$

Let $\widehat{g}_{\alpha,t}$, $\widehat{g}_{\mathrm{riskP},t}$ and $\widehat{g}_{\mathrm{factor},t}$ respectively denote the estimated functions of $g_{\alpha,t}$, $g_{\mathrm{riskP},t}$ and $g_{\mathrm{factor},t}$, whose construction will be clear in the next section. Then the conditional factor model yields the following decompsitions of the in-sample fit at each time t:

In-sample decomposition:

spot expected return:

$$\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t) = g_{\alpha,t}(\mathbf{x}_{i,t-1}) + g_{\mathrm{riskP},t}(\mathbf{x}_{i,t-1}) + g_{\mathrm{factor},t}(\mathbf{x}_{i,t-1}),$$

long-term expected return:

$$\begin{aligned} \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}) &= \mathbb{E}\left(\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)\middle|\mathbf{x}_{i,t-1}\right) \\ &= \underbrace{g_{\alpha,t}(\mathbf{x}_{i,t-1})}_{\text{mispricing}} + \underbrace{g_{\mathrm{riskP},t}(\mathbf{x}_{i,t-1})}_{\text{risk premium}}, \end{aligned}$$

$$\begin{aligned} \text{returns:} \quad y_{it} &\approx \underbrace{\widehat{g}_{\alpha,t}(\mathbf{x}_{i,t-1}) + \widehat{g}_{\mathrm{riskP},t}(\mathbf{x}_{i,t-1}) + \widehat{g}_{\mathrm{factor},t}(\mathbf{x}_{i,t-1})}_{\approx \widehat{y}_{it}} + e_{it}, \\ \widehat{y}_{it} &:= \widehat{m}_t(\mathbf{x}_{i,t-1}) \end{aligned} \tag{2.5}$$

where $e_{it} = \gamma_{\alpha,i,t-1} + \boldsymbol{\gamma}'_{\beta,i,t-1}\boldsymbol{\lambda}_{t-1} + \boldsymbol{\gamma}'_{\beta,i,t-1}(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t) + u_{it}$ and $\widehat{y}_{it}$ is the in-sample expected return by plugging the in-sample characteristic $\mathbf{x}_{i,t-1}$ into the machine learning function. This decomposition takes into account the fact that the factors in the in-sample period have been realized and are estimable under some mild conditions.

The first equality is what the model implies, which leads to a decomposition of what we call “spot expected return”. It is clear from the decomposition that the spot return depends on realized factor returns, but does not depend on the components in the betas and alphas that are orthogonal to the characteristics ($\boldsymbol{\gamma}_{it}$), neither does it depend on idiosyncratic errors (u_{it}). Later on, we will show that the spot expected return can be learned by period-by-period cross-sectional deep neural networks (DNN), via regressing returns on characteristics.

The second expected return, which we call “long-term expected return”, depends only on characteristics and the factor risk premia, so can be learned by taking the local time-series average (around time t) of the spot expected return. It clearly shows that the conditional expected return evolves with characteristics through two components. Finally, the third (approximate) equality shows that plugging the in-sample $\mathbf{x}_{i,t-1}$ into the DNN estimated function yields the in-sample fit $\widehat{y}_{it}$ for the realized return.

Next we discuss the out-of-sample decomposition. The realized out-of-sample returns, $y_{i,t+1}$, has the following decomposition.

Out-of-sample decomposition

$$\begin{aligned} y_{i,t+1} &= g_{\alpha,t+1}(\mathbf{x}_{i,t}) + g_{\text{riskP},t+1}(\mathbf{x}_{i,t}) + g_{\text{factor},t+1}(\mathbf{x}_{i,t}) + e_{i,t+1} \\ &\approx g_{\alpha,t}(\mathbf{x}_{i,t}) + g_{\text{riskP},t}(\mathbf{x}_{i,t}) + g_{\beta,t}(\mathbf{x}_{i,t})'(\mathbf{f}_{t+1} - \mathbb{E}\mathbf{f}_t) + e_{i,t+1} \\ &= m_t^0(\mathbf{x}_{i,t}) + g_{\beta,t}(\mathbf{x}_{i,t})'(\mathbf{f}_{t+1} - \mathbf{f}_t) + e_{i,t+1}, \end{aligned} \tag{2.6}$$

where $\approx$ holds if the functions $g_{\alpha,t}(\cdot)$ and $g_{\text{riskP},t}(\cdot)$ change slowly over time. Therefore, the return to be predicted approximately equals the conditional mean function $m_t^0(\mathbf{x})$ evaluated at the new characteristic $\mathbf{x} = \mathbf{x}_{i,t}$, plus two noises: $e_{i,t+1}$ being the idiosyncratic noise in the mean regression, and $g_{\beta,t}(\mathbf{x}_{i,t})'(\mathbf{f}_{t+1} - \mathbf{f}_t)$ arising from the factor innovation.

While the last line of (2.6) justifies the use of machine learning predictor $\widehat{y}_{i,t+1|t}$ to predict returns, this predictor also contains the factor shocks which does not contribute to predictability. In fact, our approach admits a refined out-of-sample decomposition that specifies all components constituting $\widehat{y}_{i,t+1|t}$. Equations (2.4) and (2.6) yield:

$$\begin{aligned} \widehat{y}_{i,t+1|t} &= \widehat{g}_{\alpha,t}(\mathbf{x}_{i,t}) + \widehat{g}_{\text{riskP},t}(\mathbf{x}_{i,t}) + \widehat{g}_{\text{factor},t}(\mathbf{x}_{i,t}), && (2.7) \\ y_{i,t+1} &= \widehat{g}_{\alpha,t}(\mathbf{x}_{i,t}) + \widehat{g}_{\text{riskP},t}(\mathbf{x}_{i,t}) + \underbrace{g_{\text{factor},t+1}(\mathbf{x}_{i,t}) + e_{i,t+1}}_{\xi_{i,t+1}} + o_P(1), && (2.8) \end{aligned}$$

where the $o_P(1)$ term converges to zero when $N, T \to \infty$ and $\widehat{y}_{i,t+1|t}$ is the out-of-sample machine learning predicted return by plugging the "new" $\mathbf{x}_{i,t}$ into the machine learning function. The decomposition Equation (2.7) justifies two main sources of the predicting power inside $\widehat{y}_{i,t+1|t}$: the mispricing component $\widehat{g}_{\alpha,t}(\mathbf{x}_{i,t})$ and the risk-premia component $\widehat{g}_{\text{riskP},t}(\mathbf{x}_{i,t})$. But the last term, $\widehat{g}_{\text{factor},t}(\mathbf{x}_{i,t}) \approx g_{\beta,t}(\mathbf{x}_{i,t})'(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t)$, has little predictive power. Indeed, (2.8) classifies two prediction noises:

$$\begin{aligned} \xi_{i,t+1} &:= g_{\text{factor},t+1}(\mathbf{x}_{i,t}) + e_{i,t+1}, \text{ where} \\ g_{\text{factor},t+1}(\mathbf{x}_{i,t}) &:= g_{\beta,t+1}(\mathbf{x}_{i,t})'(\mathbf{f}_{t+1} - \mathbb{E}\mathbf{f}_{t+1}) \text{ is future factor realization,} \\ e_{i,t+1} &:= \gamma_{\alpha,it} + \boldsymbol{\gamma}'_{\beta,it}\boldsymbol{\lambda}_t + \boldsymbol{\gamma}'_{\beta,it}(\mathbf{f}_{t+1} - \mathbb{E}\mathbf{f}_{t+1}) + u_{i,t+1} \\ &\quad\ \text{is orthogonal and idiosyncratic components.} \end{aligned}$$

The future factor realization is often either nearly martingale difference or very weakly dependent on $\widehat{g}_{\text{factor},t}(\mathbf{x}_{i,t})$, so is unpredictable. This leads to a major difference between the in-sample and out-of-sample decompositions. [2]

We aim to estimate each component of the above decompositions, both in-sample and out-of-sample. In particular, we shall apply DNN to respectively estimate functions $g_{\alpha,t+1}(\cdot)$ and $g_{\text{riskP},t+1}(\cdot)$. Our method serves two goals of this paper: first, it allows to quantify the role of each component in constituting asset returns; second, it allows to remove the factor-realizations and use only the alphas and risk-premium components for predictions. As one of the theoretical results, we show that the out-of-sample true returns satisfy (2.8) for the future period $T+1$, where $\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T})$ and

[2] In traditional prediction approaches, the past realizations of latent factors are used to predict the future factor innovations, which doubles the prediction variance and is worse than the predictor zero when the factor innovations are nearly martingale. Our structural decomposition and learning techniques allow us to untangle the contributions of factor realizations and replace them by zero, the best predictor when the underlying factor process is martingale. If there are nontrivial temporal correlations among factor realizations, then we can fit an autoregressive model: $g_{\text{factor},t+1}(\mathbf{x}_{i,t}) = \rho g_{\text{factor},t}(\mathbf{x}_{i,t-1}) + \epsilon_{f,t+1}$. The first term is predictable at period t, while the new shock $\epsilon_{f,t+1}$ has smaller volatility than $g_{\text{factor},t+1}(\mathbf{x}_{i,t})$ does. Hence adding $g_{\text{factor},t}(\mathbf{x}_{i,t-1})$ may improve the out-of-sample prediction in applications when ρ is relatively large. Unfortunately factor realizations in asset pricing tend to have short memories whose autocorrelations are very low in magnitude.

$\widehat{g}_{\mathrm{riskP},T}(\mathbf{x}_{i,T})$ are respectively the DNN-predictions of mispricing and risk premium. Yet, the remaining component $\xi_{i,T}$ satisfies $\mathbb{E}(\xi_{i,T+1}|\mathcal{F}_T) = 0$ for the filtration information up to the prediction period. This shows: (i) the DNN-prediction can capture the predictable components of future returns: mispricing and risk premium; (ii) other than the two DNN-prediction components, other components in future returns are not predictable.

3 The Methodology

In this section we describe our methods for estimating the components of the return decomposition, i.e. the mispricing component, factors, risk exposures and risk premia. Define

$$\begin{aligned}
\mathbb{E}(\mathbf{Y}_t|\mathbf{X}_{t-1}, \mathbf{f}_t) &= (\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t) : i \leq N), \quad N \times 1 \text{ vector},\\
\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1}) &= (g_{\beta,t}(\mathbf{x}_{i,t-1}) : i \leq N), \quad N \times \dim(\mathbf{f}_t) \text{ matrix},
\end{aligned}$$

which are the matrices stacking the high-dimensional spot expected returns and characteristic-betas at each period. The building block of our methodology is the following local approximation: By (2.5), fix a period t, for all periods s that are "close to t":

$$\begin{aligned}
&\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1}, \mathbf{f}_s) - \mathbb{E}\left(\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1}, \mathbf{f}_s)\middle|\mathbf{X}_{s-1}\right)\\
= \ &\mathbf{G}_{\beta,s}(\mathbf{X}_{s-1})(\mathbf{f}_s - \mathbb{E}\mathbf{f}_s)\\
\approx \ &\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1})(\mathbf{f}_s - \mathbb{E}\mathbf{f}_s),
\end{aligned}$$

where "≈" follows from the assumption that characteristic-betas are varying much slower than factor realizations. Therefore, columns of $\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1})$ are locally proportional to the top eigenvectors (left singular vectors) of the matrix of demeaned spot expected returns. This motivates us to estimate the model in three steps:

(1) apply DNN to estimate the nonparametric spot returns $\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1}, \mathbf{f}_s)$;

(2) apply local averages to estimate the long-term returns $\mathbb{E}\left(\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1}, \mathbf{f}_s)\middle|\mathbf{X}_{s-1}\right)$;

(3) apply local PCA to estimate betas $\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1})$.

However, estimation details are technically challenging due to the possibly high nonlinearity and time-varying nature of long-term expected returns and characteristic-betas. These issues are to be addressed using deep neural networks and kernel smoothing. In the following subsections we outline additional details of our methodology. Formal properties are established in Section 4.

3.1 Learning spot returns by deep neural networks

The first step of our method is to estimate the spot expected return function

$$m_t^0(\mathbf{x}) := \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1} = \mathbf{x}, \mathbf{f}_t).$$

As the latent factors are realized at time t, this conditional mean function can be estimated via the following cross sectional regression:

$$y_{it} = m_t^0(\mathbf{x}_{i,t-1}) + e_{it}, \quad \mathbb{E}(e_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t) = 0, \quad i = 1, \cdots, N.$$

Because of the nonlinearity and high-dimensionality of $\mathbf{x}_{i,t-1}$, the deep neural network is an appealing nonparametric machine learning technique to employ here. Deep learning can be viewed as a family of nonlinear statistical models that are able to encode highly nontrivial representations of data. A prototypical example is a *feed-forward neural network* with J layers, which is a family of functions taking form:

$$m(\mathbf{x}) = \sigma_J(\boldsymbol{\theta}_J h_{J-1}(\mathbf{x})), \quad h_{j-1}(\mathbf{x}) = \sigma_{j-1}(\boldsymbol{\theta}_{j-1} h_{j-2}(\mathbf{x})), \cdots, h_0(\mathbf{x}) = \mathbf{x}$$

where the parameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_J)$ with $\boldsymbol{\theta}_j \in \mathbb{R}^{d_j \times d_{j-1}}$, and $\sigma_j : \mathbb{R}^{d_j} \to \mathbb{R}^{d_{j+1}}$ is a vector-value nonlinear activation functions, usually the same across components and layers. One of the popularly used activation functions is known as ReLu, defined as $\sigma(x) = \max(0, x)$. The number of *neurons* being used in layer j, denoted by d_j, is called the width of that layer. For presentational simplicity, we shall just assume $d_1 = \cdots = d_J = L$, but in practice they can be chosen to vary across layers.

Let $\mathcal{M}_{J,L}$ denote the neural network space with depth J and width L that collect functions taking the form $m(\mathbf{x})$ parametrized by $\boldsymbol{\theta}$. We estimate $m_t^0(\cdot)$ period-by-period via

$$\widehat{m}_t(\cdot) = \arg\min_{m \in \mathcal{M}_{J,L}} \sum_{i=1}^{N} (y_{it} - m(\mathbf{x}_{i,t-1}))^2,$$

where we drop the superscript 0 for simplicity. Let $\widehat{\mathbf{m}}_t(\mathbf{X}_{t-1})$ be the $N \times 1$ vector of fitted values $\{\widehat{m}_t(\mathbf{x}_{i,t-1})\}$, which estimates the N-dimensional spot expected return at given time t. It can be shown that

$$\widehat{m}_t(\mathbf{x}_{i,t-1}) \to^P g_{\alpha,t}(\mathbf{x}_{i,t-1}) + g_{\mathrm{riskP},t}(\mathbf{x}_{i,t-1}) + g_{\beta,t}(\mathbf{x}_{i,t-1})'(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t), \quad \text{as } N \to \infty.$$

which demonstrates at least three appealing features of the period-by-period DNN for expected returns: (1) it eliminates idyosyncratic noises with vanishing statistical errors that only depends on the cross-sectional dimensions, so works well even if T is finite; (2) it retains the factor realizations, and (3) it is not affected by the nature of time-varyingness under conditional models.

As has been documented in the literature, learning using deep neural networks brings several advantages compared to classical nonparametric methods. First, as an important statistical advantage (Bauer and Kohler, 2019; Schmidt-Hieber, 2020), the art of deep-learning alleviates the curse of dimensionality arising from the high-dimensional nonparametric regression, and finds an estimator with good generalization power. See more details in the theory section. Secondly, both the asymptotic theoretical performance and the finite sample performance of deep neural networks are much less sensitive to the choice of tuning parameters than the kernel-based methods. Finally, it has been proved beneficial to use deep neural networks (Telgarsky, 2016), which can approximate various classes of functions (Yarotsky, 2017, 2018; Shen et al., 2021; Lu et al., 2020; Shen et al., 2022). [3]

3.2 Learning long-term returns by kernel smoothing

The next step is to estimate the long-term conditional expected return

$$\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}) = \mathbb{E}\left(\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)\middle|\mathbf{x}_{i,t-1}\right).$$

We apply the kernel smoothing method following the seminal work of Ang and Kristensen (2012). Note that the kernel smoothing technique being employed here is

[3] Telgarsky (2016) shows that a tooth function with $O(2^k)$ oscillations can be expressed as a ReLU-DNN with depth $O(k)$ and $O(1)$ nodes, but needs $O(2^k)$ nodes for a two-layer network to realize it. Using Taylor expansion and approximating polynomials via tooth functions, nonasymptotic approximation error bounds for various classes of functions have been derived in the aforementioned references.

not motivated by the usual nonparametric regression for estimating conditional mean functions. Rather, it is the time-domain smoothing, motivated by the fact that the conditional alpha and beta $g_{\alpha,t}(\mathbf{x}_{i,t})$ and $g_{\beta,t}(\mathbf{x}_{i,t})$ vary slowly across time. We assume that for each individual i, there are twice differentiable functions $m_i(\cdot)$ and $g_i(\cdot)$ so that almost surely, we can write

$$\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}) = m_i\left(\frac{t}{T}\right), \quad g_{\beta,t}(\mathbf{x}_{i,t-1}) = g_i\left(\frac{t}{T}\right).$$

Then, we have $m_i\left(\frac{t}{T}\right) \approx m_i\left(\frac{s}{T}\right)$, $g_i\left(\frac{t}{T}\right) \approx g_i\left(\frac{s}{T}\right)$ for all $\frac{s}{T} \approx \frac{t}{T}$. Thus, we have the local approximation:

$$\begin{aligned} m_s^0(\mathbf{x}_{i,s-1}) &= m_i\left(\frac{s}{T}\right) + g_{\beta,s}(\mathbf{x}_{i,s-1})'(\mathbf{f}_s - \mathbb{E}\mathbf{f}_s) \\ &\approx m_i\left(\frac{t}{T}\right) + g_{\beta,t}(\mathbf{x}_{i,t-1})'(\mathbf{f}_s - \mathbb{E}\mathbf{f}_s), \quad \frac{s}{T} \approx \frac{t}{T}. \end{aligned}$$

Therefore, averaging the DNN functions $\widehat{m}_s(\mathbf{x}_{i,s-1})$ locally over time can lead to a consistent estimation of the long-term expected return $m_i(t/T)$.

To carry out the local-average, we adopt a kernel function $K : [-1, 1] \to [0, \infty)$ with bandwidth h. We estimate the long-term expected return $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})$ by

$$\bar{m}_{i,t} = \frac{1}{Th}\sum_{s=1}^{T} \widehat{m}_s(\mathbf{x}_{i,s-1}) K\left(\frac{s-t}{Th}\right) A_t^{-1}, \quad A_t = \frac{1}{Th}\sum_{s=1}^{T} K\left(\frac{s-t}{Th}\right).$$

This estimator is well motivated from the time-domain nonparametric kernel estimation literature (Fan and Yao, 2003). The estimation performance is not sensitive to the specific choice of kernel functions. In our empirical application, we use the quartic kernel:

$$K(x) = \frac{15}{16}(1 - x^2)^2, \quad -1 \le x \le 1$$

adjusted to alleviate the boundary effects. The boundary kernel we employ reduces the estimation bias on boundary time periods near both $t = 0$ and $t = T$, which is relevant for out-of-sample forecasts. In addition, the bandwidth h controls the time window used to locally average the DNN functions. A small bandwidth means only observations close to t are used in the weighted averages, so the bandwidth controls the bias and variance of the estimator. In particular, as sample size grows, the bandwidth should shrink towards zero at a suitable rate. [4]

In time-varying asset pricing models with observable factors, Ang and Kristensen (2012) applied a similar approach to estimate the unconditional alphas and betas. Different from their approach, we are estimating the *conditional* expected returns *given* characteristics. This is particularly valuable as extant recent literature, e.g. Gagliardini et al. (2016), Chaieb et al. (2021), Kelly et al. (2019). Bakalli et al. (2021) show that standard factor models can be improved significantly by considering the additional information about the variation of risk over time contained in characteristics.

3.3 Local principal components analysis

After respectively estimating the spot and long-term returns, we now discuss how the conditional alphas, betas and risk premia can be estimated. We propose to use local PCA combined with DNN to estimate these quantities in conditional models. As outlined earlier, the difference between the spot and long-term expected returns equals:

$$\begin{aligned}\mathbb{E}(y_{is}|\mathbf{x}_{i,s-1},\mathbf{f}_s)-\mathbb{E}(y_{is}|\mathbf{x}_{i,s-1}) &= g_{\beta,s}(\mathbf{x}_{i,s-1})'(\mathbf{f}_s-\mathbb{E}\mathbf{f}_s)\\ &\approx g_{\beta,t}(\mathbf{x}_{i,t-1})'(\mathbf{f}_s-\mathbb{E}\mathbf{f}_s), \quad \forall\frac{s}{T}\approx\frac{t}{T},\end{aligned} \tag{3.1}$$

which is an approximate *noise-free* factor model locally around period t. Therefore, locally $\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1})$ is approximately the top-eigenvector matrix of

$$\mathrm{var}\left(\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1},\mathbf{f}_s)-\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1})\middle|\mathbf{X}_{s-1}\right)\approx\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1})\mathrm{var}(\mathbf{f}_s|\mathbf{X}_{s-1})\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1})'.$$

Define

$$\begin{aligned}\widehat{\mathbf{m}}_s(\mathbf{X}_{s-1}) &= (\widehat{m}_s(\mathbf{x}_{i,s-1}):i\le N), \quad N\times 1\\ \bar{\mathbf{m}}_s &= (\bar{m}_{i,s}:i\le N), \quad N\times 1.\end{aligned}$$

[4] For the baseline kernel $K(u)$, we adjust it by taking the boundary effect and define the boundary kernel $K_t(u)=1\{|u|\le 1\}[K(u)-a_t]$, where $a_t=\int_{l(t)}^{u(t)}xK_0(x)dx/\int_{l(t)}^{u(t)}x1\{|x|<1\}dx$ is a boundary adjusting constant with $l_t=(1-t)/(Th)$ and $u(t)=(T-t)/(Th)$. Alternatively, the boundary kernel can also be defined as $K_t(u)=K(u)(b_t-uc_t)$, where $b_t=\int_{l(t)}^{u(t)}x^2K(x)dx$ and $c_t=\int_{l(t)}^{u(t)}xK(x)dx$. The latter method is indeed the equivalent kernel induced by the local linear fitting (Fan and Gijbels, 1996).

With the demeaned expected return $\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1}, \mathbf{f}_s) - \mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1})$ estimated by $\widehat{\mathbf{m}}_s(\mathbf{X}_{s-1}) - \bar{\mathbf{m}}_s$, we define the conditional-beta estimator $\widehat{\mathbf{G}}_{\beta,t}(\mathbf{X}_{t-1})$ as the eigenvectors corresponding to the first K eigenvalues of

$$\frac{1}{Th}\sum_{s=1}^{T}[\widehat{\mathbf{m}}_s(\mathbf{X}_{s-1}) - \bar{\mathbf{m}}_s][\widehat{\mathbf{m}}_s(\mathbf{X}_{s-1}) - \bar{\mathbf{m}}_s]' K\left(\frac{s-t}{Th}\right) A_t^{-1}.$$

With the estimated conditional-beta, the conditional-alpha and risk premia can be estimated following a cross-sectional procedure based on the decomposition (2.5) for the long-term expected returns. We present the formal algorithm in the next subsection.

It is important to note that the heuristic $s \approx t$ in (3.1) by no means restricts our method to only being applicable to short panels such as the usual moving-window approach. Instead, the "local" nature is naturally possessed by the use of kernel smoothing, and in fact allows much longer time series and more volatile parameters than the usual moving-window approach would do.

In the presence of latent factors, PCA is often used to combine with cross-sectional regressions for unconditional models (Giglio and Xiu, 2021; Giglio et al., 2021). But ordinary PCA works well only in unconditional factor models, because factor-betas can be regarded as eigenvectors of the data matrix *only if* betas are time-invariant. In the context of conditional factor model, we recall that betas have the following decomposition

$$\boldsymbol{\beta}_{t-1} = \mathbf{G}_{\beta,t}(\mathbf{X}_{t-1}) + \boldsymbol{\gamma}_{\beta,t-1}.$$

In this context, PCA would not work well for two reasons. First, betas cannot be represented as the eigenvectors of the return covariance matrix in time-varying models. Secondly, PCA does not distinguish characteristic effects and its orthogonal effects (arising from $\boldsymbol{\gamma}_t$). To improve over regular PCA, Fan et al. (2016); Kim et al. (2021) proposed to use "projected PCA", which removes the effects of $\boldsymbol{\gamma}_{\beta,t}$ but does not takes into account time-varying characteristics or varying $g_{\beta,t}(\cdot)$ function. Kelly et al. (2019) and Chen et al. (2021) extended their methods to incorporate time-varying characteristics, in which dynamic characteristics are mapped to alphas and betas and the mapping is assumed to be time-invariant, and do not work with high-dimensional expected returns.

3.4 The full estimation algorithm

Following the previous discussions, we propose the following algorithm to estimate the conditional factor model. For simplicity, we denote the in-sample data as:

$$\begin{pmatrix} y_{i,1} \\ \mathbf{x}_{i,0} \end{pmatrix}, \cdots, \quad \begin{pmatrix} y_{i,T} \\ \mathbf{x}_{i,T-1} \end{pmatrix}, \quad i = 1, \cdots, N.$$

Algorithm 3.1. Estimate the model parameters and functions following these steps.

