Discussion of:

High Dimensional Factor Models with an Application to Mutual Fund Characteristics

by Martin Lettau

Christian Julliard

London School of Economics

BI-SHoF Conference 2022

In a nutshell

A brief ode to tensors:

- Great refresher/starter on the use of tensors for data representation
- Extension of 2-dimensional (typically T x assets), orthogonalised (linear) latent factor models to higher-dimension via Tucker / CP compression algos

  ⇒ 2-D factor models along the (unfolded) modes

Note: Factors are "reduced" form (in the VAR sense) i.e. not orthogonal conditional on the mode and across modes

The data application:

- 3D sample (T x characteristic x fund) of mutual funds return

⇒ "compressed" (un-orthogonalized) representation (by a factor of 97%) "explains" a large share of the data (93% of MSE).

⇒ extracted factors seem to capture salient feature of the characteristics

In a nutshell

A brief ode to tensors:

- Great refresher/starter on the use of tensors for data representation
- Extension of 2-dimensional (typically T x assets), orthogonalised (linear) latent factor models to higher-dimension via Tucker / CP compression algos

  $\Rightarrow$ 2-D factor models along the (unfolded) modes

Note: Factors are "reduced" form (in the VAR sense) i.e. not orthogonal conditional on the mode and across modes

The data application:

- 3D sample (T x characteristic x fund) of mutual funds return

$\Rightarrow$ "compressed" (un-orthogonalized) representation (by a factor of 97%) "explains" a large share of the data (93% of MSE).

$\Rightarrow$ extracted factors seem to capture salient feature of the characteristics

What's a tensor anyway?

"Tensors are the facts of the universe"

Lillian Lieber

Def 1 : "Multi-dimensional array of numbers" (aka a grid of numbers)

- Scalar = tensor of rank 0; Vector = tensor of rank 1 (1 index); Matrix = tensor of rank 2 (2 indexes); 3D array = tensor of rank 3 (3 indexes); ...
- ...misses the geometry of it...

Def 2 : "An object that is invariant under a change of coordinates, and has components that change in special, predictable way under a change of coordinates"

E.g. : (Euclidian) vectors (aka, arrows) are invariant (e.g., length and direction) but the vector components are not invariant

Def 3 : "a collection of (column) vectors and covectors (row vectors) combined together using the tensor product"

$\Rightarrow$ the working definition here for data encoding and compression

Def 4 : "partial derivatives and gradients that transform with the Jacobian matrix"

$\Rightarrow$ nice connection to tangency and Sharpe ratios (more on this later)

What's a tensor anyway?

"Tensors are the facts of the universe"

Lillian Lieber

Def 1 : "Multi-dimensional array of numbers" (aka a grid of numbers)

- Scalar = tensor of rank 0; Vector = tensor of rank 1 (1 index); Matrix = tensor of rank 2 (2 indexes); 3D array = tensor of rank 3 (3 indexes); ...
- ...misses the geometry of it...

Def 2 : "An object that is invariant under a change of coordinates, and has components that change in special, predictable way under a change of coordinates"

E.g. : (Euclidian) vectors (aka, arrows) are invariant (e.g., length and direction) but the vector components are not invariant

Def 3 : "a collection of (column) vectors and covectors (row vectors) combined together using the tensor product"

$\Rightarrow$ the working definition here for data encoding and compression

Def 4 : "partial derivatives and gradients that transform with the Jacobian matrix"

$\Rightarrow$ nice connection to tangency and Sharpe ratios (more on this later)

What's a tensor anyway?

"Tensors are the facts of the universe"

Lillian Lieber

Def 1 : "Multi-dimensional array of numbers" (aka a grid of numbers)

- Scalar = tensor of rank 0; Vector = tensor of rank 1 (1 index); Matrix = tensor of rank 2 (2 indexes); 3D array = tensor of rank 3 (3 indexes); ...
- ...misses the geometry of it...

