Tensor decompositions and their applications

Lecture 2: Tucker decomposition

Nick Vannieuwenhoven (KU Leuven)

1. Introduction (5')
2. Multilinear algebra* (40')
3. Tucker decomposition (15')
4. Higher-order singular value decomposition (40')
5. Application: dimensionality reduction (5')
6. Conclusions


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
JORGE CHAM © 2013
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
WWW.PHDCOMICS.COM

Overview

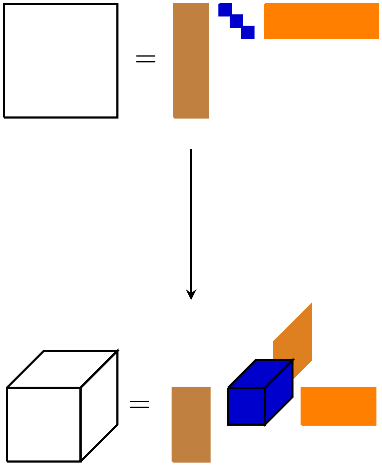
1. Introduction (5')
2. Multilinear algebra* (40')
3. Tucker decomposition (15')
4. Higher-order singular value decomposition (40')
5. Application: dimensionality reduction (5')
6. Conclusions

Linear algebra

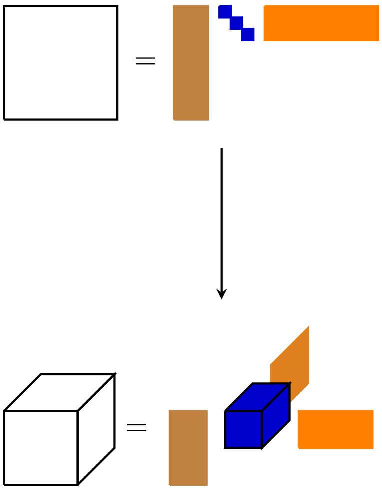
is extended to

Multilinear algebra

Singular value decomposition



Tucker decomposition



One interpretation of the compact SVD USV^T of a rank-r matrix $A \in \mathbb{k}^{m\times n}$ is that it identifies

- an orthonormal basis U of the column space of A,
- an orthonormal basis V of the row space of A, and
- the coordinates S of A relative to the bases U and V.

This interpretation generalizes straightforwardly to tensors. The resulting orthogonal **Tucker decomposition** can be used for **dimensionality reduction** to the smaller blue core tensor.


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Overview

1. Introduction (5')
2. Multilinear algebra* (40')
3. Tucker decomposition (15')
4. Higher-order singular value decomposition (40')
5. Application: dimensionality reduction (5')
6. Conclusions

Tensor product

The **tensor product of two vector spaces** V and W with respective bases $\{\mathsf{v}_1, \ldots, \mathsf{v}_m\}$ and $\{\mathsf{w}_1, \ldots, \mathsf{w}_n\}$ is defined as the vector space

$$V \otimes W := \operatorname{span}(\mathsf{v}_1 \otimes \mathsf{w}_1, \ldots, \mathsf{v}_1 \otimes \mathsf{w}_n, \ldots, \mathsf{v}_m \otimes \mathsf{w}_1, \ldots, \mathsf{v}_m \otimes \mathsf{w}_n),$$

where the **tensor product** of vectors $\otimes$ is **bilinear**:

$$\begin{aligned}(\mathsf{a}+\mathsf{b}) \otimes \mathsf{c} &= \mathsf{a} \otimes \mathsf{c} + \mathsf{b} \otimes \mathsf{c},\\ \mathsf{a} \otimes (\mathsf{b}+\mathsf{c}) &= \mathsf{a} \otimes \mathsf{b} + \mathsf{a} \otimes \mathsf{c},\\ (\alpha\mathsf{a}) \otimes \mathsf{b} &= \alpha(\mathsf{a} \otimes \mathsf{b}) = \mathsf{a} \otimes (\alpha\mathsf{b}).\end{aligned}$$

It can be shown that $\dim V \otimes W = \dim V \cdot \dim W$.

Every tensor $\mathcal{T} \in V \otimes W$ that is of the form

$$\mathcal{T} = \mathrm{a} \otimes \mathrm{b}$$

can also be expressed as a linear combination of the foregoing vectors $\mathrm{v}_i \otimes \mathrm{w}_j$. Indeed, if

$$\mathrm{a} = a_1\mathrm{v}_1 + a_2\mathrm{v}_2 + \cdots + a_m\mathrm{v}_m \quad \text{and} \quad \mathrm{b} = b_1\mathrm{w}_1 + b_2\mathrm{w}_2 + \cdots + a_n\mathrm{w}_n$$

then we have

$$\mathcal{T} = \left(\sum_{i=1}^{m} a_i\mathrm{v}_i\right) \otimes \mathrm{b} = \sum_{i=1}^{m}(a_i\mathrm{v}_i) \otimes \mathrm{b} = \sum_{i=1}^{m} a_i\mathrm{v}_i \otimes \left(\sum_{j=1}^{n} b_j\mathrm{w}_j\right)$$
$$= \sum_{i=1}^{m}\sum_{j=1}^{n}(a_i\mathrm{v}_i) \otimes (b_j\mathrm{w}_j) = \sum_{i=1}^{m}\sum_{j=1}^{n}(a_i b_j)\mathrm{v}_i \otimes \mathrm{w}_j$$

In other words, $\mathcal{T} = \mathrm{a} \otimes \mathrm{b}$ can be represented in coordinates by a rank-1 matrix ab^T!