S1. Spot expected returns. Run cross-sectional deep NN regression:

$$\widehat{m}_t(\cdot) = \arg \min_{m \in \mathcal{M}_{J,L}} \sum_{i=1}^{N} (y_{it} - m(\mathbf{x}_{i,t-1}))^2, \quad t = 1, \cdots, T.$$

Let $\widehat{\mathbf{m}}_t(\mathbf{X}_{t-1})$ be the $N \times 1$ vector of $\widehat{m}_t(\mathbf{x}_{i,t-1})$.

S2. Long-term expected returns. Run time-domain smoothing:

$$\bar{\mathbf{m}}_t = \frac{1}{Th} \sum_{s=1}^{T} \widehat{\mathbf{m}}_t(\mathbf{X}_{t-1}) K\left(\frac{s-t}{Th}\right) A_t^{-1}, \quad A_t = \frac{1}{Th} \sum_{s=1}^{T} K\left(\frac{s-t}{Th}\right).$$

S3. Beta and Factors. Define $\frac{1}{\sqrt{N}} \widehat{\mathbf{G}}_{\beta,t-1}$ as an $N \times K$ matrix whose columns are the eigenvectors of $\frac{1}{Th} \mathbf{Z}_t \mathbf{K}_t \mathbf{Z}_t'$, corresponding to the top K eigenvalues, where

$$\mathbf{Z}_t = (\widehat{\mathbf{m}}_1(\mathbf{X}_0) - \bar{\mathbf{m}}_t, \cdots, \widehat{\mathbf{m}}_T(\mathbf{X}_{T-1}) - \bar{\mathbf{m}}_t),$$ [5]

and $\mathbf{K}_t$ is a $T \times T$ diagonal matrix consisting of $\{K(\frac{s-t}{Th}) : s = 1, \cdots, T\}$ as the diagonal entries. Define the factor estimator at time t as:

$$\widehat{\mathbf{f}}_t = \widehat{\mathbf{G}}_{\beta,t-1}' (\widehat{\mathbf{m}}_t(\mathbf{X}_{t-1}) - \bar{\mathbf{m}}_t).$$

S4. Alpha and Risk Premia. Run cross-sectional regression to estimate the factor risk premium and $g_{\alpha,t}$:

$$\widehat{\boldsymbol{\lambda}}_{t-1} = \frac{1}{N} \widehat{\mathbf{G}}_{\beta,t-1}' \bar{\mathbf{m}}_t, \quad \widehat{\mathbf{G}}_{\alpha,t-1} := \bar{\mathbf{m}}_t - \widehat{\mathbf{G}}_{\beta,t-1} \widehat{\boldsymbol{\lambda}}_{t-1}. \tag{3.2}$$

[5] We assume the number of factors is consistently estimable. In our empirical analysis we conduct robustness study for various choices of the number factors.

Steps 1 through 3 have been well motivated from our previous discussions. Step 4 estimates the alphas and the factor risk premium. While this step is similar to that in the usual Fama and MacBeth (1973) procedure, here we apply the cross-sectional regression on the average return after DNN projections. In estimating the risk premia in (3.2), we impose $\widehat{\mathbf{G}}'_{\beta,t-1}\widehat{\mathbf{G}}_{\alpha,t-1} = 0$.

Let $\widehat{g}_{\alpha,t-1,i}$ denote the ith element of $\widehat{\mathbf{G}}_{\alpha,t-1}$, which is the estimated in-sample alpha, $g_{\alpha,t}(\mathbf{x}_{i,t-1})$, driven by characteristics, $\widehat{\mathbf{g}}'_{\beta,t-1,i}$ denotes the ith of $\widehat{\mathbf{G}}_{\beta,t-1}$. We define the in-sample risk-related components as in (2.4):

$$\begin{aligned} g_{\text{riskP},t}(\mathbf{x}_{i,t-1}) &:= g_{\beta,t}(\mathbf{x}_{i,t-1})'\boldsymbol{\lambda}_{t-1}, \\ g_{\text{factor},t}(\mathbf{x}_{i,t-1}) &:= g_{\beta,t}(\mathbf{x}_{i,t-1})'(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t), \end{aligned}$$

which can be estimated using[6]

$$\begin{aligned} \widehat{g}_{\text{riskP},t,i} &= \widehat{\mathbf{g}}'_{\beta,t-1,i}\widehat{\boldsymbol{\lambda}}_{t-1} \\ \widehat{g}_{\text{factor},t,i} &= \widehat{\mathbf{g}}'_{\beta,t-1,i}\widehat{\mathbf{f}}_t. \end{aligned}$$

Next, we present the out-of-sample algorithm. The key differences are that the factor innovations are unpredictable and mispricing and risk premia need to be extrapolated. To utilize the state-domain regression for the out-of-sample prediction, we additionally train two neural networks by regressing the estimated in-sample alphas $\widehat{\mathbf{G}}_{\alpha,T-1}$ and risk premia $\widehat{\mathbf{G}}_{\beta,T-1}\widehat{\boldsymbol{\lambda}}_{T-1}$ on $\mathbf{X}_{T-1}$. This gives rise to DNN estimated nonparametric functions: the mispricing function and risk-functions:

$$g_{\alpha,T}(\mathbf{x}), \quad g_{\text{riskP},T}(\mathbf{x}).$$

Note these functions are not required for in-sample estimation of alpha and risk but are required for the out-of-sample decomposition and prediction. The out-of-sample prediction can be constructed by plugging in $\mathbf{X}_T$ to these estimated functions.

Algorithm 3.2. Predict out-of-sample alpha and risk following these steps.

S5. Estimate the in-sample alpha and risk premium $\widehat{\mathbf{G}}_{\alpha,T-1}$ and $\widehat{\mathbf{G}}_{\beta,T-1}\widehat{\boldsymbol{\lambda}}_{T-1}$ as in

[6] While it looks as if we dropped $\mathbb{E}\mathbf{f}_t$ in the estimation below, we actually did not. It is only a matter of notation. The estimators for the latent factors are actually estimating the demeaned ones.

Algorithm 3.1, and write elements of the $N \times 1$ vectors as:

$$\widehat{\mathbf{G}}_{\alpha,T-1} = (\widehat{g}_{\alpha,T-1,1}, \cdots, \widehat{g}_{\alpha,T-1,N})', \quad \widehat{\mathbf{G}}_{\beta,T-1}\widehat{\boldsymbol{\lambda}}_{T-1} = (\widehat{g}_{\text{riskP},T,1}, \cdots, \widehat{g}_{\text{riskP},T,N})'.$$

S6. Run cross-sectional deep NN regression:

$$\widehat{g}_{\text{riskP},T}(\cdot) \;=\; \arg\min_{r \in \mathcal{M}_{J,L}} \sum_{i=1}^{N} (\widehat{g}_{\text{riskP},T,i} - r(\mathbf{x}_{i,T-1}))^2.$$

Run constrained cross-sectional deep NN regression: for some tuning parameter $\nu \to 0$,

$$\begin{aligned} \widehat{g}_{\alpha,T}(\cdot) \;&=\; \arg\min_{g \in \mathcal{M}_{J,L}} \sum_{i=1}^{N} (\widehat{g}_{\alpha,T-1,i} - g(\mathbf{x}_{i,T-1}))^2 \\ &\quad \text{subject to } \left\| \frac{1}{N} \sum_{i=1}^{N} g(\mathbf{x}_{i,T}) \widehat{g}_{\beta,T,i} \right\| \leq \nu. \end{aligned} \tag{3.3}$$

S7. Using the new characteristic $\mathbf{x}_{i,T}$, predict the out-of-sample alphas and compensations for risk as:

$$\begin{aligned} \widehat{\mathbf{G}}_{\alpha,T+1} &= (\widehat{g}_{\alpha,T}(\mathbf{x}_{1,T}), \cdots, \widehat{g}_{\alpha,T}(\mathbf{x}_{N,T}))', \\ \widehat{\mathbf{g}}_{\text{riskP},T+1} &= (\widehat{g}_{\text{riskP},T}(\mathbf{x}_{1,T}), \cdots, \widehat{g}_{\text{riskP},T}(\mathbf{x}_{N,T}))'. \end{aligned}$$

The alpha-functions need be estimated subject to constraints, which restrict the estimation of

$$\left\| \frac{1}{N} \sum_{i=1}^{N} \widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) \widehat{g}_{\beta,T-1,i} \right\|.$$

This restriction ensures that the predicted alphas should be approximately orthogonal to betas, though we are using the in-sample estimated beta $\widehat{g}_{\beta,T-1,i}$ here in the constraint which facilitates computations.

3.5 Double descent of the risk curve for overparametrized learning models

The success of the deep learning revolution builds on a surprising empirical discovery. The best performing deep neural networks are trained with no explicit regularization to control their statistical complexity, and they display excellent prediction performance in the highly-overparametrized regime, that is, the number of parameters is much higher than the number of training samples. In fact, the prediction risk for deep neural networks and other general machine learning methods often appear to present a "double descent" shape as the degree of model complexity increases, where the first descent appears in the classical under-fitting regime, casting the traditional statistical wisdom on the bias-variance tradeoff. Yet as the number of parameters continues to grow, risk starts decreasing again, so a second descent of the prediction risk occurs in the extremely overparametrized regime. Such double-descent phenomena of DNN predictions are illustrated in a recent empirical work by (Belkin et al., 2019). In fact, this scenario is far from being specific to neural networks, and has been observed in quite a few statistical machine learning models including nonparametric regression (Belkin et al., 2019), kernel learning (Belkin et al., 2018), and factor modeling and matrix factorization (Arora et al., 2019), ridge regression in random features model (Mei and Montanari, 2019), and even for linear models (Hastie et al., 2019; Belkin et al., 2020). In the asset pricing literature, Kelly et al. (2021) documented that Sharpe ratios of machine learning portfolios may increase for overparametrized models.

Below we demonstrate the "double descent" scenario using the diffusion index model of Stock and Watson (2002). Our demonstration is perhaps of independent interest itself, because the diffusion index model is one of the most popular economic models for big-data forecasts, and to our best knowledge, the double descent scenario has not been observed in this context. Consider forecasting an index return Y_{t+1} that is generated from a dynamic factor model:

$$Y_{t+1} = \mathbf{b}'\mathbf{f}_t + \varepsilon_t, \quad \mathbf{f}_t = \rho\mathbf{f}_{t-1} + \mathbf{v}_t. \tag{3.4}$$

As for the working model, we assume that the true DGP (3.4) is unknown, and forecast

Y_{t+1} using lagged returns of a large number of asset returns $\mathbf{X}_t = (X_{1,t}, \cdots, X_{p,t})'$:

$$Y_{t+1} \quad = \quad \mathbf{X}_t'\boldsymbol{\theta} + e_t, \tag{3.5}$$

with $X_{j,t} = \boldsymbol{\beta}_j'\mathbf{f}_t + u_{j,t}$ follows a factor model with latent factors $\mathbf{f}_t$ and unknown factor loading vector $\boldsymbol{\beta}_j$. It is well known that when $p > T$, OLS is not defined. So to estimate model (3.5) when p is large, we employ the "minimum-norm interpolation" least squares:

$$\begin{aligned} \widehat{\boldsymbol{\theta}}_p &= \arg\min\{\|\boldsymbol{\theta}\| : \boldsymbol{\theta} \text{ minimizes } \sum_t (Y_{t+1} - \mathbf{X}_t'\boldsymbol{\theta})^2\} \\ &= \left(\sum_{t=1}^{T} \mathbf{X}_t\mathbf{X}_t'\right)^+ \sum_{t=1}^{T} \mathbf{X}_t Y_{t+1} \end{aligned} \tag{3.6}$$

where $\mathbf{A}^+$ denotes the generalized inverse of matrix $\mathbf{A}$. Note that the solution (3.6) always exists regardless of $p := \dim(\mathbf{X}_t)$ and reduces to OLS when $p \leq T$. But unlike ridge regressions, this estimator overfits the model when p is large. In fact when $p > T$, this estimator interpolates the in-sample data as $Y_{t+1} = \mathbf{X}_t'\widehat{\boldsymbol{\theta}}_p$ for $t = 1, \cdots, T$ in this case. For the out-of-sample data $\{\mathbf{X}_t : t = T+1, \cdots, T+s\}$, we evaluate the out-of-sample prediction risk

$$R(p) := \frac{1}{s} \sum_{t=T+1}^{T+s} (Y_{t+1} - \mathbf{X}_t\widehat{\boldsymbol{\theta}}_p)^2.$$

Figure 1 plots $R(p)$ as p increases, averaged over one hundred simulations, where all parameters are calibrated to data of monthly U.S. equity returns. The plot clearly shows the double-descent pattern of the prediction risk: The prediction risk first decreases because the model is less biased, but then increases because of a variance explosion, and reaches a peak at the the interpolation threshold, where the model completely interpolates the in-sample data, corresponding to zero in-sample error but large prediction risk. As more assets are included as predictors, the prediction risk decreases again, and appears to be "at infinite complexity": the more overparametrized is the model, the smaller is the prediction risk.

There have been two interpretations in the literature: the first being that machine learning algorithms often rely on gradient descent algorithms, which induces implicit regularizations that select the simplest overparametrized model in a suitable sense

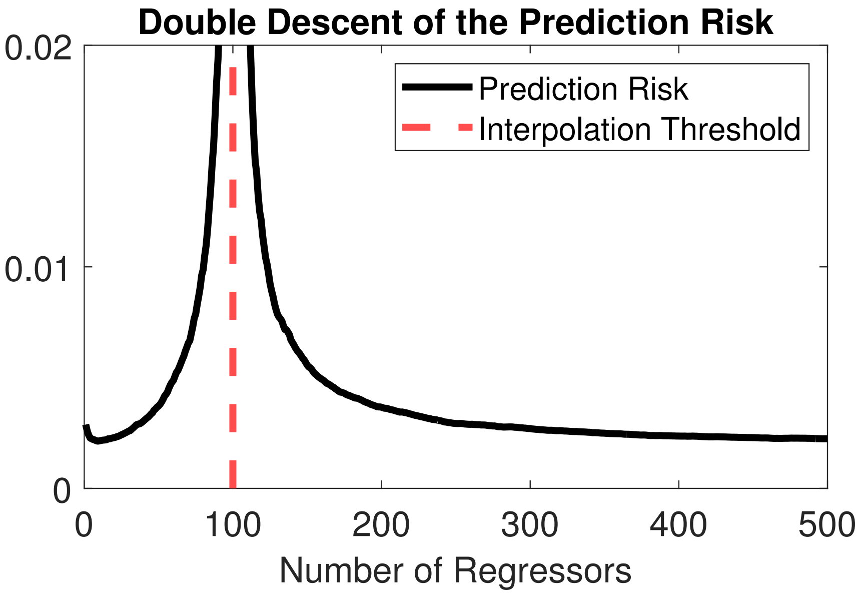


Figure 1: The prediction risk $R(p)$ for predicting the asset Y_{t+1} using lagged returns $(X_{1,t}, \cdots, X_{p,t})$ with the minimum-norm interpolation estimator, averaged over one hundred simulations. The number of in-sample fitted data is $T = 100$ and the number of out-of-sample predicted data is $s = 25$. The interpolation threshold refers to the occurance $p = T$, where the estimator completely interpolates the in-sample data. The true model for Y_{t+1} is given by (3.4), and $X_{j,t} = \boldsymbol{\beta}_j'\mathbf{f}_t + u_t$, with $(\varepsilon_t, \mathbf{v}_t, e_t, u_t)$ being generated independent from the multivariate normal distribution. All parameters are calibrated from the monthly data of 2140 asset returns and Fama-French-three factors from January 2015 to December 2017. The prediction risk first descents and achieves a local minimum at $p = 10$, and increases as p approaches to the in-sample size. As the number of predictors continues increasing, it descents again.

(see, e.g., Du et al. (2018)). As for the linear model (3.4) and (3.6), we note that the minimum-norm interpolation estimator can be seen as the limit of ridge regressions with vanishing tuning parameters, which is in fact the convergence point of gradient descent for least squares loss. The second interpretation is similar to the diversification effect in finance. As the number of regressors grows, more predictors/neurons generally result in decreasing magnitudes for the components of $\boldsymbol{\theta}_p$, by distributing signals over more parameters, so the variance of the estimator decreases, which also leads to descending prediction risks. On the other hand, due to the interpolation, the biases are generally small as p gets bigger. However, even though the double descent has been documented widely, understanding it from a theoretical perspective is still at the forefront of the statistical machine learning literature.

4 Asymptotic Theory

While our methods seem intuitively appealing and appear heuristic, they are actually provable. In this section we present the formal theory to justify the proposed procedure. We show that the theoretical results depend on three key quantities:

$$\begin{aligned}
\varphi_T &= \max_t \sup_{\mathbf{x}} \left[\inf_{g\in\mathcal{M}_{J,L}} |g_{\beta,t}(\mathbf{x}) - g(\mathbf{x})| + \inf_{g\in\mathcal{M}_{J,L}} |g_{\alpha,t}(\mathbf{x}) - g(\mathbf{x})| \right]. \\
\delta_T &= \sqrt{\frac{p(\mathcal{M}_{J,L})\log(NT)}{N}}, \quad (p(\mathcal{M}_{J,L}) \text{ to be defined soon}) \\
\eta_T &= \frac{1}{\sqrt{Th}} + h^2.
\end{aligned}$$

The first term φ_T denotes the approximation error using deep neural networks to nonparametric functions of interests. The approximation error normally *does not* suffer from the curse of dimensionality when $g_{\beta,t}$ and $g_{\alpha,t}$ belong to a broad class of functions with low intrinsic dimensions. For instance, Schmidt-Hieber (2020) showed that if the true regression function, say g_0 is a composition of several functions:

$$g_0 = f_q \circ f_{q-1} \circ \cdots \circ f_1$$

where each component f_j is a multi-dimensional and multivariate function, then a multilayer feedforward network with ReLu activation functions at each layer would lead to the approximation error:

$$\varphi_T \leq C \max_{i\leq q} N^{-\kappa_i/(2\kappa_i+s_i)},$$

for a flexible collection of width and length of layers. Namely the approximation errors are dominated by the hardest functions in the composition, where s_i is the maximum number of input variables that f_i may depend on, and κ_i measures the smoothness of f_i. This approximation error holds for a robust choice of the growth of the width J and depth L of the network. Excitingly, t_i is the "intrinsic dimension" which can be much smaller than $\dim(\mathbf{x}_{i,t-1})$. For instance if f_i depends on the input f_{i-1} through its linear combinations (such as the single index model), then $\max_{i\leq q} t_i = 1$, so the curse of dimensionality is adaptively avoided. Apparently, components f_i in the composition are not separately identified, but we are only interested in g_0 so this causes no problems. Schmidt-Hieber (2020) showed that among all possible representations, the neural network picks one that leads to the fastest possible approximation rate. [7]

The second term δ_T represents the complexity of the deep neural network space growing with the number of layers and neurons. The complexity is measured by the *pseudo dimension* $p(\mathcal{M}_{J,L})$ of the network, defined as the Vapnik-Chervonenkis dimension of the subgraph class $\{f(x,y) := \operatorname{sgn}(h(x)-y) : h \in \mathcal{M}_{J,L}\}$. Bartlett et al. (2019) showed that for a ReLu network with depth L and the maximum width J across layers, for some constant $C > 0$,

$$p(\mathcal{M}_{J,L}) \leq CJ^2L^2\log(JL).$$

The third term η_T is the usual nonparametric rate for time-domain smoothing. Thanks to the use of boundary-adjusted kernels, this result holds for not only the "interior periods" $(Th, T-Th)$, which is the focus of Ang and Kristensen (2012), but also the "boundary periods" (either $[1, Th]$ or $[T-Th, T]$). We extend the in-sample studies of Ang and Kristensen (2012) to the out-of-sample context, where incorporating the boundary case $t = T$ is essential.

The formal assumptions are listed as follows.

Assumption 4.1 (Cross-sectional and serial dependences)**.** *(i) For each fixed t, the sequence $\{\mathbf{x}_{i,t}\}$ is cross-sectionally i.i.d.*

(ii) Let $e_{it} = y_{it} - \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)$. Then for each fixed t, $\{e_{it}\}$ is cross-sectionally independent and sub-Gaussian, conditioning on $\mathbf{f}_t$. More specifically, there are $c_1, c_2 > 0$, $\forall x > 0$ such that almost surely for $\mathbf{f}_t$, $\max_{it}\mathbb{P}(|e_{it}| > x|\mathbf{f}_t) \leq c_1\exp(-c_2x^2)$.

(iii) Let $\mathcal{F}_t$ be the filtration generated by $\{\mathbf{x}_{i,t} :$ for all $i\}$ up to time t. Then for all i, t, $\mathbb{E}(u_{i,t}|\mathcal{F}_{t-1}, \mathbf{f}_t) = 0$, $\mathbb{E}(\gamma_{\alpha,i,t-1}|\mathcal{F}_{t-1}, \mathbf{f}_t) = 0$, and $\mathbb{E}(\boldsymbol{\gamma}_{\beta,i,t-1}|\mathcal{F}_{t-1}, \mathbf{f}_t) = 0$.

[7] Suppose unknown to us, the function $m_t^0(\cdot)$ is additive, the neural network will find an estimator that has one-dimensional rate. As another example, suppose, unknown to us, that $m(x_1, \cdots, x_5) = f_1(x_1, x_2) + f_3(x_3) + f_4(f_5(x_1, x_3), f_6(x_2, x_4), x_5)$ (dropping the subscript t), the neural network will automatically find an estimator $\hat{m}(\cdot)$ with three dimensional nonparametric rate or more generally the rate that is denominated by the hardest component in the composition. The main reason for this automatic adaptation is that the composition of the neural networks is still a neural network. Take the second case as an example, if the structure were known, we can construct an optimal neural network $\mathcal{N}_0$ to approximate the function $m(\cdot)$ by constructing optimal DNN for each component and taking a composition of these networks. When it is unknown, we can take the maximum depth and maximum width of the network $\mathcal{N}_0$ and construct a fully connected DNN $\mathcal{N}_1$ that has at least order of the approximation error as $\mathcal{N}_0$, yet the complexity of $\mathcal{N}_1$ remains the same order as $\mathcal{N}_0$. In other words, we achieve the same order of approximation without inflating significantly the variance.

(iv) The factor process $\{\mathbf{f}_t : t = 1, \cdots, T\}$ is stationary and $\mathbb{E}(\mathbf{f}_t|\mathcal{F}_{t-1}) = \mathbb{E}\mathbf{f}_t$. In addition, it is weakly dependent in the sense that for any $\mathbf{v}_s \in \{\mathbf{f}_s, \mathrm{vec}(\mathbf{f}_s\mathbf{f}_s')\}$, at each fixed t,

$$\max_k \mathrm{var}\left(\frac{1}{Th}\sum_s \frac{(s-t)}{T}(v_{s,k} - \mathbb{E}v_{s,k})K\left(\frac{t-s}{Th}\right)\right) = O\left(\frac{h}{T}\right). \tag{4.1}$$

Condition (4.1) is straightforward to verify if factors are serially independent with bounded second moments. Meanwhile, we present this high-level assumption to allow some possible weak dependence over time. Above all, the factor process should have very low persistence and hard to predict.

Assumption 4.2 (Smoothness over time)**.** *(i) For each fixed i, there exist functions $m_i(\cdot)$ and $\mathbf{g}_i(\cdot)$ so that almost surely for $\mathbf{x}_{i,t-1}$, we have*

$$\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}) = m_i\left(\frac{t}{T}\right), \quad g_{\beta,t}(\mathbf{x}_{i,t-1}) = \mathbf{g}_i\left(\frac{t}{T}\right), \forall t = 1, \cdots, T,$$

where the functions are continuously twice-differentiable:

$$\sup_{v,i}\left[\left|\frac{dm_i(v)}{dv}\right| + \left|\frac{d^2m_i(v)}{dv^2}\right| + \|\nabla\mathbf{g}_i(v)\| + \|\nabla^2\mathbf{g}_i(v)\|\right] < C.$$

(ii) For out-of-sample predictions:

$$\sup_{\mathbf{x}}|g_{\alpha,T}(\mathbf{x}) - g_{\alpha,T+1}(\mathbf{x})| = O_P(T^{-1/2}), \quad \sup_{\mathbf{x}}|g_{riskP,T}(\mathbf{x}) - g_{riskP,T+1}(\mathbf{x})| = O_P(T^{-1/2}),$$

Assumption 4.2 extends the condition A.1 of Ang and Kristensen (2012) to the characteristic-based time-varying models, which assumes that the long-term expected return and characteristic-betas should be varying smoothly over time. But different from them, we do *not* require the entire betas or alphas to be smooth functions since our approach is not based on time-series OLS. Rather, our approach, when integrated with the neural network projection, allows us to impose such conditions only on the characteristic-driven components, leaving the remaining components ($\boldsymbol{\gamma}_t$) to be possibly varying nonsmoothly over time.

In the assumption below, we recall that $m_t^0(\mathbf{x}) := \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1} = \mathbf{x}, \mathbf{f}_t)$.

