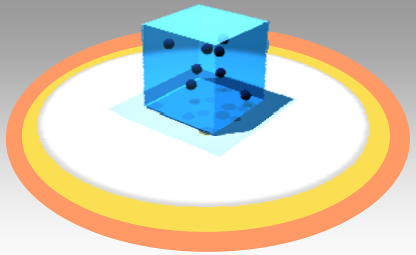
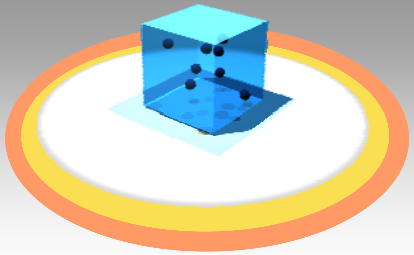
SAND2007-7046C



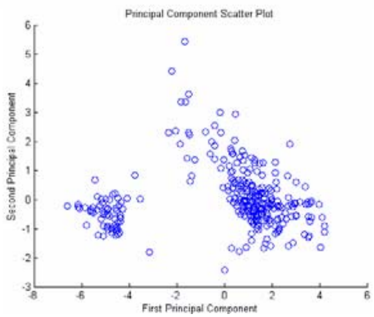
Tensor Decompositions and Data Mining

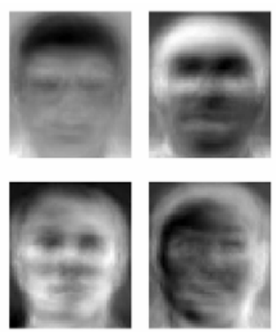
Tammy Kolda
Sandia National Laboratories
tgkolda@sandia.gov
http://csmr.ca.sandia.gov/~tgkolda/

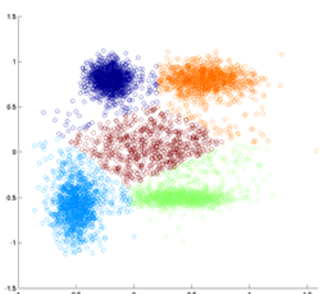
Sandia National Laboratories



Matrix decompositions are important to data mining







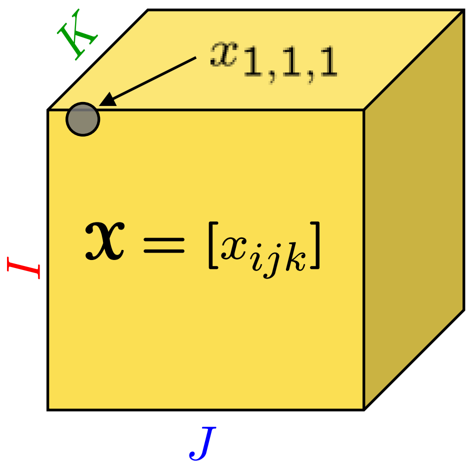
- **Principal Component Analysis (PCA)** – singular value decomposition (SVD) of object-feature matrix to define primary features
- **Latent semantic indexing (LSI)** – SVD of term-document matrix to project terms and documents into conceptual space (Dumais et al., CHI88)
- **Eigenfaces** – EVD of covariance matrix derived from facial images (Turk & Pentland, CVPR91)
- **PageRank** – EVD of specialized Markov matrix representing the web (Page et al., WWW7, 1998)
- **HITS** – SVD of adjacency matrix of the web graph to compute hubs and authorities (Kleinberg, JACM,1999)
- **K-means and SVD equivalency** – Under certain conditions, the SVD is shown to be the relaxed solution of K-means clustering problem (Ding and He, ICML04)

Sandia National Laboratories

Tensor Basics

A tensor is a multidimensional array

An $I \times J \times K$ tensor

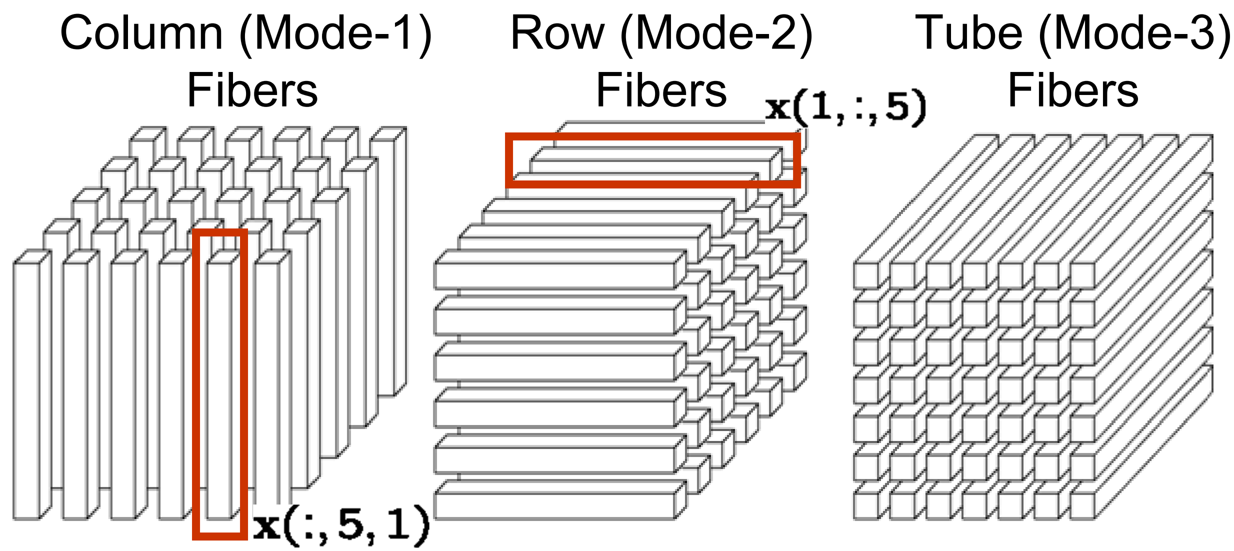


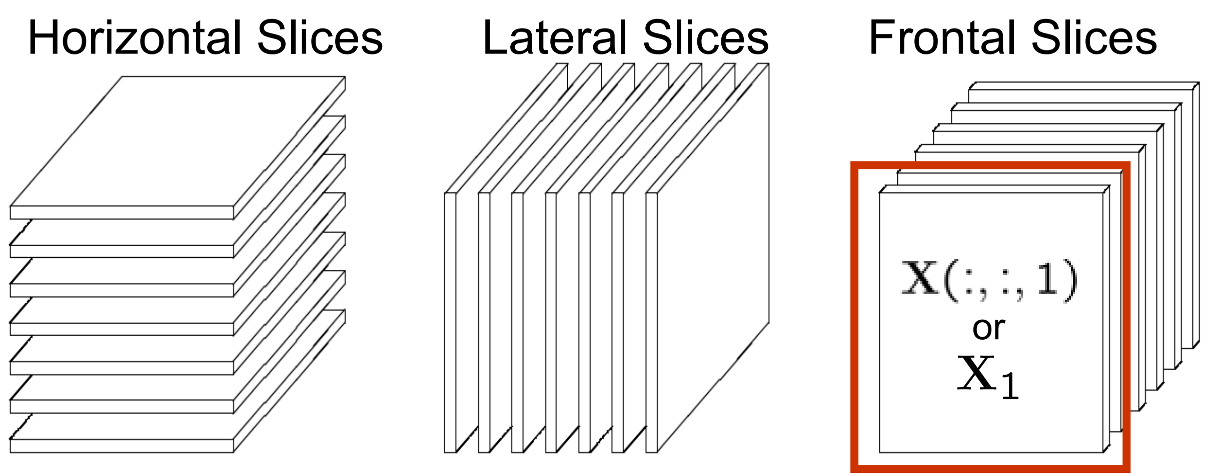
3rd order or 3-way tensor

Mode 1 has dimension I

Mode 2 has dimension J

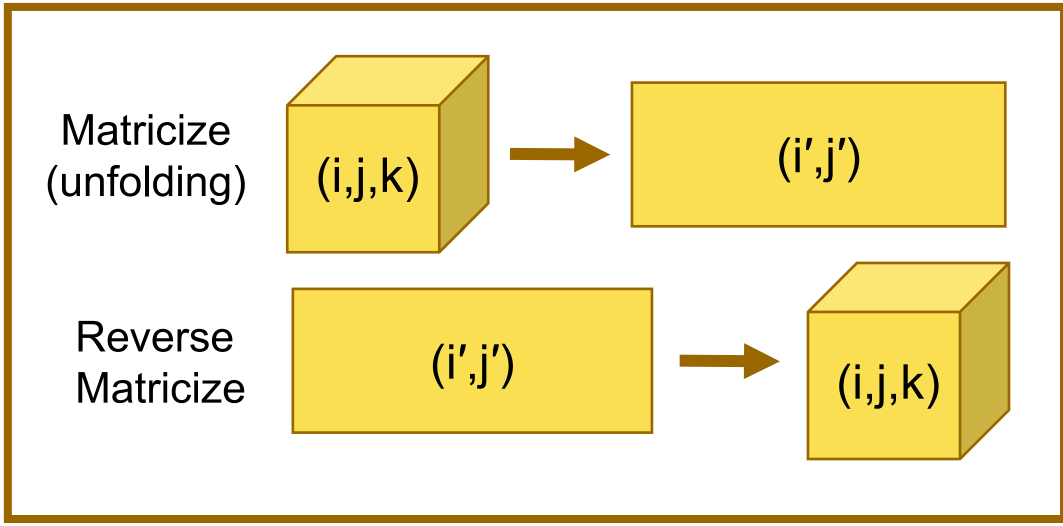
Mode 3 has dimension K



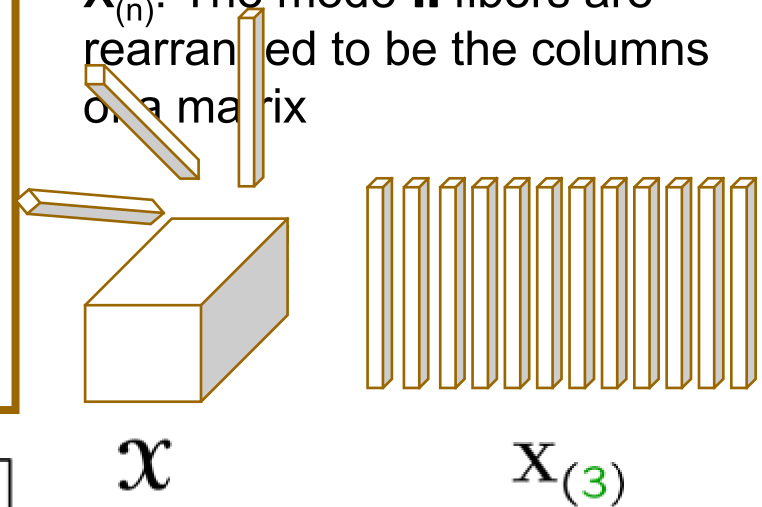


Sandia National Laboratories

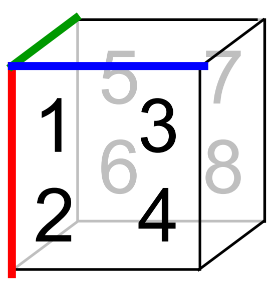
Unfolding: Converting a Tensor to a Matrix



$\mathbf{X}_{(n)}$: The mode-**n** fibers are rearranged to be the columns of a matrix



$\mathcal{X} =$



$$\mathbf{X}_{(1)} = \begin{bmatrix} 1 & 3 & 5 & 7 \\ 2 & 4 & 6 & 8 \end{bmatrix}$$

$$\mathbf{X}_{(2)} = \begin{bmatrix} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \end{bmatrix}$$

$$\mathbf{X}_{(3)} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$

$$\text{vec}(\mathcal{X}) = \text{vec}\left(\mathbf{X}_{(1)}\right) = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{bmatrix}$$

Sandia National Laboratories

Tensor Mode-n Multiplication

$$\mathcal{X} \in \mathbb{R}^{I \times J \times K}, \ \mathbf{B} \in \mathbb{R}^{M \times J}, \ \mathbf{a} \in \mathbb{R}^{I}$$

