NBER WORKING PAPER SERIES

3D-PCA: HIGH-DIMENSIONAL FACTOR MODELS WITH RESTRICTIONS

Martin Lettau

Working Paper 32261
http://www.nber.org/papers/w32261

NATIONAL BUREAU OF ECONOMIC RESEARCH
1050 Massachusetts Avenue
Cambridge, MA 02138
March 2024, Revised August 2026

I thank Anand Bharadwaj for research assistance and Ben Hebert, Ali Kakhbod, Valentina Raponi, and the seminar participants at Berkeley-Haas, Columbia, IESE, Johns Hopkins, Pompeu Fabra, Toulouse, and the CEMFI Big Data, INFORMS, and SITE conferences for their helpful comments. The views expressed herein are those of the author and do not necessarily reflect the views of the National Bureau of Economic Research.

3D-PCA: High-Dimensional Factor Models with Restrictions
Martin Lettau
NBER Working Paper No. 32261
March 2024, Revised August 2026
JEL No. C38, G0, G12

ABSTRACT

This paper proposes latent factor models for multidimensional panels called 3D-PCA. Factor weights are constructed from a small set of dimension-specific building blocks, which give rise to proportionality restrictions of factor weights. While the set of feasible factors is restricted, factors with long/short structures often found in pricing factors are admissible. I estimate the model using a 3-dimensional data set of double-sorted portfolios of 11 characteristics. Factors estimated by 3D-PCA have higher Sharpe ratios and smaller cross-sectional pricing errors than models with PCA or Fama-French factors. Since factor weights are subject to restrictions, the number of free parameters is small. Consequently, the model produces robust results in short time series and performs well in recursive out-of-sample estimations.

Martin Lettau
University of California, Berkeley
Haas School of Business
and CEPR
and also NBER
lettau@haas.berkeley.edu

1. Introduction

Factor models have been a central framework for modeling stock returns. Many asset pricing theories, such as the CAPM and Intertemporal CAPM, imply factor structures for returns. The arbitrage pricing theory (APT) of Ross (1976) implies that risk premia are linked to common factors via no-arbitrage conditions, leading to statistical representations of approximate factor models (Huberman (1982), Chamberlain and Rothschild (1983)). Building on these theoretical insights, Connor and Korajczyk (1986, 1988) proposed estimating latent factors using Principal Component Analysis (PCA). Recently, latent factor models have been used to address the "factor zoo" conundrum in asset pricing (Subrahmanyam (2010), Cochrane (2011)). This literature aims to condense the large number of characteristics associated with return spreads into a few pricing factors. While some recent papers use machine learning tools, e.g., elastic nets (Kozak, Nagel, and Santosh (2020)), regression trees (Bryzgalova, Huang, and Julliard (2023)), and neural nets (Chen, Pelger, and Zhu (2024)), most estimation methods for latent factor models are based on PCA. However, PCA estimations have several drawbacks. First, PCA factors often have little economic interpretation. Second, PCA factors have limited success in capturing the cross-section of expected returns (see, e.g., Lettau and Pelger (2020a,b)). Third, PCA estimations require a relatively long time series to obtain reliable results. Fourth, results from recursive out-of-sample estimations on short samples tend to be poor.

This paper proposes an estimation method for multidimensional panels called 3D-PCA that addresses some of these shortcomings. Standard PCA is based on eigenvectors and eigenvalues of the data's covariance (or second-moment) matrix, which "flattens" the data and removes any multidimensionality of the data set. If the data set is "large", PCA estimates factors and factor weights based on a "large" covariance matrix, which involves estimating many parameters and requires long samples. In contrast, 3D-PCA is specifically designed to exploit the multidimensional panel structure of the data and estimates dimension-specific factor weights and factors that involve only a small number of parameters. Intuitively, 3D-PCA estimates factor weights for each dimension separately, which serve as "building blocks" for the overall factor structure. The full factor weights are then constructed as combinations of the building blocks across the different dimensions. Different combinations of these building blocks yield different factor weights and, therefore, different factors. Factors that share a subset of building blocks share features implied by the common building blocks, yielding interpretable factors.

Since factors are built from the same building blocks, their factor weights are related and subject to constraints. The restrictions can be expressed as proportionality conditions along the dimensions of the panel data. As a result, 3D-PCA is more parsimonious than PCA and yields more precise estimations with smaller standard errors. In particular, 3D-PCA can be estimated in short samples. In the data set used in this paper, 3D-PCA can be reliably estimated with as few as six time series observations, whereas standard PCA requires much longer samples.

3D-PCA generalizes conventional two-dimensional factor models to higher-order data structures. To fix ideas, consider a three-dimensional data set of dimensions $(T \times P \times Q)$. Since PCA applies only to 2-dimensional matrices, a conventional approach requires "flattening" the data. If $T \gg PQ$, the three-dimensional data can be folded into a two-dimensional $(T \times PQ)$ matrix. Alternatively, one could estimate PCA for each time period $t = 1, \dots, T$ based on the series of $P \times Q$ matrices of cross-sectional data in each t. Similarly, PCA can also be estimated for each $p = 1, \dots, P$ or $q = 1, \dots, Q$ based on the series of $T \times Q$ or $T \times P$ matrices, respectively. In all cases, PCA ignores correlations across the dimension that is "fixed" and, therefore, discards information embedded in the joint dependence structure across all three dimensions. Examples of this approach include Jiang, Tang, and Zhou (2018), Balasubramaniam, Campbell, Ramadorai, and Ranish (2023), and Bryzgalova, Lerner, Lettau, and Pelger (2025). 3D-PCA resolves this inefficiency by *simultaneously* estimating the full system jointly across all dimensions. The resulting factor structure is *mutually consistent*: 3D-PCA imposes cross-equation restrictions that link the individual two-dimensional models and ensure they are jointly coherent. Moreover, the implied system of interconnected two-dimensional factor models is recovered as a by-product; see Lettau (2023) for more details.

In the empirical part of the paper, I estimate 3D-PCA using portfolios constructed from size/ characteristic quintile double-sorts for a set of characteristics. Hence, the panel of portfolios has three dimensions: the characteristic, the size quintile, and the quintile of the characteristic sort. 3D-PCA specifies a small set of vectors for each of the three dimensions that serve as “building blocks” for portfolio weights that form the factor. Different combinations of these building blocks yield different factor weights and, therefore, distinct factors. Since factors are based on the same building blocks, their factor weights are related and subject to restrictions. The restrictions can be expressed as proportionality conditions along the dimensions of the panel data. Factors that share a subset of building blocks share features implied by the common building blocks. The restrictions imply that the number of free parameters in 3D-PCA is several orders of magnitude smaller than that for comparable PCA estimations. While these restrictions limit the set of possible factor weights, I show that factors with familiar long-short structures are permissible. In particular, the Fama-French factors such as SMB and HML satisfy the restrictions and are, therefore, special cases of 3D-PCA factors.

I find that 3D-PCA factors have appealing features and dominate models based on PCA and Fama-French factors across several dimensions. First, 3D-PCA factors have straightforward economic interpretations given by the structures of the estimated building blocks. These building-block vectors exhibit the level, slope, and curvature patterns observed in many PCA estimates. Factor weights are determined by combinations of level, slope, and curvature across the different data dimensions. Some resulting factors have familiar patterns along the size and characteristic quintile dimensions related to long-only averages, small-minus-big, and high-minus-low. In contrast to the SMB and HML factors, 3D-PCA factors combine these patterns with multiple characteristics. For example, some factors are based on (approximately) equal weights across all characteristics, while others are based on only a few characteristics. However, all factors can be interpreted as interactions among the estimated building-block vectors. Second, Sharpe ratios of individual 3D-PCA factors are substantially higher than those of PCA and Fama-French factors. The highest annualized Sharpe ratio of estimated out-of-sample 3D-PCA factors is 1.21, and four 3D-PCA factors have annualized Sharpe ratios above 0.5. In contrast, none of the PCA factors exceeds this benchmark, consistent with Lettau and Pelger (2020b,a), who also find low Sharpe ratios of PCA factors.

Third, while PCA estimations typically require long time series, 3D-PCA can be estimated in short samples. In the data set used in this paper, 3D-PCA can be robustly estimated in rolling windows with as few as six time series observations. 3D-PCA estimates are substantially more stable for a given sample length than those from PCA models. The reason is that the construction of 3D-PCA factors imposes restrictions on factor weights, as mentioned above. Hence, fewer free parameters must be estimated. In contrast, PCA can be interpreted as an unrestricted estimation of factor weights. The difference in degrees of freedom is stark across all specifications I consider. For example, in the benchmark case, the weight matrix underlying 3D-PCA factors has only 48 free parameters, while the PCA weight matrix requires estimating 7,372 parameters.

Fourth, 3D-PCA factors explain the cross-section of portfolio returns substantially better than PCA factors. Parsimonious out-of-sample 3D-PCA factor models capture up to 80% of the variation in mean returns, while comparable PCA and Fama-French factor models capture only 50% to 60%. While 3D-PCA pricing errors are smaller for all characteristics, the fit for momentum, reversals, and variance-related portfolios is substantially better than that of PCA and Fama-French factor models. Many factor models do not capture returns of small/low portfolios (e.g., the small/low-BM portfolio). While these portfolios are also the most mispriced in 3D-PCA models, their pricing errors are substantially smaller than those for PCA and Fama-French models. Finally, Sharpe ratios for 3D-PCA factors are substantially higher than those for PCA factors. 3D-PCA models yield annualized out-of-sample Sharpe ratios of up to 2, while PCA and RP-PCA factors have Sharpe ratios below one. The 3D-PCA Sharpe ratios are comparable to those generated by state-of-the-art machine learning methods; see Gu, Kelly, and Xiu (2020) for a recent survey. All empirical results are robust to different sample lengths and different numbers of factors.

The 3D-PCA model can be estimated using tensor methods, which generalize vector and matrix algebra to higher dimensions. Formally, an n-dimensional panel data set forms an n-dimensional tensor. For example, a 3-dimensional tensor is a cuboid. The standard PCA of a 2-dimensional matrix can be written in terms of its singular value decomposition (SVD). Tucker (1966) provides a tensor decomposition that generalizes the matrix SVD to n-dimensional tensors; see Kolda and Bader (2009) for a survey of tensor methods and Lettau (2023) for an application to mutual funds. The 3D-PCA model is an implication of the Tucker decomposition and can be estimated using standard iterative numerical methods. Finally, I extend the model to allow for arbitrary dimensions and factorizations.

In Monte Carlo simulations, a 3-dimensional version of 3D-PCA, called 3D-PCA, yields more accurate estimates with smaller standard errors of the underlying factors and weights than standard PCA across a wide range of designs. The advantage of 3D-PCA is particularly pronounced in small samples, where it substantially outperforms PCA. The superior performance of 3D-PCA is likely due to its more parsimonious representation of the data, which requires estimating far fewer parameters than PCA.

The rest of the paper is organized as follows. Section 2 reviews the related literature. Section 3 presents an example that illustrates the intuition behind high-dimensional factor models, followed by a description of the general 3D-PCA model and its estimation in Section 4. Section 5 presents the estimation of the 3D-PCA model using a 3-dimensional panel of double-sorted portfolios for in-sample and out-of-sample estimations in rolling windows, followed by the description of the asset pricing results in Section 6. Section 7 uses 2D-PCA to estimate factors for a larger set of single-sorted portfolios and extends 3D-PCA to arbitrary dimensions, and Section 8 concludes.

2. Literature review

While there are many applications of tensor methods or high-dimensional factor models in other fields, there are only a few applications in finance. Lettau (2023) interprets Tucker decompositions in terms of 2-dimensional factor models and applies them to a panel of mutual fund characteristics. Babii, Ghysels, and Pan (2025) provide an asymptotic theory for a different tensor decomposition and apply it to the problem of imputing missing values in a multidimensional panel of firm characteristics. Varolgunes, Zhou, Yu, and Uddin (2023) use a non-linear tensor model to infer missing values and apply it to a data set of firm fundamentals. Jang and Seong (2023) use a Tucker decomposition as part of a deep learning model to predict stock returns. They apply the method to a small sample of 29 Chinese stocks with technical analysis indicators as features. Brandi, Gramatica, and Matteo (2020) propose a tensor decomposition method, Slice-Diagonal Tensor factorization, as an alternative to the Tucker decomposition. They apply the method to a 3-dimensional covariance matrix of asset returns. Dees (2019) estimates a tensor model using interest rate swap data of different maturities across several currencies.

This paper contributes to the literature on extensions of PCA for latent factor estimation. Recent contributions include Connor, Hagmann, and Linton (2012), Fan, Liao, and Wang (2016), and Kim, Korajczyk, and Neuhierl (2021), who allow factor betas to be nonparametric functions of observed characteristics. Kelly, Pruitt, and Su (2019) propose a related Instrumented-PCA (I-PCA) model in which factor betas are linear in characteristics. Gu, Kelly, and Xiu (2021) extend I-PCA to nonlinear specifications. Lettau and

Pelger (2020b,a) show that standard PCA may fail to identify weak factors. Their Risk Premium PCA (RP-PCA) estimator can overcome this limitation when weak factors have high Sharpe ratios. 3D-PCA can be combined with methods based on PCA estimates. For example, 3D-PCA factors can be used in place of PCA factors in Kozak, Nagel, and Santosh (2020)'s shrinkage estimator of the stochastic discount factor and the three-pass model of Giglio and Xiu (2021). Zaffaroni (2025) extends the asymptotic theory of PCA to allow for time-varying factor loadings, numbers of factors, and idiosyncratic risk.

3. Example: 2D-PCA

Consider the nine portfolios formed by a 2-dimensional portfolio sort as the intersection of three size (ME) and three book-to-market (BM) portfolios. Let $\mathbf{R}_t = [R_{pq,t}]$ be the 3-by-3 matrix of excess returns of (p,q)-portfolios where p and q are size and BM terciles, respectively. Factors are constructed as linear combinations of the test assets. Let $\mathbf{W} = [w_{pq}]$ be a 3-by-3 matrix of weights so that the associated factor is given by

$$F_t = \sum_{p,q} w_{pq}\, R_{pq,t} = \operatorname{vec}(\mathbf{W})^\top \operatorname{vec}(\mathbf{R}_t). \tag{1}$$

The literature has considered various methods to determine factor weights $\mathbf{W}$. For example, Fama and French (1993) use fixed weights to construct their SMB and HML factors:

$$\mathbf{W}_{\text{SMB}} = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}, \qquad \mathbf{W}_{\text{HML}} = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix}. \tag{2}$$

Alternatively, factors can be estimated by PCA, so that factor weights are given by the eigenvectors of the variance-covariance matrix or the second-moment matrix of returns. Instead, I consider factor weights that are constructed from separate vectors for the ME and BM dimensions. Let

$$\mathbf{v}^{(P)} = \begin{pmatrix} \mathbf{v}_1^{(P)} \\ \mathbf{v}_2^{(P)} \\ \mathbf{v}_3^{(P)} \end{pmatrix}, \qquad \mathbf{v}^{(Q)} = \begin{pmatrix} \mathbf{v}_1^{(Q)} \\ \mathbf{v}_2^{(Q)} \\ \mathbf{v}_3^{(Q)} \end{pmatrix}, \tag{3}$$

$$\mathbf{W} = \mathbf{v}^{(P)}\, \mathbf{v}^{(Q)\top}, \tag{4}$$

so that $\mathbf{W}$ is the 3-by-3 matrix of weights given by the outer product of $\mathbf{v}^{(P)}$ and $\mathbf{v}^{(Q)}$. The factor weight of portfolio (p,q) is therefore $w_{pq} = v_p^{(P)}\, v_q^{(Q)}$. Suppose there are two such vectors for the size dimension, as well as for the BM dimension, and collect them in the $(3{\times}2)$ matrices

$$\mathbf{V}^{(P)} = \left[\mathbf{v}_1^{(P)}, \mathbf{v}_2^{(P)}\right], \qquad \mathbf{V}^{(Q)} = \left[\mathbf{v}_1^{(Q)}, \mathbf{v}_2^{(Q)}\right]. \tag{5}$$

Factors are formed by combining the columns of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$. Since both matrices have two columns, there are four possible combinations, so four different factors can be created. In other words, the columns of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ form the building blocks of factor weights.

To develop further intuition, consider the following specific example:

$$\mathbf{V}^{(P)} = \mathbf{V}^{(Q)} = \begin{pmatrix} 1 & -1 \\ 1 & 0 \\ 1 & 1 \end{pmatrix}. \tag{6}$$

The two columns have the familiar "level" and "slope" patterns often encountered in PCA estimations.

Combining the columns of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ yields weights of four factors:

$$\mathbf{W}_{11} = \mathbf{v}_1^{(P)}\mathbf{v}_1^{(Q)^\top} = \begin{pmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{pmatrix} \qquad \mathbf{W}_{12} = \mathbf{v}_1^{(P)}\mathbf{v}_2^{(Q)^\top} = \begin{pmatrix}-1 & 0 & 1\\ -1 & 0 & 1\\ -1 & 0 & 1\end{pmatrix} \tag{7}$$

$$\mathbf{W}_{21} = \mathbf{v}_2^{(P)}\mathbf{v}_1^{(Q)^\top} = \begin{pmatrix}-1 & -1 & -1\\ 0 & 0 & 0\\ 1 & 1 & 1\end{pmatrix} \qquad \mathbf{W}_{22} = \mathbf{v}_2^{(P)}\mathbf{v}_2^{(Q)^\top} = \begin{pmatrix}1 & 0 & -1\\ 0 & 0 & 0\\ -1 & 0 & 1\end{pmatrix}, \tag{8}$$

which can be collected in the (9×4)-matrix of factor weights $\mathbf{W}^{2\text{D}} = [\text{vec}(\mathbf{W}_{11}), \text{vec}(\mathbf{W}_{21}), \text{vec}(\mathbf{W}_{12}), \text{vec}(\mathbf{W}_{22})]$. Note that $\mathbf{W}^{2\text{D}}$ can also be expressed using the Kronecker product: $\mathbf{W}^{2\text{D}} = \mathbf{V}^{(Q)} \otimes \mathbf{V}^{(P)}$. The vector of four factors is given by $\mathbf{F}_t^{2\text{D}} = \mathbf{W}^{2\text{D}^\top}\text{vec}(\mathbf{R}_t)$.

Comparing the weight matrices in (7) and (8) to (2) shows that $\mathbf{W}_{21}$ is identical to SMB weights and that $\mathbf{W}_{12}$ is equal to HML weights. Hence, the Fama-French factors are special cases of 2D-PCA. Furthermore, the rows and columns of each weight matrix are proportional. In addition, the rows of $\mathbf{W}_{11}$ and $\mathbf{W}_{21}$ are proportional as are the rows of $\mathbf{W}_{12}$ and $\mathbf{W}_{22}$. The columns have similar proportionality properties. Of course, these patterns arise because the weight matrices are constructed from the same building blocks, i.e., the columns of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$.

The patterns in the weight matrices can also be understood as restrictions that are imposed by the construction of the weight matrices in (3) and (4). The (9×4)-matrix of factor weights $\mathbf{W}^{2\text{D}}$ has 36 elements, while the $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ matrices have a total of 12 elements. Hence, the elements of $\mathbf{W}^{2\text{D}}$ are subject to 24 restrictions. In the general model introduced below, $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ are assumed to be orthonormal, which imposes six additional restrictions, so that $\mathbf{W}^{2\text{D}}$ has six degrees of freedom. On the one hand, the $\mathbf{W}^{2\text{D}}$ matrix is highly restricted, and the set of feasible $\mathbf{W}^{2\text{D}}$ is limited. On the other hand, the parametrization allows for long/short patterns often found in pricing factors. In PCA, the weight matrix $\mathbf{W}$ is composed of eigenvectors and is, therefore, orthonormal but not subject to other restrictions. The PCA weight matrix has 36 elements, and orthonormality imposes seven restrictions. Hence, the weight matrix has $36 - 7 = 29$ free parameters, almost five times the number of free parameters of $\mathbf{W}^{2\text{D}}$. In standard factor analysis, the full (9×4) weight matrix $\mathbf{W}$ is estimated, for example by PCA. In contrast, 2D-PCA estimates the small $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ matrices, and then constructs $\mathbf{W}^{2\text{D}}$ as their Kronecker product. The empirical estimates of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$, shown in Appendix A.4, have a "level" and "slope" structure similar to the example in (6).

The model considered in this paper generalizes the 2D-PCA example to data sets with an n-dimensional panel structure and is therefore called nD-PCA. Factor weights are constructed from n matrices $\mathbf{V}^{(i)}$ for $i = 1, \ldots, n$ so that the weight matrix is given by the Kronecker product $\mathbf{W}^{n\text{D}} = \mathbf{V}^{(1)} \otimes \mathbf{V}^{(2)} \otimes \cdots \otimes \mathbf{V}^{(n)}$. The matrices $\mathbf{V}^{(i)}$ contain the building blocks of factor weights for dimension i. The number of free parameters of $\mathbf{W}^{n\text{D}}$ is small since the number of elements in the $\mathbf{V}^{(i)}$ matrices is typically much smaller than the number of elements in $\mathbf{W}^{n\text{D}}$. Nevertheless, long/short factors commonly found in asset pricing are admissible. Instead of estimating the large $\mathbf{W}^{n\text{D}}$ matrix as in PCA, nD-PCA estimates the small $\mathbf{V}^{(i)}$ matrices. Given the estimated $\mathbf{V}^{(i)}$ matrices, the weight matrix $\mathbf{W}^{n\text{D}}$ is constructed as their Kronecker product.

4. High-dimensional factor models: 3D-PCA

This section introduces the 3D-PCA model for estimating latent factors from multidimensional panels of asset returns. For ease of exposition, I focus on the 3D-PCA model, which is applicable to three-dimensional portfolio constructions, such as double-sorted portfolios with multiple characteristics. Section 7.2 shows how the model can be extended to arbitrary dimensions. The data set used in this paper consists of double-sorted portfolios of $C = 11$ characteristics. For each characteristic, the portfolios are based on $P = 5$ size quintiles and $Q = 5$ characteristic quintiles, creating a three-dimensional structure of the portfolio construction. The 3D-PCA model can be understood as an extension of the 2D-PCA model described in section 3 to three dimensions. In principle, one could estimate 2D-PCA models for each characteristic separately and then combine the resulting factors. However, this approach ignores potential correlations across characteristics and thus does not use the full information in the data. Instead, the 3D-PCA model estimates factors *jointly* across all characteristics in one step. The advantage of the 3D-PCA model is that it leverages information across characteristics while still accounting for the three-dimensional structure of portfolio construction.

The data are assembled in a four-dimensional panel $\boldsymbol{\mathcal{X}} = [x_{tcpq}]$, where the first dimension is time, the second dimension corresponds to characteristics, and the third and fourth dimensions are the size and characteristic-percentile portfolios. Hence, x_{tcpq} is the excess return of the portfolio given by the p-th size and q-th characteristic portfolio of characteristic c in period t. The total number of portfolios is $N = CPQ$. Let $\mathbf{X}_{(T)}$ be the $(T \times CPQ)$-dimensional matrix whose columns are the time series of portfolio returns.

4.1. *From matrices to tensors*

Multidimensional panels can be represented as *tensors*, which generalize vectors and matrices to higher dimensions. For example, a 3-dimensional tensor is a cuboid, while the data set used in this paper forms a 4-dimensional tensor. While tensor notation is more complex and differs from the familiar matrix notation in several ways, tensor algebra is a straightforward extension of matrix algebra; see Kolda and Bader (2009) and Lettau (2023). Throughout the paper, I use the following notation:

$$\begin{aligned} \text{scalar: } & x \in \mathbb{R} \\ \text{vector: } & \mathbf{x} \in \mathbb{R}^{I} \\ \text{matrix: } & \mathbf{X} \in \mathbb{R}^{I_1} \times \mathbb{R}^{I_2} \\ m\text{-th order tensor: } & \boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1} \times \mathbb{R}^{I_2} \times \cdots \times \mathbb{R}^{I_m}. \end{aligned}$$

Hence, a zero-order tensor is a scalar, a first-order tensor is a vector, a second-order tensor is a matrix, and a third-order tensor is a cuboid. Each of the m dimensions of a tensor is called a *mode*.

The *slice* of an m-dimensional tensor is an $(m-1)$-dimensional tensor obtained by fixing one of the indices to a given value. For example, a t-slice of $\boldsymbol{\mathcal{X}}$ is a 3-dimensional tensor of size $(C \times P \times Q)$, $\boldsymbol{\mathcal{X}}_{(t)cpq}$, consisting of N portfolio returns in period t. A *fiber* of $\boldsymbol{\mathcal{X}}$ is a vector obtained by fixing all indices except one. For example, the mode-t fiber, $\mathbf{x}_{(cpq)t}$, is the $(T \times 1)$ vector of the time series for the p-th-size quintile and q-th-quintile portfolio of characteristic c. Figure C.1 shows a 3-dimensional tensor, its fibers, and its slices. When the context is clear, I will sometimes write $\boldsymbol{\mathcal{X}}_{(t)cpq}$ and $\mathbf{x}_{(cpq)t}$ as $\boldsymbol{\mathcal{X}}_t$ and $\mathbf{x}_t$.

A tensor can be *matricized* by stacking its fibers in a matrix. For example, $\mathbf{X}_{(T)} = \text{mat}_T(\boldsymbol{\mathcal{X}})$ is the $(T \times CPQ)$ matrix whose rows are the t-fibers $\mathbf{x}_{(cpq),t}$. In other words, the columns of $\mathbf{X}_{(T)}$ are the time series of returns of the CPQ portfolios. Figure C.2 shows the matricization of a 3-dimensional tensor along each dimension. Similarly, $\mathbf{x} = \text{vec}(\boldsymbol{\mathcal{X}})$ denotes the vectorized tensor of dimension $(TCPQ \times 1)$. The n-mode product, denoted $\times_n$, is the multiplication of a tensor and a conforming matrix along the n-th dimension. For example, the mode-1 product of a $(I \times J \times K)$ tensor $\boldsymbol{\mathcal{X}}$ and a $(L \times I)$ matrix $\mathbf{A}_1$ is equal to a $(L \times J \times K)$ tensor $\boldsymbol{\mathcal{X}} \times_1 \mathbf{A}_1$ (see Figure C.3). Note that the standard matrix product can be written in tensor notation: $\mathbf{A}_1 \mathbf{X} \mathbf{A}_2^{\top} = \mathbf{X} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2$.

Let $\mathbf{a}_1, \ldots, \mathbf{a}_m$ be m vectors of lengths $i_1, \ldots, i_m$. The outer product $\circ$ of the m vectors is an m-dimensional tensor of size $(i_1 \times \cdots \times i_m)$: $\boldsymbol{\mathcal{X}} = \mathbf{a}_1 \circ \cdots \circ \mathbf{a}_m$. Panel C of Figure C.3 illustrates the outer product of three vectors. Note that the outer product of two vectors $\mathbf{a}_1 \circ \mathbf{a}_2 = \mathbf{a}_1 \mathbf{a}_2^{\top} = \mathbf{X}$ yields an $(i_1 \times i_2)$ matrix $\mathbf{X}$.

4.2. The 3D-PCA model

High-dimensional factor models are a natural extension of PCA for matrices. Consider first the singular value decomposition (SVD) of a matrix $\mathbf{X}$ of asset returns:

$$\mathbf{X} = \mathbf{U}^{(1)}\,\mathbf{H}\,\mathbf{U}^{(2)^\top}, \tag{9}$$

where the columns of $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are the eigenvectors of $\mathbf{X}\mathbf{X}^\top$ and $\mathbf{X}^\top\mathbf{X}$, respectively, and $\mathbf{H}$ is the diagonal matrix of the square roots of the corresponding non-zero eigenvalues. Since $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are orthonormal, (9) can be written in factor form. Factors are linear combinations of asset returns and are given by $\mathbf{F} = \mathbf{U}^{(1)}\,\mathbf{H} = \mathbf{X}\,\mathbf{U}^{(2)}$ where the columns of $\mathbf{U}^{(2)}$ are the factor weights. The truncated K-factor model uses only eigenvectors of the K largest eigenvalues so that (9) becomes an approximation: $\mathbf{X} \approx \mathbf{U}_K^{(1)}\,\mathbf{H}_K\,\mathbf{U}_K^{(2)^\top}$. Note that the SVD can be written in tensor notation as

$$\mathbf{X} = \mathbf{H} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \tag{10}$$

$$\mathbf{F} = \mathbf{H} \times_1 \mathbf{U}^{(1)} = \mathbf{X} \times_2 \mathbf{U}^{(2)^\top}. \tag{11}$$

Tucker (1966) shows that the matrix SVD can be extended to higher dimensional tensors. The Tucker decomposition states that an m-dimensional tensor can be written as a "small" m-dimensional "core" tensor and m matrices. The Tucker decomposition with (K_T, K_C, K_P, K_Q) factors is given by

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{G}} \times_1 \mathbf{V}^{(T)} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)} \times_4 \mathbf{V}^{(Q)}, \tag{12}$$

where $\times_n$ denotes the n-mode tensor product. $\boldsymbol{\mathcal{G}}$ is a $(K_T{\times}K_C{\times}K_P{\times}K_Q)$ *core* tensor and $\mathbf{V}^{(T)}$, $\mathbf{V}^{(C)}$, $\mathbf{V}^{(P)}$, $\mathbf{V}^{(Q)}$ are $(T{\times}K_T)$, $(C{\times}K_C)$, $(P{\times}K_P)$, $(Q{\times}K_Q)$ matrices, respectively. As in the SVD, each $\mathbf{V}^{(i)}$ is normalized to be orthonormal. When $K_T = T, K_C = C, K_P = P, K_Q = Q$, the approximation in (12) becomes an equality. Figure C.4 shows the Tucker decomposition of a 3-dimensional tensor. The $\mathbf{V}^{(i)}$ matrices correspond to $\mathbf{U}^{(i)}$ and $\boldsymbol{\mathcal{G}}$ corresponds to $\mathbf{H}$. Note that the matrix SVD is a special case of the Tucker decomposition (12) when $\boldsymbol{\mathcal{X}}$ is a 2-dimensional matrix. In this case, the core tensor $\boldsymbol{\mathcal{G}}$ is diagonal and is equivalent to (10). Kolda and Bader (2009) provide a detailed treatment of the Tucker decomposition, while Lettau (2023) develop the intuition in the context of a finance application.

The 3D-PCA model is based on a simplified version of the Tucker decomposition (12) and is defined as follows. Since we are interested in the factor structure of the portfolios, the time dimension does not have to be factored, which suggests a *partial* Tucker decomposition with K_C, K_P, and K_Q factors for the C, P, and Q dimensions:[1]

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{F}} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)} \times_4 \mathbf{V}^{(Q)}. \tag{13}$$

Since the time dimension is not factored, the core tensor $\boldsymbol{\mathcal{F}}$ has T elements in the first dimension, is of dimension $(T{\times}K_C{\times}K_P{\times}K_Q)$, and is interpreted as a factor tensor. The $\mathbf{V}^{(i)}$ matrices are identical to the $\mathbf{V}^{(i)}$ matrices in the Tucker decomposition (12). As in PCA, the factor tensor $\boldsymbol{\mathcal{F}}$ is a linear combination of the data tensor $\boldsymbol{\mathcal{X}}$ and the columns of the $\mathbf{V}^{(i)}$ matrices are the factor weights:

$$\boldsymbol{\mathcal{F}} \approx \boldsymbol{\mathcal{X}} \times_2 \mathbf{V}^{(C)^\top} \times_3 \mathbf{V}^{(P)^\top} \times_4 \mathbf{V}^{(Q)^\top}, \tag{14}$$

which generalizes (11) to three dimensions. The difference is that the matrix of PCA factors is the product of the data matrix and a weight matrix, while 3D-PCA factors are given by multiplying the data tensor by three weight matrices. Using the properties of the n-mode product, (13) can be written in terms of matrices. Define $\mathbf{W}^{3\mathrm{D}}$ as the Kronecker product of $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$:

$$\mathbf{W}^{3\mathrm{D}} = \mathbf{V}^{(Q)} \otimes \mathbf{V}^{(P)} \otimes \mathbf{V}^{(C)}, \tag{15}$$

[1] All results reported in the paper are similar when the full Tucker model is used.

which is a $(CPQ \times K_C K_P K_Q)$-dimensional matrix. Note that $\mathbf{W}^{3D}$ is orthonormal since the $\mathbf{V}^{(i)}$ matrices are orthonormal. Matricizing $\boldsymbol{\mathcal{X}}$ along the time dimension gives the $(T \times CPQ)$ matrix $\mathbf{X}_{(T)}$, whose columns are the time series of portfolio returns. Multiplying $\mathbf{X}_{(T)}$ by $\mathbf{W}^{3D}$ yields a matrix of factors:

$$\mathbf{F}^{3D}_{(T)} = \mathbf{X}_{(T)} \, \mathbf{W}^{3D}. \tag{16}$$

$\mathbf{F}^{3D}_{(T)}$ is a $(T \times K_C K_P K_Q)$-dimensional matrix with columns F^{3D}_{cpq}. Note that $\mathbf{F}^{3D}_{(T)}$ is the matricized factor tensor, $\mathbf{F}^{3D}_{(T)} = \text{mat}_T(\boldsymbol{\mathcal{F}})$ and $\mathbf{W}^{3D}$ is the matrix of factor weights. Each of its $K_C K_P K_Q$ columns is a $(CPQ \times 1)$ vector of weights that yields a factor when multiplied by $\mathbf{X}_{(T)}$.

This derivation shows that the 3D-PCA model is a direct extension of the 2D-PCA model in section 3. The main difference is that the matrix of factor weights of the 3D-PCA model, $\mathbf{W}^{3D}$, is given by the Kronecker product of three matrices, $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}, \mathbf{V}^{(Q)}$ matrices, while $\mathbf{W}^{2D}$ is the Kronecker product of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$. Moreover, the intuition behind the factor construction carries over to the 3D-PCA model, in that the columns of the $\mathbf{V}^{(i)}$ matrices can be understood as building blocks that form the set of factor weights.

In 2D-PCA, the weights of a factor are given by a matrix created by the outer product of two vectors, one from $\mathbf{V}^{(P)}$ and one from $\mathbf{V}^{(Q)}$. In contrast, 3D-PCA weights of a factor are given by the outer product of three vectors, one from each of the $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$ matrices. Let $\mathbf{v}^{(C)}_c, \mathbf{v}^{(P)}_p$, and $\mathbf{v}^{(Q)}_q$ be the c-th, p-th, and q-th columns of $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$, respectively. Their outer product $\boldsymbol{\mathcal{W}}_{cpq} = \mathbf{v}^{(C)}_c \circ \mathbf{v}^{(P)}_p \circ \mathbf{v}^{(Q)}_q$ is a three-dimensional tensor of dimension $(C \times P \times Q)$ with elements

$$w_{cpq,ijk} = v^{(C)}_{ci} \, v^{(P)}_{pj} \, v^{(Q)}_{qk} \quad \text{for} \quad i = 1, \dots, C, j = 1, \dots, P, k = 1, \dots, Q, \tag{17}$$

where $v^{(C)}_{ci}, v^{(P)}_{pj}$ and $v^{(Q)}_{qk}$ are the i-th, j-th, and k-th elements of $\mathbf{v}^{(C)}_c, \mathbf{v}^{(P)}_p$, and $\mathbf{v}^{(Q)}_q$, respectively. $\boldsymbol{\mathcal{W}}_{cpq}$ comprises weights that can be used to construct factors. The (c, p, q)-factor is given by

$$F_{cpq,t} = \sum_i \sum_j \sum_k w_{cpq,ijk} \, x_{tijk} \tag{18}$$

$$= \text{vec}(\boldsymbol{\mathcal{W}}_{cpq})^\top \, \mathbf{x}_t. \tag{19}$$

Repeating this procedure for all combinations of (c, p, q) yields $K_C K_P K_Q$ factors that are collected in $\boldsymbol{\mathcal{F}}$. The columns of the $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$ matrices can be understood as building blocks that create the set of factor weights. The total number of vectors is $K_C + K_P + K_Q$, yielding $K_C K_P K_Q$ distinct factors.

