SAND2015-5146C

Illustration by Chris Brigman

Survey: Optimization Challenges in Tensor Decomposition

Tamara G. Kolda

Sandia National Laboratories

Livermore, CA

26th Biennial Numerical Analysis Conference – Glasgow

June 23, 2015

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Acknowledgements

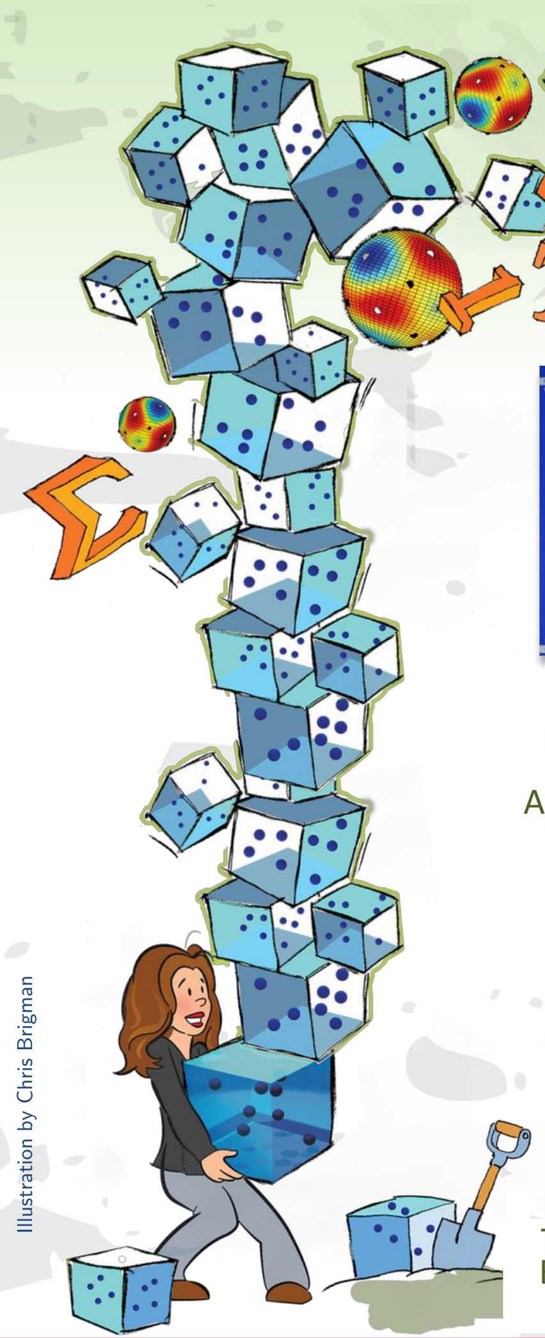
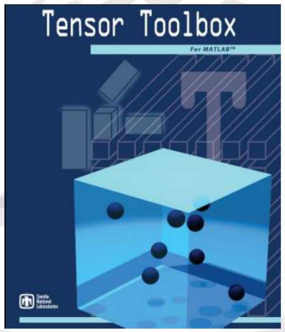


Illustration by Chris Brigman

Kolda and Bader, Tensor Decompositions and Applications, *SIAM Review*, 2009



Tensor Toolbox for MATLAB
Bader, Kolda, Acar, Dunlavy, and others

<u>Co-authors</u>

- Evrim Acar (Univ. Copenhagen*)
- Woody Austin (Univ. Texas Austin*)
- Brett Bader (Digital Globe*)
- Grey Ballard (Sandia)
- Eric Chi (NC State Univ.*)
- Danny Dunlavy (Sandia)
- Sammy Hansen (IBM*)
- Joe Kenny (Sandia)
- Jackson Mayo (Sandia)
- Morten Mørup (Denmark Tech. Univ.)
- Todd Plantenga (FireEye*)
- Martin Schatz (Univ. Texas Austin*)
- Teresa Selee (GA Tech Research Inst.*)
- Jimeng Sun (GA Tech)

Plus many more collaborators for workshops, tutorials, etc.

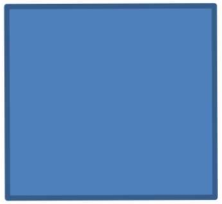
** = Worked for Sandia at some point*

A Tensor is an d-Way Array

Vector
d = 1

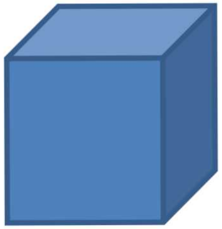
$\mathbf{a}$

Matrix
d = 2



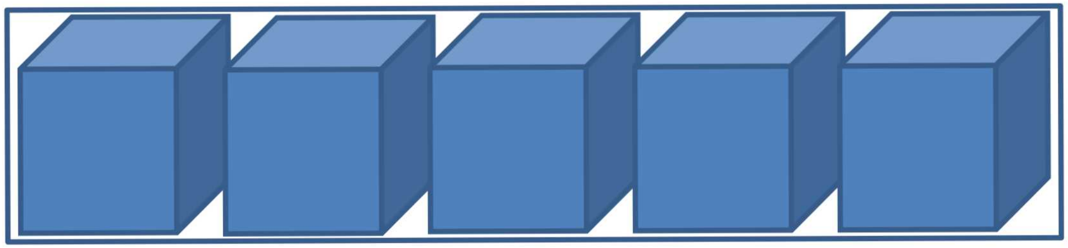
$\mathbf{A}$

3rd-Order Tensor
d = 3



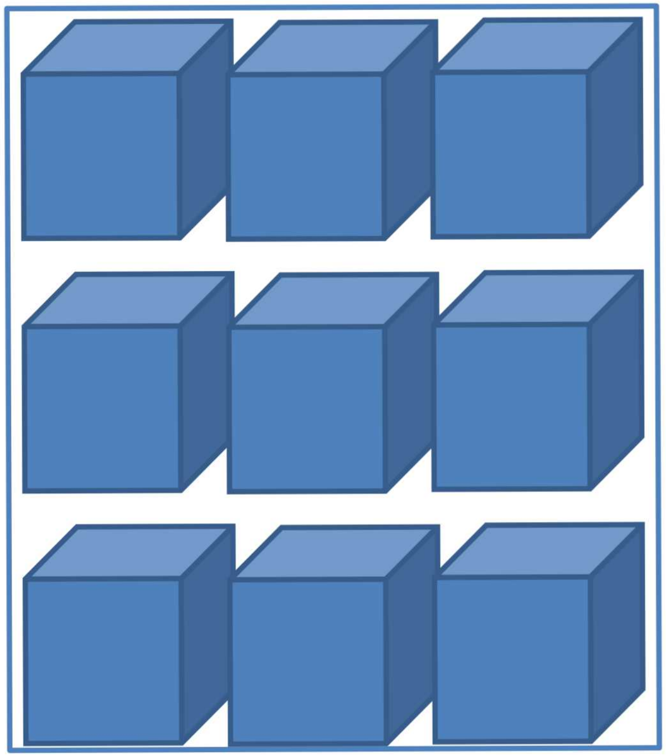
$\mathcal{A}$

4th-Order Tensor
d = 4



$\mathcal{A}$

5th-Order Tensor
d = 5

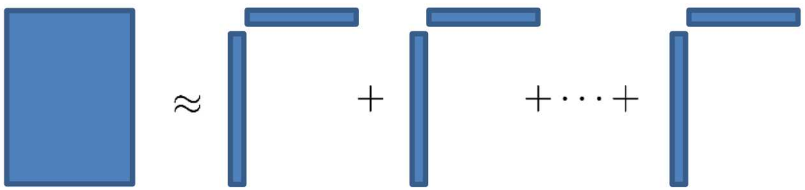


$\mathcal{A}$

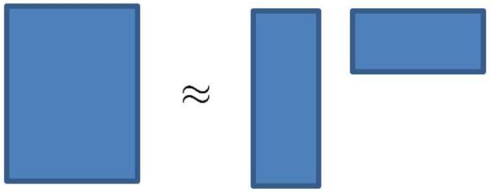
Tensor Decompositions are the New Matrix Decompositions

Singular value decomposition (SVD), eigen-decomposition (EVD), nonnegative matrix factorization (NMF), sparse SVD, etc.

