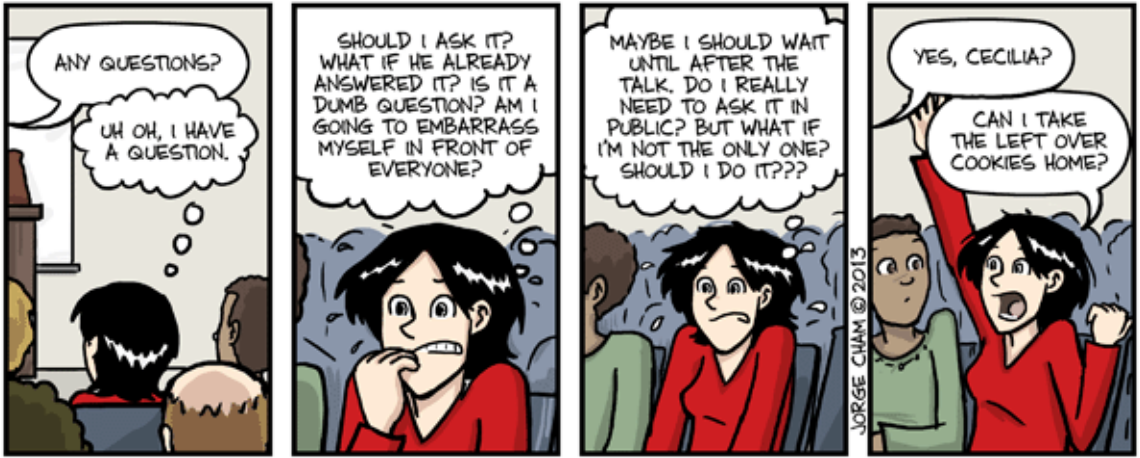
Tensor decompositions and their applications

Lecture 4: Tensor rank decomposition

Nick Vannieuwenhoven (KU Leuven)

1. Introduction (5')
2. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. 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???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Overview

1. Introduction (5')
2. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. Conclusions

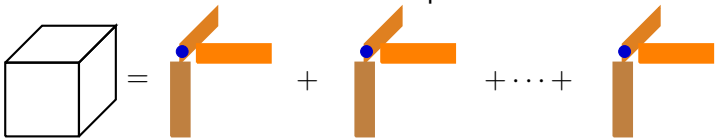
Linear algebra

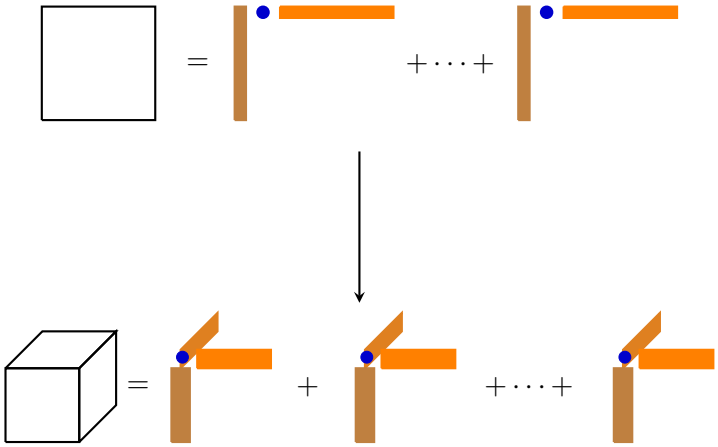
is extended to

Multilinear algebra

Singular value decomposition

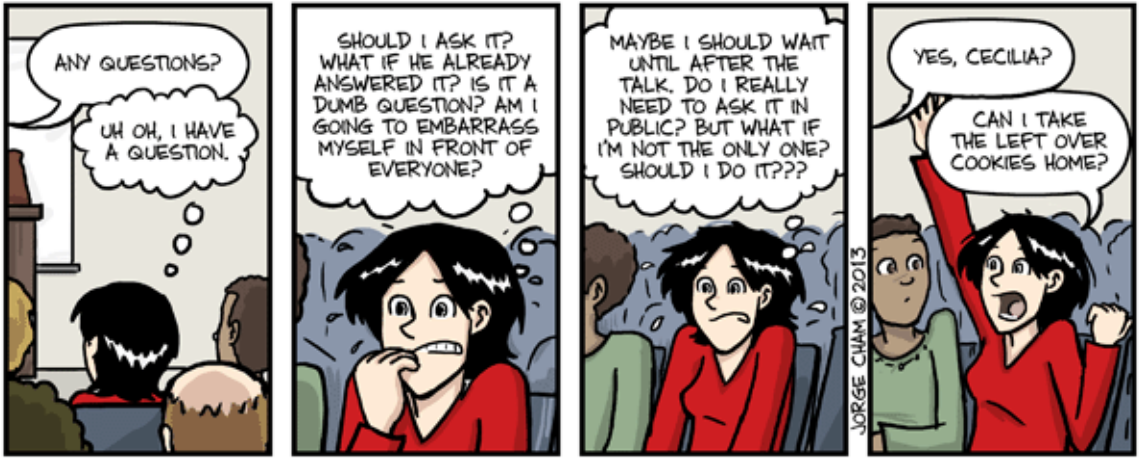
Tensor rank decomposition





One interpretation of the compact SVD of a matrix A is that it reveals a minimum-length expression of A as **a sum of rank-1 matrices**.

This interpretation generalizes immediately to tensors. The resulting **tensor rank decomposition** can be used for **exploratory data analysis**.

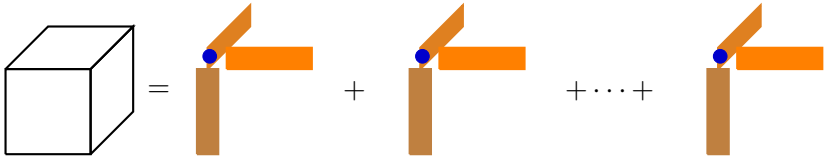

ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES,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

Overview

1. Introduction (5')
2. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. Conclusions

Tensor rank decomposition

The decomposition introduced by Hitchcock (1927) that expresses a tensor as a **minimum-length** linear combination of rank-1 tensors,



$$\mathcal{A} = a_1^1 \otimes \cdots \otimes a_1^d + \; a_2^1 \otimes \cdots \otimes a_2^d \; + \cdots + \; a_r^1 \otimes \cdots \otimes a_r^d$$

was rediscovered several times and hence goes by many names:

- tensor rank decomposition,
- parallel factor analysis (PARAFAC),
- canonical decomposition (CANDECOMP),
- CP decomposition,
- separable representation, and
- canonical polyadic decomposition (CPD).

Tensor rank

Since

$$V_1 \otimes \cdots \otimes V_d := \operatorname{span}(\{v_{i_1}^1 \otimes v_{i_2}^2 \otimes \cdots \otimes v_{i_d}^d \mid 1 \leq i_k \leq \dim V_k, k = 1, \ldots, d\})$$

where $\{v_1^k, v_2^k, \ldots, v_{n_k}^k\}$ is a basis for V_k, every tensor $\mathcal{A} \in V_1 \otimes V_2 \otimes \cdots \otimes V_d$ has an expression

$$\mathcal{A} = \sum_{i_1, i_2, \ldots, i_d} c_{i_1, i_2, \ldots, i_d} v_{i_1}^1 \otimes v_{i_2}^2 \otimes \cdots \otimes v_{i_d}^d.$$

Hence, there exist polyadic decompositions for every tensor $\mathcal{A}$.