Since $\mathbf{W}^{3D}$ is orthonormal, 3D-PCA implies a standard factor representation $\mathbf{X}_{(T)} \approx \mathbf{F}^{3D}_{(T)} \, \mathbf{W}^{3D^\top}$. The difference relative to a standard matrix factor model is that $\mathbf{F}^{3D}_{(T)}$ and $\mathbf{W}^{3D^\top}$ are derived from the tensor factorization of $\boldsymbol{\mathcal{X}}$. In matrix factor models, the only restriction on the matrix of factor weights is that its columns are orthonormal. Since the factor weights of 3D-PCA are given by outer products of a small number of vectors, they are subject to restrictions similar to those in the 2D-PCA example in section 3. For example, all 2-dimensional slices of the $\boldsymbol{\mathcal{W}}_{cpq}$ factor tensors are proportional. To see this, fix c, p, and q, and consider the slices of $\boldsymbol{\mathcal{W}}_{cpq}$ corresponding to the characteristics. Recall that $\boldsymbol{\mathcal{W}}_{cpq} = \mathbf{v}^{(C)}_c \circ \mathbf{v}^{(P)}_p \circ \mathbf{v}^{(Q)}_q$. Hence, the slice corresponding to characteristic i is given by $v^{(C)}_{ci} \left(\mathbf{v}^{(P)}_p \circ \mathbf{v}^{(Q)}_q \right)$.[2] Thus, all c-slices are proportional to the $(P \times Q)$-dimensional matrix $\mathbf{v}^{(P)}_p \circ \mathbf{v}^{(Q)}_q$. Next, consider the weight tensor $\boldsymbol{\mathcal{W}}_{c'pq}$ that is created by the outer product of the same $v^{(P)}_p$ and $v^{(Q)}_q$ vectors but a different vector $v^{(C)}_{c'}$. Since $\boldsymbol{\mathcal{W}}_{c'pq} = \mathbf{v}^{(C)}_{c'} \circ \mathbf{v}^{(P)}_p \circ \mathbf{v}^{(Q)}_q$, all c-slices $v^{(C)}_{c'i} \left(\mathbf{v}^{(P)}_p \circ \mathbf{v}^{(Q)}_q \right)$ are also proportional to $\mathbf{v}^{(P)}_p \circ \mathbf{v}^{(Q)}_q$. Therefore, all factor weights that stem from the same columns of $\mathbf{V}^{(P)^\top}$ and $\mathbf{V}^{(Q)^\top}$ are proportional. The same argument holds for combinations of columns of $\mathbf{V}^{(C)^\top}$ and $\mathbf{V}^{(P)^\top}$, and columns of $\mathbf{V}^{(C)^\top}$, and $\mathbf{V}^{(Q)^\top}$. Therefore, the set of factor weights of the 3D-PCA model is subject to within-factor and across-factor restrictions.

In matrix-PCA, the contribution of a factor to the overall variation in the data is given by the associated

[2] See Figure C.5 for an illustration.

eigenvalue of the eigenvector that is used to construct the factor. A similar result can be derived for the 3D-PCA model. Consider the "full" Tucker decomposition (12) with $K_T = T, K_C = C, K_P = P, K_Q = Q$, so that it holds with equality. Since the $\mathbf{V}^{(i)}$ matrices are orthonormal, the overall variation of the data tensor $\boldsymbol{\mathcal{X}}$ is equal to the squared norm of the core tensor $\boldsymbol{\mathcal{G}}$:

$$\|\boldsymbol{\mathcal{X}}\|^2 = \sum_{t,c,p,q} x_{tcpq}^2 = \sum_{t,c,p,q} g_{tcpq}^2 = \|\boldsymbol{\mathcal{G}}\|^2. \tag{20}$$

Therefore, the squared elements of the core tensor $\boldsymbol{\mathcal{G}}$ can be interpreted as the contribution of the corresponding factor to the overall variation in the data and play a role similar to eigenvalues in matrix-PCA. In the partial Tucker decomposition (13) with $K_C = C, K_P = P, K_Q = Q$, $\boldsymbol{\mathcal{F}}$ corresponds to the core tensor, so that $\|\boldsymbol{\mathcal{X}}\|^2 = \|\boldsymbol{\mathcal{F}}\|^2$. Recall that $\mathbf{X}_{(T)} = \text{mat}_T(\boldsymbol{\mathcal{X}})$ is the $(T \times CPQ)$ matrix whose columns are time series of portfolio returns. Similarly, $\mathbf{F}_{(T)} = \text{mat}_T(\boldsymbol{\mathcal{F}})$ is the $(T \times K_C K_P K_Q)$ matrix of time series of factors. Hence,

$$\|\boldsymbol{\mathcal{X}}\|^2 = \|\mathbf{X}_{(T)}\|^2 = \|\mathbf{F}_{(T)}\|^2 \tag{21}$$

$$= \sum_{c,p,q} \left(\sum_t F_{cpq,t}^2 \right), \tag{22}$$

which implies that the contribution of a factor F_{cpq} to the overall variation in the data is given by its time series sum of squares, $\sum_t F_{cpq,t}^2$.

4.3. Number of parameters

It is instructive to compare the number of free parameters in the 3D-PCA and SVD/PCA models. To simplify the notation, let N_i be the size of dimension $i \in \{C, P, Q\}$ and K_i be the number of components in dimension i. Thus, the total number of test assets is $N = \prod_i N_i$ and the total number of factors is $K = \prod_i K_i$. The factor tensor $\boldsymbol{\mathcal{F}}$ is of dimension $T \times \prod_i K_i$ and the loadings matrices $\mathbf{V}^{(i)}$ are $N_i \times K_i$-dimensional.

Consider first the SVD in (9). The total number of parameters in the SVD is $TK + K^2 + NK$. However, the SVD is not unique and can be rotated, which imposes $2K^2$ restrictions on the parameters.[3] Thus, the number of free parameters in the SVD is $TK + K^2 + NK - 2K^2 = TK + (N-K)K$.[4] Next, consider the number of parameters in the 3D-PCA model (13). The 3D-PCA model in (13) can also be rotated by arbitrary nonsingular $K_i \times K_i$ matrices $\mathbf{S}_i$, which imposes $\sum_i K_i^2$ restrictions on the parameters. Thus, the number of free parameters in the 3D-PCA model is $T \prod_i K_i + \sum_i N_i K_i - \sum_i K_i^2 = T \prod_i K_i + \sum_i (N_i - K_i) K_i$. The difference in the number of parameters between the SVD and 3D-PCA models is $(\prod_i N_i - \prod_i K_i) \prod_i K_i - \sum_i (N_i - K_i) K_i$.

From a factor model perspective, it is not necessary to compute all elements of the SVD or 3D-PCA decompositions. Only the weighting matrices are needed to compute the factors as linear combinations of the data. In the 2-dimensional case, the matrix of weights is $\mathbf{W}^{\text{PC}} = \mathbf{U}_K^{(2)}$ while in the 3D-PCA case, the weight matrix $\mathbf{W}^{\text{3D}}$ is given by the Kronecker product of the three $\mathbf{V}^{(i)}$ matrices as shown in (15). Since $\mathbf{W}^{\text{PC}}$ is an orthonormal matrix, it has $NK - K(K+1)/2$ parameters while $\mathbf{W}^{\text{3D}}$ has $\sum_i N_i K_i - \sum_i K_i(K_i+1)/2$ parameters. The results are summarized as follows:

---

[3] $\mathbf{X} \approx \mathbf{U}_K^{(1)} \mathbf{S}_1 \mathbf{S}_1^{-1} \mathbf{H}_K \mathbf{S}_2 \mathbf{S}_2^{-1} \mathbf{U}_K^{(2)^\top} = \tilde{\mathbf{U}}_K^{(1)} \tilde{\mathbf{H}}_K \tilde{\mathbf{U}}_K^{(2)^\top}$ for arbitrary nonsingular $K \times K$ matrices $\mathbf{S}_1, \mathbf{S}_2$.

[4] An alternative derivation is as follows. Since $\mathbf{U}_K^{(1)}$ and $\mathbf{U}_K^{(2)}$ are orthonormal matrices, they are subject to $K(K+1)/2$ restrictions each. Hence, the SVD has $TK - K(K+1)/2$ parameters for $\mathbf{U}_K$ plus $NK - K(K+1)/2$ parameters for $\mathbf{V}_K$, plus K parameters for the diagonal matrix $\mathbf{H}_K$. The total number of parameters in this case is therefore $(T + N - K)K = TK + (N-K)K = T \prod_i K_i + (\prod_i N_i - \prod_i K_i) \prod_i K_i$.

	Decomposition	Weight matrix $\mathbf{W}$
SVD/PCA	$T\prod_i K_i + (\prod_i N_i - \prod_i K_i)\prod_i K_i$	$\prod_i N_i K_i - \prod_i K_i(\prod_i K_i + 1)/2$
3D-PCA	$T\prod_i K_i + \sum_i (N_i - K_i)K_i$	$\sum_i N_i K_i - \sum_i K_i(K_i + 1)/2$

Note that the number of parameters in the 3D-PCA model grows linearly with the number of dimensions, while it grows multiplicatively in the SVD/PCA model. It is easy to see that the number of parameters in the 3D-PCA model is much smaller than in the SVD/PCA model if $K_i \ll N_i$ since $\prod_i N_i \gg \sum_i N_i, \prod_i N_i K_i \gg \sum_i N_i K_i$, and $\prod_i (N_i - K_i) \gg \sum_i (N_i - K_i)$. Alternatively, consider the special case where $N_i = N$ and $K_i = K$ for all i and $T = N = n^3$, so that the number of observations is $TN = n^6$. Let $\alpha = K/N = (k/n)^3$ be the ratio of the numbers of factors and test assets. The number of parameters divided by the number of observations for SVD is $2\alpha - \alpha^2 = \alpha(2-\alpha)$, and for 3D-PCA it is $\alpha + 3\alpha^{1/3}/n^4 - 3\alpha^{2/3}/n^4 \longrightarrow \alpha < \alpha(2-\alpha)$ as $n \to \infty$. Thus, the number of parameters relative to the number of observations is smaller for 3D-PCA than for SVD/PCA for large n.

For the data set used in the paper, the differences in parameter counts are stark. The SVD/PCA model has 7,047 parameters, while the 3D-PCA model has only 45, and the SVD decomposition has 6,660 more parameters than the 3D-PCA decomposition. Table B.1 in the Appendix shows the number of parameters in SVD/PCA and 3D-PCA models for different values of N_i and K_i, which will be used in the Monte Carlo simulations in section 4.5 of the paper. In all cases, the number of parameters in the 3D-PCA model is significantly smaller than that of the SVD/PCA model. As we will see in the empirical part of the paper, this difference drastically affects the estimation of both models.

4.4. Estimation

Given (K_C, K_P, K_Q), the Tucker decomposition (13) can be written as

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{F}} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)} \times_4 \mathbf{V}^{(Q)} + \boldsymbol{\mathcal{E}}, \tag{23}$$

where $\boldsymbol{\mathcal{E}}$ is the approximation error. The objective is to find $\boldsymbol{\mathcal{F}}, \mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$ that minimize the L^2-norm of $\boldsymbol{\mathcal{E}}$:

$$\hat{\boldsymbol{\mathcal{F}}}, \hat{\mathbf{V}}^{(C)}, \hat{\mathbf{V}}^{(P)}, \hat{\mathbf{V}}^{(Q)} = \operatorname{argmin} \|\boldsymbol{\mathcal{E}}\|, \tag{24}$$

subject to the restriction that $\hat{\mathbf{V}}^{(C)}, \hat{\mathbf{V}}^{(P)}, \hat{\mathbf{V}}^{(Q)}$ are orthonormal. Since there is no closed-form solution to (24), the problem has to be solved numerically. One candidate solution is the Higher-Order SVD (HOSVD), as suggested by De Lathauwer, De Moor, and Vandewalle (2000). HOSVD solves a series of 2-dimensional eigenvalue problems that yield estimates of the $\mathbf{V}^{(i)}$ matrices followed by the computation of $\boldsymbol{\mathcal{F}}$. In the 3-dimensional case considered here, HOSVD is given by the following procedure:

1. Unfold $\boldsymbol{\mathcal{X}}$ along the C dimension: $\mathbf{X}_{(C)} = \text{mat}_C(\boldsymbol{\mathcal{X}})$. Compute the SVD of $\mathbf{X}_{(C)}$ and set $\hat{\mathbf{V}}^{(C)}$ to the first K_C left singular vectors, i.e., the eigenvectors of $\mathbf{X}_{(C)}\mathbf{X}_{(C)}^\top$.
2. Unfold $\boldsymbol{\mathcal{X}}$ along the P dimension: $\mathbf{X}_{(P)} = \text{mat}_P(\boldsymbol{\mathcal{X}})$. Compute the SVD of $\mathbf{X}_{(P)}$ and set $\hat{\mathbf{V}}^{(P)}$ to the first K_P left singular vectors, i.e., the eigenvectors of $\mathbf{X}_{(P)}\mathbf{X}_{(P)}^\top$.
3. Unfold $\boldsymbol{\mathcal{X}}$ along the Q dimension: $\mathbf{X}_{(Q)} = \text{mat}_Q(\boldsymbol{\mathcal{X}})$. Compute the SVD of $\mathbf{X}_{(Q)}$ and set $\hat{\mathbf{V}}^{(Q)}$ to the first K_Q left singular vectors, i.e., the eigenvectors of $\mathbf{X}_{(Q)}\mathbf{X}_{(Q)}^\top$.
4. Set $\hat{\boldsymbol{\mathcal{F}}} = \boldsymbol{\mathcal{X}} \times_2 \hat{\mathbf{V}}^{(C)\top} \times_3 \hat{\mathbf{V}}^{(P)\top} \times_4 \hat{\mathbf{V}}^{(Q)\top}$.

In general, HOSVD does not minimize $\|\boldsymbol{\mathcal{E}}\|$ and is therefore not optimal. Kroonenberg and de Leeuw (1980) suggested an iterative procedure called Higher-Order Orthogonal Iteration (HOOI) that has proven to yield efficient solutions in most applications.[5] They show that one $\hat{\mathbf{V}}^{(i)}$ can be solved for if the other $\hat{\mathbf{V}}^{(j)}$

[5]HOOI is also known as Alternating Least Squares (ALS).

and $\hat{\mathbf{V}}^{(k)}$ are known; see Appendix A.3 for more details. Each step solves a linear problem, making HOOI computationally feasible even for large tensors. Typically, HOOI uses the HOSVD estimates as starting values and iterates until a convergence criterion is satisfied.[6] Zhang and Xia (2018) show that the HOOI algorithm is consistent and converges within a logarithmic factor of iterations. They also provide explicit upper bounds of the estimation error.

4.5. Monte Carlo simulations

Before presenting the empirical results, I perform a Monte Carlo simulation to evaluate the performance of 3D-PCA in recovering the true underlying factors and weights from simulated data. I compare the estimation accuracy of 3D-PCA to that of standard PCA. The data generating process is a tensor factor model in (13) plus an idiosyncratic error term:

$$\boldsymbol{\mathcal{X}}_e = \boldsymbol{\mathcal{X}} + \tilde{\sigma}_e\, \boldsymbol{\mathcal{E}}, \tag{25}$$

$$\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{F}} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)} \times_4 \mathbf{V}^{(Q)}, \tag{26}$$

where $\boldsymbol{\mathcal{F}}$ is a $(T{\times}K_C{\times}K_P{\times}K_Q)$-dimensional tensor of i.i.d. normal random variables. The means and standard deviations of the factors are set to match the sample properties. I set the $\mathbf{V}^{(i)}$ to their sample counterparts estimated from data (see Figure 3). $\boldsymbol{\mathcal{E}}$ is a $(T{\times}C{\times}P{\times}Q)$-dimensional tensor of i.i.d. standard normal random variables. The size of $\boldsymbol{\mathcal{X}}$ is set to match the empirical application: $C = 11$, $P = 5$, and $Q = 5$. I consider different combinations of factor numbers $(K_C, K_P, K_Q) = (2,2,2), (3,3,3), (4,2,2)$ and sample sizes $T = 25, 50, 100, 250, 750$. I set the idiosyncratic error standard deviation so that $\tilde{\sigma}_e = \sigma_e\, \sigma(\boldsymbol{\mathcal{X}})$ and consider different values of σ_e.

Recall that the matrix of factor weights implied by (26) is given by $\mathbf{W}^{3D} = \mathbf{V}^{(Q)} \otimes \mathbf{V}^{(P)} \otimes \mathbf{V}^{(C)}$ and the matrix of factors is given by $\mathbf{F}^{3D} = \mathrm{mat}_T(\boldsymbol{\mathcal{F}})$. In each simulation sample, I estimate $\hat{\boldsymbol{\mathcal{F}}}$ and $\hat{\mathbf{V}}^{(i)}$ of the 3D-PCA model using the HOOI method described in section 5.2 and compute the implied $\hat{\mathbf{W}}^{3D}$ and $\hat{\mathbf{F}}^{3D}$. I also estimate the standard PCA factor and weight matrices $\hat{\mathbf{F}}^{PC}$ and $\hat{\mathbf{W}}^{PC}$. Recall that the 3D-PCA setup has a 2-dimensional factor structure, so that the PCA is correctly specified.[7] To evaluate the estimation accuracy of the two methods, I compute the root mean squared errors (RMSE) and average correlations between the true and estimated weights and factors across the 1,000 simulations.

Table 2 reports results when the idiosyncratic error standard deviation σ_e is set to 0.15, which is close to the value estimated in the data. Table B.2 in the Appendix reports results for $\sigma_e = 0.5$. Consider first the case $(K_C, K_P, K_Q) = (2,2,2)$ for different sample sizes T. The RMSEs of the 3D-PCA estimated weight matrices $\mathbf{W}^{3D}$ and factors $\mathbf{F}^{3D}$ are substantially lower than those of the PCA estimates $\mathbf{W}^{PC}$ and $\mathbf{F}^{PC}$ across all sample sizes by an order of magnitude. The 3D-PCA estimates of $\mathbf{W}^{3D}$ are very precise even in short samples, while the RMSE of $\mathbf{F}^{3D}$ are slightly higher. However, the estimated factors are highly correlated with the true factors, with correlations above 95% for all values of T. On the other hand, PCA estimates are much less accurate. The RMSE for $T = 750$ are comparable to those of 3D-PCA with $T = 25$ for the weight matrix and $T = 50$ for factors. In addition, the correlations of the PCA estimates with the true weights and factors are also much lower than those of 3D-PCA and above 90% only for a large sample size of $T = 750$. Recall from section 4.3 that the number of free parameters in the tensor factor model is smaller than in the standard matrix factor model. For the case of $T = 100$, the RMSE of the PCA weight matrix is substantially larger than that of the 3D-PCA method: 0.037 versus 0.0004. The number of parameters in the weight matrix in 3D-PCA and PCA are 33 and 2,164, respectively.[8] The more

[6] There are other solution approaches, but HOOI remains the most widely used method; see Kolda and Bader (2009).

[7] Note that the 3D-PCA model has a standard factor model representation in two dimensions that can be estimated by PCA. However, the order of the 3D-PCA and PCA factors can differ. In the simulations, I match the correct columns of $\hat{\mathbf{W}}^{3D}$ and $\hat{\mathbf{W}}^{PC}$.

[8] The numbers of parameters of the full Tucker and SVD decompositions are 830 and 2,936, respectively.

parsimonious representation of the 3D-PCA model likely contributes to the high estimation accuracy of the 3D-PCA method. When higher-order factors are included, the estimation accuracy of both methods declines, but the 3D-PCA method continues to yield precise estimates and outperform PCA.

The DGP in (25)-(26) has a tensor structure, so that the 3D-PCA method is correctly specified. Next, I consider the case when the DGP has a 2-dimensional factor structure that does not satisfy the restrictions of the 3D-PCA model.[9] The misspecified DGP is constructed as follows. I take the $\mathbf{V}^{(C)}$ and $\mathbf{V}^{(P)}$ matrices from the previous DGP and construct the implied weight matrix $\mathbf{W}$, as before. I then add random noise to $\mathbf{W}$ to break the tensor structure:

$$\mathbf{W}_e = \mathbf{W} + \tilde{\sigma}_w \; \mathbf{E}_w \tag{27}$$

$$\mathbf{X} = \mathbf{F}\,\mathbf{W}_e^\top + \tilde{\sigma}_e \; \mathbf{E}_x, \tag{28}$$

where $\mathbf{W}_e$ is orthonormalized, so that it satisfies the standard conditions of a factor weight matrix. As for σ_e, I set the standard deviation so that $\tilde{\sigma}_w = \sigma_w \, \sigma(\mathbf{W})$ and consider different values of σ_w. Finally, I reshape $\mathbf{X}$ into a $(T{\times}C{\times}P{\times}Q)$-dimensional tensor $\boldsymbol{\mathcal{X}}$, estimate the 3D-PCA and PCA models, and compute the RMSE and correlations as before. I find that 3D-PCA continues to yield more accurate estimates than PCA, even when the DGP does not satisfy the restrictions of the 3D-PCA model unless σ_w is very large. Table 3 reports results when σ_w is set to 25%. While the precision of 3D-PCA estimates is lower than in the correctly specified case, they are still substantially more accurate than PCA estimates. The RMSEs of the 3D-PCA estimates are lower than those of PCA across all sample sizes, and the correlations with the true weights and factors are higher for 3D-PCA than for PCA. This is true even when $\sigma_w = 50\%$; see Table B.3 in the Appendix. These results show that the 3D-PCA method is robust to misspecification and can still provide useful estimates even when the underlying factor structure does not satisfy the restrictions of the model.

In summary, the Monte Carlo results demonstrate that the 3D-PCA method can accurately recover the true underlying factors and weights from tensor data, even in small samples, and it does so more effectively than PCA.

5. Empirical results

5.1. Data

The data set consists of excess returns of value-weighted double-sorted portfolios of 11 characteristics that are available on Ken French's website: book-to-market (BM), operating profitability (OP), investment (INV),[10] momentum (MOM), long- and short-term reversal (REV, SREV), accruals (AC), beta (BETA), net share issuance (NSI), daily variance (VAR), and daily residual variance (RVAR). Each set of double-sorted portfolios is based on the intersection of five size (ME) quintile portfolios and five characteristic c quintiles, where $c \in \{\text{BM, OP, ..., RVAR}\}$. The total number of portfolios is $11 \times 25 = 275$. The size and characteristic quintiles are denoted P1 to P5 and Q1 to Q5, respectively. The sorts are arranged so that the high-return portfolios are in the fifth quintile and the low-return portfolios are in the first quintile. For example, since small stocks have higher average returns than big stocks, P5 refers to the quintile of the smallest stocks, and P1 is the quintile of the largest stocks.[11] The data set has 724 monthly observations from July 1963 to October 2023. In Section 7.1, I consider a larger set of single-sorted portfolios based on 78 characteristics.

---

[9] I thank a referee for suggesting the estimation of misspecified models.
[10] INV is defined as asset growth.
[11] Thus, P1 (P5) corresponds to high (low) OP, NSI, REV, and SREV.

Figure 1 shows heatmaps of annualized mean returns of the 5-by-5 quintiles for each of the 11 characteristics. The bottom right panel shows the mean across all characteristics. Portfolios with mean returns that are higher (lower) than the overall mean return are plotted in blue (red). The highest and lowest returns are associated with the highest and lowest quintiles, although the pattern is not always monotonic. The small-high momentum portfolios MOM-P5Q5 and MOM-P4Q5 have the highest mean returns (15.2% and 14.3%, respectively) followed by the P4Q5 RVAR and VAR portfolios (13.8% and 13.6%). The RVAR and VAR P5Q1 portfolios have negative average excess returns, while the means of the SREV and MOM P5Q1 portfolios are below 1%. Note that the interaction of size and characteristic quintiles varies across characteristics. While the size effect is positive for all Q5 quintiles, it reverses in Q1 portfolios for some characteristics but not others. For example, the return of the big/low-BM portfolio (BM-P1Q1) is 6.9%, which is higher than the mean of the small/low-BM portfolio (BM-P5Q1, 3.0%). On the other hand, the returns of the AC-P5Q1 accruals portfolio are higher than those of the AC-P1Q1 portfolio.

Table 1 reports annualized means and Sharpe ratios of long-short portfolios for each characteristic. SMB_c is the difference between the small-stock portfolios (P5) and the big-stock portfolios (P1) of the double sort of characteristic c averaged over all characteristic quintiles. HML_c is constructed analogously, but is the difference of Q5 and Q1 portfolios averaged over all size quintiles. Finally, $CROSS_c$ is the average of the P1Q1 and P5Q5 portfolios minus the average of the P1Q5 and P5Q1 portfolios. SMB_c is positive for all characteristics; hence, small stocks earn, on average, higher returns than big stocks in double-sorts. However, size premia are modest, ranging from 1.30% for SREV to 3.91% for OP. The Sharpe ratios of SMB_c range from 0.09 for SREV to 0.28 for OP. The average SMB_c across all characteristics is 2.75%. For most characteristics, the premium in the characteristic dimension exceeds the size premium. The average HML_c for MOM is 10.43%, by far the largest in the sample, followed by SREV at 7.37%. In contrast, HML_c of BETA is only 0.7%. The three characteristics with the highest HML_c Sharpe ratios are MOM, SREV, and NSI. The average HML_c premium is 4.83%, about twice the average SMB_c premium.

Recall that $CROSS_c$ captures the difference between the low/low and high/high, and the low/high and high/low corner portfolios. The third column of Table 1 shows that there is large variation in mean $CROSS_c$ returns across characteristics. The reason is that the size premium inverts across the Q1 to Q5 quintiles for some characteristics but not for others. $CROSS_c$ is high for characteristics with inverted size premia and low for those without. The highest $CROSS_c$ returns of 13.70% and 13.10% are associated with VAR and RVAR, respectively, which are also the highest among all long-short portfolios shown in the table. As shown in Figure 1, the size premium is positive for their Q5 portfolios (P5Q5-P1Q5 is positive) but strongly negative for the Q1 portfolios (P5Q1-P1Q1 is negative). As a result, the return of the $CROSS_c$ portfolios is particularly high, outstripping their HML_c by a factor of three. $CROSS_c$ of SREV, BM, and MOM are also high. On the other hand, the $CROSS_c$ of AC and BETA are negative because the size premium is positive for the Q1 and Q5 portfolios. The average $CROSS_c$ premium is 5.93%. Furthermore, the highest Sharpe ratios among all long-short portfolios are for $CROSS_c$ of VAR and RVAR, at 0.78 and 0.77, respectively.

The standard deviations of all SMB_c and $CROSS_c$ portfolios are between 12% and 18% and thus comparable. In contrast, the volatilities of the HML_c vary across characteristics. The volatility of the AC HML_c is only 4.91% and thus particularly low. The reason is that the corresponding Q1 and Q5 AC portfolios are highly correlated; the mean correlation across the five size portfolios is 0.91. As a result, the volatility of the long-short portfolio is low. This pattern is also true for OP, INV, and NSI portfolios but to a lesser extent (their mean correlations are between 0.85 and 0.90). However, the correlations of the MOM and VAR Q1 to Q5 portfolios are below 0.7, so their HML_c portfolios are more volatile. The differences in the volatilities of HML_c portfolios will be important for estimating the factor models considered in the rest of the paper.

5.2. 3D-PCA estimation

I start by estimating the 3D-PCA model with different numbers of factors (K_C, K_P, K_Q) using the full sample. Since the in-sample factors are subject to look-ahead bias, I also estimate out-of-sample factors that are not. I estimate the 3D-PCA model in rolling samples of length h. For a subsample ending in $t' = h, ..., T-1$, I construct factor returns in $t'+1$ by multiplying the portfolio returns in period $t'+1$ by the factor weights from the subsample ending in t'. An important feature of 3D-PCA is its parameter parsimony, as shown above. Since it has few free parameters, it can be estimated in much shorter samples than PCA.

The 3D-PCA model approximates the data tensor with 275 portfolios when $K_C < C, K_P < P, K_Q < Q$. Therefore, I first compute the proportion of the variation in $\boldsymbol{\mathcal{X}}$ that the model captures for different combinations of (K_C, K_P, K_Q). The time series R^2 is defined as $R^2_{ts} = 1 - \|\boldsymbol{\mathcal{E}}\| / \operatorname{Var}(\boldsymbol{\mathcal{X}})$, where $\boldsymbol{\mathcal{E}}$ is the approximation error in (23). I estimate 3D-PCA models for all combinations of $K_C \in \{1, 2, \dots, C\}, K_P \in \{1, 2, \dots, P\}, K_Q \in \{1, 2, \dots, Q\}$. The total number of factors of a given model is $K = K_C K_P K_Q$. Figure 2 shows the heatmap of the R^2_{ts} with the number of characteristic factors K_C on the x-axis and combinations of size and characteristic quintile factors (K_P, K_Q) on the y-axis. Similar to PCA models, 3D-PCA models with few factors capture a large proportion of the data's variation. The model with a single factor in each dimension, $K_C = K_P = K_Q = 1$, in the upper-left cell, captures 82.6% of the data variation. Increasing K_C while keeping $K_P = K_Q = 1$ has a marginal impact on the R^2_{ts}, suggesting that higher values of (K_P, K_Q) are required. The most parsimonious models with an R^2_{ts} of at least 90% and 95% are $K_C = K_P = K_Q = 2$ with a total of 8 factors and $K_C = 5, K_P = K_Q = 3$ with 45 factors, respectively. Increasing the number of factors improves the fit only marginally.

The rest of this section is organized as follows. I first investigate the structure and properties of the estimated factors and factor weights in full-sample and rolling-sample settings, followed by the asset-pricing results for 3D-PCA and PCA-based factor models.

5.3. Factor weights

The estimation of the 3D-PCA model yields the $(T \times K_C \times K_P \times K_Q)$-dimensional tensor of factors $\boldsymbol{\mathcal{F}}$ and three matrices $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$ of dimensions $(K_C \times C), (K_P \times P)$, and $(K_Q \times Q)$, respectively. Recall that the weights that generate the factors are determined by the transposes of $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$. I first present the in-sample estimates of the three $\mathbf{V}$ matrices and then study the implications for factor weights. Weights derived from the rolling estimation are discussed in the following subsection. The empirical results in Sections 6.1 to 6.2 below show that the first three factors, C, P, and Q, are relevant for asset pricing. Hence, I estimate the 3D-PCA model with $K_C, K_P, K_Q = 3$ as the benchmark specification.

Figure 3 shows the estimated matrices $\hat{\mathbf{V}}^{(i)}$ as heatmaps with the point estimates displayed in each cell. Positive values are in blue, and negative values are in red. While the columns are orthogonal and have unit norm, their signs are not identified. I set the signs so that the factor interpretations align with standard portfolio sorts and risk factors whenever possible. The rows and columns of $\hat{\mathbf{V}}^{(C)}$ in Panel A correspond to the 11 characteristics and C-factors, $K_C = 1, 2, 3$. All elements of the first column, $\mathbf{v}_1^{(C)}$, are positive and represent "long-only" portfolio weights. The weights range from 0.29 to 0.31, thus similar across characteristics, resulting in an almost equal-weighting scheme. The second and third columns contain positive and negative values and therefore correspond to "long-short" weights. The largest positive weights of the second factor, $\mathbf{v}_2^{(C)}$, are for VAR and RVAR, while the weight of BETA is by far the lowest, followed by SREV, REV, and AC. The elements of the other characteristics are small. Hence, the second factor has the interpretation of an (approximately) long-VAR and RVAR, with a short-BETA weighting scheme. The third factor, $\mathbf{v}_3^{(C)}$, is dominated by MOM (0.88), followed by BM (-0.36) and NSI (-0.25), and can be interpreted as a long-MOM, short-BM factor.

The characteristic weights of the second and third factors are related to the volatilities of long-short portfolios. Results in Table 1 show that volatilities of SMB_c portfolios by characteristic are similar across characteristics; however, the volatilities of characteristic-specific HML_c portfolios vary significantly. For example, the HML_c portfolio created by a size/accruals double-sort has a volatility of 4.91%, whereas HML_c for several other characteristics has a standard deviation over 15%. In fact, the characteristics with the highest HML_c volatilities have the highest weights in $\mathbf{v}_2^{(C)}$ and $\mathbf{v}_3^{(C)}$ (in absolute value), namely, VAR (19.55%), RVAR (17.93%), MOM (17.22%), and BETA (16.86%), followed by SREV (12.96%) and BM (12.59%). Of course, factor weights depend on the complex correlation structure of the test assets; however, we will see below that factors with high-minus-low structures play an essential role.

The $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ matrices determine factor weights along the size and characteristic dimensions, respectively. Both follow similar patterns that have a familiar structure. The first P-factor, $\mathbf{v}_1^{(P)}$, is long-only, but the weights increase in P-quintiles, so that the corresponding factor overweights the small-stock quintiles and underweights the big-stock quintiles. $\mathbf{v}_2^{(P)}$ is monotonically increasing across size quintiles, ranging from -0.66 for the P1 quintile to 0.62 for the P5 quintile, and therefore exhibits a small-minus-big pattern. In contrast, the P1 and P5 elements of $\mathbf{v}_3^{(P)}$ are positive, while the P2, P3, and P4 elements are negative, creating a long-short/short-medium pattern.

The shape of the $\mathbf{V}^{(Q)}$ matrix is similar to that of $\mathbf{V}^{(P)}$. The first Q-factor, $\mathbf{v}_1^{(Q)}$, is positive and thus long-only. The second column, $\mathbf{v}_2^{(Q)}$, increases from Q1 to Q5 and represents a high-minus-low factor. Like the third P-factor, the third Q-factor is positive in Q1 and Q5 and negative in Q2, Q3, and Q4. The columns of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ therefore have a level-slope-curvature pattern often seen in PCA applications. The first "level" factor is long-only and has approximately similar values; the second factor has a "slope" pattern that is monotonically increasing across the quintile portfolios; and the third factor is convex and resembles "curvature".

