Structual Deep Learning in Conditional Asset Pricing

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

Preliminary, Please do NOT post or circulate

Abstract

We develop new nonparametric methodology for estimating conditional asset pricing models using deep neural networks, by employing time-varying conditional information on alphas and betas carried by firm-specific characteristics. Contrary to many applications of neural networks in economics, we can open the "black box" of machine learning predictions, 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 deep learning, followed by local PCAs to caputure time-varying features of the model. We formally establish the asymptotic theory of the 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

Keywords: sparse alternatives, thresholding, large covariance matrix estimation, Wald-test, screening, cross-sectional dependence, factor pricing model

*Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ 08544, USA. jqfan@princeton.edu. His research is supported by NSF grants DMS-1662139 and DMS-1712591.

†Department of Statistics, Harvard University, 75 Hamilton St., New Brunswick, NJ 08901, USA. yuan.liao@rutgers.edu

‡Department of Economics, Rutgers University, 75 Hamilton St., New Brunswick, NJ 08901, USA. yuan.liao@rutgers.edu

§Department of Finance, Washington University at St Louis, andreas.neuhierl@wustl.edu

1 Introduction

In this paper, we develop new nonparametric methodology to obtain structural, economic interpretations of predictions obtained from deep neural networks. 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 the most successful approaches to prediction for high dimensional and unstructured prediction problems. In the analysis of financial market data, while deep learning approaches have lead to increased explanatory power in predicting asset returns, they are often criticized as *black boxes*, 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 improved predictive power (mispricing v.s. risk prize), nor it is clear to us whether there have nontrivial portions of unpredictable noises inside the black box.

We take the success of deep learning methods as given, i.e. we do not aim to produce better predictions than the successful deep learning models in asset pricing. Our primary contribution is to open the black box and obtain economic understanding for *why* deep learning models have been shown to produce successful prediction in financial economics. 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 and that some of features 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 alpha 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 will arise naturally in a setting in which investors learn about predictors and take advantage of arbitrage opportunities to (eventually) eliminate them.

We also contribute to the econometric literature by rigorously deriving the rate of convergence for deep learning estimations of predicted returns, alphas and compensation for risk. A novel theoretical result is that out-of-sample rates of convergence is 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, as existing literature mostly concentrates on in-sample convergence. The derived rates of convergence depends on three key ingredients, (1) the approximation error for the unknown functions using 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, the third component also appears naturally when we apply 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 95% of the explained variation can be attributed to risk. Within this 95% the bulk of the explanatory power is driven by the factor realization ($\approx 80\%$) and roughly 20% by the long-term risk premium. About 5% 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: while the component of factor realization plays the major role inside the usual deep learning predictor, it has little predictive power. For out-of-sample predictions, we find the predictive success is driven almost exclusively by the risk premium component and the mispricing part. 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. This results has implications for the forecasting practice. The standard approach of taking the parameters of an estimated model and plugging in new data, will lead to a suboptimal prediction which is "noised up" by the past factor realization. Our results show that we can obtain better predictions, by focusing only on the risk premium and mispricing component.

Related Literature

Deep learning models have achieved remarkable success in science and engineering. Theoretically, deep learning have 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) and Schmidt-Hieber (2020) demonstrate the advantage of using deep neural network for being able to circumvent the curse of dimensionality arising from high-dimensional predictors in nonparametric regression. In asset pricing, deep learning methods have shown great promise. Freyberger et al. (2020), Gu et al. (2020), Bianchi et al. (2021) and Chen et al. (2020) show that equity and bond return predictions are increased significantly via the application of neural networks relative to linear (or other parametric) models.

There has been an extensive literature on factor pricing models, with both time-invariant and time-varying factor betas. In the most recent literature, Giglio and Xiu (2021); Giglio et al. (2021); Kim et al. (2021) have studied the unconditional model in the presence of latent factors, where they focused on inferences about risk premium and alphas from a large dimensional cross-sectional assets. Meanwhile, conditional linear factor models have been popularly used to capture time-varying effects of financial variables and firm-specific characteristics, e.g., (Shanken, 1990; Ferson and Harvey, 1999; Lettau and Ludvigson, 2001; Ghysels, 1998; Gagliardini et al., 2016). To account for dynamic factor betas, Kelly et al. (2019, 2020) proposed the "instrumental PCA" method (IPCA), which assumes the mappings from characteristics to alphas and betas are linear. Gu et al. (2019) accounts for more general nonlinear characteristics effects. Both IPC and autoencoder models assume that the mappings from the characteristics are time-invariant. Theoretical properties of IPCA have been recently studied by Kelly et al. (2020), and by (Bai, 2009; Chernozhukov et al., 2019) for similar problems in the panel data literature. See Gagliardinia et al. (2020) for an excellent survey for econometric methodologies for large-dimensional conditional factor models. Besides, there has been an extensive literature on econometrics and statistics for estimating latent factor models, including, e.g., Connor and Korajczyk (1986); Bai (2003); Stock and Watson (2002); Connor et al. (2012); Fan et al. (2016) among many others.

2 The Model

2.1 The conditional factor pricing model

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

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

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; $\boldsymbol{\lambda}_t$ is the vector of factor risk premium. u_{it} is the idiosyncratic component.

We consider the scenario where alphas and betas can be (partially) observed 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 non-parametric functions of characteristics; $\gamma_{\alpha,it}$ and $\boldsymbol{\gamma}_{\beta,it}$ represent the source of alphas and betas that cannot be explained by the characteristics.

This model extends the arbitrage model of Kim et al. (2021) and Li and Linton (2020) to conditional models where not only characteristic effects should be dynamic but their effects $g_{\alpha,t}$ and $g_{\beta,t}$ might be nonlinear and also time-varying. 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 fast over time due to the high-frequency change of $\gamma_{\alpha,i,t}$ and $\boldsymbol{\gamma}_{\beta,i,t}$.

Machine learning methods, in particular deep learning, has been successfully employed to predict asset returns using large amount of conditional information in the characteristics (e.g., Chen et al. (2020); Gu et al. (2020); Bali et al. (2021)). 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 data $\{(y_{i,t}, \mathbf{x}_{i,t-1}) : i = 1, ..., N\}$. 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 essentially 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 regression model:

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

where e_{it} is the regression noise.

However, little interpretations have been given regarding the source of predictability for such 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.

2.2 Structural machine learning predictions

To start, the mean function $m_t^0(\mathbf{x})$ can be presented as:

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

Let $\widehat{g}_{\alpha,t}$, $\widehat{r}_{\text{riskP},t}$ and $\widehat{g}_{\text{factor},t}$ respectively denote the estimated functions of $g_{\alpha,t}$, $g_{\text{riskP},t}$ and $g_{\text{factor},t}$, whose construction will be clear in the next section.

Then the conditional factor model yields the following in-sample decompositions at each time t:

In-sample decomposition:

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

$$\text{returns:} \quad y_{i,t} \approx \underbrace{\widehat{g}_{\alpha,t}(\mathbf{x}_{i,t-1}) + \widehat{r}_{\text{riskP},t}(\mathbf{x}_{i,t-1}) + \widehat{r}_{\text{factor},t}(\mathbf{x}_{i,t-1})}_{\widehat{y}_{i,t}} + e_{i,t}. \tag{2.5}$$

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 not depend on idiosyncratic errors (u_{it}). Later on, we will show that the spot return can be learned by period-by-period cross-sectional deep neural networks (DNN), via regressing returns on characteristics. Because $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)$ may be highly nonlinear in $\mathbf{x}_{i,t-1}$ and the number of characteristics can be high-dimensional, DNN is an appealing nonparametric machine learning technique to employ here.

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 (round time t) of the spot expected return. In particular, it decomposes the long-term expected return into the alpha and risk, both depend on characteristics. It clearly shows that the conditional expected return evolves with characteristics through two components. Our model enables to analytically characterize the percentage contributed from each component.

Moving on to the out-of-sample returns, the true out-of-sample return $y_{i,t+1}$ has the following 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 $g_{\alpha,t}$ and $g_{\beta,t}$ change slowly. 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, 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:

Out-of-sample decomposition:

$$\widehat{y}_{i,t+1|t} = \widehat{g}_{\alpha,t}(\mathbf{x}_{i,t}) + \widehat{r}_{\text{riskP},t}(\mathbf{x}_{i,t}) + \widehat{r}_{\text{factor},t}(\mathbf{x}_{i,t}) \tag{2.7}$$

$$y_{i,t+1} = \widehat{g}_{\alpha,t}(\mathbf{x}_{i,t}) + \widehat{r}_{\text{riskP},t}(\mathbf{x}_{i,t}) + \underbrace{r_{\text{factor},t+1}(\mathbf{x}_{i,t}) + e_{i,t+1}}_{\text{noises}} + o_P(1) \tag{2.8}$$

where the $o_P(1)$ term converges to zero when $N, T \to \infty$. Equation (2.7) is a decomposition of the neural network prediction, justifying the 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-prize component $\widehat{r}_{\text{riskP},t}(\mathbf{x}_{i,t})$. But the last term, which estimates $\widehat{g}_{\text{factor},t}(\mathbf{x}_{i,t}) \approx g_{\beta,t}(\mathbf{x}_{i,t})'(\mathbf{f}_t - \mathbb{E}\mathbf{f}_t)$, has no predictive power because $y_{i,t+1}$ depends on the future factor realization $\mathbf{f}_{t+1} - \mathbb{E}\mathbf{f}_{t+1}$.