The tensor product generalizes to an **arbitrary number of vector spaces** $V_1, \ldots, V_d$. If V_k has basis $\{\mathrm{v}_1^k, \ldots, \mathrm{v}_{n_k}^k\}$, then

$$V_1 \otimes \cdots \otimes V_d := \mathrm{span}\Big(\{\otimes(\mathrm{v}_{i_1}^1, \ldots, \mathrm{v}_{i_d}^d)\}_{i_1,\ldots,i_d=1}^{n_1,\ldots,n_d} \Big),$$

where the tensor product $\otimes(\cdot, \ldots, \cdot)$ can be defined as

$$\otimes(\mathrm{a}^1, \ldots, \mathrm{a}^d) = \mathrm{a}^1 \otimes (\mathrm{a}^2 \otimes (\cdots \otimes (\mathrm{a}^{d-1} \otimes \mathrm{a}^d))) = \mathrm{a}^1 \otimes \mathrm{a}^2 \otimes \cdots \otimes \mathrm{a}^d$$

The tensor product generalizes to an **arbitrary number of vector spaces** $V_1, \ldots, V_d$. If V_k has basis $\{\mathrm{v}_1^k, \ldots, \mathrm{v}_{n_k}^k\}$, then

$$V_1 \otimes \cdots \otimes V_d := \operatorname{span}\Big(\{\otimes(\mathrm{v}_{i_1}^1, \ldots, \mathrm{v}_{i_d}^d)\}_{i_1,\ldots,i_d=1}^{n_1,\ldots,n_d} \Big),$$

where the tensor product $\otimes(\cdot, \ldots, \cdot)$ can be defined as

$$\otimes(\mathrm{a}^1, \ldots, \mathrm{a}^d) = \mathrm{a}^1 \otimes (\mathrm{a}^2 \otimes (\cdots \otimes (\mathrm{a}^{d-1} \otimes \mathrm{a}^d))) = \mathrm{a}^1 \otimes \mathrm{a}^2 \otimes \cdots \otimes \mathrm{a}^d$$

The **tensor product is multilinear**:

$$\begin{aligned} \mathrm{a}^1 \otimes \cdots \otimes \mathrm{a}^{k-1} \otimes \Big(\alpha \mathrm{a}^k + \beta \mathrm{b}^k\Big) \otimes \mathrm{a}^{k+1} \otimes \cdots \otimes \mathrm{a}^d \\ = \alpha \mathrm{a}^1 \otimes \cdots \otimes \mathrm{a}^d + \beta \mathrm{a}^1 \otimes \cdots \otimes \mathrm{a}^{k-1} \otimes \mathrm{b}^k \otimes \mathrm{a}^{k+1} \otimes \cdots \otimes \mathrm{a}^d \end{aligned}$$

for all $k = 1, 2, \ldots, d$.

An element $\mathcal{A} \in V_1 \otimes \cdots \otimes V_d$ that is expressed as

$$\mathcal{A} = \mathrm{a}^1 \otimes \cdots \otimes \mathrm{a}^d$$

is called a **pure**, **simple**, **elementary**, or **rank**-1 tensor.

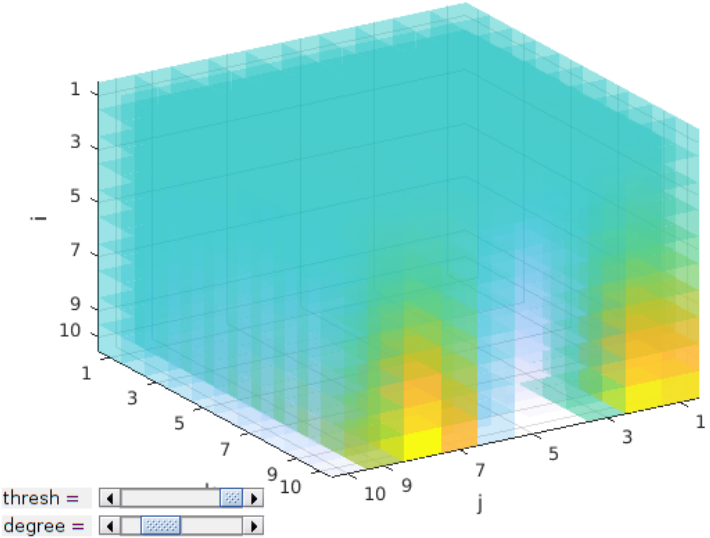
If $\mathrm{a}^i = a^i_1 \mathrm{v}^i_1 + a^i_2 \mathrm{v}^i_2 + \cdots + a^i_{n_i} \mathrm{v}^i_{n_i}$, then, as before, we have

$$\mathcal{A} = \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} (a^1_{i_1} \cdots a^d_{i_d})\, \mathrm{v}^1_{i_1} \otimes \cdots \otimes \mathrm{v}^d_{i_d}.$$

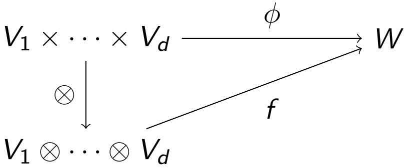
Hence, $\mathcal{A}$ is represented in coordinates by a special $n_1 \times \cdots \times n_d$ coordinate d-array $\mathcal{A}$ in which

$$\mathcal{A}_{i_1, i_2, \ldots, i_d} = a^1_{i_1} a^2_{i_2} \cdots a^d_{i_d}.$$

A rank-1 tensor visualized in Tensorlab:



The **universal property** of the tensor product states that for every **multilinear map** $\phi : V_1 \times \cdots \times V_d \to W$ there is a **unique linear map** $f : V_1 \otimes \cdots \otimes V_d \to W$ such that the diagram



commutes.

A nice consequence is that it enables easy definitions of linear maps acting on tensors.

Flattening

A **flattening** is the linear map induced via the universal property of the multilinear map

$$\cdot_{(\pi;\tau)} : V_1 \times \cdots \times V_d \to (V_{\pi_1} \otimes \cdots \otimes V_{\pi_k}) \otimes (V_{\tau_1} \otimes \cdots \otimes V_{\tau_{d-k}})^*$$
$$(\mathrm{a}^1, \ldots, \mathrm{a}^d) \mapsto (\mathrm{a}^{\pi_1} \otimes \cdots \otimes \mathrm{a}^{\pi_k})(\mathrm{a}^{\tau_1} \otimes \cdots \otimes \mathrm{a}^{\tau_{d-k}})^T$$

where $\cdot^*$ denotes the dual. It is a technique to **turn a tensor into a matrix** in many ways, by **forgetting some of the tensor structure**.