Def 2 : "An object that is invariant under a change of coordinates, and has components that change in special, predictable way under a change of coordinates"

E.g. : (Euclidian) vectors (aka, arrows) are invariant (e.g., length and direction) but the vector components are not invariant

Def 3 : "a collection of (column) vectors and covectors (row vectors) combined together using the tensor product"

$\Rightarrow$ the working definition here for data encoding and compression

Def 4 : "partial derivatives and gradients that transform with the Jacobian matrix"

$\Rightarrow$ nice connection to tangency and Sharpe ratios (more on this later)

The Tucker (1966) decomposition

Representation:

Let $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_n}$, then

$$\mathcal{Y} \equiv \tilde{\mathcal{Y}} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_n U^{(n)}$$

where $\tilde{\mathcal{Y}} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_n}$ is the core tensor, $U^{(k)} \in \mathbb{R}^{d_k \times d_k}$ are unitary matrices, $\times_k$ denotes the k-mode product (multiplies each mode-k fiber of $\tilde{\mathcal{Y}}$ by $U^{(k)}$).

Approximation / compression:

$\hat{\mathcal{Y}} := \mathcal{G} \times_1 V^{(1)} \times_2 V^{(2)} \ldots \times_n V^{(n)}$ with $\mathcal{G} \in \mathbb{R}^{K_1 \times K_2 \ldots \times K_n}$, $K_j \leq I_j$

s.t. $\hat{\mathcal{Y}} = \arg\min ||\mathcal{Y} - \hat{\mathcal{Y}}||$

$\Rightarrow$ compression from $\mathbb{R}^{I_1 \times I_2 \ldots \times I_n}$ to $\mathbb{R}^{K_1 \times K_2 \ldots \times K_n}$

Note:
- components are neither 1) ordered, 2) orthogonal or 3) unique.
- $\mathcal{G}$ is not diagonal (but CP, with $K_j = \kappa, \ \forall j$) nor linked to e-values/vectors
- for 2D case, SVD-PCA yields same "type" of representation

$\Rightarrow$ But the latter is more interpretable: ordering of orthogonal SR contributions.

Cf. reduced form VAR vs S-VAR via Choleski decomp.

The Tucker (1966) decomposition

Representation:
Let $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_n}$, then

$$\mathcal{Y} \equiv \tilde{\mathcal{Y}} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_n U^{(n)}$$

where $\tilde{\mathcal{Y}} \in \mathbb{R}^{I_1 \times I_2 \ldots \times I_n}$ is the core tensor, $U^{(k)} \in \mathbb{R}^{d_k \times d_k}$ are unitary matrices, $\times_k$ denotes the k-mode product (multiplies each mode-k fiber of $\tilde{\mathcal{Y}}$ by $U^{(k)}$).

Approximation / compression:
$\hat{\mathcal{Y}} := \mathcal{G} \times_1 V^{(1)} \times_2 V^{(2)} \ldots \times_n V^{(n)}$ with $\mathcal{G} \in \mathbb{R}^{K_1 \times K_2 \ldots \times K_n}$, $K_j \leq I_j$
s.t. $\hat{\mathcal{Y}} = \arg\min ||\mathcal{Y} - \hat{\mathcal{Y}}||$

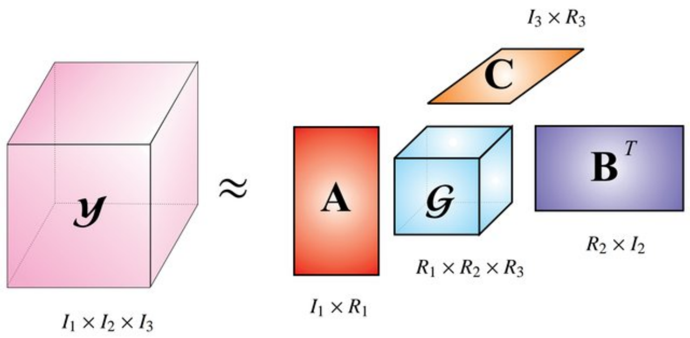
$\Rightarrow$ compression from $\mathbb{R}^{I_1 \times I_2 \ldots \times I_n}$ to $\mathbb{R}^{K_1 \times K_2 \ldots \times K_n}$

Note:
- components are neither 1) ordered, 2) orthogonal or 3) unique.
- $\mathcal{G}$ is not diagonal (but CP, with $K_j = \kappa, \ \forall j$) nor linked to e-values/vectors
- for 2D case, SVD-PCA yields same "type" of representation

$\Rightarrow$ But the latter is more interpretable: ordering of orthogonal SR contributions.

