Computing and decomposing tensors

— Decomposition basics

Nick Vannieuwenhoven
(FWO / KU Leuven)
1 Introduction

2 Basic tensor operations

3 Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms

4 Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability

5 References
Overview

1. Introduction
2. Basic tensor operations
3. Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms
4. Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability
5. References
Multidimensional data appear in many applications:
- image and signal processing;
- pattern recognition, data mining and machine learning;
- chemometrics;
- biomedicine;
- psychometrics; etc.

There are two major problems associated with this data:
1. Storage cost is very high, and
2. analysis and interpretation of patterns in data.

**Tensor decompositions** can *identify* and *exploit* useful structures in the tensor that may not be apparent from its given coordinate representation.
Different decompositions have different strengths.

A Tucker decomposition

\[ \text{Tensor} = \text{Factor}_1 \times \text{Factor}_2 \times \cdots \]

can reduce storage costs.

A tensor rank decomposition

\[ \text{Tensor} = \text{Factor}_1 + \text{Factor}_2 + \cdots \]

may uncover interpretable patterns.
Overview

1 Introduction

2 Basic tensor operations

3 Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms

4 Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability

5 References
Flattenings

A tensor $\mathcal{A}$ of order $d$ lives in the tensor product of $d$ vector spaces:

$$\mathcal{A} \in \mathbb{F}^{n_1} \otimes \mathbb{F}^{n_2} \otimes \cdots \otimes \mathbb{F}^{n_d} \cong \mathbb{F}^{n_1 \times n_2 \times \cdots \times n_d}$$

A 3\textsuperscript{rd} order tensor has 3 associated vector spaces:

- Mode-1 vectors ($\mathbb{F}^{n_1}$)
- Mode-2 vectors ($\mathbb{F}^{n_2}$)
- Mode-3 vectors ($\mathbb{F}^{n_3}$)
Computing and decomposing tensors: Decomposition basics

Basic tensor operations

\[ A = A^{(2)} \in \mathbb{F}^{n_1 \times n_2 \times n_3} \]

\[ \text{Mode-2 flattening} \]

\[ A^{(2)} = \quad \in \mathbb{F}^{n_2 \times n_1 n_3} \]
Formally, a flattening is the linear map induced via the universal property of the multilinear map

\[ (\pi; \tau) : V_1 \times \cdots \times V_d \rightarrow (V_{\pi_1} \otimes \cdots \otimes V_{\pi_k}) \otimes (V_{\tau_1} \otimes \cdots \otimes V_{\tau_{d-k}}) \]

\[(a_1, \ldots, a_d) \mapsto (a_{\pi_1} \otimes \cdots \otimes a_{\pi_k})(a_{\tau_1} \otimes \cdots \otimes a_{\tau_{d-k}})^T\]

It is common to use the following shorthand notations in the literature:

\[ \mathcal{A}(k) := \mathcal{A}(k;1,\ldots,k-1,k+1,\ldots,d) \quad \text{and} \quad \text{vec}(\mathcal{A}) := \mathcal{A}(1,\ldots,d;\emptyset). \]

Be aware that some authors still define \( \mathcal{A}(k) = \mathcal{A}(k;k+1,\ldots,d,1,\ldots,k-1) \).
For example, if \( A = \sum_{i=1}^{r} a_i \otimes b_i \otimes c_i \) then

\[
A_{(2)} = \sum_{i=1}^{r} b_i (a_i \otimes c_i)^T.
\]

Flattenings can be implemented on a computer for tensors expressed in coordinates simply by rearranging the elements in the \( d \)-array of size \( n_1 \times \cdots \times n_d \) to form a 2-array of size \( n_{\pi_1} \cdots n_{\pi_k} \times n_{\tau_1} \cdots n_{\tau_{d-k}} \).

In fact, all flattenings \( A_{(1,\ldots,k; k+1,\ldots,d)} \) in which the order of the factors is not changed can be implemented on a computer with 0 computational cost (time and memory).
Multilinear multiplication

As mentioned in the first lecture, **multilinear multiplication** is synonymous with the **tensor product of linear maps** $A_i : V_i \rightarrow W_i$, where $V_i, W_i$ are finite-dimensional vector spaces.

This is the unique linear map from $V_1 \otimes \cdots \otimes V_d$ to $W_1 \otimes \cdots \otimes W_d$ induced by the universal property by the multilinear map

$$V_1 \times \cdots \times V_d \rightarrow W_1 \otimes \cdots \otimes W_d,$$

$$(v_1, \ldots, v_d) \mapsto (A_1 v_1) \otimes \cdots \otimes (A_d v_d).$$

The induced linear map is $A_1 \otimes \cdots \otimes A_d$. 
The notation

\[(A_1, \ldots, A_d) \cdot \mathcal{A} := (A_1 \otimes \cdots \otimes A_d)(\mathcal{A})\]

is commonly used in the literature, specifically when working in coordinates.

The shorthand notation

\[A_k \cdot_k \mathcal{A} := (\text{Id}, \ldots, \text{Id}, A_k, \text{Id}, \ldots, \text{Id}) \cdot \mathcal{A}\]

is also used in the literature.
By definition, the action on rank-1 tensor is

\[(A_1 \otimes \cdots \otimes A_d)(v_1 \otimes \cdots \otimes v_d) = (A_1 v_1) \otimes \cdots \otimes (A_d v_d).\]

The \textit{composition} of multilinear multiplications behaves like

\[(A_1 \otimes \cdots \otimes A_d)(B_1 \otimes \cdots \otimes B_d)(A) = ((A_1 B_1) \otimes \cdots \otimes (A_d B_d))(A),\]

which follows immediately from the definition.