Indeed, (2.8) classifies two prediction noises:

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

The future factor realization is often unpredictable, as it is often either independent of or very weakly dependent with $\widehat{g}_{\text{factor},t}(\mathbf{x}_{i,t})$. This leads to a major difference between the in-sample and out-of-sample decompositions. [1]

3 The Methodology

We now describe our methods for estimating the related quantities, including the mispricing components, factors, risk exposures and risk premia. Define

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

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

[1] If there is 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.

following equation: 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 "$\approx$" 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 of the matrix of demeaned spot expected returns. This motivates us to estimate the model by three major 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 the major steps for our methodology.

3.1 Applying deep neural networks

The first step of our method is to estimate the spot expected return $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1},\mathbf{f}_t)$. Let

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

which can be viewed as the conditional mean function in the following cross-sectional regression:

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

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:

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

where the parameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_J)$ with $\boldsymbol{\theta}_j \in \mathbb{R}^{d_j \times d_{j-1}}$, and $\sigma_j :\to \mathbb{R}^{d_j} \to \mathbb{R}^{d_{j+1}}$ is a vector-value nonlinear activation function. One of the popularly used activation function 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 $d_1 = ... = d_J = L$, 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

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

Let $\widehat{\mathbf{m}}_t(\mathbf{X}_{t-1})$ be the $N \times 1$ vector of $\widehat{m}_t(\mathbf{x}_{i,t-1})$. It is then our estimator for the N-dimensional spot expected return at given time t. As has been documented in the literature, learning using deep neural networks bring several advantages compared to classical nonparametric methods As an important statistical advantage (Bauer and Kohler, 2019; Schmidt-Hieber, 2020): first, least squares estimates based on multilayer feedforward neural networks are able to circumvent the curse of dimensionality arising from the high-dimensional predictors in nonparametric regressions. 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 in the machine learning literature that multilayer neural networks are able to approximate much larger classes of nonlinear functions.

3.2 Kernel smoothing

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

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

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 not motivated by the usual nonparametric regression for estimating conditional mean functions. Rather, it is motivated by the fact that the conditional alpha and beta $g_{\alpha,t}(\mathbf{x}_{i,t})$ and $g_{\beta,t}(\mathbf{x}_{i,t})$ may slowly vary 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 heuristically, $m_i\left(\frac{t}{T}\right) \approx m_i\left(\frac{s}{T}\right)$ and $g_i\left(\frac{t}{T}\right) \approx g_i\left(\frac{s}{T}\right)$ for all $\frac{s}{T} \approx \frac{t}{T}$. Thus we have

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

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 nonparametric kernel estimation literature. For instance, consider a time-varying mean-model $y_t = \mu_t + e_t$ with $\mathbb{E}(e_t) = 0$ and μ_t may vary over time. A standard practice to estimate μ_t via kernel smoothing is to use:

$$\widehat{\mu}_t = \frac{1}{Th}\sum_{s=1}^{T} y_s K\left(\frac{s-t}{Th}\right) A_t^{-1},$$

which essentially uses weighted averages of observations near time t. The estimation performance is not sensitive to the specific choice of kernel functions. In our empirical application, we use Epanechnikov kernel

$$K(u) = \frac{3}{4}(1 - u^2), \quad |u| \leq 1.$$

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.

In time-varying asset pricing models with observable factors, Ang and Kristensen (2012) applied a similar approach to estimate the unconditional alphas and betas. Contrary to 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 risk (and 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. In the presence of latent factors, the principal components analysis (PCA) is often used to combine with cross-sectional regressions for unconditional models (Giglio and Xiu, 2021; Giglio et al., 2021). But 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, PCA cannot capture the time-varying betas. More specifically, 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 $\boldsymbol{\gamma}_t$). Instead, 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 a *noise-free, no-arbitrarge* factor model locally around period t. Therefore, locally $\mathbf{G}_{\beta,t}(\mathbf{X}_{t-1})$ is approximately the top- eigenvector matrix of

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

Define

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

The demeaned expected return $\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1},\mathbf{f}_s)-\mathbb{E}(\mathbf{Y}_s|\mathbf{X}_{s-1})$ is estimated using $\widehat{\mathbf{m}}_s(\mathbf{X}_{s-1})-\bar{\mathbf{m}}_s$, we define the conditional-beta estimator $\widehat{\mathbf{G}}_{\beta,t}(\mathbf{X}_{t-1})$ as the eigenvectors corresponding to the first r 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 of our method 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 firm-specific characteristics, 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. To incorporate time-varying characteristics, a standard approach is to fix estimation windows, typically set to at least twelve months, and assumes characteristics are time-invariant

within a fixed window. This still cannot keep track of updated characteristics which in fact update in higher frequencies. For instance, some characteristics are averages of past returns, which change monthly other rely on the quarterly updated accounting information in Compustat and thus change every three months. In sharp contrasts, we employ kernel smoothing over time, which allows to update estimates of betas and alphas in month-by-month basis.

3.4 The full estimation algorithm

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

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

Algorithm 3.1. Estimate the model following these steps.

S1. Spontaneous expected returns. Run cross-sectional deep NN:

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

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.

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

S3. Beta and Factors. Define $\frac{1}{\sqrt{N}} \widehat{\mathbf{G}}_{\beta,t-1}$ as an $N \times r$ 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

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

and $\mathbf{K}_t$ is a $T \times T$ diagonal matrix consisting of $\{K(\frac{s-t}{Th}) : s = 1, ..., T\}$ 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 $g_{\alpha,t}$:

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

Steps 1 through 3 have been well motivated from our previous discussions. In addition, Step 4 estimates the alphas and the factor risk premium. While this step is similar to that in the usual Fama-Macbeth procedure, here we apply the cross-sectional regression on the average return after DNN projections.

Let $\widehat{g}_{\alpha,t-1,i}$ denote the i th 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. Also, $\widehat{\mathbf{g}}'_{\beta,t-1,i}$ denotes the i th row of $\widehat{\mathbf{G}}_{\beta,t-1}$. We define the in-sample risk and its estimator as

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

which can be estimated using

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

Next, we present the out-of-sample algorithm. For the out-of-sample prediction, we additionally train three neural networks by regressing the estimated in-sample alphas $\widehat{\mathbf{G}}_{\alpha,T-1}$, risks $\widehat{\mathbf{G}}_{\beta,T-1}\widehat{\boldsymbol{\lambda}}_T$ and $\widehat{\mathbf{G}}_{\beta,T-1}\widehat{\mathbf{f}}_T$ on $\mathbf{X}_{T-1}$. This gives rise to DNN estimated nonparametric functions: the α-function and risk-functions:

$$g_{\alpha,T}(\mathbf{x}), \quad g_{\text{riskPP},T}(\mathbf{x}) := g_{\beta,T}(\mathbf{x})'\boldsymbol{\lambda}_T, \quad g_{\text{factor},T}(\mathbf{x}) := g_{\beta,T}(\mathbf{x})'(\mathbf{f}_T - \mathbb{E}\mathbf{f}_T).$$

Note these functions are not required for in-sample estimation of alpha and risk but are required for the out-of-sample decomposition. 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 $\widehat{\mathbf{G}}_{\alpha,T-1}$ and $\widehat{\mathbf{G}}_{\beta,T-1}\widehat{\boldsymbol{\lambda}}_T$ as in Algorithm 3.1, and write elements of the

$N \times 1$ vectors as:

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

S6. Run cross-sectional deep NN regression:

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

S7. Run constraint cross-sectional deep NN regression: for some tuning parameter $\nu \to 0$,

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

S8. Using the new characteristic $\mathbf{x}_{i,T}$, predict the out-of-sample alphas and risks as:

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

It is worthwhile to emphasize that 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.

3.5 Double descent of the risk curve for overparametrized learning models

The success of deep learning revolution builds on a surprising empirical discovery that the best performing deep neural networks are trained with no explicit regularization to control their statistical complexity, and they produce 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. But as the number of parameters continues to grow, the risk starts decreasing again, so a second descent of the prediction risk occurs in the extremely overparametrized regime. Such a double-descent phenomena of DNN predictions was 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 machine learning models including random forests and kernel regressions, e.g., (Mei and Montanari, 2019), and even for linear models (Hastie et al., 2019; Belkin et al., 2020).

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 a return Y_{t+1} that is generated from a dynamic factor model:

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

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 assets $\mathbf{X}_t = (X_{1,t}, ..., X_{p,t})'$:

$$Y_{t+1} \quad = \quad \mathbf{X}_t'\boldsymbol{\theta} + e_t, \quad X_{j,t} = \boldsymbol{\beta}_j'\mathbf{f}_t + u_{j,t}. \tag{3.5}$$

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:

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

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 Lasso, this estimator does not reduce the dimensionality and 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 $t = 1, ..., T$ in this case. For the out-of-sample data $\{\mathbf{X}_t : t = T+1, ..., T+s\}$, 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 from real data of monthly 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 (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 in in fact the convergence point of gradient descent for least squares loss. The second interpretation is that as the number of regressors grows, more predictors/neurons generally result in decreasing components of $\boldsymbol{\theta}_p$, by distributing signals over more parameters, so the variance of the estimator

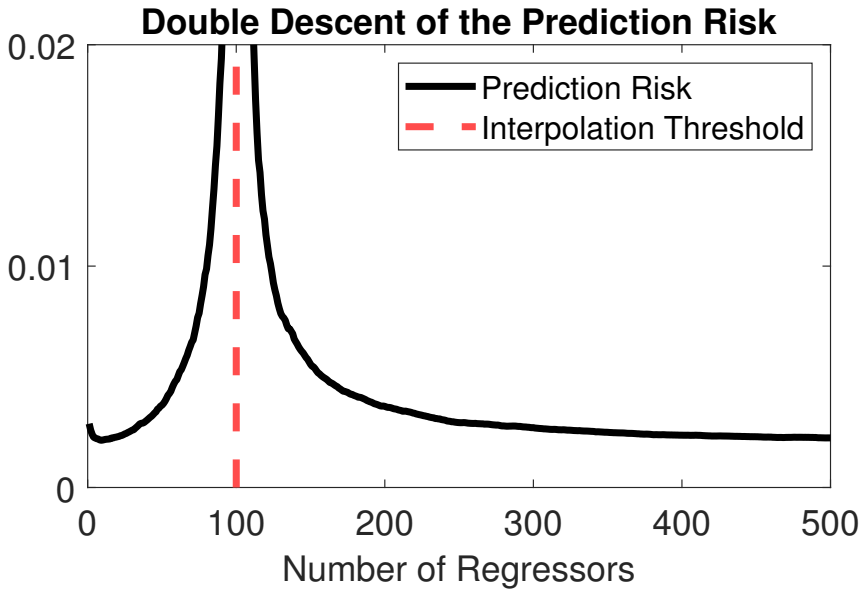


Figure 1: The prediction risk $R(p)$ for predicting the asset Y_{t+1} using lagged returns $(X_{1,t}, ..., X_{p,t})$ with the minimum-norm interpolation estimator, averaged over one hundred simulations. The number of in-sample fitted data is $T = 100$ and the number of out-of-sample predicted data is $s = 25$. 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.

decreases, which also leads to descending prediction risks. Above all, understanding the double descent phenomena from a theoretical perspective is still at the forefront stage in the machine learning literature.