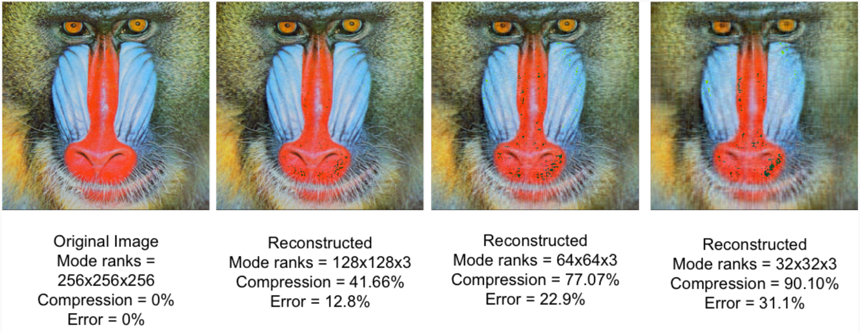
Cf. reduced form VAR vs S-VAR via Choleski decomp.

Example 1: Tucker compression of 3D Array



Tensor can be decomposed as a core tensor G and factor matrices, one for each mode.

Example 2: Tucker compression of 3D mandril

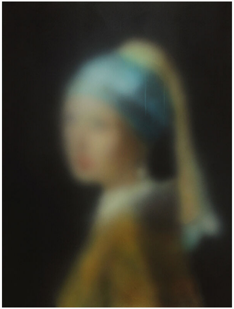


$\Rightarrow$ Efficient compression with typically small errors

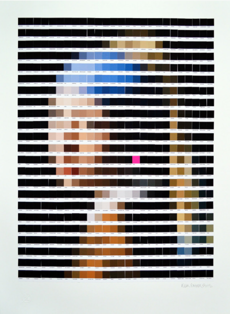
But: only one of the many compression tools available (e.g., compression via low-frequency Fourier coefficients)

Why Tucker?

Given the lack of economic interpretability, and the multiplicity of available methods, why should Tucker/CP be preferred?



Miaz Brothers: *The Muse*, 2020



Nick Smith: *Girl with the Pink Earring*, 2019

J. Vermeer: *Girl with a Pearl Earring*, c. 1665

Does Tucker/CP outperforms e.g. simple PCs? And what are the metrics of success? Just in sample MSE? X-Section? Predictability? OSS?

A Multilinear SVD

We can actually perform a decomposition of tensors that:

1. is ordered, fast, accurate and has the canonical SVD/PC as a particular case
2. is as economically "interpretable" as canonical PCs, and can be "shrunk" accordingly (cf. Kozak, Nagel, Santosh (2020))

Theorem (HOSVD, De Lathauwer, De Moor, Vandewalle (2000))

Every $(I_1 \times I_2 \ldots \times I_n)$-tensor $\mathcal{Y}$ can be written as the product

$$\mathcal{Y} = \mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_n U^{(n)} \quad \textit{where}$$

1. *$U^{(n)}$ is a unitary $I_n \times I_n$ matrix*
2. *the $(I_1 \times I_2 \ldots \times I_n)$-tensor $\mathcal{S}$ of which the sub-tensors $\mathcal{S}_{i_n=\alpha}$, obtained fixing the nth index to α, have the properties of:*
   (i) *all-orthogonality: $\mathcal{S}_{i_n=\alpha} \perp \mathcal{S}_{i_n=\beta}\ \forall n,\ \alpha \neq \beta$: i.e., $\langle \mathcal{S}_{i_n=\alpha}, \mathcal{S}_{i_n=\beta} \rangle = 0$*
   (ii) *ordering: $||\mathcal{S}_{i_n=1}|| \geq ||\mathcal{S}_{i_n=2}|| \geq \ldots \geq ||\mathcal{S}_{i_n=I_n}||\ \forall n$*