Practically, multilinear multiplications are often \textbf{computed} by exploiting

\[\left[ (A_1, \ldots, A_d) \cdot A \right]_{(k)} = A_k A_{(k)}(A_1 \otimes \cdots \otimes A_{k-1} \otimes A_{k+1} \otimes \cdots \otimes A_d)^T\]
Overview

1. Introduction

2. Basic tensor operations

3. Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms

4. Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability

5. References
Overview

1. Introduction
2. Basic tensor operations
3. Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms
4. Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability
5. References
Multilinear rank

Assume that $\mathcal{A}$ lives in a separable tensor subspace

$$\mathcal{A} \in W_1 \otimes W_2 \otimes \cdots \otimes W_d \subset \mathbb{F}^{n_1} \otimes \mathbb{F}^{n_2} \otimes \cdots \otimes \mathbb{F}^{n_d}.$$ 

Since the mode-$k$ flattening

$$\mathcal{A}(k) \in W_k \otimes (W_1 \otimes \cdots \otimes W_{k-1} \otimes W_{k+1} \otimes \cdots \otimes W_d)^*,$$

which is a subspace of the $n_k \times (n_1 \cdots n_{k-1} n_{k+1} \cdots n_d)$ matrices, it follows that the column span

$$\text{span}(\mathcal{A}(k)) \subset W_k.$$
In fact, the *smallest* separable tensor subspace that $\mathcal{A}$ lives in is $W_1 \otimes \cdots \otimes W_d$ with

$$W_k := \text{span}(\mathcal{A}(k)).$$

The dimension of this subspace is

$$r_k := \dim W_k = \dim \text{span}(\mathcal{A}(k)) = \text{rank}(\mathcal{A}(k)).$$

**Definition (Hitchcock, 1928)**

The **multilinear rank** of $\mathcal{A}$ is the tuple containing the dimensions of the minimal subspaces that the standard flattenings of $\mathcal{A}$ live in:

$$\text{mlrank}(\mathcal{A}) := (r_1, r_2, \ldots, r_d).$$
In the case \( A \in W_1 \otimes W_2 \subset \mathbb{F}^{n_1 \times n_2} \) is a matrix, the multilinear rank is, by definition,

\[
\text{mlrank}(A) = (\dim W_1, \dim W_2) = (\rank(A_{(1)}), \rank(A_{(2)})) = (\rank(A), \rank(A^T)).
\]

In the matrix case, we attach special names to \( W_1 \) and \( W_2 \):
- \( W_1 \) is the \textbf{column space} or \textbf{range}, and
- \( W_2 \) is the \textbf{row space}.

The \textbf{fundamental theorem of linear algebra} states that \( \dim W_1 = \dim W_2 \). Therefore,

\[
\text{mlrank}(A) = (\dim W_1, \dim W_2) = (r, r).
\]

Consequently, not all tuples are feasible multilinear ranks!
Proposition (Carlini and Kleppe, 2011)

Let $\mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d}$ with multilinear rank $(r_1, \ldots, r_d)$. Then, for all $k = 1, \ldots, d$ we have

$$r_k \leq \prod_{j \neq k} r_j.$$ 

The proof is left as an exercise.
Connection to algebraic geometry

The set of tensors of bounded multilinear rank

\[ M_{r_1,\ldots,r_d} := \{ \mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d} \mid \text{mlrank}(\mathcal{A}) \leq (r_1, \ldots, r_d) \} \]

is easily seen to be an algebraic variety, i.e., the solution set of a system of polynomial equations, because it is the intersection of the determinantal varieties

\[ M_{r_k} := \{ \mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d} \mid \text{rank}(\mathcal{A}_{(k)}) \leq r_k \} \]

for \( k = 1, \ldots, d \).
Higher-order singular value decomposition

If \( \mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d} \) lives in a separable tensor subspace \( V_1 \otimes \cdots \otimes V_d \) with \( r_k := \dim V_k \), then there exist bases

\[
A_k = [a_{ij}^k]_{j=1}^{r_k} \in \mathbb{F}^{n_k \times r_k} \text{ for } V_k \subset \mathbb{F}^{n_k}
\]

such that

\[
\mathcal{A} = \sum_{i_1=1}^{r_1} \cdots \sum_{i_d=1}^{r_d} c_{i_1,\ldots,i_d} a_{i_1}^1 \otimes \cdots \otimes a_{i_d}^d =: (A_1, A_2, \ldots, A_d) \cdot C
\]

for some \( C \in \mathbb{F}^{r_1 \times r_2 \times \cdots \times r_d} \).

This is equivalent to stating that

\[
\text{mlrank}(\mathcal{A}) = (r_1, r_2, \ldots, r_d).
\]
Recall that the **Moore–Penrose pseudoinverse** of matrix $A \in \mathbb{F}^{m \times n}$ of rank $n$ is given by

$$A^\dagger = (A^H A)^{-1} A^H.$$

Then, the coefficients $C$ of $A$ with respect to the basis $A_1 \otimes \cdots \otimes A_d$ satisfy

$$\mathcal{A} = (A_1, A_2, \ldots, A_d) \cdot C,$$

so that

$$(A_1^\dagger, A_2^\dagger, \ldots, A_d^\dagger) \cdot \mathcal{A} = (A_1^\dagger, A_2^\dagger, \ldots, A_d^\dagger) \cdot (A_1, A_2, \ldots, A_d) \cdot C$$

$$= (A_1^\dagger A_1, A_2^\dagger A_2, \ldots, A_d^\dagger A_d) \cdot C$$

$$= C.$$
In other words, if we know that \( \mathcal{A} \) lives in \( V_1 \otimes \cdots \otimes V_d \), and we have chosen some bases \( A_k \) of \( V_k \), then the coefficients (also called \textbf{core tensor}) are given by \( C = (A_1^\dagger, A_2^\dagger, \ldots, A_d^\dagger) \cdot \mathcal{A} \).