4 The Asymptotic Theory

In this section we present the formal theory for the proposed procedure.

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 $\varepsilon_{it} = y_{it} - \mathbb{E}(y_{it}|\mathbf{x}_{it}, \mathbf{f}_t)$. Then $\{\varepsilon_{it}\}$ is cross-sectionally independent, con-

ditioning on $\mathbf{f}_t$. In addition, there are $c_1, c_2 > 0$, $\forall x > 0$, $\max_{it} \mathbb{P}(|\varepsilon_{it}| > x) \leq c_1 \exp(-c_2 x^2)$.

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

$$\max_k \text{var} \left(\frac{1}{Th} \sum_s \frac{(s-t)}{T} (v_{s,k} - \mathbb{E}v_{s,k}) K \left(\frac{t-s}{Th} \right) \right) = O\left(\frac{h^2}{Th} \right). \tag{4.1}$$

While condition (4.1) allows weak serial dependences for the factor process, in the appendex we verify this assumption when the factor process is independent over time.

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

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

where the functions are continuously twice-differentiable:

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

(ii) For out-of-sample predictions:

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

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 Ang and Kristensen (2012), 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 $L > 0$,

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

(ii) The dimension of the neural network space satisfies: $p(\mathcal{M}_{J,L}) \log^{3/2}(NT) = o(N)$.

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 $c > 0$,

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

In addition,

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

(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.

(iv) Out-of-sample alphas should be properly centered: $|\frac{1}{N}\sum_{i=1}^N g_{\alpha,T+1}(\mathbf{x}_{i,T})| = O_P(N^{-1/2})$.

Below we show that the theoretical results depend on three key quantities:

$$\varphi_T \;=\; \max_t \sup_{\mathbf{x}} \left[\inf_{\mathbf{g} \in \mathcal{M}_{J,L}} |g_{\beta,t}(\mathbf{x}) - g(\mathbf{x})| + \inf_{g \in \mathcal{M}_{J,L}} |g_{\alpha,t}(\mathbf{x}) - g(\mathbf{x})| \right].$$

$$\delta_T = \sqrt{\frac{p(\mathcal{M}_{J,L})\log(NT)}{N}},$$

$$\eta_T = \begin{cases} \frac{1}{\sqrt{Th}} + h^2 & \text{if } t \in (Th, T - Th) \\ \frac{1}{\sqrt{Th}} + h & \text{for all other } t. \end{cases}$$

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. For instance, Schmidt-Hieber (2020) showed that if the true regression function, say g_0 belongs is a composition of several functions, that is,

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

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:

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

where t_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.

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 $\{f(x, y) := \text{sgn}(h(x) - y) : h \in \mathcal{M}_{J,L}\}$. Bartlett et al. (2019) showed that for a ReLu network with depth L and the maximum width

J across layers,

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

In addition, the PCA step also depends on time-series rate η_T, which is the usual nonparametric rate for kernel smoothing. If the time of interest t is in the "interior" $(Th, T-Th)$, which is the focus of Ang and Kristensen (2012), then it converges relatively fast. Meanwhile we also allow the case when t is on the boundary (either $[1, Th]$ or $[T-Th, T]$), where the time-series rate of convergence is slower. The "boundary" case is also relevant in the context as we are also interested in out-of-sample forecasts.

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

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

(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 bias 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,*

$$\frac{1}{N}\sum_{i=1}^{N}[\widehat{g}_{\alpha,t-1,i} - g_{\alpha,t}(\mathbf{x}_{i,t-1})]^2 \;\; = \;\; O_P(\delta_T^2 + \varphi_T^2 + \eta_T^2),$$

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

The next theorem shows the prediction property of the alphas and risks. While the literature on deep neural networks mostly concentrates on in-sample convergence, to our best knowledge, this is the first time the out-of-sample prediction rate is established,

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

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

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

$$\begin{aligned}\max_{i\leq N}|\widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) - g_{\alpha,T+1}(\mathbf{x}_{i,T})| &= O_P\left((\delta_T + \varphi_T + \eta_T)^{s_0}\right)\\ \max_{i\leq N}|\widehat{g}_{riskP,T}(\mathbf{x}_{i,T}) - g_{riskP,T+1}(\mathbf{x}_{i,T})| &= O_P\left((\delta_T + \varphi_T + \eta_T)^{s_0}\right).\end{aligned}$$

As shown in Theorem 4.3, the prediction rate has an additional parameter s_0 compared to that of the in-sample result. This parameter slightly slows down the rate of convergence, as it depends on the smoothness of the functions (q,γ) and the dimensionality of characteristics.

The final result formally present the out-of-sample decomposition for the future returns. 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. Let $\mathcal{F}_T$ be 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. In addition, $\mathbb{E}(a|\mathcal{F}_T) = 0$, for $a \in \{\gamma_{\alpha,i,T}, \boldsymbol{\gamma}_{\beta,i,T}, \boldsymbol{\gamma}'_{\beta,i,T}\mathbf{f}_{T+1}, \mathbf{f}_{T+1} - \mathbb{E}\mathbf{f}_{T+1}\}$. Then there exists $\xi_{i,T+1}$ so that uniformly for $i \leq N$,*

$$y_{i,T+1} = \widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) + \widehat{g}_{riskP,T}(\mathbf{x}_{i,T}) + \xi_{i,T+1} + O_P\left((\delta_T + \varphi_T + \eta_T)^{s_0}\right),$$

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. 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 before we include it in the paper to avoid survivorship biases, which may arise from backfilling. Our overall sample ranges from 1965 through 2018. Table V 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.

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, $g_{\beta,t}(\mathbf{x})'\boldsymbol{\lambda}_t$, 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 at all. We therefore compute the R^2 from a regression of $y_{it} = \beta_0 + \beta_1 \widehat{y}_{it} + \varepsilon_{it}$ for each period, where $\widehat{y}_{it}$ is the fitted return by regressing returns on the characteristics at month t. As noted in equation (2.4), $\widehat{y}_{it}$ contains a component related to risk and possibly mispricing.

We next quantify their magnitudes. We first ask how much of the explained variation can be attributed to risk-related components, i.e. the average risk premium and the factor shock. To obtain an estimate of this quantity, we run the following regression each period:

$$y_{it} = \beta_0 + \beta_1 \widehat{g}_{\text{riskP},t,i} + \beta_2 \widehat{g}_{\text{factor},t,i} + \varepsilon_{it},$$

where $\widehat{g}_{\text{riskP},t,i} = \widehat{\mathbf{g}}'_{\beta,t-1,i}\widehat{\boldsymbol{\lambda}}_t$ and $\widehat{g}_{\text{factor},t,i} = \widehat{\mathbf{g}}'_{\beta,t-1,i}\widehat{\mathbf{f}}_t$ are our in-sample estimates of the relevant quantities. To obtain an interpretation of the economic magnitude of the mispricing component, we compute the returns to an arbitrage portfolio (Kim et al. (2021)) by computing

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

Since $\mathbf{y}_t$ are excess returns, any positive multiple of $\widehat{\mathbf{G}}_{\alpha,t-1}$ would also result in excess returns. We therefore scale the volatility of $r_{\alpha,t}$ to be 20% over the full sample to gain easy interpretability. We show the results in Table I for the full sample and early and late subsample separately. In addition, we cut by firm size, since the size distribution is highly skewed, we define the 20% largest firms as large firms and the 80% smallest as small.

Table I: In-Sample Decomposition - Realized Returns

This table shows empirical estimates for the in-sample decomposition of realized returns (equation (2.5)). R^2_{total} is time series average of the coefficient of determination of a regression of realized returns onto the average compensation for risk ($g_{\beta,t}(\mathbf{x})'\lambda_t$), the factor innovation ($g_{\beta,t}(\mathbf{x})'(f_t - \mathbb{E}f_t)$) and the mispricing estimate ($g_{\alpha,t}(\mathbf{x})$). R^2_{riskP} is the time series average of coefficient of determination of realized returns on the risk related components standardized by R^2_{total} in each period. $g'_{\alpha,t}y_t$ are the average monthly percentage returns to the arbitrage portfolio scaled to an average annual volatility of 20%. $\frac{g'_{\alpha,t}y_t}{\sigma(g'_{\alpha,t}y_t)}$ is the Sharpe ratio of the arbitrage portfolio. We report the results separately for all, large (20% largest) and small firms (all except 20% largest).