- Tensor Times Matrix

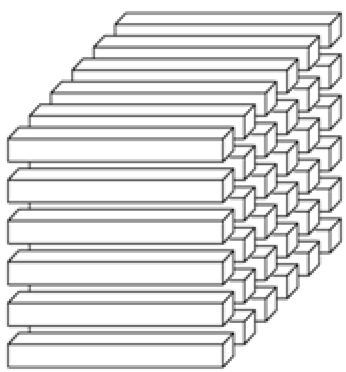
$$\mathcal{Y} = \mathcal{X} \times_2 \mathbf{B} \in \mathbb{R}^{I \times M \times K}$$

$$y_{imk} = \sum_j x_{ijk} \, b_{mj}$$

$$\mathbf{Y}_{(2)} = \mathbf{B}\mathbf{X}_{(2)}$$

Multiply each mode-2 fiber by **B**:

$$\mathbf{y}_{i:k} = \mathbf{B}\mathbf{x}_{i:k}$$



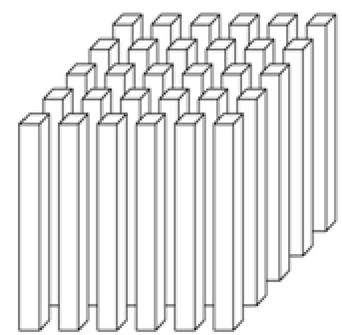
- Tensor Times Vector

$$\mathbf{Y} = \mathcal{X} \bullet_1 \mathbf{a} \in \mathbb{R}^{J \times K}$$

$$y_{jk} = \sum_i x_{ijk} \, a_i$$

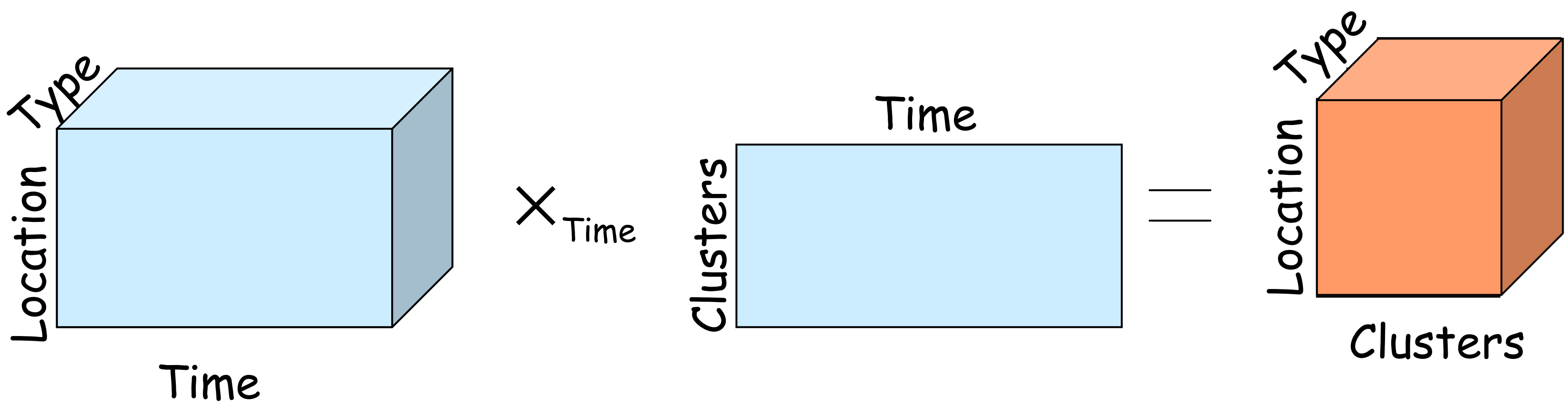
Compute the dot product of a and each mode-1 fiber:

$$y_{jk} = (\mathbf{x}_{:jk})^{\mathsf{T}} \mathbf{a}$$

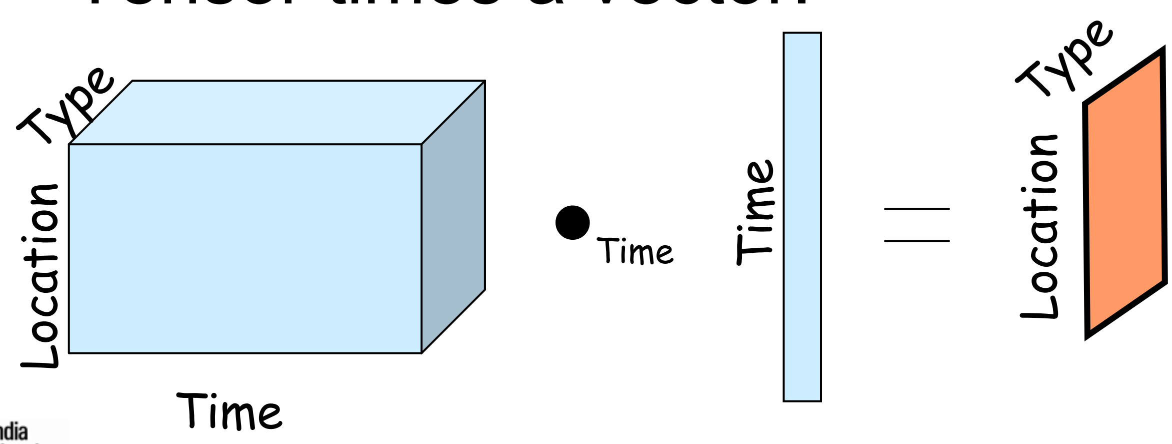


Mode-n Multiplication Example

Tensor times a matrix:



Tensor times a vector:



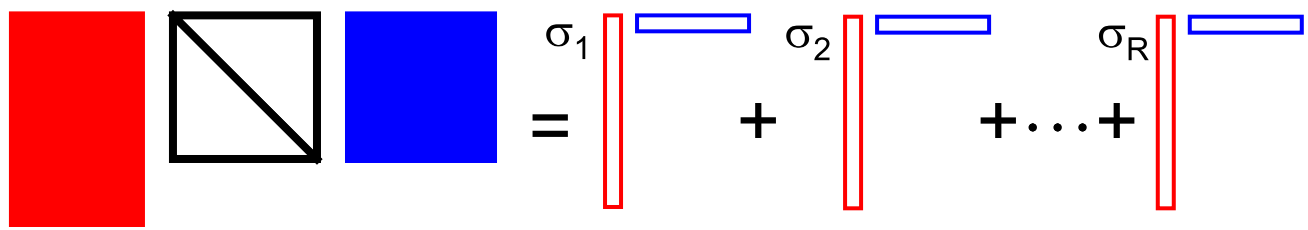
Sandia National Laboratories

(Thanks to Jimeng Sun and Christos Faloutsos for this slide.)

What is the higher-order analogue of the Matrix SVD?

Two views of the matrix SVD:

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathsf{T}} =$$



Finding bases for row and column subspaces:

$$\mathbf{X} = \boldsymbol{\Sigma} \times_1 \mathbf{U} \times_2 \mathbf{V}$$

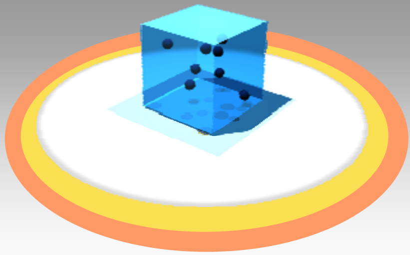
→ **Tucker Decomposition**

Sum of R rank-1 matrix factors (where R is the rank):

$$\mathbf{X} = \sum_{r=1}^{R} \sigma_r \, \mathbf{u}_r \circ \mathbf{v}_r$$

→ **CANDECOMP/ PARAFAC**

Tucker Decomposition

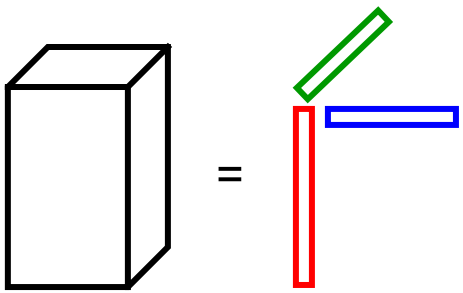


Outer Products & Kronecker Products

3-Way Outer Product

$$\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$$

$$x_{ijk} = a_i b_j c_k$$



Rank-1 Tensor

Vector Kronecker Product

$$(\mathbf{a} \otimes \mathbf{b})^{\mathsf{T}} = \begin{bmatrix} a_1 \mathbf{b}^{\mathsf{T}} & a_2 \mathbf{b}^{\mathsf{T}} & \cdots & a_N \mathbf{b}^{\mathsf{T}} \end{bmatrix}$$

Matrix Kronecker Product

$$\underset{M \times N}{\mathbf{A}} \otimes \underset{P \times Q}{\mathbf{B}} = \underset{MP \times NQ}{\begin{bmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1N}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2N}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M1}\mathbf{B} & a_{M2}\mathbf{B} & \cdots & a_{MN}\mathbf{B} \end{bmatrix}}$$

$$= \begin{bmatrix} \mathbf{a}_1 \otimes \mathbf{b}_1 & \mathbf{a}_1 \otimes \mathbf{b}_2 & \cdots & \mathbf{a}_N \otimes \mathbf{b}_Q \end{bmatrix}$$

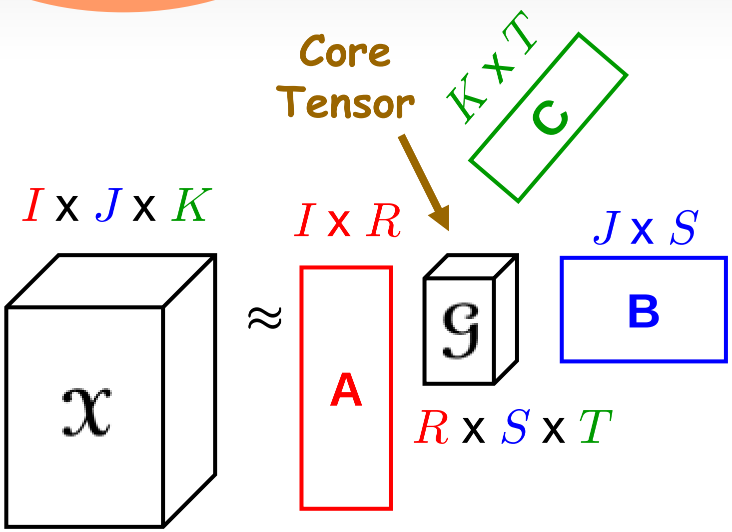
Observe

The Kronecker product and outer product yield the same results, just shaped differently:

$$\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} \Rightarrow \mathrm{vec}(\mathcal{X}) = \mathbf{c} \otimes \mathbf{b} \otimes \mathbf{a}$$

Sandia National Laboratories

Tucker Decomposition



$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$$

$$\mathcal{X} = \sum_{r=1}^{R} \sum_{s=1}^{S} \sum_{t=1}^{T} g_{rst}\, \mathbf{a}_r \circ \mathbf{b}_s \circ \mathbf{c}_t$$

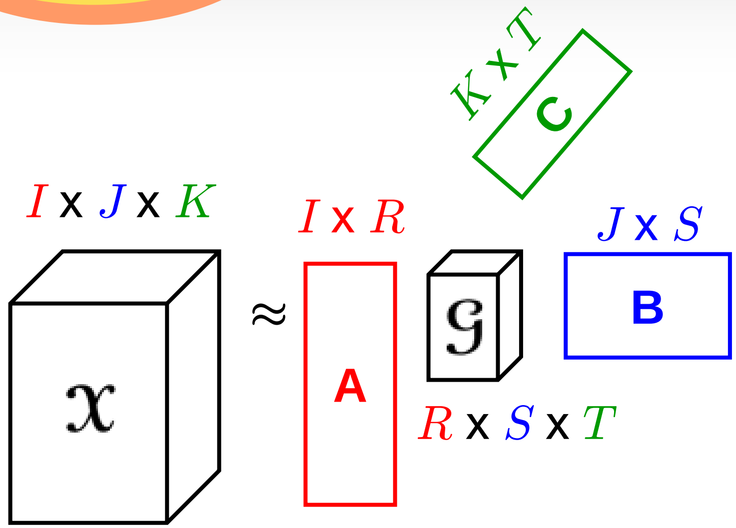
RST rank-1 factors

- Also known as: three-mode factor analysis, three-mode PCA, orthogonal array decomposition
- Dimensions R, S, T chosen by the user.
- **A**, **B**, and **C** may be orthonormal (generally assume full column rank)
- $\mathcal{G}$ is <u>not</u> diagonal
- Not unique

See Tucker, *Psychometrika*, 1966; see also Hitchcock, 1927.