The factorization

\[
\mathcal{A} = (A_1, \ldots, A_d) \cdot C
\]

reveals the separable subspace \( V = V_1 \otimes \cdots \otimes V_d \) that tensor \( \mathcal{A} \) lives in, as \( A_k \) provides a basis of \( V_k \) from which a tensor product basis of \( V \) can be constructed. The factorization is called a (rank-revealing) \textbf{Tucker decomposition} of \( \mathcal{A} \) in honor of L. Tucker (1963).
The **higher-order singular value decomposition** (HOSVD), popularized by De Lathauwer, De Moor, and Vandewalle (2000) but already introduced by Tucker (1966), is a particular strategy for choosing orthonormal bases $A_k$.

The HOSVD chooses as orthonormal basis for $V_k$ the left singular vectors of $A_{(k)}$. That is, let the thin SVD of $A_{(k)}$ be

$$A_{(k)} = U_k \Sigma_k Q_k^H.$$

Then, the HOSVD orthogonal basis for $V_k$ is given by $U_k$. 
An advantage of choosing orthonormal bases $A_k$, beyond improved numerical stability, is that the Moore–Penrose inverse reduces to

$$U_k^\dagger = (U_k^H U_k)^{-1} U_k^H = U_k^H,$$

so that

$$\mathcal{A} = (U_1, U_2, \ldots, U_d) \cdot ((U_1, U_2, \ldots, U_d)^H \cdot \mathcal{A})$$

$$= (U_1 U_1^H, U_2 U_2^H, \ldots, U_d U_d^H) \cdot \mathcal{A}$$

$$= \pi_1 \pi_2 \cdots \pi_d \mathcal{A}$$

where

$$\pi_k \mathcal{A} := (U_k U_k^H) \cdot_k \mathcal{A}$$

is the HOSVD mode-$k$ orthogonal projection.
The coefficients $d$-array

$$S = (U_1, U_2, \ldots, U_d)^H \cdot \mathcal{A}$$

is called the core tensor.

The orthogonal basis of $V_1 \otimes \cdots \otimes V_d$,

$$U_1 \otimes U_2 \otimes \cdots \otimes U_d := [u_{i_1}^1 \otimes \cdots \otimes u_{i_d}^d]_{i_1, \ldots, i_d = 1}^{r_1, \ldots, r_d}$$

is called the HOSVD basis.

By definition of the thin SVD, we have

$$r_k = \dim V_k = \text{rank}(U_k)$$

and so $U_k \in \mathbb{R}^{n_k \times r_k}$. 
Algorithm 1: HOSVD Algorithm

input : A tensor $\mathcal{A} \in \mathbb{F}^{n_1 \times n_2 \times \cdots \times n_d}$

output: The components $(U_1, U_2, \ldots, U_d)$ of the HOSVD basis

output: Coefficients array $S \in \mathbb{F}^{r_1 \times r_2 \times \cdots \times r_d}$

for $k = 1, 2, \ldots, d$ do
  Compute the compact SVD $\mathcal{A}_{(k)} = U_k \Sigma_k Q_k^H$;
end

$S \leftarrow (U_1^H, U_2^H, \ldots, U_d^H) \cdot \mathcal{A}$;
The HOSVD provides a natural **data sparse representation** of tensors $\mathcal{A}$ living in a separable subspace.

If $\mathcal{A} \in \mathbb{F}^{n_1 \times n_2 \times \cdots \times n_d}$ has multilinear rank $(r_1, r_2, \ldots, r_d)$, then it can be represented exactly via the HOSVD as

$$\mathcal{A} = (U_1, U_2, \ldots, U_d) \cdot S$$

using only

$$\prod_{k=1}^{d} r_k + \sum_{k=1}^{d} n_k r_k$$

storage (for $S$ and the $U_i$).
Overview

1 Introduction

2 Basic tensor operations

3 Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms

4 Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability

5 References
Consider the mathematically simple task of computing the multilinear rank of a tensor $\mathcal{A}$. For example, $r_k$ equals the number of nonzero singular values of $\mathcal{A}(k)$.

Let us take the rank-1 tensor

$$
\mathcal{A} = \begin{bmatrix}
1 & \sqrt{2} & \sqrt{2} & 2 \\
\sqrt{2} & 2 & 2\sqrt{2} & 2
\end{bmatrix} = \mathbf{v} \otimes \mathbf{v} \otimes \mathbf{v}, \quad \text{where } \mathbf{v} = \begin{bmatrix} 1 \\ \sqrt{2} \end{bmatrix}.
$$

Its 1-flattening is

$$
\mathcal{A}(1) = \mathbf{v}(\mathbf{v} \otimes \mathbf{v})^T = \begin{bmatrix}
1 & \sqrt{2} & \sqrt{2} & 2 \\
\sqrt{2} & 2 & 2\sqrt{2} & 2
\end{bmatrix}.
$$
Computing the singular values of $\mathcal{A}(1)$ in Matlab R2017b, we get the next result:

```matlab
>> svd([[1 sqrt(2) sqrt(2) 2];[sqrt(2) 2 2 2*sqrt(2)]])
ans =
  5.196152422706632e+00
  1.805984985273179e-16
```

Both singular values are nonzero, so the computed rank is 2!

However, the rank of $\mathcal{A}(1)$ is 1, so what have we computed? Can we make sense of this result?
There are two sources of error that entered our computation:

1. **representation errors**, and
2. **computation errors**.

The **representation error** is incurred because $\mathcal{A}_1$ cannot be represented with (IEEE double-precision) floating-point numbers; indeed, $\sqrt{2} \notin \mathbb{Q}$.