Definition (Tensor rank)

The **rank** of $\mathcal{A}$ is the minimum number of rank-1 summands in a polyadic decomposition of $\mathcal{A}$.

If $\mathcal{A} = \sum_{i=1}^{r} a_i^1 \otimes \cdots \otimes a_i^d$ is a tensor rank decomposition and (π, σ) partitions $\{1, \ldots, d\}$, then

$$\mathcal{A}_{(\pi;\sigma)} = \sum_{i=1}^{r} (a_i^{\pi_1} \otimes \cdots \otimes a_i^{\pi_k})(a_i^{\sigma_1} \otimes \cdots \otimes a_i^{\sigma_l})^T.$$

This proves the following

Lemma

If $\mathcal{A} \in V_1 \otimes \cdots \otimes V_d$ has rank r, then for every partition (π, σ) we have

$$\operatorname{rank}(\mathcal{A}_{(\pi;\sigma)}) \leq r.$$

In particular, if $\mathcal{A}$'s multilinear rank is $(r_1, \ldots, r_d)$ and its tensor trains rank is $(s_1, \ldots, s_{d-1})$, then

$$\max\{r_1, r_2, \ldots, r_d, s_1, \ldots, s_{d-1}\} \leq r.$$

Tucker compression

A very **practical observation** is the following fact.

Lemma

Let $A_i : V_i \to W_i$ and $\mathcal{A} \in V_1 \otimes \cdots \otimes V_d$. Then,

$$\operatorname{rank}((A_1, \ldots, A_d) \cdot \mathcal{A}) \leq \operatorname{rank}(\mathcal{A})$$

When all A_i are injective, i.e., their matrices have linearly independent columns, then applying the lemma first to the A_i's and then to their left inverses $A_i^\dagger$ results in the next

Lemma

Let $A_i : V_i \to W_i$ be injective and $\mathcal{A} \in V_1 \otimes \cdots \otimes V_d$. Then,

$$\operatorname{rank}(\mathcal{A}) = \operatorname{rank}((A_1^\dagger, \ldots, A_d^\dagger) \cdot ((A_1, \ldots, A_d) \cdot \mathcal{A})) \leq \operatorname{rank}((A_1, \ldots, A_d) \cdot \mathcal{A}) \leq \operatorname{rank}(\mathcal{A})$$

In particular, if $\mathcal{A}$ has a compact higher-order singular value decomposition (HOSVD)

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

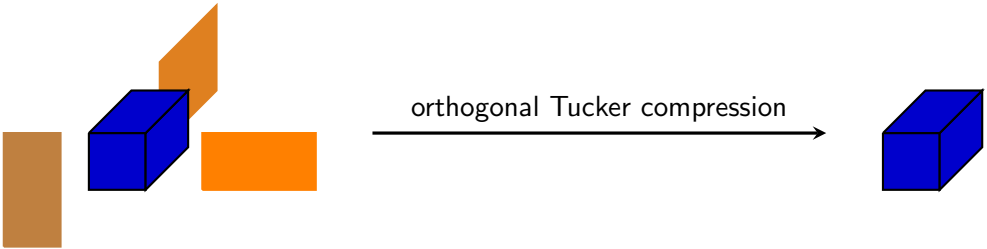
then

$$\operatorname{rank}(\mathcal{A}) = \operatorname{rank}(\mathcal{C}).$$

This is not the only useful property that is **invariant under orthogonal Tucker compression**. When U_i has orthonormal columns, we have invariance of

- **rank** (de Silva and Lim, 2008),
- **identifiability** (Chiantini, Ottaviani and Vannieuwenhoven, 2014), and
- **sensitivity** (Dewaele, Breiding and Vannieuwenhoven, 2021)

under orthogonal Tucker compression.



That is, several essential properties of the tensor rank decomposition of $\mathcal{A} = (U_1, \ldots, U_d) \cdot \mathcal{C}$ and its core tensor $\mathcal{C}$ are the same.

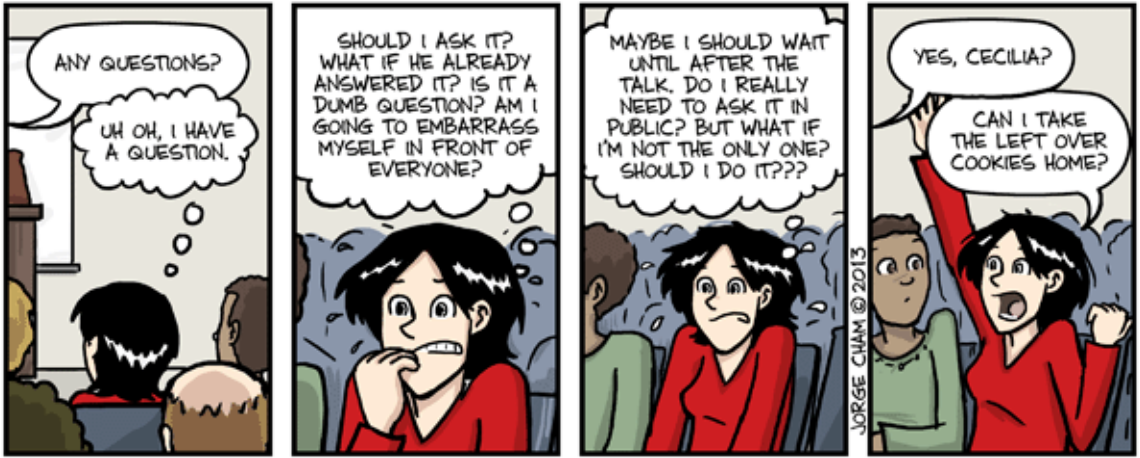
The previous observation is exploited by Bro and Andersson (1998) in the **CDE approach** to computing a tensor rank decomposition:

C: **Compress** $\mathcal{A}$ with an HOSVD $\mathcal{A} = (U_1, \ldots, U_d) \cdot \mathcal{C}$ to the core tensor $\mathcal{C}$.

D: **Decompose** the core tensor $\mathcal{C} = \sum_{i=1}^{r} c_i^1 \otimes \cdots \otimes c_i^d$.

E: **Expand** this decomposition of $\mathcal{C}$ to $\mathcal{A} = \sum_{i=1}^{r} (U_1 c_i^1) \otimes \cdots \otimes (U_d c_i^d)$.

This approach is **much faster** than computing a tensor rank decomposition of $\mathcal{A}$ directly in the large space $V_1 \otimes \cdots \otimes V_d$.