1970 - 2018				1970 - 1999				2000 - 2018			
R^2_{total}	R^2_{riskP}	$g'_{\alpha,t}y_t$	$\frac{g'_{\alpha,t}y_t}{\sigma(g'_{\alpha,t}y_t)}$	R^2_{total}	R^2_{riskP}	$g'_{\alpha,t}y_t$	$\frac{g'_{\alpha,t}y_t}{\sigma(g'_{\alpha,t}y_t)}$	R^2_{total}	R^2_{riskP}	$g'_{\alpha,t}y_t$	$\frac{g'_{\alpha,t}y_t}{\sigma(g'_{\alpha,t}y_t)}$
All Firms											
11.89	95.86	2.04	1.23	11.43	95.40	2.03	1.40	12.78	96.75	2.08	1.02
Large Firms											
16.00	95.38	1.92	0.78	14.84	94.89	1.80	0.91	18.28	96.32	2.15	0.66
Small Firms											
11.08	95.59	2.08	1.05	10.55	95.05	2.08	1.24	12.14	96.65	2.06	0.83

From the results in Table I we see that the bulk of realized returns is explained by risk related components. Throughout all subsamples and across firm sizes we see that more than 95% of the explained variation is due to exposure to systematic risk. Naturally, as the explained variation in returns is due to mispricing and the two risk related components, we can also conclude from Table I that about 5% of the explained variation is due to mispricing (1-R^2_{riskP}). While 5% may not appear to be large from a statistical point of view, it can be economically meaningful. To illustrate this point, we compute the return to the arbitrage portfolio. For the full sample, we find an average excess return of approximately 2 percent per month for this portfolio and an annual Sharpe ratio in excess of one. On average the Sharpe ratios are larger for the group of smaller firms and appear to be declining slightly over time.

In the previous analysis, we looked exclusively at the decomposition of realized returns. However, as can be seen from Table I, we can only explain a small fraction of the variation therein - the rest is noise. We therefore now also look at the decomposition of expected returns $\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)$. We already know that the bulk of the explained variation is due to risk (recall the R^2_{riskP} in Table I is normalized by the overall explanatory power). We now also break down, how much of the explained variation is due to the risk premium and the factor shock. Since the factor shock is only available in-sample, this will also inform about the differences between the in- vs. out-of-sample decomposition. To investigate the relative importance of the risk premium component and the factor related component, we normalize $\widehat{y}_{it}$, $\widehat{g}_{\text{riskP},t,i}$ and $\widehat{g}_{\text{factor},t,i}$ to have mean zero and unit standard deviation each period and then regress the normalized expected return onto the normalized risk premium estimates and normalized factor shocks. Due to the normalization, we can compare the magnitude of the estimated coefficients. The time series averages of these coefficient estimates are reported in Table II. The estimates in Table II show that the factor shock is much more important in explaining variation in expected returns. While there is some variation in the relative importance between small and large firms, all subsamples show a consistent picture. On average the factor shock explains at least five times as much of the variation in expected returns relative to the risk premium component.

Since the factor shock is only know contemporaneously with the return, we also compare the relative explanatory power of the risk premium component and the mispricing component for expected returns. To this end, we regress the expected return, $\widehat{y}_{it}$, on the risk premium component, $g_{\beta,t}(\mathbf{x}_{i,t-1})'\boldsymbol{\lambda}_t$, and the mispricing component,

$g_{\alpha,t}(\mathbf{x}_{i,t-1})$ separately. The results are consistent across time and firm size. We find an R^2 for the risk-premium component ranging between approximately 12% and 15% and of roughly 4% to 5.5% for the mispricing component.

Table II: In-Sample Decomposition - Risk Premia vs. Factor Shock

This table shows a breakdown of the relative importance of the risk based components of expected returns, the risk premium ($g_{\text{riskP},t,i}$) and the exposure to the factor shock ($g_{\text{factor},t,i}$). We present the time series averages of the following regression:

$$\widehat{y}_{it} = \beta_{\text{rp}}\widehat{g}_{\text{riskP},t,i} + \beta_{\text{factor}}\widehat{g}_{\text{factor},t,i} + \varepsilon_{it},$$

where all variables are normalized to have mean zero and unit standard deviation each period.

1970 - 2018		1970 - 1999		2000 - 2018	
$\beta_{\text{risk premium}}$	$\beta_{\text{factor shock}}$	$\beta_{\text{risk premium}}$	$\beta_{\text{factor shock}}$	$\beta_{\text{risk premium}}$	$\beta_{\text{factor shock}}$
All Firms					
0.172	0.973	0.180	0.968	0.157	0.984
Large Firms					
0.127	0.972	0.126	0.967	0.129	0.981
Small Firms					
0.170	0.973	0.175	0.968	0.161	0.983

We also present in-sample estimates of

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

Note that this quantity does not have a direct interpretation as an excess return to an arbitrage portfolio since $\widehat{\mathbf{y}}_t$ is not the return of a traded asset. We therefore do not apply any normalization to it and report the raw numbers. However, $\widehat{r}_\alpha$ can be interpreted as an estimate of the squared pricing error. Another interpretation is that it is a "de-noised" version of $r_\alpha := \frac{1}{N_t}\widehat{\mathbf{G}}'_{\alpha,t-1}\mathbf{y}_t$, because the idiosyncratic components have been removed due to the projected step. Figure 2 plots the evolution of $\widehat{r}_\alpha$ over time. From the plot we see an asymmetric "bell" shape. Overall, a large fraction of the pricing error comes from the sample of firms excluding the largest 20% of firms. The pricing error increases up until the early 2000s, and decays towards the end of the sample.

It is also evident from Figure 2 that the temporal evolution of alphas is not simply

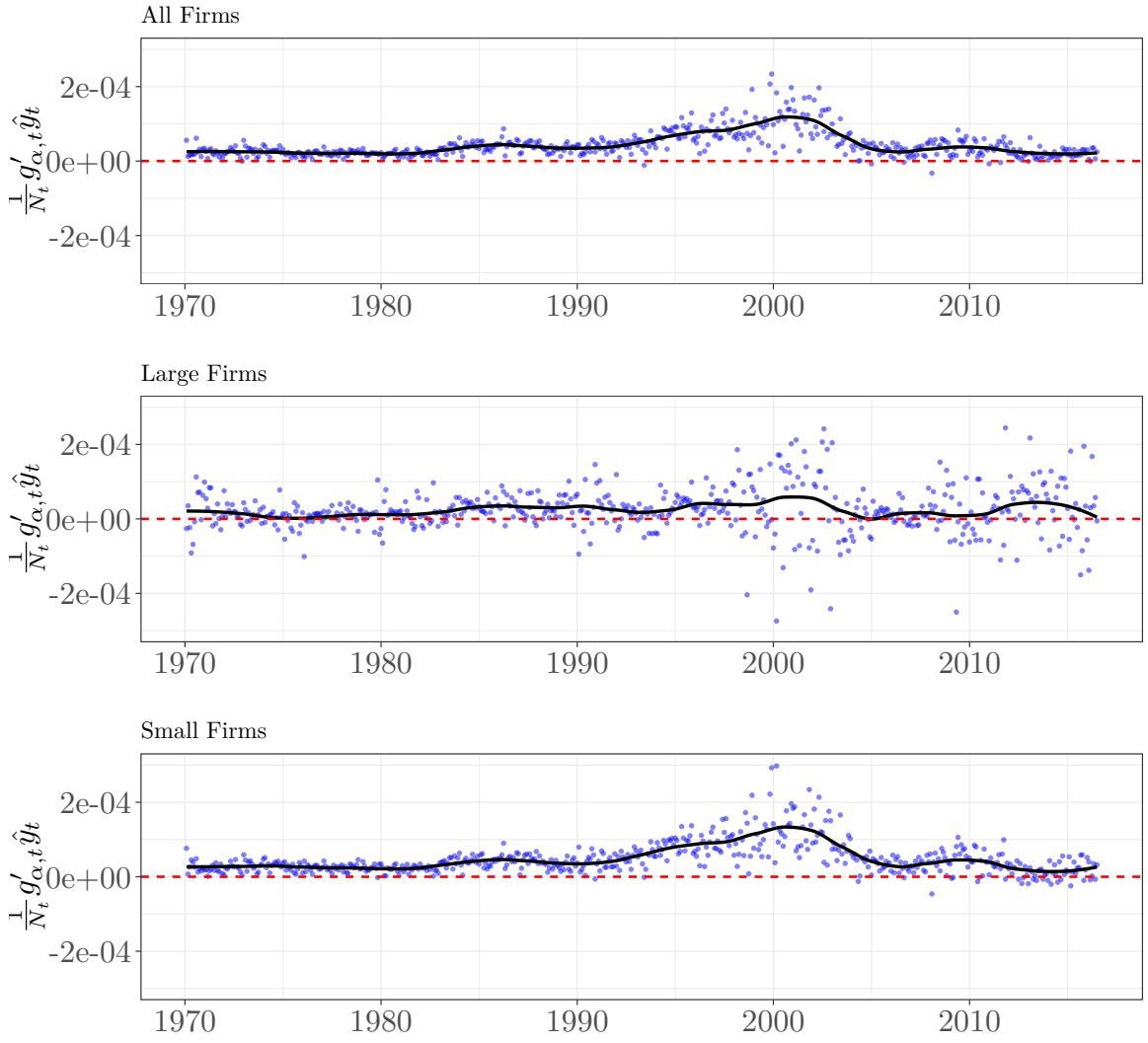
due to the evolution of characteristics, but that the alpha-function is also changing over time. In fact, since we are ranking characteristics at each period t, whose cross-sectional density is uniform on $[0,1]^{\dim(\mathbf{x})}$, as $N_t \to \infty$,

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

where $p(\mathbf{x})$ denotes the density of multivariate uniform distribution on $[0,1]^{\dim(\mathbf{x})}$. The displayed convergence shows that the evolution of $\widehat{r}_\alpha$ should be mainly driven by that of $g_{\alpha,t}$ rather than the characteristics.