It is common to use the following shorthand notations in the literature:[1]

$$\mathcal{T}_{(k)} := \mathcal{T}_{(k;1,\ldots,k-1,k+1,\ldots,d)} \quad \text{and} \quad \mathrm{vec}(\mathcal{T}) := \mathcal{T}_{(1,\ldots,d;\emptyset)}.$$

[1] Some authors define $\mathcal{T}_{(k)} = \mathcal{T}_{(k;k+1,\ldots,d,1,\ldots,k-1)}$.

For example, if

$$\mathcal{T} = \sum_{i=1}^{r} \mathrm{a}_i \otimes \mathrm{b}_i \otimes \mathrm{c}_i \quad \in V_1 \otimes V_2 \otimes V_3$$

then the three standard flattenings are

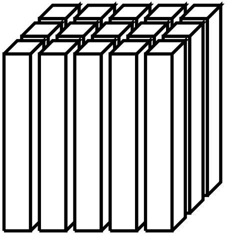
$$\mathcal{T}_{(1)} = \sum_{i=1}^{r} \mathrm{a}_i(\mathrm{b}_i \otimes \mathrm{c}_i)^T \quad \in V_1 \otimes (V_2 \otimes V_3)^*,$$

$$\mathcal{T}_{(2)} = \sum_{i=1}^{r} \mathrm{b}_i(\mathrm{a}_i \otimes \mathrm{c}_i)^T \quad \in V_2 \otimes (V_1 \otimes V_3)^*,$$

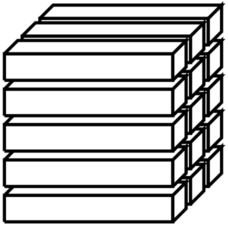
$$\mathcal{T}_{(2)} = \sum_{i=1}^{r} \mathrm{c}_i(\mathrm{a}_i \otimes \mathrm{b}_i)^T \quad \in V_3 \otimes (V_1 \otimes V_2)^*.$$

In coordinates, flattenings can be defined as follows. Let $\mathcal{T}$ be an $n_1 \times n_2 \times \cdots \times n_d$ be a d-array over $\mathbb{k}$. Then, we can associate d vector spaces defined by these coordinates.

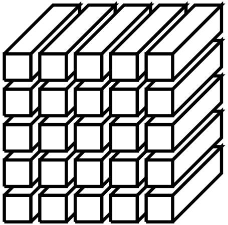
For example, a third-order tensor has 3 **associated vector spaces**:



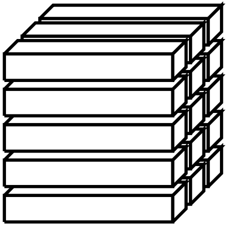
Mode-1 vectors (in $\mathbb{k}^{n_1}$)



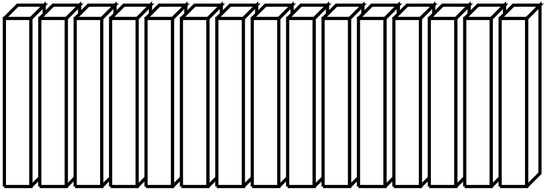
Mode-2 vectors (in $\mathbb{k}^{n_2}$)

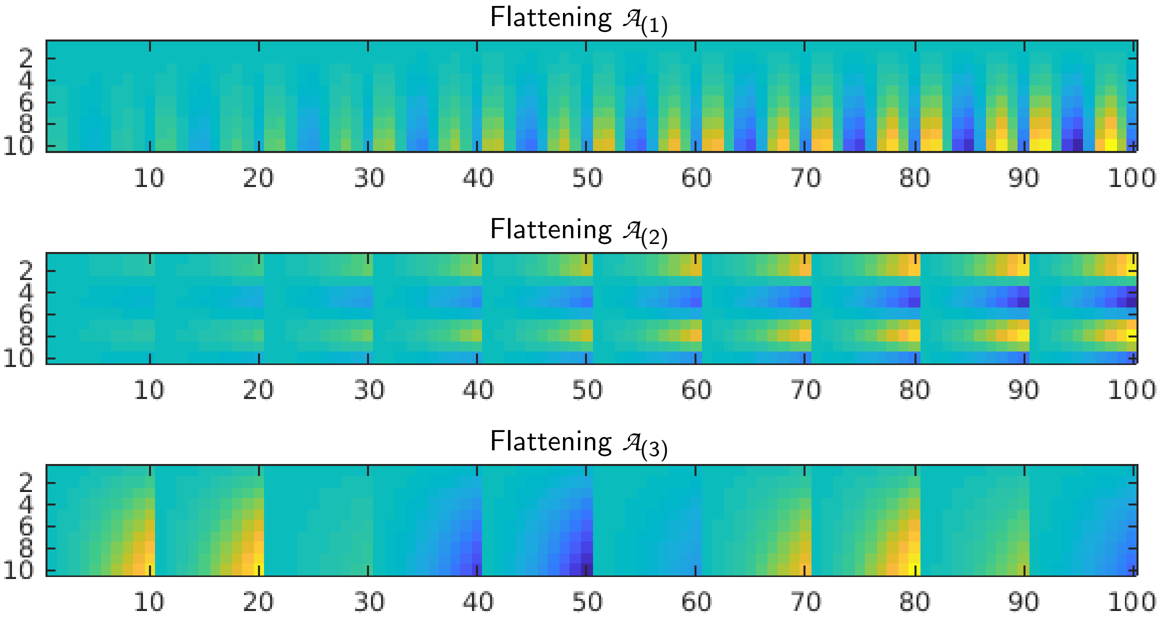


Mode-3 vectors (in $\mathbb{k}^{n_3}$)

$\mathcal{A} =$  $\in \mathbb{k}^{n_1 \times n_2 \times n_3}$

Mode-2 **flattening**

$\mathcal{A}_{(2)} =$  $\in \mathbb{k}^{n_2 \times n_1 n_3}$


Flattening $\mathcal{A}_{(1)}$
2 4 6 8 10
10 20 30 40 50 60 70 80 90 100
Flattening $\mathcal{A}_{(2)}$
2 4 6 8 10
10 20 30 40 50 60 70 80 90 100
Flattening $\mathcal{A}_{(3)}$
2 4 6 8 10
10 20 30 40 50 60 70 80 90 100

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

For example, an implementation of flattenings in Julia looks like this:

```
function flatten(A, pi, tau)
    Aperm = permutedims([pi; tau])
    Ak = reshape(Aperm, prod(size(A)[pi]), :)
    return Ak
end
```

All flattenings $\mathcal{A}_{(1,\ldots,k;k+1,\ldots,d)}$ in which the order of the factors is not changed can be implemented on a computer for free, i.e., they only need `reshape`.