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. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. Conclusions

Identifiability

We say that $\mathcal{A} \in V_1 \otimes \cdots \otimes V_d$ is r-**identifiable** if there is a **unique set** (of cardinality r) of rank-1 tensors $\{\mathcal{A}_1, \ldots, \mathcal{A}_r\}$ such that

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

In other words, if $\mathcal{A}$ is r-identifiable and we have

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

then necessarily

$$\{\mathcal{A}_1, \ldots, \mathcal{A}_r\} = \{\mathcal{A}'_1, \ldots, \mathcal{A}'_r\}.$$

Identifiability of tensors is **radically different** from the matrix case ($d = 2$). Indeed, if $A \in \mathbb{k}^{m \times n}$ is a rank-r matrix, then

$$A = UV^T = (UX)(X^{-1}V^T) \quad \text{for all } X \in \mathrm{GL}(r, \mathbb{k})$$

For most choices of X, we have that $(UX)_i \neq \alpha \mathrm{u}_{\pi_i}$, so that the decompositions are distinct.

For example,

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 2 \end{bmatrix}$$

Kruskal's lemma

A classic result on identifiability is **Kruskal's lemma**, which relies on the notion of the **Kruskal rank** of a set of vectors.

Definition (Kruskal, 1977)

The Kruskal rank $k(\mathcal{V})$ of a set of vectors $\mathcal{V} = \{\mathrm{v}_1, \ldots, \mathrm{v}_r\} \subset V$ is the largest k integer such that every subset of k vectors of $\mathcal{V}$ is linearly independent.

For example,

- $\{0\}$ has Kruskal rank 0;
- $\{\mathrm{v}, \mathrm{v}\}$ has Kruskal rank 1;
- $\{\mathrm{v}, \mathrm{w}, \mathrm{v}\}$ has Kruskal rank 1; and
- $\{\mathrm{v}, \mathrm{w}, \mathrm{v} + \mathrm{w}\}$ has Kruskal rank 2 if v and w are linearly independent.
- a "general" set of k vectors in V has Kruskal rank $\min\{k, \dim V\}$.

Kruskal proved, among others, the following result.

Theorem (Kruskal, 1977)

Let $\mathcal{A} = \sum_{i=1}^{r} \mathrm{a}_i^1 \otimes \mathrm{a}_i^2 \otimes \mathrm{a}_i^3$ *and* $A_k := [\mathrm{a}_i^k]_{i=1}^r$. *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 $\mathcal{A}$ *has rank* r *and it is* r*-identifiable.*

The condition $k(A_1) > 1$ is necessary for $r \geq 2$ because otherwise there exist $i \neq j$ with $a_i^1 = \alpha v$ and $a_j^1 = \beta v$ such that

$$\alpha v \otimes a_i^2 \otimes a_i^3 + \beta v \otimes a_j^2 \otimes a_j^3 \in \langle \mathrm{v} \rangle \otimes V_2 \otimes V_3 \simeq V_2 \otimes V_3^*$$

is like a matrix (which are not identifiable), and likewise for the other factors.

Kruskal's lemma can also be applied to higher-order tensors

$$\mathcal{A} \in V_1 \otimes \cdots \otimes V_d$$

by **grouping the factors** or **reshaping the tensor**:

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

where $1 \leq s < t \leq d$ are fixed and π is a permutation of $\{1, \ldots, d\}$.

Since the Kruskal rank of vectors in V is bounded above by $\dim V$, the **reshaped Kruskal lemma** suggests that r-identifiability can hold up to

$$2r \leq \left(\prod_{k=1}^{s} \dim V_{\pi_k}\right) + \left(\prod_{k=s+1}^{t} \dim V_{\pi_k}\right) + \left(\prod_{k=t+1}^{d} \dim V_{\pi_k}\right) - 2$$

For example, if all $\dim V_k = n$ and n is sufficiently large, this yields

$$r \leq n^{\lfloor \frac{d-1}{2} \rfloor} + \frac{1}{2} n^{d-2\lfloor \frac{d-1}{2} \rfloor} - 1$$

in the best case (Chiantini, Ottaviani, Vannnieuwenhoven, 2017).

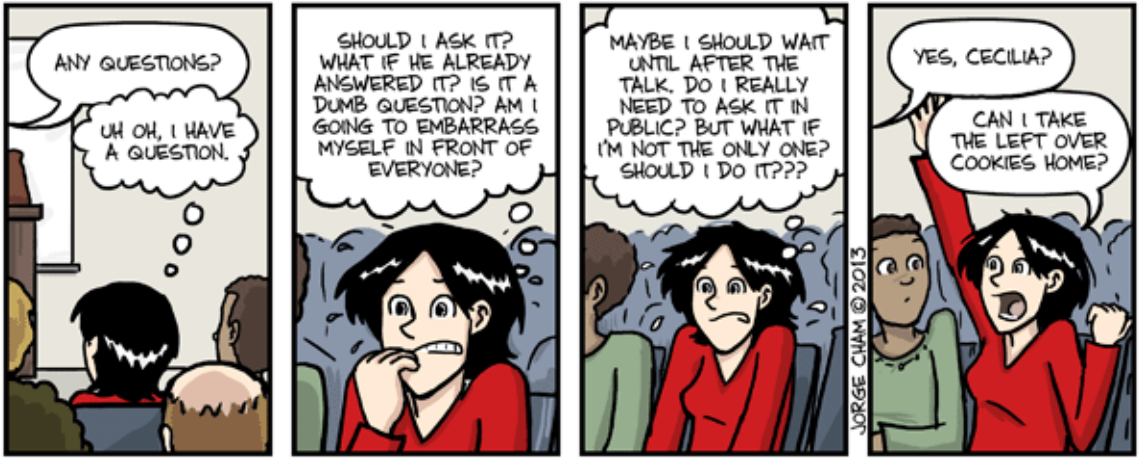
Generic identifiability conjectures

There is a **conjecture** that rank-r higher-order tensors in $V_1 \otimes \cdots \otimes V_d$ are mostly r-identifiable. Specifically, Bocci, Chiantini and Ottaviani (2014) and Chiantini, Ottaviani and Vannieuwenhoven (2014, 2017) conjecture the following. Let $n_k = \dim V_k$. If

$$n_1 \geq \cdots \geq n_d \geq 2, \quad r_{\text{cr}} = \frac{n_1 \cdots n_d}{1 + \sum_{k=1}^{d}(n_k - 1)}, \quad \text{and} \quad r_{\text{ub}} = n_2 \cdots n_d - \sum_{k=2}^{d}(n_k - 1),$$

then the **general rule**, modulo a few exceptions, is:

- matrices ($d = 2$) $\rightarrow$ nowhere r-identifiable
- rank too high ($r \geq r_{\text{cr}}$) $\rightarrow$ nowhere r-identifiable
- unbalanced and high rank ($n_1 > r_{\text{ub}}$ and $r \geq r_{\text{ub}}$) $\rightarrow$ nowhere r-identifiable
- $(n, n, 2, 2)$ tensors with $r = 2n - 1$ $\rightarrow$ nowhere r-identifiable
- 4 special exceptions $\rightarrow$ nowhere r-identifiable
- "regular" tensors of subgeneric rank $\rightarrow$ r-identifiable almost everywhere


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. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. Conclusions

Sensitivity

Because of identifiability, the tensor rank decomposition can be employed for **data analysis** purposes, by **identifying dominant features** or explanatory factors.