Table 4 reports bootstrapped standard errors of the estimated weight matrices $\mathbf{V}^{(i)}$. Since columns of these matrices are normalized to have unit length but have different numbers of rows, I express the standard errors as percentages of the average absolute value of the elements of the corresponding column, which is equal to the inverse of the t-statistics. First, note that lower-order factors, e.g., $\mathbf{v}_1^{(i)}$, are estimated more precisely than higher-order factors $\mathbf{v}_3^{(i)}$, as is the case for standard PCA estimations. Second, the columns of $\mathbf{V}^{(Q)}$ are estimated with the greatest precision, with standard errors of less than 1% of the average estimated weight: 0.001, 0.004, and 0.007. The standard errors of $\mathbf{V}^{(P)}$ are only slightly higher at 0.002, 0.009, and 0.023. Third, the standard errors of the first two columns of $\mathbf{V}^{(C)}$ are 0.002 and 0.024, and are also small relative to the average estimated weights. The weights in the third column of $\mathbf{V}^{(C)}$ have the largest standard errors at 0.121. The mean standard error across all $\mathbf{v}_j^{(i)}$ columns is 2.02% of the average element of the three $\mathbf{V}^{(i)}$ matrices. In contrast, PCA factor weights are substantially noisier. The mean of the standard errors of the PCA weight matrix $\mathbf{W}^{PC}$ is 70.53% of its average element, which is larger than the average standard error of 3D-PCA weights by a factor of 33. These results are consistent with the Monte Carlo simulations in Section 4.5, which show that 3D-PCA weights are estimated significantly more precisely than PCA weights.

As mentioned above, the estimated weight matrices $\mathbf{V}^{(i)}$ can differ in models with different numbers of factors. However, in most applications, the difference is small. Figure C.6 in the Appendix shows the first three columns of the estimated $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$ for the exact 3D-PCA model with $(K_C, K_P, K_Q) = (C, P, Q)$. Without exception, the elements of the two weight matrices are very close and follow the same patterns. I verified that the same holds for other specifications of K_C, K_P, and K_Q. Moreover, none of the results in the paper are sensitive to the choice of the number of factors. Hence, the results are robust to the choice of K_C, K_P, K_Q as long as they are large enough to capture the main patterns in the data.

Section 4 showed that the columns of $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$, and $\mathbf{V}^{(Q)}$ form building blocks for factor weights.

Next, I study the portfolios that can be constructed from the estimated $\hat{\mathbf{V}}^{(i)}$. Consider first the P and Q-dimensions. The outer product of the p-th and q-th columns of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ is the (5×5) matrix $\mathbf{W}_{pq}^{(PQ)} = \mathbf{v}_p^{(P)} \circ \mathbf{v}_q^{(Q)}$. The elements of $\mathbf{W}_{pq}^{(PQ)}$ are weights for the 25 possible $(P\times Q)$ combinations of quintile portfolios. Since $K_P = K_Q = 3$, there are nine possible combinations of the columns of $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$. Each of the nine combinations represents a different weighting scheme for the P and Q dimensions, constructed from the estimates $\hat{\mathbf{V}}^{(P)}$ and $\hat{\mathbf{V}}^{(Q)}$.

Figure 4 shows all nine $\mathbf{W}_{pq}^{(PQ)}$ matrices that are generated by the estimated $\hat{\mathbf{V}}^{(P)}$ and $\hat{\mathbf{V}}^{(Q)}$ shown in Figure 3. The first matrix $\mathbf{W}_{11}^{(PQ)}$ combines the first columns $\mathbf{v}_1^{(P)}$ and $\mathbf{v}_1^{(Q)}$. Since both columns are long-only, $\mathbf{W}_{11}^{(PQ)}$ is a positive matrix. Since $\mathbf{v}_1^{(P)}$ underweights P1, the P1 weights of $\mathbf{W}_{11}^{(PQ)}$ are somewhat lower than the P2 to P5 weights. Next, consider combining the first column of $\hat{\mathbf{V}}^{(P)}$ and the second column of $\hat{\mathbf{V}}^{(Q)}$. $\mathbf{v}_1^{(P)}$ is long-only, while $\mathbf{v}_2^{(Q)}$ is monotonic in the Q1 to Q5 quintiles. Hence, the columns of $\mathbf{W}_{12}^{(PQ)}$ (Panel B) retain the monotonic structure in the Q-dimension while its rows, representing the P-dimension, are almost identical. The smaller values (in absolute terms) of the P1 row are again due to the underweighting of the P1 portfolio in $\mathbf{v}_1^{(P)}$. Therefore, $\mathbf{W}_{12}^{(PQ)}$ has a high-minus-low structure that is long in the high-characteristic quintile Q5 and short in the low-characteristic quintile Q1, combined with equal weighting of the size dimension. By the same token, the outer product of the second P-factor and the first Q-factor, $\mathbf{W}_{21}^{(PQ)}$, has a similar structure but with the P and Q dimensions reversed since $\mathbf{v}_2^{(P)}$ is monotonic in P1 to P5 and $\mathbf{v}_1^{(Q)}$ is long-only. Thus, $\mathbf{W}_{21}^{(PQ)}$ has a small-minus-big pattern with (approximately) equal weights in the Q dimension. In other words, $\mathbf{W}_{12}^{(PQ)}$ and $\mathbf{W}_{21}^{(PQ)}$ are similar to HML and SMB but are estimated rather than constructed from fixed weights. Combining the first and third columns of $\hat{\mathbf{V}}^{(P)}$ and $\hat{\mathbf{V}}^{(Q)}$ generates $\mathbf{W}_{13}^{(PQ)}$, shown in Panel C, which retains the approximately equal weighting in the P dimension with the convex structure of $\mathbf{v}_3^{(Q)}$. Hence, all P1-P5 rows have positive Q1 and Q5 weights and negative Q2, Q3, and Q4 weights. As for $\mathbf{W}_{21}^{(PQ)}$, the pattern is reversed for $\mathbf{W}_{31}^{(PQ)}$, so that each column has the curvature pattern in the size dimension.

The $\mathbf{W}_{pq}^{(PQ)}$ weight matrices mentioned so far combine a "long-only" vector with a vector in the other dimension that has a level, slope, or curvature structure. Thus, all portfolios that can be generated retain a level, slope, or curvature structure in one dimension, with approximately equal weighting in the other. The outer product of the second columns, $\mathbf{v}_2^{(P)}$ and $\mathbf{v}_2^{(Q)}$, combines two monotonic (slope) vectors. The resulting matrix $\mathbf{W}_{22}^{(PQ)}$ is shown in Panel E. The weights of the diagonal corner portfolios P1Q1 and P5Q5 are positive, while the weights of the off-diagonal corner portfolios P1Q5 and P5Q1 are negative. The other elements of $\mathbf{W}_{22}^{(PQ)}$ are an order of magnitude smaller. Hence, the pattern is similar to the CROSS portfolios introduced in Section 5.1 (see Table 1).

Since the $\mathbf{W}_{pq}^{(PQ)}$ are based on combinations of the same column vectors, they share common structures that combine the level, slope, and curvature patterns of those vectors. Hence, all $\mathbf{W}_{pq}^{(PQ)}$ have straightforward economic interpretations. Moreover, some weight matrices exhibit patterns that resemble long-short portfolios studied in the literature. Rather than using ad hoc weights, the weights are estimated from the data.

To obtain weights for all $N = 275$ portfolios, the outer product of an (11×1)-dimensional column vector $\mathbf{v}_i^{(C)}$ and a (5×5)-dimensional matrix $\mathbf{W}_{pq}^{(PQ)}$ yields an $(11\times5\times5)$-dimensional tensor of weights $\boldsymbol{\mathcal{W}}_{ijk}$. Since $K_C = K_P = K_Q = 3$, each of the 27 combinations of $i, j, k = 1, 2, 3$ represents the weight tensor $\boldsymbol{\mathcal{W}}_{ijk}$ of a factor F_{ijk}^{3D}. Figure 5 shows a subset of rows of $\mathbf{W}^{3D}$ as a heatmap. I plot weights of the P1Q1, P1Q3, P1Q5, P3Q1, P3Q3, P3Q5, P5Q1, P5Q3, and P5Q5 portfolios of BM, MOM, REV, SREV, NSI, VAR, and RVAR. Each column of the heatmap corresponds to the weights of a factor F_{ijk}^{3D}. Positive weights are in blue, and negative weights are in red. While the heatmap is complex, common structures emerge from the figure. The first nine columns show factors $F_{1\bullet\bullet}^{3D}$ that are based on the first column $\mathbf{v}_1^{(C)}$, the next nine columns correspond to factors generated by $\mathbf{v}_2^{(C)}$, and the last nine columns are computed from $\mathbf{v}_3^{(C)}$. The heatmap shows that factors $F^{3D}_{2\bullet\bullet}$ are dominated by VAR, RVAR, and BETA, while factors $F^{3D}_{3\bullet\bullet}$ depend on MOM, BM, and NSI.

By the same token, factors $F^{3D}_{\bullet 12}$ and $F^{3D}_{\bullet 21}$ retain the HML and SMB structures of $\mathbf{W}^{(PQ)}_{12}$ and $\mathbf{W}^{(PQ)}_{21}$. For example, the Q5 weights of $F^{3D}_{112}, F^{3D}_{212}$ and F^{3D}_{312} are positive, the Q3 weights are close to 0, and the Q1 weights are negative. Similarly, $F^{3D}_{121}, F^{3D}_{221}$ and F^{3D}_{321} have the same structure but for size quintiles P5, P3, and P1. The other factors have similar interpretations.

For comparison, Figure 6 plots the corresponding weight matrix derived from a PCA estimation with 27 factors. Aside from the first long-only factor, PCA weights show little common structure. For example, factors are not related to particular characteristics but involve portfolios of many characteristics. In contrast, 3D-PCA factors are associated with all characteristics ($F^{3D}_{1\bullet\bullet}$) or small subsets of characteristics ($F^{3D}_{2\bullet\bullet}$ and $F^{3D}_{3\bullet\bullet}$). Moreover, the weights of most PCA factors do not have clear patterns across size and characteristic quintiles. Figure C.7 shows heatmaps of the first nine PCA factor weights. Only two factors have easily interpretable patterns.[12] The weights of the second and fifth factors are related to size, but none of the factors is related to the Q dimension. F^{PC}_{2} reflects big-minus-small since its weights are positive for P1 and negative for P5 quintiles. The weights of the fifth factor are positive for the P1 and P5 quintiles and negative for the P3 quintile, creating a "size-curvature" pattern.

Recall from (22) that the contribution of a 3D-PCA factor to the overall variation in the data is given by its sum of squares, $\sum_t F^2_{cpq,t}$. Table 5 reports the sum of squares for the 27 3D-PCA factors. Consider first the results for in-sample factors in Panel A. As in many PCA applications, the first factor accounts for most of the variation in the data. The sum of squares of F^{3D}_{111} is 573.92, an order of magnitude larger than the second-largest sum of squares, 28.52 for F^{3D}_{121}. It is instructive to compare the two factors that are created by slope factors in the P and Q dimensions with the level factors in the other dimensions, i.e., F^{3D}_{121}, which has a small-minus-big structure, and F^{3D}_{112}, which has a high-minus-low structure (see Figure 3). The sum of squares of F^{3D}_{121} is 28.52, considerably higher than that of F^{3D}_{112} (5.45). Therefore, F^{3D}_{121} captures a higher portion of the data variation than F^{3D}_{112}.

5.4. Factor weights in rolling samples

In the out-of-sample estimation, factor models are estimated in rolling windows with h data points. Figure 7 plots the factor weights of the first four 3D-PCA (left column) and PCA (right column) factors for the benchmark case of $h = 24$. Figures C.8, C.9, and C.10 in the Appendix show results for $h = 6, 12, 60$. Each panel shows the time series of the weights of the P1Q1, P1Q5, P5Q1, and P5Q5 portfolios for each of the eleven characteristics. The weights of the four portfolios are color-coded as indicated in the legends. For example, the black lines show the weights of the P5Q5 portfolios over time. The average time-series standard deviations of the weights across all 27 factors are reported in the top-left corner of each panel.

Consider first the F^{3D}_{111} factor in Panel A. For the full-sample estimation, the weights are positive for all portfolios and characteristics. The weights of the P1Q1 portfolios (in red) are the highest, followed by the P1Q5 and P5Q1 portfolios. The P5Q5 portfolios (in black) have the lowest weights. This pattern is consistent with the level structure of $\mathbf{W}^{(PQ)}_{11}$ shown in Figure 4, and the mean weights in rolling samples are close to the full-sample estimates. Moreover, the estimates of the weights are stable over time; however, they "fan" out in the early 2000s, which coincides with a sharp decline in the correlations of portfolio returns; see below. The behaviors of the weights of the $F^{3D}_{121}, F^{3D}_{112}$, and F^{3D}_{122} 3D-PCA factors in Panels C, E, and G are similar. The weights are stable and show little variation over time. Figure C.9 in the Appendix shows that the 3D-PCA weights are also stable for windows with only $h = 12$ time series observations.

[12] Since the signs of PCA factor weights are not identified, I normalize the signs so that the mean returns of PCA factors are positive.

The weights of the first four PCA factors are plotted in the right column of Figure 7. In contrast to the 3D-PCA weights, the PCA factor weights are substantially more unstable over time. Their standard deviations (reported in the top-left corner of each panel) range from 1.48% to 7.41%, compared to between 1.07% and 2.82% for 3D-PCA factors. Moreover, they are less autocorrelated and often change signs. The weights in longer windows of $h = 60$ months shown in Figure C.10 are only slightly more stable.

Next, I compare the stability of 3D-PCA and PCA weights across different window lengths. Figure 8 plots the average standard deviation and autocorrelations of the weights across all factors for different window lengths h. The blue line in Panel A shows the average standard deviation of all $K = 27$ 3D-PCA factor weights. Standard deviations of PCA weights are plotted in orange. Note that if $h < 27$, the number of PCA parameters exceeds the number of observations, so that only h factors are identified. Since 3D-PCA has fewer parameters, all 27 3D-PCA factors are always identified.[13] The standard deviations of 3D-PCA weights decrease roughly linearly with the window length h. For the benchmark case of $h = 24$, the average standard deviation of 3D-PCA weights is 2.83%.

It may be surprising that the standard deviations of the PCA weights do not decrease as the window length increases when $h < 36$. This is because lower-order factors are more precisely estimated than higher-order factors. As mentioned above, only the first h PCA factors are identified for $h < 36$. Hence, the average standard deviation of PCA weights is computed from a different set of factors for each window length, and since lower-order factors are more precisely estimated, PCA weights appear more stable for shorter windows. To facilitate an appropriate comparison, the dashed blue line in Figure 8 shows the average standard deviation of the 3D-PCA weights for only the first h factors, for $h < 27$.

The average volatility of PCA weights is substantially higher than that of 3D-PCA weights across all window lengths. For example, the average standard deviation of PCA weights for $h = 24$ is 5.48%, which is about twice that of 3D-PCA weights. Strikingly, the standard deviations of 3D-PCA weights for a window length of six months are lower than those of PCA weights for $h = 240$ months. The advantage of 3D-PCA seems to decrease for low h; however, this is due to the fact that only the first h PCA factors are identified for $h < 27$, as mentioned above. Once the same set of factors is compared, 3D-PCA weights (dashed blue) retain the same advantage as for higher h.

Another measure of stability is the autocorrelation of the weights, which is plotted in Panel B of Figure 8. Since adjacent rolling windows overlap by $h - 1$ months, their factor weights should be similar if the estimation is stable, implying high autocorrelations. The comparison of autocorrelations is even starker than the comparison of standard deviations. The average autocorrelation of the 3D-PCA weights, shown in blue, exceeds 0.9 for $h \geq 24$. Even for $h = 6$, the average autocorrelation of 3D-PCA weights is 0.67, indicating that the 3D-PCA estimation is stable and yields reliable estimates even in short windows. In contrast, average autocorrelations of PCA weights are below 0.5 for $h \leq 36$ and reach 0.8 only for very long windows of $h = 180$ and $h = 240$.

As noted above, the period between 2000 and 2010 showed unusual behavior of subsample weights in Figure 7. To investigate this behavior, I compute pairwise correlations of all $N = 275$ portfolios in rolling windows of length 24. Until the early 1990s, the average correlation was above 0.8, then declined to between 0.7 and 0.8. In early 2000, there is a precipitous drop to below 0.5, followed by a rise until the end of the sample. See Panel A of Figure C.11 in the Appendix for a plot of the mean correlation across time. The decline in the average correlation coincides with the fanning out of portfolio weights in subsamples that include the early 2000s. To pinpoint changes in return behavior, I compute the cross-sectional dispersion measured as the cross-sectional standard deviation of portfolio returns each month. The time series of dispersion is plotted in Panel B. Dispersion of portfolio returns spikes in February

[13] For example, for $h = 6\,(24)$, 3D-PCA has 198 (684) free parameters, and each rolling sample has 1,650 (6,600) observations, whereas PCA with 27 factors has 6,858 (7,344) parameters.

2000 to 12.33, by far the highest value in the sample. The reason for this spike is that many small stock portfolios had substantially higher returns in February 2000 than big stock portfolios (e.g., returns of seven SMB_c were above 25%), while HML_c of several characteristics had returns below -20% (e.g., the HML_c of VAR and RVAR were -29.78% and -29.65%, respectively). While such a large spread is unprecedented in the sample, the return dispersion remains relatively high until December 2001. This unusual behavior of returns is reflected in the factor weights of the rolling samples that include the early 2000s, which show more variation than in other periods.

In summary, the 3D-PCA weights are stable over time and show little variation across rolling samples even in very short windows. In contrast, comparable PCA weights are substantially more volatile and often change signs. Long windows are required to achieve stability comparable to 3D-PCA. These empirical results are consistent with the Monte Carlo simulations in Section 4.5, which demonstrated that the 3D-PCA factors and weights are more precisely estimated than those of PCA in short samples.

6. Asset pricing results

This section focuses on the asset pricing implications of 3D-PCA factors. I first present the Sharpe ratios of individual 3D-PCA factors, followed by the Sharpe ratio that is generated by all factors in a given model. I compare results for 3D-PCA factors with those for PCA factors based on the complete set of 275 portfolios and on the 22 long-short SMB_c and HML_c portfolios (labeled PCA-LS). I then analyze the cross-sectional fit of 3D-PCA factors and compare them with the fit of PCA factors. Finally, I compare the results with those of machine learning methods that are estimated using firm-level data.[14] The benchmark 3D-PCA model is based on $(K_C, K_P, K_Q) = (3,3,3)$ factors but I also consider more parsimonious specifications. Additional factors do not significantly improve the results.

6.1. Factor returns

For the estimation of factor returns, I set the benchmark window length for out-of-sample estimations to $h = 24$ months. Figure 9 plots the 10 in-sample and out-of-sample 3D-PCA and PCA factors with the highest annualized Sharpe ratios (in absolute value). Note that the annualized Sharpe ratio of the CRSP-VW index over the sample period is 0.42.[15] Consider first the in-sample 3D-PCA factors in Panel A (in blue). The 3D-PCA model with $(K_C, K_P, K_Q) = (3,3,3)$ has 27 factors, which are denoted as F^{3D}_{cpq} for $c, p, q \in \{1,2,3\}$. Factor with a $(-)$ superscript have negative Sharpe ratios. In standard PCA applications, it is often difficult to interpret the factors. In contrast, 3D-PCA factors have a clear economic interpretation, as discussed in section 5.3. In particular, the factor weights in the characteristic dimension (Panel A of Figure 3) and the (P, Q) dimensions (Figure 4) are helpful to understand the factor structures and interpretation of 3D-PCA factors.

The four factors with the highest Sharpe ratios, F^{3D}_{112}, F^{3D}_{122}, F^{3D}_{113}, and F^{3D}_{123}, are based on the first column of $\hat{\mathbf{V}}^{(C)}$ ($\mathbf{v}^C_1$ in Panel A of Figure 3), which implies an (approximately) equal weighting of all characteristics, or a "level" pattern in the characteristic dimension. Factor F^{3D}_{112} has the highest Sharpe ratio of 1.23, which is almost three times that of the CRSP-VW index. F^{3D}_{112} combines the first column of $\hat{\mathbf{V}}^{(P)}$ $(P = 1)$ and the second column of $\hat{\mathbf{V}}^{(Q)}$, $(Q = 2)$, corresponding to "level" in the size dimension and "slope" in the characteristic dimension. The corresponding (P, Q) weights are shown in Panel B of Figure 4. Putting the three dimensions together yields a level $\times$ level $\times$ slope pattern so that F^{3D}_{112} has the interpretation of a weighted HML factor across all characteristics and size portfolios. Correspondingly, the portfolios with

[14] Previous versions of this paper also included RP-PCA factors suggested by Lettau and Pelger (2020a,b). However, the improvement over standard PCA is minor, and thus I do not report results for RP-PCA factors for brevity.

[15] The Sharpe ratios of the SMB, HML, RMW, CMA, and MOM factors are 0.19, 0.34, 0.45, 0.45, and 0.49, respectively.

the highest (smallest) factor weights are Q5 (Q1) portfolios, see Table B.5. Of course, the difference from Fama-French factors is that the weights are estimated and combine all (P, Q) quintiles.

F^{3D}_{122} has a Sharpe ratio of 0.90 and combines the second columns of $\hat{\mathbf{V}}^{(P)}$, $P=2$, and $\hat{\mathbf{V}}^{(Q)}$, $Q=2$, creating a "cross" factor that is long in the P1Q1 and P5Q5 quintiles and short in P1Q5 and P5Q1 quintiles, as shown in Panel E of Figure 4. Thus, the weights of F^{3D}_{122} have a level $\times$ slope $\times$ slope structure. Why does this factor have a high Sharpe ratio? Table 1 shows that Fama-French-style"cross" factors have positive returns and high Sharpe ratios for most characteristics.[16] Hence, it is not surprising that a factor that has a similar structure has a positive mean return and a high Sharpe ratio.

The Sharpe ratio of F^{3D}_{113} is -0.72. F^{3D}_{113} combines the first column of $\hat{\mathbf{V}}^{(P)}$ and the third column of $\hat{\mathbf{V}}^{(Q)}$ to create a "curvature" factor that is long in the Q1 and Q5 quintiles and short in Q2, Q3, and Q4 quintiles (Panel C of Figure 4) following a level $\times$ level $\times$ curvature pattern. To investigate why this factor has a negative mean return and a high Sharpe ratio (in absolute value), I compute the mean returns of curvature factors for each characteristic.[17] Table B.4 in Appendix B shows that curvature factors have negative returns across all characteristics with Sharpe ratios (in absolute values) for some characteristics.[18] As a consequence, F^{3D}_{113} has a negative mean return and a negative Sharpe ratio that is high in absolute value. The other factors can be interpreted similarly, and I do not discuss them in detail for brevity.

The Sharpe ratios of PCA factors are plotted in orange in Figure 9. None of the Sharpe ratios are as high as those of the two 3D-PCA factors with the highest Sharpe ratios. The 16th, third, and seventh PCA factors have Sharpe ratios of 0.71, 0.71, and 0.7, respectively. The Sharpe ratios of the other factors are comparable to those of 3D-PCA factors. It is difficult to detect patterns in the weights of PCA factors (see Figure 6). Table B.6 shows the portfolios with the largest and smallest weights of the PCA factors. The only factor with a clear interpretation is F^{PC}_{7}, which is long in Q5 and short in Q1 SREV portfolios. The other PCA factors have a mixture of characteristics and quintiles, making it difficult to interpret them economically. The Sharpe ratios based on PCA for the 22 long-short SMB_c and HML_c portfolios (PCA-LS) are plotted in black. The highest Sharpe ratio is 1.06 for F^{LS}_{5}, but the subsequent factors have lower Sharpe ratios than the 3D-PCA and PCA factors.

Sharpe ratios of out-of-sample factors estimated in windows of length $h = 48$ are plotted in Panel B of Figure 9. On one hand, estimating factors in rolling windows increases estimation noise since the samples are shorter. On the other hand, it may incorporate any variation in the correlation structure over time that is not captured by the in-sample factors estimated in the entire sample, which may increase Sharpe ratios. Consider first the results for 3D-PCA factors plotted in blue. First, note that factors that have high in-sample Sharpe ratios also have high out-of-sample Sharpe ratios, as nine of the top-10 factors coincide. Second, the Sharpe ratios of the out-of-sample 3D-PCA factors are comparable to their in-sample counterparts. For example, the out-of-sample Sharpe ratios of F^{3D}_{112}, F^{3D}_{122}, and F^{3D}_{113} are 1.21, 0.69, and -0.83, respectively. Third, the out-of-sample factors are highly correlated with their in-sample counterparts, with correlations above 0.85, as reported in Table 6. This evidence suggests that the out-of-sample 3D-PCA estimation of factors yields stable and robust results. Strikingly, the estimation provides reliable results even for shorter windows. Panel C of Figure 9 plots the top-10 Sharpe ratios of out-of-sample 3D-PCA factors for window lengths of $h = 3, 6, 24$, and 60 months. The Sharpe ratios are close to their in-sample counterparts even when the factors are estimated with as few as six months of data. Even for $h = 3$, the Sharpe ratios are still reasonably high, confirming that 3D-PCA can be estimated in very short samples.

[16] Cross factors are the returns of the (P1Q1+P5Q5−(P1Q5+P5Q1) portfolios for each characteristic.

[17] For size-curvature I compute (P1Qi+P5Qi)/2−P3Qi) for each $i = 1..., 5$ and then take the mean across i. For characteristic-curvature I compute (PiQ1+PiQ5)/2−PiQ3) for each $i = 1..., 5$ and then take the mean across i.

[18] The Sharpe ratios of HML curvature factors of NSI, RVAR, and VAR are -0.86, -0.69, and -0.64, respectively.

In contrast, the out-of-sample PCA factors have substantially lower Sharpe ratios than their in-sample counterparts. The highest out-of-sample Sharpe ratio is 0.45 for F_1^{PC}, which is the only PCA factor with a Sharpe ratio above 0.4. Furthermore, the ranking of the high-SR factors is different from the in-sample ranking, and the correlations of in-sample and out-of-sample PCA factors are low for most factors, as reported in Table 6. The results for PCA-LS factors are similar.

The results presented in this section can be summarized as follows. First, 3D-PCA factors have higher Sharpe ratios than factors estimated by standard PCA. Second, 3D-PCA estimations yield stable and robust results even in very short samples. Third, 3D-PCA factors have clear economic interpretations and can be related to well-known factors such as SMB, HML, and curvature factors. Next, I analyze the Sharpe ratios of mean-variance optimal portfolios based on the factor models.

6.2. Sharpe ratios of factor models

In this section, I analyze the maximal Sharpe ratios that are implied by 3D-PCA and PCA factors. Given a set of in-sample factors, I compute the mean-variance optimal portfolio weights and the corresponding Sharpe ratio using the unconditional factor mean and covariance matrix. For out-of-sample factors estimated in rolling samples, I compute the optimal mean-variance weights in each rolling window. For a subsample ending in month t, I compute the optimal weights based on the factors estimated in the previous h months.[19] I then apply the weights to the factor returns in month $t+1$ to obtain the out-of-sample realized return of the mean-variance optimal portfolio. I then roll the window forward by one month and repeat the procedure until the end of the sample, yielding a time series of out-of-sample mean-variance optimal returns. Since mean-variance optimization requires reasonably long samples to obtain stable estimates, I choose a benchmark window length of $h = 48$ months for the out-of-sample estimations, but I also consider other window lengths. For comparison, the annualized Sharpe ratio of the CRSP-VW index is 0.42, and the in-sample and out-of-sample Sharpe ratios of the 6-factor Fama-French model are 1.17 and 0.89, respectively.

Panel A of Table 7 reports Sharpe ratios of in-sample 3D-PCA factors for combinations of (K_C, K_P, K_Q). Consider first the marginal effects of the number of factors in each dimension. Sharpe ratios for specifications with $K_Q = 1$ are well below one, indicating that at least two Q-factors are required to achieve high Sharpe ratios. Compared with the specification with $K_Q = 2$, adding the third characteristic factor ($K_Q = 3$) has a smaller effect, highlighting the importance of the second characteristic, the "slope" factor. Increasing K_P from 1 to 2 improves most Sharpe ratios more than adding a third size factor ($K_P = 3$), while the effects of increasing K_C from 1 to 2 are about the same as those of increasing K_C from 2 to 3. Note that all specifications with $K_P, K_Q \geq 2$ have Sharpe ratios in excess of 1.5. Hence, the results do not rely on a particular choice of (K_C, K_P, K_Q), but rather indicate that the 3D-PCA factors are robust to different specifications. The Sharpe ratio of the benchmark specification with $(K_C, K_P, K_Q) = (3, 3, 3)$ is 2.30, which is more than twice that of the 5-factor Fama-French model. However, more parsimonious specifications with fewer factors also yield high Sharpe ratios. For example, the specifications with six factors, $(K_C, K_P, K_Q) = (1, 2, 3)$, and nine factor, $(K_C, K_P, K_Q) = (1, 3, 3)$, have Sharpe ratios of 1.77 and 1.91, respectively.

Of course, the Sharpe ratios of in-sample factors are subject to look-ahead bias. Panel B reports results for out-of-sample factors estimated in rolling windows. I choose a benchmark window length of $h = 48$ months but report results for different choices of h below. The out-of-sample Sharpe ratios are lower than their in-sample counterparts, but the differences are relatively small, and in some cases the out-of-sample

[19] The optimal weights are computed using a shrinkage covariance matrix, with the shrinkage parameter selected by the method proposed by Chen, Wiesel, Eldar, and Hero (2010). Given the covariance matrix, I follow Kozak, Nagel, and Santosh (2020) and estimate weights using l_1 and l_2 regularizations.

Sharpe ratio is higher than the in-sample counterpart. In contrast to the in-sample results, increasing K_C beyond 1 lowers the Sharpe ratios for most specifications, while increasing K_P and K_Q has similar effects as in the in-sample case. The highest Sharpe ratio is 1.94 for the specification with $(K_C, K_P, K_Q) = (1,3,3)$ and $K = 9$ factors. Adding additional factors does not significantly improve the Sharpe ratios. For example, the Sharpe ratio of the benchmark specification with $(K_C, K_P, K_Q) = (3,3,3)$ and $K = 27$ factors is 1.89. As for the in-sample case, Sharpe ratios are also high for more parsimonious specifications with fewer factors. For example, the specifications with $(K_C, K_P, K_Q) = (1,2,2)$ and $(K_C, K_P, K_Q) = (1,3,2)$ have Sharpe ratios of 1.63 and 1.83, respectively, with $K = 4$ and $K = 6$ factors. As I will discuss later, these out-of-sample Sharpe ratios compare favorably with those reported in the literature for machine learning methods with many more characteristics.[20]

The results so far are based on models that include all $K = K_C K_P K_Q$ factors. However, some of the factors could be redundant, and thus a subset of the K factors might yield similar Sharpe ratios to the model with all K factors. I use three methods to obtain "sparse" models with $L = 1, \ldots, K$ factors. First, I set the weights of the $K - L$ factors with the smallest absolute mean-variance weights in each rolling window to zero and compute the out-of-sample Sharpe ratio of the resulting portfolio. Second, I select factors based on their individual Sharpe ratios. Third, I rank factors by their contribution to the overall variation of the data, i.e., by the sums of squared F^{3D}_{cpq}, $\sum_t F^2_{cpq,t}$. This corresponds to ordering PCA factors by their eigenvalues. I then select the L factors that contribute most to the variation and estimate the model using these L factors. Panel A of Figure 10 plots results for $(K_C, K_P, K_Q) = (1,3,3)$ and shows that high Sharpe ratios can be achieved with parsimonious models with few factors. When factors are selected by their mean-variance weights (blue line), the model with only $L = 2$ factors achieves a Sharpe ratio of 1.76. Adding further factors only gradually increases the Sharpe ratio. For the other two selection methods, the Sharpe ratios are lower but still reach 1.5 with only $L = 2$ factors for SR selection and $L = 3$ for F^{3D}_{cpq} selection. These results show that the Sharpe ratios of 3D-PCA factors are driven by a few factors, and parsimonious models with as few as two factors yield Sharpe ratios close to those of the full model.

Panel B of Figure 10 compares the out-of-sample Sharpe ratios of 3D-PCA models with those of PCA models as a function of the total number of factors K. If there are several combinations of (K_C, K_P, K_Q) with the same K, the figure shows the 3D-PCA specification with the highest Sharpe ratio. The horizontal lines show the Sharpe ratios of the CRSP-VW index and the Fama-French 6-factor model as benchmarks. The highest Sharpe ratio of PCA and PCA-LS models is 0.68 for $K = 6$ factors, whereas the Sharpe ratio of the 3D-PCA model with $K = 6$ factors is 1.83. For PCA-LS, the Sharpe ratio is maximized at $K = 4$ factors with a value of 0.52. Increasing the number of factors further does not improve the Sharpe ratios of PCA and PCA-LS models. Note that the Sharpe ratio of the 6-factor Fama-French model is 0.87, higher than those of the PCA and PCA-LS models, further evidence that the PCA models yield poor out-of-sample results.

The results above are based on rolling samples with a window length of $h = 48$ months. Figure 11 shows Sharpe ratios of out-of-sample 3D-PCA, PCA, and RP-PCA factors for different window lengths h. The 3D-PCA Sharpe ratios are high, reaching 1.62 even when the rolling windows contain only 12 months of data. However, increasing the window length beyond $h = 48$ yields only small improvements. Lengthening the rolling samples improves the results of the PCA model across all 275 portfolios, but the resulting Sharpe ratios remain below 1 even for $h = 120$. The Sharpe ratios of PCA-LS models are somewhat higher but top out at 1.14, below the Sharpe ratio of 1.19 for the 6-factor Fama-French model for $h = 120$.

[20] Table B.7 reports p-values for the null hypothesis of equal Sharpe ratios for out-of-sample 3D-PCA and PCA factors with the same number of factors. The three panels report results for the Jobson and Korkie (1981) test (corrected by Memmel (2003)), and the HAC (without pre-whitening) and the Bootstrap-TS tests proposed in Ledoit and Wolf (2008) test. The null is rejected at conventional significance levels in all specifications with $K_C > 1$.

I conclude that 3D-PCA factor models generate sizable out-of-sample Sharpe ratios that reach two on an annualized basis. Sharpe ratios of this magnitude compare favorably with those reported in the literature for machine learning methods that use firm-level returns and have many more characteristics. Gu, Kelly, and Xiu (2020) survey the performance of several machine learning methods, including regularized linear models, regression trees, random forests, and neural nets. To compare the results of 3D-PCA estimates based on portfolio returns to those of machine learning methods estimated with firm-level data, I replicate the best-performing models in Gu, Kelly, and Xiu (2020) using a sample with the same 11 characteristics (plus size) as in this paper. Since machine learning methods require longer samples for estimation, I use a window length of $h = 240$ months for their out-of-sample estimations. Details are in Appendix A.5. The results are reported in Table 8. The Sharpe ratios of machine learning models range from 1.81 for the random forest model to 1.89 for a neural net with two layers. Hence, machine learning methods do not outperform the relatively simple and linear 3D-PCA models, which have a Sharpe ratio of 1.86 in the matching sample. Furthermore, returns of machine learning models are highly leptokurtic, with excess kurtosis values over 6, whereas the 3D-PCA model has an excess kurtosis of 0.71 and is therefore close to a normal distribution.