Sandia National Laboratories

Matrix & Vector Forms of Tucker Decomposition



$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$$

"Matricized" Tucker

$$\mathbf{X}_{(1)} \approx \mathbf{A}\mathbf{G}_{(1)}(\mathbf{C} \otimes \mathbf{B})^{\mathsf{T}}$$

$$\mathbf{X}_{(2)} \approx \mathbf{B}\mathbf{G}_{(2)}(\mathbf{C} \otimes \mathbf{A})^{\mathsf{T}}$$

$$\mathbf{X}_{(3)} \approx \mathbf{C}\mathbf{G}_{(3)}(\mathbf{B} \otimes \mathbf{A})^{\mathsf{T}}$$

"Vectorized" Tucker

$$\text{vec}(\mathcal{X}) \approx (\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A})\text{vec}(\mathcal{G})$$

Given **A**, **B**, **C**, with orthonormal columns, the optimal core is:

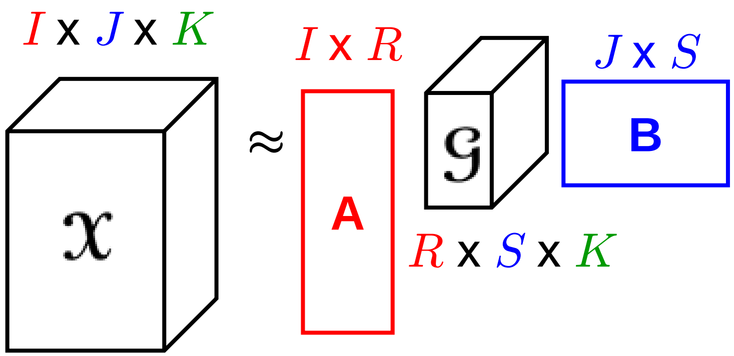
$$\mathcal{G} = \mathcal{X} \times_1 \mathbf{A}^{\dagger} \times_2 \mathbf{B}^{\dagger} \times_3 \mathbf{C}^{\dagger}$$

Sandia National Laboratories

Tucker Variations

Details

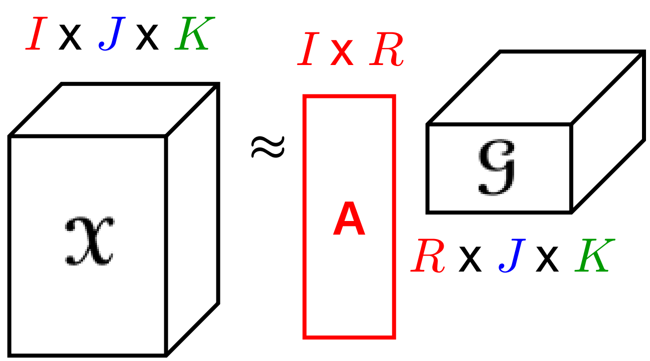
Tucker 2



$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{I}$$

$$\mathbf{X}_{(3)} \approx \mathbf{G}_{(3)} (\mathbf{B} \otimes \mathbf{A})^{\mathsf{T}}$$

Tucker 1



$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{I} \times_3 \mathbf{I}$$

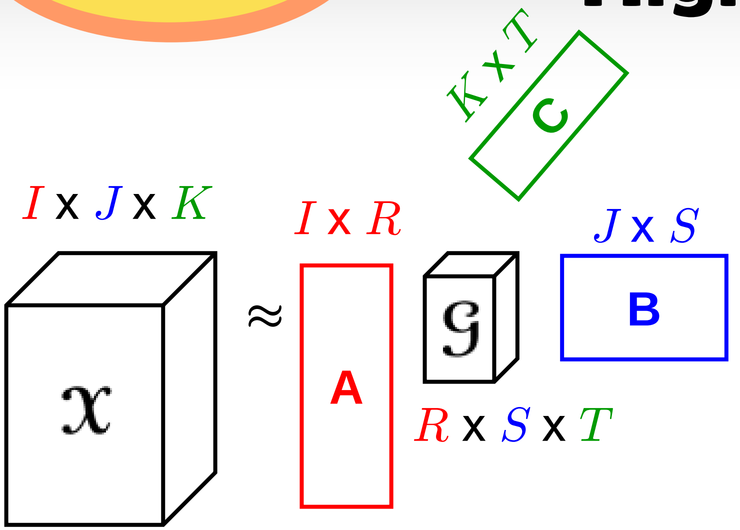
$$\underbrace{\mathbf{X}_{(1)} \approx \mathbf{A}\mathbf{G}_{(1)}}$$

Can be solved via SVD.

See Kroonenberg & De Leeuw, *Psychometrika*,1980 for discussion.

Sandia National Laboratories

Fitting Tucker via the Higher Order SVD (HO-SVD)



$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$$

Not optimal, but often used to initialize other algorithms.

$\mathbf{A}$ = leading R left singular vectors of $\mathbf{X}_{(1)}$

$\mathbf{B}$ = leading S left singular vectors of $\mathbf{X}_{(2)}$

$\mathbf{C}$ = leading T left singular vectors of $\mathbf{X}_{(3)}$

(Observe connection to Tucker 1)

$$\mathcal{G} = \mathcal{X} \times_1 \mathbf{A}^\dagger \times_2 \mathbf{B}^\dagger \times_3 \mathbf{C}^\dagger$$

De Lathauwer, De Moor, & Vandewalle, SIMAX, 2000.
Also known as “Method 1” in Tucker, 1966.

Sandia National Laboratories

Fitting Tucker via Alternating Least Squares (ALS)

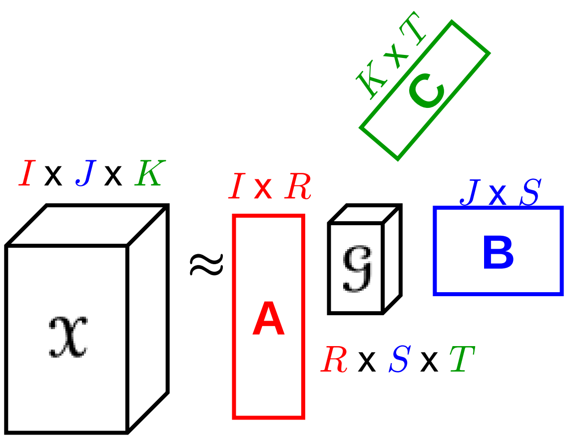
1. Initialize A, B, C.

2. Repeat the following steps until “convergence”:

$$\mathbf{A} = \text{leading } R \text{ left singular vectors of } \mathbf{X}_{(1)}(\mathbf{C} \otimes \mathbf{B})$$

$$\mathbf{B} = \text{leading } S \text{ left singular vectors of } \mathbf{X}_{(2)}(\mathbf{C} \otimes \mathbf{A})$$

$$\mathbf{C} = \text{leading } T \text{ left singular vectors of } \mathbf{X}_{(3)}(\mathbf{B} \otimes \mathbf{A})$$



3. Finally, set:

$$\mathcal{G} = \mathcal{X} \times_1 \mathbf{A}^\dagger \times_2 \mathbf{B}^\dagger \times_3 \mathbf{C}^\dagger$$

See Kroonenberg & De Leeuw, *Psychometrika*, 1980, and De Lathauwer, De Boor, and Vandewalle, *SIMAX*, 2000.

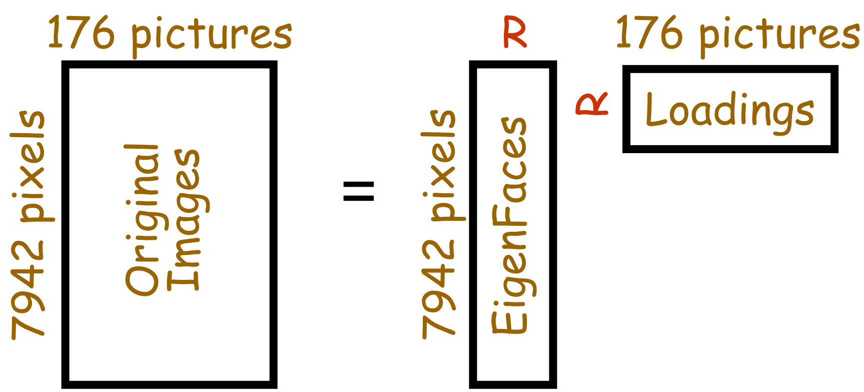
New!

Newton-Grassmann method proposed by Eldén and Savas, 2007.

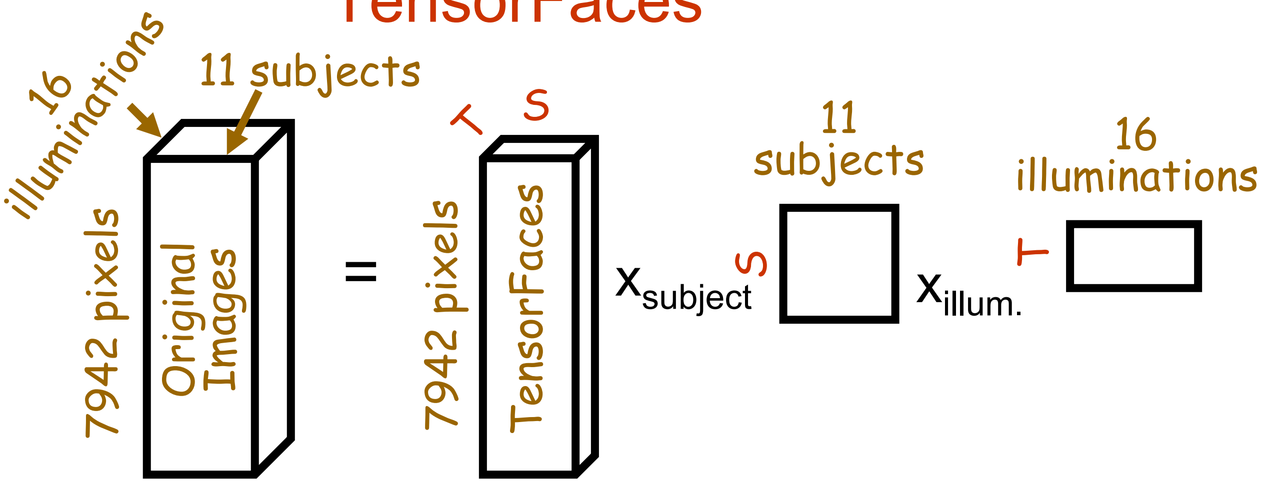
Application: TensorFaces

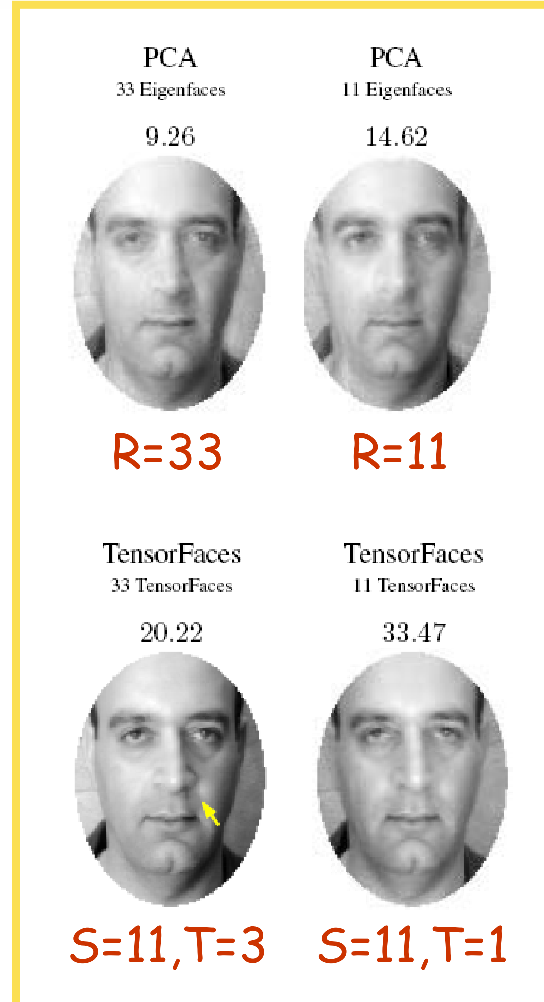
TensorFaces: An Application of the Tucker2 Decomposition