However, tensors arising in applications can be **corrupted** by

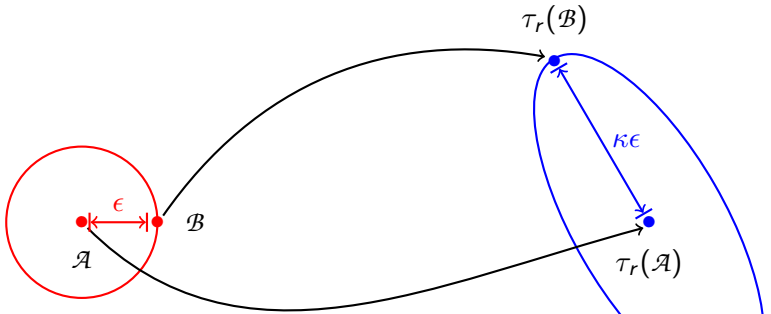
- measurement errors,
- roundoff errors,
- computation and simulation errors.

Consequently, there will be **slight differences** between the **true mathematical tensor** and the **perturbed tensor** that you want to analyze automatically with a computer.

And so a fundamental question arises:

> *How can we be* mathematically certain *that the computed decomposition is still close to the true rank decomposition?*

Since tensors are almost always r-identifiable, there exists some **tensor decomposition function** τ_r taking a rank-r tensor $\mathcal{A}$ to its decomposition.



$$\kappa[\tau_r](\mathcal{A}) := \lim_{\epsilon \to 0} \sup_{\mathcal{B} \in B_\epsilon(\mathcal{A})} \frac{\|\tau_r(\mathcal{B}) - \tau_r(\mathcal{A})\|}{\|\mathcal{B} - \mathcal{A}\|}.$$

The **condition number** quantifies the **worst-case sensitivity** of τ_r to input perturbations.

The condition number implies the asymptotically sharp **error bound**:

$$\|\tau_r(\mathcal{A}) - \tau_r(\mathcal{B})\| \leq \kappa[\tau_r](\mathcal{A}) \cdot \|\mathcal{A} - \mathcal{B}\| + o(\|\mathcal{A} - \mathcal{B}\|)$$

for small $\|\mathcal{A} - \mathcal{B}\|$.

A numerical experiment ...

Consider the matrix

$$A = \frac{1}{177147}\begin{bmatrix} 88574 & 88574 & 2 \\ 88574 & 88574 & 2 \\ 2 & 2 & 177146 \end{bmatrix}$$

Computing the eigenvalue decomposition $\widehat{V}\widehat{\Lambda}\widehat{V}^{-1}$ of A numerically using Octave, we find $\|A - \widehat{V}\widehat{\Lambda}\widehat{V}^{-1}\|_2 \approx 1.1 \cdot 10^{-15}$.

The eigenvalues are

numerical	exact
0.0000000000000000011..	0
0.9999830649121912	$0.999983064912191569... = 1 - 3^{-10}$
1.000016935087810	$1.000016935087808430... = 1 + 3^{-10}$

We found 15 correct digits of the exact solution.

However, when comparing the computed eigenvector corresponding to $\lambda_1 = 1 + 3^{-10}$ to the exact solution, we get

numerical	exact
0.577350269187273	$\frac{1}{\sqrt{3}}$
0.577350269187273	$\frac{1}{\sqrt{3}}$
0.577350269194331	$\frac{1}{\sqrt{3}}$

We recovered only 11 digits correctly, even though the matrix $\widehat{V}\widehat{\Lambda}\widehat{V}^{-1}$ contains at least 15 correct digits of each entry.

It seems that the eigenvalues are **computed more accurately** than the eigenvectors. Clearly not all functions are created equal!

Assuming the eigenvalues are distinct, our computational problems can be **modeled locally as analytic functions** by Kato (1995):

$$\lambda_1 : \mathrm{Sym}(\mathbb{R}^{m\times m}) \to \mathbb{R}, \quad \text{resp.} \quad v_1 : \mathrm{Sym}(\mathbb{R}^{m\times m}) \to \mathbb{S}^{m-1}.$$

What we observed above is that

$$0.41 \approx \frac{|\lambda_1(A) - \lambda_1(A+\Delta)|}{\|\Delta\|_2} \ll \frac{\|v_1(A) - v_1(A+\Delta)\|}{\|\Delta\|_2} \approx 5.33 \cdot 10^3$$

where $\|\Delta\|_2 \approx 1.1 \cdot 10^{-15}$ in this case.

For eigenvalues and eigenvectors, in the foregoing distances, it is known from e.g. Bürgisser and Cucker (2013) that the condition numbers are respectively

$$\kappa[\lambda_1](A) = 1,$$
$$\kappa[v_1](A) = \frac{1}{\min_{j \neq 1} |\lambda_1(A) - \lambda_j(A)|} = \frac{1}{2 \cdot 3^{-10}} \approx 2.95 \cdot 10^4$$

This largely explains the difference in accuracy between eigenvalues and eigenvectors in the foregoing numerical experiment.

Condition number of tensor rank decomposition

For the problem of computing a tensor rank decompostion of a rank-r tensor $\mathcal{A} = \mathcal{A}_1 + \cdots + \mathcal{A}_r$ the condition number was computed by Breiding and Vannnieuwenhoven (2018).

Let $\mathcal{A}_i = a_i^1 \otimes \cdots \otimes a_i^d$. Then the **Terracini matrix** of $(\mathcal{A}_1, \ldots, \mathcal{A}_r)$ is

$$T_{\mathcal{A}_1,\ldots,\mathcal{A}_r} = \begin{bmatrix} T_{\mathcal{A}_1} & T_{\mathcal{A}_2} & \cdots & T_{\mathcal{A}_r} \end{bmatrix}$$

where

$$T_{\mathcal{A}_i} = \begin{bmatrix} a_i^1 \otimes \cdots \otimes a_i^d & U_1 \otimes a_i^2 \otimes \cdots \otimes a_i^d & \cdots & a_i^1 \otimes \cdots \otimes a_i^{d-1} \otimes U_d \end{bmatrix}$$

and the columns of U_k contain an orthonormal basis of the vector space $V_k / \langle a_i^k \rangle$.