Assumption 4.3 (For the Neural Network learning)**.** *(i) For each fixed t, almost* *surely for all* $\mathbf{f}_t$*, functions* $m_t^0, g_{\alpha,t}$ *and* $g_{\beta,t}$ *belong to the Hölder ball: for some* $q \in \mathbb{R}$*,* $\gamma \in (0,1]$ *and* $L > 0$*,* [8]

$$\mathcal{H}(q,\gamma,L) = \{f : [a,b]^d \to \mathbb{R}, \|f\|_{q,\gamma} \leq L\}, \quad \|f\|_{q,\gamma} = \max_{s\leq q} \sup_{\mathbf{a},\mathbf{b}} \frac{|f^{(s)}(\mathbf{a}) - f^{(s)}(\mathbf{b})|}{\|\mathbf{a}-\mathbf{b}\|^{\gamma}}.$$

(ii) The dimension of the neural network space satisfies: $J^2L^2\log(JL)\log^{3/2}(NT) = o(N)$*, where* J, L *respectively denote the width and depth of the network.*

Assumption 4.3 is the technical condition that ensures that the spot expected returns and alpha-, beta- functions can be learned sufficiently well by employing cross-sectional DNN. Indeed, it has been proved in the machine learning literature that functions in the Hölder space can be approximately well using fully connected neural networks. Condition (ii) on the other hand, regulates the complexity of the type of neural networks we shall use. Namely, the network cannot be too deep or too wide, which is a technical condition for mathematical proofs.

Assumption 4.4. *(i) Identification: For any* $\epsilon > 0$ *and for some* $c > 0$*,*

$$\min_t \inf_{\|m-m_t^0\|_{q,\gamma}>\epsilon} \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^2 > c$$

In addition, we impose

$$\max_t \left\| \frac{1}{N}\sum_{i=1}^{N} g_{\beta,t}(\mathbf{x}_{i,t-1})g_{\alpha,t}(\mathbf{x}_{i,t-1}) \right\| = O_P(N^{-1/2}).$$

(ii) Moments: For some $C > 0$*,*

$$\sup_{\mathbf{x}} |g_{\alpha,t}(\mathbf{x})| + \sup_{\mathbf{x}} \|g_{\beta,t}(\mathbf{x})\| + \|\boldsymbol{\lambda}_t\| + \mathbb{E}\|\mathbf{f}_t\|^2 < C.$$

(iii) The eigenvalues of $\frac{1}{N}\sum_{i=1}^{N} g_{\beta,t}(\mathbf{x}_{i,t-1})g_{\beta,t}(\mathbf{x}_{i,t-1})'$ *and* $\mathbb{E}\mathbf{f}_t\mathbf{f}_t'$ *are bounded away from zero and infinity uniformly over* t*.*

Recall that $\widehat{m}_t(\cdot)$ is the neural network estimated function by cross-sectionally

[8] With additional assumptions, the functional space can be extent to the *hierarchical composition space*, which consists of compositions of several functions, such as $f = f_q \circ f_{q-1} \circ \cdots \circ f_1$ where each of f_k belongs to a Hölder ball. See Kohler and Langer (2021) for additional details.

regressing y_{it} onto $\mathbf{x}_{i,t-1}$ at time t, which estimates the spot expected return; $\bar{m}_{i,t}$ is the weighted average of $\widehat{m}_s(\cdot)$ locally around t.

Theorem 4.1 (Expected Returns)**.** *Suppose Assumptions 4.1-4.4 hold. Then*

(i) For spot expected returns: at each fixed t,

$$\mathbb{E}[\widehat{m}_t(\mathbf{x}_{i,t-1}) - \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)]^2 = O_P(\delta_T^2 + \varphi_T^2).$$

(ii) For long-term expected returns: at each fixed t,

$$\mathbb{E}[\bar{m}_{i,t} - \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})]^2 = O_P(\delta_T^2 + \varphi_T^2 + \eta_T^2).$$

Theorem 4.1 respectively presents the quality for learning the spot and long-term expected returns using DNN projections. We see that the spot expected return $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)$ is learned period-by-period, so its learning quality depends on both the complexity (δ_T) and the approximation error (φ_T) of the DNN space. In addition, estimating the long-term expected return $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})$ requires an additional step of kernel averaging over time, so its quality further involves the smoothing estimation error η_T.

As for the in-sample alpha and risk, we have

Theorem 4.2 (In-Sample Alpha and Risk)**.** *Suppose Assumptions 4.1-4.4 hold. Then at each time t of interest,*

$$\begin{aligned}
\frac{1}{N}\sum_{i=1}^{N}[\widehat{g}_{\alpha,t-1,i} - g_{\alpha,t}(\mathbf{x}_{i,t-1})]^2 &= O_P(\delta_T^2 + \varphi_T^2 + \eta_T^2), \\
\frac{1}{N}\sum_{i=1}^{N}[\widehat{g}_{riskP,t,i} - g_{riskP,t}(\mathbf{x}_{i,t-1})]^2 &= O_P(\delta_T^2 + \varphi_T^2 + \eta_T^2) \\
\frac{1}{N}\sum_{i=1}^{N}[\widehat{g}_{factor,t,i} - g_{factor,t}(\mathbf{x}_{i,t-1})]^2 &= O_P(\delta_T^2 + \varphi_T^2 + \eta_T^2).
\end{aligned}$$

The next theorem establishes the properties of the predictions for alpha and compensation for risk. While the literature on deep neural networks mostly concentrates on in-sample analysis, to our best knowledge, the out-of-sample predictive rate has not been established in any context. We develop a new technical argument to achieve a sharp out-of-sample rate for DNN predictions, and show that it can be as fast as the in-sample convergence rate.

Theorem 4.3 (Out-of-Sample Prediction)**.** *Suppose the tuning parameter ν in the constraint (3.3) satisfies: for some sufficiently large $C > 0$,*

$$\nu \geq C\left[\varphi_N + \eta_T + \delta_T + \left(\frac{1}{N}\sum_{i=1}^{N}[g_{\alpha,T+1}(\mathbf{x}_{i,T}) - g_{\alpha,T}(\mathbf{x}_{i,T-1})]^2\right)^{1/2}\right].$$

In addition, Assumptions 4.1- 4.4 and Assumption **??** *hold. Then*

$$\begin{aligned}&\max_{i\leq N}|\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) - g_{\alpha,T+1}(\mathbf{x}_{i,T})| = O_P\left(\delta_T + \varphi_T + \eta_T\right)\\ &\max_{i\leq N}|\widehat{g}_{riskP,T}(\mathbf{x}_{i,T}) - g_{riskP,T+1}(\mathbf{x}_{i,T})| = O_P\left(\delta_T + \varphi_T + \eta_T\right).\end{aligned}$$

As a consequence of Theorem 4.3, the out-of-sample decomposition follows. It shows that $y_{i,T+1}$ relies on the two DNN-forecasters $\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T})$ and $\widehat{g}_{\text{riskP},T}(\mathbf{x}_{i,T})$, plus noises that are not predictable. We present it in the theorem below. Recall that $\mathcal{F}_T$ is the sigma-algebra generated by characteristics $\{\mathbf{x}_{i,t} : t = 1, ..., T, \text{ for all } i\}$ up to time T.

Theorem 4.4 (Prediction decomposition)**.** *Suppose assumptions of Theorem 4.3 hold. Then there exists $\xi_{i,T+1}$ so that uniformly for $i \leq N$,*

$$y_{i,T+1} = \widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) + \widehat{g}_{riskP,T}(\mathbf{x}_{i,T}) + \xi_{i,T+1} + O_P\left(\delta_T + \varphi_T + \eta_T\right),$$

where $\mathbb{E}(\xi_{i,T+1}|\mathcal{F}_T) = 0$.

5 Empirical Analysis

5.1 Data

Our main data set is the same as in Freyberger et al. (2020) and has been updated through 2018, spanning 648 months with about 4261 firms on average per month. Asset returns are obtained from the Center for research in Security Prices (CRSP) monthly file and accounting data are from Compustat. As in most empirical asset pricing studies, we limit the analysis to common equity which is trading on NYSE, Nasdaq or Amex. We also limit the analysis to U.S. firms. As in Freyberger et al. (2020), we use accounting data from the fiscal year ending in calendar year $t - 1$ for estimation starting from the end of June of year t until the end of May of year $t + 1$, predicting returns from the beginning of July of year t until the end June of year $t+1$. We require that firms have at least two years of data in Compustat to include in the paper to mitigate survivorship biases, which may arise from backfilling. Our overall sample ranges from 1965 through 2018. Table **??** in the Appendix provides an overview of the characteristics.

5.2 Return Decomposition

In the following, we estimate the in- and out-of-sample return decompositions (2.4) and (2.6) respectively. Throughout, we use a 60 months window for estimation, which we slide forward by one month, after each estimation. In the implementation, we use a one, two and three layer network with four nodes on each hidden layer. Throughout, we employ a learning rate of 0.001 and use 2000 epochs. We also use a constant bandwith $h = 0.75$, which approximately minimizes the in-sample mean squared error for estimating the spot expected returns.

5.2.1 In-Sample Decomposition

We decompose both expected returns and realized returns into a mispricing component, $g_{\alpha,t}$, and a risk-based component which is driven by the risk premium component, $g_{\beta,t}(\mathbf{x})'\boldsymbol{\lambda}_{t-1}$, and the exposure to the factor shock, $g_{\beta,t}(\mathbf{x})'(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t)$. Since returns are noisy, we first establish a benchmark of how much of realized return can be explained, relying on the the following measure of R^2:

$$R^2 = 1 - \frac{\sum_{i,t}(y_{it} - \mathtt{prediction}_{it})^2}{\sum_{i,t} y_{it}^2}, \tag{5.1}$$

where we set $\mathtt{prediction}_{it}$ to equal the different parts of the return decomposition, (2.4), to assess their explanatory strength from a statistical perspective. For the in-sample decomposition, prediction is the same as fitting.

Table I shows the results for varying the number of factors and different layer networks. We see that, for the in-sample decomposition, the bulk of the variation is due to risk related components. For instance, Panel A shows the results for all firms, and when we use ten factors ($K = 10$) and a single layer, the in-sample R^2 for $\beta'(F+\lambda)$ is 23.67 percent, which explains about 90% of the explained variation of $\widehat{y}$. Within this 90%, the bulk of the explanatory power is driven by the factor realization

($\approx 95\%$) and roughly 5% by the long-term risk premium. Panel B shows the results for the 80% largest firms by (lagged) market capitalization. We can explain a lot more variation in excess returns for large firms. Panel C of Table I shows the corresponding measures for the 20% smallest firms. While the qualitative finding that the bulk of the variation can be explained by the risk-related components is still true, the explained variation slightly drops. For instance, for ten factors and a single layer model, it is 79% ($= 17.17/21.64$). This is in line with the intuition that returns of large firms are in general better explained by exposure to systematic risk relative to small firms.

In addition, up to 1% of the explained in-sample return can be attributed to mispricing for all firms if we use a single factor, otherwise the R^2_α is typically less than zero, and small firms have larger R^2_α than large firms. This is in line with many findings in empirical asset pricing that mispricing tends to concentrate in smaller firms. Additionally, when we compare $R^2_{\widehat{y}}$ with $R^2_{\beta'(F+\lambda)}$ and R^2_α, the general pattern is that $R^2_{\widehat{y}}$ is closer to the sum of the two for large number of factors and large firms. Meanwhile, Panel C shows that the difference in $R^2_{\widehat{y}}$ and $R^2_{\beta'(F+\lambda)}$ is still relatively large even for 10 factors. A sensible explanation to this is that small firms tend to load weakly on factors so the spot expected returns have relatively large idiosyncratic components and errors in variable issues in the estimated factors.[9]

Overall, we can see from Table I that across all specifications and cuts of the data, factor sensitivity times factor realization takes on the largest fraction of the explanatory power. Since factors are only available contemporaneously with the return realization and are themselves excess returns they are almost completely unpredictable. This carries an important lesson for out-of-sample prediction, the components that we can forecast are the slower moving pieces, i.e. the risk-premium component ($\mathbf{g}'_{\beta,t-1}\boldsymbol{\lambda}_{t-1}$) and mispricing ($\mathbf{g}_{\alpha,t-1}$). We will revisit this issue in Section 5.2.3 when we study out-of-sample predictability.

5.2.2 Pricing Error

Leitch and Tanner (1991) point out that studying purely statistical measures of fit may sometimes be at odds with economic measures of success. In particular, some of the R^2_αs in Table I are negative which may at first sight suggest that mispricing is

[9] One can show the in-sample $\widehat{y}_{it} = \widehat{g}_{\alpha,t}(\mathbf{x}_{i,t-1}) + \widehat{g}_{\text{riskP},t}(\mathbf{x}_{i,t-1}) + \widehat{g}_{\text{factor},t}(\mathbf{x}_{i,t-1}) + \widetilde{u}_{it} + o_P(1)$, where $\widetilde{u}_{it}$ is the estimation error from the projected idiosyncratic term and the $o_P(1)$ term contains other estimation errors. The last two components are relatively large for small firms.

Table I: In-Sample Decomposition - Realized Returns (Full Sample)

This table shows empirical estimates for the in-sample decomposition of realized returns (equation (2.5)). $R^2_{\hat{y}}$ measures the quality of the in-sample fit from the period-by-period DNN regresssions of excess returns onto characteristics. $R^2_{\beta' F}$ measures how much of the variation in excess returns can be explained by exposure to common factors. $R^2_{\beta' \lambda}$ measures how much of excess returns is explained by the factor risk premia, $R^2_{\beta'(F+\lambda)}$ measures how much can be explained by all risk-based components. R^2_{α} measures how much in-sample variation of excess returns can be explained by mispricing. Panel A shows the results for all firms in our CRSP/Compustat sample, Panel B focuses on the 80% largest firms and Panel C shows the results for the 20% smallest firms. All R^2 measure are in percentage. The sample period is 1970 - 2018.

	1 Layer					2 Layers					3 Layers				
K	$R^2_{\hat{y}}$	$R^2_{\beta' F}$	$R^2_{\beta' \lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}	$R^2_{\hat{y}}$	$R^2_{\beta' F}$	$R^2_{\beta' \lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}	$R^2_{\hat{y}}$	$R^2_{\beta' F}$	$R^2_{\beta' \lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}
Panel A: All firms															
1	26.34	16.31	1.03	17.33	0.03	27.57	16.36	1.03	17.39	0.02	30.36	17.22	1.00	18.20	0.17
6	26.34	21.46	1.21	22.61	-0.14	27.57	21.80	1.20	22.95	-0.16	30.36	23.85	1.25	25.04	-0.07
10	26.34	22.50	1.23	23.67	-0.16	27.57	23.15	1.24	24.34	-0.19	30.36	25.52	1.26	26.73	-0.09
Panel B: Large firms															
1	35.9	23.12	0.86	24.05	-0.06	36.21	23.31	0.87	24.25	-0.10	36.45	22.37	0.83	23.26	0.20
6	35.9	33.22	1.13	34.58	-0.21	36.21	33.17	1.11	34.56	-0.24	36.45	32.73	1.17	34.12	-0.06
10	35.9	34.38	1.18	35.75	-0.27	36.21	34.27	1.20	35.65	-0.33	36.45	33.81	1.18	35.21	-0.09
Panel C: Small firms															
1	21.64	10.01	1.35	11.30	0.23	23.76	10.04	1.34	11.32	0.19	29.75	12.99	1.30	14.20	0.34
6	21.64	14.68	1.49	15.94	0.01	23.76	15.19	1.48	16.42	-0.04	29.75	20.26	1.55	21.53	0.08
10	21.64	15.91	1.51	17.17	-0.02	23.76	17.14	1.50	18.38	-0.07	29.75	22.63	1.57	23.89	0.05

Table II: In-Sample Sharpe Ratios of Mispricing Portfolio

This table presents estimates for the annualized Sharpe ratio of the mispricing portfolio for the full sample (Panel a), the early sample (Panel B) and the late sample (Panel C). K denotes the number of estimated factors. Small firms are the 20% smallest firms and large firms are the 80% largest firms in each month. The returns of the mispricing portfolio are computed as $r_{\alpha,t} = \mathbf{G}'_{\alpha,t-1}\mathbf{y}_t$. The annualized Sharpe ratio is then computed as $SR_{ann} = \sqrt{12}\bar{r}_\alpha/s$ where $\bar{r}_\alpha, s$ respectively denote the in-sample return average and standard deviation.

	1 Layer			2 Layers			3 Layers		
K	All	Large	Small	All	Large	Small	All	Large	Small
Panel A: 1970 - 2018									
1	1.45	0.50	1.51	1.52	0.51	1.41	1.63	0.63	1.33
6	1.11	0.43	1.22	1.19	0.42	1.00	1.57	0.54	1.32
10	0.96	0.30	1.11	0.91	0.29	0.83	1.55	0.52	1.21
Panel B: 1970 - 1999									
1	1.57	0.62	1.32	1.71	0.67	1.33	1.69	0.80	1.15
6	1.07	0.65	1.08	1.29	0.55	0.98	1.86	0.71	1.26
10	0.95	0.45	0.98	0.99	0.37	0.79	1.83	0.65	1.19
Panel C: 2000 - 2018									
1	1.41	0.29	1.86	1.40	0.24	1.58	1.52	0.32	1.71
6	1.20	0.23	1.45	1.07	0.28	1.03	1.21	0.38	1.43
10	0.99	0.21	1.32	0.79	0.22	0.90	1.21	0.45	1.25

completely negligible. To investigate this, we study $\mathbf{G}_{\alpha,t-1}$, more closely. In the first step, we compute the (in-sample) returns to the mispricing portfolio,

$$r_{\alpha,t} := \frac{1}{N_t}\widehat{\mathbf{G}}'_{\alpha,t-1}\mathbf{y}_t.$$

To asses the economic magnitude of the mispricing, we compute the annualized Sharpe ratio for this portfolio in Table II. We see that many annualized Sharpe ratios are greater than one and therefore strongly economically significant in magnitude. The results in Table II also show that mispricing is more strongly concentrated in small rather than larger firms. Note that it can be the case that the Sharpe ratio is higher for the portfolio using all stocks rather than just small stocks. This happens because the standard deviation of the portfolio returns for all stocks can be smaller in some cases than for the portfolio of small stocks due to greater diversification benefits.

At first sight, results from Table II might suggest that mispricing is more or less constant across time. To investigate the time-variation more closely, it is advantageous to study a "denoised" version of the mispricing portfolio return, defined as

$$\widehat{r}_{\alpha,t} := \frac{1}{N_t}\widehat{\mathbf{G}}'_{\alpha,t-1}\widehat{\mathbf{y}}_t.$$

This quantity can be interpreted as an estimate of the squared pricing error, which is often used in the examination of factor models, e.g. Gibbons et al. (1989) or Dybvig and Ross (1985). While $\widehat{r}_{\alpha,t}$ cannot be interpreted as an excess return to a traded portfolio (because $\widehat{\mathbf{y}}_t$ are not the returns of traded assets), it is a good measurement of the returns' magnitude because the idiosyncratic components have been removed due to the projected step: $\widehat{r}_{\alpha,t} = r_{\alpha,t} + o_P(1)$. Figure 2 plots the evolution of $\widehat{r}_{\alpha,t}$ over time. From the plot, we see that the magnitude of the pricing error varies over time and increases with times or large market volatility such as the dot-com episode and the 2008/2009 financial crises. When we compare the three panels of Figure 2, we observe that the magnitude of the pricing error is much smaller for six estimated factors ($K = 6$) relative to just one estimated factor ($K = 1$). However, it appears that the difference between six and ten estimated factors does not affect the shape or magnitude of the pricing error in a material way. In Figure **??** in the Appendix, we present the corresponding analysis for the 80% largest firms. While the shape of the local average is similar across different firm sizes, the magnitude of the pricing error is larger for small firms.

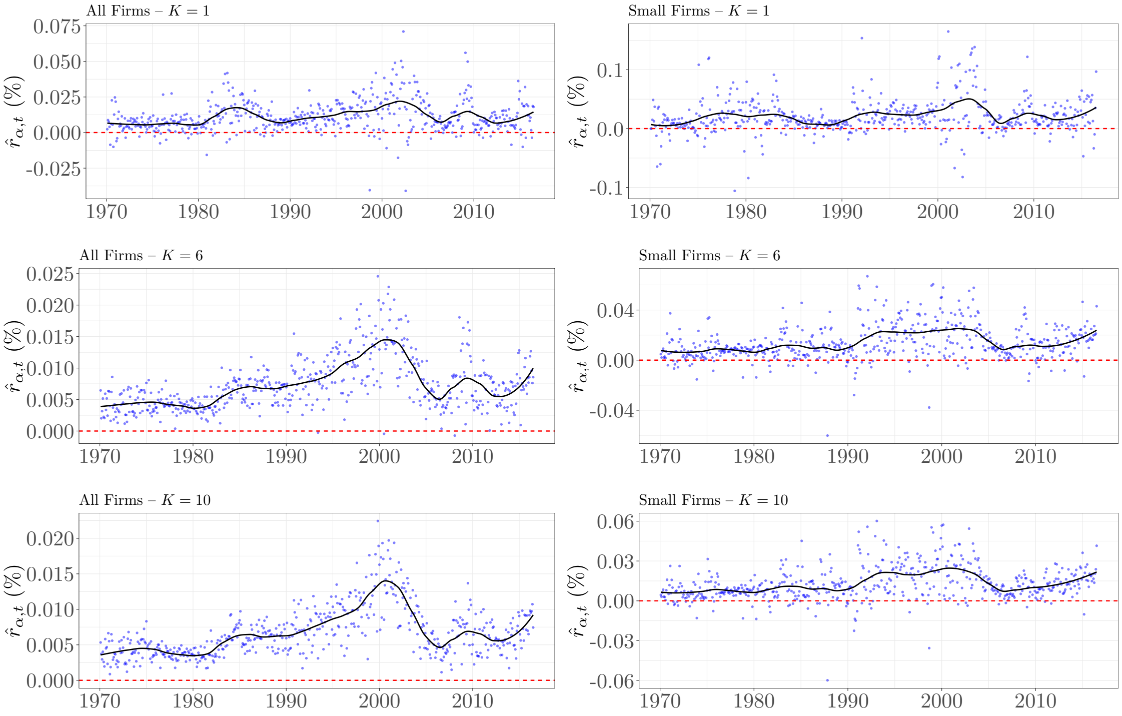
It is also evident from Figure 2 that the temporal evolution of alphas is not simply due to the evolution of characteristics, but that the alpha-function, $g_{\alpha,t}$, is also changing over time. In fact, since we are ranking characteristics at each period t, whose cross-sectional density is uniform on $[0,1]^{\dim(\mathbf{x})}$, as $N_t \to \infty$,

$$\widehat{r}_{\alpha,t} = \frac{1}{N_t}\sum_{i=1}^{N_t}\widehat{g}_{\alpha,t-1,i}\widehat{y}_{i,t} = \frac{1}{N_t}\sum_{i=1}^{N_t} g_{\alpha,t}(\mathbf{x}_{i,t-1})^2 + o_P(1) \to^P \int g_{\alpha,t}(\mathbf{x})^2 p(\mathbf{x})d\mathbf{x},$$

where $p(\mathbf{x})$ denotes the density of multivariate uniform distribution on $[0,1]^{\dim(\mathbf{x})}$. The probability limit of the convergence depends on $g_{\alpha,t}$ rather than the characteristics. This highlights the importance of our time-varying approach, which allows for varying mappings from characteristics to mispricing over time.

Figure 2: Evolution of Pricing Error over Time

This figures shows estimates of the average squared pricing error computed as $\frac{1}{N_t}\widehat{\mathbf{G}}_{\alpha,t-1}(\mathbf{x})'\widehat{\mathbf{y}}_t$ for all firms and $K = 1$, $K = 6$ and $K = 10$ for the full sample (blue dots). We also present a local regression smoothing curve as an estimate of the local average (black line). The red dashed horizontal line is at zero.



Finally, it appears from Figure 2 that the pricing error is correlated with market volatility. To analyze this relationship, we run the following time series regression:

$$\widehat{r}_{\alpha,t} = a + b \times \text{VIX}_t + \epsilon_t$$

where VIX_t is the implied volatility index. We normalize both the estimated pricing error and VIX_t to have unit standard deviation for easier interpretability. The estimates are $\widehat{a} = 1.245$ and $\widehat{b} = 0.2783$, both statistically significant at the 1% level, and confirms our intuition that higher mispricing tends to occur during economic stress.

5.2.3 Out-of-Sample Decomposition

We now implement the out-of-sample decomposition developed in Section 2.2. The prevailing practice in the literature is to estimate a model on some part of the data and then take the estimated model and plug in new data, i.e. data that has not been used in estimation, to obtain an out-of-sample forecast. Deep neural networks have been shown to be among most of successful methods in terms of predictive accuracy in such explorations. We aim to understand the sources of the success better through the lens of our return decompositions, and provide an economically meaningful explanation of the source of predictability.