Nevertheless, the numerical representation of $\mathcal{A}_1$ is very close to the latter. By the properties of floating-point arithmetic, we have

$$\|\mathcal{A}_1 - \text{fl}(\mathcal{A}_1)\|_F^2 \leq 3(\sqrt{2}\delta)^2 + ((2\sqrt{2})\delta)^2 = 14\delta^2,$$

where $\delta \approx 1.1 \cdot 10^{-16}$ is the unit roundoff.
The **computation error** arises in the computation of the singular values of the matrix with floating-point elements. The magnitude of this error strongly depends on the algorithm. Numerically “stable” algorithms will only introduce “small” errors.

Matlab’s `svd` likely implements an algorithm satisfying\(^1\)

\[
|\tilde{\sigma}_k(\tilde{A}) - \sigma_k(\tilde{A} + E)| \leq p(m, n) \cdot \sigma_1(\tilde{A} + E) \cdot \delta
\]

with

\[
\|E\|_2 \leq p(m, n) \cdot \sigma_1(\tilde{A}) \cdot \delta
\]

where \(\sigma_k(A)\) is the \(k\)th exact singular value of the matrix \(A\) and \(\tilde{\sigma}_k(A)\) is the numerically obtained \(k\)th singular value, and \(p(m, n)\) is a “modest growth factor.”

---

\(^1\)See [http://www.netlib.org/lapack/lug/node97.html](http://www.netlib.org/lapack/lug/node97.html).
For brevity, write $A := A_{(1)}$ and $\tilde{A} := \text{fl}(A_{(1)})$.

Even in light of these representation and computation errors, we can extract useful information from our result by using the error bounds and **Weyl’s perturbation lemma**:

$$|\sigma_k(X) - \sigma_k(X + Y)| \leq \|Y\|_2.$$

We have

$$|\sigma_k(A) - \tilde{\sigma}_k(\tilde{A})| = |\sigma_k(A) - \sigma_k(\tilde{A}) + \sigma_k(\tilde{A}) - \tilde{\sigma}_k(\tilde{A})|$$

$$\leq \sqrt{14}\delta + |\sigma_k(\tilde{A}) - \tilde{\sigma}_k(\tilde{A})|$$

$$= \sqrt{14}\delta + |\sigma_k(\tilde{A}) - \sigma_k(\tilde{A} + E) + \sigma_k(\tilde{A} + E) - \tilde{\sigma}_k(\tilde{A})|$$

$$\leq (p(m, n)\sigma_1(\tilde{A}) + \sqrt{14})\delta + |\sigma_k(\tilde{A} + E) - \tilde{\sigma}_k(\tilde{A})|$$

$$\leq (4p(m, n)\tilde{\sigma}_1(\tilde{A}) + \sqrt{14})\delta,$$

assuming $p(m, n) \max\{\sigma_1(\tilde{A} + E), \sigma_1(\tilde{A})\} \leq 2$. 
Applying this to our case, and assuming that $p(m, n) \leq 10(m + n)$, we find

$$\left| \sigma_1(A_1) - 5.196152422706632 \right| \leq 1.517 \cdot 10^{-13}$$

$$\left| \sigma_2(A_1) - 1.805984985273179 \cdot 10^{-16} \right| \leq 1.517 \cdot 10^{-13};$$

hence, $\sigma_1(A_1) \neq 0$, but based on our error bounds we cannot exclude that $\sigma_2(A_1)$ might be 0.

We thus conclude that $r_1 \geq 1$ and that the distance of $A_1$ to the locus of rank-1 matrices is at most about $1.517 \cdot 10^{-13}$. 
Overview

1 Introduction

2 Basic tensor operations

3 Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms

4 Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability

5 References
It is uncommon to encounter tensors $\mathcal{A} \in \mathbb{F}^{n_1 \times n_2 \times \cdots \times n_d}$ with a multilinear rank that is exactly smaller than $(n_1, n_2, \ldots, n_d)$ because of numerical errors. However, tensors $\mathcal{A}$ can often lie close to a separable subspace $V_1 \otimes V_2 \otimes \cdots \otimes V_d$. This leads naturally to

The low multilinear rank approximation (LMLRA) problem

Given $\mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d}$ and a target multilinear rank $(r_1, \ldots, r_d)$, find a minimizer of

$$\min_{\text{mlrank}(\mathcal{B}) \leq (r_1, \ldots, r_d)} \| \mathcal{A} - \mathcal{B} \|_F$$

In other words, find the separable subspace $V_1 \otimes \cdots \otimes V_d$ with $\dim V_k = r_k$ that is closest to $\mathcal{A}$. 
Since $\text{mlrank}(B) = (r_1, \ldots, r_d)$ is equivalent to the existence of a separable subspace $V_1 \otimes \cdots \otimes V_d$ in which $B$ lives, we can write

$$B = (U_1, U_2, \ldots, U_d) \cdot S$$

where $U_k \in \mathbb{F}^{n_k \times r_k}$ can be chosen orthonormal by the existence of the HOSVD.

So graphically we want to approximate $A$ by

$$A \approx (U_1, U_2, U_3) \cdot S$$
After choosing the separable subspace, the optimal approximation is the **orthogonal projection** onto this subspace. Hence, the LMLRA problem is equivalent to

\[
\min_{U_k \in \text{St}_{n_k, r_k}} \left\| \mathcal{A} - P \langle U_1 \otimes \cdots \otimes U_d \rangle \mathcal{A} \right\|_F
\]

where \( \langle U \rangle \) denotes the linear subspace spanned by the basis \( U \), and \( \text{St}_{m,n} \) is the Stiefel manifold of \( m \times n \) matrices with orthonormal columns.
Proposition (V, Vandebril, and Meerbergen, 2012)

Let $U_1 \otimes \cdots \otimes U_d$ be a tensor basis of the separable subspace $V = V_1 \otimes \cdots \otimes V_d$. Then, the approximation error

$$\| A - PV A \|_F^2 = \sum_{k=1}^{d} \| \pi_{p_{k-1}} \cdots \pi_{p_1} A - \pi_{p_k} \pi_{p_{k-1}} \cdots \pi_{p_1} A \|_F^2,$$

where $\pi_j A = (U_j U_j^H) \cdot j A$ and $p$ is any permutation of $\{1, 2, \ldots, d\}$.