Theorem (Breiding and Vannieuwenhoven, 2018)

The condition number of τ_r at an r-identifiable tensor $\mathcal{A} = \mathcal{A}_1 + \cdots + \mathcal{A}_r$ is

$$\kappa[\tau_r](\mathcal{A}) = \frac{1}{\sigma_m(T_{\mathcal{A}_1,\ldots,\mathcal{A}_r})}, \qquad \text{where } m = r\left(1 + \sum_{k=1}^{d} (\dim V_k - 1)\right).$$

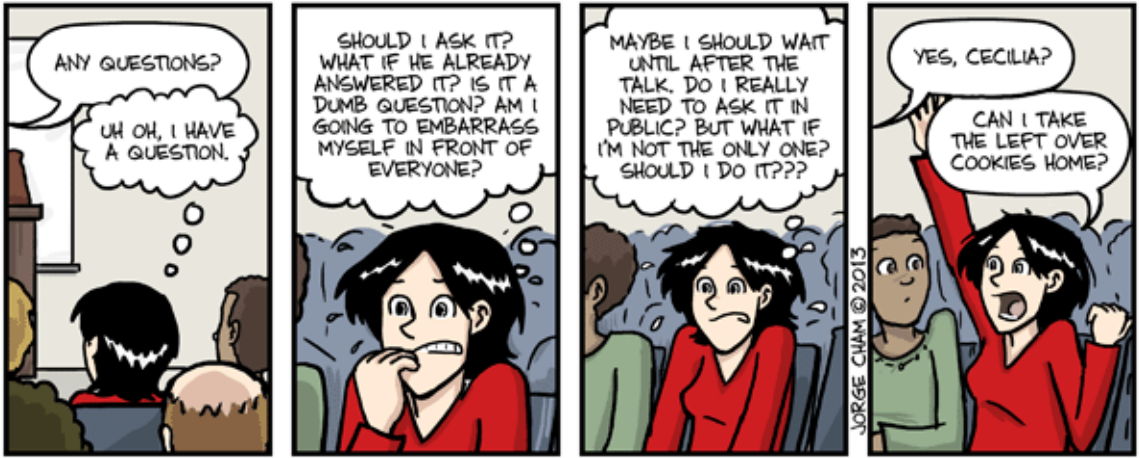
If

$$\mathcal{A} = \mathcal{A}_1 + \cdots + \mathcal{A}_r = \sum_{i=1}^{r} \mathsf{a}_i^1 \otimes \cdots \otimes \mathsf{a}_i^d$$

$$\mathcal{B} = \mathcal{B}_1 + \cdots + \mathcal{B}_r = \sum_{i=1}^{r} \mathsf{b}_i^1 \otimes \cdots \otimes \mathsf{b}_i^d$$

are r-identifiable tensors, then for $\|\mathcal{A} - \mathcal{B}\|_F \approx 0$ we have the **asymptotically sharp bound**

$$\underbrace{\min_{\pi \in \mathfrak{S}_r} \sqrt{\sum_{i=1}^{r} \|\mathcal{A}_i - \mathcal{B}_{\pi_i}\|_F^2}}_{\text{forward error}} \lesssim \underbrace{\kappa[\tau_r](\mathcal{A})}_{\text{condition number}} \cdot \underbrace{\|\mathcal{A} - \mathcal{B}\|_F}_{\text{backward error}}$$


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. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. Conclusions

Jennrich-type algorithms

An early algorithm for computing an exact tensor rank decomposition of some order-3 rank-r tensors $\mathcal{A}$ is due to Jennrich and was described by Harshman (1970).

By using the **reshaping trick**, this algorithm extends to r-identifiable higher-order tensors as well (Chiantini, Ottaviani and Vannieuwenhoven, 2017). Indeed, if

$$\mathcal{A}_i = \mathcal{A}_i^1 \otimes \mathcal{A}_i^2 \otimes \mathcal{A}_i^3 \in (V_{\pi_1} \otimes \cdots \otimes V_{\pi_s}) \otimes (V_{\pi_{s+1}} \otimes \cdots \otimes V_{\pi_t}) \otimes (V_{\pi_{t+1}} \otimes \cdots \otimes V_{\pi_d})$$

are the rank-1 terms in the decomposition of the reshaped tensor, then by r-identifiability the $\mathcal{A}_i^k$ will be rank-1 tensors themselves. Their decomposition can be obtained from a compact HOSVD.

Assume a tensor is given in coordinates as the 3-array $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$. Without loss of generality, we assume that $n_1 \geq n_2 \geq n_3$. Let

$$\mathcal{A} = \sum_{i=1}^{r} \mathsf{a}_i \otimes \mathsf{b}_i \otimes \mathsf{c}_i.$$

The matrices $A = [\mathsf{a}_i]$, $B = [\mathsf{b}_i]$ and $C = [\mathsf{c}_i]$ of $\mathcal{A}$ are called **factor matrices**.

A **fundamental assumption** in Jennrich's algorithm is that

$$\operatorname{rank}(A) = \operatorname{rank}(B) = r.$$

The assumption implies that the compact HOSVD of $\mathcal{A}$ yields a core tensor $\mathcal{C} \in \Bbbk^{r \times r \times r'}$ for some $r' \leq \min\{r, n_3\}$. Hence, we can assume w.l.o.g. that $\mathcal{A} \in \Bbbk^{r \times r \times r'}$ with $r' \geq 2$ (otherwise r-identifiability fails).

Consider two vectors q_1 and q_2 in $\mathbb{k}^{r'}$. Then,

$$X := (I, I, q_1^*) \cdot \mathcal{A} = \sum_{i=1}^{r} \mathsf{a}_i \otimes \mathsf{b}_i \otimes (q_1^* \mathsf{c}_i), \text{ and}$$
$$Y := (I, I, q_2^*) \cdot \mathcal{A} = \sum_{i=1}^{r} \mathsf{a}_i \otimes \mathsf{b}_i \otimes (q_2^* \mathsf{c}_i),$$

Observe that both X and Y are matrices, with respective decompositions

$$X = A \underbrace{\operatorname{diag}(q_1^* c_1, \ldots, q_1^* c_r)}_{D_X} B^T \text{ and } Y = A \underbrace{\operatorname{diag}(q_2^* c_1, \ldots, q_2^* c_r)}_{D_Y} B^T.$$

By the fundamental assumption in Jennrich's algorithm, $A, B \in \mathbb{k}^{r \times r}$ are invertible matrices. Assume additionally that $D_X, D_Y \in \mathbb{k}^{r \times r}$ are invertible. Then,

$$\begin{aligned} XY^{-1} &= (AD_X B^T)(AD_Y B^T)^{-1} \\ &= (AD_X B^T)(B^{-T} D_Y^{-1} A^{-1}) \\ &= A(D_X D_Y^{-1}) A^{-1} \end{aligned}$$

In other words, XY^{-1} has an eigenvalue decomposition with eigenvalues on the diagonal of $D_X D_Y^{-1}$ and eigenvectors A.

Next, consider the 1-flattening of $\mathcal{A}$:

$$\mathcal{A}_{(1)} = \sum_{i=1}^{r} \mathrm{a}_i (\mathrm{b}_i \otimes \mathrm{c}_i)^T = A(B \odot C)^T,$$

where $B \odot C := [\mathrm{b}_i \otimes \mathrm{c}_i]_{i=1}^{r} \in \mathbb{R}^{rr' \times r}$.

Since A is invertible, we get

$$A^{-1}\mathcal{A}_{(1)} = A^{-1}A(B \odot C)^T = (B \odot C)^T.$$

By viewing the columns of $B \odot C$ at tensors in $\mathbb{k}^r \otimes \mathbb{k}^{r'}$, we see that

$$b_i \otimes c_i \simeq b_i c_i^T$$

so b_i and c_i can be obtained from a rank-1 matrix decomposition (compact SVD) of the ith column of $B \odot C$, viewed as an $r \times r'$ matrix.

The foregoing discussion can be formulated as **Jennrich's algorithm**.