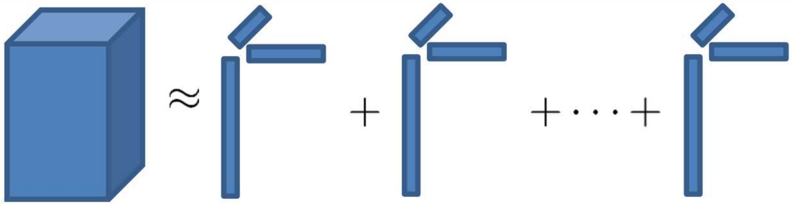
Viewpoint 1: Sum of outer products, useful for interpretation



Viewpoint 2: High-variance subspaces, useful for compression

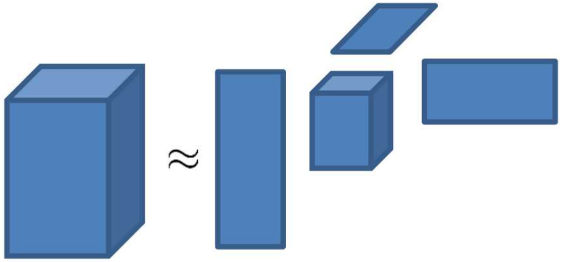


CP Model: Sum of d-way outer products, useful for interpretation



CANDECOMP, PARAFAC, Canonical Polyadic, CP

Tucker Model: Project onto high-variance subspaces to reduce dimensionality

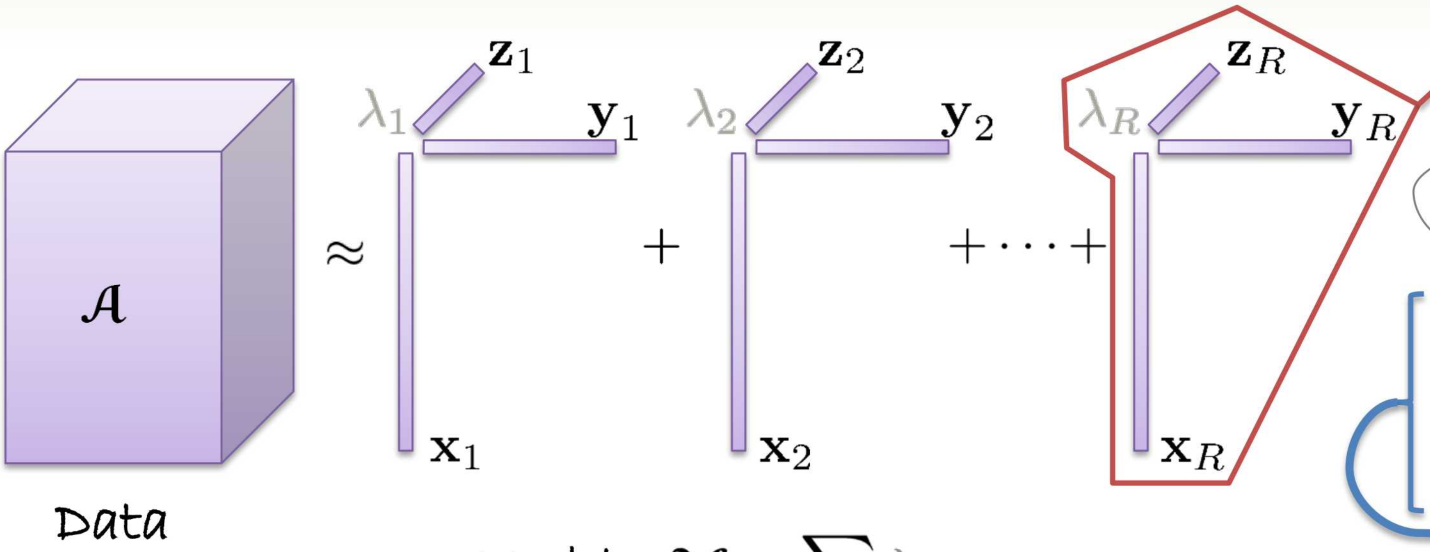


HO-SVD, Best Rank-(R1,R2,…,RN) decomposition

Other models for compression include hierarchical Tucker and tensor train.

CP: Sum of Outer Products

CANDECOMP/PARAFAC or canonical polyadic (CP) Model



Optional Weights

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 & \cdots & \lambda_R \end{bmatrix}^{\mathsf{T}}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \cdots & \mathbf{x}_R \end{bmatrix}$$
$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1 & \cdots & \mathbf{y}_R \end{bmatrix}$$
$$\mathbf{Z} = \begin{bmatrix} \mathbf{z}_1 & \cdots & \mathbf{z}_R \end{bmatrix}$$

Factor Matrices

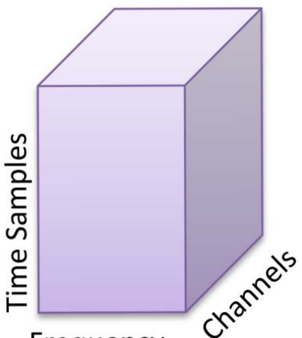
Model: $\mathcal{M} = \sum_r \lambda_r \, \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$

$$\boxed{\min_{\mathcal{M}} \sum_{ijk} (x_{ijk} - m_{ijk})^2 \quad \text{subject to} \quad m_{ijk} = \sum_r \lambda_r \, x_{ir} \, y_{jr} \, z_{kr}}$$

Key references: Hitchcock, 1927; Harshman, 1970; Carroll and Chang, 1970

Tensor Factorization "Sorts Out" Comingled Data

Data measurements are recorded at multiple sites (channels) over time. The data is transformed via a continuous wavelet transform.

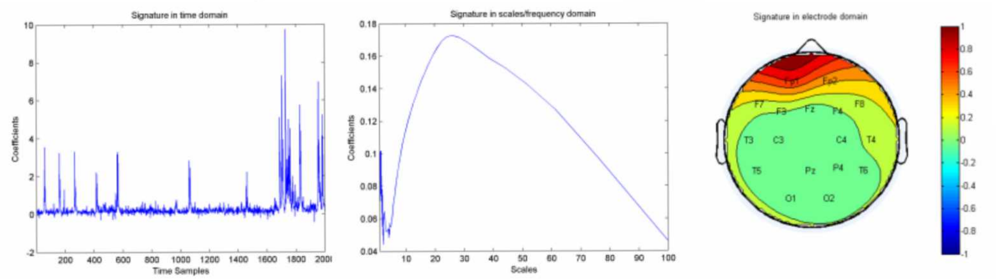


$$\mathcal{A} = \mathbf{x}_1 \circ \mathbf{y}_1 \circ \mathbf{z}_1 + \mathbf{x}_2 \circ \mathbf{y}_2 \circ \mathbf{z}_2 + \mathcal{E}$$

Acar, Bingol, Bingol, Bro and Yener, *Bioinformatics*, 2007

Eye Artifact

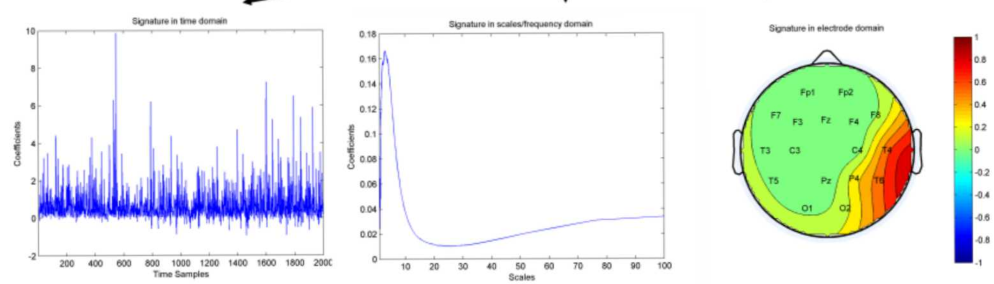
$$\mathcal{A} = \mathbf{x}_1 \circ \mathbf{y}_1 \circ \mathbf{z}_1 + \mathbf{x}_2 \circ \mathbf{y}_2 \circ \mathbf{z}_2 + \mathcal{E}$$