The proof is left as an exercise.
Note that \( \mathcal{A} - \pi_j \mathcal{A} = (I - U_j U_j^H) \cdot j \mathcal{A} \) is also a projection, which we denote by

\[
\pi_j \mathcal{A} := (I - U_j U_j^H) \cdot j \mathcal{A}.
\]

We may intuitively understand the proposition as follows. If

\[
\mathcal{A} \approx \hat{\mathcal{A}} := \pi_1 \pi_2 \pi_3 \mathcal{A} = (U_1 U_1^H, U_2 U_2^H, U_3 U_3^H) \cdot \mathcal{A},
\]

then an error expression is

\[
\| \mathcal{A} - \pi_1 \pi_2 \pi_3 \mathcal{A} \|^2 = \| \pi_1 \mathcal{A} \|^2 + \| \pi_2 \pi_1 \mathcal{A} \|^2 + \| \pi_3 \pi_1 \pi_2 \mathcal{A} \|^2.
\]
Since orthogonal projections only decrease unitarily invariant norms, we also get the following corollary.

**Corollary**

Let $U_1 \otimes \cdots \otimes U_d$ be a tensor basis of the separable subspace $V = V_1 \otimes \cdots \otimes V_d$. Then, the approximation error satisfies

$$\| \mathcal{A} - \mathcal{P}_V \mathcal{A} \|_F^2 \leq \sum_{k=1}^{d} \| \pi_k^\perp \mathcal{A} \|_F^2,$$

where $\pi_j \mathcal{A} = (U_j U_j^H) \cdot j \mathcal{A}$. 

We may intuitively understand this corollary as follows. If

\[ \mathcal{A} \approx \hat{\mathcal{A}} := \pi_1 \pi_2 \pi_3 \mathcal{A} = (U_1 U_1^H, U_2 U_2^H, U_3 U_3^H) \cdot \mathcal{A}, \]

then an upper bound is

\[ \| \mathcal{A} - \pi_1 \pi_2 \pi_3 \mathcal{A} \|^2 \leq \| \pi_1 \mathcal{A} \|^2 + \| \pi_2 \mathcal{A} \|^2 + \| \pi_3 \mathcal{A} \|^2 \]
A closed solution of the LMLRA problem

$$\min_{U_k \in \text{St}_{n_k}, r_k} \| A - P \langle u_1 \otimes \cdots \otimes u_d \rangle A \|_F$$

is not known.

However, we can use foregoing error expressions for choosing good, even quasi-optimal, separable subspaces to project onto.
The idea of the **truncated HOSVD** (T-HOSVD) is minimizing the upper bound on the error:

$$\|\mathcal{A} - \pi_1 \pi_2 \pi_3 \mathcal{A}\|^2 \leq \|\pi_{1}^\perp \mathcal{A}\|^2 + \|\pi_{2}^\perp \mathcal{A}\|^2 + \|\pi_{3}^\perp \mathcal{A}\|^2$$

If the upper bound is small, then evidently the error is also small.
Minimizing the upper bound results in

\[
\min_{\pi_1, \ldots, \pi_d} \| \mathcal{A} - \pi_1 \cdots \pi_d \mathcal{A} \|_F^2 \leq \min_{\pi_1, \ldots, \pi_d} \sum_{k=1}^{d} \| \pi_k^\perp \mathcal{A} \|_F^2
\]

\[
= \sum_{k=1}^{d} \min_{\pi_k} \| \pi_k^\perp \mathcal{A} \|_F^2
\]

\[
= \sum_{k=1}^{d} \min_{U_k \in \text{St}_{n_k, r_k}} \| \mathcal{A}(k) - U_k U_k^H \mathcal{A}(k) \|_F^2
\]

This has a closed form solution, namely the optimal $\overline{U}_k$ should contain the $r_k$ dominant left singular vectors. That is, writing the compact SVD of $\mathcal{A}(k)$ as

\[
\mathcal{A}(k) = U_k \Sigma_k Q_k^T,
\]

then $\overline{U}_k$ contains the first $r_k$ columns of $U_k$. 
The resulting **T-HOSVD algorithm** is thus but a minor modification of the HOSVD algorithm.

**Algorithm 2: T-HOSVD Algorithm**

**input**: A tensor $\mathcal{A} \in \mathbb{F}^{n_1 \times n_2 \times \cdots \times n_d}$

**input**: A target multilinear rank $(r_1, r_2, \ldots, r_d)$.

**output**: The components $(\overline{U}_1, \overline{U}_2, \ldots, \overline{U}_d)$ of the T-HOSVD basis

**output**: Coefficients array $\overline{S} \in \mathbb{F}^{r_1 \times r_2 \times \cdots \times r_d}$

**for** $k = 1, 2, \ldots, d$ **do**

- Compute the compact SVD $\mathcal{A}_{(k)} = U_k \Sigma_k Q_k^H$;
- Let $\overline{U}_k$ contain the first $r_k$ columns of $U_k$;

**end**

$\overline{S} \leftarrow (\overline{U}_1^H, \overline{U}_2^H, \ldots, \overline{U}_d^H) \cdot \mathcal{A}$;
Assume that we truncate a tensor in $\mathbb{F}^{n \times \cdots \times n}$ to multilinear rank $(r, \ldots, r)$. The computational complexity of standard T-HOSVD is

$$\mathcal{O} \left( dn^{d+1} + \sum_{k=1}^{d} n^{d+1-k} r^k \right)$$ operations.
The resulting approximation is quasi-optimal.