Algorithm 1: Standard PBA Algorithm

input : A tensor $\mathcal{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, $n_1 \geq n_2 \geq r$, of rank r.

output: Factor matrices (A, B, C) such that the rank-1 tensors in the CPD of $\mathcal{A}$ are $\mathcal{A}_i = \mathrm{a}_i \otimes \mathrm{b}_i \otimes \mathrm{c}_i$.

Compute a compact HOSVD $\mathcal{A} = (U_1, U_2, U_3) \cdot \mathcal{C}$;
Sample a random $r' \times 2$ matrix $Q = [q_1 \;\; q_2]$ with orthonormal columns;
$X \leftarrow (I, I, q_1^*) \cdot \mathcal{C}$;
$Y \leftarrow (I, I, q_2^*) \cdot \mathcal{C}$;
Compute eigendecomposition $XY^{-1} = ADA^{-1}$;
$B \odot C \leftarrow (A^{-1}\mathcal{C}_{(1)})^T$;
$A \leftarrow U_1 A$;
for *For each column z_i of $B \odot C$* **do**
    Let Z_i be the matrix such that $\mathrm{vec}(Z_i) = z_i$;
    Compute rank-1 truncated SVD $Z_i = \sigma u v^T$;
    $b_i \leftarrow \sigma U_2 u$;
    $c_i \leftarrow U_3 v$;
end

Numerical issues

A variant of this algorithm is implemented in Tensorlab v3.0 as `cpd_gevd`. Let us perform an experiment with it.

We create the first tensor that comes to mind: a rank-25 random tensor $\mathcal{A}$ of size $25 \times 25 \times 25$:

```
>> Ut{1} = randn(25,25);
>> Ut{2} = randn(25,25);
>> Ut{3} = randn(25,25);
>> A = cpdgen(Ut);
```

Compute $\mathcal{A}$'s decomposition and compare its distance to the input decomposition, relative to the machine precision $\epsilon \approx 2 \cdot 10^{-16}$:

```
>> Ur = cpd_gevd(A, 25);
>> E = kr(Ut) - kr(Ur);
>> norm( E(:), 2 ) / eps
ans =
     8.6249e+04
```

This large number can arise because of a high condition number. However,

```
>> kappa = condition_number( Ut )
ans =
     2.134
```

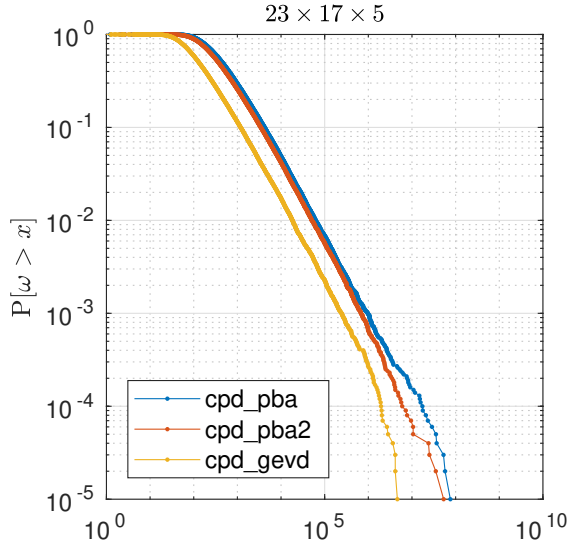
It thus appears that **there is something wrong with the algorithm** when running in floating-point arithmetic, even though it is a mathematically sound algorithm for computing low-rank CPDs. The reason is the following.

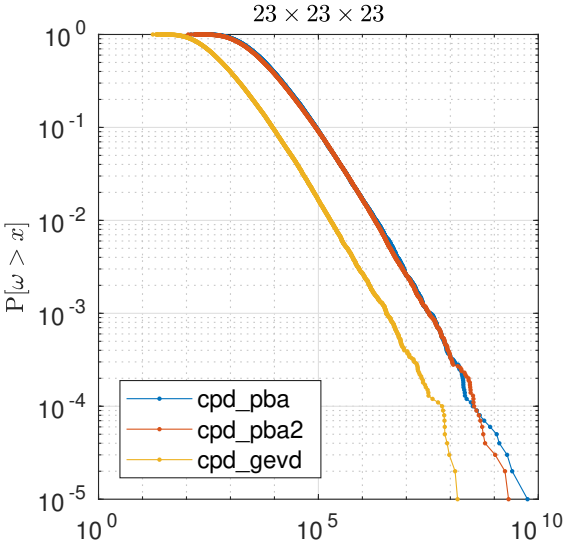
Theorem (Beltrán, Breiding and Vannieuwenhoven, 2019)

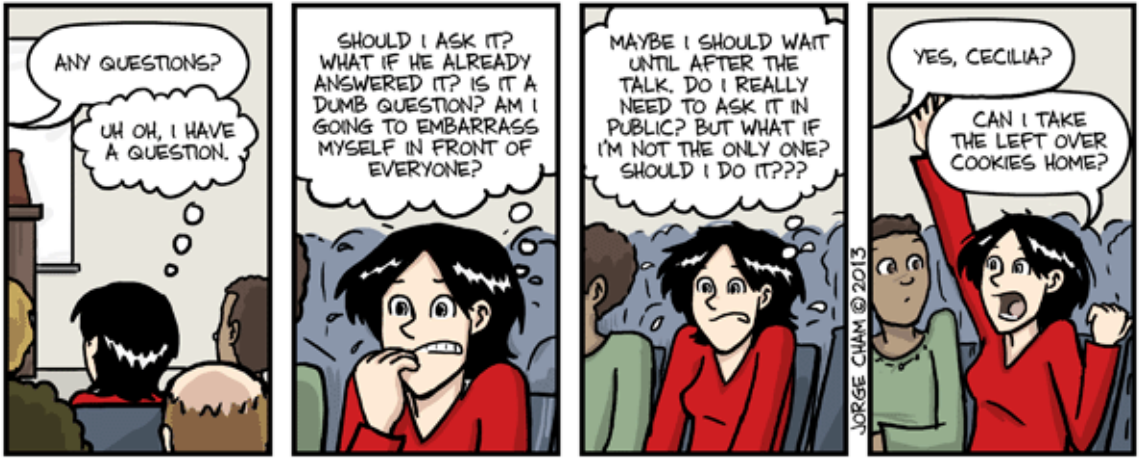
Many tensor rank decomposition algorithms based on a reduction to $\mathbb{R}^{n_1 \times n_2 \times 2}$ *are* numerically unstable*: the forward error produced by the algorithm divided by the backward error is "much" larger than the condition number on an open set of inputs.*

Aforementioned so-called pencil-based algorithms (including Jennrich's algorithm) **should be used with care** as they do not necessarily yield the highest **attainable precision**.

The instability of the algorithm leads to an **excess factor** ω on top of the condition number of the computational problem:






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. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. Conclusions

Application: reduced order models of expensive functions

Because each rank-1 summand in a tensor rank decomposition of a tensor is usually unique and always easy to interpret, the tensor rank decomposition is often used as a tool for **explorative data analysis** or a **reduced order model** of (potentially incomplete) multidimensional data.