EigenFaces



TensorFaces

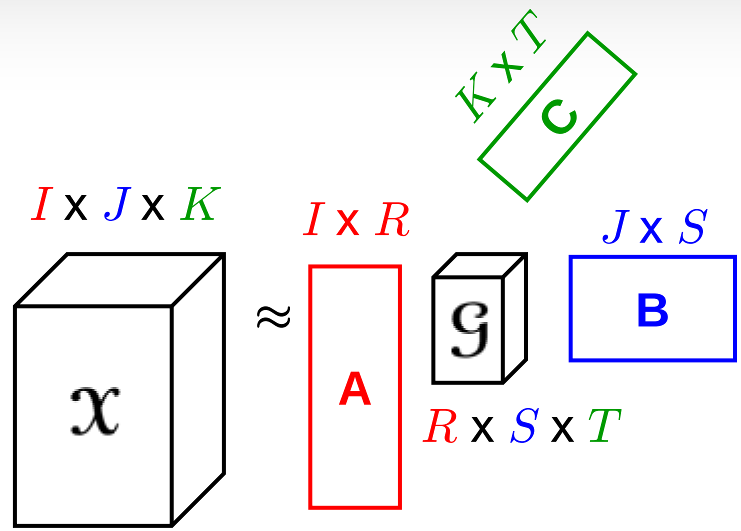




M.A.O. Vasilescu & D. Terzopoulos, CVPR'03

Issue: Tucker is Not Unique

Details



Tucker decomposition is *not* unique. Let Y be an R x R orthogonal matrix. Then…

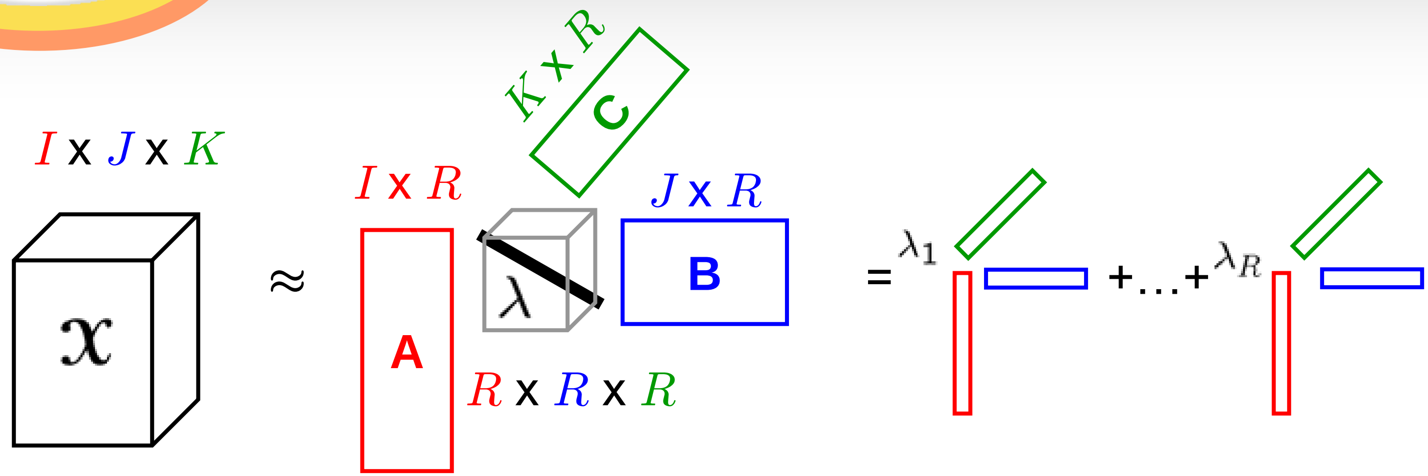
$$\mathcal{X} \approx \mathcal{G} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C} = \left(\mathcal{G} \times_1 \mathbf{Y}^{\mathsf{T}}\right) \times_1 (\mathbf{A}\mathbf{Y}) \times_2 \mathbf{B} \times_3 \mathbf{C}$$

$$\mathbf{X}_{(1)} \approx \mathbf{A}\mathbf{G}_{(1)}(\mathbf{C} \otimes \mathbf{B})^{\mathsf{T}} = \mathbf{A}\mathbf{Y}\mathbf{Y}^{\mathsf{T}}\mathbf{G}_{(1)}(\mathbf{C} \otimes \mathbf{B})^{\mathsf{T}}$$

Sandia National Laboratories

CANDECOMP/PARAFAC Decomposition

CANDECOMP/PARAFAC (CP)

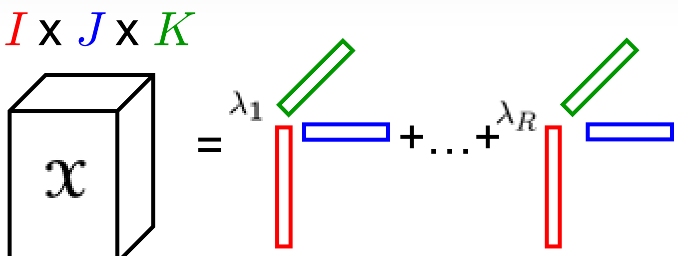


$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

- CANDECOMP = Canonical Decomposition,
- PARAFAC = Parallel Factors
- Optional core is <u>diagonal</u> (specified by the vector λ)
- Columns of **A**, **B**, and **C** are <u>not</u> orthonormal
- Exact decomposition is often unique

Carroll & Chang, *Psychometrika*, 1970, Harshman, 1970 – plus Hitchcock, 1927.

CP & Tensor Rank



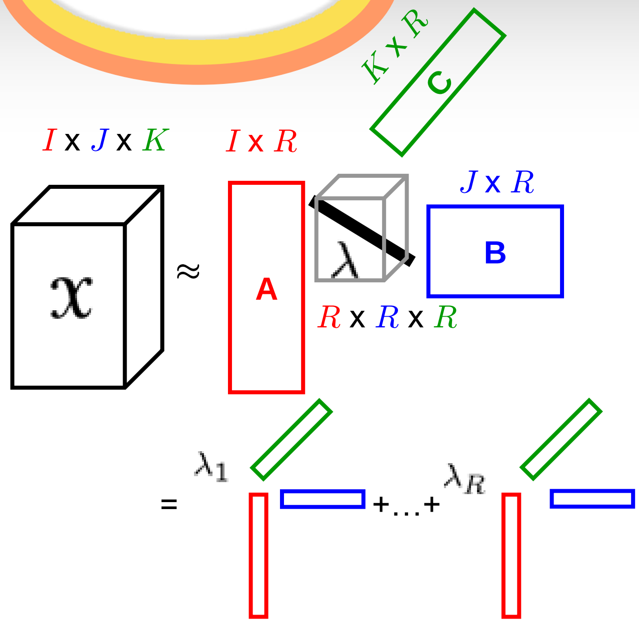
$$\mathcal{X} = \sum_{r=1}^{R} \lambda_r \, \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

> Tensor Rank: The rank of a tensor $\mathcal{X}$, denoted rank($\mathcal{X}$), is the smallest number of rank-1 factors that generate $\mathcal{X}$ as their sum.

- No straightforward method to determine the rank of a specific tensor
- Maximum rank = maximum achievable rank
  - 2 x 2 x 2 tensor = 3, 3 x 3 x 2 tensor = 4
- Typical rank = occurs with probability greater than zero
  - 2 x 2 x 2 tensor = {2,3}, 3 x 3 x 2 = {3,4}
- Border Rank = Minimum number of rank-1 tensors sufficient to approximate given tensor with arbitrarily small nonzero error
  - "Best" low-rank approximation may not exist

> See also Kruskal, *LAA*, 1977; Kruskal, 1989; J. M. F. Ten Berge, *Psychometrika*, 1991; Bini et al., *Inform. Process. Lett., 1979*

Matrix & Vector Forms of CP Decomposition



$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

Matrix Khatri-Rao Product

$$\mathbf{A} \odot \mathbf{B} = \left[\mathbf{a}_1 \otimes \mathbf{b}_1 \quad \mathbf{a}_2 \otimes \mathbf{b}_2 \quad \cdots \quad \mathbf{a}_R \otimes \mathbf{b}_R \right]$$

M x R   N x R   MN x R

Useful Fact

$$(\mathbf{A} \odot \mathbf{B})^{\dagger} = ((\mathbf{A}^{\mathsf{T}}\mathbf{A}) * (\mathbf{B}^{\mathsf{T}}\mathbf{B}))^{\dagger} (\mathbf{A} \odot \mathbf{B})^{\mathsf{T}}$$

"Matricized" CP

$$\mathbf{X}_{(1)} \approx \mathbf{A}\Lambda(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$$

$$\mathbf{X}_{(2)} \approx \mathbf{B}\Lambda(\mathbf{C} \odot \mathbf{A})^{\mathsf{T}}$$

$$\mathbf{X}_{(3)} \approx \mathbf{C}\Lambda(\mathbf{B} \odot \mathbf{A})^{\mathsf{T}}$$

$$\Lambda = \mathrm{diag}(\lambda)$$

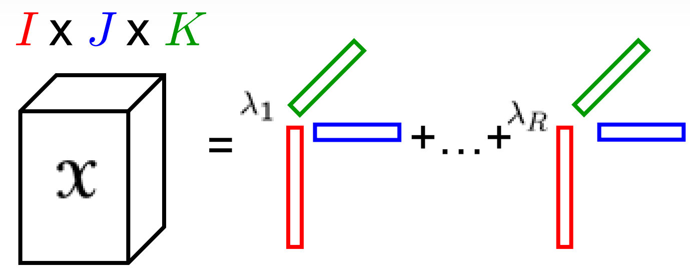
"Vectorized" CP

$$\mathrm{vec}(\mathcal{X}) \approx (\mathbf{C} \odot \mathbf{B} \odot \mathbf{A})\lambda$$

Sandia National Laboratories

Fitting CP via Alternating Least Squares (ALS)

$$\mathcal{X} = \sum_{r=1}^{R} \lambda_r \, \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$



1. Initialize A, B, C.
2. Repeat the following steps until "convergence":

$$\mathbf{A} = \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B})(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{B}^{\mathsf{T}}\mathbf{B})^{\dagger}$$

$$\mathbf{B} = \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A})(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{A}^{\mathsf{T}}\mathbf{A})^{\dagger}$$

$$\mathbf{C} = \mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A})(\mathbf{B}^{\mathsf{T}}\mathbf{B} * \mathbf{A}^{\mathsf{T}}\mathbf{A})^{\dagger}$$

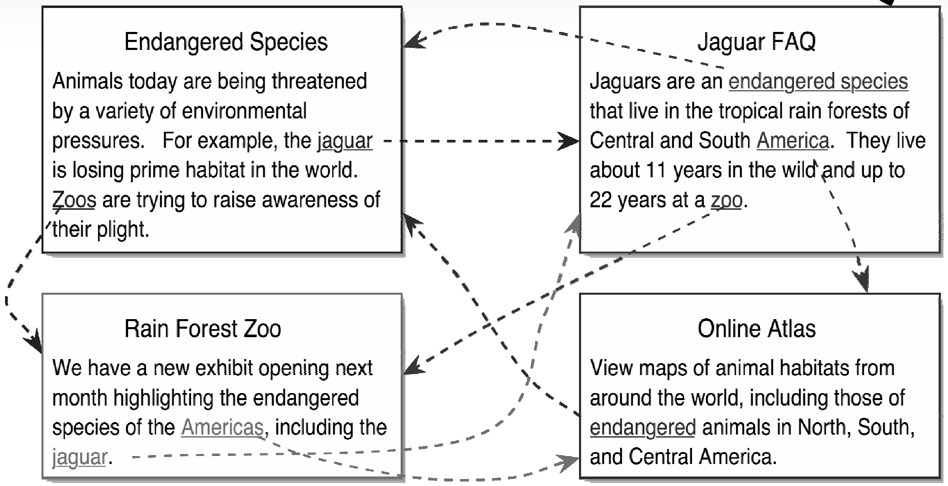
(normalize each matrix in turn to get Λ)

Survey: Tomasi & Bro, Computational Statistics & Data Analysis, 2006.