Multilinear multiplication

The **tensor product of linear maps** $A_i : V_i \to W_i$, where V_i, W_i are finite-dimensional vector spaces, 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 applied to the multilinear map

$$\begin{aligned}(A_1, \ldots, A_d) : V_1 \times \cdots \times V_d &\to W_1 \otimes \cdots \otimes W_d,\\ (\mathrm{v}^1, \ldots, \mathrm{v}^d) &\mapsto (A_1 \mathrm{v}^1) \otimes \cdots \otimes (A_d \mathrm{v}^d).\end{aligned}$$

We denote the induced linear map by $A_1 \otimes \cdots \otimes A_d$.

Consequently, by the universal property,

$$(A_1 \otimes \cdots \otimes A_d)(\mathrm{v}^1 \otimes \cdots \otimes \mathrm{v}^d) = (A_1 \mathrm{v}^1) \otimes \cdots \otimes (A_d \mathrm{v}^d).$$

For general tensors $\mathcal{A} = \sum_{i=1}^r \mathrm{a}_i^1 \otimes \cdots \otimes \mathrm{a}_i^d \in V_1 \otimes \cdots \otimes V_d$ we then have

$$\begin{aligned}(A_1 \otimes \cdots \otimes A_d)(\mathcal{A}) &= (A_1 \otimes \cdots \otimes A_d) \left(\sum_{i=1}^r \mathrm{a}_i^1 \otimes \cdots \otimes \mathrm{a}_i^d \right) \\ &= \sum_{i=1}^r (A_1 \otimes \cdots \otimes A_d)(\mathrm{a}_i^1 \otimes \cdots \otimes \mathrm{a}_i^d) \\ &= \sum_{i=1}^r (A_1 \mathrm{a}_i^1) \otimes \cdots \otimes (A_d \mathrm{a}_i^d)\end{aligned}$$

The shorthand notation

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

is commonly used in the literature. This operation is called **multilinear multiplication**.

The notation

$$A_k \cdot_k \mathcal{A} := (\mathrm{Id}, \ldots, \mathrm{Id}, A_k, \mathrm{Id}, \ldots, \mathrm{Id}) \cdot \mathcal{A}$$

is also used in the literature. This operation is called a **mode-k multiplication**.

Note that

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

Hence, a multilinear multiplication can be computed in practice as follows:

```
function multilinear_multiplication(As, T)
    n = size(T)
    m = [size(A,1) for A ∈ As]
    for k = 1 : length(As)
        T = reshape(T, n[k], :)
        T = transpose(T) * transpose(As[k])
    end
    T = reshape(T, m)
end
```

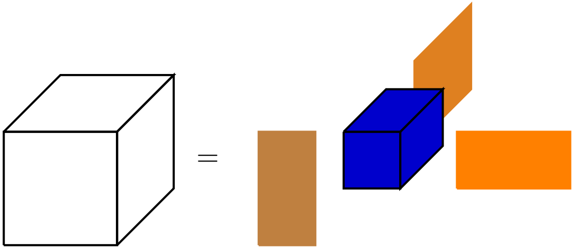

ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
JORGE CHAM © 2013
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
WWW.PHDCOMICS.COM

Overview

1. Introduction (5')
2. Multilinear algebra* (40')
3. Tucker decomposition (15')
4. Higher-order singular value decomposition (40')
5. Application: dimensionality reduction (5')
6. Conclusions

Tucker decomposition

The Tucker decomposition of $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ reveals a tensor product basis $A_1 \otimes \cdots \otimes A_d$ and the coordinates $\mathcal{C}$ of a separable subspace $V_1 \otimes \cdots \otimes V_d$ in which $\mathcal{A}$ lives.



Assume we have a basis $\{\mathrm{v}_1^k, \ldots, \mathrm{v}_{r_k}^k\}$ of the r_k-dimensional vector subspace $V_k \subset W_k$ and that $\mathcal{A} \in V_1 \otimes \cdots \otimes V_d \subset W_1 \otimes \cdots \otimes W_k$. Then, there exist coefficients $c_{i_1,\ldots,i_d} \in \mathbb{k}$ such that

$$\mathcal{A} = \sum_{i_1=1}^{r_1} \cdots \sum_{i_d=1}^{r_d} c_{i_1,\ldots,i_d} \mathrm{v}_{i_1}^1 \otimes \cdots \otimes \mathrm{v}_{i_d}^d.$$

This is called a **Tucker decomposition** of $\mathcal{A}$.

The $r_1 \times \cdots \times r_d$ d-array $\mathcal{C}$ is called the **core tensor**.

Another viewpoint is as follows. Let $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$. If there exist linear maps $A_i : V_i \to W_i$ and a tensor $\mathcal{C} \in V_1 \otimes \cdots \otimes V_d$ such that

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

then this expression is a **Tucker decomposition** of $\mathcal{A}$.

Multilinear rank

We say that $V_1 \otimes \cdots \otimes V_d$ is the **minimal separable tensor subspace** $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ lives in if

$$\mathcal{A} \in V_1 \otimes \cdots \otimes V_d \subset W_1 \otimes \cdots \otimes W_d.$$

and there are no $V_k' \subset V_k$ with at least one of these containments strict such that $\mathcal{A} \in V_1' \otimes \cdots \otimes V_d'$.

Lemma

Let $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$. The minimal separable tensor subspace in which $\mathcal{A}$ lives is $V_1 \otimes \cdots \otimes V_d$ if and only if

$$V_k = \operatorname{span}(\mathcal{A}_{(k)})$$

for all $k = 1, 2, \ldots, d$.