- **factor analysis** for analyzing underlying causes in psychology (Carroll and Chang, 1970; Harshman, 1970);
- identifying spectra of fluorophores in **fluorenscence spectroscopy** (Appellof and Davidson, 1981);
- **estimation of the parameters of some latent variable models** like single exchangeable topic models, mixtures of Gaussians, and hidden Markov models (Anandkumar et al., 2014);
- tensor-based **collaborative filtering and recommender systems** (Frolov and Oseledets, 2017);
- exploratory analysis in **sports analytics applications** (Verstraete, Decroos, Coussement, Vannieuwenhoven and Davis, 2020; Geens, 2020);
- reduced order model for **thermodynamic quantities of multicomponent alloys** (Coutinho, Vervliet, De Lathauwer and Moelans, 2020);
- reduced order model for **power output of curved solar panels** (De Coppel, 2021);
- reduced order model for finite element matrices of **parameterized joint elements in lattices** (De Weer, Vannieuwenhoven, Lammens, Meerbergen, 2021).

Reduced order model

Many problems in engineering and science consist of evaluating a **complex multivariate function** that describes some aspect of reality and which can only be understood through **proxies**. Such functions are expensive to evaluate, because they either require

- **costly physical measurements** or
- **expensive numerical computations** like solving (stochastic) partial differential equations.

These functions typically feature in design, process, shape or topology optimization and are to be evaluated often.

Reduced order models attempt to approximate these complicated functions by simpler, explicit models that are **computationally inexpensive** to evaluate on some domain.

Tensor decompositions can be used to construct reduced order models. In particular, the functional version of the tensor rank decomposition provides a **separable approximation**

$$f(x_1, \ldots, x_d) = \sum_{i=1}^{r} f_i^1(x_1) f_i^2(x_2) \cdots f_i^d(x_d) \cdot \mathcal{A}_i$$

where $f : \mathbb{k}^d \to \mathbb{k}^{m_1 \times \cdots \times m_p}$, the f_i^k are univariate functions, and $\mathcal{A}_i \in \mathbb{k}^{m_1 \times \cdots \times m_p}$ are tensors (they could be structured, e.g., rank-1).

For constructing the approximation, there are two main discretization approaches:

1. construct a (sparsely sampled) tensor product grid, or
2. choose a finite-dimensional function space.

The first approach has the advantage that **optimally shaped functions** will be detected automatically through the tensor decomposition. Its disadvantage is that only sampling on a tensor product grid is possible, which can be incompatible with known function evaluations.

The second approach has the advantage that the training samples do **not need to lie on a tensor product grid**. Its disadvantage is that choosing the function space optimally may require some hand tuning.

I will discuss only option 1, discretizing the domain.

The domain is implicitly discretized with a **tensor product grid**. The number and type of nodes (Chebychev nodes, unfiformly spaced, etc) can be chosen for each variable separately.

A functional tensor rank decomposition can be obtained as follows.

1. **Sample N nodes** of the tensor product grid at random (without replacement).
2. Evaluate the function at these nodes, resulting in an **incomplete tensor** $\mathcal{A}$.
3. Compute an approximate tensor rank decomposition of $\mathcal{A}$ using **optimization algorithms**.
4. **Norm-balance the vectors** a_i^k in the rank-1 tensors $\mathcal{A}_i = a_i^1 \otimes \cdots \otimes a_i^d$. The vectors a_i^k represent evaluations of some unknown function in the nodes of the tensor grid.
5. Interpolate or approximate these evaluations using any **univariate interpolation** / least-squares approximation technique.

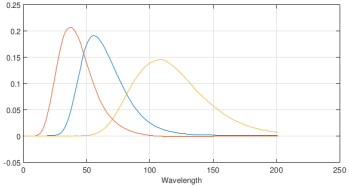
Steps 1–3 are an example of **tensor completion**, where an unknown tensor is fitted with a low-rank model for predicting the missing entries.

For a potentially **unique recovery** of a rank-r tensor from only N measurements on an $n_1 \times \cdots \times n_d$ tensor product grid, there should be at least

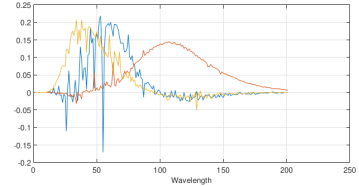
$$N \geq N_{\min} = r \left(1 + \sum_{k=1}^{d} (n_k - 1) \right),$$

measurements for this **algebraic compressed sensing problem** (Breiding, Gesmundo, Michałek, Vannieuwenhoven, 2021).

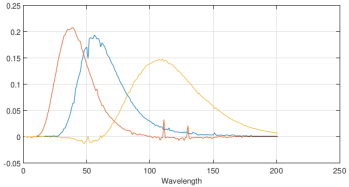
Oversampling ($N = \phi N_{\text{min}}$ with $\phi > 1$) is usually required due to the nonoptimality of random sampling, as in this experiment from Swijsen, Van der Veken, Vannieuwenhoven (2021):



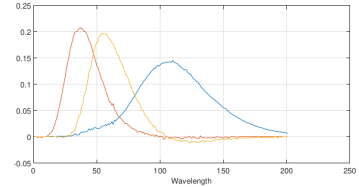
(A) Full tensor, relative error of 0.0251



(B) $\phi = 5$, relative error of 0.1342



(C) $\phi = 7$, relative error of 0.0350



(D) $\phi = 10$, relative error of 0.0296

This problem can be mitigated by adding a **penalty term** to punish high **finite differences** (Yokota, Zhao, Cichocki, 2016).

De Weer, Vannieuwenhoven, Lammens and Meerbergen (2021) used the foregoing setup for constructing a numerical model of a new type of **parameterized finite element**, to be used in the **finite element method** for solving partial differential equations:

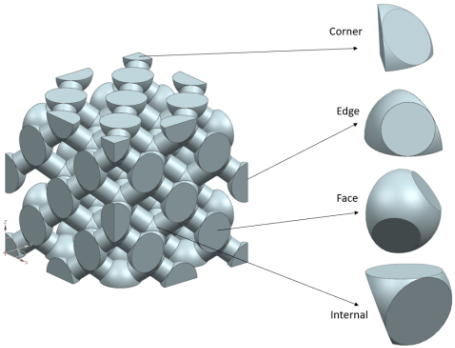


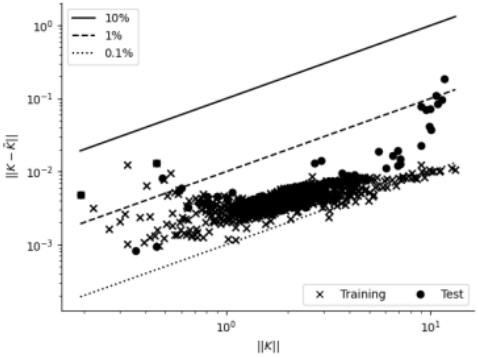
Fig. 4: Different types of lattice joints inside a $2 \times 2 \times 2$ diamond lattice. The joint radii are relatively large wrt the beam radii to highlight the joint geometries. Visualization with Simcenter 3D [29].

The small 24×24 element matrix of these joints can be computed essentially through a numerical simulation that involves solving a system in a few thousands of unknowns (depending on the desired accuracy).