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, large and small firms for the full sample (blue dots). We also estimate a local regression model as an estimate of the local average (black line). The red dashed horizontal line is at zero.



5.2.2 Out-of-Sample Decomposition

We now implement of 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 the most of successful models 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.

We know that the plug-in forecast decomposes into the following parts (see equation 2.8): The predicted mispricing component, $g_{\alpha,t}(\mathbf{x})$, the predicted risk premium $g_{\text{riskP},t}(\mathbf{x})$ and $g_{\text{factor},t}(\mathbf{x})$. We apply Algorithm 3.2 to obtain out-of-sample predictions of the individual quantities as well as the "aggregate" forecast for returns. From the decomposition, it is clear that the empirical success of DNN predictions may in principle be stemming from all three components. 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. For the factor component, it is important to remember that the factors are themselves excess returns. We can therefore draw on the large empirical literature analyzing the time series properties of returns. It is well known since Fama (1965) that returns exhibit very low temporal dependence in the time series. As predicting the factor component amounts to essentially predicting the time series of returns, we expect that this component is very hard to forecast.

In empirical analyses, a standard measure of performance is the R^2 from regressing realized returns on predicted returns. In Table III we present time series averages of the R^2 from regressing $\mathbf{y}_{t+1}$ on $\widehat{\mathbf{y}}_{t+1|t}$ and separately on the components of $\widehat{\mathbf{y}}_{t+1|t}$ at each period.

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

This table shows the predictive accuracy for out-of-sample forecasts. $R^2_{\hat{y}}$ is the average coefficient of determination for regressing realized returns on $\widehat{y}_{t+1|t}$, $R^2_{g_\alpha}$ is the average for regressing realized returns on $g_\alpha(\mathbf{x})$, $R^2_{g_\beta}$ is the average coefficient of determination for regressing realized returns on $g'_\beta \lambda_t$ and $R^2_{g_\beta, g_\alpha}$ is the average coefficient of determination for regressing realized returns on the predicted mispricing component and the predicted risk premium. All R^2 are presented in percentage.

1970 - 2018				1970 - 1999				2000 - 2018			
$R^2_{\hat{y}}$	$R^2_{g_\alpha}$	$R^2_{g_\beta}$	$R^2_{g_\beta,g_\alpha}$	$R^2_{\hat{y}}$	$R^2_{g_\alpha}$	$R^2_{g_\beta}$	$R^2_{g_\beta,g_\alpha}$	$R^2_{\hat{y}}$	$R^2_{g_\alpha}$	$R^2_{g_\beta}$	$R^2_{g_\beta,g_\alpha}$
All Firms											
1.619	0.826	1.801	2.376	1.520	0.620	1.627	2.027	1.778	1.154	2.079	2.931
Large Firms											
3.106	1.132	3.746	4.571	2.608	0.926	3.066	3.768	3.900	1.460	4.829	5.849
Small Firms											
1.485	0.743	1.557	2.087	1.386	0.525	1.355	1.708	1.643	1.091	1.880	2.690

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 $R^2_{a,b}$ shows the coefficient of determination from regressing realized returns on the predicted mispricing and risk premium component. Note that it does not have to equal the sum of R^2_a and R^2_b because the orthogonality only holds exactly in-sample. However, comparing this column with the first column ($R^2_{\hat{y}}$) 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 reason for this is that the plugin forecast also contains the 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 $\hat{\mathbf{y}}_t$ is about 7 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.

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 $\mathbf{x}_{i,t} = (x_{i,t,1}, ..., x_{i,t,5})$ as follows:

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

where $\epsilon_{x,i,t,k} \sim \mathcal{N}(0,1)$ and $\text{rank}(\bar{x}_{i,t,k})$ is the ranking of $\bar{x}_{i,t,k}$ in $(\bar{x}_{1,t,k}, ..., \bar{x}_{N,t,k})$. We take five factors, where the j th β-function ($j \leq 5$) $g_{\beta,t,j}$ are generated as follows:

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

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 $g_{\alpha,t}(\mathbf{x}) = [\phi_1(\mathbf{x}), ..., \phi_5(\mathbf{x})]\boldsymbol{\theta}_{\alpha,t}$, where $\boldsymbol{\theta}_{\alpha,t}$ is obtained by solving the following constraint least squares problem:

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

with $\widehat{a}_i$ being the estimated alpha for firm i using the Fama-French 5-factor model during 2001-2018. So the $g_{\alpha,t}(\cdot)$ function is time-varying. Furthermore, we generate $\boldsymbol{\gamma}_{\alpha,i,t-1} \sim \mathcal{N}(0, \sigma_{\gamma 1}^2)$ and $\boldsymbol{\gamma}_{\beta,i,t-1,j} \sim \mathcal{N}(0, \sigma_{\gamma 2,j}^2)$, 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 $f_{j,t} \sim \mathcal{N}(\mu_{f,j}, \sigma_{f,j}^2)$ and $u_{i,t} \sim \mathcal{N}(0, \sigma_{u,i}^2)$, with parameters generated using the Fama-French 5-factor model using data during 2001-2018. The factor risk premium is set to $\boldsymbol{\lambda}_t = a_t \boldsymbol{\mu}_f$, where we calibrate the constant a_t so that $g_{\alpha,t}(\mathbf{x}_{i,t-1})$ explains about 20% variations in the decomposition

$\mathbb{E}(y_{it}|\mathbf{x}_{i,t-1}) = g_{\alpha,t}(\mathbf{x}_{i,t-1}) + g_{\beta,t}(\mathbf{x}_{i,t-1})'\boldsymbol{\lambda}_t$ at each period. Throughout we fix $N = 500$ firms and $T = 200$ periods.

We examine the performance of estimating four quantities: spot expected return (1) $\mathbb{E}(y_{i,t}|\mathbf{x}_{i,t-1}, \mathbf{f}_t)$, (2) the long-term expected return $\mathbb{E}(y_{i,t}|\mathbf{x}_{i,t-1})$, (3) the alpha $g_{\alpha,t}(\mathbf{x}_{i,t-1})$ and the risk (4) $g_{\text{riskP},t}(\mathbf{x}_{i,t-1})$. For each estimated quantity $\widehat{r}_{i,t}$ for $r_{i,t}$ being one of the above four quantities, we report the in-sample relative mean squared error

$$\text{RMSE}(\widehat{r}) = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T}(\widehat{r}_{it} - r_{it})^2}{\sum_{i=1}^{N}\sum_{t=1}^{T} r_{it}^2}.$$

We compare the proposed method ("DNN-varying") with three additional benchmark methods. The first is linear varying method (called "Linear-varying") where the DNN projection is replaced by linear regressions on the characteristics period-by-period. The second benchmark is the DNN moving-window method ("DNN-mw") which estimates quantities at time t by fixing a moving-window of twelve months $[t-11, ..., t]$ as the in-sample period, and estimates quantities (2)(3)(4) 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.

In addition, we conduct out-of-sample comparisons, by refitting the estimated functions for quantities in (1)-(4) using the new data $x_{i,t}$ and compare them with the true values. We then compute

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

for new sampling periods $T+1, ..., T+s$ with $s = 180$, where $\widehat{r}_t(\cdot)$ is estimated using cross-sectional regressions on $\mathbf{x}_{i,t-1}$ within period t using one of the four compared methods, and then evaluated at $\mathbf{x}_{i,t}$ and compare with the true value $r_t(\mathbf{x}_{i,t})$. When $\widehat{r}_t(\mathbf{x}_{i,t}) = \widehat{y}_{i,t+1|t}$ is the predicted spot return, we set $r_t(\mathbf{x}_{i,t}) = y_{i,t+1}$, so RMSE$(\widehat{r})$ in this case is the relative predictive error of the true returns. For other $\widehat{r}_t(\cdot)$ that is the estimated function $r_t(\mathbf{x})$ of one of $\{\mathbb{E}(y_{i,t}|\mathbf{x}), g_{\alpha,t}(\mathbf{x}), g_{\text{riskP},t}(\mathbf{x})\}$, we plug in the out-of-sample $\mathbf{x}_{i,t}$ and compute the corresponding RMSE as well.

Table IV reports the RMSE(r) 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 all four quantities, closely followed by the Linear-varying method, which accounts for the time-varying characteristics but not the nonlinearity. For in-sample estimation, the gain of using DNN over linear projections is more pronounced in estimating the spot expected returns. The two moving-window methods, DNN-mw and Linear-mw, use the same estimators of the period-by-period as spot expected returns (which is why they have the same RMSE as for the first two methods in the first column). However, the moving-window methods do not capture sufficient dynamics as the first two methods do. For predicting the out-of-sample returns, the race is very close. But in general the DNN-varying method still performs better than competing ones. Finally, observe that for estimating $\mathbb{E}(y|\mathbf{x}, \mathbf{f})$, the out-of-sample prediction error is significantly larger than that of the in-sample error. This is not surprising because the out-of-sample RMSE($\widehat{r}$) in this case is relative to the true return $y_{i,t+1}$ rather than the expected return $\mathbb{E}(y_{i,t+1}|\mathbf{x}_{i,t}, \mathbf{f}_{t+1})$, so is also affected by out-of-sample idiosyncratic errors.