Definition (Hitchcock, 1928)

The **multilinear rank** of $\mathcal{A}$ is the tuple containing the dimensions of the minimal subspaces V_k that comprise the minimal separable tensor subspace that $\mathcal{A}$ lives in:

$$\operatorname{mlrank}(\mathcal{A}) := (\dim V_1, \dim V_2, \dots, \dim V_d).$$

In case the matrix A lives in the minimal separable tensor subspace $V_1 \otimes V_2$, the multilinear rank is, by definition,

$$\mathrm{mlrank}(A) = (\dim V_1, \dim V_2) = \big(\mathrm{rank}(A_{(1)}), \mathrm{rank}(A_{(2)})\big) = \big(\mathrm{rank}(A), \mathrm{rank}(A^T)\big).$$

In the matrix case, we attach special names to V_1 and V_2:

- V_1 is the **column space** or **range**, and
- V_2 is the **row space**.

When $\mathcal{A} \in V_1 \otimes V_2$ lives in the minimal separable tensor subspace $V_1 \otimes V_2$, the **fundamental theorem of linear algebra** states that $\dim V_1 = \dim V_2$. Therefore,

$$\operatorname{mlrank}(A) = (\dim V_1, \dim V_2) = (r, r).$$

That is, **not all tuples are feasible multilinear ranks**! This observation generalizes to higher-order tensors.

Proposition (Carlini and Kleppe, 2011)

Let $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ 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.


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
JORGE CHAM © 2013
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
WWW.PHDCOMICS.COM

Overview

1. Introduction (5')
2. Multilinear algebra* (40')
3. Tucker decomposition (15')
4. Higher-order singular value decomposition (40')
5. Application: dimensionality reduction (5')
6. Conclusions

Higher-order singular value decomposition

The compact **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** of V_k for a tensor

$$\mathcal{A} \in V_1 \otimes \cdots \otimes V_d \subset W_1 \otimes \cdots \otimes W_d.$$

The HOSVD chooses as **orthonormal basis** for V_k the left singular vectors of $\mathcal{A}_{(k)}$. That is, let the compact SVD of $\mathcal{A}_{(k)}$ be

$$\mathcal{A}_{(k)} = U_k \Sigma_k Q_k^*.$$

Then a basis of V_k is given by $U_k \in \Bbbk^{n_k \times r_k}$.

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

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

is called an **HOSVD basis**. It reveals (a basis for) the minimal separable tensor product subspace in which $\mathcal{A}$ lives.

Since $\mathcal{A}$ lives in $V_1 \otimes \cdots \otimes V_d$ and $\mathrm{span}(U_k) = V_k$, there must exist coordinates $\mathcal{C} \in \mathbb{k}^{r_1 \times \cdots \times r_d}$

$$\mathcal{A} = (U_1 \otimes \cdots \otimes U_d)(\mathcal{C}) = (U_1, \ldots, U_d) \cdot \mathcal{C}$$

so that

$$\begin{aligned}(U_1^*, \ldots, U_d^*) \cdot \mathcal{A} &= (U_1^*, \ldots, U_d^*) \cdot ((U_1, \ldots, U_d) \cdot \mathcal{C}) \\ &= (U_1^* U_1, \ldots, U_d^* U_d) \cdot \mathcal{C} \\ &= \mathcal{C}.\end{aligned}$$

By definition of the compact SVD, we have

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

so the HOSVD reveals the multilinear rank as well.

Algorithm 1: HOSVD Algorithm

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

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

output: Coefficients array $\mathcal{C} \in \mathbb{k}^{r_1 \times r_2 \times \cdots \times r_d}$

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

  Compute the compact SVD $\mathcal{A}_{(k)} = U_k \Sigma_k Q_k^*$;

end

$\mathcal{C} \leftarrow (U_1^*, U_2^*, \ldots, U_d^*) \cdot \mathcal{A}$;

The HOSVD provides a **data sparse representation** of tensors $\mathcal{A}$ living in a separable subspace.

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

$$\underbrace{\prod_{k=1}^{d} r_k}_{\text{core tensor}} + \underbrace{\sum_{k=1}^{d} n_k r_k}_{\text{basis vectors}} \ll \prod_{k=1}^{d} n_k$$

storage (for $\mathcal{C}$ and the U_i).

Approximation algorithms (by truncation)

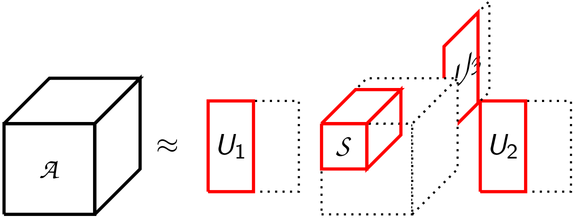
In applications, tensors $\mathcal{A}$ often (only) lie close to a separable subspace $V_1 \otimes \cdots \otimes V_d$. This leads naturally to

The low multilinear rank approximation (LMLRA) problem

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

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

Visually, we want to approximate $\mathcal{A}$ by



Since $\text{mlrank}(\mathcal{B}) \le (r_1, \ldots, r_d)$ is equivalent to the existence of a separable subspace $V_1 \otimes \cdots \otimes V_d$ in which $\mathcal{B}$ lives, we can write $\mathcal{B} = (U_1, U_2, \ldots, U_d) \cdot \mathcal{C}$ where $U_k \in \mathbb{k}^{n_k \times r_k}$ can be chosen orthonormal by the existence of the HOSVD.

After finding the subspace, the optimal approximation $\mathcal{B}$ is the **orthogonal projection** of $\mathcal{A}$ onto this subspace:

$$\mathcal{B} = \mathrm{P}_{U_1 \otimes \cdots \otimes U_d} \mathcal{A}.$$

Consequently, the problem is equivalent to

$$\min_{U_k \in \mathrm{St}_{n_k, r_k}} \|\mathcal{A} - \mathrm{P}_{U_1 \otimes \cdots \otimes U_d} \mathcal{A}\|_F$$

where $\mathrm{St}_{m,n}$ is the Stiefel manifold of $m \times n$ matrices with orthonormal columns.