**Proposition (Hackbusch, 2012)**

Let $\mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d}$, and let $\mathcal{A}^*$ be the best rank-$(r, \ldots, r)$ approximation to $\mathcal{B}$, i.e.,

$$
\| \mathcal{A} - \mathcal{A}^* \|_F = \min_{\text{mlrank}(\mathcal{B}) \leq (r, \ldots, r)} \| \mathcal{A} - \mathcal{B} \|_F.
$$

Then, the rank-$(r, \ldots, r)$ T-HOSVD approximation $\mathcal{A}_T$ is a quasi best approximation:

$$
\| \mathcal{A} - \mathcal{A}_T \|_F \leq \sqrt{d} \| \mathcal{A} - \mathcal{A}^* \|_F.
$$
The idea of the **sequentially truncated HOSVD** (ST-HOSVD) is sequentially choosing projections with the aim of minimizing the error expression:

\[
\| \mathcal{A} - \pi_1 \pi_2 \pi_3 \mathcal{A} \|^2 = \| \pi_1^\perp \mathcal{A} \|^2 + \| \pi_2^\perp \pi_1 \mathcal{A} \|^2 + \| \pi_3^\perp \pi_1 \pi_2 \mathcal{A} \|^2
\]
ST-HOSVD greedily minimizes the foregoing error expression. That is, it computes

\[ \hat{\pi}_1 = \arg \min_{\pi_1} \| \pi_1 A_1 \|_2^2 \]

\[ \hat{\pi}_2 = \arg \min_{\pi_2} \| \pi_2 \hat{\pi}_1 A_1 \|_2^2 \]

\[ \vdots \]

\[ \hat{\pi}_d = \arg \min_{\pi_d} \| \pi_d \hat{\pi}_{d-1} \cdots \hat{\pi}_2 \hat{\pi}_1 A_1 \|_2^2 \]
In practice, \( \min_{\pi_k} \| \pi_k^{\perp} \pi_{k-1} \cdots \pi_1 \mathcal{A} \|_F \) is computed as follows. As \( \pi_j \) are orthogonal projections, we can write them as

\[
\pi_j \mathcal{A} := (\hat{U}_j \hat{U}_j^H) \cdot_j \mathcal{A} = \hat{U}_j \cdot_j (\hat{U}_j^H \cdot_j \mathcal{A}).
\]

Therefore,

\[
\min_{U_k \in \text{St}_{n_k, r_k}} \| U_k U_k^H \mathcal{A}(k)(\hat{U}_1 \hat{U}_1^H \otimes \cdots \otimes \hat{U}_{k-1} \hat{U}_{k-1}^H \otimes I \otimes \cdots \otimes I)^T \|_F
\]

\[
= \min_{U_k} \| U_k U_k^H \mathcal{A}(k)(\hat{U}_1^H \otimes \cdots \otimes \hat{U}_{k-1}^H \otimes I \otimes \cdots \otimes I)^T \|_F
\]

\[
= \min_{U_k} \| U_k U_k^H S^{k-1}_{(k)} \|_F,
\]

where we define

\[
S^{k-1} := (\hat{U}_1, \ldots, \hat{U}_{k-1}, I, \ldots, I)^H \cdot \mathcal{A} = \hat{U}_{k-1}^H \cdot_{k-1} S^{k-2}.
\]

Recall that the solution of \( \min_{U_k} \| U_k U_k^H S^{k-1}_{(k)} \|_F \) is given by the rank-\(r_k\) truncated SVD of \( S^{k-1}_{(k)} \).
Visually, here’s what happens for a third-order tensor.

\[ S^0 = A \]
\[ S^1_{(1)} = \hat{U}_1^H S^0 \]
\[ S^2_{(2)} = \hat{U}_2^H S^1 \]
\[ S^3_{(3)} = \hat{U}_3^H S^2 \]
The resulting **ST-HOSVD algorithm** is thus but a minor modification of the T-HOSVD algorithm.

**Algorithm 3: ST-HOSVD Algorithm**

- **input**: A tensor \( \mathcal{A} \in \mathbb{F}^{n_1 \times n_2 \times \cdots \times n_d} \)
- **input**: A target multilinear rank \((r_1, r_2, \ldots, r_d)\).
- **output**: The components \((\hat{U}_1, \hat{U}_2, \ldots, \hat{U}_d)\) of the ST-HOSVD basis
- **output**: Coefficients array \( \hat{S} \in \mathbb{F}^{r_1 \times r_2 \times \cdots \times r_d} \)

\(
\hat{S} \leftarrow \hat{A}; \\
\text{for } k = 1, 2, \ldots, d \text{ do} \\
\quad \text{Compute the compact SVD } S_{(k)} = U_k \Sigma_k Q_k^H; \\
\quad \text{Let } \hat{U}_k \text{ contain the first } r_k \text{ columns of } U_k; \\
\quad \hat{S} \leftarrow \hat{U}_k^H \cdot_k \hat{S}; \\
\text{end}
\)
Assume that we truncate a tensor in $\mathbb{F}^{n_1 \times \cdots \times n_d}$ to multilinear rank $(r, \ldots, r)$. The computational complexity of ST-HOSVD is

$$O \left( n^{d+1} + 2 \sum_{k=1}^{d} n^{d+1-k} r^k \right)$$ operations,

which compares favorably versus T-HOSVD’s

$$O \left( dn^{d+1} + \sum_{k=1}^{d} n^{d+1-k} r^k \right)$$ operations.

Note that much larger speedups are possible for uneven mode sizes $n_1 \geq n_2 \geq \cdots \geq n_d \geq 2$, as you will show in the problem sessions.
The resulting approximation is also quasi-optimal.