The fact that complex nonlinear machine learning methods estimated with firm-level data do not systematically outperform the relatively simple and linear 3D-PCA model based on sorted portfolios raises several questions. For example, what are the trade-offs between using firm-level and portfolio-level data? What drives the performance of machine learning methods? Are they able to capture complex nonlinear interactions in firm-level data, or do they mainly exploit linear patterns that can be captured by simpler models such as 3D-PCA? The results in this paper suggest that these questions are not as clear-cut as some of the literature suggests. Since most machine learning methods are black boxes that are difficult to interpret, answering these questions is challenging.[21]

6.3. The cross-section of returns

Although the recent literature has focused on Sharpe ratios, the original motivation for factor models was to explain the cross-section of returns. Consider a linear asset pricing model with L factors, $\mathbf{F}_t^L$. Given that 3D-PCA and PCA factors are linear combinations of excess portfolio returns, they are themselves excess returns. I therefore run N time-series regressions of excess returns of portfolios $i = 1, \dots, N$ on a constant and factor returns $\mathbf{F}_t^L$:

$$R_{i,t}^e = \alpha_i + \boldsymbol{\beta}_i^\top \mathbf{F}_t^L + u_{i,t+1}, \quad i = 1, ..., N, \tag{29}$$

where $R_{i,t}^e$ is the excess return of portfolio i in month t. The pricing error of portfolio i is the intercept α_i. I evaluate the performance of the model by the (pseudo) cross-sectional R^2:

$$R_{xs}^2 = 1 - \frac{\frac{1}{N}\sum_{i=1}^N \alpha_i^2}{\mathrm{Var}_{xs}(\bar{R}_i)}, \tag{30}$$

where $\bar{R}_i$ is the mean excess return of portfolio i and $\mathrm{Var}_{xs}(\bar{R}_i)$ is the cross-sectional variance of mean returns. Note that the pricing errors in the numerator of (30) are not demeaned, so the mean pricing error is accounted for. As a result, R_{xs}^2 is not a "proper" R^2 and need not be positive.[22]

I follow the same procedure as in the previous section and start by computing pricing errors of 3D-PCA models for all 27 combinations of $K_C, K_P, K_Q \in \{1, 2, 3\}$. The results are reported in Table 9. Consider

[21] In ongoing work, Bharadwaj and Lettau (2026) explore these issues in more detail.

[22] The null hypothesis that all pricing errors are zero is rejected by the GRS test in all cross-sectional models considered in this paper. It is well-known that the GRS test typically rejects models even when pricing errors are economically small and is therefore not reported in most of the literature. Instead, the focus is on the economic magnitudes of the pricing errors.

first the results for in-sample factors in Panel A. Cases with only one factor in the characteristic dimension, $K_Q = 1$, have relatively low R^2_{xs}, indicating that at least two Q factors are required to achieve a reasonable fit. There are many specifications with R^2_{xs} above 70%, indicating that the fit of the 3D-PCA factors is robust and does not depend on a specific choice of (K_C, K_P, K_Q). The benchmark specification with $(K_C, K_P, K_Q) = (3, 3, 3)$ has 27 factors and yields a cross-sectional R^2_{xs} of 83%, indicating that the 3D factors explain a large portion of the cross-sectional variation in excess returns. Several specifications with fewer factors also yield fits that are almost as good. For example, the specifications with $(K_C, K_P, K_Q) = (3, 1, 2)$ and $(K_C, K_P, K_Q) = (2, 3, 3)$ have six and 18 factors with R^2_{xs} of 70% and 81%, respectively.

The results for out-of-sample factors are shown in Panel B. As for the in-sample results, at least two characteristic factors are necessary. For most specifications with $K_Q > 1$, the R^2_{xs} of out-of-sample factors are only slightly lower than those for in-sample factors. The R^2_{xs} of the benchmark specification with 27 factors is 74%, slightly lower than the 83% R^2_{xs} for in-sample factors. More parsimonious specifications with fewer factors are $(K_C, K_P, K_Q) = (1, 3, 3)$ and $(K_C, K_P, K_Q) = (3, 2, 2)$ with 9 and 12 factors and R^2_{xs} of 65% and 70%, respectively. These results demonstrate that parsimonious out-of-sample 3D-PCA models successfully capture the cross-section of mean returns of the 275 portfolios in the sample.

The results so far are based on models that include all $K = K_C K_P K_Q$ factors. However, some of the factors could be redundant, and thus a subset of the K factors might yield a similar or even higher R^2_{xs} than the model with all K factors. I therefore also consider models with subsets with $L < K$ factors. I consider three selection methods. First, in each rolling sample, I rank factors by their contribution to the overall variation of the data, i.e., by the sums of squared F^{3D}_{cpq}, $\sum_t F^2_{cpq,t}$. This corresponds to ordering PCA factors by their eigenvalues. I then select the L factors that contribute most to the variation and estimate the model using these L factors. Second, I select factors based on their individual Sharpe ratios. Third, I select factors based on the average absolute value of betas in the time series regressions of portfolio returns on factor returns. The results are reported in Figure 12. The results for out-of-sample factors for the case $(K_C, K_P, K_Q) = (1, 3, 3)$ are shown in Panel A of Figure 12. Depending on the selection method, three to four factors are sufficient to capture the majority of the cross-sectional variation in mean returns, which is consistent with the results for Sharpe ratios in the previous section.

Panel B of Figure 12 compares the cross-sectional R^2_{xs} of out-of-sample 3D-PCA, PCA, and PCA-LS models for different numbers of factors K. If there are several 3D-PCA specifications of (K_C, K_P, K_Q) with the same total number of factors K, the figure shows the specification with the highest R^2_{xs}. Note that the 22 long-short factors do not contain a "mean" factor; I therefore add the monthly mean across all 275 portfolios as a factor to all cross-sectional regressions with PCA-LS factors. As discussed above, at most h factors are identified in PCA models. The fit of all PCA models based on all 275 portfolios is poor, with R^2_{xs} below 0.25. PCA-LS models perform somewhat better, reaching $R^2_{xs} = 0.49$ for $K = 5$, but adding more factors does not significantly improve the fit. However, all PCA-based models have substantially lower R^2_{xs} than 3D-PCA models.

Since PCA models have more free parameters than 3D-PCA models, the poor performance of PCA models could be due to the relatively short window length of $h = 48$ months. I therefore consider the fit for different rolling-sample lengths and report the results in Figure 13. Panel A shows the cross-sectional R^2_{xs} of 3D-PCA models for $h = 6, 12, 48, 120$. The fit is remarkably stable across different window lengths. The benchmark case of $h = 48$ discussed above is plotted in black. Even with $h = 6$, so that the rolling samples span only six time-series observations, R^2_{xs} reaches 55% for $K = 6$ factors. Lengthening the window to 12 months improves the fit for all K and reaches 76% for $K = 12$. Increasing the window length to 120 months has a negligible effect on the R^2_{xs}.

The PCA results for all 275 portfolios are reported in Panel B. Two patterns stand out. First, the fit improves with longer window lengths, reaching 49% for $h = 120$. Second, adding factors beyond five or six does not improve the fit, suggesting that higher-order factors do not capture additional variation in mean returns. The results for PCA-LS reported in Panel C follow the same pattern but have somewhat higher R^2_{xs}. However, PCA models perform worse than comparable 3D-PCA models. In fact, the 3D-PCA model estimated with rolling samples with three time series observations performs about as well as the best-performing PCA-LS model with window lengths of 240 months.

The results in this section show that the 3D-PCA model yields interpretable factors that produce high Sharpe ratios and explain a large portion of the cross-sectional variation in mean returns. The results are robust to different choices of (K_C, K_P, K_Q) and rolling-sample lengths. In particular, 3D-PCA models can be reliably estimated in very short samples. 3D-PCA factors dominate factors from standard 2-dimensional PCA models. Moreover, the 3D-PCA Sharpe ratios are comparable to those of machine learning methods that use firm-level returns.

7. Extensions

7.1. 2D-PCA: Single-sorted portfolios

The empirical analysis so far has used a relatively small data set of double-sorted portfolios using data from Ken French's data library. As a robustness check, I consider next a larger cross-section of single-sorted portfolios using data from the Open Source Asset Pricing database (Chen and Zimmermann (2022)). The sample period is from July 1963 to December 2024. The data set consists of 390 single-sorted quintile portfolios formed on 78 characteristics with fewer than 240 months of missing data and statistically significant characteristic premia over the sample period. The characteristics used to form the single-sorted portfolios are listed in Table B.9 in Appendix B.

It is straightforward to adapt the 3D-PCA methodology to single-sorted portfolios. In this case, the data tensor $\boldsymbol{\mathcal{X}}$ has dimensions $(T\times C\times P)$, where $T = 738$ months, $C = 78$ is the number of characteristics, and $P = 5$ is the number of portfolios (quintiles) for each characteristic. Since this data set has one fewer dimension than the double-sorted portfolios, the 3D-PCA model (13) simplifies to 2D-PCA:

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{F}} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)}. \tag{31}$$

In this case, $\boldsymbol{\mathcal{F}}$ is a $(T\times K_C\times K_P)$-dimensional tensor and $\mathbf{V}^{(C)}$ and $\mathbf{V}^{(P)}$ are matrices of dimensions $(K_C\times C)$ and $(K_P\times P)$, respectively. The estimation and interpretation of the factors are analogous to the 3D-PCA case. Since the data set has missing values, I use the estimator proposed by Lettau (2023) for tensor factor models with missing data.

Table 11 reports the cross-sectional R^2_{xs} and Sharpe ratios for the 2D-PCA model with $(K_C, K_P) = (3, 2)$ factors, as well as those for PCA and RP-PCA models with six factors.[23] The results are qualitatively similar to those of double-sorted portfolios reported in Table 9. Parsimonious models with in-sample 2D-PCA factors capture the cross-section of mean returns well. For example, the model with $L = 5$ 2D-PCA factors has an in-sample cross-sectional R^2_{xs} of 79.9%. In-sample PCA and RP-PCA factor models perform somewhat worse. However, as in the case of double-sorted portfolios, the out-of-sample performance of PCA and RP-PCA factor models deteriorates sharply. For example, the R^2_{xs} of the models with five PCA and RP-PCA factors are only 19.2% and 22.5%, respectively. In contrast, the out-of-sample 2D-PCA factors maintain strong pricing performance, with R^2_{xs} comparable to those of the in-sample models.

7.2. (m,n)D-PCA

The 3D-PCA model can be extended to tensors of arbitrary dimensions. Let $\boldsymbol{\mathcal{X}}$ be an m-dimensional tensor of size $(I_1\times I_2\times \cdots \times I_m)$. Suppose we are interested in factoring only a subset $\mathcal{N}$ of the dimensions.

[23] I omit results for Fama-French factors since they are substantially worse than those of the other models.

Let $\mathcal{N}'$ be the set of unfactored dimensions. Without loss of generality, assume that $\boldsymbol{\mathcal{X}}$ is arranged so that the first n dimensions are factored. Hence, $\mathcal{N} = \{1, \dots, n\}$ and $\mathcal{N}' = \{n+1, \dots, m\}$. The implied factor model is called (m, n)D-PCA, where n of m dimensions are factored.

Let K_j, $j \in \mathcal{N}$ be the number of factors for the factored dimensions. Then, the partial-$\mathcal{N}$ Tucker decomposition is given by

$$\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{F}}_{\mathcal{N}} \times_1 \mathbf{V}_{\mathcal{N}}^{(1)} \times_2 \mathbf{V}_{\mathcal{N}}^{(2)} \cdots \times_n \mathbf{V}_{\mathcal{N}}^{(n)} \tag{32}$$

where $\mathbf{V}_{\mathcal{N}}^{(j)}$ are $(I_j \times K_j)$-dimensional matrices and $\boldsymbol{\mathcal{F}}_{\mathcal{N}}$ is a $(K_1 \times \cdots \times K_n \times I_{n+1} \times \cdots \times I_m)$-dimensional tensor. Hence, the factored dimensions of $\boldsymbol{\mathcal{F}}_{\mathcal{N}}$ are of length K_j while the unfactored dimensions have the same size as $\boldsymbol{\mathcal{X}}$, I_j. As for 3D-PCA, the factors of the (m, n)D-PCA model are given by $\boldsymbol{\mathcal{F}}_{\mathcal{N}}$ and can be constructed as linear combinations of the elements of $\boldsymbol{\mathcal{X}}$:

$$\boldsymbol{\mathcal{F}}_{\mathcal{N}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{V}_{\mathcal{N}}^{(1)^\top} \times_2 \mathbf{V}_{\mathcal{N}}^{(2)^\top} \cdots \times_n \mathbf{V}_{\mathcal{N}}^{(n)^\top}. \tag{33}$$

As before, the columns of the $\mathbf{V}_{\mathcal{N}}^{(1)^\top}, \dots, \mathbf{V}_{\mathcal{N}}^{(n)^\top}$ matrices are the building blocks of factor weights. The difference between (m, n)D-PCA and 3D-PCA is that the factor weights are based on a subset of the m dimensions of $\boldsymbol{\mathcal{X}}$. The factor tensor $\boldsymbol{\mathcal{F}}_{\mathcal{N}}$ can be written as a matrix by folding the unfactored dimensions in $\mathcal{N}'$ as rows and the factored dimensions in $\mathcal{N}$ as columns: $\mathbf{F}_{\mathcal{N}} = \text{mat}_{\mathcal{N}}(\boldsymbol{\mathcal{F}}_{\mathcal{N}})$. The matrix of factors $\mathbf{F}_{\mathcal{N}}$ has $\prod_{k \in \mathcal{N}'} I_k$ rows and $\prod_{j \in \mathcal{N}} K_j$ columns. Each column of $\mathbf{F}_{\mathcal{N}}$ corresponds to one of the $\prod_{j \in \mathcal{N}} K_j$ factors.

We can apply this model to the data set used in this paper. $\boldsymbol{\mathcal{X}}$ is an $m = 4$-dimensional panel with dimensions time, characteristics, size quintiles, and characteristic quintiles. The 3D-PCA model in Section 4 factored all dimensions but time; hence, it is equivalent to a $(4, 3)$D-PCA model. As an alternative, consider the case where only the size quintile and characteristic quintile dimensions are factored, yielding a $(4, 2)$D-PCA model. $\boldsymbol{\mathcal{F}}_{\mathcal{N}}$ is a $(T \times C \times K_P \times K_Q)$-dimensional factor tensor and the factor matrix $\mathbf{F}_{\mathcal{N}} = \text{mat}_{\mathcal{N}}(\boldsymbol{\mathcal{F}}_{\mathcal{N}})$ has TC rows and $K_P K_Q$ columns. In the 3D-PCA model, the factor matrix $\mathbf{F}_{(T)}$ is $(T \times K_C K_P K_Q)$-dimensional so that each row corresponds to factors in a given month. In contrast, in the $(4, 2)$D-PCA model, each row of $\mathbf{F}_{\mathcal{N}}$ corresponds to the $K_P K_Q$ factors of a characteristic in a month. In other words, factors are estimated for each characteristic in each month. The interpretation of the factors is as follows. Suppose a factor's weights have a high-minus-low structure across characteristic quintiles. Then, the corresponding column of $\mathbf{F}_{\mathcal{N}}$ includes monthly factor observations of this high-minus-low factor for all characteristics. For example, the BM elements represent the time series of the estimated high-minus-low factor for the book-to-market ratio, which is related to the Fama-French HML factor. The difference is that factor weights are estimated by the $(4, 2)$D-PCA model rather than fixed. By the same token, RMW, CMA, and MOM are related to the OP, INV, and MOM entries of the columns of $\mathbf{F}_{\mathcal{N}}$.

Figure 14 shows the estimates of the $\hat{\mathbf{V}}_{\mathcal{N}}^{(P)}$ and $\hat{\mathbf{V}}_{\mathcal{N}}^{(Q)}$ matrices, which are almost identical to the corresponding matrices of the 3D-PCA shown in Figure 3. Hence, they have the same level, slope, and curvature interpretation, so that factor weights, given by the outer products of combinations of columns of $\hat{\mathbf{V}}_{\mathcal{N}}^{(P)}$ and $\hat{\mathbf{V}}_{\mathcal{N}}^{(Q)}$, are similar to those of 3D-PCA shown in Figure 4. Table 10 shows the Sharpe ratios of the nine in-sample and out-of-sample factors for each characteristic. The $F_{11}^{\mathcal{N}}$ factor is a combination of the first columns of $\hat{\mathbf{V}}_{\mathcal{N}}^{(P)}$ and $\hat{\mathbf{V}}_{\mathcal{N}}^{(Q)}$. Since both columns are long-only and (approximately) equal-weighted, $F_{11}^{\mathcal{N}}$ is (approximately) an equal-weighted average of the 25 double-sorted portfolios for each characteristic. The Sharpe ratios range from 0.41 for NSI to 0.51 for REV. The combination of the first column of $\hat{\mathbf{V}}_{\mathcal{N}}^{(P)}$ and the second column of $\hat{\mathbf{V}}_{\mathcal{N}}^{(Q)}$ yields a factor, $F_{12}^{\mathcal{N}}$, with weights that have the familiar high-minus-low pattern along the characteristic dimension. As mentioned above, this factor is related to the Fama-French factors. The in-sample Sharpe ratio of BM-$F_{12}^{\mathcal{N}}$ is 0.48, which is substantially higher than that of HML (0.32). In fact, Sharpe ratios of all $F_{12}^{\mathcal{N}}$ factors are higher than those of the corresponding Fama-French factors: 0.52

for OP-$F_{12}^{\mathcal{N}}$ compared to 0.46 for RMW, 0.67 for INV-$F_{12}^{\mathcal{N}}$ compared to 0.45 for CMA, and 0.69 for MOM-$F_{12}^{\mathcal{N}}$ compared to 0.50 for MOM.

The Sharpe ratios of the $F_{13}^{\mathcal{N}}$ factors are all negative. Since the weights underlying $F_{13}^{\mathcal{N}}$ have a convex curvature structure along characteristic quintiles, the inverse weights that are long in the middle quintiles and short in the top and bottom quintiles generate a factor with a positive mean return. The three highest (absolute) in-sample Sharpe ratios in Panel A are due to $F_{13}^{\mathcal{N}}$ factors. The NSI-$F_{13}^{\mathcal{N}}$ factor has the highest (absolute) Sharpe ratio among all factors at -0.94, followed by RVAR-$F_{13}^{\mathcal{N}}$ and VAR-$F_{13}^{\mathcal{N}}$ with Sharpe ratios of -0.86 and -0.78, respectively. Sharpe ratios of the other $(4,2)$D-PCA factors are generally smaller except for VAR and RVAR $F_{22}^{\mathcal{N}}$ and $F_{23}^{\mathcal{N}}$ factors, which combine slope in size with slope in characteristic quintiles, and slope in size with curvature in characteristic quintiles, respectively. The Sharpe ratios of out-of-sample $(4,2)$D-PCA factors follow the same patterns as their in-sample counterparts, confirming that the recursive estimation in short rolling windows yields stable results.

8. Conclusion

This paper considers an estimation method, 3D-PCA, for latent factors in multidimensional panels that exploits the structure of the data set. Factor weights are constructed from a small number of dimension-specific building blocks. The implied factor weights are subject to a set of constraints. Consequently, 3D-PCA has a relatively small number of free parameters compared to standard models based on PCA. The model is an implication of the Tucker tensor decomposition, which generalizes the matrix SVD to higher dimensions. The weights and factors can be estimated by iterative numerical methods.

I estimate 3D-PCA using a 4-dimensional panel of time series of size and characteristic-sorted portfolio returns for a set of characteristics. The estimated building blocks that determine the factor weights have familiar level, slope, and curvature patterns. The factor weights are given by interactions of level, slope, and curvature across size and characteristic quintiles. The factor weights can be estimated in short samples and are stable over time. In contrast, PCA requires longer samples and exhibits instability over time.

3D-PCA factors have substantially higher in-sample and out-of-sample Sharpe ratios than Fama-French and PCA factors, and are in line with Sharpe ratios of machine learning methods. The 3D-PCA factors with the highest Sharpe ratios are related to slope and curvature along characteristic quintiles. Moreover, 3D-PCA factor models capture the cross-section of expected returns in in-sample and out-of-sample estimations. Parsimonious models with 3D-PCA factors yield smaller pricing errors than larger models with Fama-French or PCA factors.

References

Andersson, Claus A., and Rasmus Bro, 1998, Improving the speed of multi-way algorithms: Part I. Tucker3, *Chemometrics and Intelligent Laboratory Systems* 42, 93–103.

Babii, Andrii, Eric Ghysels, and Junsu Pan, 2025, Tensor PCA for factor models, *Journal of Econometrics*.

Balasubramaniam, Vimal, John Y. Campbell, Tarun Ramadorai, and Benjamin Ranish, 2023, Who Owns What? A Factor Model for Direct Stockholding, *The Journal of Finance* 78, 1545–1591.

Bharadwaj, Anand, and Martin Lettau, 2026, To sort, or not to sort, Discussion paper Haas School of Business.

Brandi, Giuseppe, Ruggero Gramatica, and Tiziana Di Matteo, 2020, Unveil stock correlation via a new tensor-based decomposition method, *Journal of Computational Science* 46.

Bryzgalova, Svetlana, Jiantao Huang, and Christian Julliard, 2023, Bayesian Solutions for the Factor Zoo: We Just Ran Two Quadrillion Models, *Journal of Finance* 78, 487–557.

Bryzgalova, Svetlana, Sven Lerner, Martin Lettau, and Markus Pelger, 2025, Missing financial data, *Review of Financial Studies* 38, 803–882.

Chamberlain, Gary, and Michael Rothschild, 1983, Arbitrage, Factor Structure, and Mean-Variance Analysis on Large Asset Markets, *Econometrica* 51, 1281–1304.

Chen, Andrew Y., and Tom Zimmermann, 2022, Open Source Cross-Sectional Asset Pricing, *Critical Finance Review* 11.

Chen, Luyang, Markus Pelger, and Jason Zhu, 2024, Deep Learning in Asset Pricing, *Management Science* 20, 714–750.

Chen, Yilun, Ami Wiesel, Yonina C. Eldar, and Alfred O. Hero, 2010, Shrinkage algorithms for MMSE covariance estimation, *IEEE Transactions on Signal Processing* 58.

Cochrane, John H., 2011, Discount Rates, *Journal of Finance* 66, 1047–1108.

Connor, Gregory, Matthias Hagmann, and Oliver Linton, 2012, Efficient Semiparametric Estimation of the Fama–French Model and Extensions, *Econometrica* 80, 713–754.

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

——— , 1988, Risk and return in an equilibrium APT: Application of a new test methodology, *Journal of Financial Economics* 21, 255–289.

De Lathauwer, Lieven, Bart De Moor, and Joos Vandewalle, 2000, A multilinear singular value decomposition, *SIAM Journal on Matrix Analysis and Applications* 21, 1253–1278.

Dees, Bruno Scalzo, 2019, Analysing Global Fixed Income Markets with Tensors, *arXiv.org* 1908.02101.

Eckart, Carl, and Gale Young, 1936, The Approximation of One Matrix by Another of Lower Rank, *Psychometrika* 1, 211–218.

Elden, Lars, and Berkant Savas, 2009, A Newton–Grassmann Method for Computing the Best Multilinear Rank (r1, r2, r3) Approximation of a Tensor, *SIAM Journal on Matrix Analysis and Applications* 31, 248–271.

Fama, Eugene F., and Kenneth R. French, 1993, Common Risk Factors in the Returns on Stocks and Bonds, *Journal of Financial Economics* 33, 3–56.

Fan, Jianqing, Yuan Liao, and Weichen Wang, 2016, Projected principal component analysis in factor models, *The Annals of Statistics* 44, 219–254.

Giglio, Stefano, and Dacheng Xiu, 2021, Asset pricing with omitted factors, *Journal of Political Economy* 129, 1947–1990.

Gu, Shihao, Bryan Kelly, and Dacheng Xiu, 2020, Empirical Asset Pricing via Machine Learning, *The Review of Financial Studies* 33, 2223–2273.

——— , 2021, Autoencoder asset pricing models, *Journal of Econometrics* 222, 429–450.

Huberman, Gur, 1982, A Simple Approach to Arbitrage Pricing Theory, *Journal of Economic Theory* 28, 183–191.

Jang, Junkyu, and Noh Yoon Seong, 2023, Deep reinforcement learning for stock portfolio optimization by connecting with modern portfolio theory, *Expert Systems with Applications* 218.

Jiang, Fuwei, Guohao Tang, and Guofu Zhou, 2018, Firm Characteristics and Chinese Stocks, *Journal of Management Science and Engineering* 3, 259–283.

Jobson, J. D., and Bob M. Korkie, 1981, Performance Hypothesis Testing with the Sharpe and Treynor Measures, *The Journal of Finance* 36.

Kelly, Bryan, Seth Pruitt, and Yinan Su, 2019, Characteristics are covariances: A unified model of risk and return, *Journal of Financial Economics* 134, 501–524.

Kim, Soohun, Robert A. Korajczyk, and Andreas Neuhierl, 2021, Arbitrage Portfolios, *The Review of Financial Studies* 34, 2813–2856.

Kolda, Tamara G., and Brett W. Bader, 2009, Tensor decompositions and applications, *SIAM Review* 51, 455–500.

Kozak, Serhiy, Stefan Nagel, and Shrihari Santosh, 2020, Shrinking the Cross-Section, *Journal of Financial Economics* 135, 271–292.

Kroonenberg, Pieter M., and Jan de Leeuw, 1980, Principal component analysis of three-mode data by means of alternating least squares algorithms, *Psychometrika* 45, 69–97.

Ledoit, Oliver, and Michael Wolf, 2008, Robust performance hypothesis testing with the Sharpe ratio, *Journal of Empirical Finance* 15.

Lettau, Martin, 2023, High-Dimensional Factor Models and the Factor Zoo, *National Bureau of Economic Research* No. 31719.

——— , and Markus Pelger, 2020a, Estimating latent asset-pricing factors, *Journal of Econometrics* 218, 1–31.

——— , 2020b, Factors That Fit the Time Series and Cross-Section of Stock Returns, *The Review of Financial Studies* 33, 2274–2325.

Memmel, Christoph, 2003, Performance Hypothesis Testing with the Sharpe Ratio, *Finance Letters* pp. 21–23.

Ross, Stephen A., 1976, The arbitrage theory of capital asset pricing, *Journal of Economic Theory* 13, 341–360.

Subrahmanyam, Avanidhar, 2010, The Cross-Section of Expected Stock Returns: What Have We Learnt from the Past Twenty-Five Years of Research?, *European Financial Management* 16, 27–42.

Tucker, Ledyard R., 1966, Some mathematical notes on three-mode factor analysis, *Psychometrika* 31, 279–311.

Varolgunes, Uras, Dan Zhou, Dantong Yu, and Ajim Uddin, 2023, NMTucker: Non-linear Matryoshka Tucker Decomposition for Financial Time Series Imputation, *ICAIF 2023 - 4th ACM International Conference on AI in Finance* 23, 516–523.

Zaffaroni, Paolo, 2025, Factor Models for Conditional Asset Pricing, *Journal of Political Economy* 133.

Zhang, Anru, and Dong Xia, 2018, Tensor SVD: Statistical and Computational Limits, *IEEE Transactions on Information Theory* 64, 7311–7338.

Table 1. Long-short portfolios

	Mean			Std. dev.			Sharpe ratio		
c	SMB_c	HML_c	$CROSS_c$	SMB_c	HML_c	$CROSS_c$	SMB_c	HML_c	$CROSS_c$
AC	3.10	2.61	-1.28	15.68	4.91	12.86	0.20	0.53	-0.10
BETA	3.33	0.70	-0.00	13.42	16.86	16.86	0.25	0.04	-0.00
BM	2.23	4.93	9.10	15.27	12.59	17.61	0.15	0.39	0.52
INV	2.58	4.17	4.97	14.58	8.05	13.99	0.18	0.52	0.36
MOM	2.91	10.43	8.30	13.31	17.22	16.21	0.22	0.61	0.51
NSI	3.28	5.22	4.72	15.82	9.28	13.64	0.21	0.56	0.35
OP	3.91	3.86	0.22	14.19	9.16	14.30	0.28	0.42	0.02
REV	2.42	4.06	3.17	13.18	10.41	15.83	0.18	0.39	0.20
RVAR	2.60	5.31	13.10	14.47	17.93	16.97	0.18	0.30	0.77
SREV	1.30	7.37	9.26	13.84	12.96	15.51	0.09	0.57	0.60
VAR	2.58	4.48	13.70	14.69	19.55	17.52	0.18	0.23	0.78
Mean	2.75	4.83	5.93	14.40	12.63	15.57	0.19	0.41	0.36

Note: The table reports annualized means, standard deviations, and Sharpe ratios of long-short portfolios. SMB_c is the difference of the small stock portfolios (P5) and the big stock portfolios (P1) of the double sort of characteristic c averaged of all characteristic quintiles. HML_c is constructed accordingly as the difference of Q5 and Q1 portfolios averaged over all size quintiles. $CROSS_c$ is the average of the P1Q1 and P5Q5 portfolios minus the average of the P1Q5 and P5Q1 portfolios. The sample is from July 1963 to October 2023.

Table 2. Monte Carlo simulations: $\sigma_e = 0.15$

(K_C, K_P, K_Q)	T	RMSE				Correlations			
		$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$	$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$
(2, 2, 2)	25	0.007	0.051	0.122	0.481	0.977	0.585	0.978	0.608
	50	0.005	0.044	0.073	0.324	0.984	0.698	0.988	0.713
	100	0.004	0.037	0.047	0.221	0.989	0.779	0.993	0.787
	250	0.002	0.029	0.025	0.126	0.997	0.864	0.997	0.866
	750	0.001	0.022	0.015	0.070	0.999	0.914	0.998	0.914
(3, 3, 3)	25	0.021	0.075	0.167	0.531	0.915	0.207	0.910	0.212
	50	0.014	0.072	0.100	0.425	0.959	0.283	0.956	0.279
	100	0.010	0.069	0.060	0.335	0.981	0.343	0.978	0.340
	250	0.006	0.064	0.032	0.245	0.993	0.428	0.990	0.424
	750	0.003	0.057	0.018	0.166	0.998	0.549	0.995	0.545
(4, 2, 2)	25	0.030	0.068	0.252	0.543	0.836	0.328	0.832	0.337
	50	0.023	0.065	0.161	0.419	0.896	0.403	0.897	0.410
	100	0.016	0.061	0.098	0.324	0.945	0.471	0.943	0.475
	250	0.010	0.057	0.049	0.234	0.980	0.553	0.978	0.553
	750	0.005	0.045	0.022	0.141	0.995	0.712	0.993	0.711

Note: This table shows results of 1,000 Monte Carlo simulations based on the misspecified data generating process in equations (27) and (28) with $\sigma_e = 0.15$ and $\sigma_w = 0.25$ for different combinations of factor numbers (K_C, K_P, K_Q) and sample sizes T. The dimensions of the data tensor are $(T, 11, 5, 5)$. The table reports root mean squared errors (RMSE) and correlations between the true and estimated factors for both 3D-PCA and standard PCA methods.

Table 3. Monte Carlo simulations under misspecification: $\sigma_w = 0.25$

(K_C, K_P, K_Q)	T	RMSE				Correlations			
		$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$	$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$
(2, 2, 2)	25	0.017	0.051	0.133	0.476	0.902	0.615	0.974	0.610
	50	0.016	0.044	0.082	0.322	0.906	0.717	0.985	0.712
	100	0.015	0.038	0.055	0.222	0.913	0.788	0.990	0.784
	250	0.015	0.029	0.036	0.125	0.916	0.872	0.993	0.868
	750	0.015	0.023	0.024	0.071	0.918	0.914	0.995	0.912
(3, 3, 3)	25	0.026	0.075	0.177	0.531	0.870	0.216	0.897	0.212
	50	0.021	0.072	0.110	0.426	0.913	0.290	0.944	0.279
	100	0.018	0.069	0.070	0.334	0.937	0.349	0.969	0.341
	250	0.016	0.064	0.043	0.245	0.948	0.431	0.982	0.425
	750	0.015	0.057	0.028	0.166	0.952	0.551	0.986	0.546
(4, 2, 2)	25	0.034	0.068	0.255	0.542	0.789	0.346	0.828	0.340
	50	0.028	0.065	0.163	0.421	0.852	0.414	0.894	0.409
	100	0.023	0.062	0.103	0.327	0.893	0.474	0.939	0.470
	250	0.018	0.057	0.055	0.236	0.927	0.551	0.975	0.547
	750	0.016	0.045	0.031	0.139	0.939	0.715	0.988	0.713

Note: This table shows results of 1,000 Monte Carlo simulations based on the data generating process in equations (25) and (26) with idiosyncratic error standard deviation $\sigma_e = 0.25$ for different combinations of factor numbers (K_C, K_P, K_Q) and sample sizes T. The dimensions of the data tensor are $(T, 11, 5, 5)$. The table reports root mean squared errors (RMSE) and correlations between the true and estimated factors for both 3D-PCA and standard PCA methods.

Table 4. Standard errors of $\mathbf{V}^{(i)}$

	$\mathbf{V}^{(C)}$	$\mathbf{V}^{(P)}$	$\mathbf{V}^{(Q)}$
$\mathbf{v}_1$	0.002	0.002	0.001
$\mathbf{v}_2$	0.024	0.009	0.004
$\mathbf{v}_3$	0.121	0.023	0.007

Note: The table shows bootstrapped standard errors of the columns of the weight matrices $\mathbf{V}^{(i)}$ based on the 3D-PCA weight with $(K_C, K_P, K_Q) = (3, 3, 3)$. The table reports the average standard errors of the row elements in $\mathbf{v}_j^{(i)}$ divided by the average absolute value of the elements in $\mathbf{v}_j^{(i)}$ across 1,000 replications. The sample is from July 1963 to October 2023.

Table 5. 3D-PCA factors: Sums of squared F^{3D}_{cpq}

	$F^{3D}_{\bullet 11}$	$F^{3D}_{\bullet 12}$	$F^{3D}_{\bullet 13}$	$F^{3D}_{\bullet 21}$	$F^{3D}_{\bullet 22}$	$F^{3D}_{\bullet 23}$	$F^{3D}_{\bullet 31}$	$F^{3D}_{\bullet 32}$	$F^{3D}_{\bullet 33}$
				A: In-sample					
$F^{3D}_{1\bullet\bullet}$	573.92	5.45	3.48	28.52	0.54	0.50	5.80	0.35	0.33
$F^{3D}_{2\bullet\bullet}$	0.50	15.15	1.05	0.35	1.07	0.26	0.11	0.53	0.20
$F^{3D}_{3\bullet\bullet}$	0.68	5.03	0.73	0.33	0.70	0.22	0.13	0.29	0.14
				B: Out-of-sample					
$F^{3D}_{1\bullet\bullet}$	573.57	28.42	6.15	0.42	0.32	0.11	1.08	0.43	0.17
$F^{3D}_{2\bullet\bullet}$	5.14	0.60	0.37	15.89	1.17	0.50	3.79	0.69	0.35
$F^{3D}_{3\bullet\bullet}$	2.93	0.51	0.32	0.86	0.27	0.21	1.15	0.30	0.19

Note: The table reports the sum of squared factors of the 27 in-sample and out-of-sample 3D-PCA factors based on a model with $K_C = K_P = K_Q = 3$ factors. F^{3D}_{cpq} factor corresponding to the c-th column of $\mathbf{V}^{(C)^\top}$, the p-th column of $\mathbf{V}^{(P)^\top}$, and the q-th column of $\mathbf{V}^{(Q)^\top}$. The $\bullet$ symbol can take values of 1, 2, or 3. Out-of-sample factors are estimated in rolling samples of length $h = 48$ months. The sample is from July 1963 to October 2023.