The Frobenius-norms $||\mathcal{S}_{i_n=i}|| =: \sigma_i^{(n)}$ are the n-mode singular values of $\mathcal{Y}$ and the vector $U_i^{(n)}$ is an ith n-mode singular vector

A Multilinear SVD

We can actually perform a decomposition of tensors that:

1. is ordered, fast, accurate and has the canonical SVD/PC as a particular case
2. is as economically "interpretable" as canonical PCs, and can be "shrunk" accordingly (cf. Kozak, Nagel, Santosh (2020))

Theorem (HOSVD, De Lathauwer, De Moor, Vandewalle (2000))

Every $(I_1 \times I_2 \ldots \times I_n)$-tensor $\mathcal{Y}$ can be written as the product

$$\mathcal{Y} = \mathcal{S} \times_1 U^{(1)} \times_2 U^{(2)} \ldots \times_n U^{(n)} \quad \textit{where}$$

1. *$U^{(n)}$ is a unitary $I_n \times I_n$ matrix*
2. *the $(I_1 \times I_2 \ldots \times I_n)$-tensor $\mathcal{S}$ of which the sub-tensors $\mathcal{S}_{i_n=\alpha}$, obtained fixing the nth index to α, have the properties of:*
   (i) *all-orthogonality: $\mathcal{S}_{i_n=\alpha} \perp \mathcal{S}_{i_n=\beta}$ $\forall n$, $\alpha \neq \beta$: i.e., $\langle \mathcal{S}_{i_n=\alpha}, \mathcal{S}_{i_n=\beta} \rangle = 0$*
   (ii) *ordering: $||\mathcal{S}_{i_n=1}|| \geq ||\mathcal{S}_{i_n=2}|| \geq \ldots \geq ||\mathcal{S}_{i_n=I_n}||$ $\forall n$*

The Frobenius-norms $||\mathcal{S}_{i_n=i}|| =: \sigma_i^{(n)}$ are the n-mode singular values of $\mathcal{Y}$ and the vector $U_i^{(n)}$ is an ith n-mode singular vector

A Multilinear SVD cont'd

For a tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the Thm implies that:

$$\mathcal{S} = \mathcal{Y} \times_1 U^{(1)^\top} \times_2 U^{(2)^\top} \times_n U^{(3)^\top}$$

is all orthogonal and sorted: i.e., the different "horizontal"/"frontal"/"vertical" matrices of $\mathcal{S}$ (fix first/second/thid index $i_1/i_2/i_3$, while others are free), are mutually orthogonal.

$\Rightarrow$ orthogonal portfolios, ordered by their relevance, for any given dimension of the data, e.g., characteristic specific PCs

Corollary of the HOSVD Thm:

if we construct a tensor $\hat{\mathcal{Y}}$ with n-mode rank of R_n ($1 \leq n \leq N$) by discarding the smallest n-mode singular values $\sigma^{(n)}_{I'_n+1}$, $\sigma^{(n)}_{I'_n+1}$, ..., $\sigma^{(n)}_{R_n}$ for given values of I'_n, i.e. set the corresponding parts of $\mathcal{S}$ equal to zero, then we have

$$||\mathcal{Y} - \hat{\mathcal{Y}}||^2 \leq \sum_{i_1=I'_1+1}^{R_1} (\sigma^{(1)}_{i_1})^2 + \sum_{i_2=I'_1+1}^{R_2} (\sigma^{(2)}_{i_2})^2 + \ldots + \sum_{i_N=I'_N+1}^{R_N} (\sigma^{(1)}_{i_1})^2$$