Time | Frequency | Channel

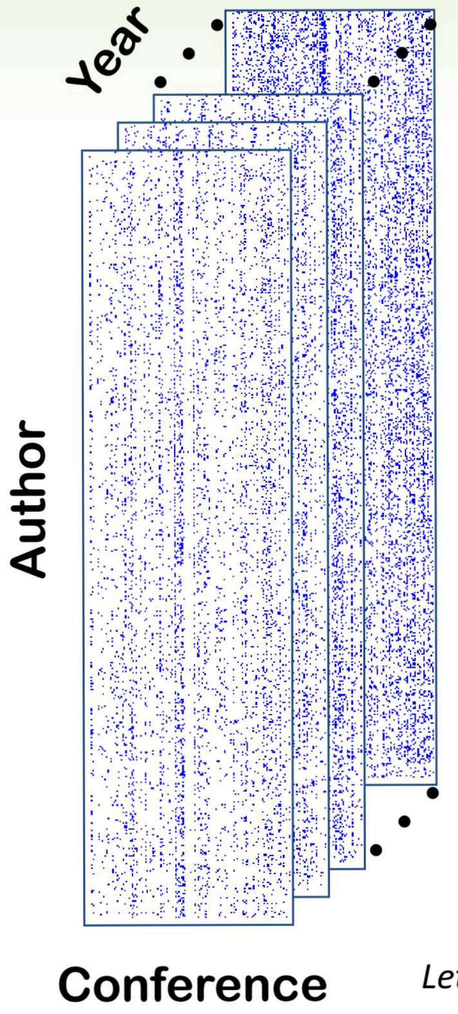
Seizure

$$\mathcal{A} = \mathbf{x}_1 \circ \mathbf{y}_1 \circ \mathbf{z}_1 + \mathbf{x}_2 \circ \mathbf{y}_2 \circ \mathbf{z}_2 + \mathcal{E}$$



Time | Frequency | Channel

Temporal Networks & Analysis



<u>Tasks</u>: Principal Components, Multidimensional Scaling, Clustering, Classification, Temporal Link Prediction

DBLP has data from 1936-2007
(used only "inproceedings" from 1991-2000)

Data	10 Years: 1991-2000
# Authors (min 10 papers)	7108
# Conferences	1103
Links	113k (0.14% dense)

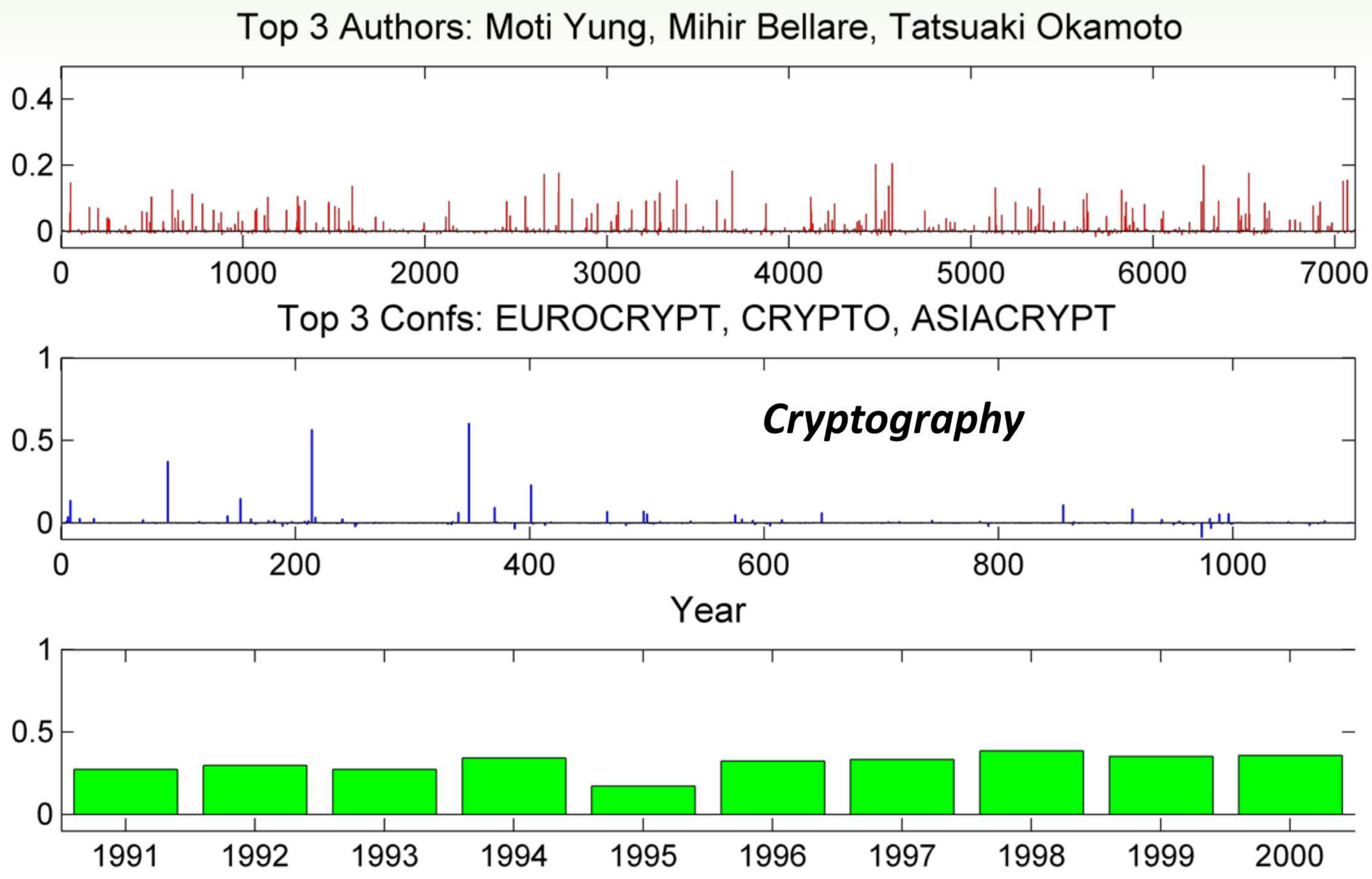
c_{ijk} = # papers by author i at conference j in year k

$$a_{ijk} = \begin{cases} \log(c_{ijk}) + 1 & \text{if } c_{ijk} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Let's look at some components sorted by size from a 50-component (R=50) factorization…

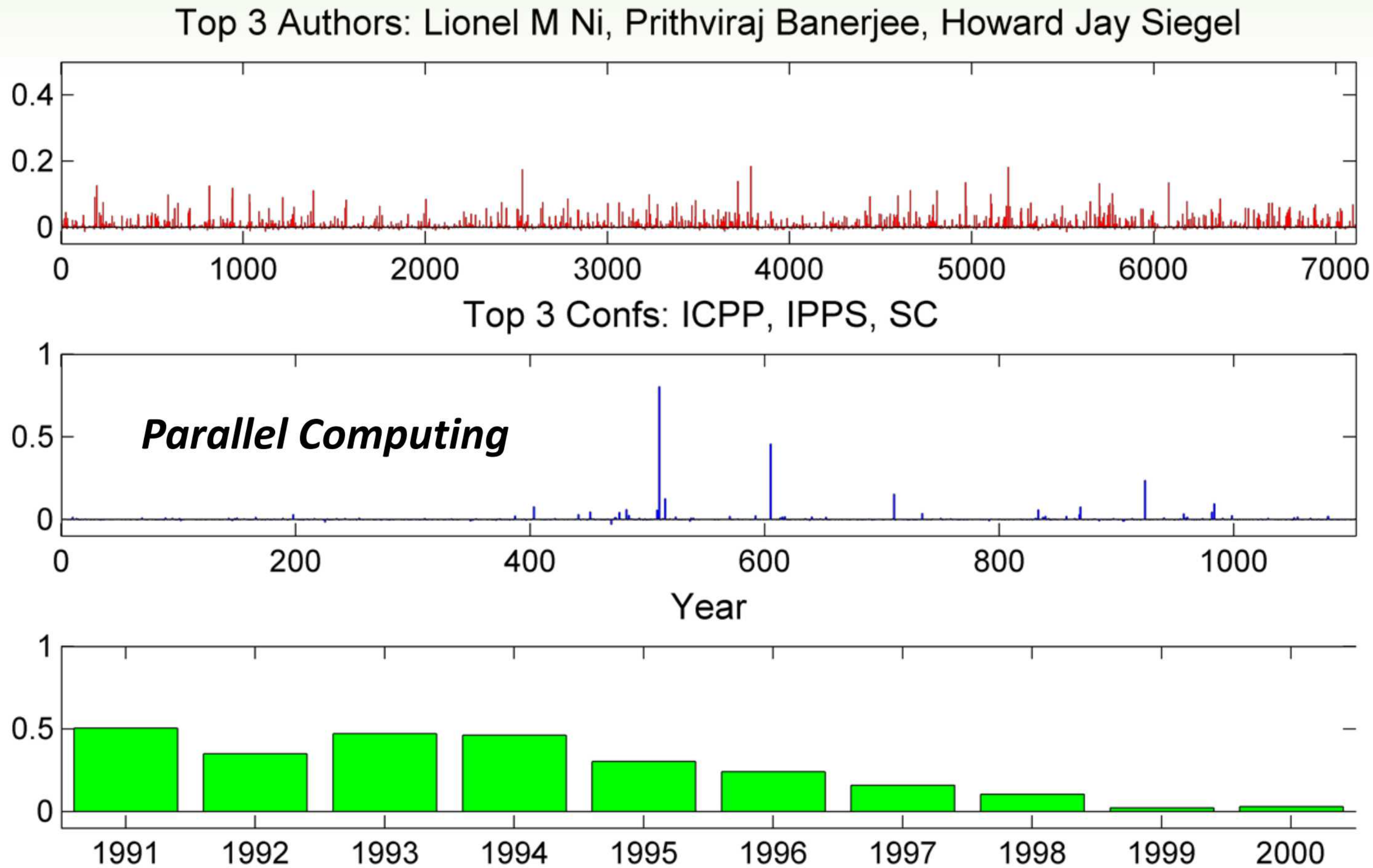
Acar, Dunlavy, & Kolda, Temporal Link Prediction using Matrix and Tensor Factorizations, *ACM TKDD*, 2010