Table 6. Correlations of in-sample and out-of-sample 3D-PCA and PCA factors

	3D-PCA		PCA		PCA-LS	
1	F^{3D}_{112}	0.86	F^{PC}_{16}	0.04	F^{LS}_{5}	0.29
2	F^{3D}_{122}	0.89	F^{PC}_{3}	0.60	F^{LS}_{9}	0.23
3	F^{3D}_{113}	0.87	F^{PC}_{7}	−0.31	F^{LS}_{2}	0.81
4	F^{3D}_{123}	0.93	F^{PC}_{4}	−0.41	F^{LS}_{21}	0.17
5	F^{3D}_{222}	0.86	F^{PC}_{21}	0.08	F^{LS}_{4}	−0.30

Note: The table reports plots the correlations between in-sample and out-of-sample factors for 3D-PCA, PCA, and PCA-LS of the five factors with the highest Sharpe ratios. The underlying 3D-PCA model is based on $(K_C, K_P, K_Q) = (3, 3, 3)$. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB_c and HML_c portfolios. Out-of-sample factors are estimated in rolling samples of length $h = 24$ months. The sample is from July 1963 to October 2023.

Table 7. Sharpe ratios of 3D-PCA models

	$K_Q = 1$			$K_Q = 2$			$K_Q = 3$		
	$K_P = 1$	$K_P = 2$	$K_P = 3$	$K_P = 1$	$K_P = 2$	$K_P = 3$	$K_P = 1$	$K_P = 2$	$K_P = 3$
A. In-sample									
$K_C = 1$	0.46	0.46	0.47	1.21	1.53	1.72	1.34	1.77	1.91
$K_C = 2$	0.48	0.50	0.53	1.37	1.75	1.95	1.42	1.89	2.08
$K_C = 3$	0.76	0.81	0.85	1.51	1.88	2.09	1.63	2.12	2.30
B. Out-of-sample									
$K_C = 1$	0.42	0.26	0.44	1.38	1.63	1.83	1.53	1.76	1.94
$K_C = 2$	0.23	0.42	0.42	1.27	1.59	1.78	1.57	1.75	1.90
$K_C = 3$	0.41	0.48	0.42	1.35	1.68	1.70	1.43	1.87	1.89

Note: The table shows in-sample and out-of-sample Sharpe ratios for models with 3D-PCA factors for different combinations of K_C, K_P, and K_Q. Factors are estimated in out-of-sample in rolling windows of length $h = 48$ months. The sample is from July 1963 to October 2023.

Table 8. Out-of-sample returns of 3D-PCA and machine learning models

	Sharpe ratio	Skewness	Kurtosis
3D-PCA	1.86	0.12	0.71
Boosted trees	1.82	0.11	6.35
Random forest	1.81	−0.14	6.14
Neural net	1.89	0.04	6.18

Note: The table shows descriptive statistics of returns of out-of-sample returns of 3D-PCA, PCA, PCA-LS, and machine learning methods. 3D-PCA factors and machine learning models are estimated in out-of-sample in rolling windows of length $h = 48$ months and $h = 240$ months, repectively. The moments in the table are based on observations from July 1983 to October 2023.

Table 9. R^2_{xs} of 3D-PCA models

	$K_Q = 1$			$K_Q = 2$			$K_Q = 3$		
	$K_P = 1$	$K_P = 2$	$K_P = 3$	$K_P = 1$	$K_P = 2$	$K_P = 3$	$K_P = 1$	$K_P = 2$	$K_P = 3$
A. In-sample factors									
$K_C = 1$	-0.45	-0.41	-0.36	0.32	0.40	0.59	0.47	0.63	0.71
$K_C = 2$	-0.31	-0.20	-0.16	0.57	0.62	0.71	0.59	0.67	0.76
$K_C = 3$	0.39	0.40	0.42	0.70	0.72	0.81	0.68	0.74	0.83
B. Out-of-sample factors									
$K_C = 1$	-0.44	-0.43	-0.37	0.31	0.43	0.60	0.36	0.53	0.67
$K_C = 2$	-0.34	-0.42	-0.35	0.39	0.56	0.63	0.30	0.57	0.64
$K_C = 3$	0.16	-0.26	-0.15	0.58	0.68	0.72	0.60	0.68	0.77

Note: The table shows cross-sectional R^2_{xs} for models with 3D-PCA factors for different combinations of K_C, K_P, and K_Q. The underlying 3D-PCA model is based on $(K_C, K_P, K_Q) = (3, 3, 3)$. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB_c and HML_c portfolios. Factors are estimated in out-of-sample in rolling windows of length $h = 24$ months. The sample is from July 1963 to October 2023.

Table 10. (4, 2)D-PCA – Sharpe ratios of factors

	$F_{11}^{\mathcal{N}}$	$F_{12}^{\mathcal{N}}$	$F_{13}^{\mathcal{N}}$	$F_{21}^{\mathcal{N}}$	$F_{22}^{\mathcal{N}}$	$F_{23}^{\mathcal{N}}$	$F_{31}^{\mathcal{N}}$	$F_{32}^{\mathcal{N}}$	$F_{33}^{\mathcal{N}}$
BM	0.44	0.47	−0.30	−0.10	0.40	−0.22	−0.09	0.08	−0.08
OP	0.44	0.56	−0.43	0.04	0.01	−0.29	−0.08	0.02	−0.30
INV	0.45	0.71	−0.50	−0.07	0.17	−0.44	0.01	0.28	0.05
MOM	0.41	0.67	−0.30	−0.01	0.29	−0.08	−0.02	0.14	−0.05
REV	0.49	0.45	−0.26	−0.07	0.07	−0.14	−0.09	0.04	−0.12
SREV	0.40	0.62	−0.45	−0.12	0.30	0.00	−0.26	−0.09	−0.03
AC	0.42	0.61	−0.47	−0.02	−0.02	−0.16	−0.04	0.27	−0.14
BETA	0.47	0.07	−0.59	−0.01	0.07	−0.14	−0.03	0.02	0.00
NSI	0.38	0.74	−0.88	0.03	0.20	−0.12	−0.02	0.18	−0.16
VAR	0.44	0.36	−0.94	−0.05	0.64	−0.64	−0.17	0.44	−0.16
RVAR	0.44	0.43	−1.03	−0.05	0.64	−0.71	−0.16	0.36	0.01

Note: The table reports annualized Sharpe ratios of (4, 2)D-PCA factors. Factors are estimated out-of-sample in rolling windows of length $h = 24$ months. The sample is from July 1963 to October 2023.

Table 11. 2D-PCA for single-sorted portfolios

	R^2_{xs}				Sharpe ratio			
	$K_C = 1$	$K_C = 2$	$K_C = 3$	$K_C = 4$	$K_C = 1$	$K_C = 2$	$K_C = 3$	$K_C = 4$
A: 2D-PCA								
$K_Q = 1$	0.31	0.38	0.37	0.37	0.73	0.80	0.69	0.70
$K_Q = 2$	0.58	0.76	0.76	0.76	1.59	1.58	1.59	1.47
$K_Q = 3$	0.66	0.77	0.77	0.78	1.61	1.55	1.70	1.60
B: PCA								
$K_Q = 1$	0.30	0.38	0.39	0.40	0.73	0.64	0.58	0.68
$K_Q = 2$	0.38	0.40	0.41	0.41	0.64	0.68	0.58	0.49
$K_Q = 3$	0.39	0.41	0.40	0.41	0.58	0.58	0.52	0.50

Note: The table shows cross-sectional R^2_{xs} of models with 2D-PCA with $(K_C, K_P) = (4, 3)$ factors and PCA with $K = 12$ factors for single-sorted quintile portfolios of 74 characteristics. Factors are estimated out-of-sample in rolling windows of length $h = 24$ months for R^2_{xs} and $h = 48$ for the Sharpe ratios. The sample is from July 1963 to December 2024.

Figure 1. Means returns of portfolios

BM

	Q1	Q2	Q3	Q4	Q5
P1	6.9	6.4	6.8	6.0	7.7
P2	7.5	7.6	8.4	9.7	10.1
P3	6.2	9.4	8.7	10.2	11.5
P4	5.9	9.1	10.0	10.4	11.8
P5	3.0	8.9	8.9	11.3	12.9

OP

	Q1	Q2	Q3	Q4	Q5
P1	4.1	5.1	6.4	6.5	7.6
P2	6.7	8.1	8.3	8.7	9.7
P3	6.2	8.5	8.9	9.2	10.8
P4	6.8	9.2	9.6	9.5	11.3
P5	6.0	11.1	10.4	12.0	9.7

INV

	Q1	Q2	Q3	Q4	Q5
P1	6.3	6.6	6.4	7.0	8.7
P2	7.1	9.0	8.8	8.9	9.2
P3	6.3	9.2	9.1	10.4	10.6
P4	5.7	10.6	10.7	10.3	10.2
P5	4.1	10.0	11.3	11.1	11.5

MOM

	Q1	Q2	Q3	Q4	Q5
P1	3.2	6.3	5.6	6.9	9.5
P2	2.8	7.6	8.1	9.2	11.9
P3	3.5	7.4	8.6	8.9	13.5
P4	2.2	8.2	9.9	11.7	14.3
P5	0.6	7.9	10.3	12.0	15.2

REV

P1	6.1	7.0	7.0	7.9	8.6
P2	7.1	9.0	9.2	8.4	10.3
P3	7.9	9.3	9.5	10.5	11.6
P4	7.9	11.0	10.9	10.1	13.0
P5	6.1	9.3	11.0	10.5	11.8

SREV

P1	4.3	6.3	6.8	7.8	7.0
P2	3.8	7.1	9.0	11.0	10.4
P3	4.1	7.0	9.5	11.1	11.8
P4	4.4	7.6	9.7	12.1	12.4
P5	0.5	7.4	8.7	9.7	12.4

AC

P1	4.1	6.2	6.5	7.0	8.6
P2	8.7	9.1	8.0	8.4	8.9
P3	7.4	9.0	9.8	9.1	9.9
P4	7.3	9.0	9.5	10.0	9.8
P5	7.0	10.5	9.2	10.9	10.3

BETA

P1	6.0	6.9	6.8	7.4	6.3
P2	7.7	9.4	8.7	7.5	9.4
P3	7.7	10.0	9.8	9.6	8.6
P4	7.9	9.9	11.2	10.6	8.3
P5	8.7	10.0	10.5	11.6	9.1

NSI

P1	2.7	6.0	8.2	6.1	6.1
P2	4.4	8.1	9.1	8.2	8.1
P3	4.1	8.4	9.4	11.0	9.3
P4	4.2	10.2	9.9	9.8	10.0
P5	2.5	9.3	12.2	11.0	10.6

VAR

P1	6.8	6.1	6.8	7.5	5.8
P2	6.4	8.9	9.3	8.8	8.6
P3	5.7	10.4	11.6	9.9	8.7
P4	4.7	11.1	12.1	12.2	10.1
P5	-1.2	9.5	12.4	13.6	11.6

RVAR

P1	6.7	6.3	6.3	7.0	6.6
P2	5.6	8.9	9.1	9.4	8.8
P3	4.7	10.6	11.4	10.9	8.5
P4	3.8	11.4	12.4	12.0	10.5
P5	-1.3	9.5	12.3	13.8	11.7

Mean

P1	5.2	6.3	6.7	7.0	7.5
P2	6.2	8.4	8.7	8.9	9.6
P3	5.8	9.0	9.7	10.1	10.4
P4	5.5	9.8	10.5	10.8	11.0
P5	3.3	9.4	10.7	11.6	11.5

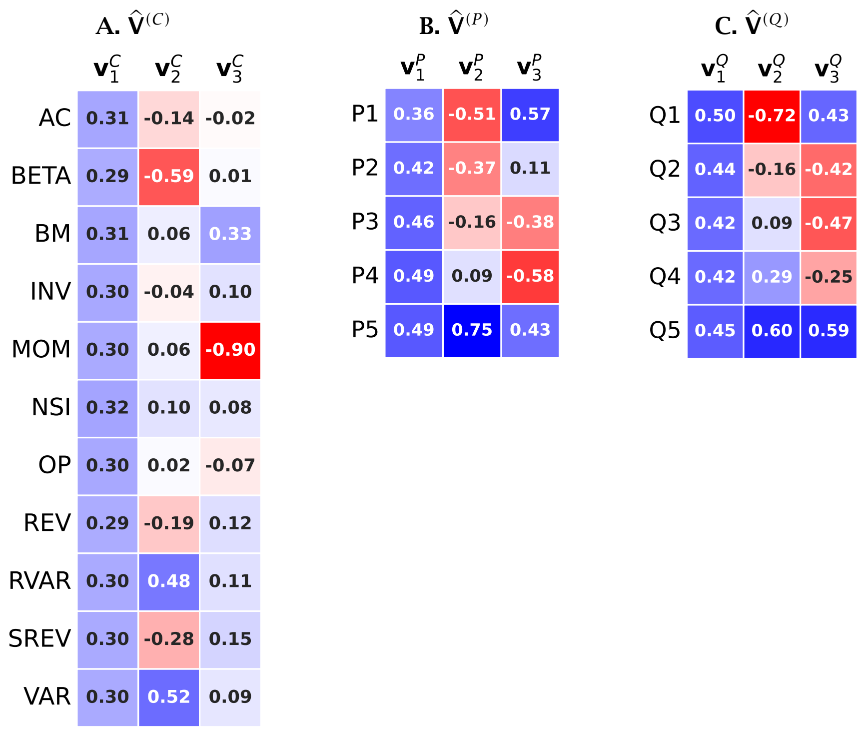
Notes: The figure shows heatmaps of annualized mean returns of the 275 portfolios. Each panel shows the 5×5-double sorted portfolios of a characteristic. The bottom-right panel shows the mean across all 11 characteristics. Portfolios with means that are lower (higher) than the overall mean portfolio return (8.54%)are in red (blue). The sample is from July 1963 to October 2023.

Figure 2. Time series R^2_{ts} of 3D-PCA

(K_P, K_Q) \ K_C	1	2	3	4	5	6	7	8	9	10	11
(1, 1)	82.6%	82.9%	83.0%	83.1%	83.1%	83.1%	83.1%	83.1%	83.2%	83.2%	83.2%
(1, 2)	83.8%	85.7%	86.5%	87.0%	87.4%	87.6%	87.8%	87.9%	88.0%	88.0%	88.0%
(1, 3)	83.9%	86.4%	87.3%	87.8%	88.3%	88.7%	88.9%	89.0%	89.2%	89.3%	89.3%
(1, 4)	83.9%	86.4%	87.4%	87.9%	88.4%	88.8%	89.1%	89.3%	89.4%	89.5%	89.5%
(1, 5)	83.9%	86.4%	87.4%	88.0%	88.5%	88.9%	89.2%	89.4%	89.6%	89.7%	89.7%
(2, 1)	86.8%	87.2%	87.3%	87.4%	87.5%	87.6%	87.6%	87.7%	87.7%	87.7%	87.7%
(2, 2)	88.0%	90.2%	91.2%	91.8%	92.3%	92.6%	92.9%	93.1%	93.2%	93.3%	93.3%
(2, 3)	88.2%	90.9%	92.1%	92.8%	93.4%	93.9%	94.3%	94.5%	94.8%	94.9%	95.0%
(2, 4)	88.3%	91.0%	92.2%	92.9%	93.7%	94.2%	94.6%	95.0%	95.3%	95.4%	95.5%
(2, 5)	88.3%	91.1%	92.3%	93.1%	93.8%	94.4%	94.9%	95.2%	95.6%	95.8%	95.9%
(3, 1)	87.6%	88.1%	88.2%	88.4%	88.5%	88.5%	88.6%	88.6%	88.7%	88.7%	88.7%
(3, 2)	89.0%	91.2%	92.2%	92.9%	93.5%	93.9%	94.2%	94.4%	94.6%	94.7%	94.7%
(3, 3)	89.2%	92.0%	93.2%	94.0%	94.8%	95.3%	95.8%	96.1%	96.4%	96.6%	96.6%
(3, 4)	89.2%	92.1%	93.4%	94.3%	95.1%	95.8%	96.3%	96.7%	97.1%	97.3%	97.4%
(3, 5)	89.3%	92.2%	93.5%	94.5%	95.4%	96.0%	96.6%	97.1%	97.6%	97.9%	98.0%
(4, 1)	87.9%	88.4%	88.6%	88.7%	88.8%	88.9%	88.9%	89.0%	89.1%	89.1%	89.1%
(4, 2)	89.3%	91.5%	92.6%	93.3%	94.0%	94.4%	94.7%	95.0%	95.2%	95.3%	95.3%
(4, 3)	89.5%	92.4%	93.7%	94.5%	95.4%	96.0%	96.4%	96.8%	97.2%	97.4%	97.5%
(4, 4)	89.6%	92.6%	93.9%	94.9%	95.8%	96.5%	97.1%	97.6%	98.0%	98.3%	98.4%
(4, 5)	89.7%	92.7%	94.1%	95.1%	96.1%	96.9%	97.5%	98.1%	98.6%	99.0%	99.1%
(5, 1)	88.0%	88.5%	88.7%	88.9%	89.0%	89.1%	89.1%	89.2%	89.2%	89.3%	89.3%
(5, 2)	89.4%	91.7%	92.8%	93.5%	94.2%	94.7%	95.1%	95.4%	95.6%	95.7%	95.7%
(5, 3)	89.7%	92.6%	94.0%	94.9%	95.8%	96.4%	96.9%	97.3%	97.7%	98.0%	98.0%
(5, 4)	89.8%	92.8%	94.3%	95.3%	96.3%	97.0%	97.6%	98.2%	98.7%	99.0%	99.1%
(5, 5)	89.9%	93.0%	94.5%	95.6%	96.7%	97.5%	98.2%	98.9%	99.5%	99.9%	100.0%

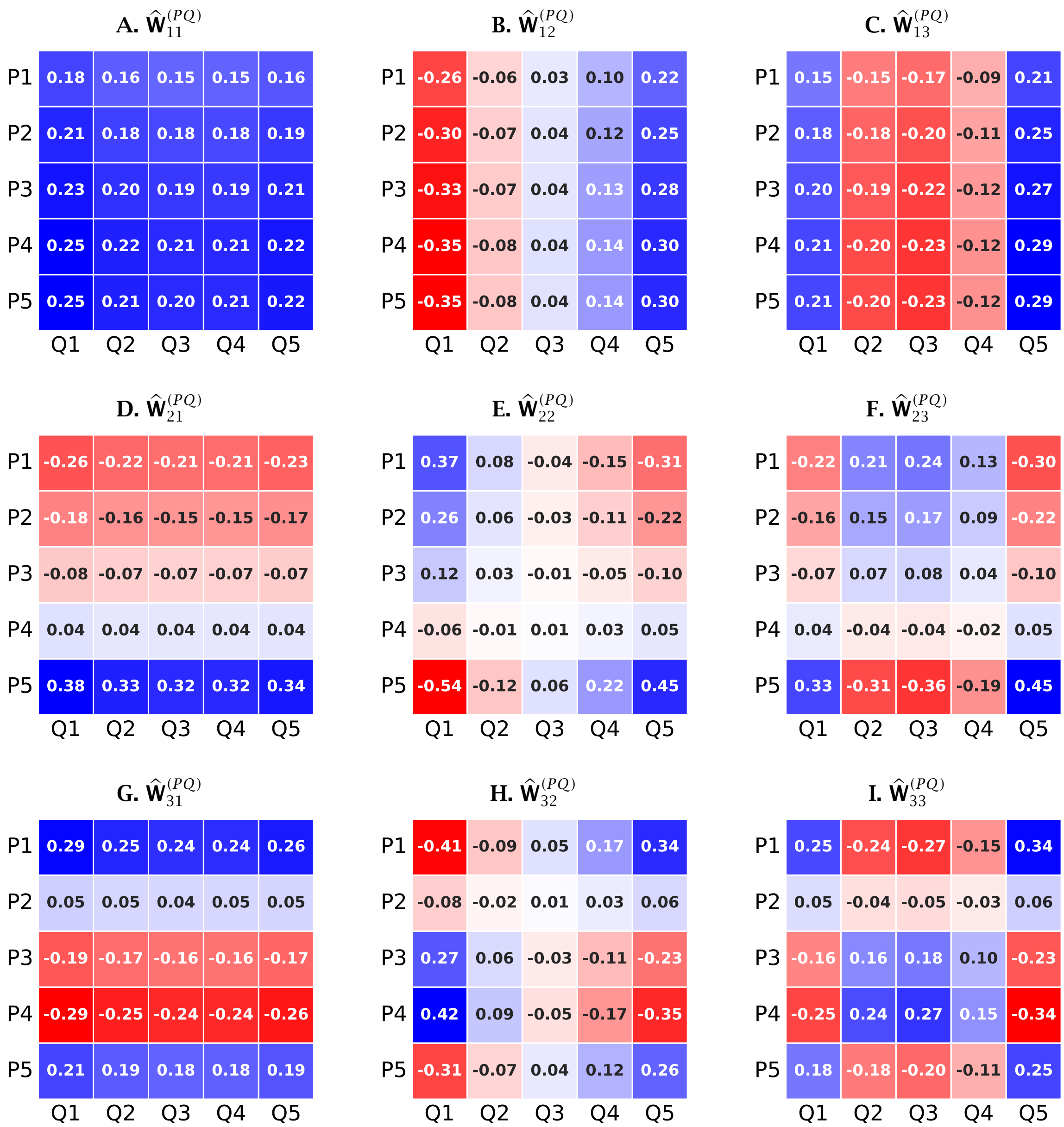
Notes: This figure shows time series R^2_{ts} of 3D-PCA models for different combinations of the number of factors, (K_C, K_P, K_Q). The x-axis shows K_C, the y-axis (K_P, K_Q). Darker colors indicate higher R^2_{ts}. The sample is from July 1967 to October 2023.

Figure 3. 3D-PCA – $\hat{\mathbf{V}}^i$ matrices



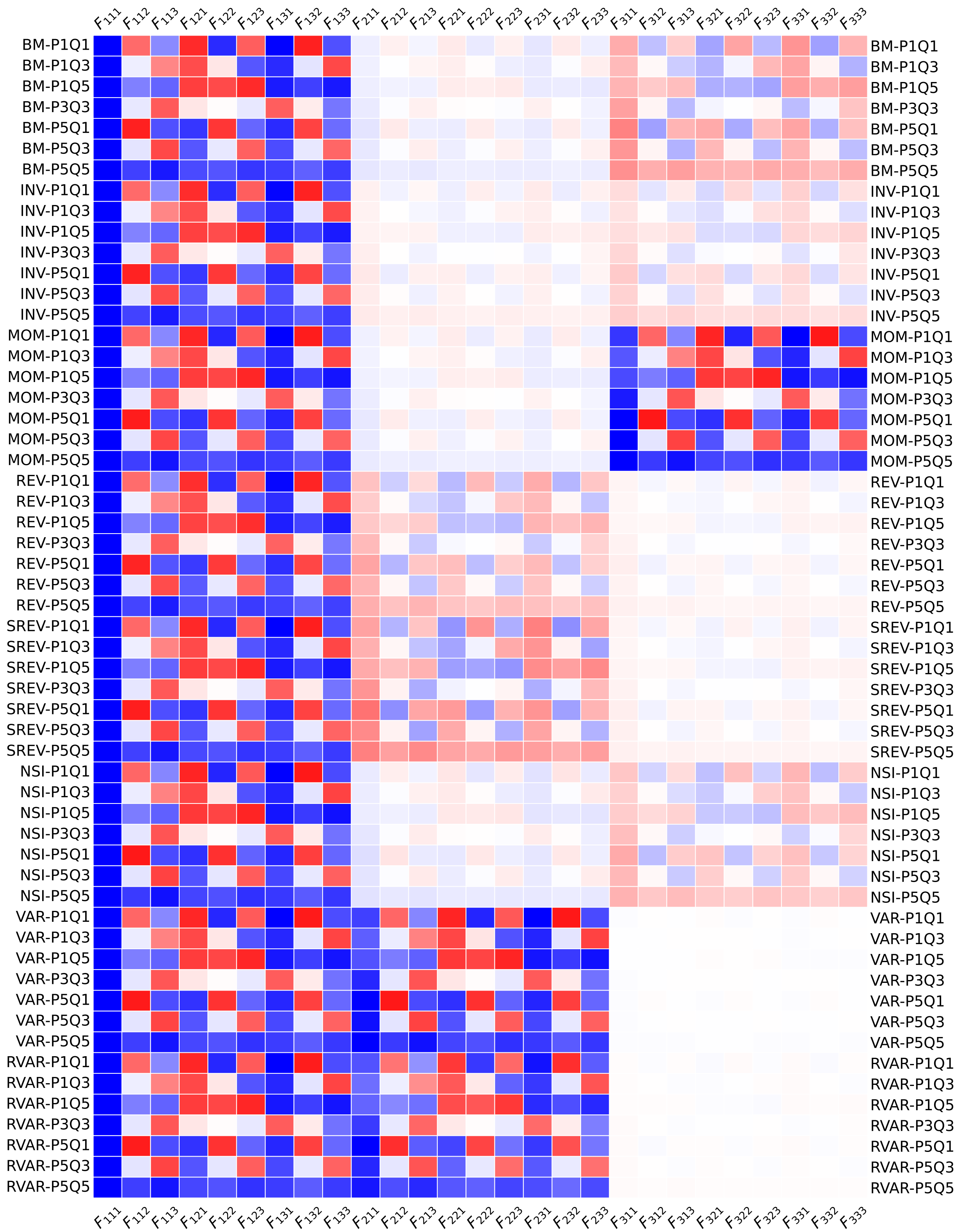
Notes: The heatmaps show estimates of the $\hat{\mathbf{V}}^{(C)}, \hat{\mathbf{V}}^{(P)}$, and $\hat{\mathbf{V}}^{(Q)}$ matrices of the 3D-PCA model (14) with $K_C = K_P = K_Q = 3$. Negative values are plotted in red and positive ones in blue. The model is estimated by HOOI. The sample is from July 1963 to October 2023.

Figure 4. 3D-PCA weights $\hat{\mathbf{W}}_{pq}^{(PQ)}$



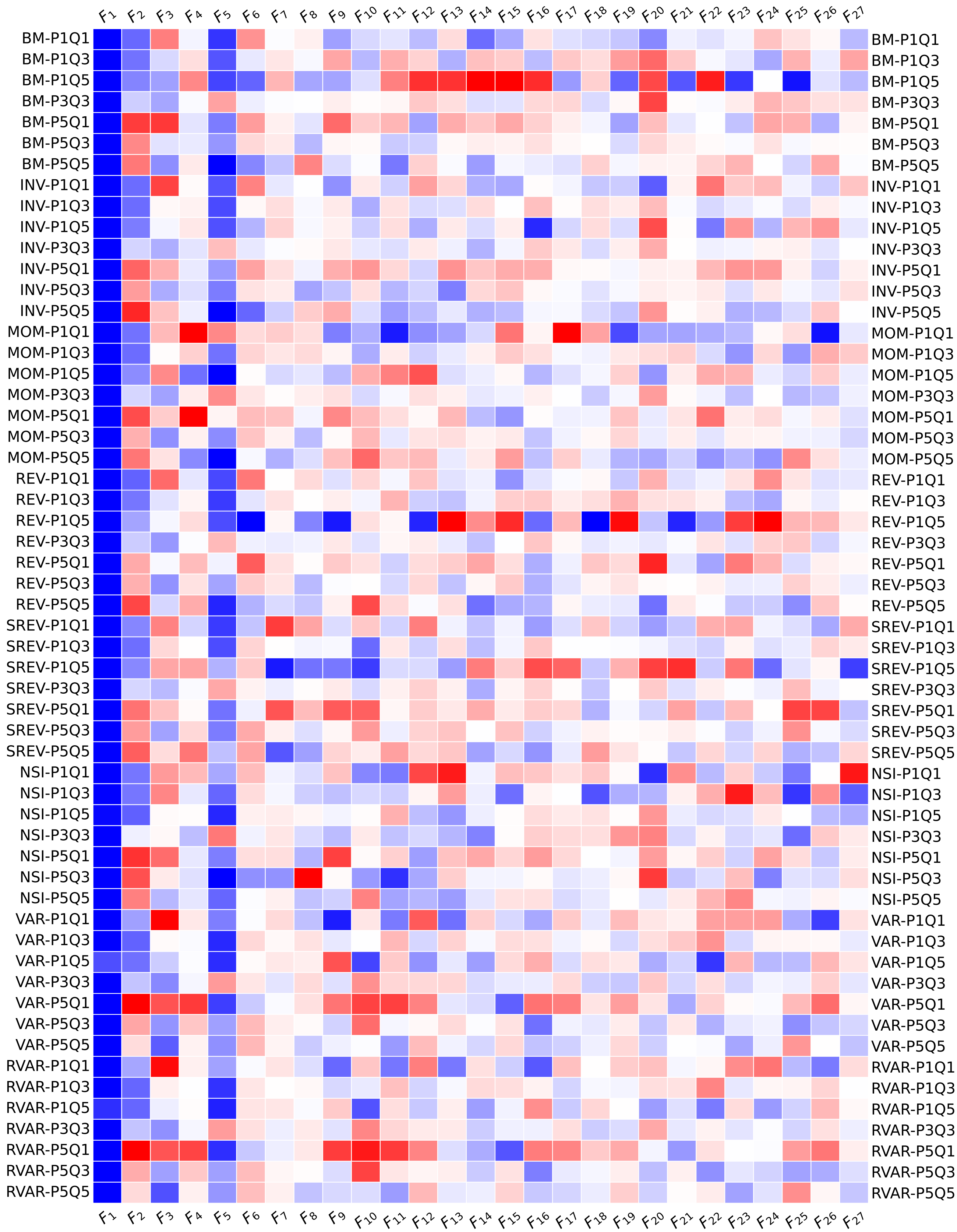
Notes: The heatmaps show the (5×5) matrices $\hat{\mathbf{W}}_{pq}^{(PQ)}$, $p, q = 1, 2, 3$, that are given by the outer product of the column vectors of $\hat{\mathbf{V}}^{(P)}$, and $\hat{\mathbf{V}}^{(Q)}$: $\mathbf{W}_{pq}^{(PQ)} = \mathbf{v}_p^{(P)} \circ \mathbf{v}_q^{(Q)}$. $\hat{\mathbf{V}}^{(P)}$, and $\hat{\mathbf{V}}^{(Q)}$ are from the estimation of a partial Tucker model (13) with $K_C = K_P = K_Q = 3$. Negative (positive) values are plotted in red (blue). The model is estimated by HOOI. The sample is from July 1963 to October 2023.

Figure 5. 3D-PCA weight matrices $\widehat{\mathbf{W}}^{3\mathrm{D}}$



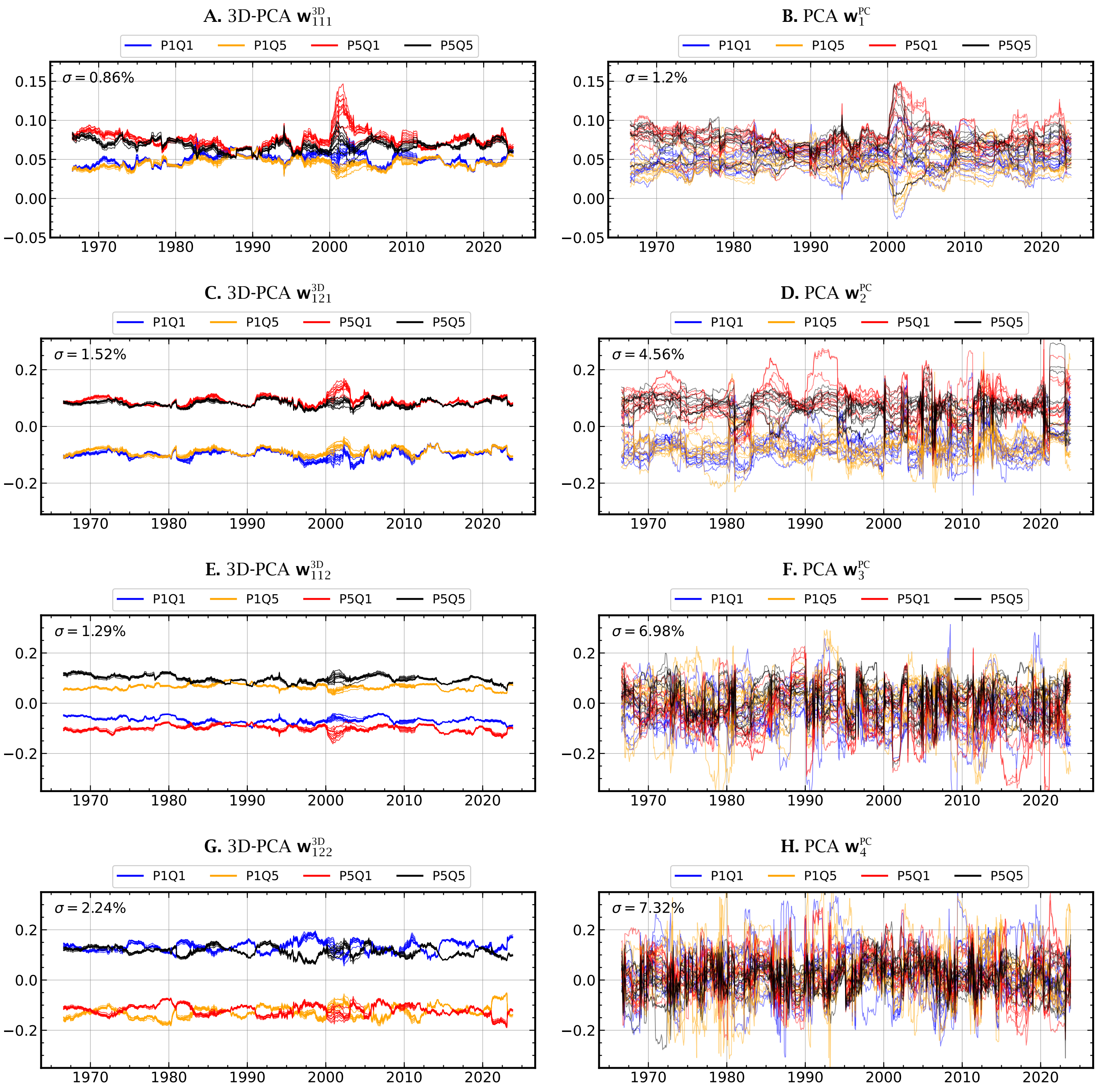
Notes: The figures show heatmaps of 3D-PCA factor weights of P1Q1, P1Q3, P1Q5, P3Q1, P3Q3, P3Q5, P51Q1, P5Q3, and P5Q5 portfolios of BM, MOM, REV, SREV, NSI, VAR, and RVAR. The factor F_{cpq} is based on the c-th column of $\mathbf{V}^{(C)^\top}$, the p-th column of $\mathbf{V}^{(P)^\top}$, and the q-th column of $\mathbf{V}^{(Q)^\top}$. Negative values are plotted in red and positive ones in blue. The underlying 3D-PCA model has $K_C = K_P = K_Q = 3$ factors. The sample is from July 1963 to October 2023.