Proposition (V, Vandebril, and Meerbergen, 2012)

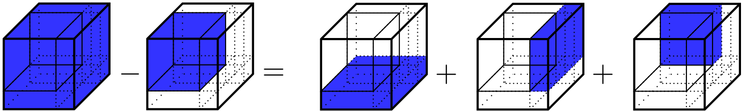
Let $U_1 \otimes \cdots \otimes U_d$ *be a tensor basis of the separable subspace* $V_1 \otimes \cdots \otimes V_d$*. Then, the approximation error*

$$\|\mathcal{A} - \mathrm{P}_{U_1 \otimes \cdots \otimes U_d} \mathcal{A}\|_F^2 = \sum_{k=1}^{d} \|\pi_{p_{k-1}} \cdots \pi_{p_1} \mathcal{A} - \pi_{p_k} \pi_{p_{k-1}} \cdots \pi_{p_1} \mathcal{A}\|_F^2 = \sum_{k=1}^{d} \|\pi_{p_k}^{\perp} \pi_{p_{k-1}} \cdots \pi_{p_1} \mathcal{A}\|_F^2,$$

where p *is any permutation of* $\{1, 2, \ldots, d\}$ *and*

$$\pi_k \mathcal{A} = (U_k U_k^*) \cdot_k \mathcal{A} \quad \text{and} \quad \pi_k^{\perp} \mathcal{A} := (I - U_k U_k^*) \cdot_k \mathcal{A}.$$

Visually, the proposition states that an error expression is



$$\|\mathcal{A} - \pi_1 \pi_2 \pi_3 \mathcal{A}\|^2 = \|\pi_1^{\perp} \mathcal{A}\|^2 + \|\pi_2^{\perp} \pi_1 \mathcal{A}\|^2 + \|\pi_3^{\perp} \pi_1 \pi_2 \mathcal{A}\|^2$$

Since orthogonal projections only decrease unitarily invariant norms, we also get the

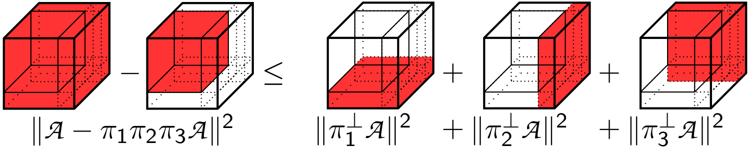
Corollary

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

$$\|\mathcal{A} - \mathrm{P}_{U_1 \otimes \cdots \otimes U_d} \mathcal{A}\|_F^2 \leq \sum_{k=1}^{d} \|\pi_k^\perp \mathcal{A}\|_F^2,$$

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

Visually, the corollary states that an upper bound is



A closed solution of the LMLRA problem

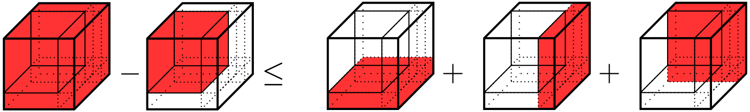
$$\min_{U_k \in \mathrm{St}_{n_k, r_k}} \|\mathcal{A} - \mathrm{P}_{U_1 \otimes \cdots \otimes U_d} \mathcal{A}\|_F$$

is not known.

Nevertheless, we can exploit the error expression and the upper bound for choosing good, even **quasi-optimal**, separable subspaces to project onto.

T-HOSVD

The idea of the **truncated HOSVD** (T-HOSVD) is minimizing the upper bound on the error:



$$\|\mathcal{A} - \pi_1\pi_2\pi_3\mathcal{A}\|^2 \le \|\pi_1^\perp \mathcal{A}\|^2 + \|\pi_2^\perp \mathcal{A}\|^2 + \|\pi_3^\perp \mathcal{A}\|^2$$

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

Minimizing the upper bound results in

$$\min_{\pi_1,\ldots,\pi_d} \|\mathcal{A} - \pi_1 \cdots \pi_d \mathcal{A}\|_F^2 \leq \min_{\pi_1,\ldots,\pi_d} \sum_{k=1}^{d} \|\pi_k^{\perp} \mathcal{A}\|_F^2$$
$$= \sum_{k=1}^{d} \min_{\pi_k} \|\pi_k^{\perp} \mathcal{A}\|_F^2$$
$$= \sum_{k=1}^{d} \min_{U_k \in \mathrm{St}_{n_k, r_k}} \|\mathcal{A}_{(k)} - U_k U_k^* \mathcal{A}_{(k)}\|_F^2$$

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 $\mathcal{A} \in \mathbb{k}^{n_1 \times n_2 \times \cdots \times n_d}$
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 $\overline{\mathcal{C}} \in \mathbb{k}^{r_1 \times r_2 \times \cdots \times r_d}$
for $k = 1, 2, \ldots, d$ **do**
    Compute the compact SVD $\mathcal{A}_{(k)} = U_k \Sigma_k Q_k^*$;
    Let $\overline{U}_k$ contain the first r_k columns of U_k;
end
$\overline{\mathcal{C}} \leftarrow (\overline{U}_1^*, \overline{U}_2^*, \ldots, \overline{U}_d^*) \cdot \mathcal{A}$;

The resulting approximation is **quasi-optimal**.