DBLP Component #30 (of 50)



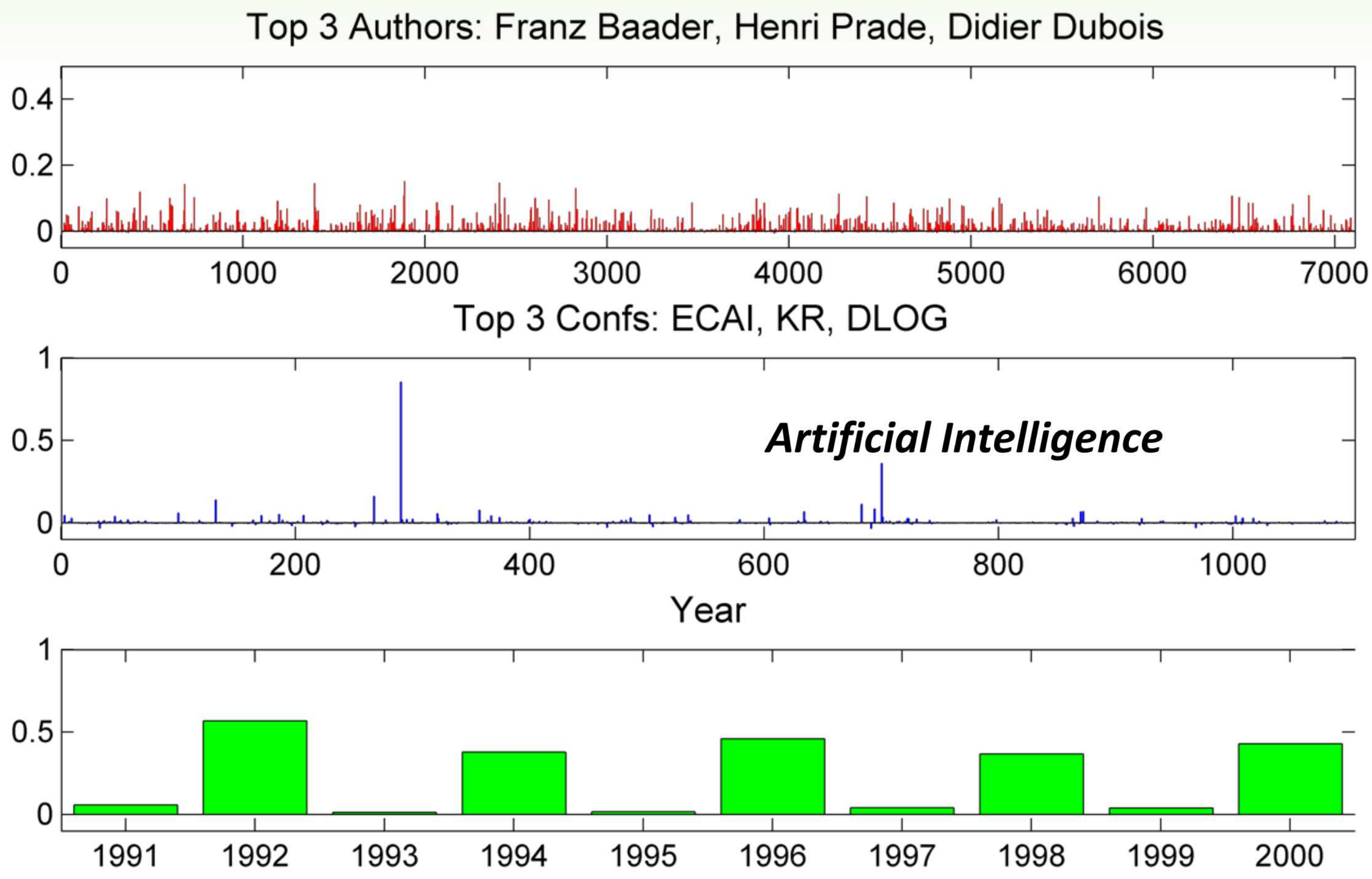
Acar, Dunlavy, & Kolda, Temporal Link Prediction using Matrix and Tensor Factorizations, *ACM TKDD*, 2010

DBLP Component #19 (of 50)



Acar, Dunlavy, & Kolda, Temporal Link Prediction using Matrix and Tensor Factorizations, *ACM TKDD*, 2010

DBLP Component #43 (of 50)

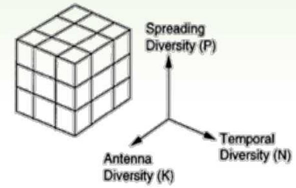


Acar, Dunlavy, & Kolda, Temporal Link Prediction using Matrix and Tensor Factorizations, *ACM TKDD*, 2010

Tensor Factorizations have Numerous Applications

- Modeling fluorescence excitation-emission data (chemometrics)
- Signal processing
- Brain imaging (e.g., fMRI) data
- Network analysis and link prediction
- Image compression and classification; texture analysis
- Text analysis, e.g., multi-way LSI
- Approximating Newton potentials, stochastic PDEs, etc.
- Collaborative filtering
- Higher-order graph/image matching

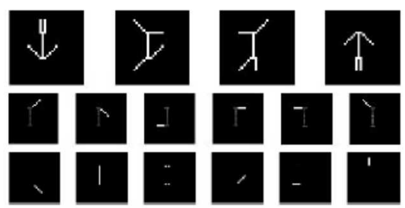
Furukawa, Kawasaki, Ikeuchi, and Sakauchi, *EGRW '02*



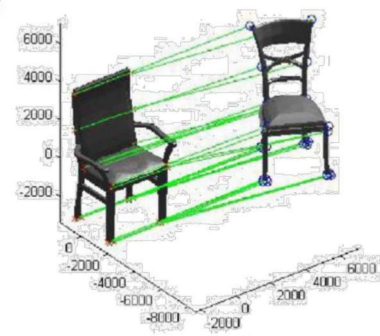
Sidiropoulos, Giannakis, Bro, *IEEE Trans. Signal Processing,* 2000

$$\mathcal{L}(x,t,\omega;u) = f(x,t,\omega) \quad (x,t) \in \mathcal{D} \times [0,T]$$
$$\mathcal{B}(x,t,\omega;u) = g(x,t) \quad (x,t) \in \partial\mathcal{D} \times [0,T]$$
$$\mathcal{I}(x,0,\omega;u) = h(x,\omega) \quad x \in \mathcal{D},$$

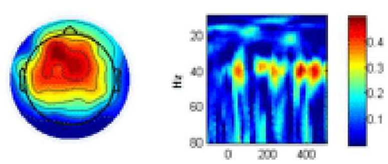
Doostan, Iaccarino, and Etemadi, *J. Computational Physics*, 2009