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

This table reports in-sample and out-of-sample RMSE($\widehat{r}$) 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 twelve month moving window, and the linear moving-window method (Linear-mw) which replaces the DNN projection in DNN-mw with linear projections. We compute the RMSE for estimating four quantities and report results averaged over 100 Monte Carlo repetitions.

	spot $\mathbb{E}(y\|\mathbf{x},\mathbf{f})$	long-term $\mathbb{E}(y\|\mathbf{x})$	alpha $g_{\alpha,t}(\mathbf{x})$	risk $g_{\text{riskP},t}(\mathbf{x})$
		In-sample		
DNN-varying	0.008	0.290	0.501	0.384
Linear-varying	0.058	0.351	0.563	0.403
DNN-mw	0.008	0.619	0.659	0.749
Linear-mw	0.058	0.651	0.654	0.768
		Out-of-sample		
DNN-varying	0.654	0.211	0.334	0.314
Linear-varying	0.641	0.268	0.361	0.325
DNN-mw	0.654	0.404	0.256	0.532
Linear-mw	0.641	0.466	0.388	0.564

A Technical proofs

A.1 Proof of Theorem 4.1

Let $\boldsymbol{\Delta}_t := \widehat{\mathbf{m}}_t - \mathbb{E}(\mathbf{y}_t|\mathbf{X}_{t-1}, \mathbf{f}_t)$, 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 $\boldsymbol{\Delta}_t$. We shall obtain the rate of convergence for $\|\boldsymbol{\Delta}_t\|$. Let

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

A.1.1 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 $\{\text{sgn}(h(x) - y) : h \in \mathcal{M}_{J,L}\}$.

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:*

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

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

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

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

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

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

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 $k = 0, 1, 2, ...$

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

Also let $E_t(f) := \frac{1}{N}\sum_{i=1} f(\varepsilon_{it}, \mathbf{x}_{i,t-1}) - \mathbb{E}f(\varepsilon_{it}, \mathbf{x}_{i,t-1})$ 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.1) imply

$$\sup_{f\in\mathcal{C}_{kt}} E_t(f) = \sup_{m\in\mathcal{E}_{kt}} [Q_{T,t}(m_t^0) - \mathbb{E}Q_{T,t}(m_t^0)] - [Q_{T,t}(m) - \mathbb{E}Q_{T,t}(m)]$$

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

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

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

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 $\max_{it}|\varepsilon_{it}| \leq B_T$,

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

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

$$
\max_t \sup_{d_t(m_t^0, m)\leq\delta} \sup_{\mathbf{x}} |m_t^0(\mathbf{x}) - m(\mathbf{x})| \leq C\delta^{s_0}, \qquad \text{(A.4)}
$$

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 $m_1, ..., m_N$ 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 $f \in \mathcal{C}_{kt}$, 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 $f_j(\varepsilon, \mathbf{x}) = (\varepsilon + m_t^0(\mathbf{x}) - m_j(\mathbf{x}))^2 - \varepsilon^2$.

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

Hence $f \in [l_j, u_j]$, where $l_j = f_j - b\delta$ and $u_j = f_j + b\delta$. Moreover, $\mathbb{E}(u_j - l_j)^2 \leq C\delta^2 \mathbb{E}\varepsilon_{it}^2$. 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 $p(\mathcal{M}_{J,L}) = o(N)$,

$$\log \mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2}) \leq D(1 + \log \frac{1}{\delta}). \tag{A.5}$$

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

$$\begin{aligned} & 2^{12} \int_{M_k/64}^{\sigma_k \sqrt{T_k}} \sqrt{\log \mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2})} d\delta \leq 2^{12} \sqrt{D} \int_{(2^k \epsilon_T)^2/c}^{\sqrt{B_T}(2^k \epsilon_T)^{s_0/2+1}} \sqrt{1 + c_0^{-1} \log \delta^{-c_0}} d\delta \\ \leq \; & C\sqrt{1 + \log(2^k \epsilon_T)^{-2c_0}} \sqrt{DB_T} (2^k \epsilon_T)^{s_0/2+1} \leq C\sqrt{DB_T} (2^k \epsilon_T)^{s_0/2+1-c_0} \\ \leq \; & \sqrt{B_T p(\mathcal{M}_{J,L}) \log N} (2^k \epsilon_T)^{s_0/2+1-c_0} \leq M_k \sqrt{N} \end{aligned} \tag{A.6}$$

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), by the union bound,

$$T\mathbb{P}\left(\sup_{f \in \mathcal{C}_{kt}} E_t(f) \geq M_k, \max_{it} |\varepsilon_{it}| \leq B_T\right) \leq CT \exp\left(-\frac{CNM_k^2}{\sigma_k^2(1 + cT_k)}\right)$$

$$\leq CT\exp\left(-\frac{CN(2^k\epsilon_T)^2}{(1+B_T(2^k\epsilon_T)^{s_0})}\right) \leq CT\exp\left(-\frac{CN(2^k)^{2-s_0}\epsilon_T^2}{\epsilon_T^{s_0}B_T}\right)$$
$$\leq CT\exp\left(-CN(2^k)^{2-s_0}\epsilon_T^{2-s_0}B_T^{-1}\right).$$

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

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

step 4 bounding A_2.

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

$$\sup_{f\in\mathcal{C}_{kt}} E_t(f) \leq |\frac{1}{N}\sum_{i=1}(m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))^2 - \mathbb{E}(m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))^2|$$
$$+|\frac{1}{N}\sum_{i=1}(m(\mathbf{x}_{i,t-1}) - m_t^0(\mathbf{x}_{i,t-1}))\varepsilon_{it}| \leq (C + \max_{it}|\varepsilon_{it}|)\|m - m_t^0\|_\infty$$
$$\leq C\max_{it}|\varepsilon_{it}|(2^k\epsilon_T)^{s_0}.$$
$$A_2 = T\sum_{k=0}^{\infty}\mathbb{P}\left(\sup_{f\in\mathcal{C}_{kt}} E_t(f) \geq (2^{k-2}\epsilon_T)^2/2, \max_{it}|\varepsilon_{it}| > B_T\right)$$
$$\leq T\sum_{k=0}^{\infty}\mathbb{P}\left(C\max_{it}|\varepsilon_{it}|(2^k\epsilon_T)^{s_0} \geq (2^{k-2}\epsilon_T)^2, \max_{it}|\varepsilon_{it}| > B_T\right)$$
$$\leq T\sum_{k=0}^{\infty}\mathbb{P}\left(\max_{it}|\varepsilon_{it}|1\{\max_{it}|\varepsilon_{it}| > B_T\} \geq c(2^{k-2}\epsilon_T)^{2-s_0}\right)$$
$$\leq \sum_{k=0}^{\infty}\frac{T}{2^{(k-2)(2-s_0)/2}}(\epsilon_T)^{-(2-s_0)/2}\mathbb{E}\max_{it}|\varepsilon_{it}|^{1/2}1\{\max_{it}|\varepsilon_{it}| > B_T\}$$
$$\leq TC(\epsilon_T)^{-(2-s_0)/2}\sqrt{\mathbb{E}\max_{it}|\varepsilon_{it}|\mathbb{P}(\max_{it}|\varepsilon_{it}| > B_T)}$$
$$\leq TC(\epsilon_T)^{-(2-s_0)/2}\log^{1/4}(NT)\sqrt{NT}\exp(-CB_T^2) \leq (NT)^c\exp(-CB_T^2)$$
$$\leq \exp(c\log(NT) - CB_T^2) \to 0$$

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

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

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

(ii) By Lemma A.1, $\mathbb{P}(\forall t, \widehat{m}_t \in \mathcal{H}(q, \gamma, L)) \to 1$ for any $L > 0$. Then by Lemma 2 of Chen and Shen (1998),

$$\max_t \sup_{\mathbf{x}} |\widehat{m}_t(\mathbf{x}) - m_t^0(\mathbf{x})| \le 2(2L)^{1-s_0} \max_t d_t(m_t^0, \widehat{m}_t)^{s_0} \le C\epsilon_T^{s_0}.$$

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

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

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

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

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

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

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 $M/64$.

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

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

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

Hence $g = O_P(\epsilon_T^2)$.

□

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

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

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

Then

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

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

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

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

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

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

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 $\max_{it} |\varepsilon_{it}| \leq B_T$,

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

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 $m_1, ..., m_N$ 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 $f_j(\varepsilon, \mathbf{x}) = \varepsilon m_j(\mathbf{x})$.

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

Hence $f \in [l_j, u_j]$, where $l_j = f_j - |\varepsilon|\delta$ and $u_j = f_j + |\varepsilon|\delta$. Moreover, $\mathbb{E}(u_j - l_j)^2 \leq 4\delta^2\mathbb{E}\varepsilon_{it}^2$. 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.5), $p(\mathcal{M}_{J,L}) \log \frac{CN}{\delta p(\mathcal{M}_{J,L})} \leq D + D\delta^{-1}$. Therefore when $\sqrt{\log(NT)}p(\mathcal{M}_{J,L}) \log N = o(N)$, we have $DB_T = o(N)$,

$$\begin{aligned} & 2^{12} \int_0^{\sqrt{B_T}} \sqrt{\log \mathcal{N}_{[]}(\delta, \mathcal{C}_{kt}, \|.\|_{L^2})} d\delta \leq 2^{12}\sqrt{D} \int_0^{\sqrt{B_T}} \sqrt{1 + \delta^{-1}} d\delta \\ \leq \; & 2^{12}\sqrt{D} \int_0^1 \sqrt{2\delta^{-1}} d\delta + 2^{12}\sqrt{D} \int_1^{\sqrt{B_T}} \sqrt{2} d\delta \leq \sqrt{CDB_T} \leq 0.5^{3/2} M\sqrt{N}. \end{aligned}$$

This verifies (a.2) in Lemma 1 of Chen and Shen (1998). Hence

$$\begin{aligned} E_1 \; &\leq \; \sum_t \mathbb{P}\left(A_t > M, \max_{it} |\varepsilon_{it}| \leq B_T\right) \leq CT \exp\left(-\frac{CNM^2}{(1 + cB_T)}\right) \\ &\leq \; C \exp\left(\log T - \frac{CNM^2}{B_T}\right) \to 0, \text{ if } \sqrt{\log(NT)} \log T = o(N). \end{aligned}$$

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

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

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

(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

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

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

A.1.2 Convergence of $\bar{m}_{i,t} - \mathbb{E}(y_{it}|\mathbf{x}_{i,t-1})$

We recall and introduce the following notation.