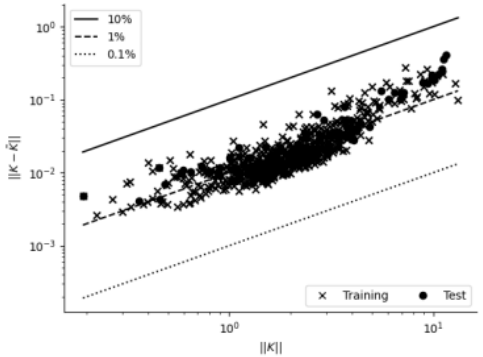
Given the large number of configurations of the joints, this direct approach is infeasible.

Instead we constructed a reduced order model for this (unknown) function.

A good approximation was possible with a low-rank tensor rank decomposition (10 or 34, depending on the setup):

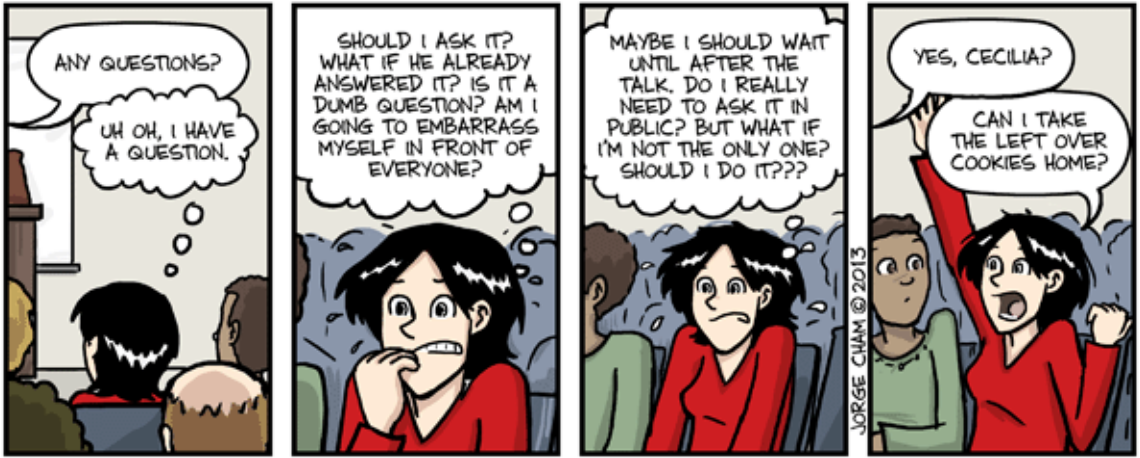


(c) Entry-level approach, combined error.



(a) Matrix-level approach, combined error.

Net gains in time are realized as soon as more than 1000 joint configurations are used. This bound is easily reached for example in topology optimization.


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. The tensor rank decomposition (15')
3. Identifiability (15')
4. Sensitivity (20')
5. Jennrich-type algorithms* (25')
6. Application: reduced order models of expensive functions (25')
7. Conclusions

Conclusions

The tensor rank decomposition is a simple decomposition that enables **exploratory and interpretative data analysis** because of its identifiability properties.

References

- Anandkumar, Ge, Hsu, Kakade, Telgarsky, *Tensor decompositions for learning latent variable models*, Journal of Machine Learning Research, 2014.
- Appellof, Davidson, *Strategies for analyzing data from video fluorometric monitoring of liquid chromatographic effluents*, Analytical Chemistry, 1981.
- Beltrán, Breiding, Vannieuwenhoven, *Pencil-based algorithms for tensor rank decomposition are not stable*, SIAM Journal on Matrix Analysis and Applications, 2019.
- Breiding, Gesmundo, Michałek, Vannieuwenhoven, *Algebraic compressed sensing*, arXiv:2108.13208, 2021.
- Breiding, Vannnieuwenhoven, *The condition number of join decompositions*, SIAM Journal on Matrix Analysis and Applications, 2018.
- Bocci, Chiantini, Ottaviani, *Refined methods for the identifiability of tensors*, Annali di Matematica Pura ed Applicata (1923-), 2014.
- Bro, Andersson, *Improving the speed of multiway algorithms part II: Compression*, Chemometrics and Intelligent Laboratory Systems, 1998.
- Bürgisser, Cucker, *Condition: The Geometry of Numerical Algorithms*, Springer, 2013.
- Carroll, Chang, *Carroll, Chang - Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart-Young decomposition*, Psychometrika, 1970.

- Chiantini, Ottaviani, Vannieuwenhoven, *An algorithm for generic and low-rank specific identifiability for complex tensors*, SIAM Journal on Matrix Analysis and Applications, 2014.
- Chiantini, Ottaviani, Vannieuwenhoven, *Effective criteria for identifiability of tensors and forms*, SIAM Journal on Matrix Analysis and Applications, 2017.
- Coutinho, Vervliet, De Lathauwer, Moelans, *Combining thermodynamics with tensor completion techniques to enable multicomponent microstructure prediction*, npj Computational Materials, 2020.
- De Coppel, Vannieuwenhoven (sup.), *Modellering van het inkomend vermogen van een gekromd en gekanteld zonnepaneel*, Master's thesis, KU Leuven, 2021.
- de Silva, Lim, *Tensor rank and the ill-posedness of the best low-rank approximation problem*, SIAM Journal on Matrix Analysis and Applications, 2008.
- De Weer, Vannieuwenhoven, Lammens, Meerbergen, *The Parametrized Superelement approach for simulating lattice structures produced by additive manufacturing*, (submitted), 2021.
- Dewaele, Breiding, Vannieuwenhoven, *The condition number of many tensor decompositions is invariant under Tucker compression*, arXiv:2106.13034, 2021.
- Frolov, Oseledets, *Tensor methods and recommender systems*, WIREs Data Mining and Knowledge Discovery, 2017.

- Geens, Davis (sup.), Vannieuwenhoven (sup.), *Het karakteriseren van voetbalspelers met tensor-gebaseerde methoden*, Master's thesis, KU Leuven, 2020.
- Harshman, *Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis*, UCLA Working Papers in Phonetics, 1970
- Kato, *Perturbation Theory for Linear Operators*, Springer, 1995.
- Kruskal, *Three-way arrays: rank and uniqueness of trilinear decompositions, with application to arithmetic complexity and statistics*, Linear Algebra and its Applications, 1977.
- Rice, *A theory of condition*, SIAM Journal on Numerical Analysis, 1966.
- Swijsen, Van der Veken, Vannieuwenhoven, *Tensor completion using geodesics on Segre manifolds*, arXiv:2108.00735, 2021.
- Yokota, Zhao, Cichocki, *Smooth PARAFAC decomposition for tensor completion*, IEEE Transactions on Signal Processing, 2016.
- Verstraete, Decroos, Coussement, Vannieuwenhoven, Davis, *Analyzing soccer players' skill ratings over time using tensor-based methods*, Machine Learning and Knowledge Discovery in Databases, 2020.
- Vervliet, Debals, Sorber, Van Barel, De Lathauwer, *Tensorlab 3.0*, Available online at `http://www.tensorlab.net`, 2016.