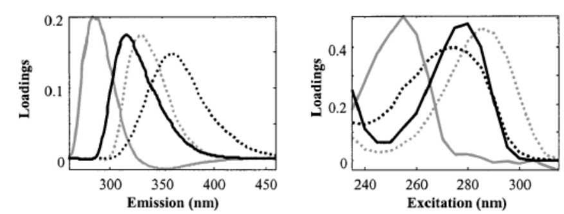
Hazan, Polak, and Shashua, *ICCV 2005*



Duchenne, Bach, Kweon, Ponce, *TPAMI 2011*

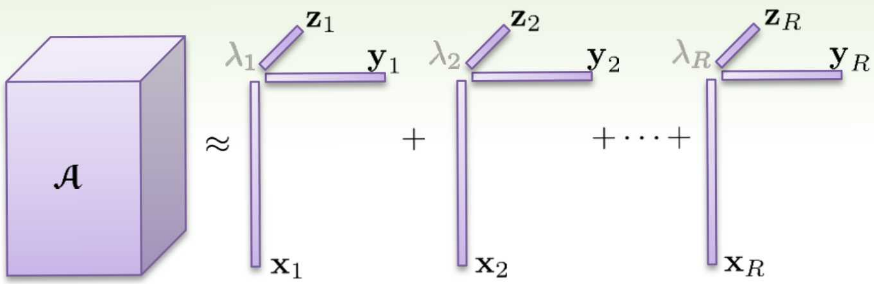


ERPWAVELAB by Morten Mørup



Andersen and Bro, *J. Chemometrics*, 2003

CP-ALS: Fitting CP via Alternating Least Squares



Convex (linear least squares) subproblems can be solved exactly
+
Structure makes easy inversion

$$f(\mathbf{X},\mathbf{Y},\mathbf{Z}) = \sum_{ijk}\Big(a_{ijk} - \sum_r x_{ir}\, y_{jr}\, z_{kr}\Big)^2$$

Repeat until convergence:

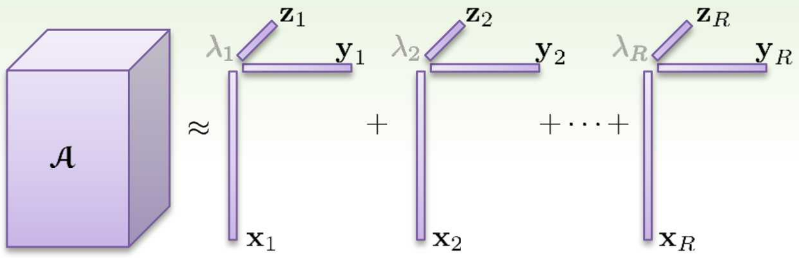
Step 1: $\displaystyle \min_{\mathbf{X}} \sum_{ijk}\Big(a_{ijk} - \sum_r x_{ir}\, y_{jr}\, z_{kr}\Big)^2$

Step 2: $\displaystyle \min_{\mathbf{Y}} \sum_{ijk}\Big(a_{ijk} - \sum_r x_{ir}\, y_{jr}\, z_{kr}\Big)^2$

Step 3: $\displaystyle \min_{\mathbf{Z}} \sum_{ijk}\Big(a_{ijk} - \sum_r x_{ir}\, y_{jr}\, z_{kr}\Big)^2$

Harshman, 1970; Carroll & Chang, 1970

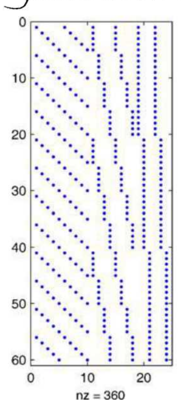
CP-OPT: Fitting CP via all-at-once optimization



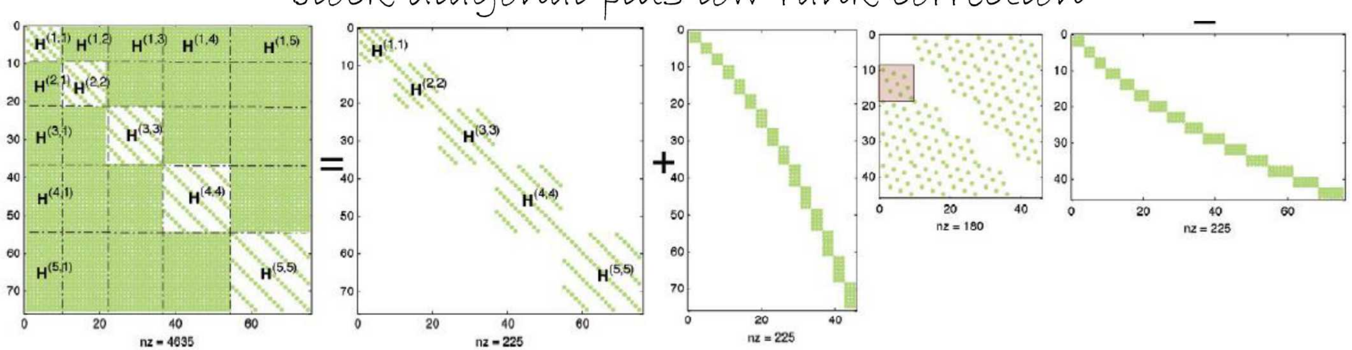
$$f(\mathbf{X},\mathbf{Y},\mathbf{Z}) = \sum_{ijk}\Big(a_{ijk} - \sum_{r} x_{ir}\, y_{jr}\, z_{kr}\Big)^2$$

- CP-OPT (Acar et al.): 1st-order method, better accuracy than ALS when R is too big
- CP-NLS (Paatero, Tomasi & Bro): Damped Gauss-Newton, accurate but slow
- CP-Newton (Phan et al.): Newton method, superior to CP-OPT for high order

Structured Jacobian

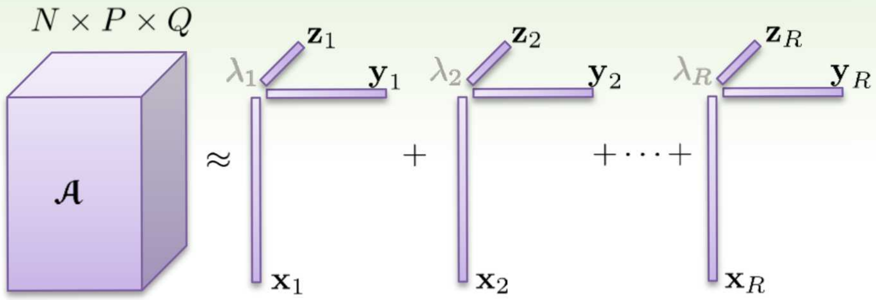


Structured Hessian can be written as block diagonal plus low-rank correction



Paatero 1997; Tomasi & Bro 2005, 2006; Acar, Dunlavy, & Kolda 2011; Phan, Tichavský, & Cichocki 2013

Challenges for CP Optimization Problem



variables = $R(N + P + Q + 1)$
data points = NPQ

Rank = minimal R to exactly reproduce tensor

- **Nonconvex:** Polynomial optimization problem ⇒ *Initialization matters*
- **Permutation and scaling ambiguities:** Can reorder the r's and arbitrarily scale vectors within each component so long as the product of the scaling is 1 ⇒ *May need regularization, # independent vars = $R(N+P+Q-2)$*
- **Rank unknown:** Determining the "rank" R that yields exact fit is NP-hard (Håstad 1990, Hillar & Lim 2009) ⇒ *No easy solution, need to try many*
- **Low-rank?** Best "low-rank" factorization may not exist (Silva & Lim 2006) ⇒ *Need bounds on components* $\|\lambda_r \, \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r\| = |\lambda_r| \, \|\mathbf{x}_r\| \, \|\mathbf{y}_r\| \, \|\mathbf{z}_r\|$
- **Not nested:** Best rank-(R-1) factorization may not be part of best rank-R factorization (Kolda 2001) ⇒ *Cannot use greedy algorithm*

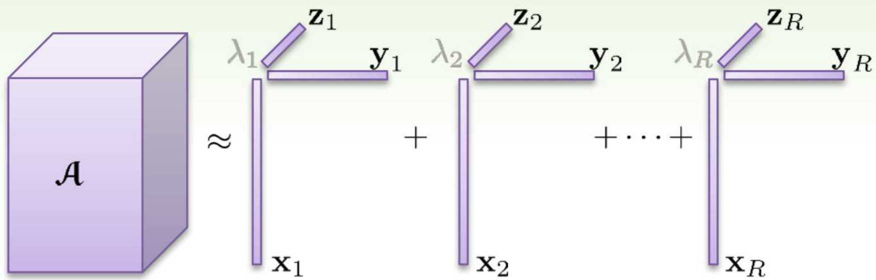
Example 9 x 9 x 9 Tensor of Unknown Rank