We apply Algorithm 3.2 to obtain out-of-sample predictions of the individual quantities as well as the "aggregate" forecast for returns. In empirical analyses, a standard measure of performance is the R^2 (in percentage) from equation (5.1). In Table III we present this measure of fit separately for the plugin forecast, $\widehat{m}(\mathbf{x})$, the predicted risk-premium, $\widehat{\mathbf{g}}'_{\beta,t-1}\widehat{\boldsymbol{\lambda}}_{t-1}$, the mispricing components as-well as the sum of the risk premium and mispricing component, $\widehat{\mathbf{g}}_{\alpha,t-1}$. Most economic models suggest that risk premia and exposures to risk premia tend to vary slowly over time (or may even be constant). It is therefore natural to suspect that this component could be (at least partially) predictable. For the mispricing components, theory offers no direct guidance, but conventional economic intuition suggests that arbitrageurs will act to eliminate such opportunities, consequently we expect it to be predictable at best over relatively short horizons. We do not present estimates of the R^2 for the component relating to the factor realization, $\mathbf{g}'_{\beta,t-1}\mathbf{f}_t$, because it is well known since Fama (1965) that factor returns exhibit very low temporal dependence in the time series.

Table III shows that most of the out-of-sample predictabilty of deep neural networks stems from the risk premium component. The predictability stemming from it is about two to three times as large as the predictive ability related to the mispricing component. The column of $R^2_{\widehat{y}}$ measures the predictability of directly plugging the out-of-sample characteristics into the DNN estimated expected return function. The column is all negative over all scenarios under study, reviewing a very bad predictability of this method. The reason for this is that the plugin forecast also contains the "factor term", i.e. the factor exposures multiplied with the past factor shock. Due to the low temporal dependence of returns, this component is unlikely to be systematically related to future returns realizations. Due to its high relative volatility (on average the standard deviation of $\widehat{\mathbf{y}}_t$ is about seven times as large as the standard deviation of $(\widehat{g}_\alpha(\mathbf{x}) + \widehat{g}'_\beta(\mathbf{x})\widehat{\boldsymbol{\lambda}}_t)$), it only adds noise to the prediction, which leads to reduced predictive accuracy.

Table III: Out-of-Sample Decomposition - Expected Returns

This table shows empirical estimates for the out-of-sample decomposition of realized returns (equation (2.6)). $R^2_{\hat{y}}$ measures the quality of the out-of-sample fit from the period-by-period DNN regresssions of excess returns onto characteristics. $R^2_{\beta'\lambda}$ measures how much of excess returns is explained by the factor risk premia, $R^2_{\beta'(F+\lambda)}$ measures how much can be explained by all risk-based components. R^2_{α} measures how much out-of-sample variation of excess returns can be explained by mispricing. All R^2 measure are in percentage. The sample period is 1970 - 2018.

	1 Layer				2 Layers				3 Layers			
K	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$
Panel A: All firms												
1	$\ll 0$	0.44	0.15	0.59	$\ll 0$	0.36	0.14	0.51	$\ll 0$	0.37	0.12	0.49
6	$\ll 0$	0.47	0.14	0.60	$\ll 0$	0.45	0.12	0.55	$\ll 0$	0.42	0.13	0.53
10	$\ll 0$	0.43	0.15	0.58	$\ll 0$	0.43	0.12	0.53	$\ll 0$	0.37	0.16	0.53
Panel B: Large firms												
1	$\ll 0$	0.82	-0.07	0.82	$\ll 0$	0.72	-0.13	0.67	$\ll 0$	0.83	-0.13	0.78
6	$\ll 0$	0.91	-0.11	0.85	$\ll 0$	0.95	-0.16	0.83	$\ll 0$	0.85	-0.07	0.80
10	$\ll 0$	0.81	-0.07	0.80	$\ll 0$	0.90	-0.13	0.79	$\ll 0$	0.69	0.01	0.73
Panel C: Small firms												
1	$\ll 0$	0.39	0.16	0.51	$\ll 0$	0.32	0.17	0.44	$\ll 0$	0.25	0.13	0.36
6	$\ll 0$	0.38	0.18	0.52	$\ll 0$	0.30	0.15	0.42	$\ll 0$	0.29	0.15	0.40
10	$\ll 0$	0.38	0.16	0.51	$\ll 0$	0.28	0.15	0.40	$\ll 0$	0.33	0.14	0.45

In contrast, $R^2_{\alpha+\beta'\lambda}$ is much larger, which also approximately equals the sum of R^2_{α} and $R^2_{\beta'\lambda}$.[10] It reveals that we can obtain greater predictive accuracy by focusing only on the risk premium component and the mispricing component rather than the prevailing practice of plugging new data into the estimated model. In addition,the predictability is nearly entirely attributed to the risk premium $\beta'\lambda$ for large firms, and is attributed to both risk premium (about 76%) and mispricing (about 24%) for small firms. This comparison is also consistent with the in-sample result that mispricing is more strongly concentrated in small firms.

Overall, our analysis shows that the main reason why neural networks are successful in predicting the cross-section of returns stems from their ability to predict risk premia. In addition, and primarily for smaller firms, we gain additional predictive ability from the mispricing component. We improve the out-of-sample forecasts by appealing to our structural decomposition as opposed to the agnostic "plugin forecast". This underscores the need for appealing to theory even in such highly practical applications as predicting the cross section of stock returns. Finally, it is particularly

[10] The orthogonality of α and β in estimations only holds exactly in-sample. It holds approximately for the out-of-sample quantities.

noteworthy that while sensible improvement on the out-of-sample forecast has been achieved, it is important to regard our analysis as a means to provide a structural decomposition of the machine learning forecast, which leads to an economically meaningful interpretation of source of predictability for asset returns.

6 Simulations

To demonstrate the finite sample performance of our method, we simulate a conditional five–factor model for excess returns as in model (2.1). We generate five characteristics $\mathbf{x}_{i,t} = (x_{i,t,1}, \cdots, x_{i,t,5})$ as follows: For each given t and k,

$$x_{i,t,k} = \frac{1}{N+1}\text{rank}(\bar{x}_{i,t,k}), \quad \bar{x}_{i,t,k} = 0.98^k \bar{x}_{i,t-1,k} + \epsilon_{x,i,t,k},$$

where $\epsilon_{x,i,t,k} \sim \mathcal{N}(0,1)$ and $\text{rank}(\bar{x}_{i,t,k})$ is the cross-sectional ranking of $\bar{x}_{i,t,k}$ in $(\bar{x}_{1,t,k}, \cdots, \bar{x}_{N,t,k})$. The five characteristics within the firm i have strong temporal dependence over time, but they are independent across firms. We take five factors, where the j^{th} β-function ($j \leq 5$) are generated as follows:

$$g_{\beta,t,j}(\mathbf{x}) = b_j\phi_j(\mathbf{x}) + a_j, \quad (b_1, \cdots, b_5) = (1, 1, \sqrt{2}, 1, 1), \quad (a_1, \cdots, a_5) = (0, -0.5, -1, 0, 0).$$

Here $\phi_j(\mathbf{x})$ is the j th basis function, chosen as

$$\begin{aligned} \phi_1(\mathbf{x}_{i,t}) &= (x_{i,t,1} - 0.5)^2, \quad \phi_2(\mathbf{x}_{i,t}) = (x_{i,t,1} - 0.5)x_{i,t,2}, \quad \phi_3(\mathbf{x}_{i,t}) = x_{i,t,3}, \\ \phi_4(\mathbf{x}_{i,t}) &= x_{i,t,4}^4, \quad \phi_5(\mathbf{x}_{i,t}) = \max\{x_{i,t,3} - 0.75, 0\}. \end{aligned}$$

To generate $g_{\alpha,t}(\mathbf{x})$ that is orthogonal to $g_{\beta,t}(\mathbf{x})$ period-by-period, we set $g_{\alpha,t}(\mathbf{x}) = [\phi_1(\mathbf{x}), \cdots, \phi_5(\mathbf{x})]\boldsymbol{\theta}_{\alpha,t}$ and calibrate it to approximate the asset's long-run alpha's, namely $\boldsymbol{\theta}_{\alpha,t}$ is obtained by solving the following constraint least squares problem:

$$\min_{\boldsymbol{\theta}_t} \sum_{i=1}^{N} (g_{\alpha,t}(\mathbf{x}_{i,t-1}) - \widehat{a}_i)^2, \qquad \begin{aligned} & g_{\alpha,t}(\mathbf{x}_{i,t-1}) = [\phi_1(\mathbf{x}_{i,t-1}), \cdots, \phi_5(\mathbf{x}_{t-1})]\boldsymbol{\theta}_t, \\ & \sum_{i=1}^{N} g_{\alpha,t}(\mathbf{x}_{i,t-1}) g_{\beta,t}(\mathbf{x}_{i,t-1}) = 0 \end{aligned}$$

with $\widehat{a}_i$ being the estimated alpha for firm i using the Fama-French 5-factor model during 2001-2018. The coefficient $\boldsymbol{\theta}_t$ is solved at each given time t and hence the function $g_{\alpha,t}(\cdot)$ is time-varying. Furthermore, we generate $\boldsymbol{\gamma}_{\alpha,i,t-1} \sim \mathcal{N}(0, \sigma^2_{\gamma 1})$ and $\boldsymbol{\gamma}_{\beta,i,t-1,j} \sim \mathcal{N}(0, \sigma^2_{\gamma 2,j})$, with variance parameters calbirated from the alphas and betas from Fama-French-5-factor-model: the residual variances in the linear regression of alphas and betas respectively regressing on characteristics. Factors and idiosyncratics are generated from $f_{j,t} \sim \mathcal{N}(\mu_{f,j}, \sigma^2_{f,j})$ and $u_{i,t} \sim \mathcal{N}(0, \sigma^2_{u,i})$, with parameters generated using the Fama-French 5-factor model using data during 2001-2018. The factor risk premium is set to $\boldsymbol{\lambda}_{t-1} = a_t \boldsymbol{\mu}_f$, where we calibrate the constant a_t so that $g_{\alpha,t}(\mathbf{x}_{i,t-1})$ explains about 20% variations in the decomposition $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}) = g_{\alpha,t}(\mathbf{x}_{i,t-1}) + g_{\beta,t}(\mathbf{x}_{i,t-1})'\boldsymbol{\lambda}_{t-1}$ at each period. Throughout we fix $N = 500$ firms and $T = 200$ periods.

In-sample estimation

We examine the performance of estimating the alpha $g_{\alpha,t}(\mathbf{x}_{i,t-1})$ and the risk premium $g_{\text{riskP},t}(\mathbf{x}_{i,t-1}) = g_{\beta,t}(\mathbf{x}_{i,t-1})'\boldsymbol{\lambda}_{t-1}$. For each estimated quantity $\widehat{g}_{i,t}$ for $g_{i,t}$ being the alpha or the risk premium, we report the relative mean squared error

$$\text{RMSE}(\widehat{g}) = \frac{\sum_{i=1}^N \sum_{t=1}^T (\widehat{g}_{it} - g_{it})^2}{\sum_{i=1}^N \sum_{t=1}^T g_{it}^2},$$

We compare the proposed method ("DNN-varying") with three additional benchmark methods. The first is linear varying method ("Linear-varying") where the DNN projections in all steps of the algorithms are replaced by linear regressions on the characteristics. The second benchmark is the DNN moving-window method ("DNN-mw") which estimates quantities at time t by fixing a moving-window of twenty-four months $[t-23, \cdots, t]$ as the in-sample period, and estimates alphas and risks by treating them constants within the period. The last benchmark we compare with is the linear moving-window method ("Linear-mw"), which replaces the DNN projection in DNN-mw with linear projections. The moving-window methods have been commonly used as a means of accounting for time-varying alphas and betas. Both DNN-based methods use feedforward three-layer neural networks with number of layers being $16, 8, 4$. We fix the learning rate as 0.1 and use the ReLu activation functions.

Out-of-sample estimation

We conduct out-of-sample comparisons, by refitting the estimated functions $g_{\alpha,t}(\cdot)$ and $g_{\text{riskP},t}(\cdot)$ using the new data $x_{i,t}$ and compare them with the true values. As "pooling" has been one of the standards for predictions in the literature, we also examine the multi-months pooling for forecasts. Specifically, for both the "DNN-varying" and "Linear-varying" methods, after obtaining the in-sample estimates of the alphas and risks as in step S5 of Algorithm 3.2, we estimate functions $g_{\alpha,t}(\cdot)$ and $g_{\text{riskP},t}(\cdot)$ by pooling the in-sample estimates of the previous M months: $t, t-1, ..., t-M$, and regressing on the corresponding pooled characteristics, for $M \in \{0, 3, 6\}$. When $M = 0$, it means we use only the in-sample estimates of the most recent month to estimate the alpha and risk functions. For the two moving-window methods "DNN-mw" and "Linear-mw", we do not pool the in-sample data because these methods treat alphas and risk constant within the estimation window. We compute

$$\text{RMSE}(\widehat{g}) = \frac{\sum_{i=1}^{N} \sum_{t=T+1}^{T+s} (\widehat{g}_t(\mathbf{x}_{i,t}) - g_t(\mathbf{x}_{i,t}))^2}{\sum_{i=1}^{N} \sum_{t=T+1}^{T+s} g_t(\mathbf{x}_{i,t})^2}.$$

for new sampling periods $T+1, \cdots, T+s$ with $s = 100$, where $\widehat{g}_t(\cdot)$ is estimated using in-sample estimates (either alphas or risks) and then evaluated at $\mathbf{x}_{i,t}$ and compare with the true value $g_t(\mathbf{x}_{i,t}) \in \{g_{\alpha,t}(\mathbf{x}_{i,t}), g_{\text{riskP},t}(\mathbf{x}_{i,t})\}$. For the two DNN-based methods, we use the same two-layer networks as in the in-sample estimation.

Table IV reports both the median and standard-deviation of the RMSEs for each method over 100 Monte Carlo repetitions, both in-sample and out-of-sample. The results show that the proposed DNN-varying method outperforms the competing benchmarks in estimating both alphas and risks, closely followed by the moving-window DNN method which accounts for the nonlinearity but not fully the time-varying features. For out-of-sample performance, the race is closer, but the DNN-varying method is still better than competing ones. In addition, we observe the following patterns from the comparisons. First, the alphas are more difficult to estimate/predict than the risk, with larger RMSE for both in-sample and out-of-sample. Second, the two DNN- based methods produce more stable estimates than the linear-based methods, evidenced by the smaller standard deviations of the repeated RMSE. Finally, pooling does not help in improving the out-of-sample predictions: the three pooling approaches (no-pooling, pooling three months, and pooling six months) perform very similarly. This is not surprising because our kernel-smoothing based estimations lead to estimated alphas and risk, both in-sample and out-of-sample, very close in any local window. It provides justifications of not using pooling in our method.

Table IV: In-sample and Out-of-sample RMSE

This table reports the median and standard-deviations of in-sample and out-of-sample RMSE over 100 replications, for four competing methods: the proposed method (DNN-varying); the linear varying method (Linear-varying) where the DNN projection is replaced by linear regressions on the characteristics period-by-period; the DNN moving-window method (DNN-mw) which treats quantities constants in the fixed twenty-four month moving window, and the linear moving-window method (Linear-mw) which replaces the DNN projection in DNN-mw with linear projections. For the out-of-sample estimation, both "DNN-varying" and "Linear-varying" pooled the in-sample estimates of the previous M months: $t, t-2, ..., t-M$ to estimate the alpha and risk functions. Here reported $M \in \{0, 3, 6\}$. The two moving-window methods do not do pooling because they treat in-sample estimates constant in the estimation window.

		alpha g_α		risk $g'_\beta\boldsymbol{\lambda}$	
		median	std ×10	median	std ×10
		In-sample			
DNN-varying		0.880	0.160	0.589	0.145
Linear-varying		1.334	0.687	0.657	0.175
DNN-mw		0.884	0.128	0.763	0.104
Linear-mw		1.420	0.823	0.925	0.209
		Out-of-sample			
DNN-varying	$M=0$	0.893	0.200	0.518	0.271
pooled	$M=3$	0.879	0.204	0.517	0.274
	$M=6$	0.872	0.193	0.519	0.275
Linear-varying	$M=0$	0.972	0.444	0.547	0.286
pooled	$M=3$	0.973	0.448	0.548	0.290
	$M=6$	0.973	0.452	0.551	0.291
DNN-mw		0.946	0.199	0.624	0.216
Linear-mw		1.047	0.539	0.717	0.385

7 Conclusion

While this paper focuss on the estimation using neural networks, our method is suitable for a generic machine learning method. For example, we could derive similar results if predictions were made using random forests.

Our main application in this paper is cross-sectional asset pricing of U.S. equities. The methods developed are however more broadly applicable. In finance a further application (besides the cross-section of other assets) is modeling implied volatility surfaces as in Park et al. (2009). Another possible application is the estimation of consumer demand as in Lewbel (1991), or the in-sample study for portfolio allocations as in the recent work of Lopez-Lira and Roussanov (2020).

References

Ang, A. and D. Kristensen (2012). Testing conditional factor models. *Journal of Financial Economics 106*(1), 132–156.

Arora, S., N. Cohen, W. Hu, and Y. Luo (2019). Implicit regularization in deep matrix factorization. *Advances in Neural Information Processing Systems 32*, 7413–7424.

Bai, J. and S. Ng (2002). Determining the number of factors in approximate factor models. *Econometrica 70*, 191–221.

Bakalli, G., S. Guerrier, and O. Scaillet (2021). A penalized two-pass regression to predict stock returns with time-varying risk premia. *Swiss Finance Institute Research Paper* (21-09).

Bansal, R. and S. Viswanathan (1993). No arbitrage and arbitrage pricing: A new approach. *The Journal of Finance 48*(4), 1231–1262.

Bartlett, P. L., N. Harvey, C. Liaw, and A. Mehrabian (2019). Nearly-tight vc-dimension and pseudodimension bounds for piecewise linear neural networks. *J. Mach. Learn. Res. 20*, 63–1.

Bauer, B. and M. Kohler (2019). On deep learning as a remedy for the curse of dimensionality in nonparametric regression. *The Annals of Statistics 47*(4), 2261–2285.

Belkin, M., D. Hsu, S. Ma, and S. Mandal (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences 116*(32), 15849–15854.

Belkin, M., D. Hsu, and J. Xu (2020). Two models of double descent for weak features. *SIAM Journal on Mathematics of Data Science 2*(4), 1167–1180.

Belkin, M., S. Ma, and S. Mandal (2018). To understand deep learning we need to understand kernel learning. In *International Conference on Machine Learning*, pp. 541–549. PMLR.

Belkin, M., A. Rakhlin, and A. B. Tsybakov (2019). Does data interpolation contradict statistical optimality? In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1611–1619. PMLR.

Bianchi, D., M. Büchner, and A. Tamoni (2021). Bond risk premiums with machine learning. *The Review of Financial Studies 34*(2), 1046–1089.

Chaieb, I., H. Langlois, and O. Scaillet (2021). Factors and risk premia in individual international stock returns. *Journal of Financial Economics 141*(2), 669–692.

Chen, L., M. Pelger, and J. Zhu (2020). Deep learning in asset pricing. *Available at SSRN 3350138*.

Chen, N.-F., R. Roll, and S. A. Ross (1986). Economic forces and the stock market. *Journal of business*, 383–403.

Chen, Q., N. L. Roussanov, and X. Wang (2021). Semiparametric conditional factor models: Estimation and inference. *Available at SSRN*.

Chen, X. and S. C. Ludvigson (2009). Land of addicts? an empirical investigation of habit-based asset pricing models. *Journal of Applied Econometrics 24*(7), 1057–1093.

Connor, G., M. Hagmann, and O. Linton (2012). Efficient semiparametric estimation of the fama–french model and extensions. *Econometrica 80*(2), 713–754.

Connor, G. and R. A. Korajczyk (1986). Performance measurement with the arbitrage pricing theory: A new framework for analysis. *Journal of Financial Economics 15*(3), 373–394.

Du, S. S., X. Zhai, B. Poczos, and A. Singh (2018). Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*.

Dybvig, P. H. and S. A. Ross (1985). Yes, the apt is testable. *The Journal of Finance 40*(4), 1173–1188.

Fama, E. F. (1965). The behavior of stock-market prices. *The journal of Business 38*(1), 34–105.

Fama, E. F. and K. R. French (1992). The cross-section of expected stock returns. *the Journal of Finance 47*(2), 427–465.

Fama, E. F. and J. D. MacBeth (1973). Risk, return, and equilibrium: Empirical tests. *Journal of political economy 81*(3), 607–636.

Fan, J. and I. Gijbels (1996). *Local polynomial modelling and its applications.* Chapman & Hall.

Fan, J., Y. Gu, and W.-X. Zhou (2022). How do noise tails impact on deep relu networks? *arXiv preprint arXiv:2203.10418*.

Fan, J., Y. Liao, and W. Wang (2016). Projected principal component analysis in factor models. *Annals of Statistics 44*(1), 219–254.

Fan, J., C. Ma, and Y. Zhong (2021). A selective overview of deep learning. *Statistical Science 36*(2), 264–290.

Fan, J. and Q. Yao (2003). *Nonlinear time series: nonparametric and parametric methods.* Springer Science & Business Media.

Farrell, M. H., T. Liang, and S. Misra (2021). Deep neural networks for estimation and inference. *Econometrica 89*(1), 181–213.

Ferson, W. E. and C. R. Harvey (1999). Conditioning variables and the cross section of stock returns. *The Journal of Finance 54*(4), 1325–1360.

Forni, M., M. Hallin, M. Lippi, and L. Reichlin (2000). The generalized dynamic factor model: identification and estimation. *The Review of Economics and Statistics 82*, 540–554.

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

Gagliardini, P., E. Ossola, and O. Scaillet (2016). Time-varying risk premium in large cross-sectional equity data sets. *Econometrica 84*(3), 985–1046.

Gagliardini, P., E. Ossola, and O. Scaillet (2020). Estimation of large dimensional conditional factor models in finance. *Handbook of Econometrics*, 219.

Ghysels, E. (1998). On stable factor structures in the pricing of risk: do time-varying betas help or hurt? *The Journal of Finance 53*, 549–573.

Gibbons, M. R., S. A. Ross, and J. Shanken (1989). A test of the efficiency of a given portfolio. *Econometrica: Journal of the Econometric Society*, 1121–1152.

Giglio, S., Y. Liao, and D. Xiu (2021). Thousands of alpha tests. *The Review of Financial Studies 34*(7), 3456–3496.

Giglio, S. and D. Xiu (2021). Asset pricing with omitted factors. *Journal of Political Economy 129*(7), 000–000.

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

Gu, S., B. T. Kelly, and D. Xiu (2019). Autoencoder asset pricing models.

Guijarro-Ordonez, J., M. Pelger, and G. Zanotti (2021). Deep learning statistical arbitrage. *Available at SSRN 3862004*.

Hastie, T., A. Montanari, S. Rosset, and R. J. Tibshirani (2019). Surprises in high-dimensional ridgeless least squares interpolation. *arXiv preprint arXiv:1903.08560*.

Kelly, B. T., S. Malamud, and K. Zhou (2021). The virtue of complexity in return prediction. *Swiss Finance Institute Research Paper* (21-90).

Kelly, B. T., S. Pruitt, and Y. Su (2019). Characteristics are covariances: A unified model of risk and return. *Journal of Financial Economics 134*(3), 501–524.

Kelly, B. T., S. Pruitt, and Y. Su (2020). Instrumented principal component analysis. *Available at SSRN 2983919*.

Kim, S., R. A. Korajczyk, and A. Neuhierl (2021). Arbitrage portfolios. *The Review of Financial Studies 34*(6), 2813–2856.

Kohler, M. and S. Langer (2021). On the rate of convergence of fully connected deep neural network regression estimates. *The Annals of Statistics 49*(4), 2231–2249.

Leitch, G. and J. E. Tanner (1991). Economic forecast evaluation: profits versus the conventional error measures. *The American Economic Review*, 580–590.

Lettau, M. and S. Ludvigson (2001). Resurrecting the (c) capm: A cross-sectional test when risk premia are time-varying. *Journal of political economy 109*(6), 1238–1287.

Lewbel, A. (1991). The rank of demand systems: theory and nonparametric estimation. *Econometrica: Journal of the Econometric Society*, 711–730.

Li, S. and O. B. Linton (2020). A dynamic network of arbitrage characteristics. *Available at SSRN 3638105*.

Lin, H. W., M. Tegmark, and D. Rolnick (2017). Why does deep and cheap learning work so well? *Journal of Statistical Physics 168*(6), 1223–1247.

Lopez-Lira, A. and N. L. Roussanov (2020). Do common factors really explain the cross-section of stock returns? *Available at SSRN 3628120*.

Lu, J., Z. Shen, H. Yang, and S. Zhang (2020). Deep network approximation for smooth functions. *arXiv preprint arXiv:2001.03040*.

McLean, R. D. and J. Pontiff (2016). Does academic research destroy stock return predictability? *The Journal of Finance 71*, 5–32.

Mei, S. and A. Montanari (2019). The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*.

Mhaskar, H., Q. Liao, and T. Poggio (2016). Learning functions: when is deep better than shallow. *arXiv preprint arXiv:1603.00988*.

Onatski, A. (2012). Asymptotics of the principal components estimator of large factor models with weakly influential factors. *Journal of Econometrics 168*(2), 244–258.

Park, B. U., E. Mammen, W. Härdle, and S. Borak (2009). Time series modelling with semiparametric factor dynamics. *Journal of the American Statistical Association 104*(485), 284–298.

Rolnick, D. and M. Tegmark (2017). The power of deeper networks for expressing natural functions. *arXiv preprint arXiv:1705.05502*.

Schmidt-Hieber, J. (2020). Nonparametric regression using deep neural networks with relu activation function. *The Annals of Statistics 48*(4), 1875–1897.