$\Rightarrow$ knows exactly how much is left unexplained, i.e., the SR^2 of the (orthogonal) "alphas"

A Multilinear SVD cont'd

For a tensor $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, the Thm implies that:

$$\mathcal{S} = \mathcal{Y} \times_1 U^{(1)^\top} \times_2 U^{(2)^\top} \times_n U^{(3)^\top}$$

is all orthogonal and sorted: i.e., the different "horizontal"/"frontal"/"vertical" matrices of $\mathcal{S}$ (fix first/second/thid index $i_1/i_2/i_3$, while others are free), are mutually orthogonal.

⇒ orthogonal portfolios, ordered by their relevance, for any given dimension of the data, e.g., characteristic specific PCs

Corollary of the HOSVD Thm:

if we construct a tensor $\hat{\mathcal{Y}}$ with n-mode rank of R_n $(1 \leq n \leq N)$ by discarding the smallest n-mode singular values $\sigma^{(n)}_{I'_n+1}, \sigma^{(n)}_{I'_n+1}, ..., \sigma^{(n)}_{R_n}$ for given values of I'_n, i.e. set the corresponding parts of $\mathcal{S}$ equal to zero, then we have

$$||\mathcal{Y} - \hat{\mathcal{Y}}||^2 \leq \sum_{i_1=I'_1+1}^{R_1} (\sigma^{(1)}_{i_1})^2 + \sum_{i_2=I'_1+1}^{R_2} (\sigma^{(2)}_{i_2})^2 + \ldots + \sum_{i_N=I'_N+1}^{R_N} (\sigma^{(1)}_{i_1})^2$$

⇒ knows exactly how much is left unexplained, i.e., the SR^2 of the (orthogonal) "alphas"

A Multilinear SVD cont'd

Also: $\mathcal{Y} \equiv \hat{\mathcal{Y}} + \mathcal{E}$, with $\mathcal{E} \perp \hat{\mathcal{Y}}$ by construction of HOSVD

$\Rightarrow$ can make distributional assumptions for $\mathcal{E}$ and perform proper model selection via:

- Bayesian methods (cf., Bryzgalova et al. (2022)): prior on $\mathcal{E} \equiv$ prior on SR^2, and can base selection on the ability of pricing the cross-section
- Shrinkage (cf., Kozak et al. (2020)).

Note: paper lacks a formal selection approach, the method proposed feels "incomplete"

A Multilinear SVD cont'd

Also: $\mathcal{Y} \equiv \hat{\mathcal{Y}} + \mathcal{E}$, with $\mathcal{E} \perp \hat{\mathcal{Y}}$ by construction of HOSVD

$\Rightarrow$ can make distributional assumptions for $\mathcal{E}$ and perform proper model selection via:

- Bayesian methods (cf., Bryzgalova et al. (2022)): prior on $\mathcal{E} \equiv$ prior on SR^2, and can base selection on the ability of pricing the cross-section
- Shrinkage (cf., Kozak et al. (2020)).

Note: paper lacks a formal selection approach, the method proposed feels "incomplete"

A Multilinear SVD cont'd

Also: $\mathcal{Y} \equiv \hat{\mathcal{Y}} + \mathcal{E}$, with $\mathcal{E} \perp \hat{\mathcal{Y}}$ by construction of HOSVD

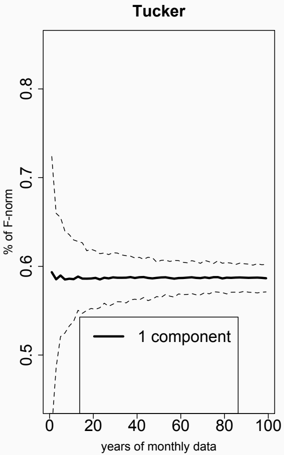
$\Rightarrow$ can make distributional assumptions for $\mathcal{E}$ and perform proper model selection via:

- Bayesian methods (cf., Bryzgalova et al. (2022)): prior on $\mathcal{E} \equiv$ prior on SR^2, and can base selection on the ability of pricing the cross-section
- Shrinkage (cf., Kozak et al. (2020)).