Figure 6. PCA weight matrices $\widehat{\mathbf{W}}^{PC}$



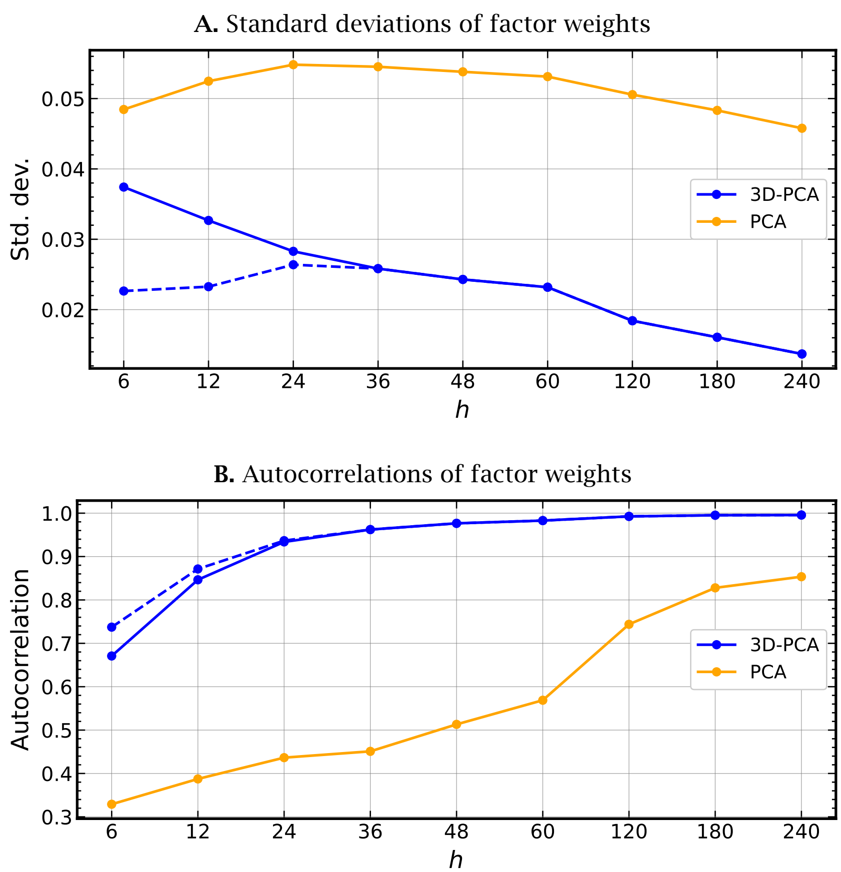
Notes: The figures show heatmaps of PCA factor weights of P1Q1, P1Q3, P1Q5, P3Q1, P3Q3, P3Q5, P51Q1, P5Q3, and P5Q5 portfolios of BM, MOM, REV, SREV, NSI, VAR, and RVAR. The factor F_{cpq} is based on the c-th column of $\mathbf{V}^{(C)^\top}$, the p-th column of $\mathbf{V}^{(P)^\top}$, and the q-th column of $\mathbf{V}^{(Q)^\top}$. Negative values are plotted in red and positive ones in blue. The underlying PCA model has 27 factors. The sample is from July 1963 to October 2023.

Figure 7. Weights – Rolling windows $h = 24$



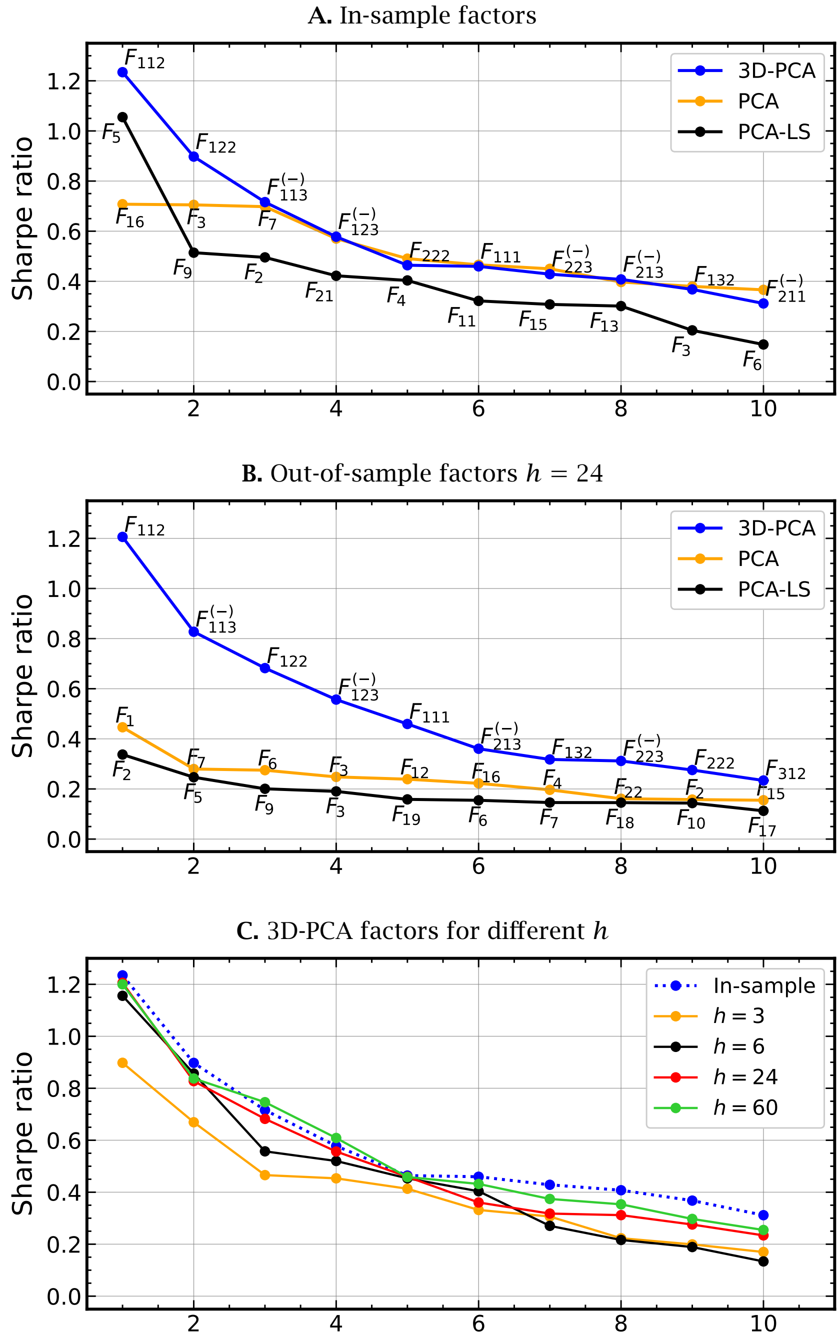
Notes: The figure plots the time series of factor weights of the P1Q1, P1Q5, P5,Q1, and P5Q5 portfolios estimated in rolling samples of length $h = 24$ of 3D-PCA factors $\mathbf{w}^{3D}_{111}, \mathbf{w}^{3D}_{112}, \mathbf{w}^{3D}_{121}$ and $\mathbf{w}^{3D}_{122}$ (left panels) and the first four PCA (right panels) factors. The 3D-PCA factor F_{cpq} is based on the c-th column of $\mathbf{V}^{(C)^\top}$, the p-th column of $\mathbf{V}^{(P)^\top}$, and the q-th column of $\mathbf{V}^{(Q)^\top}$. 3D-PCA factors are based on a partial Tucker decomposition with $K_C = K_P = K_Q = 3$ while PCA factor are based on an estimation with 27 factors. Each plot shows the time series of the factor weights of the 275 portfolios. $\bar{\sigma}$ is the average standard deviation of the 275 time series in each plot. The sample is from July 1963 to October 2023.

Figure 8. Rolling estimations for different window lengths h



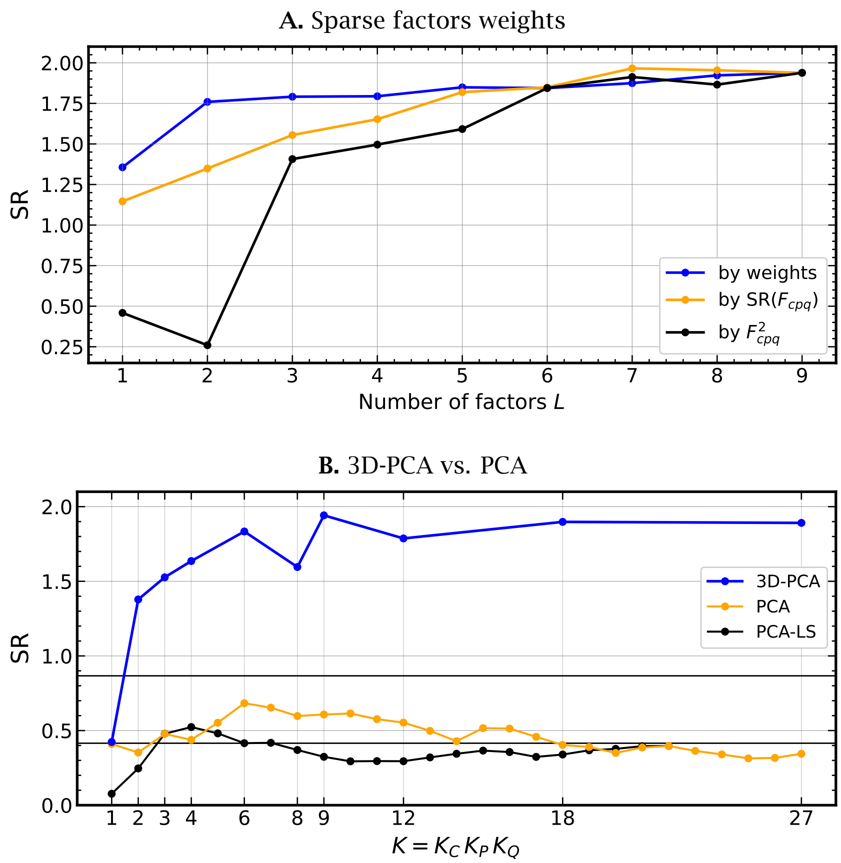
Notes: Panels A and B plot the average standard deviations and autocorrelations of the time series of 3D-PCA (in blue) and PCA (in orange) factor weights estimated in rolling samples of different lengths h, respectively. The dashed blue line shows the average standard deviation of 3D-PCA weights of the first h factors if $h < 27$. The sample is from July 1963 to December 2024.

Figure 9. Sharpe ratios of 3D-PCA and PCA factors



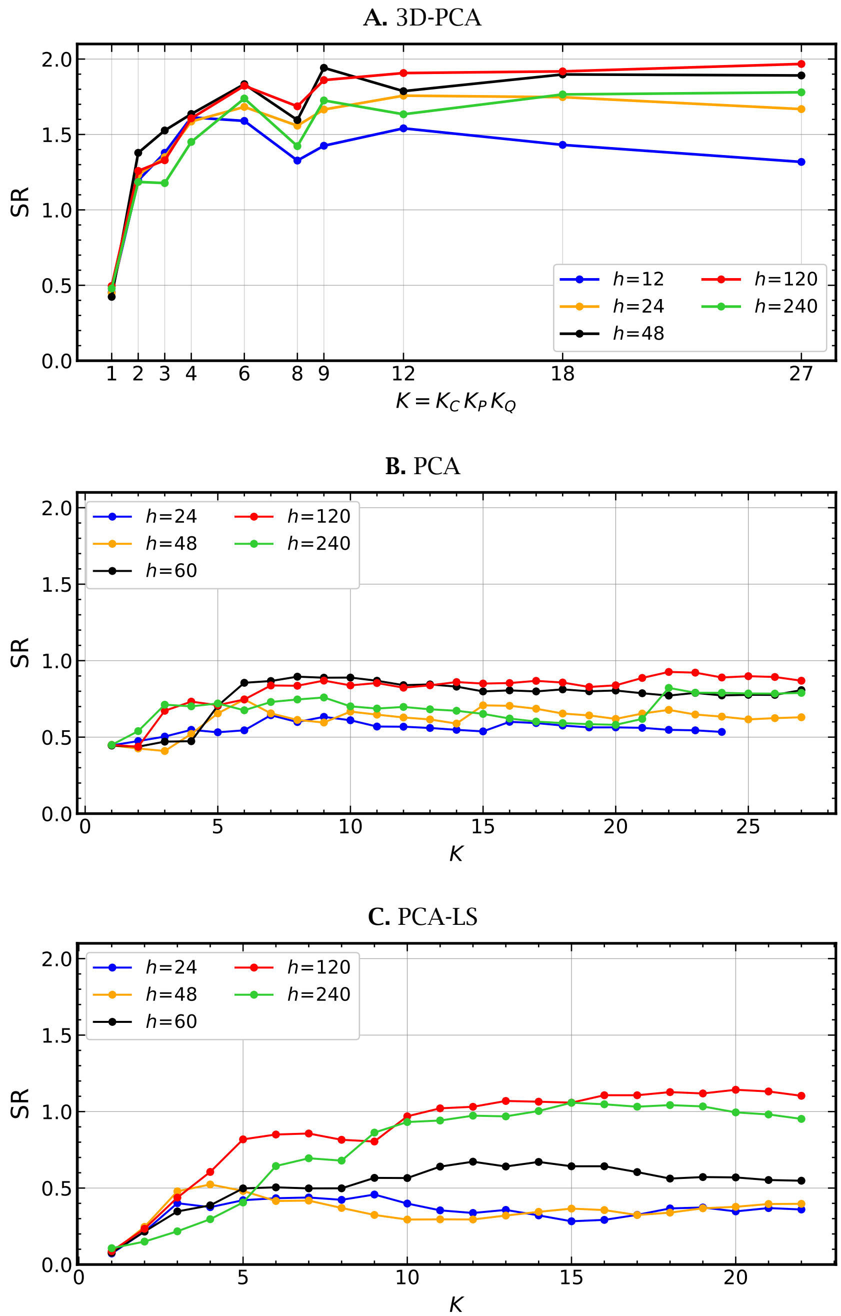
Notes: The shows Sharpe ratios of 3D-PCA, PCA, and PCA-LS in-sample and out-of-sample factors ranked by their Sharpe ratios (in absolute values). Out-of-sample factors are estimated in rolling samples of length $h = 24$ months. Panel C plots the top=10 Sharpe ratios of out-of-sample 3D-PCA factors for different window lengths. The underlying 3D-PCA model is based on $(K_C, K_P, K_Q) = (3, 3, 3)$. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB_c and HML_c portfolios. The sample is from July 1963 to October 2023.

Figure 10. Sparse factors and comparison with PCA



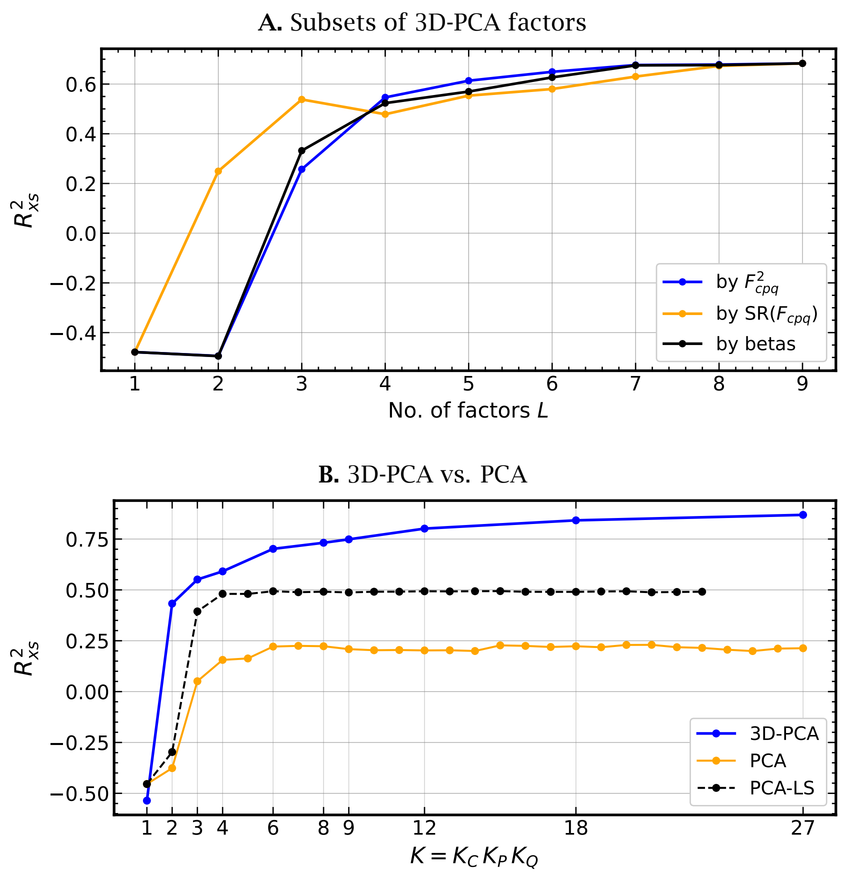
Notes: Panel A shows Sharpe ratios for 3D-PCA models with subset of L factors based on mean-variance weights. Panels B plots the Sharpe-ratios of out-of-sample 3D-PCA and PCA models. If there are several combinations of (K_C, K_P, K_Q) with the same total number of factors K, the figure shows the specification with the highest Sharpe ratio, respectively. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB_c and HML_c portfolios. The horizontal lines show the Sharpe ratios of the CRSP-VW index and the Fama-French 6-factor model. Factors are estimated out-of-sample in rolling windows of length $h = 48$ months. The sample is from July 1963 to December 2024.

Figure 11. Sharpe ratio for different window lengths



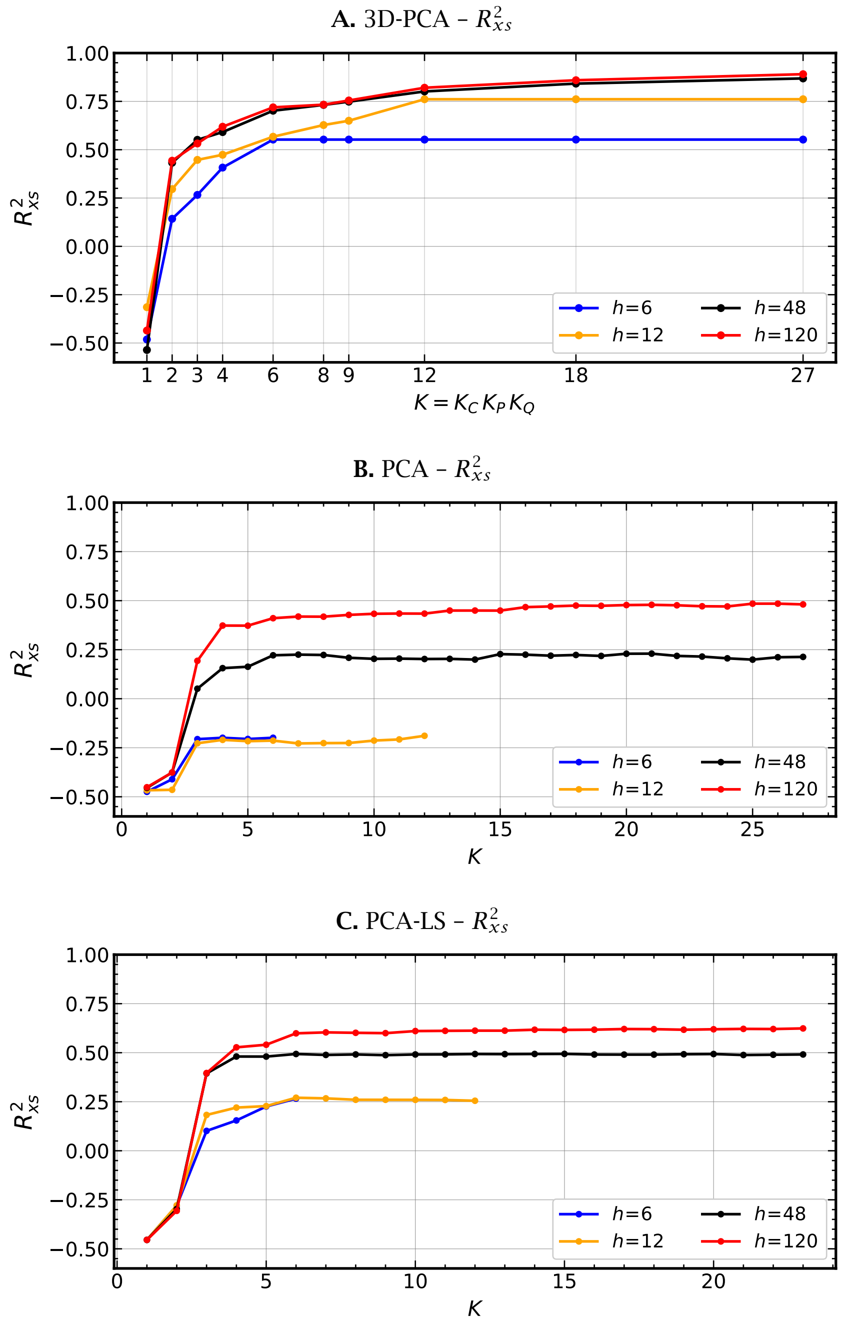
Notes: The figure shows Sharpe ratios of 3D-PCA and PCA models for different number of factors: (K_C, K_P, K_Q) for $K_C, K_P, K_Q = 1, 2, 3$. The x-axes show the total number of factors $K = K_C\, K_P\, K_Q$. If there are several combinations of (K_C, K_P, K_Q) with the same total number of factors K, the figure shows the specification with the highest Sharpe ratio. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB$_c$ and HML$_c$ portfolios. Factors are estimated out-of-sample in rolling windows of lengths $h = 12, 24, 48, 120$ months. The sample is from July 1963 to December 2024.

Figure 12. Cross-sectional R^2_{xs} of out-of-sample factors



Notes: Panel A shows the cross-sectional R^2_{xs} for subsets of factors for the 3D-PCA model with $(K_C, K_P, K_Q) = (1, 3, 3)$ factors. The factors are chosen recursively based on the best cross-sectional fit, by their Sharpe ratios, and F^2_{cpq}, respectively. Panel B shows the cross-sectional R^2_{xs} of 3D-PCA and PCA models for different number of factors. For the 3D-PCA model, (K_C, K_P, K_Q) for $K_C, K_P, K_Q = 1, 2, 3$. The x-axes show the total number of factors $K = K_C\, K_P\, K_Q$. If there are several combinations of (K_C, K_P, K_Q) with the same total number of factors K, the figure shows the specification with the highest R^2_{xs}. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB_c and HML_c portfolios. Factors are estimated out-of-sample in rolling windows of length $h = 48$ months. The sample is from July 1963 to December 2024.

Figure 13. 3D-PCA vs. PCA: Different window lengths



Notes: The figure shows the cross-sectional R^2_{xs} of 3D-PCA and PCA models for different number of factors: (K_C, K_P, K_Q) for $K_C, K_P, K_Q = 1, 2, 3$. The x-axes show the total number of factors $K = K_C\, K_P\, K_Q$. If there are several combinations of (K_C, K_P, K_Q) with the same total number of factors K, the figure shows the specification with the highest R^2_{xs}. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB_c and HML_c portfolios. Factors are estimated out-of-sample in rolling windows of lengths $h = 3, 6, 24, 60$ months. The sample is from July 1963 to December 2024.

Figure 14. (4, 2)D-PCA – $\hat{\mathbf{V}}^i$ matrices

A. $\hat{\mathbf{V}}^{(P)}$

	$\mathbf{v}_1^P$	$\mathbf{v}_2^P$	$\mathbf{v}_3^P$
1	0.34	-0.68	0.60
2	0.42	-0.34	-0.32
3	0.46	-0.08	-0.45
4	0.49	0.21	-0.26
5	0.50	0.61	0.52

B. $\hat{\mathbf{V}}^{(Q)}$

	$\mathbf{v}_1^Q$	$\mathbf{v}_2^Q$	$\mathbf{v}_3^Q$
1	0.50	-0.70	0.45
2	0.44	-0.18	-0.39
3	0.42	0.07	-0.48
4	0.42	0.29	-0.28
5	0.45	0.62	0.58

Notes: The heatmaps show estimates of the $\hat{\mathbf{V}}^{(P)}$ and $\hat{\mathbf{V}}^{(Q)}$ matrices of the (4, 2)D-PCA model with $K_P = K_Q = 3$. Negative values are plotted in red and positive ones in blue. The model is estimated by HOOI. The sample is from July 1963 to October 2023.

Appendix A.

Appendix A.1. Tensor operations

Let $\boldsymbol{\mathcal{X}}$ be a $(T\times N\times C)$-dimensional tensor. A 3-dimensional tensor can be expressed as collections of one-dimensional *fibers* and 2-dimensional *slices*. Fibers are vectors and correspond to rows and columns of a matrix, while slices are matrices. Fibers are defined by fixing every index but one so that $\boldsymbol{\mathcal{X}}$ has fibers along each mode, denoted by $\mathbf{x}_{(nc)t}, \mathbf{x}_{(tc)n}$, and $\mathbf{x}_{(tn)c}$, respectively.[24] Slices are created by fixing all but two indices and are written as $\mathbf{X}_{(t)nc}, \mathbf{X}_{(n)tc}, \mathbf{X}_{(c)tn}$.[25]

A tensor can be written as a matrix by *unfolding* one dimension. For example, unfolding $\boldsymbol{\mathcal{X}}$ along the first dimension arranges the dimension-1 fibers as columns of the unfolded matrix $\mathbf{X}_{(1)}$, which is of dimension $(T\times NC)$. Correspondingly, unfolding $\boldsymbol{\mathcal{X}}$ along the second and third dimensions yields a $(N\times TC)$-matrix $\mathbf{X}_{(2)}$ and a $(C\times TN)$-matrix $\mathbf{X}_{(3)}$, respectively.

The *inner product* of two tensors of equal dimensions is the sum of the products of the individual tensor elements as follows:

$$\langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \rangle = \sum_{t,n,c} x_{tnc}\, y_{tnc}$$

and the *norm* of $\boldsymbol{\mathcal{X}}$ is $\|\boldsymbol{\mathcal{X}}\| = \langle \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}} \rangle^{1/2}$. The *outer product* $\circ$ of two vectors $\mathbf{a} \in \mathbb{R}^T$ and $\mathbf{b} \in \mathbb{R}^N$ is defined as[26]

$$\mathbf{X} = \mathbf{a} \circ \mathbf{b} = \mathbf{a}\,\mathbf{b}^\top \in \mathbb{R}^T \times \mathbb{R}^N,$$

so that $\mathbf{X}$ is a $(T\times N)$ matrix. The *outer* product of three vectors, $\mathbf{a} \in \mathbb{R}^T, \mathbf{b} \in \mathbb{R}^N, \mathbf{c} \in \mathbb{R}^C$, yields a 3-dimensional $(T\times N\times C)$ tensor

$$\boldsymbol{\mathcal{X}} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} \in \mathbb{R}^T \times \mathbb{R}^N \times \mathbb{R}^C. \tag{A.1}$$

Tensors can be multiplied by vectors and matrices of appropriate dimensions. Since tensors have arbitrary dimensions, the mode multiplied by the matrix must be explicitly specified. The product of a tensor $\boldsymbol{\mathcal{X}}$ and a matrix $\mathbf{A}_n$ is called *n-mode multiplication*, where n specifies the mode multiplied by $\mathbf{A}_n$. For example, the mode-1 product of the $(T\times N\times C)$ tensor $\boldsymbol{\mathcal{X}}$ and the $(S\times T)$ matrix $\mathbf{A}_1$ is equal to a $(S\times N\times C)$ tensor $\boldsymbol{\mathcal{Y}}$ given by

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{A}_1.$$

The n-mode product tensor is constructed by multiplying each mode-n fiber by each row vector of $\mathbf{A}_1$. In general, the n-mode is written as $\boldsymbol{\mathcal{X}}\times_n\mathbf{A}_n$. The number of columns of $\mathbf{A}_n$ must equal the n-mode dimension of $\boldsymbol{\mathcal{X}}$ while the n-mode dimension of $\boldsymbol{\mathcal{X}}\times_n\mathbf{A}_n$ is equal to the number of rows of $\mathbf{A}_n$. The n-mode product can be chained:

$$\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{X}} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2 \times_3 \mathbf{A}_3$$

where $\mathbf{A}_2$ and $\mathbf{A}_3$ are conforming matrices. The order of the multiplications in the chain is irrelevant.

The 1-mode product of a $(2\times2\times3)$ tensor with a (5×2) matrix is illustrated in Panel A of Figure C.3. Each mode-1 fiber of $\boldsymbol{\mathcal{X}}$ is a vector of length 2 and is multiplied by each of the row vectors of $\mathbf{A}_1$ so that $\boldsymbol{\mathcal{X}}$ with mode-1 dimension T is transformed into the product tensor $\boldsymbol{\mathcal{Y}}$ with mode-1 dimension S. All other dimensions are the same. Panel C shows an example of a mode-2 product. Note that $\mathbf{A}_2$ is a (2×4) matrix but is displayed as a (4×2) matrix. It is standard practice to rotate tensors, matrices, and vectors in illustrations so that the mode dimensions match.[27]

The standard matrix products can be written in n-mode tensor notation. Let $\mathbf{X}, \mathbf{A}_1$, and $\mathbf{A}_2$ be $(T\times N)$, $(S\times T)$, and $(U\times N)$ matrices, respectively. Then $\mathbf{A}_1\,\mathbf{X} = \mathbf{X} \times_1 \mathbf{A}_1$ is a $(S\times N)$ matrix, $\mathbf{X}\,\mathbf{A}_2^\top = \mathbf{X} \times_2 \mathbf{A}_2$ is a $(T\times U)$ matrix, and $\mathbf{A}_1\,\mathbf{X}\,\mathbf{A}_2^\top = \mathbf{X} \times_1 \mathbf{A}_1 \times_2 \mathbf{A}_2$ is a $(S\times U)$ matrix.

---

[24] See Panels B, C, and D of Figure C.1.

[25] See Panels E, F, and G of Figure C.1.

[26] Panel A of Figure C.3 shows an example for $T = 5, N = 4, C = 3$.

[27] There is no "transpose" operator for tensors, and it may be helpful to think about tensor multiplications without the notion of a matrix transpose.

Appendix A.2. The Singular Value Decomposition (SVD) of a matrix

Let $\mathbf{X}$ be a $(T{\times}N)$ data matrix with TN observations x_{tn}.[28] The SVD of $\mathbf{X}$ is given by

$$\mathbf{X} = \mathbf{U}^{(1)}\,\mathbf{H}\,\mathbf{U}^{(2)^\top} = \sum_{i=1}^{\min(T,N)} h_i\,\mathbf{u}_i^{(1)}\,\mathbf{u}_i^{(2)^\top}, \tag{A.2}$$

where $\mathbf{U}^{(1)}$ is a $(T{\times}T)$ matrix of eigenvectors $\mathbf{u}_t^{(1)}$ of $\mathbf{X}\mathbf{X}^\top$ as columns, $\mathbf{U}^{(2)}$ is a $(N{\times}N)$ matrix of eigenvectors $\mathbf{u}_t^{(2)}$ of $\mathbf{X}^\top\,\mathbf{X}$ as columns, and $\mathbf{H}$ is a diagonal $(T{\times}N)$ matrix with diagonal elements h_i that are the square roots of non-zero eigenvalues of $\mathbf{X}\mathbf{X}^\top$. The eigenvalues are in descending order and the eigenvectors in $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are ordered accordingly.

The SVD of $\mathbf{X}$ implies a *factor representation*

$$\mathbf{X} = \mathbf{F}_N\,\mathbf{B}_N^\top, \tag{A.3}$$

where $\mathbf{F}_N = \mathbf{X}\mathbf{U}^{(\mathbf{2})} = \mathbf{U}^{(1)}\mathbf{H}$ and $\mathbf{B}_N = \mathbf{U}^{(2)}$ are of dimensions $(T{\times}N)$ and $(N{\times}N)$, respectively. The columns of $\mathbf{F}_N$ are *factors*, and the columns of $\mathbf{B}_N$ are *factor loadings*. Factor models (A.3) are not unique and can be rotated by any nonsingular $(N{\times}N)$ matrix $\mathbf{S}$: $\mathbf{X} = \mathbf{F}_N\,\mathbf{S}\,\mathbf{S}^{-1}\,\mathbf{B}_N^\top$. The standard normalization assumes that $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are orthonormal and that $\mathbf{H}$ is diagonal.

It is also possible to compute the SVD of the $(N{\times}T)$ matrix $\mathbf{X}^\top$ instead of $\mathbf{X}$. The isomorphic factor representation for $\mathbf{X}^\top$ is given by $\mathbf{X}^\top = \mathbf{F}_T\,\mathbf{B}_T^\top$, where $\mathbf{F}_T = \mathbf{X}^\top\mathbf{U}^{(\mathbf{1})} = \mathbf{U}^{(2)}\mathbf{H}^\top$ and $\mathbf{B}_T = \mathbf{U}^{(1)}$. The representations are equivalent, but the roles of $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are reversed so that factors of the SVD of $\mathbf{X}$ become factor loadings in the SVD of $\mathbf{X}^\top$, and *vice versa*.

Suppose we want to approximate $\mathbf{X}$ by a matrix $\hat{\mathbf{X}}_K$ that can be written in terms of lower-dimensional matrices such that

$$\mathbf{X} = \hat{\mathbf{X}}_K + \mathbf{E}_K, \tag{A.4}$$

$$\text{where }\ \hat{\mathbf{X}}_K = \mathbf{U}_K^{(1)}\,\mathbf{H}_K\,\mathbf{U}_K^{(2)^\top}, \tag{A.5}$$

and $\mathbf{H}_K, \mathbf{U}_K^{(1)}, \mathbf{U}_K^{(2)}$ are $(K{\times}K), (T{\times}K), (N{\times}K)$ matrices. The optimal $\hat{\mathbf{X}}_K$ minimizes the mean-squared-error (MSE)

$$\mathrm{MSE}(\hat{\mathbf{X}}_K) = \frac{1}{TN}\,\|\mathbf{E}_K\|^2,$$

where $\|\mathbf{E}\| = \sqrt{\sum_{t,n} e_{tn}^2}$ is the Frobenius matrix norm. Eckart and Young (1936) showed that the solution is given by the *truncated SVD*, *i.e.*, setting $\mathbf{H}_K$ to the first K rows and columns of $\mathbf{H}$ and $\mathbf{U}_K^{(1)}, \mathbf{U}_K^{(2)}$ to the first K columns of $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}$:

$$\hat{\mathbf{X}}_K = \mathbf{U}_K^{(1)}\,\mathbf{H}_K\,\mathbf{U}_K^{(2)^\top}. \tag{A.6}$$

The truncated SVD (A.6) is equivalent to the K-factor model

$$\mathbf{X} = \mathbf{F}_K\,\mathbf{B}_K^\top + \mathbf{E}_K, \tag{A.7}$$

where $\mathbf{F}_K = \mathbf{U}_K^{(1)}\,\mathbf{H}_K$ and $\mathbf{B}_K = \mathbf{U}_K^{(2)}$ are $(T{\times}K)$ and $(N{\times}K)$ matrices, respectively. Thus, the truncated SVD equals the first K principal components of $\mathbf{X}^\top\mathbf{X}$. I will refer to this model as SVD-PCA throughout the paper.