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

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)* $\text{var}(\frac{1}{Th}\sum_{s=1}^{T} g_{\beta,s}(\mathbf{x}_{i,s-1})'\mathbf{f}_s K\left(\frac{t-s}{Th}\right)) = O(1/(Th))$.
(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,

$$\frac{1}{N}\sum_i \mathbb{E}\left|\bar{m}_{i,t} - m_i\left(t/T\right)\right|^2 = O\left(\frac{1}{Th} + p_T^2 + \delta_T^2 + \varphi_T^2\right), \quad p_T^2 = \begin{cases} h^4, & t \in (Th, T-Th) \\ h^2, & \textit{for all other } t. \end{cases}$$

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

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

We have $\mathbb{E}(\mathbf{f}_s|\mathbf{x}_s) = \mathbb{E}\mathbf{f}_s$ implying $\mathbb{E}a_2 = 0$. Also, $\max_{it}\text{var}(a_2) = O(1/(Th))$. 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,

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

To bound a_{11}, write $\delta(x) = \frac{1}{Th}$, $l = (1-t)/(Th)$ and $u = (T-t)/(Th)$. Then

$$a_{11} = \dot{m}_i(\frac{t}{T})A_t^{-1}h\sum_{x=l}^{u} xK(x)\delta(x) = \dot{m}_i(\frac{t}{T})A_t^{-1}h\left[\int_l^u xK(x)dx + O(\frac{1}{Th})\right]$$

Case 1: $t \in (Th, T-Th)$. Then $l \le -1$ and $u \ge 1$. Note $\int_{-1}^{1} xK(x)dx = 0$. So

$$\max_i|a_{11}| = \max_i|\dot{m}_i(\frac{t}{T})|A_t^{-1}hO(\frac{1}{Th}) = O(T^{-1}).$$

Case 2: $t \in (0, Th]$. We have $\max_i|a_{11}| = \max_i|\dot{m}_i(\frac{t}{T})|A_t^{-1}hO(1) = O(h)$.

Case 3: $t \in [T - Th, T]$. This case is very similar to Case 2, $\max_i |a_{11}| = O_P(h)$. Together,

$$\frac{1}{N}\sum_i \mathbb{E}\left|\bar{m}^0_{i,t} - m_i\left(t/T\right)\right|^2 = O\left(\frac{1}{Th} + p_T^2\right).$$

In addition, by Proposition A.1, write $\Delta_{is} := m^0_s(\mathbf{x}_{i,s-1}) - \widehat{m}_s(\mathbf{x}_{i,s-1})$.

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

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

A.2 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 $g_{\beta,t-1}$. 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 $\mathbf{H}_t := \frac{1}{NTh}\sum_s \mathbf{f}_s\mathbf{f}_s' K(s,t) g_{\beta,t-1}'\widehat{\mathbf{G}}_{\beta,t-1}\widehat{\mathbf{V}}^{-1}$, we have

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

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

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

the following identity,

$$\begin{aligned}\widehat{\boldsymbol{\lambda}}_t - \mathbf{H}_t^{-1}\boldsymbol{\lambda}_t &= \frac{1}{N}\sum_{i=1}^{N}\widehat{g}_{\beta,t-1,i}(\bar{m}_{i,t} - m_i(t/T)) + \frac{1}{N}\sum_{i=1}^{N}(\widehat{g}_{\beta,t-1,i} - \mathbf{H}_t' g_{\beta,t}(\mathbf{x}_{i,t-1}))g_{\alpha,t}(\mathbf{x}_{i,t-1})' \\ &\quad + \frac{1}{N}\sum_{i=1}^{N}\widehat{g}_{\beta,t-1,i}(g_{\beta,t}(\mathbf{x}_{i,t-1})'\mathbf{H}_t - \widehat{g}'_{\beta,t-1,i})\mathbf{H}_t^{-1}\boldsymbol{\lambda}_t + O_P(N^{-1/2}),\end{aligned}$$

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 $\|\widehat{\boldsymbol{\lambda}}_t - \mathbf{H}_t^{-1}\boldsymbol{\lambda}_t\| = O_P(\frac{1}{\sqrt{Th}} + p_T + \delta_T + \varphi_T)$. So

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

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

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

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

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)* $\mathbb{E}\mathbf{f}_t\mathbf{f}_t'$ *does not vary across* t.

(ii) $\max_{kl}\text{var}\left(\frac{1}{Th}\sum_s \frac{(s-t)}{T}(v_{s,k} - \mathbb{E}v_{s,k})K(s,t)\right) = O(\frac{h^2}{Th})$ *for* $\mathbf{v}_s \in \{\mathbf{f}_s, \text{vec}(\mathbf{f}_s\mathbf{f}_s')\}$

Then for each fixed t,

$$\frac{1}{Th}\sum_s \mathbf{M}_s\mathbf{M}_s' K(s,t) = g_{\beta,t-1}\frac{1}{Th}\sum_s \mathbf{f}_s\mathbf{f}_s' K(s,t) g_{\beta,t-1}' + \mathbf{B}(t)$$

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

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

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),

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

Fix t of interest. By Propositions A.1, A.2,

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

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

$$\begin{aligned}
&\frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s d_2(i,s)\mathbf{f}_s K(s,t)\|^2 \le \frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s (g_{\beta,s}(\mathbf{x}_{is}) - g_{\beta,t-1}(\mathbf{x}_{it}))'\mathbf{f}_s\mathbf{f}_s' K(s,t)\|^2 \\
\le\ & \frac{2}{N}\sum_i \|\frac{1}{Th}\sum_s \frac{s-t}{T}\dot{\mathbf{g}}_i\left(\frac{t}{T}\right)'\mathbf{f}_s\mathbf{f}_s' K(s,t)\|^2 + \frac{2}{N}\sum_i \|\frac{1}{Th}\sum_s \frac{(s-t)^2}{T^2}\ddot{\mathbf{g}}_i\left(\frac{v}{T}\right)'\mathbf{f}_s\mathbf{f}_s' K(s,t)\|^2
\end{aligned}$$

$$
\begin{aligned}
&\leq C\|\frac{1}{Th}\sum_s \frac{(s-t)}{T}\mathbf{f}_s\mathbf{f}_s' K(s,t)\|^2 + C\left(\frac{1}{Th}\sum_s \frac{(s-t)^2}{T^2}\|\mathbf{f}_s\|^2 K(s,t)\right)^2 \\
&\leq C\|\frac{1}{Th}\sum_s \frac{(s-t)}{T}\mathbf{s}_s K(s,t)\|^2 + C\|\frac{1}{Th}\sum_s \frac{(s-t)}{T}\mathbb{E}\mathbf{f}_s\mathbf{f}_s' K(s,t)\|^2 + O_P(h^4) \\
&\leq O_P(1)\max_{kl}\text{var}\left(\frac{1}{Th}\sum_s \frac{(s-t)}{T}\mathbf{s}_{s,kl}K(s,t)\right) + CA_t^{-2}h^2\left[\int_l^u xK(x)dx + O_P(\frac{1}{Th})\right]^2 + O_P(h^4) \\
&\leq O_P(\frac{h^2}{Th} + h^2k_T^2 + h^4)
\end{aligned}
$$

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,

$$
\begin{aligned}
&\frac{1}{NTh}\sum_{is} d_2(i,s)^2 K(s,t) = \frac{1}{NTh}\sum_{is}\|g_{\beta,s}(\mathbf{x}_{is}) - g_{\beta,t-1}(\mathbf{x}_{it})\|^2\|\mathbf{f}_s\|^2 K(s,t) \\
\leq\ &\frac{1}{NTh}\sum_{is}\|\dot{\mathbf{g}}_i(v)\|^2\frac{(s-t)^2}{T^2}\|\mathbf{f}_s\|^2 K(s,t) \leq O_P(1)\frac{1}{Th}\sum_s \frac{(s-t)^2}{T^2}\mathbb{E}\|\mathbf{f}_s\|^2 K(s,t) = O_P(h^2).
\end{aligned}
$$

Finally,

$$
\begin{aligned}
&\frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s d_3(i,s)\mathbf{f}_s K(s,t)\|^2 \leq \frac{1}{N}\sum_i \|\frac{1}{Th}\sum_s [m_i(s/T) - m_i(t/T)]\mathbf{f}_s K(s,t)\|^2 \\
\leq\ &\frac{C}{N}\sum_i \dot{m}_i(t/T)^2\|\frac{1}{Th}\sum_s \frac{s-t}{T}\mathbf{f}_s K(s,t)\|^2 + \frac{C}{N}\sum_i \|\frac{1}{Th}\sum_s \ddot{m}_i(v)\frac{(s-t)^2}{T^2}\mathbf{f}_s K(s,t)\|^2 \\
\leq\ &O_P(1)\max_k \text{var}\left(\frac{1}{Th}\sum_s \frac{(s-t)}{T}f_{s,k}K(s,t)\right) + O_P(h^4) \leq O_P(\frac{h^2}{Th} + h^4). \\
&\frac{1}{NTh}\sum_{is} d_3(i,s)^2 K(s,t) \leq \frac{1}{NTh}\sum_{is}[m_i(s/T) - m_i(t/T)]^2 K(s,t) = O_P(h^2).
\end{aligned}
$$