Shanken, J. (1990). Intertemporal asset pricing: An empirical investigation. *Journal of Econometrics 45*(1-2), 99–120.

Shen, Z., H. Yang, and S. Zhang (2021). Neural network approximation: Three hidden layers are enough. *Neural Networks 141*, 160–173.

Shen, Z., H. Yang, and S. Zhang (2022). Optimal approximation rate of relu networks in terms of width and depth. *Journal de Mathématiques Pures et Appliquées 157*, 101–135.

Stock, J. and M. Watson (2002). Macroeconomic forecasting using diffusion indexes. *Journal of Business & Economic Statistics 20*(2), 147–162.

Telgarsky, M. (2016). Benefits of depth in neural networks. In *Conference on learning theory*, pp. 1517–1539. PMLR.

Yarotsky, D. (2017). Error bounds for approximations with deep relu networks. *Neural Networks 94*, 103–114.

Yarotsky, D. (2018). Optimal approximation of continuous functions by very deep relu networks. In *Conference on Learning Theory*, pp. 639–649. PMLR.

Supplement to "Structural Deep Learning in Conditional Asset Pricing"

Jianqing Fan[*] Zheng Tracy Ke[†] Yuan Liao[‡]
Andreas Neuhierl [§]

Abstract

This document contains all technical proofs as well as additional empirical results for robustness studies.

Contents

[*] Department of ORFE, Princeton University. jqfan@princeton.edu.
[†] Department of Statistics, Harvard University. zke@fas.harvard.edu
[‡] Department of Economics, Rutgers University. yuan.liao@rutgers.edu
[§] Olin Business School, Washington University in St. Louis, andreas.neuhierl@wustl.edu

A Technical proofs

A.1 The boundary kernel

We apply the boundary kernel $K_t(\cdot)$ which satisfies the following property: write $l(t) = (1-t)/(Th)$, and $u(t) = (T-t)/(Th)$ then

$$\int_{l(t)}^{u(t)} xK_t(x)dx = 0, \quad t = 1, 2, \cdots, T. \tag{A.1}$$

The use of boundary kernel is important as it does not slow down the rate of convergence at the end-period, which is relevant for out-of-sample forecasts.

To construct such a kernel, let $K_0(\cdot)$ be a baseline kernel function supported on $[-1, 1]$ satisfying $\int_{-1}^{1} xK_0(x)dx = 0$. Now define the boundary kernel

$$K_t(x) = 1\{-1 \leq x \leq 1\}[K_0(x) - a_t]$$

where

$$a_t = \begin{cases} 0 & t \in (Th, T - Th] \\ b_t, \ \text{where } b_t = 2\int_{l(t)}^{1} xK_0(x)dx/\left[1 - l(t)^2\right], & t \leq Th \\ c_t, \ \text{where } c_t = 2\int_{-1}^{u(t)} xK_0(x)dx/\left[u(t)^2 - 1\right], & t > T - Th. \end{cases}$$

It is straightforward to show that (A.1) is satisfied (in Lemma A.1).

As for the baseline kernel K_0, in our empirical studies we use the quartic kernel:

$$K_0(x) = \frac{15}{16}(1 - x^2)^2, \quad -1 \leq x \leq 1$$

which satisfies $\frac{d}{dx}K_0(\pm 1) = 0$, so that $K_t(x)$ is continuous with respect to t at the boundary points $t \in \{Th, T - Th\}$ as $Th \to \infty$. Lemma A.1 verifies additional properties of this kernel.

Lemma A.1. *Suppose $K_0(x) = \frac{15}{16}(1-x^2)^2 1\{-1 \leq x \leq 1\}$. Then at any t $\int_{l(t)}^{u(t)} K_t(x)dx > c_0 > 0$ and $\int_{l(t)}^{u(t)} x^2 K_t(x)dx < \infty$. In addition $\int_{l(t)}^{u(t)} xK_t(x)dx = 0$.*

Proof. When $t \in (Th, T - Th]$, $l(t) \leq -1$ and $u(t) \geq 1$ so $\int_{l(t)}^{u(t)} K_t(x)dx = 1$ and $\int_{l(t)}^{u(t)} x^2 K_t(x)dx = 1/7$. We now consider the boundary cases. Calculus yields $\int_l^u x^m K_0(x)dx = \frac{15}{16}[\frac{x^{m+1}}{m+1} + \frac{x^{m+5}}{m+5} - 2\frac{x^{m+3}}{m+3}]|_l^u$ for $m \in \{0, 1, 2\}$ and $-1 \leq l, u \leq 1$. This implies

$$\begin{aligned} b_t &= \frac{15}{8(1-l(t)^2)}[\frac{x^2}{2}+\frac{x^6}{6}-\frac{x^4}{2}]\Big|_{l(t)}^{1}, \quad \text{if } -1 < l(t) \leq 0 \\ c_t &= \frac{15}{8(u(t)^2-1)}[\frac{x^2}{2}+\frac{x^6}{6}-\frac{x^4}{2}]\Big|_{-1}^{u(t)}, \quad \text{if } 0 < u(t) \leq 1 \\ \int_{l(t)}^{u(t)} K_0(x)dx &= \frac{15}{16}[x+\frac{x^5}{5}-2\frac{x^3}{3}]\Big|_{l}^{u}, \quad \int_{l(t)}^{u(t)} x^2K_0(x)dx = \frac{15}{16}[\frac{x^3}{3}+\frac{x^7}{7}-2\frac{x^5}{5}]\Big|_{l}^{u}. \end{aligned}$$

When $t \leq Th$, $l(t) \in (-1, 0]$ and $u(t) \geq 1$. Then $\int_{l(t)}^{u(t)} K_t = \int_{l(t)}^{1} K_0 - b_t(1 - l(t)) = F(l(t))$

$$F(l) = \int_l^1 K_0(x)dx - 2(1-l)\int_l^1 xK_0/(1-l^2).$$

Note that F is a decreasing function of l since its derivative is negative when $l \in (-1, 0]$. Hence $\int_{l(t)}^{u(t)} K_t \geq F(0) = 3/16$. In addition, $\int_{l(t)}^{u(t)} x^2K_t = \int_{l(t)}^{1} x^2K_0 - (\frac{1}{3} - \frac{l(t)^3}{3})b_t$. The first term is bounded. For the second term, when $l(t)^2 \to 1$, let $F_1(y) = \frac{y}{2} + \frac{y^3}{6} - \frac{y^2}{2}$. Then $\frac{d}{dy}F_1(1) = 0$, implying

$$(\frac{1}{3} - \frac{l(t)^3}{3})b_t \leq |b_t| \leq 2\frac{F_1(1) - F_1(l(t)^2)}{1 - l(t)^2} \leq \frac{d}{dy}F_1(1) + o(1) = o(1).$$

When $l(t)^2$ is bounded away from 1, this term is also bounded. Hence $\int_{l(t)}^{u(t)} x^2K_t(x)dx < \infty$. Similarly, the conclusions hold when $t > T - Th$.

It remains to prove (A.1). We consider three cases.

Case 1: $t \in (Th, T - Th]$. Then $l(t) \leq -1$ and $u(t) \geq 1$ and $K_t(x) = K_0(x)$ where K_0 satisfies $\int_{-1}^{1} xK_0(x)dx = 0$.

Case 2: $t \in (0, Th]$. We have $u(t) \geq 1$ and $l(t) \in (-1, 0]$. This is one of the boundary cases and $K_t(x) = K_0(x) - b_t$. Then by the definition of b_t.

$$\int_{l(t)}^{u(t)} xK_t(x)dx = \int_{l(t)}^{1} xK_0(x)dx - 0.5b_t[1 - l(t)^2] = 0.$$

Case 3: $t > T - Th$. We have $l(t) \leq -1$ and $u(t) \in [0, 1)$. This is another boundary case and $K_t(x) = K_0(x) - c_t$. Then by the definition of c_t.

$$\int_{l(t)}^{u(t)} xK_t(x)dx = \int_{-1}^{u(t)} xK_0(x)dx - 0.5c_t[u(t)^2 - 1] = 0.$$

□

A.2 Proof of Theorem 4.1

Let $\mathbf{\Delta}_t := \widehat{\mathbf{m}}_t - \mathbb{E}(\mathbf{y}_t|\mathbf{X}_{t-1}, \mathbf{f}_t)$, where $\widehat{\mathbf{m}}_t$ is the DNN estimator for $\mathbb{E}(\mathbf{y}_t|\mathbf{X}_{t-1}, \mathbf{f}_t)$. Also, let $\Delta_{i,t}$ denote the i th component of $\mathbf{\Delta}_t$. We shall obtain the rate of convergence for $\|\mathbf{\Delta}_t\|$.

$$m_t^0(\mathbf{x}) := \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1} = \mathbf{x}, \mathbf{f}_t).$$

A.2.1 Convergence of $\widehat{\mathbf{m}}_t - \mathbf{m}_t^0$.

We derive bounds that require the pseudo dimension of the deep neural network class, e.g., Anthony and Bartlett (2009); Bartlett et al. (2019). Let $p(\mathcal{M}_{J,L})$ denote the pseudo dimension of $\mathcal{M}_{J,L}$, defined as the Vapnik-Chervonenkis (VC) dimension of the subgraph class $\{\mathrm{sgn}(h(x) - y) : h \in \mathcal{M}_{J,L}\}$.

The first result of Theorem 4.1 follows from (i) of Proposition A.1 below.

Proposition A.1. *Suppose m_t^0 belongs to the Hölder ball for all t:*

$$\mathcal{H}(q, \gamma, L) = \{f : [a, b]^d \to \mathbb{R}, \|f\|_{q,\gamma} \leq L\}, \quad \|f\|_{q,\gamma} = \sup_{\mathbf{a},\mathbf{b}} \frac{|f^{(q)}(\mathbf{a}) - f^{(q)}(\mathbf{b})|}{\|\mathbf{a} - \mathbf{b}\|^\gamma}.$$

Let $s_0 = \frac{2(q+\gamma)}{2(q+\gamma)+\dim(\mathbf{x}_{i,t-1})}$. Let $d_t(m_1, m_2) := \sqrt{\mathbb{E}[m_1(\mathbf{x}_{i,t-1}) - m_2(\mathbf{x}_{i,t-1})]^2}$, which does not depend on i by the assumption that $\mathbf{x}_{i,t-1}$'s identically distributed across i. Let

$$\begin{aligned} \delta_T^2 &= \frac{p(\mathcal{M}_{J,L}) \log(NT)}{N}, \\ \varphi_T^2 &= \max_t \inf_{m\in\mathcal{M}_{J,L}} \sup_{\mathbf{x}} |m_t^0(\mathbf{x}) - m(\mathbf{x})| = \max_t \|m_t^0 - \pi_N m_t^0\|_\infty. \end{aligned}$$

Suppose $p(\mathcal{M}_{J,L}) \log^{3/2}(NT) + \log^a(NT) = o(N)$ for some $a > 1 + s_0^{-1}$. Also suppose:

(a) there is $q \in \mathbb{R}$, $\gamma \in (0, 1]$ and $L > 0$ so that $m_t^0 \in \mathcal{H}(q, \gamma, L)$.

(b) For any $\epsilon > 0$, $\min_t \inf_{\|m-m_t^0\|_{q,\gamma}>\epsilon} \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^2 > c$ for some $c > 0$.

(c) $\mathbf{x}_{i,t-1}$'s are i.i.d. cross i and e_{it}'s are independent across i.

(d) There are $c_1, c_2 > 0$, $\forall x > 0$, $\max_{it} \mathbb{P}(|e_{it}| > x) \leq c_1 \exp(-c_2 x^2)$.

Then

(i) $\max_t d_t(m_t^0, \widehat{m}_t)^2 \leq O_P(\delta_T^2 + \varphi_T^2)$.

(ii) $\max_t \sup_{\mathbf{x}} |\widehat{m}_t(\mathbf{x}) - m_t^0(\mathbf{x})| = O_P(\varphi_T^{s_0} + \delta_T^{s_0})$.

(iii) $\max_t \frac{1}{N}\sum_i[\widehat{m}_t(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})]^2 = O_P(\delta_T^2 + \varphi_T^2)$.

Proof. (i) Let $\epsilon_T^2 = \bar{C}(\varphi_T^2 + \delta_T^2)$ for some large $\bar{C} > 0$. The goal is to show that

$$\mathbb{P}(\max_t d_t(m_t^0, \widehat{m}_t) > 0.5\epsilon_T) \to 0.$$

step 1 peeling device.

We apply the standard peeling device (e.g., in the proof of Theorem 3.2.5 of van der Vaart and Wellner (1996)). Note that

$$\begin{aligned} A &:= \mathbb{P}(\max_t d_t(m_t^0, \widehat{m}_t) > 0.5\epsilon_T) \leq T\mathbb{P}(d_t(m_t^0, \widehat{m}_t) > 0.5\epsilon_T) \\ &\leq \sum_{k=0}^{\infty} T\mathbb{P}(2^{k-1}\epsilon_T \leq d_t(\widehat{m}_t, m_t^0) \leq 2^k\epsilon_T). \end{aligned}$$

Let $Q_{T,t}(m) = \frac{1}{N}\sum_i (y_{it} - m(\mathbf{x}_{i,t-1}))^2$. We also have

$$\begin{aligned} \max_t |Q_{T,t}(\pi_N m_t^0) - Q_{T,t}(m_t^0)| &\leq 2\varphi_T^2 + \max_t |\frac{4}{N}\sum_i e_{it}(m_t^0(\mathbf{x}_{i,t-1}) - \pi_N m_t^0(\mathbf{x}_{i,t-1}))| \\ &\leq C_1\varphi_T^2 + C_2\frac{\log T}{N} \leq \epsilon_T^2/8 \qquad \text{(A.2)} \end{aligned}$$

for sufficiently large $\bar{C} > 0$ in the definition of ϵ_T, and this holds with probability approaching one. So we now condition on this event. For notational simplicity, all $\mathbb{P}$ throughout this proof refers to this conditional probability.

Define for $k = 0, 1, 2, \cdots$

$$\begin{aligned} \mathcal{E}_{kt} &:= \{m \in \mathcal{M}_{J,L} : 2^{k-1}\epsilon_T \leq d_t(m, m_t^0) \leq 2^k\epsilon_T\} \\ \mathcal{C}_{kt} &:= \{f : f(\varepsilon, \mathbf{x}) = \varepsilon^2 - (\varepsilon + m_t^0(\mathbf{x}) - m(\mathbf{x}))^2 : m \in \mathcal{E}_{kt}\}. \end{aligned}$$

Also let $E_t(f) := \frac{1}{N}\sum_{i=1} f(e_{it}, \mathbf{x}_{i,t-1}) - \mathbb{E}f(e_{it}, \mathbf{x}_{i,t-1})$ for $f \in \mathcal{C}_{k,t}$. Then the events $2^{k-1}\epsilon_T \leq d_t(\widehat{m}_t, m_t^0) \leq 2^k\epsilon_T$ and (A.2) imply

$$\begin{aligned} \sup_{f\in\mathcal{C}_{kt}} E_t(f) &= \sup_{m\in\mathcal{E}_{kt}} [Q_{T,t}(m_t^0) - \mathbb{E}Q_{T,t}(m_t^0)] - [Q_{T,t}(m) - \mathbb{E}Q_{T,t}(m)] \\ &\geq \mathbb{E}Q_{T,t}(\widehat{m}_t) - \mathbb{E}Q_{T,t}(m_t^0) + [Q_{T,t}(m_t^0) - Q_{T,t}(\pi_N m_t^0)] \\ &= d_t(\widehat{m}_t, m_t^0)^2 + [Q_{T,t}(m_t^0) - Q_{T,t}(\pi_N m_t^0)] \\ &\geq (2^{k-1}\epsilon_T)^2 - \epsilon_T^2/8 \geq (2^{k-2}\epsilon_T)^2/2. \end{aligned}$$

Let $B_T \to \infty$ be some truncation sequence. Then

$$\begin{aligned} A &\leq \sum_{k=0}^{\infty} T\mathbb{P}(\sup_{f\in\mathcal{C}_{kt}} E_t(f) \geq (2^{k-2}\epsilon_T)^2/2) \\ &\leq T\sum_{k=0}^{\infty}\mathbb{P}\left(\sup_{f\in\mathcal{C}_{kt}} E_t(f) \geq (2^{k-2}\epsilon_T)^2/2, \max_{it}|e_{it}| \leq B_T\right) \\ &\quad + T\sum_{k=0}^{\infty}\mathbb{P}\left(\sup_{f\in\mathcal{C}_{kt}} E_t(f) \geq (2^{k-2}\epsilon_T)^2/2, \max_{it}|e_{it}| > B_T\right) := A_1 + A_2. \end{aligned}$$

To bound A_1, we apply Lemma 1 of Chen and Shen (1998). While Lemma 1 of Chen and Shen (1998) is for β-mixing data, it admits independent data as a special case. In their notation, set $a_{n1} = 1$ and $a_{2n} = N$. When $\max_{it} |e_{it}| \leq B_T$,

$$\begin{aligned}
\sup_{f \in \mathcal{C}_{kt}} |f(e_{it}, \mathbf{x}_{i,t-1})| &\leq 4B_T |m_t^0(\mathbf{x}_{i,t-1}) - m(\mathbf{x}_{i,t-1})| \leq CB_T(2^k \epsilon_T)^{s_0} := T_k && \text{(A.3)} \\
\sup_{f \in \mathcal{C}_{kt}} \frac{1}{N} \mathrm{var}(\sum_i f(e_{it}, \mathbf{x}_{i,t-1})) &< \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^2 + C\mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^4 \\
&\leq \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^2 \\
&\quad + [\sup_{\mathbf{x}} |m(\mathbf{x}_{i,t-1})| + \sup_{\mathbf{x}} |m_t^0(\mathbf{x}_{i,t-1})|] \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^2 \\
&\leq \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^2 \leq C(2^k \epsilon_T)^2 := \sigma_k^2. && \text{(A.4)}
\end{aligned}$$

Set $M_k = (2^{k-2}\epsilon_T)^2/2$. Then in Lemma 1 of Chen and Shen (1998), condition (a.1) is satisfied for $\xi = 0.5$ and $M_k = (2^{k-2}\epsilon_T)^2/2 \leq \xi\sigma_k^2/4$ for some large C. Condition (a.3) is satisfied for $NM_k/6 > CB_T(2^k\epsilon_T)^{s_0} = T_k$ and $a_{n2} = N$, as long as $N\epsilon_T^{2-s_0} \gg B_T$.

In (A.3), we used the fact that for any $\delta > 0$, there is $s \in (0, 2)$ so that

$$\max_t \sup_{d_t(m_t^0, m) \leq \delta} \sup_{\mathbf{x}} |m_t^0(\mathbf{x}) - m(\mathbf{x})| \leq C\delta^{s_0}, \tag{A.5}$$

which is to be verified in the end.

In the next step, we verify condition (a.3) in Lemma 1 of Chen and Shen (1998).

step 2 the bracketing number.

In this step we bound the bracketing number $\mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2})$. Let $m_1, \cdots, m_{\mathcal{N}}$ be a δ-cover of $\mathcal{M}_{J,L}$ under the sup norm $\|.\|_\infty$ and $\mathcal{N} := \mathcal{N}(\delta, \mathcal{M}_{J,L}, \|.\|_\infty)$. Then for any $f \in \mathcal{C}_{kt}$, where $f(\varepsilon, \mathbf{x}) = (\varepsilon + m_t^0(\mathbf{x}) - m(\mathbf{x}))^2 - \varepsilon^2$, there is m_j such that $\|m - m_j\|_\infty \leq \delta$. Let $f_j(\varepsilon, \mathbf{x}) = (\varepsilon + m_t^0(\mathbf{x}) - m_j(\mathbf{x}))^2 - \varepsilon^2$.

$$\sup_{f \in \mathcal{C}_{kt}, \|m - m_j\|_\infty \leq \delta} |f_j(e_{it}, \mathbf{x}_{it}) - f(e_{it}, \mathbf{x}_{it})| \leq (C+2)|e_{it}|\delta := b(e_{it})\delta$$

Hence $f \in [l_j, u_j]$, where $l_j = f_j - b\delta$ and $u_j = f_j + b\delta$. Moreover, $\mathbb{E}(u_j - l_j)^2 \leq C\delta^2\mathbb{E}e_{it}^2$. This shows that $\{[l_j, u_j] : j \leq \mathcal{N}\}$ is a $C\delta$- bracket of $\mathcal{C}_{kt}$, implying that the bracketing number satisfies

$$\mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2}) \leq \mathcal{N}(C\delta, \mathcal{M}_{J,L}, \|.\|_\infty) \leq \left(\frac{CN}{\delta p(\mathcal{M}_{J,L})} \right)^{p(\mathcal{M}_{J,L})}.$$

where the last inequality follows from Theorem 12.2 of Anthony and Bartlett (2009). Let

$D := p(\mathcal{M}_{J,L}) \log \frac{CN}{p(\mathcal{M}_{J,L})}$. Because $p(\mathcal{M}_{J,L}) = o(N)$,

$$\log \mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2}) \leq D(1 + \log \frac{1}{\delta}). \tag{A.6}$$

Note that $\log y \leq y - 1$ for all $y > 0$. Hence for any small $c_0 > 0$,

$$\begin{aligned} & 2^{12} \int_{M_k/64}^{\sigma_k \sqrt{T_k}} \sqrt{\log \mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2})} d\delta \leq 2^{12} \sqrt{D} \int_{(2^k \epsilon_T)^2/c}^{\sqrt{B_T}(2^k \epsilon_T)^{s_0/2+1}} \sqrt{1 + c_0^{-1} \log \delta^{-c_0}} d\delta \\ \leq \; & C \sqrt{1 + \log(2^k \epsilon_T)^{-2c_0}} \sqrt{D B_T} (2^k \epsilon_T)^{s_0/2+1} \leq C \sqrt{D B_T} (2^k \epsilon_T)^{s_0/2+1-c_0} \\ \leq \; & \sqrt{B_T p(\mathcal{M}_{J,L}) \log N} (2^k \epsilon_T)^{s_0/2+1-c_0} \leq M_k \sqrt{N} \end{aligned} \tag{A.7}$$

where the last inequality holds if $\sqrt{B_T} \delta_T \leq C(\epsilon_T)^{1-s_0/2+c_0}$ and $\epsilon_T = \bar{C}\delta_T + \bar{C}\varphi_T$. We shall prove this claim in the end. Hence we have verified condition (a.3).

step 3 bounding A_1.

We are ready to apply Lemma 1 of Chen and Shen (1998). For $M_k = (2^{k-2}\epsilon_T)^2/2$, and $\sigma_k^2 = C(2^k \epsilon_T)^2$, and because $B_T \epsilon_T^s > c$ for some $c > 0$ (a claim to be proved in the end), so

$$\begin{aligned} & T\mathbb{P}\left(\sup_{f \in \mathcal{C}_{kt}} E_t(f) \geq M_k, \max_{it} |e_{it}| \leq B_T \right) \leq CT \exp\left(-\frac{CNM_k^2}{\sigma_k^2(1 + cT_k)} \right) \\ \leq \; & CT \exp\left(-\frac{CN(2^k \epsilon_T)^2}{(1 + B_T(2^k \epsilon_T)^{s_0})} \right) \leq CT \exp\left(-\frac{CN(2^k)^{2-s_0} \epsilon_T^2}{\epsilon_T^{s_0} B_T} \right) \\ \leq \; & CT \exp\left(-CN(2^k)^{2-s_0} \epsilon_T^{2-s_0} B_T^{-1} \right). \end{aligned}$$

This implies, with $CN\epsilon_T^{2-s_0} B_T^{-1} \geq 2\log(NT)$,

$$\begin{aligned} A_1 \;\; \leq \;\; & T \sum_{k=0}^{\infty} C \exp\left(-CN(2^k)^{2-s_0} \epsilon_T^{2-s_0} B_T^{-1} \right) \leq CT \exp(-CN\epsilon_T^{2-s_0} B_T^{-1}) \\ \leq \;\; & C \exp(-\log T - 2\log(NT)) \to 0. \end{aligned}$$

step 4 bounding A_2.