Note: paper lacks a formal selection approach, the method proposed feels "incomplete"

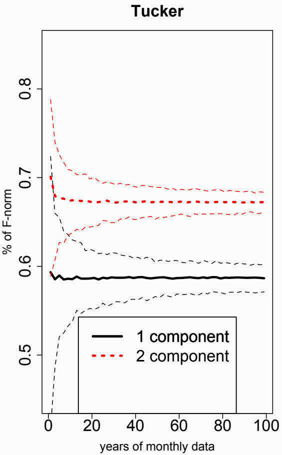
Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



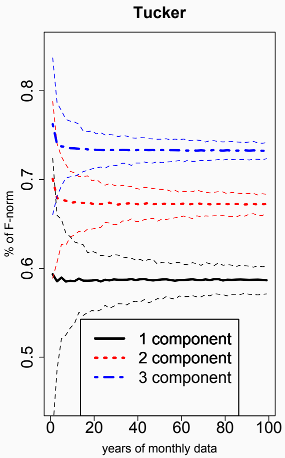
Tucker factors can explain a large share of the RMSE...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



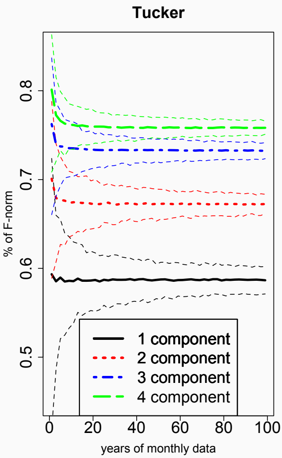
Tucker factors can explain a large share of the RMSE...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



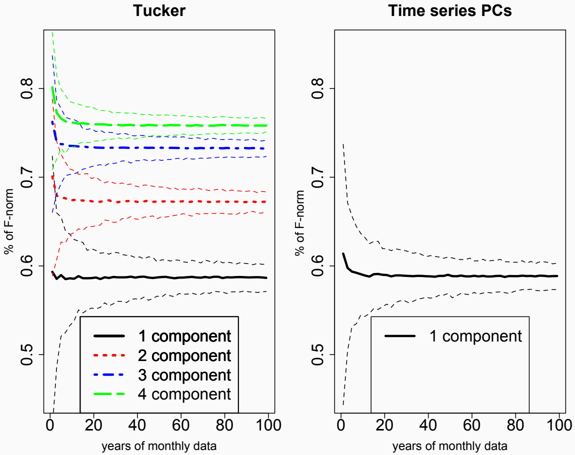
Tucker factors can explain a large share of the RMSE...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



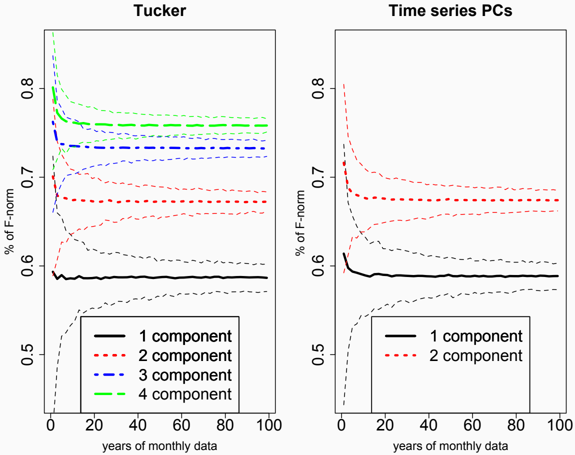
Tucker factors can explain a large share of the RMSE...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