- Specific 9 x 9 x 9 tensor factorization problem
- Corresponds to being able to do fast matrix multiplication of two 3x3 matrices
- Rank is between 19 and 23

$x_{1,1,1} = 1$	$x_{4,2,1} = 1$	$x_{7,3,1} = 1$
$x_{1,4,2} = 1$	$x_{4,5,2} = 1$	$x_{7,6,2} = 1$
$x_{1,7,3} = 1$	$x_{4,8,3} = 1$	$x_{7,9,3} = 1$
$x_{2,1,4} = 1$	$x_{5,2,4} = 1$	$x_{8,3,4} = 1$
$x_{2,4,5} = 1$	$x_{5,5,5} = 1$	$x_{8,6,5} = 1$
$x_{2,7,6} = 1$	$x_{5,8,6} = 1$	$x_{8,9,6} = 1$
$x_{3,1,7} = 1$	$x_{6,2,7} = 1$	$x_{9,3,7} = 1$
$x_{3,4,8} = 1$	$x_{6,5,8} = 1$	$x_{9,6,8} = 1$
$x_{3,7,9} = 1$	$x_{6,8,9} = 1$	$x_{9,9,9} = 1$

Laderman 1976; Bini et al. 1979; Bläser 2003; Benson & Ballard, PPoPP'15

Opportunities for the CP Optimization Problem



k-rank(**X**) = maximum value k such that *any* k columns of **X** are linearly independent

- Factorization is **essentially unique** (i.e., up to permutation and scaling) under the condition the the sum of the factor matrix k-rank values is $\geq$ 2R + d – 1 (Kruskal 1977)

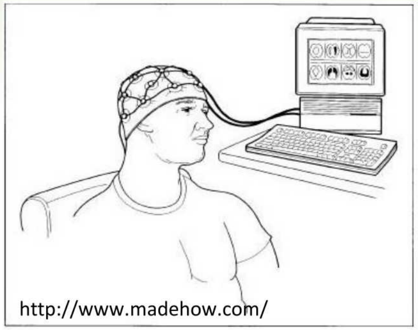
$$\text{k-rank}(\mathbf{X}) + \text{k-rank}(\mathbf{Y}) + \text{k-rank}(\mathbf{Z}) \geq 2R + 2$$

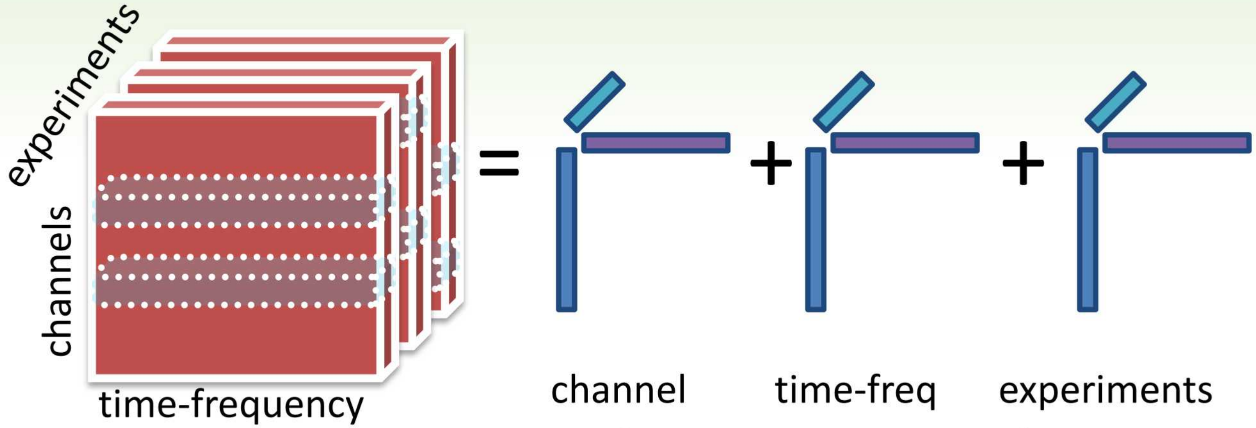
- If $R \ll N,P,Q$, then can use **compression** to reduce dimensionality before solving CP model (CANDELINC: Carroll, Pruzansky, and Kruskal 1980)
- Efficient **sparse kernels** exist (Bader & Kolda, SISC 2007)
- Recommended as optimization test problems with tunable difficulty
  - Choosing higher order (illustration for order d=3)
  - Choosing larger dimensions
  - Creating higher collinearity in the factors
  - Adding constraints for nonnegativity, sparseness, etc.

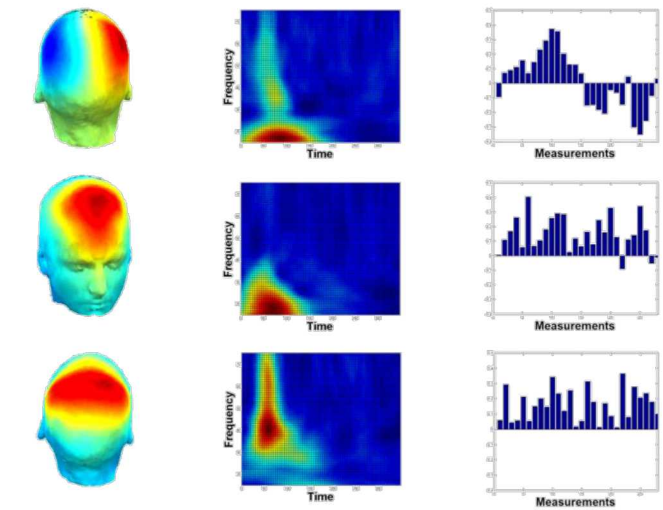
Collinear

$$\cos(\Theta(\mathbf{x}_r, \mathbf{x}_s)) \approx 0$$

Tensor Factorizations with Missing Data?







Biomedical signal processing

- EEG (electroencephalogram) signals can be recorded using electrodes placed on the scalp
- **Missing data problem** occurs when…
  - Electrodes get loose or disconnected, causing the signal to be unusable
  - Different experiments have over-lapping but not identical channels

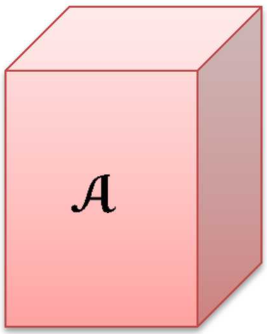
Can we still do this calculation if data are missing?

Acar, Dunlavy, Kolda, Mørup, Scalable Tensor Factorizations with Missing Data, SDM'10

The Missing Data Problem

Standard Problem:

Given all entries of tensor A, find factor matrices X, Y, and Z such that…



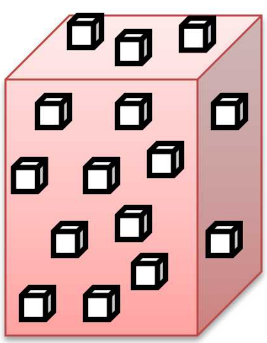
$$\min_{\mathcal{M}} \sum_{ijk} (a_{ijk} - m_{ijk})^2$$

Typically formulated as a least squares problem

$$\text{subject to} \quad m_{ijk} = \sum_r x_{ir}\, y_{jr}\, z_{kr}$$

Missing Data Problem:

Given *a subset of the entries* of A, find factor matrices X, Y, and Z such that…



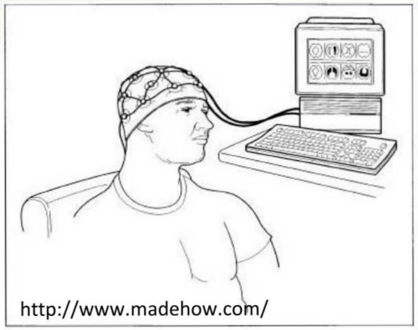
$$\min_{\mathcal{M}} \sum_{ijk \in \Omega^c} (a_{ijk} - m_{ijk})^2$$

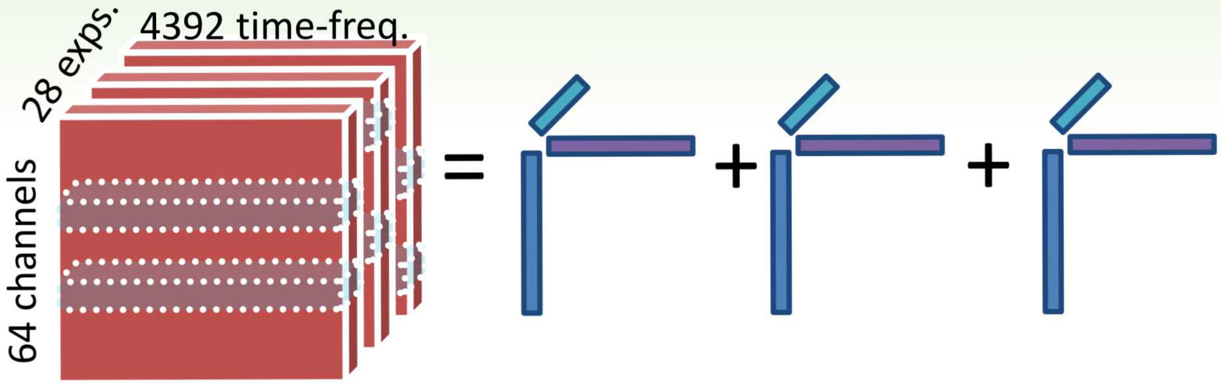
$\Omega =$ *subset of missing entries*

Not amenable to ALS, but straightforward with "all-at-once" optimization.
Key to statistical rank prediction.