The truncated SVD has an alternative representation useful for understanding tensor decompositions. $\mathbf{U}_K^{(1)}\,\mathbf{H}_K\,\mathbf{U}_K^{(2)^\top}$ is equivalent to the weighted sum of the outer products of the column vectors of $\mathbf{U}_K^{(1)}$ and the row vectors of $\mathbf{U}_K^{(2)^\top}$. This can be seen by writing (A.6) as

$$\hat{\mathbf{X}}_K = \sum_{t=1}^{K}\sum_{n=1}^{K} h_{tn}\,\underbrace{\mathbf{u}_t^{(1)}\,\mathbf{u}_n^{(2)^\top}}_{T\times N} \tag{A.8}$$

$$= \sum_{k=1}^{K} h_{kk}\,\mathbf{u}_k^{(1)}\,\mathbf{u}_k^{(2)^\top}. \tag{A.9}$$

The second equality follows from the fact that $\mathbf{H}_K$ is a diagonal matrix. $\hat{\mathbf{X}}_K$ is the weighted sum of K matrices with dimensions $(T{\times}N)$, which are the outer vector product of the eigenvectors $\mathbf{u}_k^{(1)}$ and $\mathbf{u}_k^{(2)^\top}$ of $\mathbf{X}\mathbf{X}^\top$ and $\mathbf{X}^\top\mathbf{X}$,

[28] The row index t is generic and does not necessarily have to be a "time" index.

respectively. Each k in the summation corresponds to a factor in the K-factor representation (A.7). The advantage of representation (A.8) is that it shows the contribution of each of the K factors in the model's fit. Since the eigenvectors are normalized, the K outer vector products $\mathbf{u}_k^{(1)}\,\mathbf{u}_k^{(2)^\top}$ are of the same magnitude, so the weight of the contribution of each factor k is approximately equal to the k-th eigenvalue.

In the truncated SVD (A.4)-(A.6) the number of factors is K. Note that we could define an asymmetric SVD that has different numbers of factors for the two dimensions:

$$\hat{\mathbf{X}}_{(K_1,K_2)} = \mathbf{U}_{K_1}^{(1)}\,\mathbf{H}_{K_1,K_2}\,\mathbf{U}_{K_2}^{(2)^\top}, \tag{A.10}$$

where $\mathbf{H}_{K_1,K_2}, \mathbf{U}_{K_1}^{(1)}, \mathbf{U}_{K_2}^{(2)}$ are $(K_1{\times}K_2), (T{\times}K_1), (N{\times}K_2)$ matrices. However, since $\mathbf{H}_{K_1,K_2}$ is diagonal, the asymmetric SVD reduces to a K-factor SVD where $K = \min(K_1, K_2)$. In contrast to the 2-dimensional matrix SVD, the core tensor $\boldsymbol{\mathcal{G}}$ in the Tucker decomposition is *not* diagonal. Consequently, the number of factors can differ by dimension.[29]

Appendix A.3. Higher-Order Orthogonal Iteration (HOOI)

The objective is to find $\boldsymbol{\mathcal{F}}$ and orthonormal $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}, \mathbf{V}^{(Q)}$ such that

$$\|\boldsymbol{\mathcal{E}}\| = \|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{F}} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)} \times_4 \mathbf{V}^{(Q)}\|$$

is minimized. Given the loading matrices $\mathbf{V}^{(i)}$, the optimal core tensor $\boldsymbol{\mathcal{F}}$ satisfies

$$\boldsymbol{\mathcal{F}} = \boldsymbol{\mathcal{X}} \times_2 \mathbf{V}^{(C)\top} \times_3 \mathbf{V}^{(P)\top} \times_4 \mathbf{V}^{(Q)\top}. \tag{A.11}$$

Since the $\mathbf{V}^{(i)}$ matrices are orthonormal, the squared norm of the approximation error $\boldsymbol{\mathcal{E}} = \boldsymbol{\mathcal{X}} - \hat{\boldsymbol{\mathcal{X}}}$ can be written as

$$\begin{aligned}
\|\boldsymbol{\mathcal{E}}\|^2 &= \|\boldsymbol{\mathcal{X}}\|^2 - 2\langle \boldsymbol{\mathcal{X}},\, \boldsymbol{\mathcal{F}} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)} \times_4 \mathbf{V}^{(Q)}\rangle + \|\boldsymbol{\mathcal{F}} \times_2 \mathbf{V}^{(C)} \times_3 \mathbf{V}^{(P)} \times_4 \mathbf{V}^{(Q)}\|^2 && \text{(A.12)}\\
&= \|\boldsymbol{\mathcal{X}}\|^2 - 2\langle \boldsymbol{\mathcal{X}} \times_2 \mathbf{V}^{(C)\top} \times_3 \mathbf{V}^{(P)\top} \times_4 \mathbf{V}^{(Q)\top},\, \boldsymbol{\mathcal{F}}\rangle + \|\boldsymbol{\mathcal{F}}\|^2 && \text{(A.13)}\\
&= \|\boldsymbol{\mathcal{X}}\|^2 - 2\langle \boldsymbol{\mathcal{F}},\, \boldsymbol{\mathcal{F}}\rangle + \|\boldsymbol{\mathcal{F}}\|^2 && \text{(A.14)}\\
&= \|\boldsymbol{\mathcal{X}}\|^2 - \|\boldsymbol{\mathcal{F}}\|^2 && \text{(A.15)}\\
&= \|\boldsymbol{\mathcal{X}}\|^2 - \|\boldsymbol{\mathcal{X}} \times_2 \mathbf{V}^{(C)\top} \times_3 \mathbf{V}^{(P)\top} \times_4 \mathbf{V}^{(Q)\top}\|^2. && \text{(A.16)}
\end{aligned}$$

Suppose we know $\mathbf{V}^{(C)}$ and $\mathbf{V}^{(P)}$. Then $\mathbf{V}^{(Q)}$ can be obtained as

$$\max_{\mathbf{V}^{(Q)}} \|\boldsymbol{\mathcal{X}} \times_2 \mathbf{V}^{(C)\top} \times_3 \mathbf{V}^{(P)\top} \times_4 \mathbf{V}^{(Q)\top}\|. \tag{A.17}$$

This maximization problem can be rewritten in matrix form as

$$\max_{\mathbf{V}^{(Q)}} \|\mathbf{V}^{(Q)\top}\mathbf{W}_Q\| \tag{A.18}$$

$$\text{where } \mathbf{W}_Q = \mathbf{X}_{(Q)}(\mathbf{V}^{(C)} \otimes \mathbf{V}^{(P)}). \tag{A.19}$$

The optimal $\mathbf{V}^{(Q)}$ is given by the first K_Q left singular vectors of $\mathbf{W}_Q$. Since one $\mathbf{V}^{(i)}$ can be computed if the other two are known, we can use the following iterative algorithm known as Higher-Order Orthogonal Iteration (HOOI):

1. Pick initial values for $\mathbf{V}^{(C)}, \mathbf{V}^{(P)}$.
2. Compute $\mathbf{V}^{(Q)}$ as the first K_Q left singular vectors of $\mathbf{X}_{(Q)}(\mathbf{V}^{(C)} \otimes \mathbf{V}^{(P)})$.
3. Compute $\mathbf{V}^{(C)}$ as the first K_C left singular vectors of $\mathbf{X}_{(C)}(\mathbf{V}^{(P)} \otimes \mathbf{V}^{(Q)})$.
4. Compute $\mathbf{V}^{(P)}$ as the first K_P left singular vectors of $\mathbf{X}_{(P)}(\mathbf{V}^{(C)} \otimes \mathbf{V}^{(Q)})$.
5. Repeat Steps 2 to 4 until a convergence criterion is satisfied.
6. Compute $\boldsymbol{\mathcal{F}} = \boldsymbol{\mathcal{X}} \times_2 \mathbf{V}^{(C)\top} \times_3 \mathbf{V}^{(P)\top} \times_4 \mathbf{V}^{(Q)\top}$.

The literature has developed numerous numerical improvements of the HOOI estimator; see, for example, Andersson and Bro (1998). Several other algorithms exist, including nonlinear Newton-Grassmann optimization (Elden and Savas (2009)). Starting values of the $\mathbf{V}^{(i)}$ matrices can be set by HOSVD.

---

[29] The CP tensor decomposition is a special case of the Tucker decomposition and restricts the core tensor $\boldsymbol{\mathcal{G}}$ to be diagonal, which implies that the number of factors is the same, $K_i = K$.

The HOOI estimator converges after 20 to 40 iterations for the data set used in this paper. In addition to setting the initial $\mathbf{V}^{(i)}$ using the method described above, I also choose initial values randomly. The numerical computations are robust and converge to the same optimum.

Appendix A.4. Estimation of 2D-PCA model

This section presents the estimated parameters of the 2D-PCA model in Section 3. The test assets are combinations of the first, third, and fifth size and book-to-market portfolios from the 5-by-5 quintile sorts, resulting in $3\times 3 = 9$ portfolios. The 2D-PCA model is estimated by HOOI with $K_P = 2$ size factors and $K_Q = 2$ book-to-market factors. Note that the estimation normalizes weights to have unit norm. For comparison with the example in Section 3, the weights are rescaled so that the absolute values of the weights of the corner portfolios have a mean of one.

$$\mathbf{V}^{(P)} = \begin{pmatrix} 0.75 & -1.13 \\ 1.05 & -0.18 \\ 1.20 & 0.87 \end{pmatrix} \qquad \mathbf{V}^{(Q)} = \begin{pmatrix} 1.11 & -1.01 \\ 0.90 & 0.15 \\ 0.99 & 0.99 \end{pmatrix}.$$

The columns of the estimated $\mathbf{V}^{(P)}$ and $\mathbf{V}^{(Q)}$ matrices have the familiar "level" and "slope" patterns of the Fama-French factors. The implied $\mathbf{W}_{ij}$ matrices

$$\mathbf{W}_{11} = \begin{pmatrix} 0.57 & 0.79 & 0.90 \\ 0.79 & 1.10 & 1.26 \\ 0.90 & 1.26 & 1.44 \end{pmatrix} \qquad \mathbf{W}_{12} = \begin{pmatrix} -0.85 & -0.14 & 0.65 \\ -1.18 & -0.19 & 0.91 \\ -1.36 & -0.22 & 1.04 \end{pmatrix}$$

$$\mathbf{W}_{21} = \begin{pmatrix} -0.85 & -1.18 & -1.36 \\ -0.14 & -0.19 & -0.22 \\ 0.65 & 0.91 & 1.04 \end{pmatrix} \qquad \mathbf{W}_{22} = \begin{pmatrix} 1.28 & 0.20 & -0.98 \\ 0.20 & 0.03 & -0.16 \\ -0.98 & -0.16 & 0.75 \end{pmatrix}$$

have the same patterns as in the example in Section 3: $\mathbf{W}_{11}$ is long-only, $\mathbf{W}_{21}$ has an SMB pattern, and $\mathbf{W}_{12}$ has an HML pattern.

Appendix A.5. Machine learning methods

I follow Gu, Kelly, and Xiu (2020)'s implementation of machine learning methods. Let $r_{i,t}$ be the excess return of stock i at time t and $\mathbf{z}_{i,t}$ be a vector of characteristics. The expected excess return is modeled as

$$\mathrm{E}[r_{i,t+1} | \mathbf{z}_{i,t}] = g(\mathbf{z}_{i,t}). \tag{A.20}$$

In each month t, stocks are sorted by the estimated expected excess return $\hat{g}(\mathbf{z}_{i,t})$ into 10 decile portfolios. The Sharpe ratio is based on realized excess returns of the decile-10 minus decile-1 long-short portfolio. Gu, Kelly, and Xiu (2020) use the following methods to estimate $g(\mathbf{z}_{i,t})$; see their paper for details:

1. Linear ridge: $g(\mathbf{z}_{i,t})$ is linear with L_2 regularization
2. Linear elastic net: $g(\mathbf{z}_{i,t})$ is linear with L_1 and L_2 regularization
3. Boosted regression trees: $g(\mathbf{z}_{i,t})$ is estimated using gradient boosting
4. Random forest: $g(\mathbf{z}_{i,t})$ is estimated using an ensemble of decision trees
5. Neural net: $g(\mathbf{z}_{i,t})$ is estimated using a feed-forward neural network

The split training/validation/test samples are of length 300/60/364 months. The hyperparameters of the machine learning methods are set to the values used by Gu, Kelly, and Xiu (2020). The data are from the Global Factor Data website (`jkpfactors.com`).

Table B.1: Number of parameters

(K_C, K_P, K_Q)	T	$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	3D-PCA	SVD
(2, 2, 2)	25	33	2,164	230	2,336
	50	33	2,164	430	2,536
	100	33	2,164	830	2,936
	250	33	2,164	2,030	4,136
	750	33	2,164	6,030	8,136
(3, 3, 3)	25	45	7,047	711	7,371
	50	45	7,047	1,386	8,046
	100	45	7,047	2,736	9,396
	250	45	7,047	6,786	13,446
	750	45	7,047	20,286	26,946
(4, 2, 2)	25	48	4,264	440	4,544
	50	48	4,264	840	4,944
	100	48	4,264	1,640	5,744
	250	48	4,264	4,040	8,144
	750	48	4,264	12,040	16,144

Note: This table reports the number of parameters in the weight matrices and factor matrices for both 3D-PCA and standard PCA methods for different combinations of factor numbers (K_C, K_P, K_Q) and sample sizes T. The dimensions of the data tensor are $(T, 11, 5, 5)$.

Table B.2: Monte Carlo simulations: $\sigma_e = 1$

(K_C, K_P, K_Q)	T	RMSE				Correlations			
		$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$	$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$
(2, 2, 2)	25	0.010	0.064	0.213	0.679	0.944	0.358	0.929	0.519
	50	0.007	0.057	0.159	0.445	0.979	0.497	0.940	0.617
	100	0.005	0.048	0.126	0.292	0.987	0.637	0.946	0.710
	250	0.003	0.037	0.097	0.174	0.995	0.783	0.949	0.802
	750	0.002	0.027	0.073	0.107	0.998	0.882	0.951	0.863
(3, 3, 3)	25	0.024	0.078	0.274	0.780	0.889	0.132	0.794	0.198
	50	0.017	0.076	0.201	0.581	0.943	0.178	0.845	0.236
	100	0.011	0.074	0.155	0.430	0.975	0.236	0.872	0.272
	250	0.007	0.069	0.117	0.297	0.990	0.329	0.884	0.337
	750	0.004	0.062	0.087	0.199	0.997	0.460	0.889	0.435
(4, 2, 2)	25	0.034	0.075	0.331	0.779	0.793	0.196	0.753	0.300
	50	0.025	0.071	0.233	0.555	0.882	0.271	0.826	0.350
	100	0.019	0.067	0.173	0.404	0.927	0.355	0.865	0.404
	250	0.011	0.061	0.117	0.278	0.976	0.472	0.906	0.483
	750	0.006	0.050	0.083	0.172	0.993	0.643	0.919	0.620

Note: This table shows results of 1,000 Monte Carlo simulations based on the data generating process in equations (25) and (26) with idiosyncratic error standard deviation $\sigma_e = 1$ for different combinations of factor numbers (K_C, K_P, K_Q) and sample sizes T. The dimensions of the data tensor are $(T, 11, 5, 5)$. The table reports root mean squared errors (RMSE) and correlations between the true and estimated factors for both 3D-PCA and standard PCA methods.

Table B.3. Monte Carlo simulations under misspecification: $\sigma_w = 0.5$

(K_C, K_P, K_Q)	T	RMSE				Correlations			
		$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$	$\mathbf{W}^{3D}$	$\mathbf{W}^{PC}$	$\mathbf{F}^{3D}$	$\mathbf{F}^{PC}$
(2, 2, 2)	25	0.029	0.051	0.176	0.485	0.810	0.622	0.964	0.604
	50	0.028	0.044	0.126	0.327	0.819	0.721	0.975	0.709
	100	0.028	0.038	0.095	0.219	0.824	0.794	0.981	0.786
	250	0.028	0.030	0.072	0.127	0.826	0.866	0.984	0.861
	750	0.028	0.023	0.053	0.070	0.827	0.915	0.986	0.913
(3, 3, 3)	25	0.036	0.075	0.203	0.535	0.792	0.219	0.872	0.210
	50	0.032	0.071	0.136	0.425	0.836	0.295	0.922	0.280
	100	0.030	0.069	0.097	0.334	0.856	0.352	0.946	0.342
	250	0.029	0.064	0.069	0.244	0.867	0.432	0.958	0.425
	750	0.028	0.057	0.050	0.166	0.870	0.551	0.963	0.546
(4, 2, 2)	25	0.042	0.068	0.273	0.544	0.709	0.352	0.809	0.337
	50	0.037	0.065	0.188	0.421	0.764	0.418	0.869	0.407
	100	0.033	0.062	0.126	0.326	0.810	0.475	0.922	0.469
	250	0.030	0.057	0.080	0.233	0.840	0.556	0.956	0.552
	750	0.029	0.045	0.053	0.140	0.854	0.715	0.972	0.711

Note: This table shows results of 1,000 Monte Carlo simulations based on the misspecified data generating process in equations (27) and (28) with $\sigma_e = 0.15$ and $\sigma_w = 0.5$ for different combinations of factor numbers (K_C, K_P, K_Q) and sample sizes T. The dimensions of the data tensor are $(T, 11, 5, 5)$. The table reports root mean squared errors (RMSE) and correlations between the true and estimated factors for both 3D-PCA and standard PCA methods.

Table B.4. Curvature portfolios

c	Mean SMB$_c$	Mean HML$_c$	Std. dev. SMB$_c$	Std. dev. HML$_c$	Sharpe ratio SMB$_c$	Sharpe ratio HML$_c$
AC	-0.99	-0.41	5.45	4.89	-0.18	-0.08
BETA	-0.82	-1.42	5.03	4.38	-0.16	-0.32
BM	-1.33	-0.22	4.95	5.24	-0.27	-0.04
INV	-0.82	-1.29	4.91	6.35	-0.17	-0.20
MOM	-0.61	-0.83	4.93	7.65	-0.12	-0.11
NSI	-0.96	-3.57	5.36	4.14	-0.18	-0.86
OP	-0.83	-0.82	5.08	5.83	-0.16	-0.14
REV	-1.23	-0.51	4.77	6.44	-0.26	-0.08
RVAR	-1.34	-3.74	5.02	5.45	-0.27	-0.69
SREV	-1.63	-1.63	4.88	6.60	-0.33	-0.25
VAR	-1.36	-3.72	4.96	5.77	-0.27	-0.64
Mean	-1.08	-1.65	5.03	5.70	-0.22	-0.31

Note: The table reports annualized means, standard deviations, and Sharpe ratios of long-short curvature portfolios. SMB$_c$ is the sum of the 0.5 times the small stock portfolios (P5) and 0.5 times the big stock portfolios (P1) minus the mid-size stock portfolios (P3) of the double sort of characteristic c averaged of all characteristic quintiles. HML$_c$ is constructed accordingly. The sample is from July 1963 to October 2023.

Table B.5. Largest and smallest weights of high-SR 3D-PCA factors

	Factor F^{3D}_{112}		Factor F^{3D}_{122}		Factor F^{3D}_{113}	
	Portfolio	Weight	Portfolio	Weight	Portfolio	Weight
1	(NSI, P5, Q5)	0.09	(NSI, P1, Q1)	0.15	(NSI, P5, Q5)	0.09
2	(VAR, P5, Q5)	0.09	(VAR, P1, Q1)	0.14	(VAR, P5, Q5)	0.09
3	(RVAR, P5, Q5)	0.09	(RVAR, P1, Q1)	0.14	(RVAR, P5, Q5)	0.09
4	(AC, P5, Q5)	0.09	(AC, P1, Q1)	0.14	(AC, P5, Q5)	0.09
5	(MOM, P5, Q5)	0.09	(MOM, P1, Q1)	0.14	(MOM, P5, Q5)	0.09
1	(NSI, P5, Q1)	−0.11	(NSI, P5, Q1)	−0.14	(NSI, P5, Q3)	−0.07
2	(VAR, P5, Q1)	−0.11	(VAR, P5, Q1)	−0.14	(VAR, P5, Q3)	−0.07
3	(RVAR, P5, Q1)	−0.11	(RVAR, P5, Q1)	−0.14	(RVAR, P5, Q3)	−0.07
4	(AC, P5, Q1)	−0.11	(AC, P5, Q1)	−0.13	(AC, P5, Q3)	−0.07
5	(MOM, P5, Q1)	−0.11	(MOM, P5, Q1)	−0.13	(MOM, P5, Q3)	−0.07

Note: The table reports the largest and smallest weights of the 3D-PCA factors with the highest Sharpe ratios, $F^{3D}_{112}, F^{3D}_{122}$, and F^{3D}_{113}. The sample is from July 1963 to October 2023.

Table B.6. Largest and smallest weights of high-SR PCA factors

	Factor F^{PC}_{16}		Factor F^{PC}_{3}		Factor F^{PC}_{7}	
	Portfolio	Weight	Portfolio	Weight	Portfolio	Weight
1	(AC, P1, Q5)	0.20	(RVAR, P5, Q5)	0.11	(SREV, P2, Q5)	0.33
2	(INV, P1, Q5)	0.19	(RVAR, P4, Q5)	0.11	(SREV, P3, Q5)	0.30
3	(RVAR, P1, Q1)	0.15	(VAR, P5, Q5)	0.10	(SREV, P1, Q5)	0.28
4	(OP, P3, Q1)	0.14	(VAR, P4, Q4)	0.10	(SREV, P4, Q5)	0.25
5	(AC, P3, Q5)	0.14	(BM, P3, Q5)	0.10	(SREV, P5, Q5)	0.20
1	(BETA, P1, Q5)	−0.25	(VAR, P1, Q1)	−0.16	(SREV, P1, Q1)	−0.23
2	(BM, P1, Q5)	−0.19	(RVAR, P1, Q1)	−0.16	(SREV, P4, Q1)	−0.23
3	(SREV, P1, Q5)	−0.16	(BETA, P1, Q5)	−0.14	(SREV, P3, Q1)	−0.22
4	(BETA, P1, Q4)	−0.13	(VAR, P2, Q1)	−0.14	(SREV, P2, Q1)	−0.22
5	(BETA, P2, Q5)	−0.13	(RVAR, P2, Q1)	−0.13	(SREV, P5, Q1)	−0.21

Note: The table reports the largest and smallest weights of the PCA factors with the highest Sharpe ratios, F^{PC}_{16}, F^{PC}_{3}, and F^{PC}_{7}. The sample is from July 1963 to October 2023.

Table B.7. p-values of Sharpe ratio tests

	$K_Q = 1$			$K_Q = 2$			$K_Q = 3$		
	$K_P = 1$	$K_P = 2$	$K_P = 3$	$K_P = 1$	$K_P = 2$	$K_P = 3$	$K_P = 1$	$K_P = 2$	$K_P = 3$
Panel A: Jobson-Korkie-Memmel test									
$K_C = 1$	0.00	0.03	0.74	0.00	0.00	0.00	0.00	0.00	0.00
$K_C = 2$	0.39	0.90	0.05	0.00	0.00	0.00	0.00	0.00	0.00
$K_C = 3$	0.65	0.15	0.22	0.00	0.00	0.00	0.00	0.00	0.00
Panel B: HAC test									
$K_C = 1$	0.00	0.03	0.76	0.00	0.00	0.00	0.00	0.00	0.00
$K_C = 2$	0.42	0.90	0.06	0.00	0.00	0.00	0.00	0.00	0.00
$K_C = 3$	0.73	0.14	0.21	0.00	0.00	0.00	0.00	0.00	0.00
Panel C: Ledoit-Wolf bootstrap test									
$K_C = 1$	0.00	0.03	0.75	0.00	0.00	0.00	0.00	0.00	0.00
$K_C = 2$	0.38	0.93	0.04	0.00	0.00	0.00	0.00	0.00	0.00
$K_C = 3$	0.73	0.17	0.21	0.00	0.00	0.00	0.00	0.00	0.00

Note: The table shows p-values for the null hypothesis of equal Sharpe ratios for 3D-PCA and PCA factors with the same number of factors. The three panels report results for the Jobson and Korkie (1981) test (corrected by Memmel (2003)), and the HAC (without pre-whitening) and the Bootstrap-TS tests proposed in Ledoit and Wolf (2008) test. The sample is from July 1963 to October 2023. The sample is from July 1963 to October 2023.

Table B.8. Sharpe ratios for different window lengths for estimation and mean-variance weights

	h_{MV}				
h_{3D}	24	36	48	60	120
24	1.67	1.76	1.96	1.93	1.89
36	1.71	1.78	1.98	1.95	1.91
48	1.73	1.77	1.94	1.90	1.91
60	1.71	1.75	1.92	1.87	1.89
120	1.61	1.68	1.85	1.80	1.86

Note: The table shows out-of-sample Sharpe ratios for models with 3D-PCA factors for combinations of window lengths for factor estimation h_{3D} and mean-variance weights h_{MV}. The sample is from July 1963 to October 2023.

Table B.9: Characteristics description of single-sorted portfolios

Acronym	Description	Acronym	Description
AbnormalAccruals	Abnormal Accruals	IntMom	Intermediate Momentum
Accruals	Accruals	InvestPPEInv	change in ppe and inv/assets
AnnouncementReturn	Earnings announcement return	InvGrowth	Inventory Growth
AssetGrowth	Asset growth	IO ShortInterest	Inst own among high short interest
BetaTailRisk	Tail risk beta	Mom12m	Momentum (12 month)
BidAskSpread	Bid-ask spread	Mom12mOffSeason	Momentum without the seasonal part
BM	Book to market, (Stattman 1980)	Mom6m	Momentum (6 month)
BPEBM	Leverage component of BM	Mom6mJunk	Junk Stock Momentum
Cash	Cash to assets	MomOffSeason	Off season long-term reversal
CBOperProf	Cash-based operating profitability	MomOffSeason06YrPlus	Off season reversal years 6 to 10
CF	Cash flow to market	MomSeason	Return seasonality years 2 to 5
ChEQ	Growth in book equity	MomSeason06YrPlus	Return seasonality years 6 to 10
ChInv	Inventory Growth	MomSeason11YrPlus	Return seasonality years 11 to 15
ChInvIA	Change in capital inv (ind adj)	MomSeason16YrPlus	Return seasonality years 16 to 20
ChNNCOA	Change in Net Noncurrent Op Assets	MomSeasonShort	Return seasonality last year
ChNWC	Change in Net Working Capital	NetDebtFinance	Net debt financing
ChTax	Change in Taxes	NetPayoutYield	Net Payout Yield
CompEquIss	Composite equity issuance	NOA	Net Operating Assets
Coskewness	Coskewness	OperProf	operating profits / book equity
CustomerMomentum	Customer momentum	OperProfRD	Operating profitability R&D adjusted
DelFINL	Change in financial liabilities	OPLeverage	Operating leverage
dNoa	change in net operating assets	OrgCap	Organizational capital
EarningsConsistency	Earnings consistency	PriceDelayRsq	Price delay r square
EarningsForecastDisparity	Long-vs-short EPS forecasts	realestate	Real estate holdings
EBM	Enterprise component of BM	RealizedVol	Realized (Total) Volatility
EntMult	Enterprise Multiple	ResidualMomentum	Momentum based on FF3 residuals
EP	Earnings-to-Price Ratio	retConglomerate	Conglomerate return
EquityDuration	Equity Duration	REV6	Earnings forecast revisions
FEPS	Analyst earnings per share	sfe	Earnings Forecast to price
FirmAgeMom	Firm Age - Momentum	ShareIss1Y	Share issuance (1 year)
GP	gross profits / total assets	ShareIss5Y	Share issuance (5 year)
grcapx	Change in capex (two years)	SP	Sales-to-price
grcapx3y	Change in capex (three years)	Tax	Taxable income to income
GrSaleToGrInv	Sales growth over inventory growth	TrendFactor	Trend Factor
IdioVol3F	Idiosyncratic risk (3 factor)	VolSD	Volume Variance
Illiquidity	Amihud's illiquidity	VolumeTrend	Volume Trend
IndRetBig	Industry return of big firms	XFIN	Net external financing

Note: The table lists the 78 characteristics used to form single-sorted portfolios. The data is from Chen and Zimmermann (2022).

Figure C.1. Tensor fibers and slices

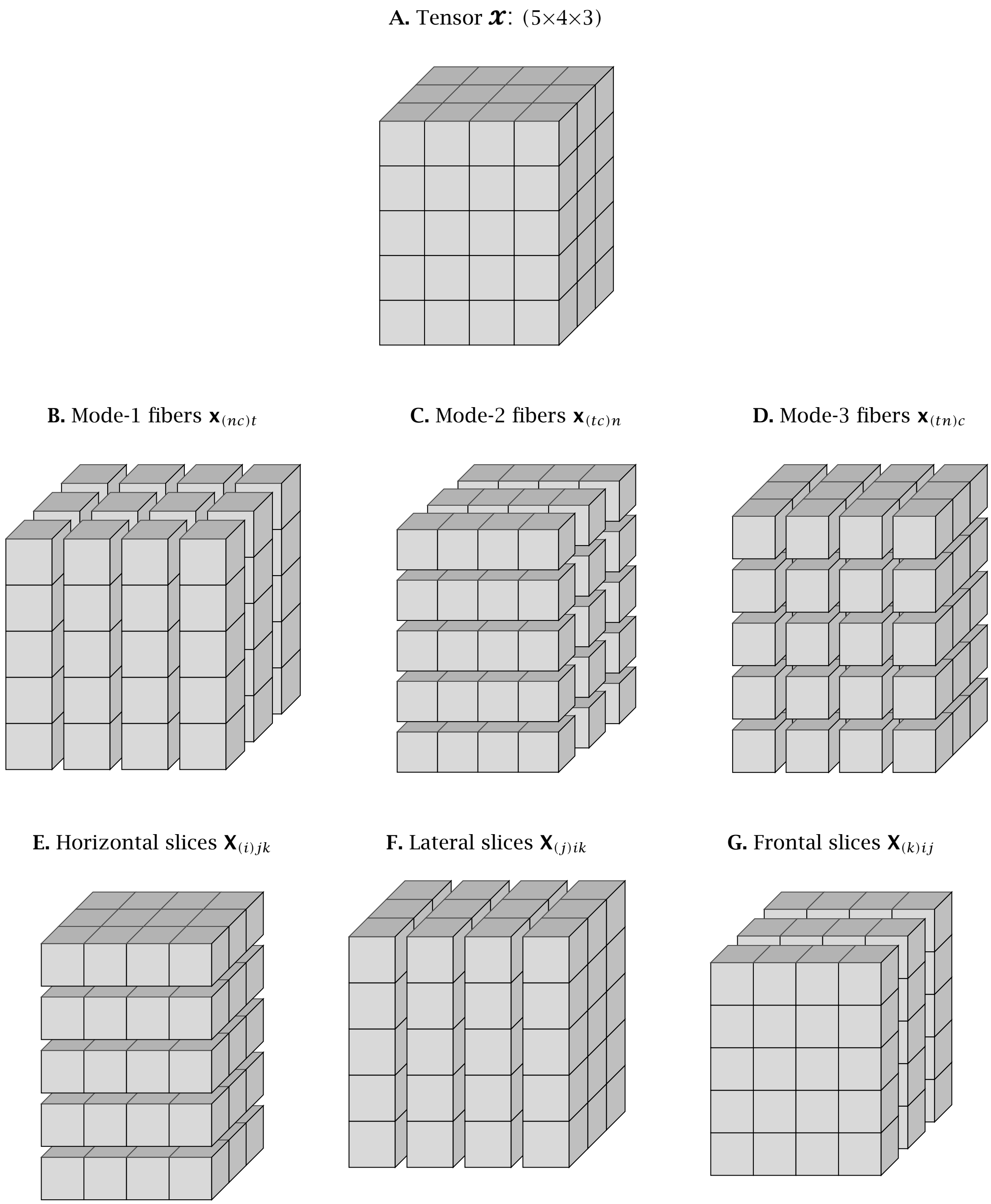
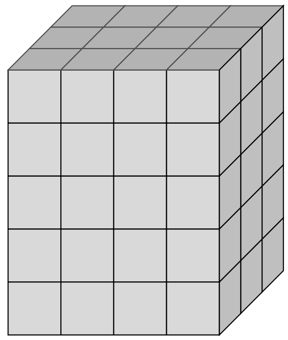
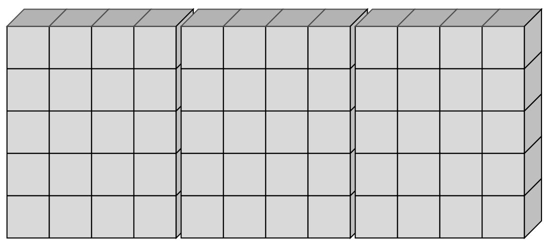


Figure C.2. Tensor as matrices

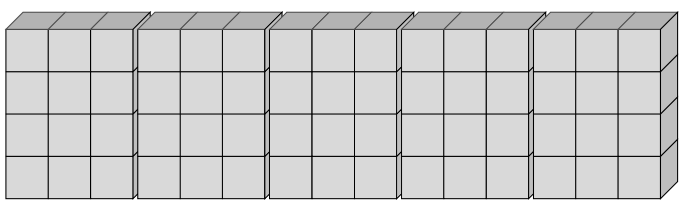
A. Tensor $\boldsymbol{\mathcal{X}}$: (5×4×3)