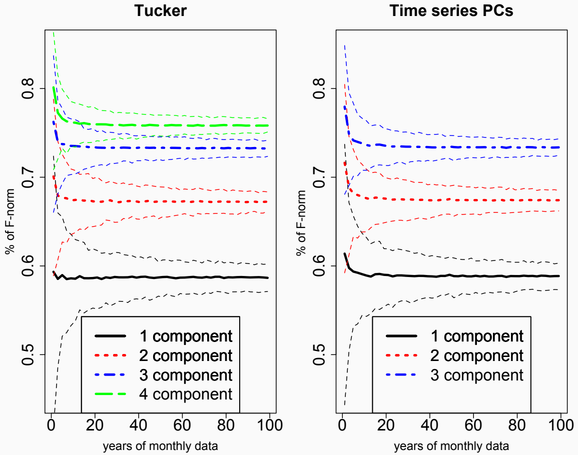
Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



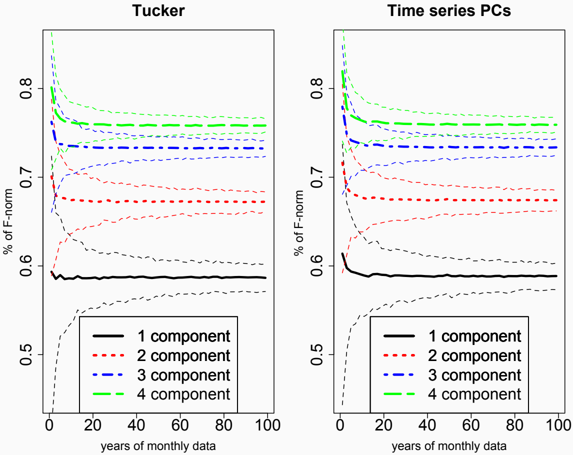
Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



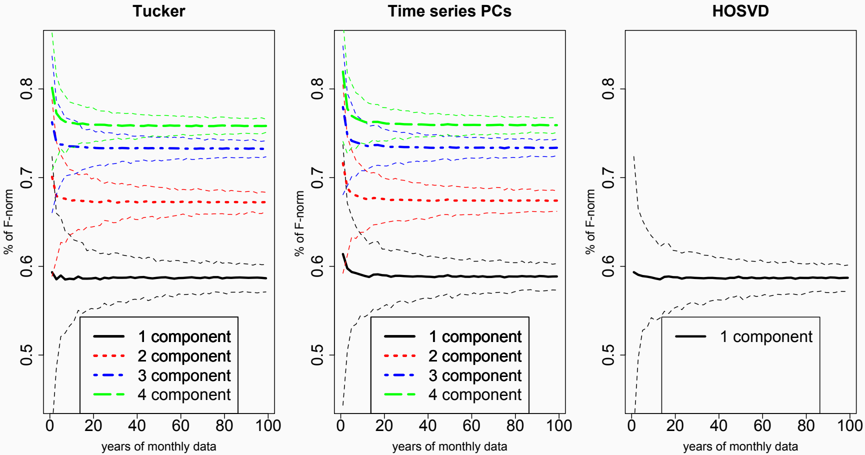
Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



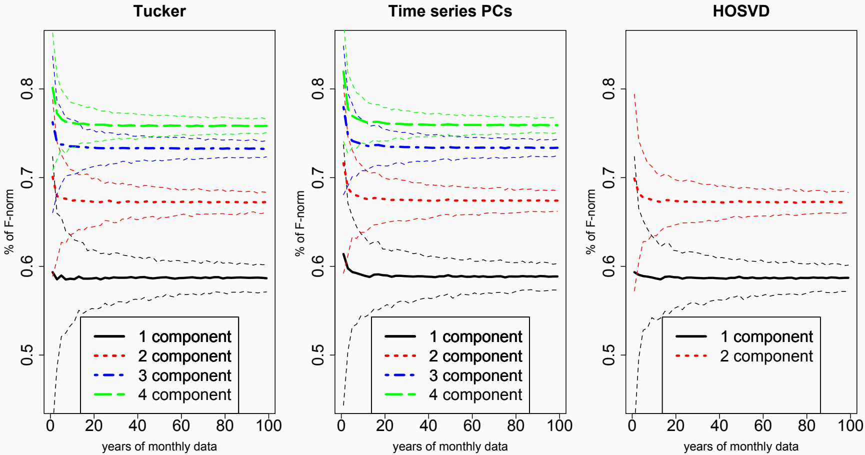
Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