Sandia National Laboratories

Application: TOPHITS

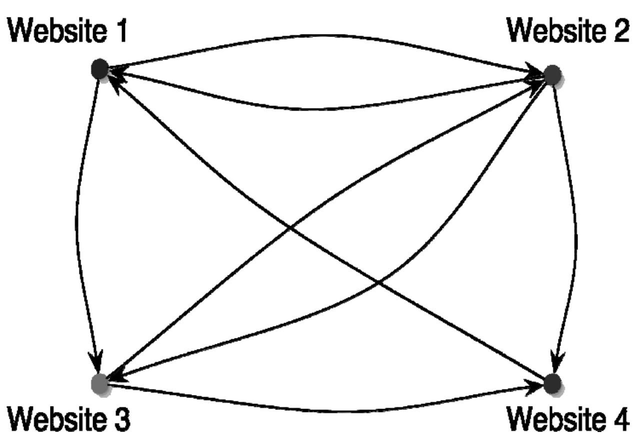
Hubs and Authorities (the HITS method)

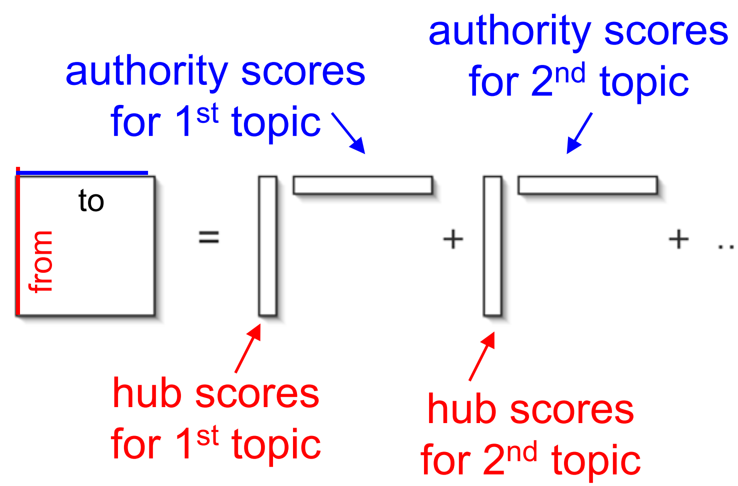


Sparse adjacency matrix and its SVD:

$$x_{ij} = \begin{cases} 1 & \text{if page } i \text{ links to page } j \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{X} = \sum_r \sigma_r \, \mathbf{h}_r \circ \mathbf{a}_r$$





Sandia National Laboratories

Jon M. Kleinberg. *JACM*, 1999

HITS Authorities on Sample Data

1st Principal Factor	
.97	www.ibm.com
.24	www.alphaw
.08	www-128.ibm
.05	www.develop
.02	www.researc
.01	www.redbook
.01	news.com.co

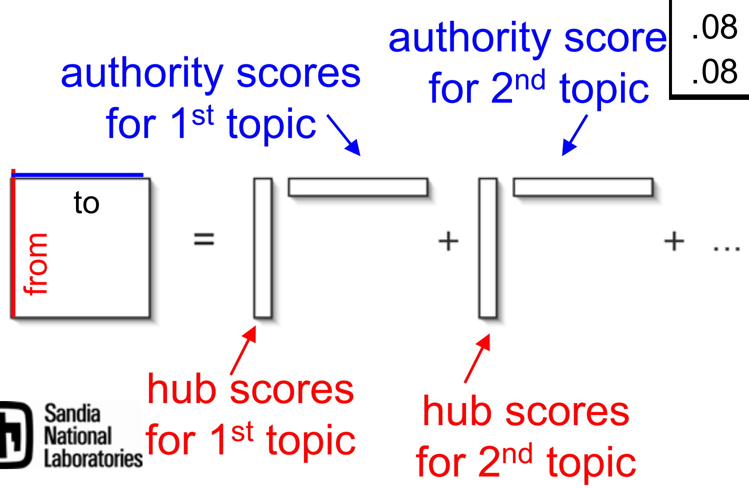
2nd Principal Factor	
.99	www.lehigh.edu
.11	www2.lehigh
.06	www.lehigha
.06	www.lehighs
.02	www.bethleh
.02	www.adobe.c
.02	lewisweb.cc.
.02	www.leo.lehi
.02	www.distanc
.02	fp1.cc.lehigh

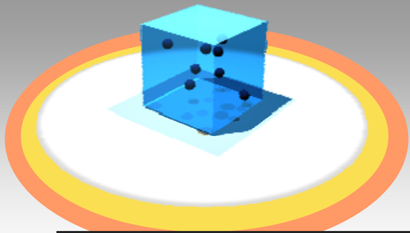
3rd Principal Factor	
.75	java.sun.com
.38	www.sun.com
.36	developers.sun.
.24	see.sun.com
.16	www.samag.co
.13	docs.sun.com
.12	blogs.sun.com
.08	sunsolve.sun.co
.08	www.sun-catalo
.08	news.com.com

4th Principal Factor	
.60	www.pueblo.gsa.gov
.45	www.whitehouse.gov
.35	www.irs.gov
.31	travel.state
.22	www.gsa.g
.20	www.ssa.g
.16	www.censu
.14	www.govbe
.13	www.kids.g
.13	www.usdoj

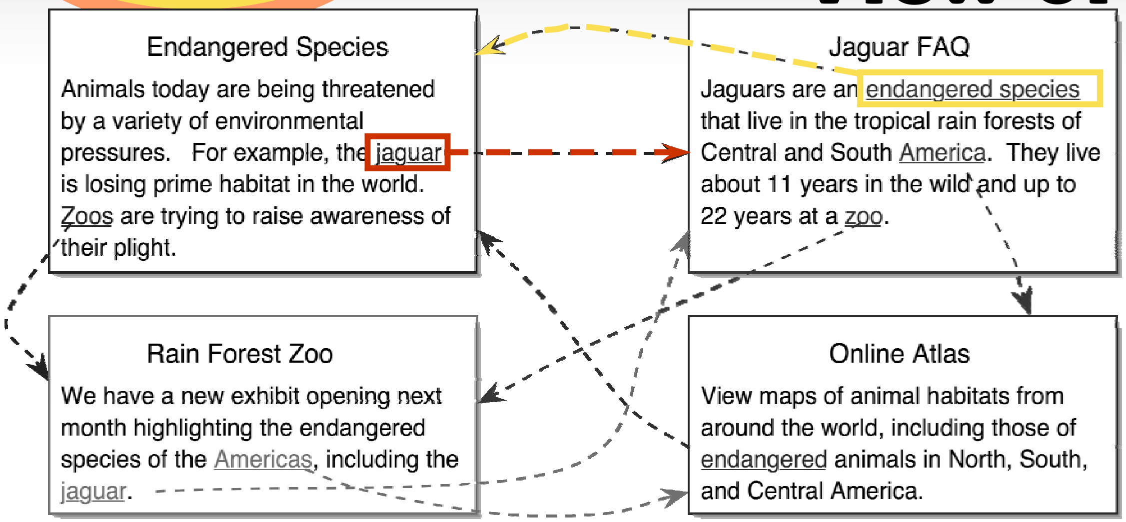
6th Principal Factor	
.97	mathpost.asu.edu
.18	math.la.asu.edu
.17	www.asu.edu
.04	www.act.org
.03	www.eas.asu.edu
.02	archives.math.utk.edu
.02	www.geom.uiuc.edu
.02	www.fulton.asu.edu
.02	www.amstat.org
.02	www.maa.org

We started our crawl from http://www-neos.mcs.anl.gov/neos, and crawled 4700 pages, resulting in 560 cross-linked hosts.





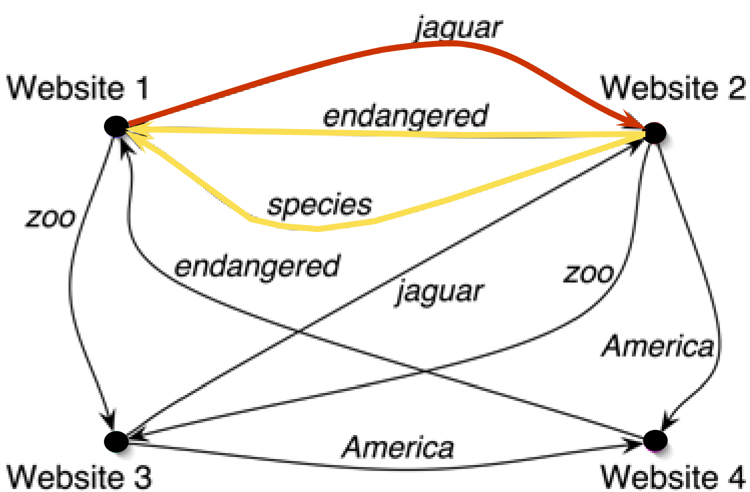
TOPHITS – A Three-Dimensional View of the Web

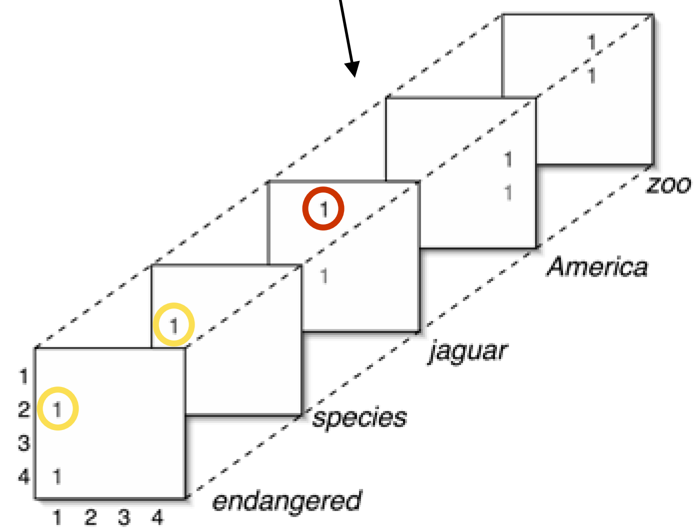


Tensor Definition

$$x_{ijk} = \begin{cases} 1 & \text{if page } i \rightarrow \text{page } j \text{ with term } k \\ 0 & \text{otherwise} \end{cases}$$

Observe that this tensor is very sparse!



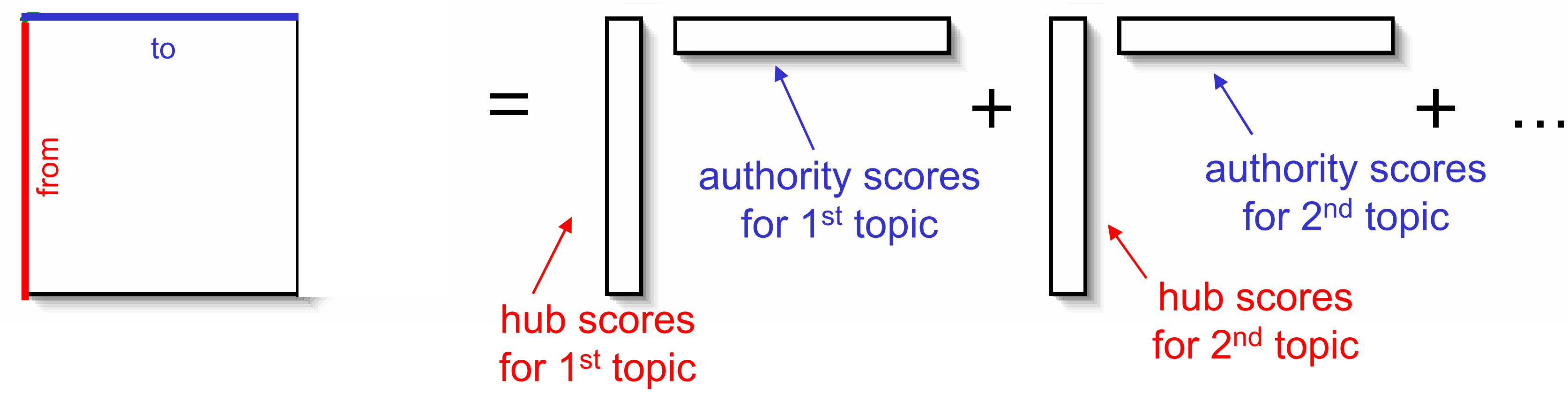


Sandia National Laboratories

Topical HITS (TOPHITS)

Main Idea: Extend the idea behind the HITS model to incorporate term (i.e., topical) information.