Proposition (Hackbusch, 2012)

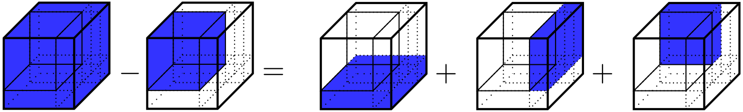
Let $\mathcal{A} \in \mathbb{k}^{n_1 \times \cdots \times n_d}$, *and let* $\mathcal{A}^*$ *be the best rank-*$(r_1, \ldots, r_d)$ *approximation to* $\mathcal{B}$, *i.e.,*

$$\|\mathcal{A} - \mathcal{A}^*\|_F = \min_{\text{mlrank}(\mathcal{B}) \le (r_1, \ldots, r_d)} \|\mathcal{A} - \mathcal{B}\|_F.$$

Then, the rank-$(r_1, \ldots, r_d)$ *T-HOSVD approximation* $\mathcal{A}_T$ *is a* quasi-best approximation*:*

$$\|\mathcal{A} - \mathcal{A}_T\|_F \le \sqrt{d}\,\|\mathcal{A} - \mathcal{A}^*\|_F.$$

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

$$
\begin{aligned}
\widehat{\pi}_1 &= \arg\min_{\pi_1} \|\pi_1^{\perp} \mathcal{A}\|^2 \\
\widehat{\pi}_2 &= \arg\min_{\pi_2} \|\pi_2^{\perp} \widehat{\pi}_1 \mathcal{A}\|^2 \\
&\vdots \\
\widehat{\pi}_d &= \arg\min_{\pi_d} \|\pi_d^{\perp} \widehat{\pi}_{d-1} \cdots \widehat{\pi}_2 \widehat{\pi}_1 \mathcal{A}\|^2
\end{aligned}
$$

In practice, $\min_{\pi_k} \|\pi_k^\perp \widehat{\pi}_{k-1} \cdots \widehat{\pi}_1 \mathcal{A}\|_F$ is computed as follows:

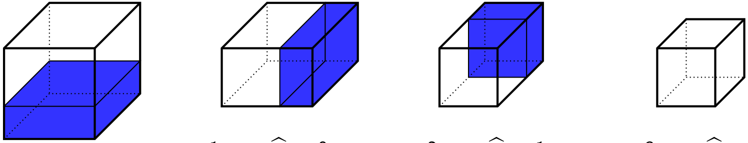
$$\begin{aligned}
&\min_{U_k \in \mathrm{St}_{n_k, r_k}} \| U_k U_k^* \mathcal{A}_{(k)} (\widehat{U}_1 \widehat{U}_1^* \otimes \cdots \otimes \widehat{U}_{k-1} \widehat{U}_{k-1}^* \otimes I \otimes \cdots \otimes I)^T \|_F \\
&\quad = \min_{U_k} \| U_k U_k^* \mathcal{A}_{(k)} (\widehat{U}_1^* \otimes \cdots \otimes \widehat{U}_{k-1}^* \otimes I \otimes \cdots \otimes I)^T \|_F \\
&\quad = \min_{U_k} \| U_k U_k^* \mathcal{C}_{(k)}^{k-1} \|_F,
\end{aligned}$$

where we define

$$\mathcal{C}^{k-1} := (\widehat{U}_1^*, \ldots, \widehat{U}_{k-1}^*, I, \ldots, I) \cdot \mathcal{A} = \widehat{U}_{k-1}^* \cdot_{k-1} \mathcal{C}^{k-2}.$$

The solution of $\min_{U_k \in \mathrm{St}_{n_k, r_k}} \| U_k U_k^* \mathcal{C}_{(k)}^{k-1} \|_F$ is given by the rank-r_k truncated SVD of $\mathcal{C}_{(k)}^{k-1}$.

Visually, here's what happens for a third-order tensor.



$\mathcal{C}^0 = \mathcal{A}$ $\qquad$ $\mathcal{C}^1_{(1)} = \widehat{U}_1^* \mathcal{C}^0_{(1)}$ $\qquad$ $\mathcal{C}^2_{(2)} = \widehat{U}_2^* \mathcal{C}^1_{(2)}$ $\qquad$ $\mathcal{C}^3_{(3)} = \widehat{U}_3^* \mathcal{C}^2_{(3)}$

The **ST-HOSVD algorithm** is thus a minor modification of the T-HOSVD algorithm.

Algorithm 3: ST-HOSVD Algorithm

input : A tensor $\mathcal{A} \in \mathbb{k}^{n_1 \times n_2 \times \cdots \times n_d}$
input : A target multilinear rank $(r_1, r_2, \ldots, r_d)$.
output: The components $(\widehat{U}_1, \widehat{U}_2, \ldots, \widehat{U}_d)$ of the ST-HOSVD basis
output: Coefficients array $\widehat{\mathcal{C}} \in \mathbb{k}^{r_1 \times r_2 \times \cdots \times r_d}$
$\widehat{\mathcal{C}} \leftarrow \widehat{\mathcal{A}}$;
for $k = 1, 2, \ldots, d$ **do**
  Compute the compact SVD $\mathcal{C}_{(k)} = U_k \Sigma_k Q_k^*$;
  Let $\widehat{U}_k$ contain the first r_k columns of U_k;
  $\widehat{\mathcal{C}} \leftarrow \widehat{U}_k^* \cdot_k \widehat{\mathcal{C}}$;
end

The resulting approximation is also **quasi-optimal**.

Proposition (Hackbusch, 2012)

Let $\mathcal{A} \in \mathbb{k}^{n_1 \times \cdots \times n_d}$, and let $\mathcal{A}^*$ be the best rank-$(r_1, \ldots, r_d)$ approximation to $\mathcal{A}$, i.e.,

$$\|\mathcal{A} - \mathcal{A}^*\|_F = \min_{\mathrm{mlrank}(\mathcal{B}) \le (r_1,\ldots,r_d)} \|\mathcal{A} - \mathcal{B}\|_F.$$

Then, the rank-$(r_1, \ldots, r_d)$ ST-HOSVD approximation $\mathcal{A}_S$ is a quasi-best approximation*:*

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

Computational performance

Assume that we truncate a tensor in $\mathbb{k}^{n\times\cdots\times n}$ to multilinear rank $(r,\ldots,r)$. The computational complexity of ST-HOSVD (with randomized truncated SVDs) is

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

which compares favorably to T-HOSVD's

$$\mathcal{O}\left(d\,rn^d\right) \text{ operations.}$$

Note that **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.