When $\max_{it} |e_{it}| > B_T$,

$$\begin{aligned} \sup_{f \in \mathcal{C}_{kt}} E_t(f) \;\; \leq \;\; & |\frac{1}{N} \sum_{i=1} (m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))^2 - \mathbb{E}(m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))^2| \\ & + |\frac{1}{N} \sum_{i=1} (m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})) e_{it}| \leq (C + \max_{it} |e_{it}|) \|m - m_t^0\|_\infty \\ \leq \;\; & C \max_{it} |e_{it}| (2^k \epsilon_T)^{s_0}. \end{aligned}$$

$$
\begin{aligned}
A_2 &= T\sum_{k=0}^{\infty}\mathbb{P}\left(\sup_{f\in\mathcal{C}_{kt}} E_t(f) \geq (2^{k-2}\epsilon_T)^2/2, \max_{it}|e_{it}| > B_T\right) \\
&\leq T\sum_{k=0}^{\infty}\mathbb{P}\left(C\max_{it}|e_{it}|(2^k\epsilon_T)^{s_0} \geq (2^{k-2}\epsilon_T)^2, \max_{it}|e_{it}| > B_T\right) \\
&\leq T\sum_{k=0}^{\infty}\mathbb{P}\left(\max_{it}|e_{it}|1\{\max_{it}|e_{it}| > B_T\} \geq c(2^{k-2}\epsilon_T)^{2-s_0}\right) \\
&\leq \sum_{k=0}^{\infty}\frac{T}{2^{(k-2)(2-s_0)/2}}(\epsilon_T)^{-(2-s_0)/2}\mathbb{E}\max_{it}|e_{it}|^{1/2}1\{\max_{it}|e_{it}| > B_T\} \\
&\leq TC(\epsilon_T)^{-(2-s_0)/2}\sqrt{\mathbb{E}\max_{it}|e_{it}|\mathbb{P}(\max_{it}|e_{it}| > B_T)} \\
&\leq TC(\epsilon_T)^{-(2-s_0)/2}\log^{1/4}(NT)\sqrt{NT}\exp(-CB_T^2) \leq (NT)^c\exp(-CB_T^2) \\
&\leq \exp(c\log(NT) - CB_T^2) \to 0
\end{aligned}
$$

The last inequality holds for $B_T^2 \geq C\log(NT)$ for sufficiently large $C > 0$.

step 5 proving claims. It remains to show claims used in the above proofs: (1)$N\epsilon_T^{2-s_0} \gg B_T$, (2) $\sqrt{B_T}\delta_T \leq C(\epsilon_T)^{1-s_0/2+c_0}$ some $c_0 > 0$, (3) $CN\epsilon_T^{2-s_0}B_T^{-1} \geq 2\log(NT)$, (4) $B_T^2 \geq C\log(NT)$, and (A.5). In fact (1)-(3) hold for any $B_T \leq c\delta_T^{-(s_0-2c_0)}$ with $s_0 > 2c_0$. Hence we can choose B_T to satisfy (1)-(4) as long as $C(\log(NT))^{1/2} \leq B_T^2 \leq \delta_T^{-(s_0-2c)}$. Such B_T always exists as long as $\log^a(NT) = O(N)$ for some $a > 1 + s_0^{-1}$.

Finally, to prove (A.5), we apply Lemma 2 of Chen and Shen (1998). By Lemma A.2, $\mathbb{P}(\forall t, \widehat{m}_t \in \mathcal{H}(q,\gamma,2L)) \to 1$. Let

$$
s_0 = \frac{2(q+\gamma)}{2(q+\gamma) + \dim(\mathbf{x}_{i,t-1})}.
$$

Then $\max_t \sup_{d_t(m_t^0,m)\leq\delta}\sup_{\mathbf{x}}|m_t^0(\mathbf{x}) - m(\mathbf{x})| \leq 2(2L)^{1-s_0}\delta^{s_0}$.

(ii) By Lemma A.2, $\mathbb{P}(\forall t, \widehat{m}_t \in \mathcal{H}(q,\gamma,L)) \to 1$ for any $L > 0$. Then by Lemma 2 of Chen and Shen (1998), $\max_t\sup_{\mathbf{x}}|\widehat{m}_t(\mathbf{x}) - m_t^0(\mathbf{x})| \leq 2(2L)^{1-s_0}\max_t d_t(m_t^0,\widehat{m}_t)^{s_0} \leq C\epsilon_T^{s_0}$.

(iii) Recall $\epsilon_T^2 = \bar{C}(\varphi_T^2 + \delta_T^2)$. Note that

$$
\max_t \frac{1}{N}\|\boldsymbol{\Delta}_t\|^2 = \max_t \frac{1}{N}\sum_i(\widehat{m}_t(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))^2 \leq O_P(\epsilon_T^2) + g
$$

where $g = \max_t\sup_{m\in\mathcal{C}}\frac{1}{N}\sum_i[(m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))^2 - \mathbb{E}(m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))^2]$, and

$$
\mathcal{C} = \{m \in \mathcal{M}_{J,L}, \max_t d_t(m_t^0,m)^2 \leq C\epsilon_T^2, \max_t\|m - m_t^0\|_\infty \leq C\epsilon_T^{s_0}\}.
$$

Let $\mathcal{F}_t := \{f : f(\mathbf{x}) = (m(\mathbf{x}) - m_t^0(\mathbf{x}))^2, m \in \mathcal{C}\}$. We now bound g using Lemma 1 of Chen and Shen (1998). Then $g \leq \max_t \sup_{f \in \mathcal{F}_t} \frac{1}{N}\sum_i f(\mathbf{x}_{i,t-1}) - \mathbb{E}f(\mathbf{x}_{i,t-1})$ and

$$\begin{aligned} \sup_{f\in\mathcal{F}_t} |f(\mathbf{x}_{i,t-1})| &\leq \sup_{m\in\mathcal{C}} \|m - m_t^0\|_\infty^2 \leq C\epsilon_T^{2s_0} := G \\ \sup_{f\in\mathcal{F}_t} \frac{1}{N}\text{var}(\sum_i f(\mathbf{x}_{i,t-1})) &< C\sup_{m\in\mathcal{C}} \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^4 \leq C\epsilon_T^2 := \sigma^2. \end{aligned}$$

Set $M = \sigma^2/8$. So their (a.1) (a.3) both are satisfied. As for (a.2), for any small $c_0 \in (0, s_0)$, note that the integral below is bounded by replacing δ with $M/64$.

$$\begin{aligned} & 2^{12}\int_{M/64}^{\sigma\sqrt{G}} \sqrt{\log \mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2})}d\delta \leq 2^{12}\sqrt{D}\int_{M/64}^{\sigma\sqrt{G}}\sqrt{1+\log\delta^{-1}}d\delta \\ \leq\ & C\sqrt{1 + c_0^{-1}\log(M)^{-c_0}}\sqrt{DG\sigma^2} \leq CM^{-c_0/2}\sqrt{DG\sigma^2} \leq M\sqrt{N} \end{aligned}$$

where the last inequality holds for $D := p(\mathcal{M}_{J,L})\log(NT) = O(N)$ and $c_0 < s_0$. We can apply Lemma 1 of Chen and Shen (1998) to reach $g = O_P(\epsilon_T^2)$, given:

$$\begin{aligned} \mathbb{P}(g > M) &\leq T\mathbb{P}(\sup_{f\in\mathcal{F}_t} \frac{1}{N}\sum_i f(\mathbf{x}_{i,t-1}) - \mathbb{E}f(\mathbf{x}_{i,t-1}) > M) \\ &\leq CT\exp(-\frac{CNM^2}{\sigma^2}) \leq CT\exp(-CN\epsilon_T^2) \to 0. \end{aligned}$$

□

Lemma A.2 (Consistency)**.** *Suppose* $\sqrt{\log(NT)}p(\mathcal{M}_{J,L})\log(NT) = o(N)$. *Also suppose:*

(a) there is $q \in \mathbb{R}$, $\gamma \in (0, 1]$ *and* $L > 0$ *so that* $m_t^0 \in \mathcal{H}(q, \gamma, L)$.

(b) For any $\epsilon > 0$, $\min_t \inf_{\|m-m_t^0\|_{q,\gamma}>\epsilon} \mathbb{E}|m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})|^2 > c$ *for some* $c > 0$.

(c) $\mathbf{x}_{i,t-1}$*'s are i.i.d. cross i and* e_{it}*'s are independent across i.*

(d) There are $c_1, c_2 > 0$, $\forall x > 0$, $\max_{it} \mathbb{P}(|e_{it}| > x) \leq c_1\exp(-c_2x^2)$.

Then

(i) $\max_t \sup_{m\in\mathcal{M}_{J,L}} |\frac{1}{N}\sum_i e_{it}m(\mathbf{x}_{i,t-1})| = o_P(1)$ *and*

(ii) $\sup_{m\in\mathcal{M}_{J,L},t} |\frac{1}{N}\sum_i d(m) - \mathbb{E}d(m)| = o_P(1)$ *where* $d(m) = [m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1})]^2$.

(iii) $\max_t \|\widehat{m}_t - m_t^0\|_{q,\gamma} = o_P(1)$.

Proof. (i) Set $B_T := \sqrt{\log(NT)}L$ for sufficiently large $L > 0$.

Let $A_t := \sup_{m\in\mathcal{M}_{J,L}} |\frac{1}{N}\sum_i e_{it}m(\mathbf{x}_{i,t-1})|$. For any $M > 0$,

$$\begin{aligned} \mathbb{P}\left(\max_t A_t > M\right) &= \mathbb{P}\left(\max_t A_t > M, \max_{it}|e_{it}| \leq B_T\right) \\ &\quad +\mathbb{P}\left(\max_t A_t > M, \max_{it}|e_{it}| > B_T\right) := E_1 + E_2. \end{aligned}$$

To bound E_1, we apply Lemma 1 of Chen and Shen (1998). While Lemma 1 of Chen and Shen (1998) is for β-mixing data, it admits independent data as a special case. In their notation, set $a_{n1} = 1$ and $a_{2n} = N$.

Step 1 verify their conditions (a.1) (a.3).

We now verify condition (a.1) in Lemma 1 of Chen and Shen (1998). Let $\mathcal{F} = \{f : f(\varepsilon, \mathbf{x}) = \varepsilon m(\mathbf{x}), m \in \mathcal{M}_{J,L}\}$. When $\max_{it} |e_{it}| \leq B_T$,

$$\sup_{f\in\mathcal{F}} |f(e_{it}, \mathbf{x}_{i,t-1})| \leq CB_T, \quad \sup_{f\in\mathcal{F}} \frac{1}{N}\text{var}(\sum_i f(e_{it}, \mathbf{x}_{i,t-1})) \leq C.$$

Hence their (a.1) and (a.3) hold for sufficiently small M, and $B_T = O(N)$.

step 2 the bracketing number.

In this step we bound the bracketing number $\mathcal{N}_{[]}(\delta, \mathcal{F}, \|.\|_{L^2})$. Let $m_1, \cdots, m_N$ be a δ-cover of $\mathcal{M}_{J,L}$ under the sup norm $\|.\|_\infty$ and $\mathcal{N} := \mathcal{N}(\delta, \mathcal{M}_{J,L}, \|.\|_\infty)$. Then for any $f \in \mathcal{F}$, where $f(\varepsilon, \mathbf{x}) = \varepsilon m(\mathbf{x})$, there is m_j such that $\|m - m_j\|_\infty \leq \delta$. Let $f_j(\varepsilon, \mathbf{x}) = \varepsilon m_j(\mathbf{x})$.

$$\sup_{f\in\mathcal{C}_{kt}, \|m-m_j\|_\infty\leq\delta} |f_j(e_{it}, \mathbf{x}_{it}) - f(e_{it}, \mathbf{x}_{it})| \leq |e_{it}|\delta.$$

Hence $f \in [l_j, u_j]$, where $l_j = f_j - |\varepsilon|\delta$ and $u_j = f_j + |\varepsilon|\delta$. Moreover, $\mathbb{E}(u_j - l_j)^2 \leq 4\delta^2\mathbb{E}e_{it}^2$. This shows that $\{[l_j, u_j] : j \leq \mathcal{N}\}$ is a $C\delta$- bracket of $\mathcal{F}$, implying that the bracketing number satisfies

$$\mathcal{N}_{[]}(\delta, \mathcal{F}, \|.\|_{L^2}) \leq \mathcal{N}(C\delta, \mathcal{M}_{J,L}, \|.\|_\infty) \leq \left(\frac{CN}{\delta p(\mathcal{M}_{J,L})}\right)^{p(\mathcal{M}_{J,L})}.$$

where the last inequality follows from Theorem 12.2 of Anthony and Bartlett (2009). Let $D := p(\mathcal{M}_{J,L}) \log \frac{CN}{p(\mathcal{M}_{J,L})} - p(\mathcal{M}_{J,L})$. By (A.6), $p(\mathcal{M}_{J,L}) \log \frac{CN}{\delta p(\mathcal{M}_{J,L})} \leq D + D\delta^{-1}$. Therefore when $\sqrt{\log(NT)}p(\mathcal{M}_{J,L}) \log N = o(N)$, we have $DB_T = o(N)$,

$$2^{12} \int_0^{\sqrt{B_T}} \sqrt{\log \mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2})}d\delta \leq 2^{12}\sqrt{D} \int_0^{\sqrt{B_T}} \sqrt{1 + \delta^{-1}}d\delta \leq 0.5^{3/2}M\sqrt{N}.$$

This verifies (a.2) in Lemma 1 of Chen and Shen (1998). Hence when $\sqrt{\log(NT)} \log T = o(N)$,

$$E_1 \quad \leq \quad \sum_t \mathbb{P}\left(A_t > M, \max_{it} |e_{it}| \leq B_T\right) \leq CT \exp\left(-\frac{CNM^2}{(1 + cB_T)}\right) \to 0.$$

step 3 bound E_2. For $B_T = \sqrt{\log(NT)}L$ and sufficiently large $L > 0$, So

$\mathbb{P}(\sup_{m\in\mathcal{M}_{J,L}}|\frac{1}{N}\sum_i e_{it}m(\mathbf{x}_{i,t-1})|>M)\to 0$ for any small $M>0$, with

$$E_2 \quad\leq\quad NT\mathbb{P}\left(|e_{it}|>B_T\right)\leq C\exp(\log(NT)-cB_T^2)\to 0.$$

(ii) The proof is very similar to that of (i) so is omitted.

(iii) The inequality $Q_{T,t}(\widehat{m}_t)\leq Q_{T,t}(\pi_N m_t^0)$ implies

$$\frac{1}{N}\sum_i(m_t^0(\mathbf{x}_{i,t-1})-\widehat{m}_t(\mathbf{x}_{i,t-1}))^2\leq\frac{1}{N}\sum_i(m_t^0(\mathbf{x}_{i,t-1})-\pi_N m_t^0(\mathbf{x}_{i,t-1}))^2$$
$$+2\frac{1}{N}\sum_i e_{it}(\widehat{m}_t(\mathbf{x}_{i,t-1})-\pi_N m_t^0(\mathbf{x}_{i,t-1})).$$

Note that $\max_t\|m_t^0-\pi_N m_t^0\|_\infty=o_P(1)$. Results (i) (ii) then imply

$$\max_t\mathbb{E}(m_t^0(\mathbf{x}_{i,t-1})-\widehat{m}_t(\mathbf{x}_{i,t-1}))^2=o(1).$$

It follows from the condition $\min_t\inf_{\|m-m_t^0\|_{q,\gamma}>\epsilon}\mathbb{E}|m(\mathbf{x}_{i,t-1})-m_t^0(\mathbf{x}_{i,t-1})|^2>c$ that for any small $\epsilon>0$, with probability approaching one, $\max_t\|\widehat{m}_t-m_t^0\|_{q,\gamma}<\epsilon$. □

A.2.2 Convergence of $\bar{m}_{i,t}-\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})$

We recall and introduce the following notation.

$$\begin{aligned}
m_i\left(\frac{t}{T}\right) &:= \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})=g_{\alpha,t}(\mathbf{x}_{i,t-1})+g_{\beta,t}(\mathbf{x}_{i,t-1})'\boldsymbol{\lambda}_{t-1}.\\
m_t^0(\mathbf{x}_{i,t-1}) &:= \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1},\mathbf{f}_t)=m_i\left(\frac{t}{T}\right)+g_{\beta,t}(\mathbf{x}_{i,t-1})'[\mathbf{f}_t-\mathbb{E}\mathbf{f}_t],\\
\mathbf{g}_i\left(\frac{t}{T}\right) &= g_{\beta,t}(\mathbf{x}_{i,t-1})\\
\bar{m}_{i,t}^0 &= \frac{1}{Th}\sum_{s=1}^T m_s^0(\mathbf{x}_{i,s-1})K_t\left(\frac{t-s}{Th}\right)A_t^{-1},\quad A_t=\frac{1}{Th}\sum_{s=1}^T K_t\left(\frac{t-s}{Th}\right)\\
\bar{m}_{i,t} &= \frac{1}{Th}\sum_{s=1}^T\widehat{m}_s(\mathbf{x}_{i,s-1})K_t\left(\frac{t-s}{Th}\right)A_t^{-1}
\end{aligned}$$

where the first lines follows from $\mathbb{E}(u_{i,t}|\mathcal{F}_{t-1},\mathbf{f}_t)=0$, $\mathbb{E}(\gamma_{\alpha,i,t-1}|\mathcal{F}_{t-1},\mathbf{f}_t)=0$, and $\mathbb{E}(\boldsymbol{\gamma}_{\beta,i,t-1}|\mathcal{F}_{t-1},\mathbf{f}_t)=0$, the model assumption.

Here $\bar{m}_{i,t}^0$ is the oracle estimator for $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})$ as if $m_t^0(\mathbf{x}_{i,t-1})$ were known. For any twice differentiable scalar function m, let $\dot{m}(v)=\frac{dm(v)}{dv}$ and $\ddot{m}(v)=\frac{d^2m_i(v)}{dv^2}$. Also for any twice differentiable vector function $\mathbf{g}$, let $\dot{\mathbf{g}}(v)=\nabla\mathbf{g}(v)$ and $\ddot{\mathbf{g}}(v)=\nabla^2\mathbf{g}(v)$.

Proposition A.2. *Suppose (i)* $\mathrm{var}(\frac{1}{Th}\sum_{s=1}^T g_{\beta,s}(\mathbf{x}_{i,s-1})'\mathbf{f}_sK_t\left(\frac{t-s}{Th}\right))=O(1/(Th))$.

(ii) $\sup_{v,i}|\dot{m}_i(v)| + |\ddot{m}_i(v)| + \sup_{v,i}\|\dot{\mathbf{g}}_i(v)\| + \|\ddot{\mathbf{g}}_i(v)\| < C.$

Then for each fixed t,

$$\frac{1}{N}\sum_i \mathbb{E}\left|\bar{m}_{i,t} - m_i\left(t/T\right)\right|^2 = O\left(\frac{1}{Th} + h^4 + \delta_T^2 + \varphi_T^2\right).$$

Proof. For notational simplicity, write $K(t,s) := K_t\left(\frac{t-s}{Th}\right)A_t^{-1}$. Then

$$\begin{aligned}
\bar{m}_{i,t}^0 - m_i\left(\frac{t}{T}\right) &= a_1 + a_2 \\
a_1 &:= \frac{1}{Th}\sum_{s=1}^{T}\left(m_i\left(\frac{s}{T}\right) - m_i\left(\frac{t}{T}\right)\right)K(t,s), \quad a_2 := \frac{1}{Th}\sum_{s=1}^{T} g_{\beta,s}(\mathbf{x}_{i,s-1})'[\mathbf{f}_s - \mathbb{E}\mathbf{f}_s]K(t,s).
\end{aligned}$$

We have $\mathbb{E}(\mathbf{f}_s|\mathbf{x}_s) = \mathbb{E}\mathbf{f}_s$ implying $\mathbb{E}a_2 = 0$. Also, $\max_{it}\operatorname{var}(a_2) = O(1/(Th))$. This shows $\frac{1}{N}\sum_i \max_t \mathbb{E}a_2^2 = O((Th)^{-1})$.

As for a_1, by the second order Taylor expansion, for some v,

$$\begin{aligned}
a_1 &= \underbrace{\dot{m}_i(\frac{t}{T})\frac{1}{Th}\sum_{s=1}^{T}\frac{(s-t)}{T}K(t,s)}_{a_{11}} + \underbrace{\frac{1}{Th}\sum_{s=1}^{T}\ddot{m}_i(\frac{v}{T})\frac{(s-t)^2}{T^2}K(t,s)}_{a_{12}} \\
\max_i |a_{12}| &\leq C\frac{1}{Th}\sum_{s=1}^{T}\frac{(s-t)^2}{T^2}K(t,s) \leq Ch^2\left[\int x^2K_t(x)dx + o(1)\right] = O(h^2).
\end{aligned}$$

To bound a_{11}, we apply the property of boundary kernels. Write $\delta(x) = \frac{1}{Th}$, $l(t) = (1-t)/(Th)$, and $u(t) = (T-t)/(Th)$. The kernel $K_t(\cdot)$ satisfies:

$$\int_{l(t)}^{u(t)} xK_t(x)dx = 0, \quad t = 1, 2, \cdots, T, \tag{A.8}$$

proved in Lemma A.1. By the same lemma, $A_t = \frac{1}{Th}\sum_s K_t(\frac{t-s}{Th})$ is bounded away from zero for any t. Hence

$$\begin{aligned}
\max_i |a_{11}| &= \max_i \dot{m}_i(\frac{t}{T})A_t^{-1}h\sum_{x=l}^{u} xK_t(x)\delta(x) = \max_i \dot{m}_i(\frac{t}{T})A_t^{-1}h\left[\int_{l(t)}^{u(t)} xK_t(x)dx + O(\frac{1}{Th})\right] \\
&\leq \max_i |\dot{m}_i(\frac{t}{T})|A_t^{-1}hO(\frac{1}{Th}) = O(T^{-1}).
\end{aligned}$$

Together, $\frac{1}{N}\sum_i \mathbb{E}\left|\bar{m}_{i,t}^0 - m_i\left(t/T\right)\right|^2 = O\left(\frac{1}{Th} + h^4\right)$. In addition, by Proposition A.1, write

$\Delta_{is} := m_s^0(\mathbf{x}_{i,s-1}) - \widehat{m}_s(\mathbf{x}_{i,s-1})$.

$$\begin{aligned}
& \mathbb{E}\frac{1}{N}\sum_i[\bar{m}_{i,t}^0 - \bar{m}_{i,t}]^2 \leq \frac{1}{N}\sum_i \mathbb{E}\left(\frac{1}{Th}\sum_s \Delta_{is}K(t,s)\right)^2 \\
\leq\ & \frac{1}{T^2h^2}\sum_s\sum_l \frac{1}{N}\sum_i \mathbb{E}|\Delta_{is}\Delta_{il}|K(t,s)K(t,l) \leq \max_{sl}\frac{1}{N}\sum_i \mathbb{E}|\Delta_{is}\Delta_{il}|\left(\frac{1}{Th}\sum_s K(t,s)\right)^2 \\
\leq\ & \max_s \frac{1}{N}\sum_i \mathbb{E}\Delta_{is}^2 = O_P(\delta_T^2 + \varphi_T^2). \qquad \text{(A.9)}
\end{aligned}$$

Hence for each fixed t, $\mathbb{E}\frac{1}{N}\sum_i |\bar{m}_{i,t} - m_i(t/T)|^2 = O_P\left(\frac{1}{Th} + h^4 + \delta_T^2 + \varphi_T^2\right)$.

□

A.3 Proof of Theorem 4.2

Proof. **Step 1.** Behavior of eigenvalues. Fix t of interest. Let $\mathbf{M}_s$ denote the $N \times 1$ vector whose i th element is $\widehat{m}_s(\mathbf{x}_{i,s-1}) - \bar{m}_{i,t}$. Let $\widehat{\mathbf{V}}$ and $\mathbf{V}$ denote the $K \times K$ diagonal matrices of the top K eigenvalues of $\frac{1}{NTh}\sum_s \mathbf{M}_s\mathbf{M}_s' K(s,t)$ and $\frac{1}{N}g_{\beta,t-1}\mathbf{S}_f g_{\beta,t-1}'$, where $\mathbf{S}_f = \frac{1}{Th}\sum_s \mathbf{f}_s\mathbf{f}_s' K(s,t)$. As, $\|\mathbf{S}_f - \mathbb{E}\mathbf{f}_t\mathbf{f}_t'\| = o_P(1)$, then the diagonals of $\mathbf{V}$ are bounded away from zero and infinity. Moreover, by Proposition A.3 to be presented below and the Weyl's inequality, for some matrix $\mathbf{B}(t)$.

$$\|\widehat{\mathbf{V}} - \mathbf{V}\| \leq \frac{1}{N}\|\mathbf{B}(t)\|_F = o_P(1).$$

Hence the diagonals of $\widehat{\mathbf{V}}$ are also bounded away from zero and infinity.

Step 2. Convergence of $\mathbf{G}_{\beta,t-1}$. By the definition of eigenvalues/vectors, the following identity holds: $\frac{1}{NTh}\sum_s \mathbf{M}_s\mathbf{M}_s' K(s,t)\widehat{\mathbf{G}}_{\beta,t-1} = \widehat{\mathbf{G}}_{\beta,t-1}\widehat{\mathbf{V}}$. Applying Proposition A.3, and letting $\mathbf{H}_t := \frac{1}{NTh}\sum_s \mathbf{f}_s\mathbf{f}_s' K(s,t)\mathbf{G}_{\beta,t-1}'\widehat{\mathbf{G}}_{\beta,t-1}\widehat{\mathbf{V}}^{-1}$, we have

$$\widehat{\mathbf{G}}_{\beta,t-1} - \mathbf{G}_{\beta,t-1}\mathbf{H}_t = \frac{1}{N}\mathbf{B}(t)\widehat{\mathbf{G}}_{\beta,t-1}\widehat{\mathbf{V}}^{-1}. \qquad \text{(A.10)}$$

This shows that $\frac{1}{N}\|\widehat{\mathbf{G}}_{\beta,t-1} - \mathbf{G}_{\beta,t-1}\mathbf{H}_t\|_F^2 = O_P(\delta_T + \varphi_T + \frac{1}{Th} + h^2)^2$.