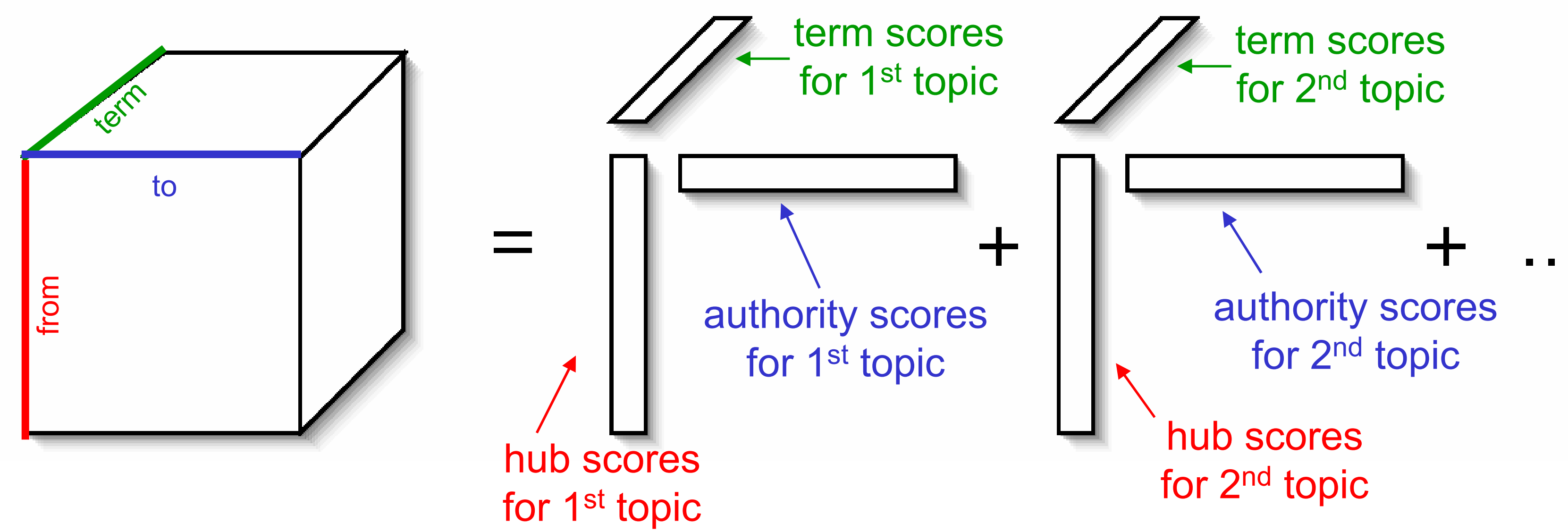
$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, \mathbf{h}_r \circ \mathbf{a}_r$$



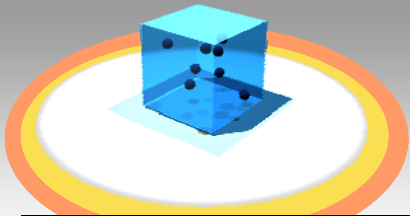
Topical HITS (TOPHITS)

Main Idea: Extend the idea behind the HITS model to incorporate term (i.e., topical) information.

$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, \mathbf{h}_r \circ \mathbf{a}_r \circ \mathbf{t}_r$$



Sandia National Laboratories



TOPHITS Terms & Authorities on Sample Data

TOPHITS uses 3D analysis to find the dominant groupings of web pages and terms.

$$x_{ijk} = \begin{cases} \frac{1}{\log(w_k)+1} & \text{if } i \rightarrow j \text{ with term } k \\ 0 & \text{otherwise} \end{cases}$$

w_k = # unique links using term k

1st Principal Factor

.23	JAVA	.86	java.sun.com
.18	SUN	.38	developers.sun.com
.17	PLATF		
.16	SOLAR		
.16	DEVEL		
.15	EDITIO		
.15	DOWN		
.14	INFO		
.12	SOFTW		
.12	NO-RE		

2nd Principal Factor

.20	NO-READABLE-TEXT	.99	www.lehigh.edu
.16	FACUL		
.16	SEARC		
.16	NEWS		
.16	LIBRAR		
.16	COMP		
.12	LEHIG		

3rd Principal Factor

.15	NO-READABLE-TEXT	.97	www.ibm.com
.15	IBM	.18	www.alphaworks.ibm.com
.12	SERVI		
.12	WEBS		
.12	WEB		
.11	DEVEL		
.11	LINUX		
.11	RESOU		
.11	TECHN		
.10	DOWN		

4th Principal Factor

.26	INFORMATION	.87	www.pueblo.gsa.gov
.24	FEDERAL	.24	www.irs.gov
.23	CITIZE		
.22	OTHER		
.19	CENTE		
.19	LANGU		
.15	U.S		
.15	PUBLI		
.14	CONS		
.13	FREE		

6th Principal Factor

.26	PRESIDENT	.87	www.whitehouse.gov
.25	NO-READABLE-TEXT	.18	www.irs.gov
.25	BUSH		
.25	WELC		
.17	WHIT		
.16	U.S		
.15	HOUS		
.13	BUDG		
.13	PRES		
.11	OFFIC		

12th Principal Factor

.75	OPTIMIZATION	.35	www.palisade.com
.58	SOFTWARE	.35	www.solver.com
.08	DECIS		
.07	NEOS		
.06	TREE		
.05	GUIDE		
.05	SEARC		
.05	ENGIN		
.05	CONTR		
.05	ILOG		

13th Principal Factor

.46	ADOBE	.99	www.adobe.com
.45	READER		
.45	ACROB		
.30	FREE		
.30	NO-RE		
.29	HERE		
.29	COPY		
.05	DOWN		

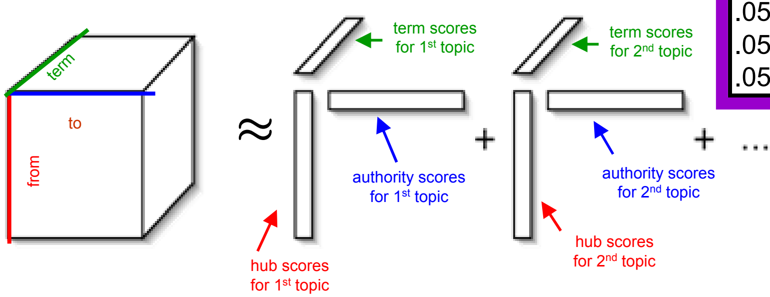
16th Principal Factor

.50	WEATHER	.81	www.weather.gov
.24	OFFICE	.41	www.spc.noaa.gov
.23	CENTER	.30	lwf.ncdc.noaa.gov
.19	NO-RE		
.17	ORGA		
.15	NWS		
.15	SEVER		
.15	FIRE		
.15	POLIC		
.14	CLIMA		

19th Principal Factor

.22	TAX	.73	www.irs.gov
.17	TAXES	.43	travel.state.gov
.15	CHILD	.22	www.ssa.gov
.15	RETIREMENT	.08	www.govbenefits.gov
.14	BENEFITS	.06	www.usdoj.gov
.14	STATE	.03	www.census.gov
.14	INCOME	.03	www.usmint.gov
.13	SERVICE	.02	www.nws.noaa.gov
.13	REVENUE	.02	www.gsa.gov
.12	CREDIT	.01	www.annualcreditreport.com

Tensor PARAFAC

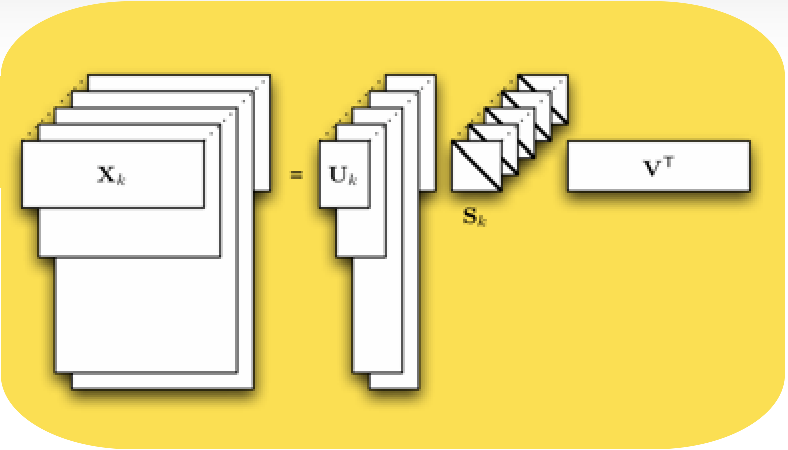


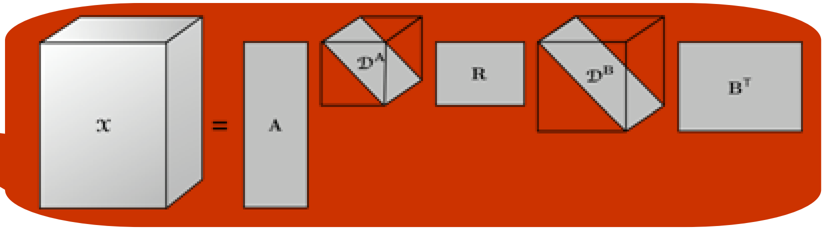
Sandia National Laboratories

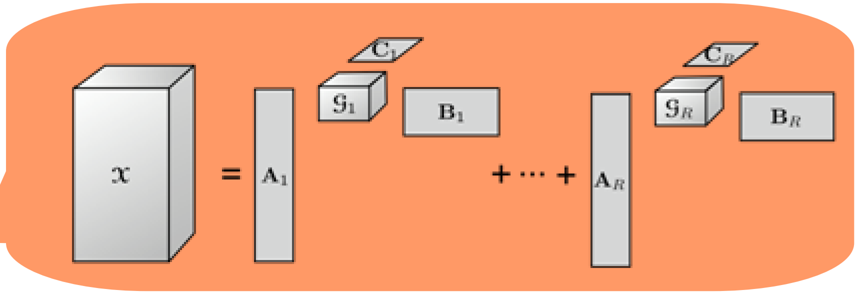
Other Decompositions

Other Decompositions

- INDSCAL: Individual Differences in Scaling (Carroll & Chang, 1972)
- PARAFAC2 (Harshman, 1978)
- CANDELINC: Linearly constrained CP (Carroll, Pruzansky, Kruskal, 1980)
- DEDICOM: Decomposition into directional components (Harshman, 1972)
- PARATUCK2: Generalization of DEDICOM (Harshman & Lundy, 1996)
- Nonnegative tensor factorizations (Bro and De Jung, 1997; Paatero, 1997; Welling and Weber, 2001; etc.)
- Block factorizations (De Lathauwer, 2007; etc.)



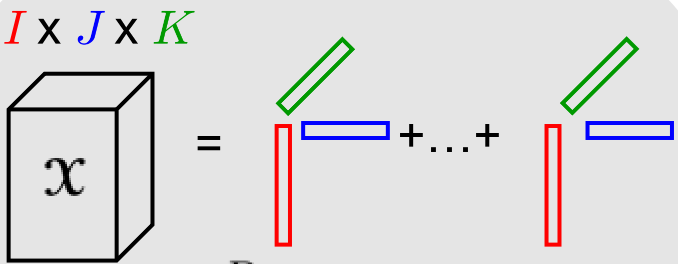




Sandia National Laboratories

PARAFAC2

Yet another view of PARAFAC



$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

$$\mathbf{X}_{(1)} \approx \mathbf{A}(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$$

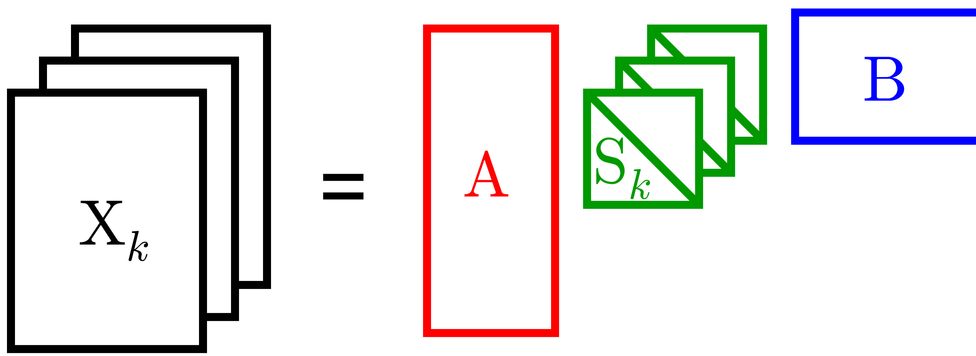
$$\mathbf{X}_{(2)} \approx \mathbf{B}(\mathbf{C} \odot \mathbf{A})^{\mathsf{T}}$$

$$\mathbf{X}_{(3)} \approx \mathbf{C}(\mathbf{B} \odot \mathbf{A})^{\mathsf{T}}$$

$$\text{vec}(\mathcal{X}) \approx (\mathbf{C} \odot \mathbf{B} \odot \mathbf{A})\mathbf{1}$$

$$\mathrm{X}_k = \mathrm{A}\mathrm{S}_k\mathrm{B}^{\mathsf{T}}$$

S_k = diag(k^{th} _row_ of C)



This representation only works for 3rd-order tensors.