**Proposition (Hackbusch, 2012)**

Let $\mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d}$, and let $\mathcal{A}^*$ be the best rank-$(r, \ldots, r)$ approximation to $\mathcal{A}$, i.e.,

$$
\| \mathcal{A} - \mathcal{A}^* \|_F = \min_{m\text{rank}(\mathcal{B}) \leq (r, \ldots, r)} \| \mathcal{A} - \mathcal{B} \|_F.
$$

Then, the rank-$(r, \ldots, r)$ ST-HOSVD approximation $\mathcal{A}_S$ is a quasi best approximation:

$$
\| \mathcal{A} - \mathcal{A}_S \|_F \leq \sqrt{d} \| \mathcal{A} - \mathcal{A}^* \|_F.
$$
Overview

1 Introduction

2 Basic tensor operations

3 Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms

4 Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability

5 References
Overview

1. Introduction
2. Basic tensor operations
3. Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms
4. Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability
5. References
The tensor rank decomposition (CPD) expresses a tensor $\mathcal{A} \in V_1 \otimes \cdots \otimes V_d$ as a **minimum-length** linear combination of rank-1 tensors:

$$\mathcal{A} = \sum_{i=1}^{r} \lambda_i a_i^1 \otimes \cdots \otimes a_i^d,$$

where $a_i^k \in V_k$.

Often the scalars $\lambda_i$ are absorbed into the $a_i^k \in V_k$.

The **rank** of $\mathcal{A}$ is the length of any of its tensor rank decompositions.
Tensor rank is a considerably more difficult subject for $d \geq 3$ than the multilinear rank. For example,

- the **maximum rank** of a tensor space $\mathbb{F}^{n_1} \otimes \cdots \otimes \mathbb{F}^{n_d}$ is not known in general;

- the **typical ranks** of a tensor space $\mathbb{F}^{n_1} \otimes \cdots \otimes \mathbb{F}^{n_d}$, i.e., those ranks occurring on nonempty Euclidean-open subsets, are not known in general;

- the rank of a real tensor can decrease when taking a **field extension**, contrary to matrix and multilinear rank; and

- computing tensor rank is **NP Hard**.
Tensor rank is invariant under invertible multilinear multiplications with \( A_1 \otimes \cdots \otimes A_d \), where \( A_k : V_k \rightarrow W_k \) are invertible linear maps.

Let \( \mathcal{A} = \sum_{i=1}^{r} b_i^1 \otimes \cdots \otimes b_i^d \). Since
\[
(A_1, \ldots, A_d) \cdot \mathcal{A} = \sum_{i=1}^{r} (A_1 b_i^1) \otimes \cdots \otimes (A_d b_i^d),
\]
we have \( \text{rank}(\mathcal{A}) \leq \text{rank}((A_1, \ldots, A_d) \cdot \mathcal{A}) \). And so
\[
\text{rank}(\mathcal{A}) \leq \text{rank}((A_1, \ldots, A_d) \cdot \mathcal{A}) \leq \text{rank}((A_1^{-1}, \ldots, A_d^{-1}) \cdot ((A_1, \ldots, A_d) \cdot \mathcal{A})) = \text{rank}(\mathcal{A}).
\]
Overview

1. Introduction
2. Basic tensor operations
3. Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms
4. Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability
5. References
Another issue with tensor rank is that the set
\[ S_{\leq r} := \{ \mathcal{A} \in \mathbb{F}_{n_1 \times \cdots \times n_d} \mid \text{rank}(\mathcal{A}) \leq r \} \]
is not closed in general, i.e., \( S_{\leq r} \neq \overline{S_{\leq r}} \).

For example, for any linearly independent \( x, y \in \mathbb{R}^n \), we have
\[
\lim_{\epsilon \to 0} \left( \frac{1}{\epsilon} (x + \epsilon y) \otimes^3 - \frac{1}{\epsilon} x \otimes^3 \right) = y \otimes x \otimes x + x \otimes y \otimes x + x \otimes x \otimes y;
\]
evidently, the tensors in the sequence have rank bounded by 2, but it can be shown that the limit has rank 3.
Consider the Euclidean closure of $S_{\leq r}$:

$$
\overline{S_{\leq r}} := \{ \lim_{\epsilon \to 0} \mathcal{A}_\epsilon, \text{ where } \mathcal{A}_\epsilon \in S_{\leq r} \}.
$$

If $\mathcal{A} \in \overline{S_{\leq r}} \setminus \overline{S_{\leq r-1}}$, then we say that $\mathcal{A}$ has **border rank** equal to $r$.

It turns out that for $\mathbb{F} = \mathbb{C}$, the Euclidean closure of $S_{\leq r}$ coincides with its closure in the Zariski topology. That is, $\overline{S_{\leq r}}$ is an **algebraic**, even **projective**, **variety**, i.e., the zero set of a system of homogeneous polynomial equations.

For $\mathbb{F} = \mathbb{R}$, both $S_{\leq r}$ and $\overline{S_{\leq r}}$ are **semi-algebraic sets**, i.e., the solution set of a system of polynomial equalities and inequalities.
Overview

1 Introduction

2 Basic tensor operations

3 Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms

4 Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability

5 References
A key property of the tensor rank decomposition is that the decomposition of $\mathcal{A}$ as a sum of rank-1 tensors $\mathcal{A}_i$ is often unique.