Step 3. The risk premium. By definition, $\widehat{\boldsymbol{\lambda}}_{t-1} = \frac{1}{N}\sum_{i=1}^N \widehat{g}_{\beta,t-1,i}\bar{m}_{i,t}$. Then from the following identity,

$$\widehat{\boldsymbol{\lambda}}_{t-1} - \mathbf{H}_t^{-1}\boldsymbol{\lambda}_{t-1} \quad = \quad \frac{1}{N}\sum_{i=1}^N \widehat{g}_{\beta,t-1,i}(\bar{m}_{i,t} - m_i(t/T)) + \frac{1}{N}\sum_{i=1}^N(\widehat{g}_{\beta,t-1,i} - \mathbf{H}_t' g_{\beta,t}(\mathbf{x}_{i,t-1}))g_{\alpha,t}(\mathbf{x}_{i,t-1})'$$

$$+\frac{1}{N}\sum_{i=1}^{N}\widehat{g}_{\beta,t-1,i}(g_{\beta,t}(\mathbf{x}_{i,t-1})'\mathbf{H}_t-\widehat{g}'_{\beta,t-1,i})\mathbf{H}_t^{-1}\boldsymbol{\lambda}_{t-1}+\frac{1}{N}\mathbf{H}_t'\mathbf{G}'_{\beta,t-1}\mathbf{G}_{\alpha,t-1}. \tag{A.11}$$

By Proposition A.2, step 2, and $\frac{1}{N}g'_{\beta,t-1}g_{\alpha,t-1}=O_P(N^{-1/2})$, we can conclude that $\|\widehat{\boldsymbol{\lambda}}_{t-1}-\mathbf{H}_t^{-1}\boldsymbol{\lambda}_{t-1}\|=O_P(\frac{1}{\sqrt{Th}}+h^2+\delta_T+\varphi_T)$. So

$$\frac{1}{N}\|\widehat{\mathbf{G}}_{\beta,t-1}\widehat{\boldsymbol{\lambda}}_{t-1}-\mathbf{G}_{\beta,t-1}\boldsymbol{\lambda}_{t-1}\|^2=O_P(\frac{1}{\sqrt{Th}}+h^2+\delta_T+\varphi_T)^2.$$

Step 4. The alpha. $\widehat{g}_{\alpha,t-1,i}=\bar{m}_{i,t}-\widehat{g}'_{\beta,t-1,i}\widehat{\boldsymbol{\lambda}}_{t-1}$. Hence

$$\widehat{g}_{\alpha,t-1,i}-g_{\alpha,t}(\mathbf{x}_{i,t-1})=\bar{m}_{i,t}-m_i(t/T)+g'_{\beta,t-1,i}\boldsymbol{\lambda}_{t-1}-\widehat{g}'_{\beta,t-1,i}\widehat{\boldsymbol{\lambda}}_{t-1}. \tag{A.12}$$

By Proposition A.2 and step 3, $\frac{1}{N}\|\widehat{g}_{\alpha,t-1}-g_{\alpha,t-1}\|^2=O_P(\frac{1}{\sqrt{Th}}+h^2+\delta_T+\varphi_T)^2$.

Step 5. The factors. Note that

$$\begin{aligned}\widehat{\mathbf{f}}_t &= \frac{1}{N}\widehat{\mathbf{G}}'_{\beta,t-1}\mathbf{M}_t=\frac{1}{N}\sum_{i=1}^{N}\widehat{g}_{\beta,t-1,i}(\bar{m}_{i,t}-\bar{m}_{i,t})\\ &= \mathbf{H}_t^{-1}[\mathbf{f}_t-\mathbb{E}\mathbf{f}_t]+\frac{1}{N}\sum_{i=1}^{N}\widehat{g}_{\beta,t-1,i}z_{it}(t)+\frac{1}{N}\sum_{i=1}^{N}\widehat{g}_{\beta,t-1,i}[g_{\beta,t}(\mathbf{x}_{i,t-1})'\mathbf{H}_t-\widehat{g}_{\beta,t-1,i}]\mathbf{H}_t^{-1}[\mathbf{f}_t-\mathbb{E}\mathbf{f}_t]\end{aligned}$$

where $z_{it}(t)$ is defined in the proof of Proposition A.3. This implies $\widehat{\mathbf{f}}_t-\mathbf{H}_t^{-1}[\mathbf{f}_t-\mathbb{E}\mathbf{f}_t]=O_P(\delta_T+\varphi_T+\eta_T)$. Then

$$\frac{1}{N}\sum_i[\widehat{r}_{factor,t,i}-r_{factor,t}(\mathbf{x}_{i,t-1})]^2=\frac{1}{N}\sum_i[\widehat{g}'_{\beta,t-1,i}\widehat{\mathbf{f}}_t-g_{\beta,t}(\mathbf{x}_{i,t-1})'(\mathbf{f}_t-\mathbb{E}\mathbf{f}_t)]^2=O_P(\delta_T+\varphi_T+\eta_T)^2.$$

□

Proposition A.3. *Suppose (i)* $\operatorname{var}(\mathbf{f}_t)$ *does not vary across t.*

(ii) $\max_{kl}\operatorname{var}\left(\frac{1}{Th}\sum_s\frac{(s-t)}{T}(R_{s,k}-\mathbb{E}R_{s,k})K(s,t)\right)=O(\frac{h^2}{Th})$ *where* $R_{s,k}$ *is the k th element of* $R_s\in\{\mathbf{v}_s,\operatorname{vec}(\mathbf{v}_s\mathbf{v}_s')\}$ *with* $\mathbf{v}_s=\mathbf{f}_s-\mathbb{E}\mathbf{f}_s$. *Then for each fixed t,*

$$\frac{1}{Th}\sum_s\mathbf{M}_s\mathbf{M}_s'K(s,t)=\mathbf{G}_{\beta,t-1}\frac{1}{Th}\sum_s[\mathbf{f}_s-\mathbb{E}\mathbf{f}_s][\mathbf{f}_s-\mathbb{E}\mathbf{f}_s]'K(s,t)\mathbf{G}'_{\beta,t-1}+\mathbf{B}(t)$$

for some $\mathbf{B}(t)$ *such that* $\frac{1}{N^2}\|\mathbf{B}(t)\|_F^2=O_P(\delta_T^2+\varphi_T^2+\frac{1}{(Th)^2}+h^4)$.

Proof. **Step 1.** Bound $\frac{1}{N}\sum_i\|\frac{1}{Th}\sum_s z_{is}(t)\mathbf{v}_sK(s,t)\|^2$ and $\frac{1}{NTh}\sum_{is}z_{is}(t)^2K(s,t)$.

Note that $\widehat{m}_s(\mathbf{x}_{i,s-1})-\bar{m}_{i,t}$ estimates the demeaned expected return, $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1},\mathbf{f}_t)-$

$\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})$, which should be approximately $g_{\beta,t}(\mathbf{x}_{it})'[\mathbf{f}_s - \mathbb{E}\mathbf{f}_s]$. The definition $z_{is}(t)$ below quantifies the estimation error. For each (s,t),

$$\begin{aligned}
z_{is}(t) &:= [\widehat{m}_s(\mathbf{x}_{i,s-1}) - \bar{m}_{i,t}] - g_{\beta,t}(\mathbf{x}_{i,t-1})'[\mathbf{f}_s - \mathbb{E}\mathbf{f}_s] = d_1(i,s) + \cdots + d_4(i) \\
d_1(i,s) &= \widehat{m}_s(\mathbf{x}_{i,s-1}) - m_s^0(\mathbf{x}_{i,s-1}) = \Delta_{is} \\
d_2(i,s) &= m_s^0(\mathbf{x}_{is}) - m_i(s/T) - g_{\beta,t}(\mathbf{x}_{i,t-1})'[\mathbf{f}_s - \mathbb{E}\mathbf{f}_s] = (g_{\beta,s}(\mathbf{x}_{i,s-1}) - g_{\beta,t}(\mathbf{x}_{i,t-1}))'\mathbf{v}_s \\
d_3(i,s) &= m_i(s/T) - m_i(t/T) \\
d_4(i) &= m_i(t/T) - \bar{m}_{i,t}. \qquad (A.13)
\end{aligned}$$

Let $\mathbf{v}_s = \mathbf{f}_s - \mathbb{E}\mathbf{f}_s$. Fix t of interest. By Propositions A.1, A.2,

$$\begin{aligned}
&\frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s d_1(i,s)\mathbf{v}_s K(s,t)\|^2 \leq \max_{sl}\frac{1}{N}\sum_i |\Delta_{is}\Delta_{il}| \left(\frac{1}{Th}\sum_s \|\mathbf{v}_s\| K(s,t)\right)^2 = O_P\left(\delta_T^2 + \varphi_T^2\right) \\
&\frac{1}{NTh}\sum_{is} d_1(i,s)^2 K(s,t) \leq \max_s \frac{1}{N}\sum_i \Delta_{is}^2 \frac{1}{Th}\sum_s K(s,t) = O_P\left(\delta_T^2 + \varphi_T^2\right) \\
&\frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s d_4(i)\mathbf{v}_s K(s,t)\|^2 \leq \frac{1}{N}\sum_i d_4(i)^2 \|\frac{1}{Th}\sum_s \mathbf{v}_s K(s,t)\|^2 \\
\leq\ & O_P\left(\frac{1}{Th} + h^4 + \delta_T^2 + \varphi_T^2\right)\frac{1}{Th} = O_P\left(\frac{1}{(Th)^2} + \frac{h^4}{Th} + \delta_T^2 + \varphi_T^2\right) \\
&\frac{1}{NTh}\sum_{is} d_4(i)^2 K(s,t) \leq O_P\left(\frac{1}{Th} + h^4 + \delta_T^2 + \varphi_T^2\right).
\end{aligned}$$

Next, write $\mathbf{s}_s := \mathbf{v}_s\mathbf{v}_s' - \mathbb{E}\mathbf{v}_s\mathbf{v}_s'$. Also note that $\mathbb{E}\mathbf{v}_s\mathbf{v}_s'$ does not depend on s due to the stationarity. By Taylor expansion, for some v,

$$\begin{aligned}
&\frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s d_2(i,s)\mathbf{v}_s K(s,t)\|^2 \leq \frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s (g_{\beta,s}(\mathbf{x}_{is}) - g_{\beta,t-1}(\mathbf{x}_{it}))'\mathbf{v}_s\mathbf{v}_s' K(s,t)\|^2 \\
\leq\ & \frac{2}{N}\sum_i \|\frac{1}{Th}\sum_s \frac{s-t}{T}\dot{\mathbf{g}}_i\left(\frac{t}{T}\right)'\mathbf{v}_s\mathbf{v}_s' K(s,t)\|^2 + \frac{2}{N}\sum_i \|\frac{1}{Th}\sum_s \frac{(s-t)^2}{T^2}\ddot{\mathbf{g}}_i\left(\frac{v}{T}\right)'\mathbf{v}_s\mathbf{v}_s' K(s,t)\|^2 \\
\leq\ & O_P(1)\max_{kl}\operatorname{var}\left(\frac{1}{Th}\sum_s \frac{(s-t)}{T}\mathbf{s}_{s,kl}K(s,t)\right) + CA_t^{-2}h^2\left[\int_l^u xK(x)dx + O_P(\frac{1}{Th})\right]^2 + O_P(h^4) \\
\leq\ & O_P(\frac{h^2}{Th} + h^2 k_T^2 + h^4)
\end{aligned}$$

where $l = (1-t)/(Th)$ and $u = (T-t)/(Th)$; $k_T = \frac{1}{Th}$ if $t \in (Th, T-Th)$ and $k_T = 1$ for all other t.

Now for some v,

$$\frac{1}{NTh}\sum_{is} d_2(i,s)^2 K(s,t) = \frac{1}{NTh}\sum_{is} \|g_{\beta,s}(\mathbf{x}_{is}) - g_{\beta,t-1}(\mathbf{x}_{it})\|^2 \|\mathbf{v}_s\|^2 K(s,t)$$

$$\le \quad \frac{1}{NTh}\sum_{is}\|\dot{\mathbf{g}}_i(v)\|^2\frac{(s-t)^2}{T^2}\|\mathbf{v}_s\|^2K(s,t) \le O_P(1)\frac{1}{Th}\sum_s\frac{(s-t)^2}{T^2}\mathbb{E}\|\mathbf{v}_s\|^2K(s,t) = O_P(h^2).$$

Finally,

$$\begin{aligned}
&\frac{1}{N}\sum_i\|\frac{1}{Th}\sum_s d_3(i,s)\mathbf{v}_sK(s,t)\|^2 \le \frac{1}{N}\sum_i\|\frac{1}{Th}\sum_s[m_i(s/T)-m_i(t/T)]\mathbf{v}_sK(s,t)\|^2 \\
\le \quad & O_P(1)\max_k \operatorname{var}\left(\frac{1}{Th}\sum_s\frac{(s-t)}{T}v_{s,k}K(s,t)\right) + O_P(h^4) \le O_P(\frac{h^2}{Th}+h^4). \\
&\frac{1}{NTh}\sum_{is}d_3(i,s)^2K(s,t) \le \frac{1}{NTh}\sum_{is}[m_i(s/T)-m_i(t/T)]^2K(s,t) = O_P(h^2).
\end{aligned}$$

Putting together,

$$\begin{aligned}
\frac{1}{N}\sum_i\|\frac{1}{Th}\sum_s z_{is}(t)\mathbf{v}_sK(s,t)\|^2 &= O_P\left(\delta_T^2+\varphi_T^2+\frac{1}{(Th)^2}+h^4\right) \\
\frac{1}{NTh}\sum_{is}z_{is}(t)^2K(s,t) &= O_P\left(\delta_T^2+\varphi_T^2+\frac{1}{Th}+h^2\right).
\end{aligned}$$

Step 2. A decomposition. Now let $\mathbf{M}_s$ and $\mathbf{Z}_s$ denote the $N\times 1$ vectors whose i th elements are respectively $\widehat{m}_s(\mathbf{x}_{i,s-1})-\bar{m}_{i,t}$ and $z_{is}(t)$. Let $\mathbf{G}_{\beta,t-1}$ denote the $N\times K$ matrix of $g_{\beta,t-1}(\mathbf{x}_{it})$. Then $\mathbf{M}_s=\mathbf{Z}_s+\mathbf{G}_{\beta,t-1}(\mathbf{f}_s-\mathbb{E}\mathbf{f}_s)$, $\mathbf{v}_s=(\mathbf{f}_s-\mathbb{E}\mathbf{f}_s)$,

$$\begin{aligned}
\frac{1}{Th}\sum_s\mathbf{M}_s\mathbf{M}_s'K(s,t) &= \mathbf{G}_{\beta,t-1}\frac{1}{Th}\sum_s\mathbf{v}_s\mathbf{v}_s'K(s,t)\mathbf{G}_{\beta,t-1}' + \underbrace{\mathbf{b}_1+\mathbf{b}_2+\mathbf{b}_2'}_{\mathbf{B}(t)} \\
\mathbf{b}_1 &= \frac{1}{Th}\sum_s\mathbf{Z}_s\mathbf{Z}_s'K(s,t), \quad \mathbf{b}_2=\frac{1}{Th}\sum_s\mathbf{Z}_s\mathbf{v}_s'K(s,t)\mathbf{G}_{\beta,t-1}' \\
\frac{1}{N^2}\|\mathbf{b}_1\|_F^2 &\le \frac{1}{N^2}[\frac{1}{Th}\sum_s\|\mathbf{Z}_s\|^2K(s,t)]^2 = [\frac{1}{NTh}\sum_{is}z_{is}(t)^2K(s,t)]^2 \\
&= O_P\left(\delta_T^4+\varphi_T^4+\frac{1}{(Th)^2}+h^4\right) \\
\frac{1}{N^2}\|\mathbf{b}_2\|_F^2 &\le O_P(1)\frac{1}{N}\|\frac{1}{Th}\sum_s\mathbf{Z}_s\mathbf{v}_s'K(s,t)\|_F^2 = O_P(1)\frac{1}{N}\sum_{i=1}^N\|\frac{1}{Th}\sum_s z_{is}(t)\mathbf{v}_sK(s,t)\|^2 \\
&= O_P\left(\delta_T^2+\varphi_T^2+\frac{1}{(Th)^2}+h^4\right). \qquad \text{(A.14)}
\end{aligned}$$

Hence $\frac{1}{N^2}\|\mathbf{b}_1\|_F^2+\frac{1}{N^2}\|\mathbf{b}_2\|_F^2=O_P(\delta_T^2+\varphi_T^2+\frac{1}{(Th)^2}+h^4)$.

□

A.4 Proof of Theorem 4.3: out-of-sample forecasts

Let $b_T := \varphi_N + \eta_T + \delta_T$.

A.4.1 Slower rate of convergence

We start with a result whose proof requires weaker assumptions, with the cost of a slower rate of convergence.

Theorem A.1 (Out-of-Sample Prediction). *Suppose the tuning parameter ν in the constraint (3.3) satisfies: for some sufficiently large $C > 0$,*

$$\nu \geq C \left[b_T + \left(\frac{1}{N} \sum_{i=1}^{N} [g_{\alpha,T+1}(\mathbf{x}_{i,T}) - g_{\alpha,T}(\mathbf{x}_{i,T-1})]^2 \right)^{1/2} \right].$$

Let $s_0 = \frac{2(q+\gamma)}{2(q+\gamma)+\dim(\mathbf{x}_{i,t-1})}$, with (q, γ) as defined in Assumption 4.3. Then

$$\max_{i \leq N} |\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) - g_{\alpha,T+1}(\mathbf{x}_{i,T})| = O_P\left(b_T^{s_0}\right)$$
$$\max_{i \leq N} |\widehat{g}_{riskP,T}(\mathbf{x}_{i,T}) - g_{riskP,T+1}(\mathbf{x}_{i,T})| = O_P\left(b_T^{s_0}\right).$$

As shown in Theorem A.1, the prediction rate has an additional parameter $s_0 < 1$ compared to that of the in-sample result. This parameter slightly slows down the rate of convergence.

A.4.2 Proof of Theorem 4.3: Sharp predictive rate of convergence

To achieve sharp prediction rate of convergence at $O_P(b_T)$, we rely on the Riesz representation theorem and requires one more assumption. First, for a generic function h, let $\pi_N h$ denote its projection on the DNN space $\mathcal{M}_{J,L}$. Define the space

$$\mathcal{A} = \text{span}(\mathcal{B}_1 \cup \mathcal{B}_2), \quad \mathcal{B}_1 = \mathcal{M}_{J,L} - \{\pi_N g_{\alpha,T}\}, \quad \mathcal{B}_2 = \mathcal{M}_{J,L} - \{\pi_N g_{\text{riskP},T}\},$$

where $\text{span}(\mathcal{B}_1 \cup \mathcal{B}_2)$ denotes the closed linear span of $\mathcal{B}_1 \cup \mathcal{B}_2$. Define an inner product:

$$\langle h_1, h_2 \rangle := \mathbb{E}(h_1(\mathbf{x}_{i,T-1}) h_2(\mathbf{x}_{i,T-1})), \quad \forall h_1, h_2 \in \mathcal{A}.$$

Hence $\mathcal{A}$ is a Hilbert space endowed with this inner product. Evaluated at the out-of-sample $\mathbf{x}_{i,T}$, define the following linear functional on $\mathcal{A}$:

$$\mathcal{T}_i(h) := h(\mathbf{x}_{i,T}), \quad h \in \mathcal{A}, \quad \forall i = 1, \cdots, N.$$

Because $\mathcal{T}_i$ is a linear functional on the Hilbert space, the Riesz representation theorem implies that there is a function $v_i^* \in \mathcal{A}$, called Riesz representer, so that

$$\mathcal{T}_i(h) = \langle h, v_i^* \rangle, \forall h \in \mathcal{A}, \quad \forall i = 1, \cdots, N. \tag{A.15}$$

We impose the following assumption.

Assumption A.1. *The Riesz representer v_i^* satisfies:* $\max_{i \leq N} \mathbb{E} v_i^*(\mathbf{x}_{j,T-1})^2 < C$.

Proof of Theorem 4.3:

Proof. The proof requires Assumption A.1.

We focus on the proof for estimating $g_{\alpha,T}()$; the proof for estimating $g_{\beta,T}()$ is similar. By the Riesz representation theorem, uniformly for $j \leq N$,

$$\begin{aligned}
\widehat{g}_{\alpha,T}(\mathbf{x}_{j,T}) - \pi_N g_{\alpha,T}(\mathbf{x}_{j,T}) &= \mathcal{T}_j(\widehat{g}_{\alpha,T} - \pi_N g_{\alpha,T}) = \langle \widehat{g}_{\alpha,T} - \pi_N g_{\alpha,T}, v_j^* \rangle \\
&= \mathbb{E}(\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T-1}) - \pi_N g_{\alpha,T}(\mathbf{x}_{i,T-1})) v_j^*(\mathbf{x}_{i,T-1}) \\
&\leq O(1) \left[\mathbb{E}(\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T-1}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2\right]^{1/2} + O_P(\varphi_T) \\
&= O_P(b_T).
\end{aligned}$$

where the inequality follows from $\max_j \mathbb{E} v_j^*(\mathbf{x}_{i,T-1})^2 < C$; the last equality is due to (A.17).

$$\max_{j \leq N} |\widehat{g}_{\alpha,T}(\mathbf{x}_{j,T}) - g_{\alpha,T}(\mathbf{x}_{j,T})| \leq O_P(b_T + \varphi_T) = O_P(b_T).$$

□

A.4.3 Proof of Theorem A.1

Proof. The proof of Theorem A.1 does not require Assumption A.1.

We focus on the convergence for predicting the alpha $g_{\alpha,T+1}(\mathbf{x}_{i,T+1})$. The proof for

$$\sup_{\mathbf{x}} |\widehat{g}_{\text{riskP},T}(\mathbf{x}) - g_{\text{riskP},T}(\mathbf{x})| = O_P(b_T^{s_0})$$

is mostly the same (but is simpler as it does not require constraints).

The proof is based on an interpolation result (see Step 3 below, cited from Chen and Shen (1998)), which bounds $\sup_{\mathbf{x}} |\widehat{g}_{\alpha,T}(\mathbf{x}) - g_{\alpha,T+1}(\mathbf{x})|$ using $\mathbb{E}|\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) - g_{\alpha,T+1}(\mathbf{x}_{i,T}))^2$ directly. Recall that $g_{\alpha,T+1}(\cdot)$ is the true out-of-sample alpha function at time $T+1$, and $\pi_N g_{\alpha,T+1}$ denotes its projection to the DNN space.

Step 1. In-sample mean squared error. Because $\pi_N g_{\alpha,T+1}(\cdot) \in \mathcal{M}_{J,L}$ satisfies the constraint, as proved in Lemma A.3,

$$\begin{aligned}
&\frac{1}{N}\sum_i (\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T-1}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2 \leq a\frac{2}{N}\sum_i (\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T-1}) - \widehat{g}_{\alpha,T-1,i})^2 \\
&+\frac{2}{N}\sum_i (\widehat{g}_{\alpha,T-1,i} - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2 \\
\leq\ & \frac{8}{N}\sum_i (g_{\alpha,T+1}(\mathbf{x}_{i,T-1}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2 + \frac{8}{N}\sum_i (g_{\alpha,T}(\mathbf{x}_{i,T-1}) - \widehat{g}_{\alpha,T-1,i})^2 \\
&+\frac{8}{N}\sum_i (\pi_N g_{\alpha,T+1}(\mathbf{x}_{i,T-1}) - g_{\alpha,T+1}(\mathbf{x}_{i,T-1}))^2 + O_P(b_T^2) \\
\leq\ & \sup_{\mathbf{x}} |g_{\alpha,T+1}(\mathbf{x}) - g_{\alpha,T}(\mathbf{x})|^2 + O_P(b_T^2) = O_P(b_T^2). \qquad \text{(A.16)}
\end{aligned}$$

For sufficiently large $\bar{C} > 0$, $\epsilon_T := \bar{C} b_T$. For any $\epsilon > 0$, we can choose $\bar{C}$ so that

$$\mathbb{P}\left(\frac{1}{N}\sum_i (\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T-1}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2 > \epsilon_T^2/8\right) < \epsilon.$$

From $\frac{1}{N}\sum_i (\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T-1}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2$ to $d_T(\widehat{g}_{\alpha,T}, g_{\alpha,T})$, we apply the peeling device in Step 3 below.