Acar, Dunlavy, Kolda, Mørup, Scalable Tensor Factorizations with Missing Data, SDM'10

Brain dynamics can be captured even extensive missing channels

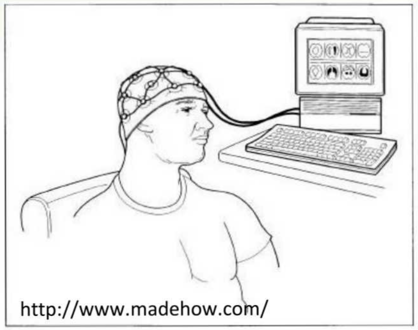


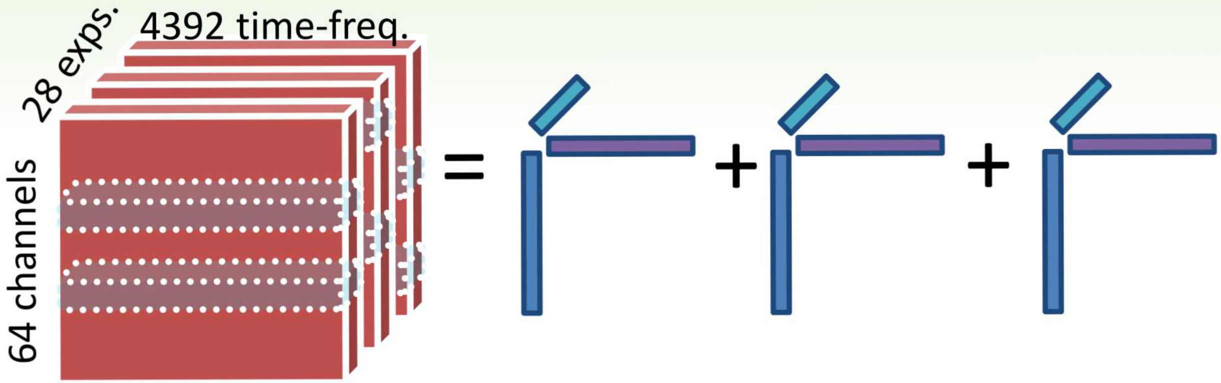


Number of Missing Channels	Replace Missing Entries with Mean
1	0.98
10	0.82
20	0.67
30	0.45
40	0.24

Acar, Dunlavy, Kolda, Mørup, Scalable Tensor Factorizations with Missing Data, SDM'10

Brain dynamics can be captured even extensive missing channels

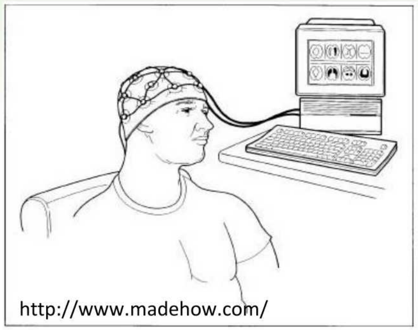


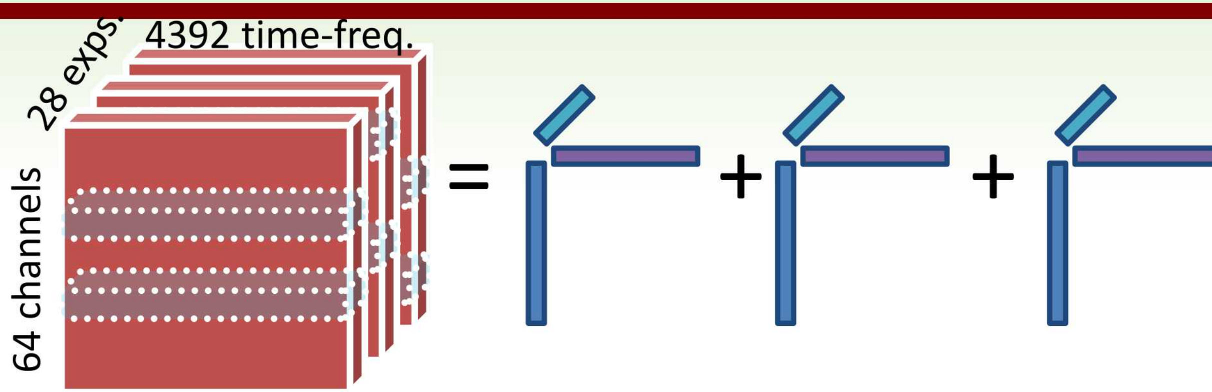


Number of Missing Channels	Replace Missing Entries with Mean	Ignore Missing Entries
1	0.98	1.00
10	0.82	0.98
20	0.67	0.95
30	0.45	0.89
40	0.24	0.65

Acar, Dunlavy, Kolda, Mørup, Scalable Tensor Factorizations with Missing Data, SDM'10

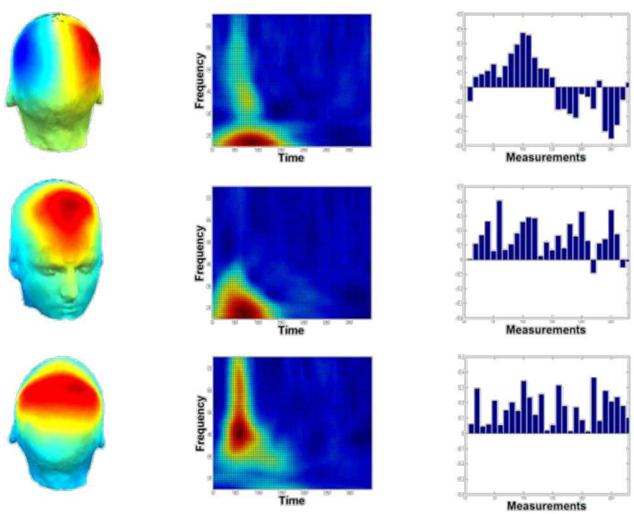
Brain dynamics can be captured even extensive missing channels





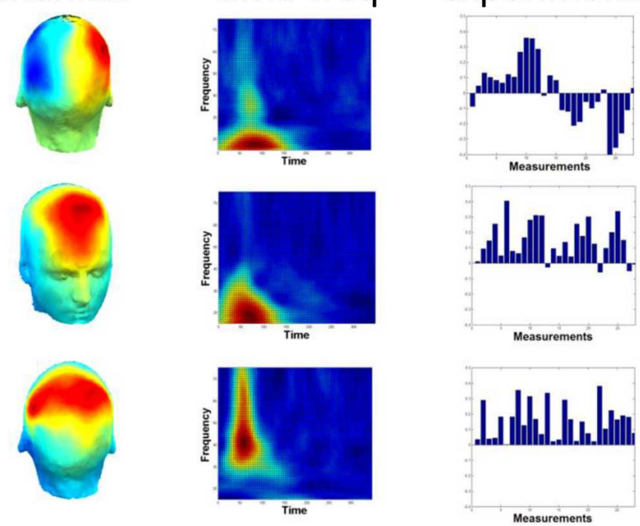
No Missing Data

channel | time-freq | experiments



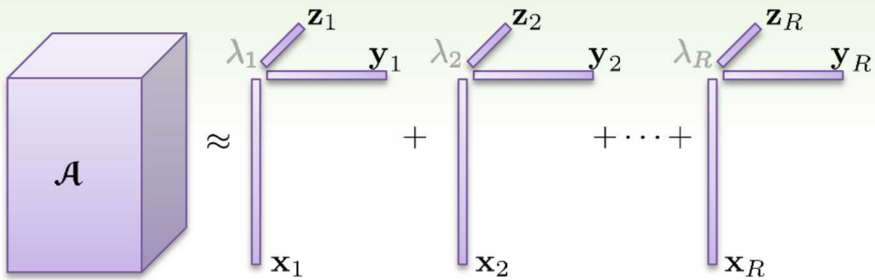
30 Chan./Exp. Missing

channel | time-freq | experiments



Acar, Dunlavy, Kolda, Mørup, Scalable Tensor Factorizations with Missing Data, SDM’10