Putting together,

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

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 $g_{\beta,t-1}$ denote the $N \times K$

matrix of $g_{\beta,t-1}(\mathbf{x}_{it})$. Then $\mathbf{M}_s = \mathbf{Z}_s + g_{\beta,t-1}\mathbf{f}_s$.

$$\begin{aligned}\frac{1}{Th}\sum_s \mathbf{M}_s\mathbf{M}_s' K(s,t) &= g_{\beta,t-1}\frac{1}{Th}\sum_s \mathbf{f}_s\mathbf{f}_s' K(s,t) g_{\beta,t-1}' + \mathbf{b}_1 + \mathbf{b}_2 + \mathbf{b}_2' \\ \mathbf{b}_1 &= \frac{1}{Th}\sum_s \mathbf{Z}_s\mathbf{Z}_s' K(s,t), \quad \mathbf{b}_2 = \frac{1}{Th}\sum_s \mathbf{Z}_s\mathbf{f}_s' K(s,t) g_{\beta,t-1}'.\end{aligned}$$

Then

$$\begin{aligned}\frac{1}{N^2}\|\mathbf{b}_1\|_F^2 &\le \frac{1}{N^2}[\frac{1}{Th}\sum_s \|\mathbf{Z}_s\|^2 K(s,t)]^2 = [\frac{1}{NTh}\sum_{is} z_{is}(t)^2 K(s,t)]^2 \\ &= O_P\left(\delta_T^4 + \varphi_T^4 + \frac{1}{(Th)^2} + h^4\right) \\ \frac{1}{N^2}\|\mathbf{b}_2\|_F^2 &\le O_P(1)\frac{1}{N}\|\frac{1}{Th}\sum_s \mathbf{Z}_s\mathbf{f}_s' K(s,t)\|_F^2 = O_P(1)\frac{1}{N}\sum_{i=1}^N \|\frac{1}{Th}\sum_s z_{is}(t)\mathbf{f}_s K(s,t)\|^2 \\ &= O_P\left(\delta_T^2 + \varphi_T^2 + \frac{1}{(Th)^2} + p_T^2\right).\end{aligned}$$

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

□

A.3 Proof of Theorem 4.3

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

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

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

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. Show the feasibility of $\pi_N g_{\alpha,T+1}(\cdot)$

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

$$
\begin{aligned}
a_3 &= \|\frac{1}{N}\sum_i g_{\alpha,T+1}(\mathbf{x}_{i,T})\| = O_P(N^{-1/2}) \\
a_4 &= \|\frac{1}{N}\sum_i (g_{\alpha,T+1}(\mathbf{x}_{i,T}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))g_{\beta,T}(\mathbf{x}_{i,T-1})\| \\
&\leq O_P(\frac{1}{N}\sum_i (g_{\alpha,T+1}(\mathbf{x}_{i,T}) - g_{\alpha,T}(\mathbf{x}_{i,T-1}))^2)^{1/2} \\
a_5 &= \|\frac{1}{N}\sum_i g_{\alpha,T}(\mathbf{x}_{i,T-1})g_{\beta,T}(\mathbf{x}_{i,T-1})\| = O_P(N^{-1/2}).
\end{aligned}
$$

Hence $\pi_N g_{\alpha,T+1}(\cdot)$ satisfies the constraint.

Step 2. In-sample mean squared error. Because $\pi_N g_{\alpha,T+1}(\cdot) \in$ DNN satisfies the constraint,

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

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 3. 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

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

$$\mathcal{C}_k \;:=\; \{f : f(\mathbf{x}) = -(m(\mathbf{x}) - g(\mathbf{x}))^2 : m \in \mathcal{E}_k\},$$

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

$$\begin{aligned}
&\sup_{f\in\mathcal{C}_k} \frac{1}{N}\sum_i f(\mathbf{x}_{i,t-1}) - \mathbb{E}f(\mathbf{x}_{i,t-1}) \\
=\;& \sup_{m\in\mathcal{E}_k} -\frac{1}{N}\sum_i (m(\mathbf{x}_{i,t-1}) - g(\mathbf{x}_{i,t-1}))^2 + \mathbb{E}(m(\mathbf{x}_{i,t-1}) - g(\mathbf{x}_{i,t-1}))^2 \\
\geq\;& -\frac{1}{N}\sum_i (\widehat{g}(\mathbf{x}_{i,t-1}) - g(\mathbf{x}_{i,t-1}))^2 + d(\widehat{g}, g)^2 \geq d(\widehat{g}, g)^2 - \epsilon_T^2/8 \geq (2^{k-2}\epsilon_T)^2/2.
\end{aligned}$$

Hence

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

We now bound the first term on the right hand side.

By Lemma 2 of Chen and Shen (1998),

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

Next, we bound the bracketing number of $\mathcal{C}_k$. Let $m_1, ..., m_{\mathcal{N}}$ 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 $f \in \mathcal{C}_k$, where $f(\mathbf{x}) = -(m(\mathbf{x}) - g(\mathbf{x}))^2$, there is m_j such that $\|m - m_j\|_\infty \leq \delta$. Let $f_j(\mathbf{x}) = -(g(\mathbf{x}) - m_j(\mathbf{x}))^2$. Then $\sup_{f\in\mathcal{C}_t:\|m-m_j\|_\infty\leq\delta} |f_j(\mathbf{x}_{i,t-1}) - f(\mathbf{x}_{i,T})| \leq (C + 2|g(\mathbf{x}_{i,t-1})|)\delta := b(\mathbf{x}_{i,t-1})\delta$. Hence $f \in [l_j, u_j]$, where $l_j = f_j - b\delta$ and $u_j = f_j + b\delta$. Moreover, $\mathbb{E}(u_j - l_j)^2 \leq C\delta^2$. This shows that $\{[l_j, u_j] : j \leq \mathcal{N}\}$ is a $C\delta$- bracket of $\mathcal{C}_k$, implying that the bracketing number satisfies

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

Then similar to the proof of (A.6), for $T_k = C(2^k\epsilon_T)^{s_0}$, $M_k = (2^{k-2}\epsilon_T)^2/2$, and

$\sigma_k^2 = C(2^k\epsilon_T)^2$,

$$2^{12}\int_{M_k/64}^{\sigma_k\sqrt{T_k}} \sqrt{\log \mathcal{N}_{[]}(\delta, \mathcal{C}_k, \|.\|_{L^2})}d\delta \leq M_k\sqrt{N}.$$

Hence all conditions of Lemma 1 of Chen and Shen (1998) are verified.

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

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

Step 4. Out-of-sample prediction. By Lemma 2 of Chen and Shen (1998), for any $\epsilon > 0$, there is $C > 0$, with probability at least $1-\epsilon$,

$$\max_i |\widehat{g}(\mathbf{x}_{i,T+1}) - g(\mathbf{x}_{i,T+1})| \leq \sup_{\mathbf{x}} |\widehat{g}(\mathbf{x}) - g(\mathbf{x})| \leq 2(2L)^{1-s_0} d(\widehat{g}, g)^{s_0} \leq Cb_T^{s_0}.$$

A.4 Proof of Theorem 4.4

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

$$\begin{aligned} y_{i,T+1} &= g_{\alpha,T+1}(\mathbf{x}_{i,T}) + g_{\text{riskP},T+1}(\mathbf{x}_{i,T}) + g_{\text{factor},T+1}(\mathbf{x}_{i,T}) + e_{i,T+1} \\ &= \widehat{g}_{\alpha,T}(\mathbf{x}_{i,T}) + \widehat{g}_{\text{riskP},T}(\mathbf{x}_{i,T}) + g_{\text{factor},T+1}(\mathbf{x}_{i,T}) + e_{i,T+1} + O_P(b_T^{s_0}). \end{aligned}$$

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

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

provided that $\mathbb{E}\mathbf{f}_{T+1} = \mathbb{E}(\mathbf{f}_{T+1}|\mathcal{F}_T)$. In addition,

$$e_{i,t+1} = \gamma_{\alpha,it} + \boldsymbol{\gamma}'_{\beta,it}\boldsymbol{\lambda}_{t+1} + \boldsymbol{\gamma}'_{\beta,it}(\mathbf{f}_{t+1} - \mathbb{E}\mathbf{f}_t) + u_{i,t+1}.$$

Hence $\mathbb{E}(e_{i,T+1}|\mathcal{F}_T) = 0$ provided that $\mathbb{E}(a|\mathcal{F}_T) = 0$, for $a \in \{\gamma_{\alpha,i,T}, \boldsymbol{\gamma}_{\beta,i,T}, \boldsymbol{\gamma}'_{\beta,i,T}\mathbf{f}_{T+1}\}$. □

B Additional Figures and Tables

References

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

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

Bai, J. (2003). Inferential theory for factor models of large dimensions. *Econometrica 71*, 135–171.

Bai, J. (2009). Panel data models with interactive fixed effects. *Econometrica 77*, 1229–1279.

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).

Bali, T., A. Goyal, D. Huang, F. Jiang, and Q. Wen (2021). Different strokes: Return predictability across stocks and bonds with machine learning and big data. *Swiss Finance Institute, Research Paper Series*, 20–110.

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.

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

Chernozhukov, V., C. B. Hansen, Y. Liao, and Y. Zhu (2019). Inference for heterogeneous effects using low-rank estimations. Technical report, CEMMAP working paper.

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.

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

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

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

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

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.

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.

Gagliardinia, P., E. Ossolac, and O. Scailletd (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.

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.

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. 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.

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.

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.

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*.

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.

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

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

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