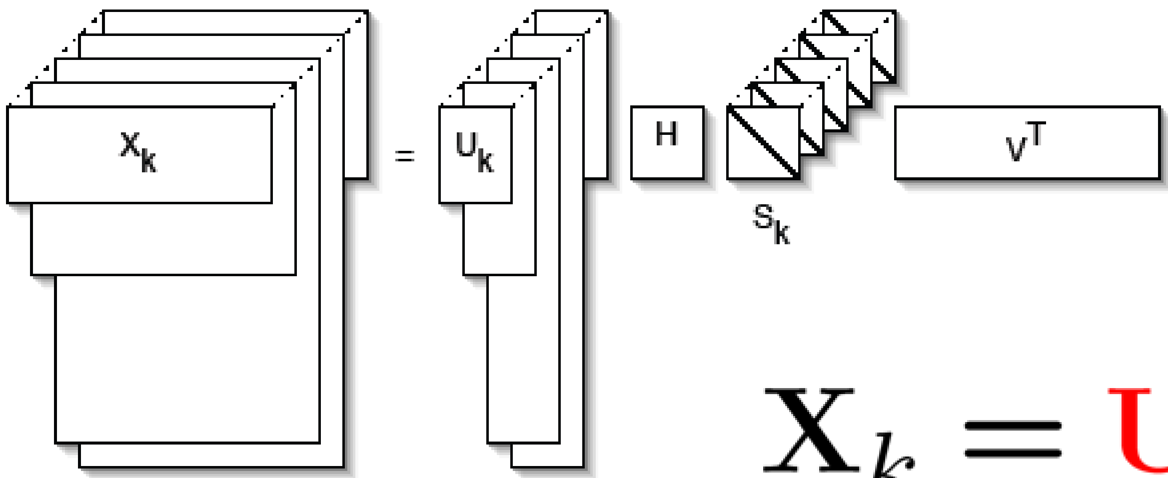
Looks like SVD.

PARAFAC2

(not, strictly speaking, a tensor decomposition)

PARAFAC

$$\mathbf{X}_k = \mathbf{A}\mathbf{S}_k\mathbf{B}^\mathsf{T}$$



$$\mathbf{X}_k = \mathbf{U}_k\mathbf{H}\mathbf{S}_k\mathbf{V}^\mathsf{T}$$

Not a tensor, but similar

orthonormal columns

used to enforce uniqueness

diagonal

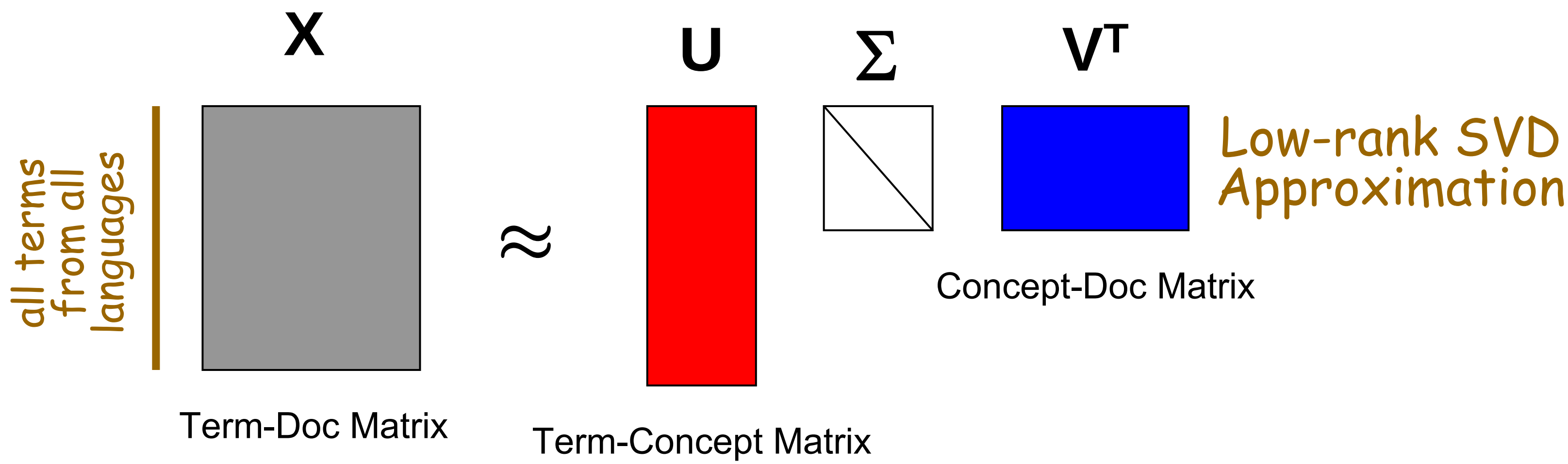
R. A. Harshman, *UCLA Working Papers in Phonetics*, 1972.

Application: Cross-Language Information Retrieval

Sandia National Laboratories

Latent Semantic Indexing (LSI) in Multilingual Environment

Step 1: Compute SVD on Parallel Corpus for training. Each “document” consists of <u>all</u> its translations.



Step 2: Map test documents to concept space. Each document is only a <u>single</u> translation.

$$\hat{Y} = UY$$

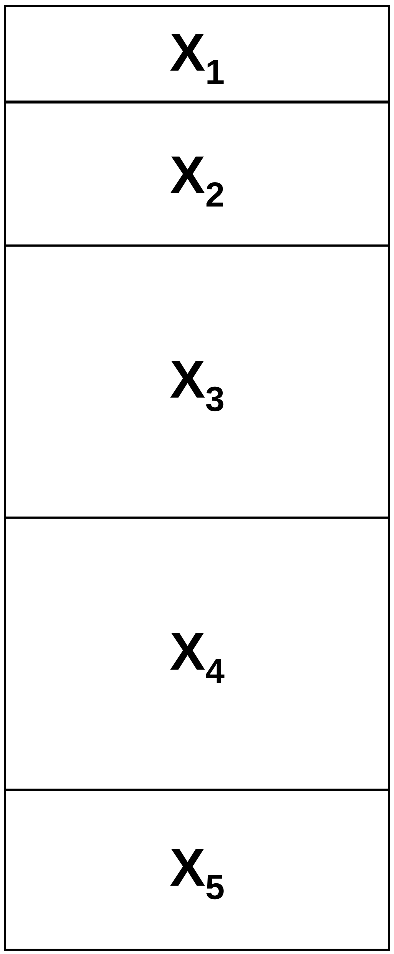
Same U for all languages.

Goal

Different translations of the same document should be nearby in concept space.

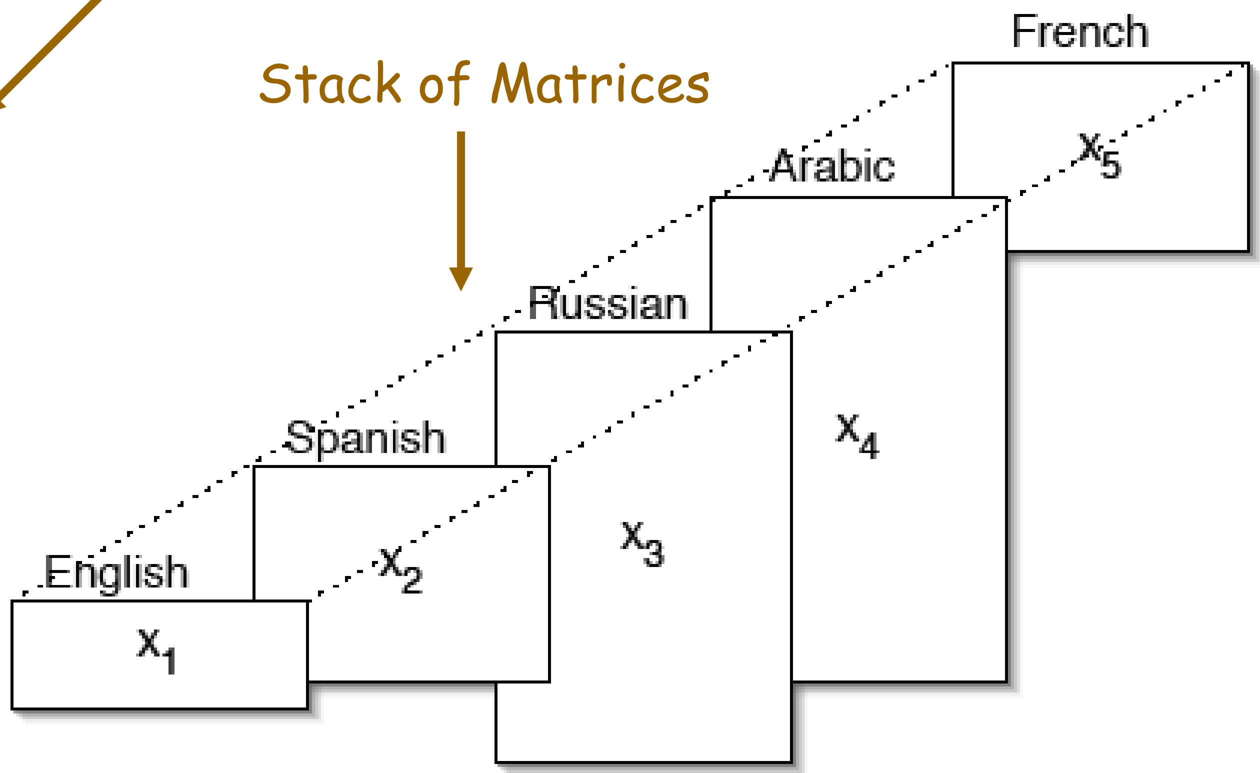
Sandia National Laboratories

A Different View



LSI Matrix (though terms are mixed)

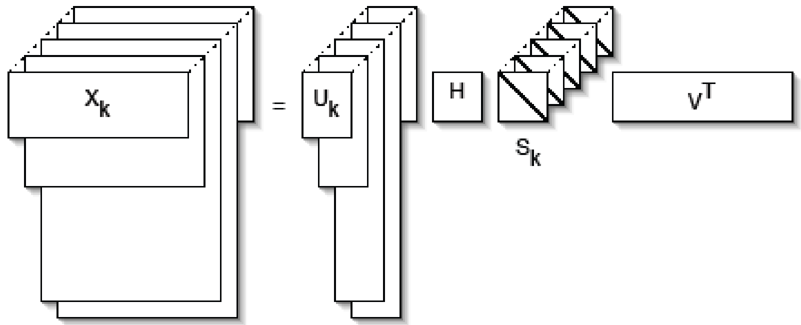
Stack of Matrices



Sandia National Laboratories

PARAFAC2 Model

Step 1: Compute PARAFAC2 on Parallel Corpus for training. Each “document” consists of <u>all</u> its translations.



Step 2: Map test documents to concept space. Each document is only a <u>single</u> translation.

$$\hat{\mathbf{Y}}_k = \mathbf{U}_k \mathbf{Y}_k$$

Minor Drawback

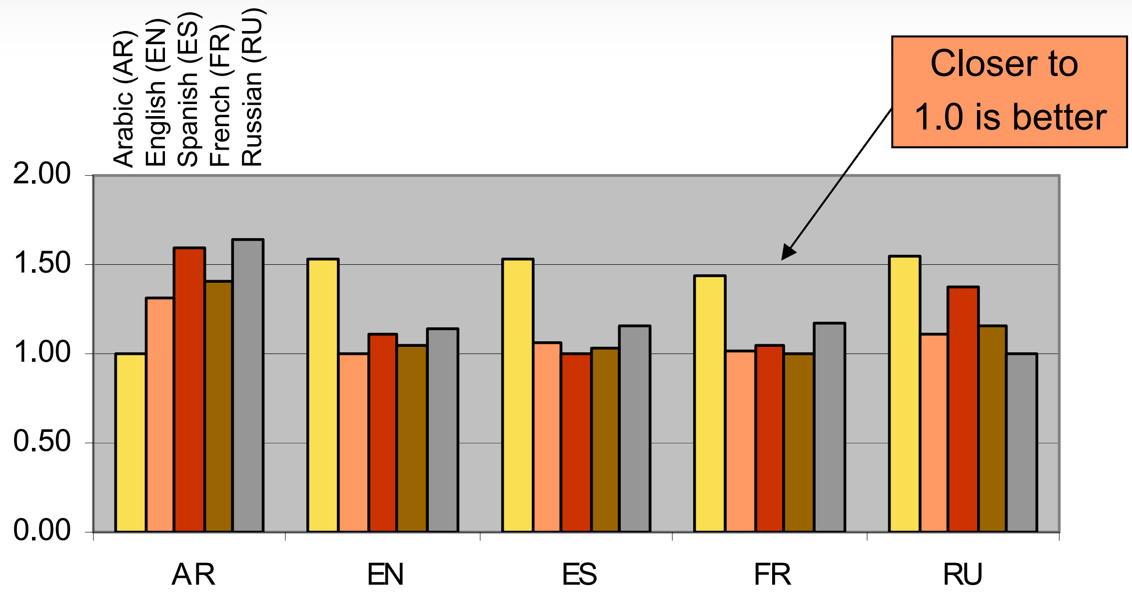
Need to know language of test document.