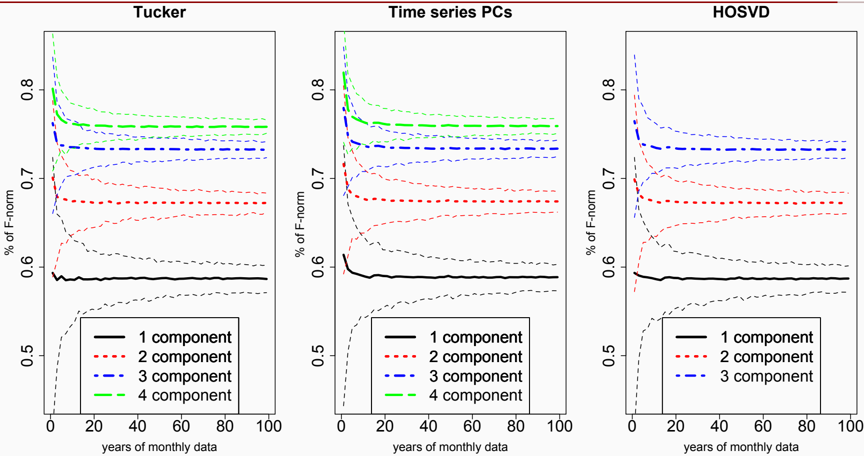
Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs... and higher order SVD has similar performance...

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs... and higher order SVD has similar performance...

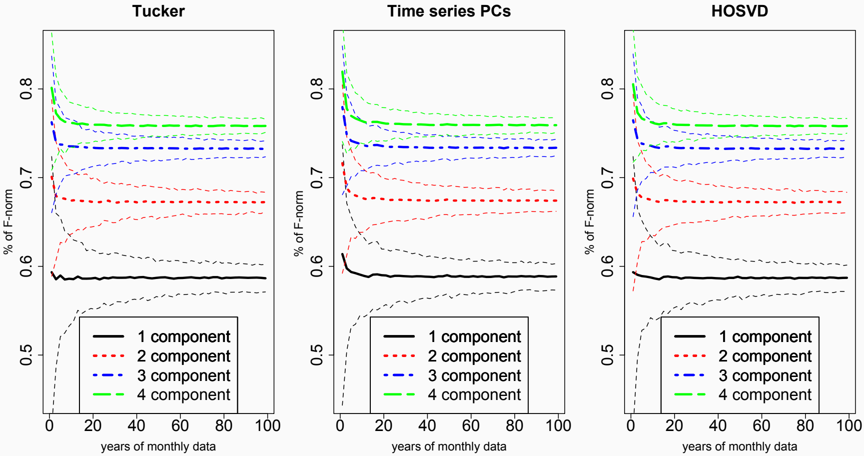
Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs... and higher order SVD has similar performance...

⇒ Needs better evaluation metric than just in sample MSE: e.g., X-sectional pricing, OSS, SRs.

Example 3: a calibrated & simulated T x Size (5) x Value (5) tensor compression



Tucker factors can explain a large share of the RMSE... roughly as much as naive time series PCs... and higher order SVD has similar performance...

Conclusion & Final Suggestions

- A great read for an intro to the potential uses of tensors in AP (cf., Bryzgalova, Kozak, Pelger, Ye (2022))
- Tucker/CP representations are not unique and harder to economically interpret than PCs/HOSVD $\rightarrow$ I'd use HOSVD (+Bayesian selection)
- Selection of dimensionality reduction should be formal
- In sample RMSE is an underwhelming metric of "success" (cf. Lettau & Pelger (2020), Bryzgalova et al. (2022))
- Needs proper comparison/horse race for methods and OSS evaluation