We say that $\mathcal{A} \in \mathbb{F}^{n_1 \times \cdots \times n_d}$ is r-identifiable if the set of rank-1 tensors $\{\mathcal{A}_1, \ldots, \mathcal{A}_r\}$ whose sum is $\mathcal{A}$, i.e.,

$$\mathcal{A} = \mathcal{A}_1 + \cdots + \mathcal{A}_r,$$

is uniquely determined by $\mathcal{A}$. 
Note that the components of a rank-1 tensor \( \mathcal{A} \in \mathbb{F}^{n_1} \otimes \cdots \otimes \mathbb{F}^{n_d} \) are themselves also uniquely determined (in projective space) by \( \mathcal{A} \). Precisely, the points

\[
[a_k] \in \mathbb{P}(\mathbb{F}^{n_k})
\]

are uniquely determined given \( \mathcal{A} = a_1 \otimes \cdots \otimes a_d \).
This $r$-identifiability is **radically different** from the matrix case ($d = 2$). Indeed, if $A \in \mathbb{F}^{m \times n}$ is a rank-$r$ matrix, then

$$A = UV^T = (UX)(X^{-1}V^T) \quad \text{for all } X \in \text{GL}_r(\mathbb{F})$$

For a generic choice of $X$, i.e., outside of some Zariski-closed set, $(UX)_i \neq \alpha u_{\pi_i}$, so that the tensor rank decompositions are distinct.

Note that in the matrix case there is even a positive-dimensional family of distinct decompositions! (Can you prove this?)
A classic result on \( r \)-identifiability of CPDs is **Kruskal’s lemma**, which relies on the notion of the **Kruskal rank** of a set of vectors.

**Definition (Kruskal, 1977)**

The Kruskal rank \( k_V \) of a set of vectors \( V = \{v_1, \ldots, v_r\} \subset \mathbb{F}^n \) is the largest \( k \) integer such that every subset of \( k \) vectors of \( V \) is linearly independent.

For example,

- \( \{v, v\} \) has Kruskal rank 1;
- \( \{v, w, v\} \) has Kruskal rank 1; and
- \( \{v, w, v + w\} \) has Kruskal rank 2 if \( v \) and \( w \) are linearly independent.
Kruskal proved, among others, the following result.

**Theorem (Kruskal, 1977)**

Let \( A = \sum_{i=1}^{r} a_i^1 \otimes a_i^2 \otimes a_i^3 \) and \( A_k := [a_i^k]_{i=1}^{r} \). If \( k_{A_1}, k_{A_2}, k_{A_3} > 1 \) and

\[
r \leq \frac{1}{2} (k_{A_1} + k_{A_2} + k_{A_3} - 2)
\]

then \( A \) is \( r \)-identifiable.

The condition \( k_{A_1} > 1 \) is necessary for \( r \geq 2 \) because otherwise \( A \in \langle \mathbf{v} \rangle \otimes \mathbb{F}^{n_2} \otimes \mathbb{F}^{n_3} \cong \mathbb{F}^{n_2 \times n_3} \), and likewise for the other factors.
Computing the Kruskal rank of $r$ vectors in $\mathbb{F}^n$ is very expensive, in general, as one needs to compute the ranks of all $\binom{r}{k}$ subsets of $k$ vectors for $k = 1, \ldots, \min\{r, n\}$. Computing one of these ranks already has a complexity of $nk^2$.

Notwithstanding this limitation, applying Kruskal’s lemma is a popular technique for verifying that a tensor given as the sum of $r$ rank-1 tensors has rank equal to $r$. Indeed, a rank-$r$ tensor is never $r'$-identifiable with $r' > r$. 
Kruskal’s lemma can also be applied to higher-order tensors

\[ \mathcal{A} \in V_1 \otimes \cdots \otimes V_d \]

simply by grouping the factors:

\[ \mathcal{A} \in (V_{\pi_1} \otimes \cdots \otimes V_{\pi_s}) \otimes (V_{\pi_{s+1}} \otimes \cdots \otimes V_{\pi_t}) \otimes (V_{\pi_{t+1}} \otimes \cdots \otimes V_{\pi_d}) \]

where \( 1 \leq s < t \leq d \) and \( \pi \) is a permutation of \( \{1, \ldots, d\} \).

In other words, Kruskal’s lemma is applied to the reshaped tensor (coordinate array).
While $r$-identifiability seems like a special property admitted by only few tensors, the phenomenon is very general. It is an open problem to prove the following conjecture:

**Conjecture (Chiantini, Ottaviani, V, 2014)**

Let $n_1 \geq n_2 \geq \cdots \geq n_d \geq 2$, $d \geq 3$. If $r < \frac{\prod_{k=1}^{d} n_k}{1+\sum_{k=1}^{d}(n_k-1)}$, then $\mathbb{F}^{n_1} \otimes \cdots \otimes \mathbb{F}^{n_d}$ is generically $r$-identifiable (there exists a proper Zariski-closed subset $Z$ of $S_{\leq r}$ such that every $\mathcal{A} \in S_{\leq r} \setminus Z$ is $r$-identifiable), unless:

1. $(n_1, n_2, n_3) = (4, 4, 3)$ and $r = 5$;
2. $(n_1, n_2, n_3) = (4, 4, 4)$ and $r = 6$;
3. $(n_1, n_2, n_3) = (6, 6, 3)$ and $r = 8$;
4. $(n_1, n_2, n_3, n_4) = (n, n, 2, 2)$ and $r = 2n - 1$, $n \geq 2$;
5. $(n_1, n_2, n_3, n_4, n_5) = (2, 2, 2, 2, 2)$ and $r = 5$; and
6. $n_1 > \prod_{k=2}^{d} n_k - \sum_{k=2}^{d}(n_k - 1) =: c$ and $r \geq c$. 
Overview

1. Introduction
2. Basic tensor operations
3. Tucker decomposition
   - Multilinear rank
   - Higher-order singular value decomposition
   - Numerical issues
   - Truncation algorithms
4. Tensor rank decomposition
   - Rank
   - Border rank
   - Identifiability
5. References
References for basic tensor operations

References for Tucker decomposition

References for tensor rank decomposition