B. $\mathbf{X}_{(1)}$: (5×12)



C. $\mathbf{X}_{(2)}$: (4×15)



D. $\mathbf{X}_{(3)}$: (3×20)

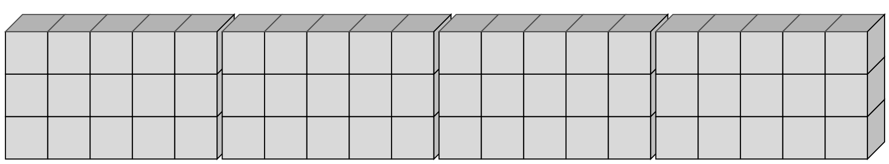


Figure C.3. n-mode Tensor multiplication

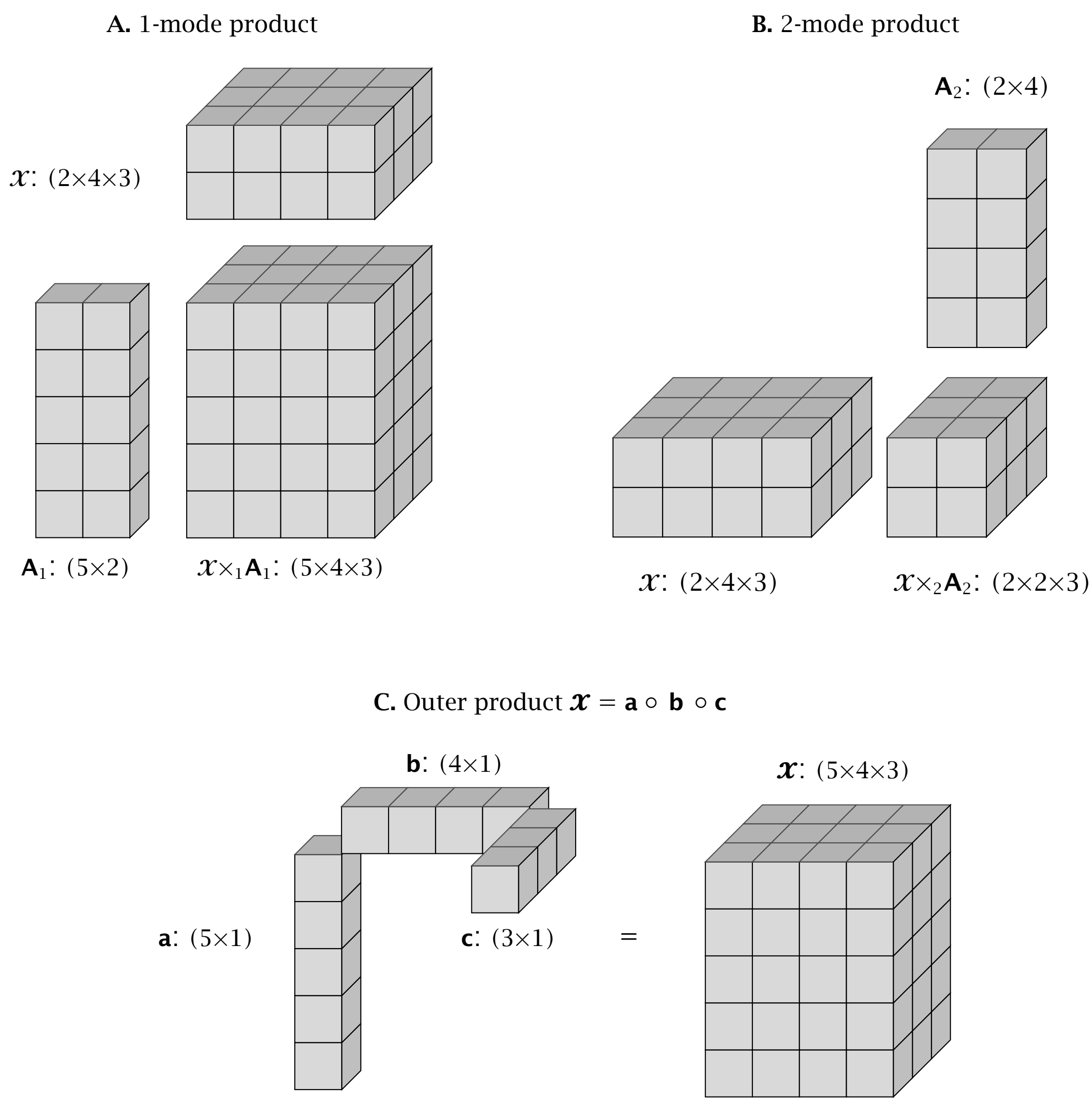


Figure C.4: Tucker Decomposition $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{G}}\times_1\mathbf{V}_1\times_2\mathbf{V}_2\times_3\mathbf{V}_3$

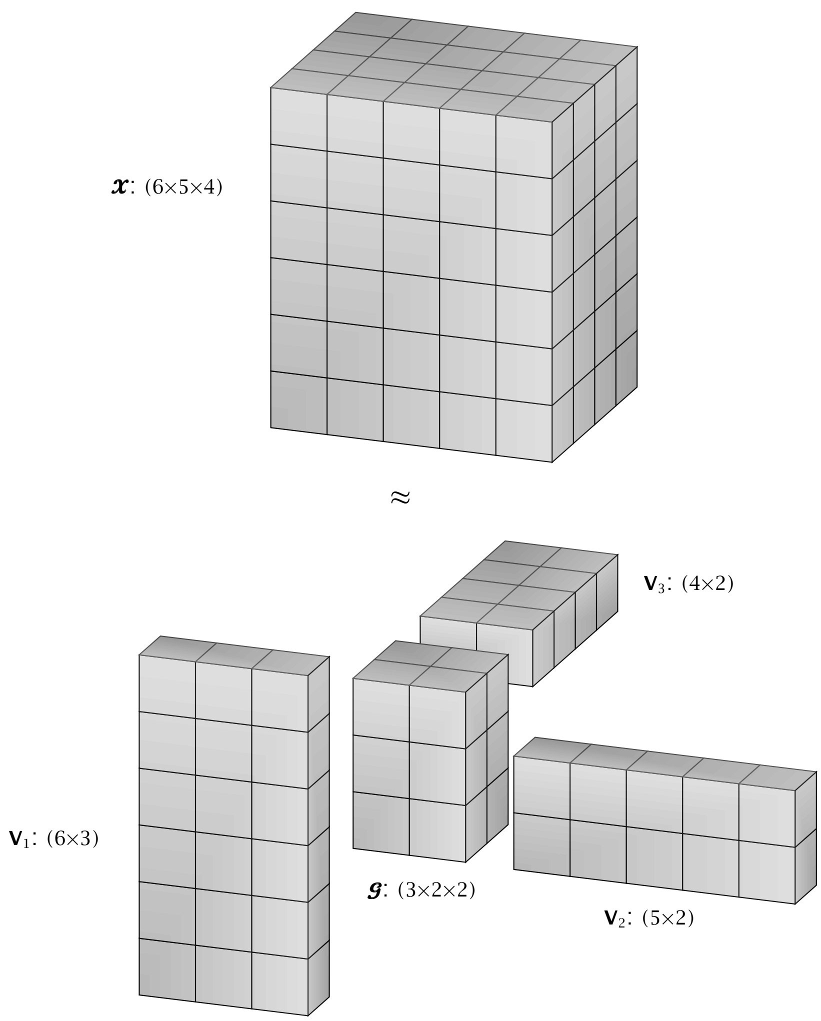


Figure C.5: Proportionality of C-slices of weight tensor $\boldsymbol{\mathcal{W}}_{cpq} = \mathbf{v}_c^{(C)} \circ \mathbf{v}_p^{(P)} \circ \mathbf{v}_q^{(Q)}$

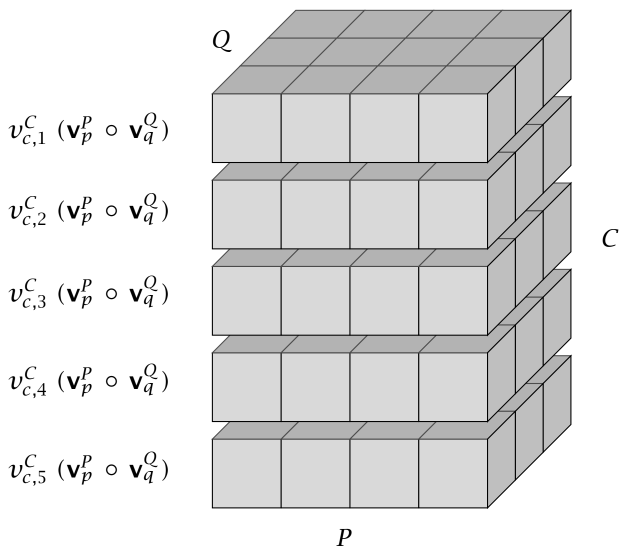
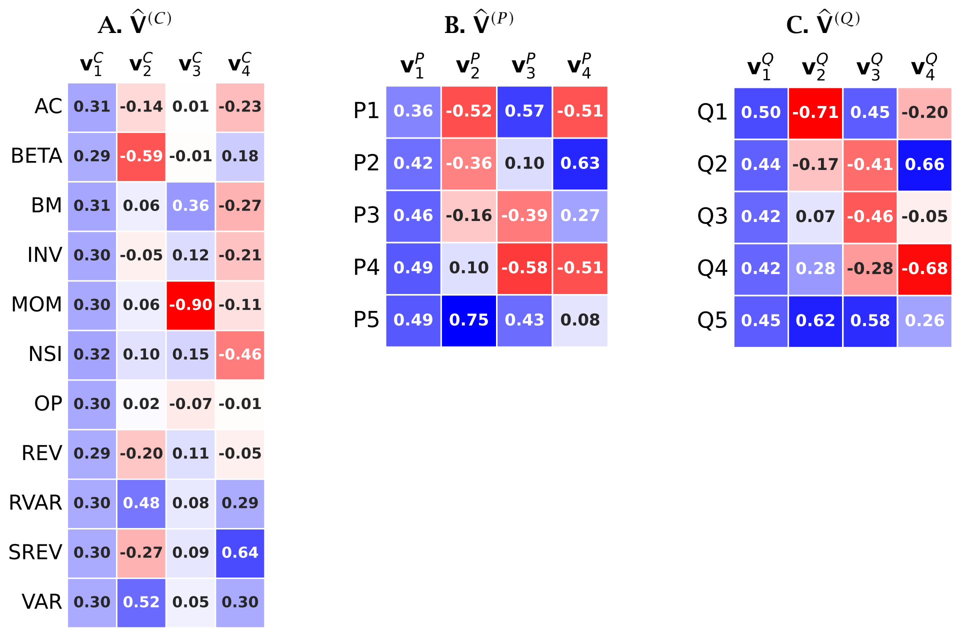
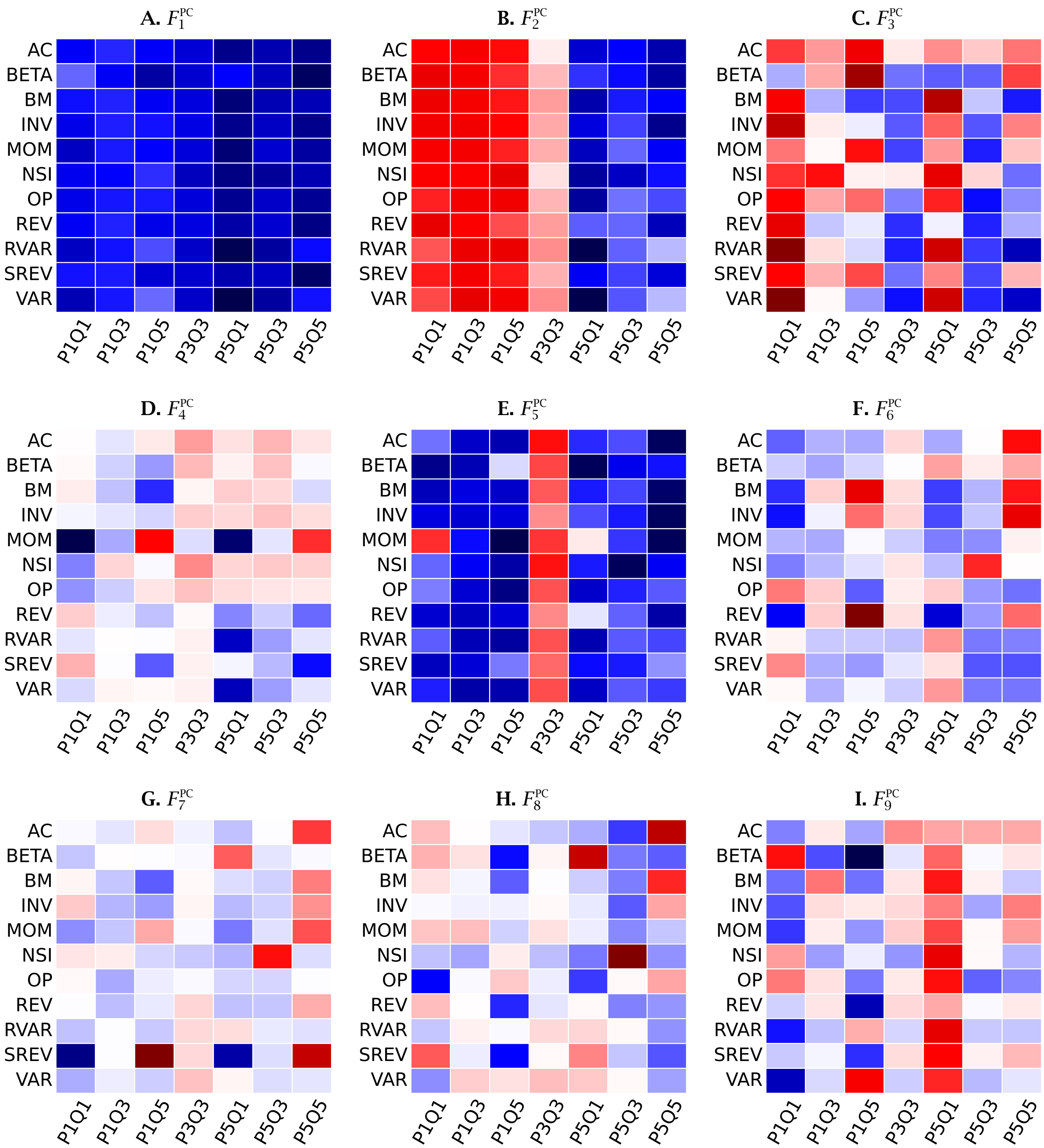


Figure C.6. $\hat{\mathbf{V}}^i$ matrices for 3D-PCA with $(K_C, K_P, K_Q) = (C, P, Q)$



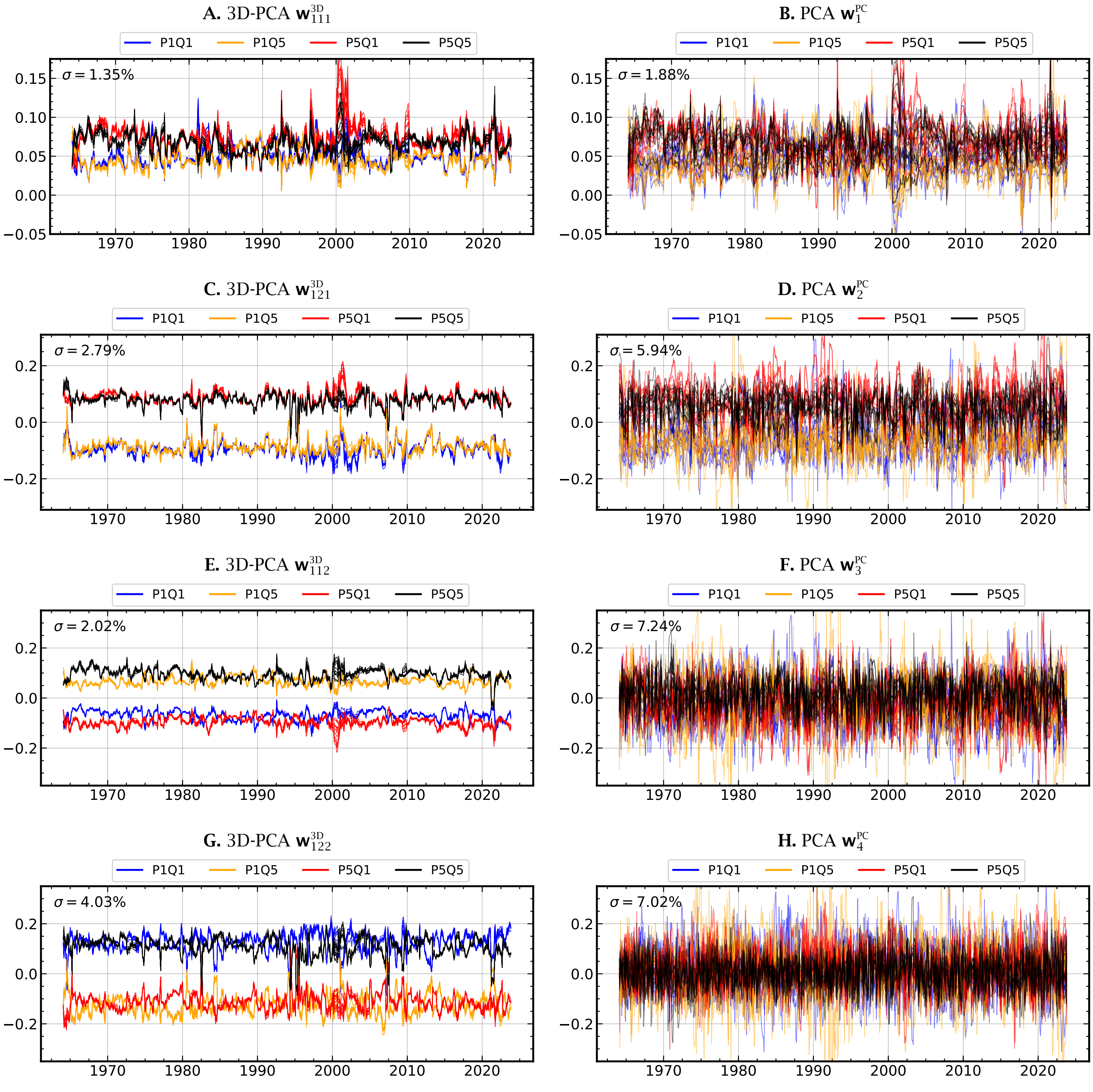
Notes: The heatmaps show estimates of the $\hat{\mathbf{V}}^{(C)}, \hat{\mathbf{V}}^{(P)}$, and $\hat{\mathbf{V}}^{(Q)}$ matrices of the 3D-PCA model (14) with $K_C = C, K_P = P, K_Q = Q$. Negative values are plotted in red and positive ones in blue. The model is estimated by HOOI. The sample is from July 1963 to October 2023.

Figure C.7. PCA – Factor weights



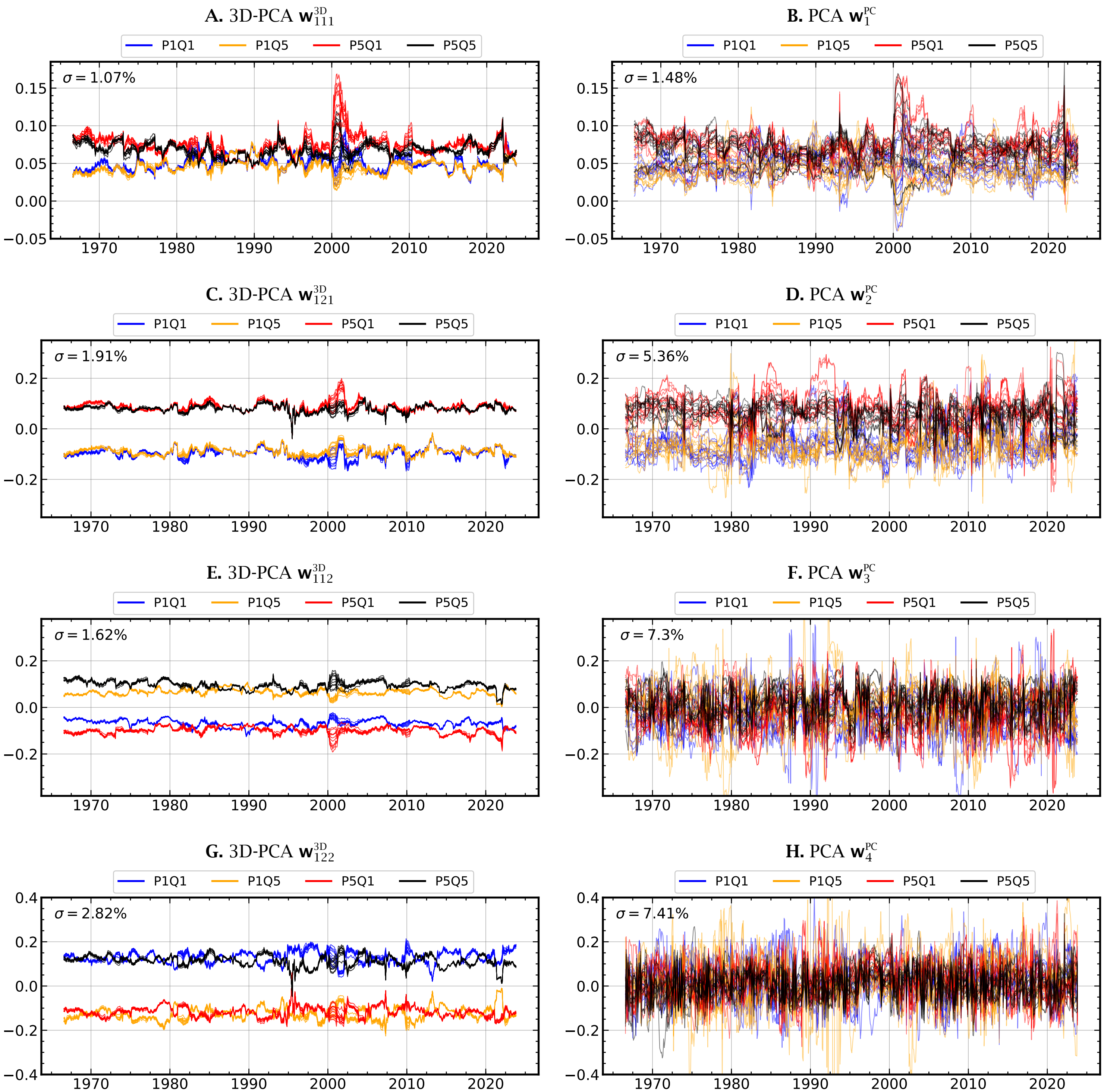
Notes: This figures shows the P1Q1, P1Q3, P1Q5, P3Q3, P5Q1, P5Q3, P5Q5 elements of the weight matrices $\hat{\mathbf{W}}_j^{PC}$ of the first nine PCA factors.

Figure C.8. Weights – Rolling windows $h = 6$



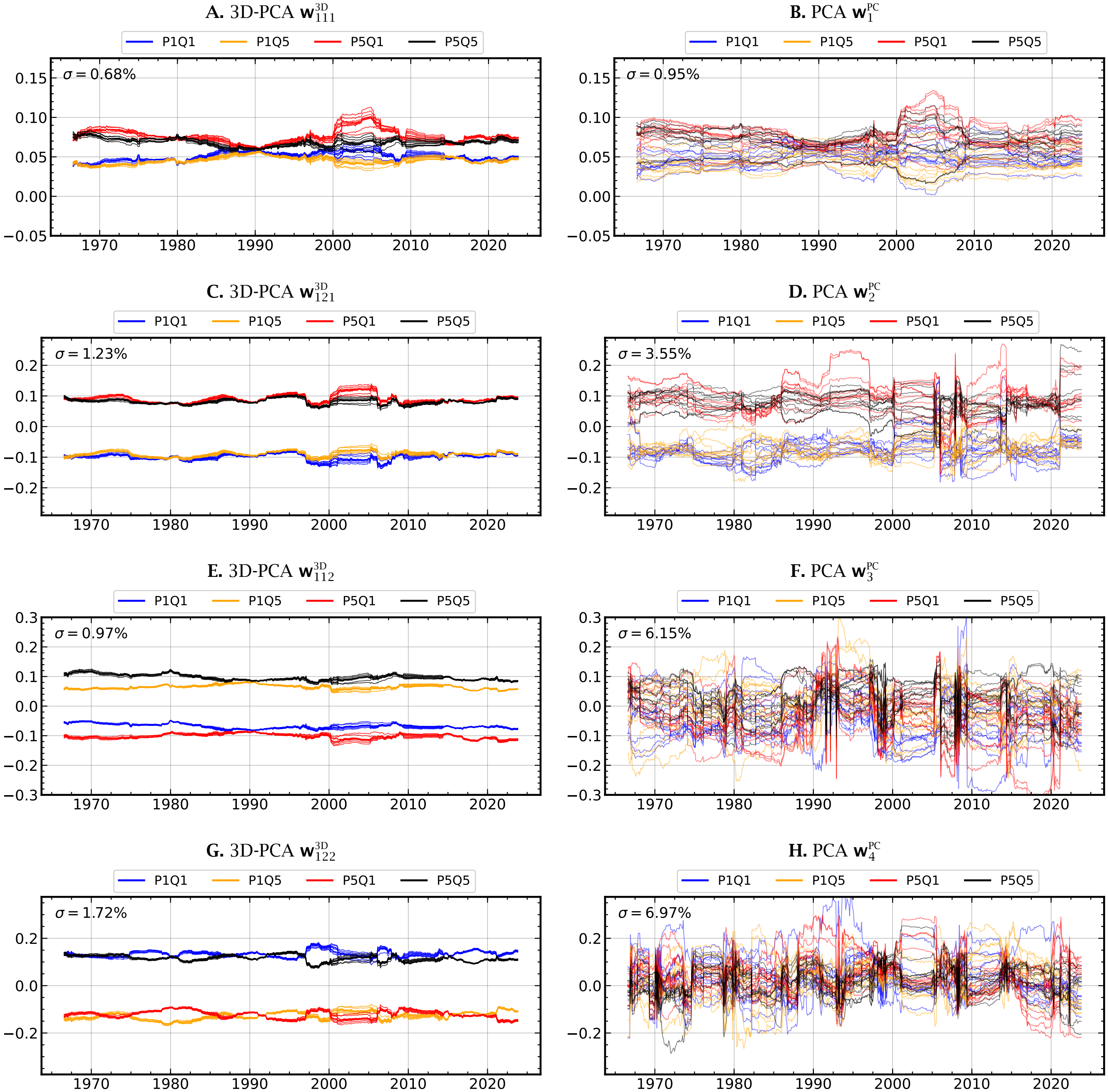
Notes: The figure plots the time series of factor weights of the P1Q1, P1Q5, P5,Q1, and P5Q5 portfolios estimated in rolling samples of length $h = 24$ of 3D-PCA factors $\mathbf{w}^{3D}_{111}, \mathbf{w}^{3D}_{112}, \mathbf{w}^{3D}_{121}$ and $\mathbf{w}^{3D}_{122}$ (left panels) and the first four PCA (right panels) factors. The 3D-PCA factor F_{cpq} is based on the c-th column of $\mathbf{V}^{(C)^\top}$, the p-th column of $\mathbf{V}^{(P)^\top}$, and the q-th column of $\mathbf{V}^{(Q)^\top}$. 3D-PCA factors are based on a partial Tucker decomposition with $K_C = K_P = K_Q = 3$ while PCA factor are based on an estimation with 27 factors. Each plot shows the time series of the factor weights of the 275 portfolios. $\bar{\sigma}$ is the average standard deviation of the 275 time series in each plot. The sample is from July 1967 to October 2023.

Figure C.9. Weights – Rolling windows h = 12



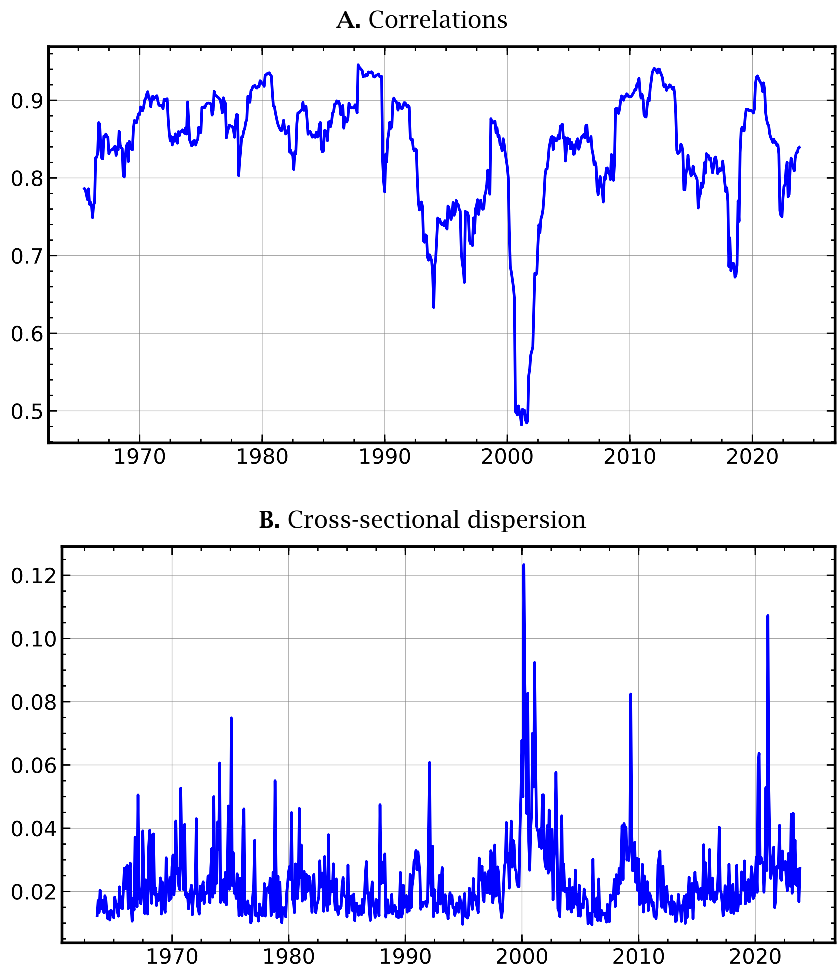
Notes: The figure plots the time series of factor weights of the P1Q1, P1Q5, P5,Q1, and P5Q5 portfolios estimated in rolling samples of length h = 24 of 3D-PCA factors $\mathbf{w}^{3D}_{111}, \mathbf{w}^{3D}_{112}, \mathbf{w}^{3D}_{121}$ and $\mathbf{w}^{3D}_{122}$ (left panels) and the first four PCA (right panels) factors. The 3D-PCA factor F_{cpq} is based on the c-th column of $\mathbf{V}^{(C)^\top}$, the p-th column of $\mathbf{V}^{(P)^\top}$, and the q-th column of $\mathbf{V}^{(Q)^\top}$. 3D-PCA factors are based on a partial Tucker decomposition with $K_C = K_P = K_Q = 3$ while PCA factor are based on an estimation with 27 factors. Each plot shows the time series of the factor weights of the 275 portfolios. $\bar{\sigma}$ is the average standard deviation of the 275 time series in each plot. The sample is from July 1967 to October 2023.

Figure C.10. Weights – Rolling windows $h = 60$



Notes: The figure plots the time series of factor weights of the P1Q1, P1Q5, P5,Q1, and P5Q5 portfolios estimated in rolling samples of length $h = 24$ of 3D-PCA factors $\mathbf{w}^{3D}_{111}, \mathbf{w}^{3D}_{112}, \mathbf{w}^{3D}_{121}$ and $\mathbf{w}^{3D}_{122}$ (left panels) and the first four PCA (right panels) factors. The 3D-PCA factor F_{cpq} is based on the c-th column of $\mathbf{V}^{(C)^\top}$, the p-th column of $\mathbf{V}^{(P)^\top}$, and the q-th column of $\mathbf{V}^{(Q)^\top}$. 3D-PCA factors are based on a partial Tucker decomposition with $K_C = K_P = K_Q = 3$ while PCA factor are based on an estimation with 27 factors. Each plot shows the time series of the factor weights of the 275 portfolios. $\bar{\sigma}$ is the average standard deviation of the 275 time series in each plot. The sample is from July 1967 to October 2023.

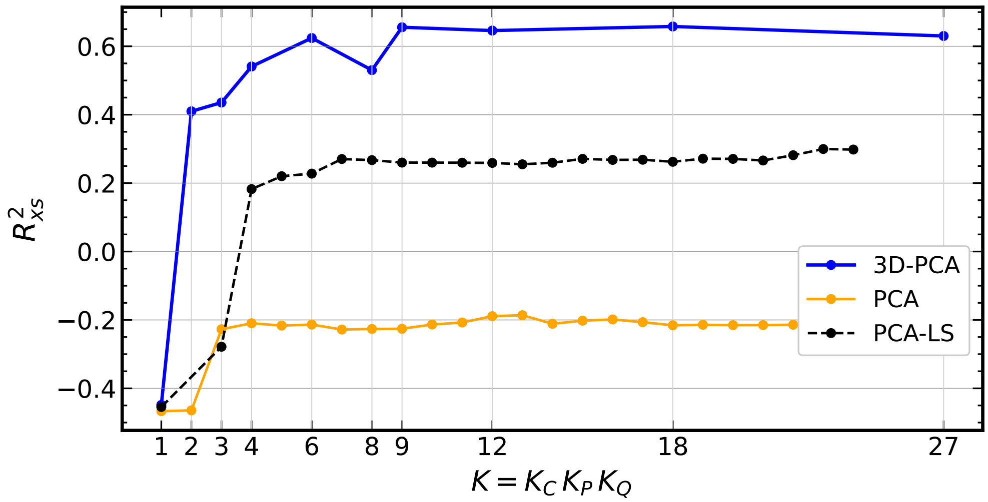
Figure C.11. Correlations and dispersion of portfolio returns



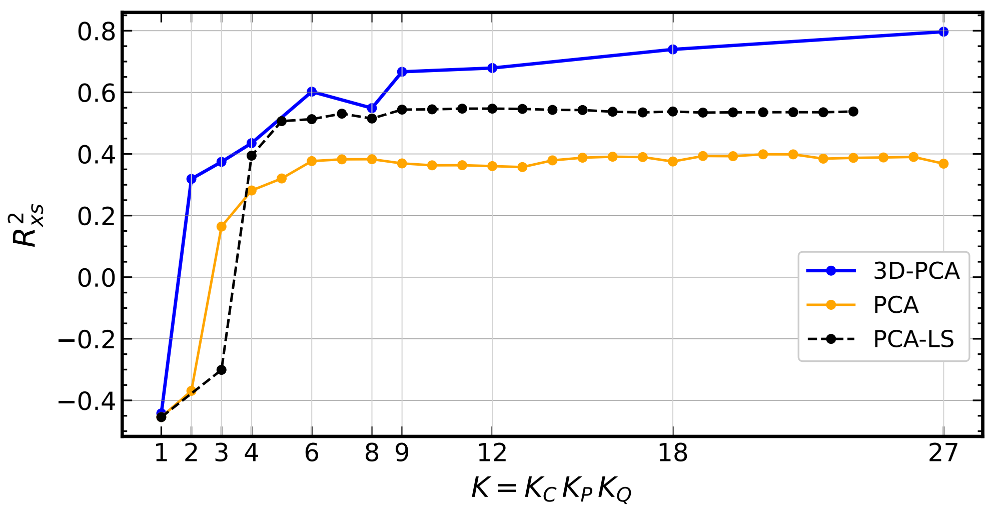
Notes: Panel A shows average correlations of the 275 portfolio returns in rolling windows of length $h = 24$ months and Panel B shows the the cross-sectional standard deviations in each month. The sample is from July 1963 to October 2023.

Figure C.12. 3D-PCA vs. PCA: R^2_{xs} for $h = 12, 60$

A. $h = 12$



B. $h = 60$



Notes: The figure shows the cross-sectional R^2_{xs} of 3D-PCA and PCA models for different number of factors. For the 3D-PCA model, (K_C, K_P, K_Q) for $K_C, K_P, K_Q = 1, 2, 3$. The x-axes show the total number of factors $K = K_C K_P K_Q$. If there are several combinations of (K_C, K_P, K_Q) with the same total number of factors K, the figure shows the specification with the highest R^2_{xs}. The PCA model is based on all 275 portfolios and up to $K = 27$ factors, while PCA-LS is based on 22 long-short SMB_c and HML_c portfolios. Factors are estimated out-of-sample in rolling windows of length $h = 12, 60$ months. The sample is from July 1963 to December 2024.

Figure C.13. 2D-PCA $\hat{\mathbf{V}}^{(P)}$

Notes: The heatmap shows the weight matrix $\hat{\mathbf{V}}^{(P)}$ from a 2D-PCA estimation of single sorted portfolios with $(K_C, K_P) = (4, 3)$ factors. The sample is from July 1967 to December 2024.