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
JORGE CHAM © 2013
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
WWW.PHDCOMICS.COM

Overview

1 Introduction (5')

2 Multilinear algebra* (40')

3 Tucker decomposition (15')

4 Higher-order singular value decomposition (40')

5 Application: dimensionality reduction (5')

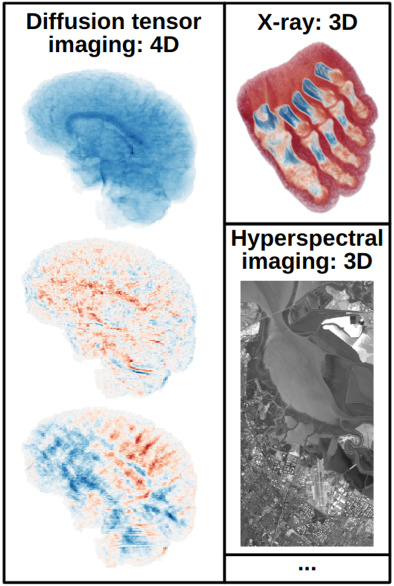
6 Conclusions

Application: dimensionality reduction

A general, main application of the truncated HOSVD consists of **dimensionality reduction**.

A truncated HOSVD identifies the minimal separable tensor subspace $V_1 \otimes \cdots \otimes V_d$ in which a tensor $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ (approximately) lives. As a **geometric principle**, geometric properties of $\mathcal{A}$ do not depend on the basis in which $\mathcal{A}$ is expressed!

Hence, most geometric analyses of $\mathcal{A}$ can be applied verbatim to the coordinate tensor $\mathcal{C}$, expressing $\mathcal{A}$ relative to $V_1 \otimes \cdots \otimes V_d$. This type of general and (usually) fast preprocessing is called **Tucker compression**.

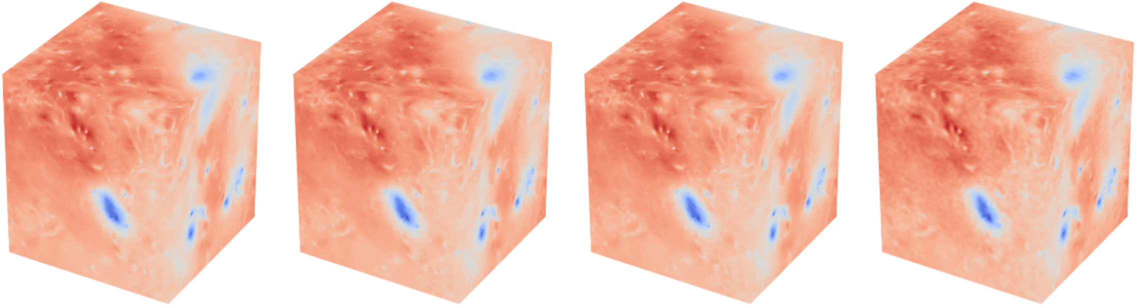


Dimensionality reduction is also a stand-alone use case. That is, **compression** of (structured) higher-order data arrays. In Baert and V (2021), we considered data from

- X-ray scans,
- diffusion tensor images,
- hyperspectral images, and
- simulation results of partial differential equations (CFD, climate, and weather).

These data sets can get large quickly!

- Isotropic-V is a 1.5GiB tensor of size $512 \times 512 \times 512 \times 3$,
- Deforest-33 is a 12.0GiB tensor of size $19 \times 79 \times 33 \times 180 \times 360$,
- Hurricane is a 24.2GiB tensor of size $13 \times 20 \times 100 \times 500 \times 500$.



Compression factor: 48.8
Relative error: 0.30%
PSNR: 69.8 dB

Compression factor: 152
Relative error: 1.00%
PSNR: 59.4 dB

Compression factor: 639
Relative error: 3.00%
PSNR: 49.8 dB

Compression factor: 4948
Relative error: 10.1%
PSNR: 39.3 dB

Fig. 2. ATC compression examples using the Isotropic-PT dataset. Each visualization only shows the first time slice of the data tensor, while the statistics in the captions represent the full data.

Uncompressed size is 800MiB.


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
JORGE CHAM © 2013
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
WWW.PHDCOMICS.COM

Overview

1 Introduction (5')

2 Multilinear algebra* (40')

3 Tucker decomposition (15')

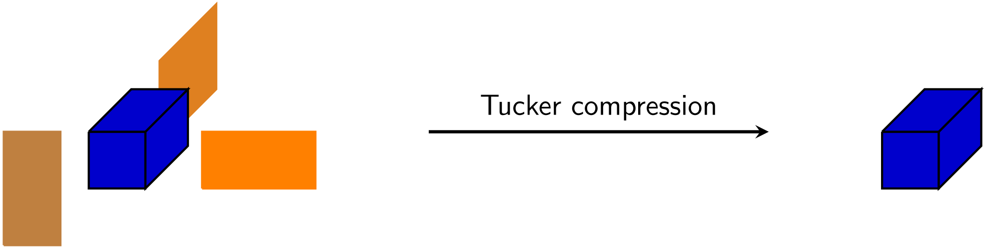
4 Higher-order singular value decomposition (40')

5 Application: dimensionality reduction (5')

6 Conclusions

Conclusions

The higher-order singular value decomposition can identify the minimal separable tensor subspace in which a given tensor lives. Most analyses can then proceed on the core tensor.



References

- Baert, Vannieuwenhoven, *ATC: an Advanced Tucker Compression library for multidimensional data*, arXiv:2107.01384, 2021.
- Carlini, Kleppe, *Ranks derived from multilinear maps*, Journal of Pure and Applied Algebra, 2011.
- De Lathauwer, De Moor, Vandewalle, *A multilinear singular value decomposition*, SIAM Journal on Matrix Analysis and Applications, 2000.
- Greub, *Multilinear Algebra*, 2nd ed., Springer, 1978.
- Kolda, Bader, *Tensor decompositions and applications*, SIAM Review, 2008.
- Landsberg, *Tensors: Geometry and Applications*, AMS, 2012.
- Hackbusch, *Tensor Spaces and Numerical Tensor Calculus*, Springer, 2012.
- Hitchcock, *Multiple invariants and generalized rank of a P-way matrix or tensor*, Journal of Mathematics and Physics, 1928.
- Tucker, *Some mathematical notes on three-mode factor analysis*, Psychometrika, 1966.
- Vannieuwenhoven, Vandebril, Meerbergen, *A new truncation strategy for the higher-order singular value decomposition*, SIAM Journal on Scientific Computing, 2012.