Step 2. Bound for $d_T(\widehat{g}_{\alpha,T}, g_{\alpha,T})$. For notational simplicity, write $(\widehat{g}, g) := (\widehat{g}_{\alpha,T}, g_{\alpha,T})$ and $d(a,b) := d_T(a,b) = \sqrt{\mathbb{E}[a(\mathbf{x}_{i,T-1}) - b(\mathbf{x}_{i,T-1})]^2}$. The proof is very similar to that of Proposition A.1. We simply write

$$\begin{aligned}
\mathcal{E}_k &:= \{m \in \mathcal{M}_{J,L} \cap H(q,\gamma,L) : 2^{k-1}\epsilon_T \leq d(m,g) \leq 2^k\epsilon_T, \frac{1}{N}\sum_i (m(\mathbf{x}_{i,T-1}) - g(\mathbf{x}_{i,T-1}))^2 < \epsilon_T^2/8\} \\
\mathcal{C}_k &:= \{f : f(\mathbf{x}) = -(m(\mathbf{x}) - g(\mathbf{x}))^2 : m \in \mathcal{E}_k\},
\end{aligned}$$

wheras g denotes the true alpha-function. Then $\widehat{g} \in \mathcal{E}_k$ implies

$$\begin{aligned}
&\sup_{f\in\mathcal{C}_k} \frac{1}{N}\sum_i f(\mathbf{x}_{i,t-1}) - \mathbb{E}f(\mathbf{x}_{i,t-1}) \\
=\ & \sup_{m\in\mathcal{E}_k} -\frac{1}{N}\sum_i (m(\mathbf{x}_{i,t-1}) - g(\mathbf{x}_{i,t-1}))^2 + \mathbb{E}(m(\mathbf{x}_{i,t-1}) - g(\mathbf{x}_{i,t-1}))^2 \\
\geq\ & -\frac{1}{N}\sum_i (\widehat{g}(\mathbf{x}_{i,t-1}) - g(\mathbf{x}_{i,t-1}))^2 + d(\widehat{g},g)^2 \geq d(\widehat{g},g)^2 - \epsilon_T^2/8 \geq (2^{k-2}\epsilon_T)^2/2. \\
A\ :=\ & \mathbb{P}(d(\widehat{g},g) > 0.5\epsilon_T) \leq \sum_{k=0}^{\infty} \mathbb{P}(\widehat{g} \in \mathcal{E}_k) + \epsilon \\
\leq\ & \sum_{k=0}^{\infty} \mathbb{P}(\sup_{f\in\mathcal{C}_k} \frac{1}{N}\sum_i f(\mathbf{x}_{i,t-1}) - \mathbb{E}f(\mathbf{x}_{i,t-1}) \geq (2^{k-2}\epsilon_T)^2/2) + \epsilon.
\end{aligned}$$

By Lemma 2 of Chen and Shen (1998),

$$\begin{aligned}
\sup_{f\in\mathcal{C}_k}|f(\mathbf{x}_{i,t-1})| &\leq \sup_{m\in\mathcal{E}_k}\sup_{\mathbf{x}}(m(\mathbf{x})-g(\mathbf{x}))^2 \leq \sup_{m\in\mathcal{E}_k}|2(2L)^{1-s_0}d(m,g)^{s_0}|^2 \leq C(2^k\epsilon_T)^{s_0}\\
\sup_{f\in\mathcal{C}_k}\frac{1}{N}\mathrm{var}(\sum_i f(\mathbf{x}_{i,t-1})) &\leq \sup_{m\in\mathcal{E}_k}\mathbb{E}|m(\mathbf{x}_{i,t-1})-g(\mathbf{x}_{i,t-1})|^4 \leq C\sup_{m\in\mathcal{E}_k}d(m,g)^2 \leq C(2^k\epsilon_T)^2.
\end{aligned}$$

From a very similar as bounding term A_1 in the proof Proposition A.1 to check that all conditions of Lemma 1 of Chen and Shen (1998) are verified (omitting details for brevity).

$$\begin{aligned}
A &\leq \sum_{k=0}^{\infty}\mathbb{P}(\sup_{f\in\mathcal{C}_k}\frac{1}{N}\sum_i f(\mathbf{x}_{i,t-1})-\mathbb{E}f(\mathbf{x}_{i,t-1})\geq M_k)+\epsilon\\
&\leq \sum_{k=0}^{\infty}\exp\left(-\frac{CN(2^k\epsilon_T)^2}{(1+(2^k\epsilon_T)^{s_0})}\right)+\epsilon \leq \sum_{k=0}^{\infty}\exp\left(-CN(2^k)^{2-s_0}\epsilon_T^{2-s_0}\right)+\epsilon\leq 2\epsilon.
\end{aligned}$$

This implies $d(\widehat{g},g)=O_P(\epsilon_T)=O_P(b_T)$, meaning

$$\mathbb{E}(\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T-1})-g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2=O_P(b_T^2). \tag{A.17}$$

Step 3. Out-of-sample prediction. By Lemma 2 of Chen and Shen (1998), for any $\epsilon>0$, there is $C>0$, with probability at least $1-\epsilon$,

$$\sup_{\mathbf{x}}|\widehat{g}_{\alpha,T}(\mathbf{x})-g_{\alpha,T}(\mathbf{x})|\leq 2(2L)^{1-s_0}d(\widehat{g}_{\alpha,T},g_{\alpha,T})^{s_0}\leq Cb_T^{s_0}. \tag{A.18}$$

$$\max_i|\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T})-g_{\alpha,T+1}(\mathbf{x}_{i,T})|\leq\sup_{\mathbf{x}}|\widehat{g}_{\alpha,T}(\mathbf{x})-g_{\alpha,T}(\mathbf{x})|+\sup_{\mathbf{x}}|g_{\alpha,T}(\mathbf{x})-g_{\alpha,T+1}(\mathbf{x})|\leq Cb_T^{s_0}.$$

□

Lemma A.3 (feasibility of $\pi_N g_{\alpha,T+1}$)**.** *The following inequality holds, which does not require Assumption A.1:* $\|\frac{1}{N}\sum_i\pi_N g_{\alpha,T+1}(\mathbf{x}_{i,T})(\widehat{g}_{\beta,T-1,i},1)\|\leq\nu$.

Proof. $\pi_N g_{\alpha,T+1}(\cdot)$ satisfies the constraint due to the following inequality:

$$\begin{aligned}
&\|\frac{1}{N}\sum_i\pi_N g_{\alpha,T+1}(\mathbf{x}_{i,T})(\widehat{g}_{\beta,T-1,i},1)\|\leq a_1+\cdots+a_5\\
a_1 &= \|\frac{1}{N}\sum_i\pi_N g_{\alpha,T+1}(\mathbf{x}_{i,T})(\widehat{g}_{\beta,T-1,i}-g_{\beta,T}(\mathbf{x}_{i,T-1})\|=O_P(b_T)\\
a_2 &= \|\frac{1}{N}\sum_i[\pi_N g_{\alpha,T+1}(\mathbf{x}_{i,T})-g_{\alpha,T+1}(\mathbf{x}_{i,T})](g_{\beta,T}(\mathbf{x}_{i,T-1}),1)\|=O_P(b_T)\\
a_3 &= \|\frac{1}{N}\sum_i g_{\alpha,T+1}(\mathbf{x}_{i,T})\|=O_P(N^{-1/2})
\end{aligned}$$

$$\begin{aligned} a_4 &= \|\frac{1}{N}\sum_i (g_{\alpha,T+1}(\mathbf{x}_{i,T}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))g_{\beta,T}(\mathbf{x}_{i,T-1})\| \leq O_P(\frac{1}{N}\sum_i (g_{\alpha,T+1}(\mathbf{x}_{i,T}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2)^{1/2} \\ a_5 &= \|\frac{1}{N}\sum_i g_{\alpha,T}(\mathbf{x}_{i,T-1})g_{\beta,T}(\mathbf{x}_{i,T-1})\| = O_P(N^{-1/2}). \end{aligned}$$

□

A.5 Proof of Theorem 4.4: out-of-sample decomposition

Proof. Theorem 4.3 shows, uniformly in $i \leq N$,

$$\begin{aligned} y_{i,T+1} &= g_{\alpha,T+1}(\mathbf{x}_{i,T}) + g_{\text{riskP},T+1}(\mathbf{x}_{i,T}) + g_{\text{factor},T+1}(\mathbf{x}_{i,T}) + \varepsilon_{i,T+1} \\ &= \widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) + \widehat{g}_{\text{riskP},T}(\mathbf{x}_{i,T}) + g_{\text{factor},T+1}(\mathbf{x}_{i,T}) + \varepsilon_{i,T+1} + O_P(b_T). \end{aligned}$$

Now let $\mathcal{F}_T$ be the filtration generated by $\{X_t : t = 1, \cdots, T\}$. Then

$$\mathbb{E}(g_{\text{factor},T+1}(\mathbf{x}_{i,T})|\mathcal{F}_T) = g_{\beta,T+1}(\mathbf{x}_{i,T})'\mathbb{E}(\mathbf{f}_{T+1} - \mathbb{E}\mathbf{f}_{T+1}|\mathcal{F}_T) = 0$$

with $\mathbb{E}\mathbf{f}_{T+1} = \mathbb{E}(\mathbf{f}_{T+1}|\mathcal{F}_T)$. In addition, $\varepsilon_{i,t+1} = \gamma_{\alpha,it} + \boldsymbol{\gamma}'_{\beta,it}\boldsymbol{\lambda}_t + \boldsymbol{\gamma}'_{\beta,it}(\mathbf{f}_{t+1} - \mathbb{E}\mathbf{f}_t) + u_{i,t+1}$. Hence $\mathbb{E}(\varepsilon_{i,T+1}|\mathcal{F}_T) = 0$ provided that $\mathbb{E}(u_{i,t}|\mathcal{F}_{t-1}, \mathbf{f}_t) = 0$, $\mathbb{E}(\gamma_{\alpha,i,t-1}|\mathcal{F}_{t-1}, \mathbf{f}_t) = 0$, and $\mathbb{E}(\boldsymbol{\gamma}_{\beta,i,t-1}|\mathcal{F}_{t-1}, \mathbf{f}_t) = 0$.

□

References

Anthony, M. and P. L. Bartlett (2009). *Neural network learning: Theoretical foundations.* cambridge university press.

Bartlett, P. L., N. Harvey, C. Liaw, and A. Mehrabian (2019). Nearly-tight vc-dimension and pseudodimension bounds for piecewise linear neural networks. *J. Mach. Learn. Res. 20*, 63–1.

Chen, X. and X. Shen (1998). Sieve extremum estimates for weakly dependent data. *Econometrica*, 289–314.

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

van der Vaart, A. and J. Wellner (1996). *Weak convergence and empirical processes* (The First Edition ed.). Springer.

B Additional Figures and Tables

Table I: In-Sample Decomposition - Realized Returns (Early Sample)

This table shows empirical estimates for the in-sample decomposition of realized returns (equation (2.5)). R^2 quantities are as defined in Table I of the paper. All R^2 measure are in percentage. The sample period is 1970 - 1999.

	1 Layer					2 Layers					3 Layers				
K	$R^2_{\hat{y}}$	$R^2_{\beta'F}$	$R^2_{\beta'\lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}	$R^2_{\hat{y}}$	$R^2_{\beta'F}$	$R^2_{\beta'\lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}	$R^2_{\hat{y}}$	$R^2_{\beta'F}$	$R^2_{\beta'\lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}
Panel A: All firms															
1	24.41	15.26	1.35	16.46	0.00	26.71	15.32	1.35	16.53	-0.01	30.39	16.66	1.31	17.80	0.21
6	24.41	19.61	1.51	20.93	-0.15	26.71	20.32	1.50	21.64	-0.16	30.39	23.29	1.57	24.66	-0.05
10	24.41	20.61	1.52	21.94	-0.17	26.71	21.82	1.54	23.18	-0.20	30.39	25.13	1.59	26.53	-0.07
Panel B: Large firms															
1	35.45	22.07	1.75	23.61	0.11	35.77	22.16	1.73	23.70	0.11	36.05	20.47	1.63	21.92	0.53
6	35.45	31.90	2.08	33.89	-0.14	35.77	31.95	2.06	33.95	-0.19	36.05	31.45	2.12	33.46	0.02
10	35.45	33.14	2.15	35.12	-0.25	35.77	33.09	2.17	35.11	-0.31	36.05	32.51	2.15	34.54	-0.06
Panel C: Small firms															
1	19.79	9.46	1.52	10.85	0.14	23.73	9.56	1.50	10.94	0.12	31.56	14.10	1.44	15.38	0.29
6	19.79	13.24	1.58	14.58	0.00	23.73	14.45	1.57	15.76	-0.05	31.56	21.59	1.69	22.95	0.08
10	19.79	14.37	1.59	15.70	-0.03	23.73	16.61	1.60	17.92	-0.09	31.56	24.13	1.70	25.50	0.06

Table II: In-Sample Decomposition - Realized Returns (Late Sample)

This table shows empirical estimates for the in-sample decomposition of realized returns (equation (2.5)). R^2 quantities are as defined in Table I of the paper. R^2 measure are in percentage. The sample period is 2000 - 2018.

	1 Layer					2 Layers					3 Layers				
K	$R^2_{\hat{y}}$	$R^2_{\beta'F}$	$R^2_{\beta'\lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}	$R^2_{\hat{y}}$	$R^2_{\beta'F}$	$R^2_{\beta'\lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}	$R^2_{\hat{y}}$	$R^2_{\beta'F}$	$R^2_{\beta'\lambda}$	$R^2_{\beta'(F+\lambda)}$	R^2_{α}
Panel A: All firms															
1	29.18	17.85	0.55	18.61	0.09	28.85	17.90	0.57	18.65	0.05	30.33	18.06	0.56	18.81	0.11
6	29.18	24.19	0.76	25.10	-0.12	28.85	23.99	0.76	24.89	-0.15	30.33	24.68	0.77	25.59	-0.10
10	29.18	25.30	0.79	26.23	-0.15	28.85	25.11	0.80	26.05	-0.18	30.33	26.10	0.78	27.03	-0.11
Panel B: Large firms															
1	36.49	24.53	-0.33	24.65	-0.30	36.8	24.86	-0.28	24.99	-0.40	36.98	24.91	-0.25	25.07	-0.24
6	36.49	34.98	-0.14	35.50	-0.31	36.8	34.81	-0.17	35.37	-0.30	36.98	34.43	-0.10	35.02	-0.17
10	36.49	36.04	-0.11	36.58	-0.31	36.8	35.84	-0.10	36.36	-0.35	36.98	35.54	-0.11	36.11	-0.12
Panel C: Small firms															
1	24.88	10.98	1.06	12.08	0.38	23.8	10.88	1.05	11.98	0.31	26.57	11.06	1.04	12.14	0.44
6	24.88	17.20	1.34	18.30	0.03	23.8	16.49	1.32	17.59	-0.01	26.57	17.94	1.31	19.03	0.09
10	24.88	18.62	1.37	19.75	-0.01	23.8	18.09	1.35	19.19	-0.04	26.57	19.99	1.35	21.08	0.04

Table III: Out-of-Sample Decomposition - Expected Returns (Early Sample)

This table shows empirical estimates for the out-of-sample decomposition of realized returns (equation (2.6)). R^2 quantities are as defined in Table I of the paper. All R^2 measure are in percentage. The sample period is 1970 - 1999.

	1 Layer				2 Layers				3 Layers			
K	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$
Panel A: All firms												
1	$\ll 0$	0.56	0.17	0.73	$\ll 0$	0.42	0.17	0.59	$\ll 0$	0.43	0.14	0.57
6	$\ll 0$	0.58	0.15	0.73	$\ll 0$	0.54	0.16	0.67	$\ll 0$	0.52	0.17	0.65
10	$\ll 0$	0.55	0.17	0.72	$\ll 0$	0.51	0.15	0.64	$\ll 0$	0.47	0.18	0.65
Panel B: Large firms												
1	$\ll 0$	1.19	0.10	1.33	$\ll 0$	0.99	0.00	1.07	$\ll 0$	1.10	-0.02	1.13
6	$\ll 0$	1.37	-0.01	1.39	$\ll 0$	1.26	-0.02	1.26	$\ll 0$	1.15	0.05	1.19
10	$\ll 0$	1.17	0.09	1.30	$\ll 0$	1.20	0.02	1.23	$\ll 0$	1.02	0.16	1.18
Panel C: Small firms												
1	$\ll 0$	0.47	0.17	0.58	$\ll 0$	0.36	0.17	0.48	$\ll 0$	0.28	0.13	0.39
6	$\ll 0$	0.42	0.18	0.56	$\ll 0$	0.36	0.16	0.48	$\ll 0$	0.34	0.15	0.45
10	$\ll 0$	0.46	0.17	0.58	$\ll 0$	0.34	0.15	0.46	$\ll 0$	0.39	0.13	0.49

Table IV: Out-of-Sample Decomposition - Expected Returns (Late Sample)

This table shows empirical estimates for the out-of-sample decomposition of realized returns (equation (2.6)). R^2 quantities are as defined in Table I of the paper. All R^2 measure are in percentage. The sample period is 2000 - 2018.

	1 Layer				2 Layers				3 Layers			
K	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$	$R^2_{\hat{y}}$	$R^2_{\beta'\lambda}$	R^2_{α}	$R^2_{\alpha+\beta'\lambda}$
Panel A: All firms												
1	$\ll 0$	0.29	0.12	0.41	$\ll 0$	0.29	0.12	0.41	$\ll 0$	0.30	0.09	0.39
6	$\ll 0$	0.32	0.13	0.45	$\ll 0$	0.34	0.08	0.40	$\ll 0$	0.31	0.09	0.37
10	$\ll 0$	0.29	0.12	0.41	$\ll 0$	0.32	0.08	0.38	$\ll 0$	0.25	0.13	0.38
Panel B: Large firms												
1	$\ll 0$	0.38	-0.26	0.21	$\ll 0$	0.39	-0.29	0.18	$\ll 0$	0.51	-0.26	0.35
6	$\ll 0$	0.35	-0.23	0.19	$\ll 0$	0.58	-0.33	0.30	$\ll 0$	0.50	-0.21	0.33
10	$\ll 0$	0.38	-0.25	0.21	$\ll 0$	0.55	-0.32	0.27	$\ll 0$	0.30	-0.17	0.19
Panel C: Small firms												
1	$\ll 0$	0.28	0.15	0.40	$\ll 0$	0.26	0.16	0.39	$\ll 0$	0.21	0.13	0.32
6	$\ll 0$	0.32	0.18	0.47	$\ll 0$	0.22	0.13	0.33	$\ll 0$	0.22	0.14	0.32
10	$\ll 0$	0.28	0.16	0.40	$\ll 0$	0.21	0.14	0.32	$\ll 0$	0.26	0.16	0.39

Figure 1: Evolution of Pricing Error over Time (Large Firms)

This figures shows estimates of the average squared pricing error computed as $\frac{1}{N_t}\widehat{\mathbf{G}}_{\alpha,t-1}(\mathbf{x})'\widehat{\mathbf{y}}_t$ for large firm and $K = 1$, $K = 6$ and $K = 10$ for the full sample (blue dots). We also present a local regression smoothing curve as an estimate of the local average (black line). The red dashed horizontal line is at zero.

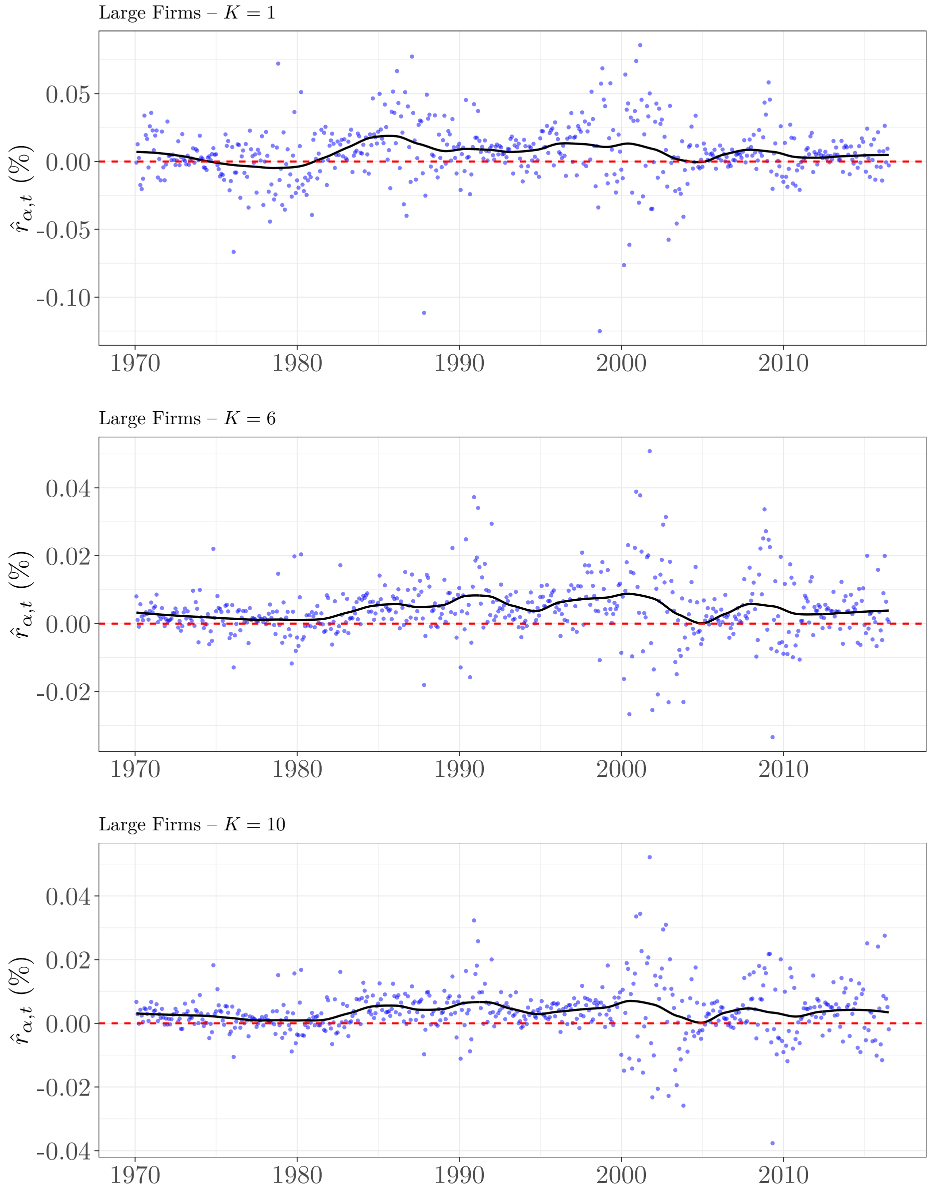


Table V: Firm Characteristics by Category

This table describes the characteristics used in our empirical analysis. They are the same as in Freyberger et al. (2020). Their online appendix details the construction of these characteristics. The sample period is January 1965 to December 2018.

	Past-returns:			**Value:**	
(1)	r_{2-1}	Return 1 month before prediction	(33)	A2ME	Total assets to Size
(2)	r_{6-2}	Return from 6 to 2 months before prediction	(34)	BEME	Book to market ratio
(3)	r_{12-2}	Return from 12 to 2 months before prediction	(35)	BEME$_{adj}$	BEME - mean BEME in Fama-French 48 industry
(4)	r_{12-7}	Return from 12 to 7 months before prediction	(36)	C	Cash to AT
(5)	r_{36-13}	Return from 36 to 13 months before prediction	(37)	C2D	Cash flow to total liabilities
			(38)	ΔSO	Log change in split-adjusted shares outstanding
	Investment:		(39)	Debt2P	Total debt to Size
(6)	Investment	% change in AT	(40)	E2P	Income before extraordinary items to Size
(7)	ΔCEQ	% change in BE	(41)	Free CF	Free cash flow to BE
(8)	ΔPI2A	Change in PP&E and inventory over lagged AT	(42)	LDP	Trailing 12-months dividends to price
(9)	ΔShrout	% change in shares outstanding	(43)	NOP	Net payouts to Size
(10)	IVC	Change in inventory over average AT	(44)	O2P	Operating payouts to market cap
(11)	NOA	Net-operating assets over lagged AT	(45)	Q	Tobin's Q
			(46)	S2P	Sales to price
	Profitability:		(47)	Sales_g	Sales growth
(12)	ATO	Sales to lagged net operating assets			
(13)	CTO	Sales to lagged total assets		**Trading frictions:**	
(14)	Δ(ΔGM-ΔSales)	Δ(% change in gross margin and % change in sales)	(48)	AT	Total assets
(15)	EPS	Earnings per share	(49)	Beta	Correlation × ratio of vols
(16)	IPM	Pre-tax income over sales	(50)	Beta daily	CAPM beta using daily returns
(17)	PCM	Sales minus costs of goods sold to sales	(51)	DTO	De-trended Turnover - market Turnover
(18)	PM	OI after depreciation over sales	(52)	Idio vol	Idio vol of Fama-French 3 factor model
(19)	PM_adj	Profit margin - mean PM in Fama-French 48 industry	(53)	LME	Price times shares outstanding
(20)	Prof	Gross profitability over BE	(54)	LME_adj	Size - mean size in Fama-French 48 industry
(21)	RNA	OI after depreciation to lagged net operating assets	(55)	Lturnover	Last month's volume to shares outstanding
(22)	ROA	Income before extraordinary items to lagged AT	(56)	Rel_to_high_price	Price to 52 week high price
(23)	ROC	Size + longterm debt - total assets to cash	(57)	Ret_max	Maximum daily return
(24)	ROE	Income before extraordinary items to lagged BE	(58)	Spread	Average daily bid-ask spread
(25)	ROIC	Return on invested capital	(59)	Std turnover	Standard deviation of daily turnover
(26)	S2C	Sales to cash	(60)	Std volume	Standard deviation of daily volume
(27)	SAT	Sales to total assets	(61)	SUV	Standard unexplained volume
(28)	SAT_adj	SAT - mean SAT in Fama-French 48 industry	(62)	Total vol	Standard deviation of daily returns
	Intangibles:				
(29)	AOA	Absolute value of operating accruals			
(30)	OL	Costs of goods solds + SG&A to total assets			
(31)	Tan	Tangibility			
(32)	OA	Operating accruals			