Results Comparison

Trained on Bible. Tested on Quran.

SVD

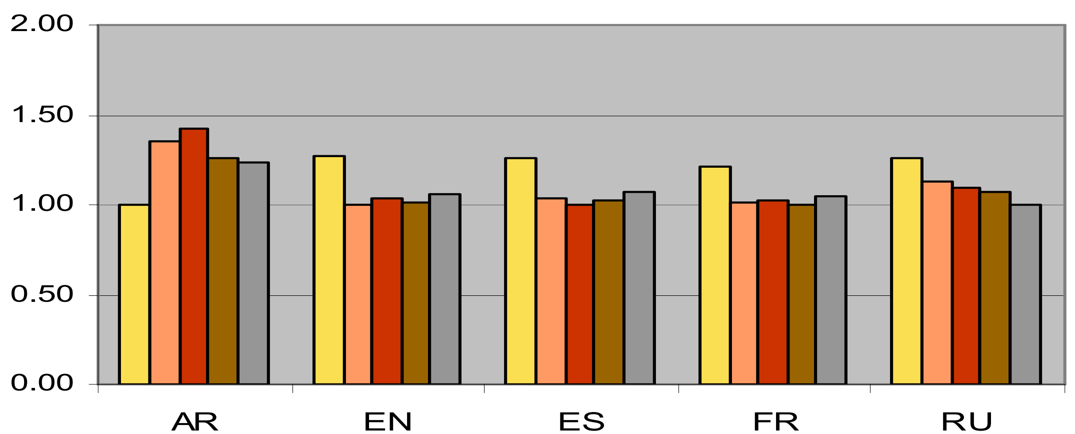
Rank-300



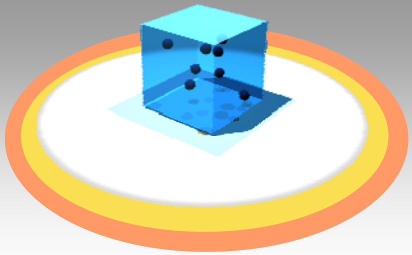
For each document in each language on the vertical axis, we ranked documents in each of the other languages. The bar represents the average rank of the correct document. Rank 1 is ideal.

PARAFAC2

Rank-240



Software



A Brief History of Tensors in MATLAB

- MATLAB (~1997)
  - Version 5.0 adds support for multidimensional arrays (MDAs)
- N-way Toolbox (<1997)
  - Extensive collection of functions and algorithms for analyzing multiway data
    - Handles constraints
    - Handles missing data
    - Etc.
  - Andersson & Bro, 2000

Few tools exist in other languages.

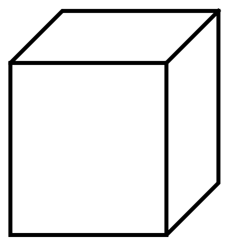
Tensor Toolbox

for MATLAB™

- Tensor Toolbox V1 (2005)
  - MATLAB classes for dense tensors, etc.
  - Extends MDA capabilities to include multiplication, matrization, etc.
- Tensor Toolbox V2 (2006)
  - Adds support for sparse and structured tensors
  - Performance enhancements
  - 1000+ registered users

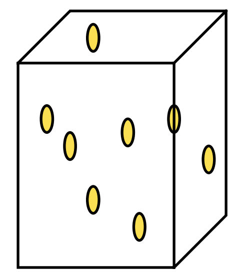
Tensor Toolbox V2.0 supports 4 types of tensors

Dense Tensors `tensor`



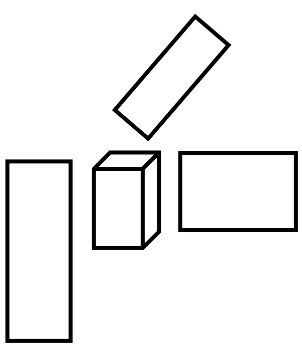
- Extends MATLAB's native MDA capabilities
- Can be converted to a matrix and vice versa

Sparse Tensors `sptensor`



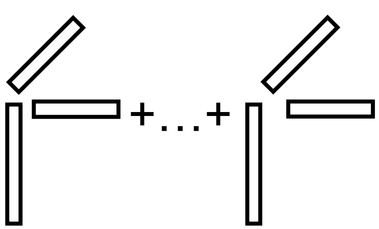
- Unique to Tensor Toolbox
- Can be converted to a (sparse) matrix and vice versa
- Effort to choose suitable representation
- Efficient functions for computation

Tucker Tensors `ttensor`



- Stores a tensor in decomposed form
- A different way to store a large-scale dense tensor
- Can do many operations in factored form

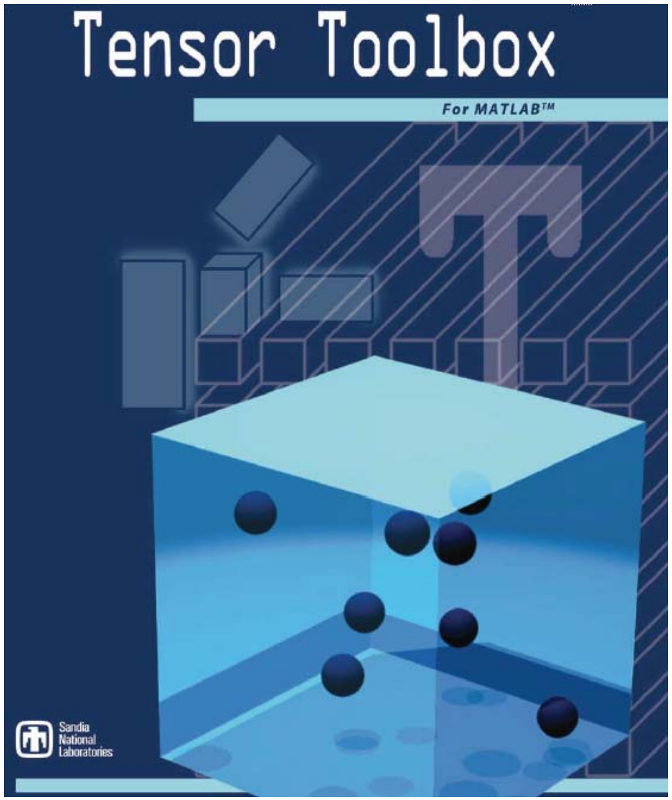
Kruskal Tensors `ktensor`



- Stores a tensor as sum of rank-1 tensors
- A different way to store a large-scale dense tensor
- Can do many operations in factored form

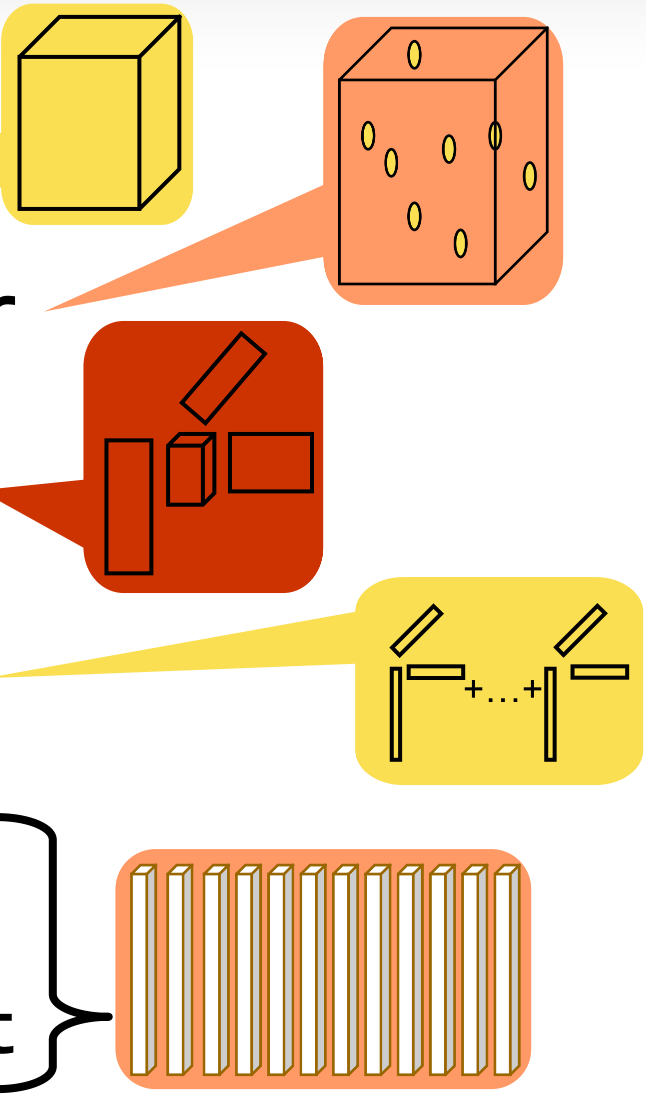
Sandia National Laboratories

Tensor Toolbox Classes

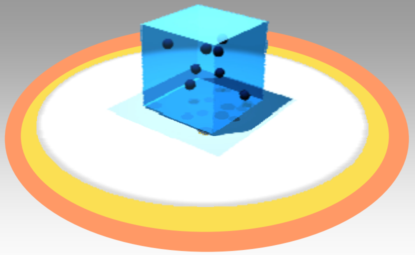


Available online.
Free for research and evaluation purposes.

- tensor
- sptensor
- ttensor
- ktensor
- tenmat
- sptenmat



Sandia National Laboratories



Examples of Other Work on Data Mining with Tensors

- **Chatroom analysis** using Tucker and CP - Acar et al., 2005 & 2006
- **Handwritten digit analysis** - Elden and Savas, (check date)
- **Window-based tensor analysis (WTA)** on streaming data - Sun, Papadimitriou, and Yu, 2006
- **Dynamic tensor analysis (DTA)** - Sun, Tao, and Papadimitriou, 2006
- **Multi-way clustering on relational graphs** - Banerjee, Basu, and Merugu, 2007
- PARAFAC and NN-PARAFAC for **Enron email analysis** - Bader, Berry, Browne, 2007
- All of the above used the Tensor Toolbox, but there are many other papers in this area starting in 2005.
- Related applications date back much further: chemometrics, EEG analysis, signal processing, etc.

Sandia National Laboratories

Acknowledgements/References

- Survey papers on tensors
  - TGK and Brett W. Bader. **Tensor decompositions and applications**. Submitted, Aug 2007.
  - TGK. **Multilinear operators for higher-order decompositions**. Tech. Rep., Apr 2006.
- Tensor Toolbox
  - Brett W. Bader and TGK. **Efficient MATLAB computations with sparse and factored tensors**. *SIAM J. Scientific Computing*, to appear.
  - Brett W. Bader and TGK. **Algorithm 862: MATLAB tensor classes for fast algorithm prototyping**. *ACM Trans. Mathematical Software*, Dec 2006.
- TOPHITS
  - TGK, Brett W. Bader, and Joseph P. Kenny. **Higher-order web link analysis using multilinear algebra**. In *ICDM 2005.*
  - TGK and Brett Bader. **The TOPHITS model for higher-order web link analysis**. In *Workshop on Link Analysis, Counterterrorism and Security*, 2006.
- Cross-Language IR with PARAFAC2
  - Peter A. Chew, Brett W. Bader, TGK, and Ahmed Abdelali. **Cross-language information retrieval using PARAFAC2**. In *KDD '07*
- Tutorial
  - Christos Faloutsos, TGK, Jimeng Sun, **Mining Large Time-evolving Data Using Matrix and Tensor Tools** at SDM07, SIGMOD07, ICML07, KDD07.

Sandia National Laboratories

Thank you.
Questions?

Tammy Kolda
tgkolda@sandia.gov

Sandia National Laboratories