Cross-Validation to Determine the Number of Components



- Create H holdout sets: $\Omega_1, \ldots, \Omega_H$
- For r=1,2,…
  - Train model for h =1,…,H

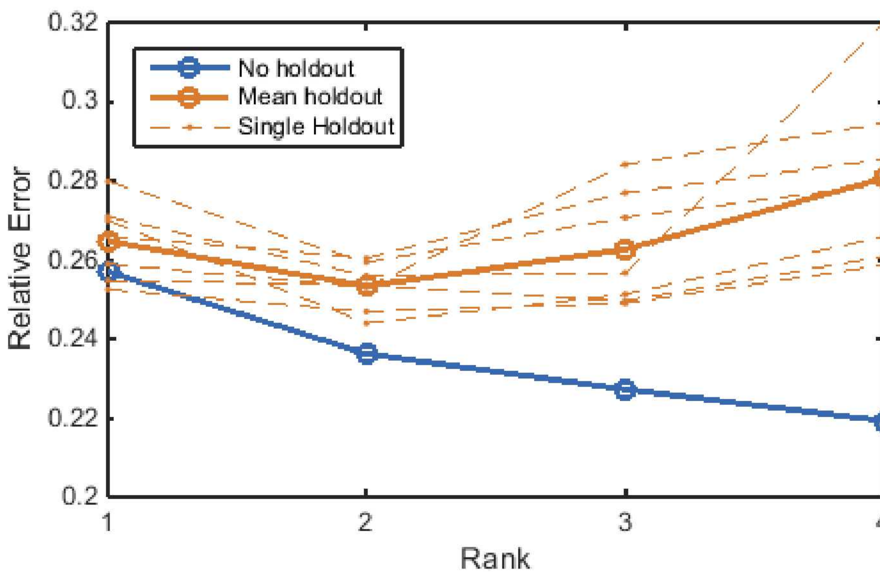
$$\mathcal{M}^{(hr)} = \arg\min_{\mathcal{M}} \sum_{ijk \in \Omega_h^c} (a_{ijk} - m_{ijk})^2$$

  - Compute error for h =1,…,H

$$e^{(hr)} = \sqrt{\frac{1}{|\Omega_h|} \sum_{ijk \in \Omega_h} (a_{ijk} - m_{ijk}^{(hr)})^2}$$

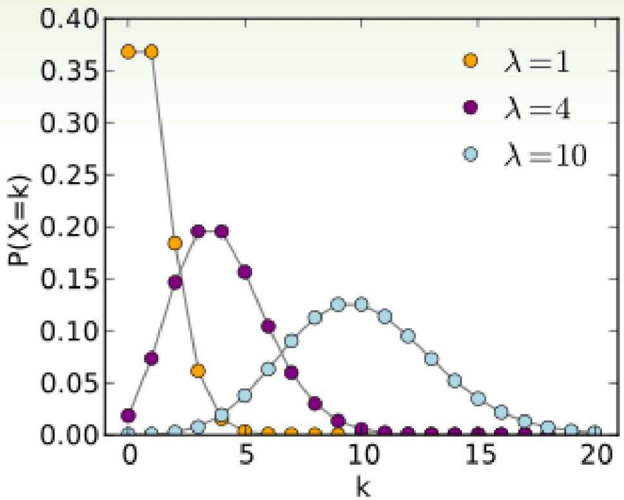
- Consider statistics on errors $e^{(hr)}$ to choose the ideal rank

Example: 10 x 10 x 10 tensor of rank-2 with component sizes of 1 and 0.1, with 25% noise. Can we tell the difference between the second small component and noise?

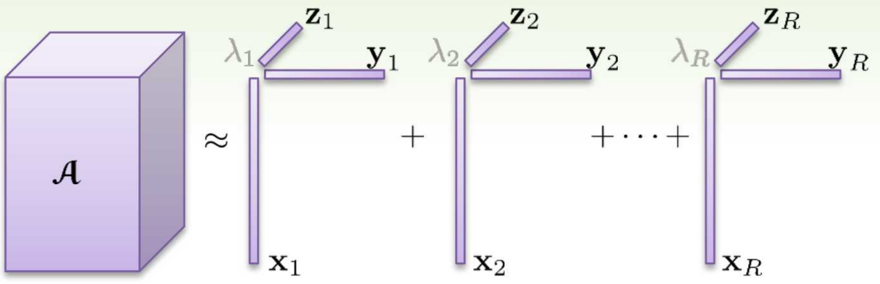


Austin and Kolda, Statistical Rank Determination for Tensor Factorizations, in progress

New "Stable" Approach: Poisson Tensor Factorization (PTF)



$$P(X = x) = \frac{\exp(-\lambda)\lambda^x}{x!}$$



$$m_{ijk} = \sum_r \lambda_r \, x_{ir} \, y_{jr} \, z_{kr}$$

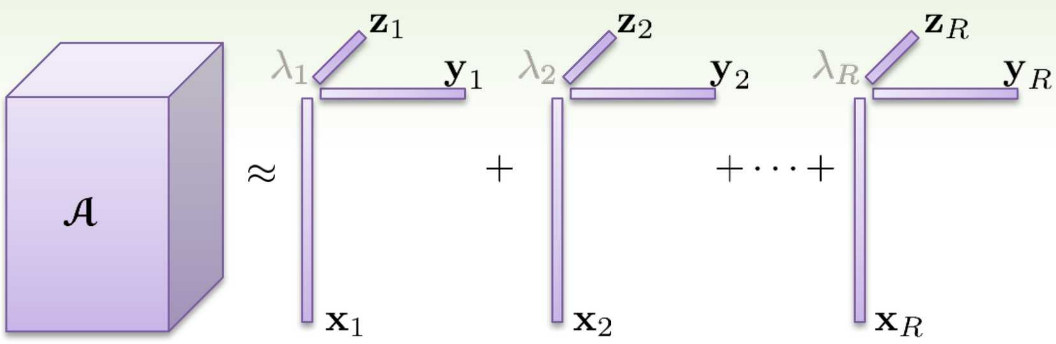
$$\boxed{a_{ijk} \sim \text{Poisson}(m_{ijk})}$$

Maximize this: $\text{likelihood}(\mathcal{M}) = \prod_{ijk} \frac{\exp(-m_{ijk}) \, m_{ijk}^{a_{ijk}}}{a_{ijk}!}$

By monotonicity of log, same as maximizing this: $\text{log-likelihood}(\mathcal{M}) = c - \sum_{ijk} m_{ijk} - a_{ijk} \log(m_{ijk})$

This objective function is also known as Kullback-Liebler (KL) divergence.
The factorization is automatically nonnegative.

Solving the Poisson Regression Problem



$$\mathcal{M} = \sum_r \lambda_r \, \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$$

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 & \cdots & \lambda_R \end{bmatrix}^{\mathsf{T}}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 & \cdots & \mathbf{x}_R \end{bmatrix}$$

$$\mathbf{Y} = \begin{bmatrix} \mathbf{y}_1 & \cdots & \mathbf{y}_R \end{bmatrix}$$

$$\mathbf{Z} = \begin{bmatrix} \mathbf{z}_1 & \cdots & \mathbf{z}_R \end{bmatrix}$$

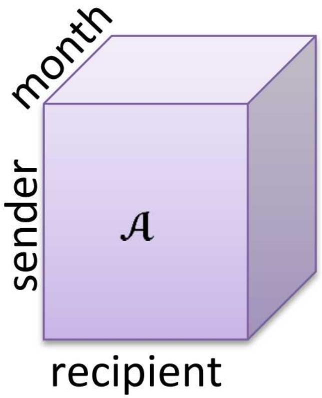
$$\boxed{\min_{\mathcal{M}} \sum_{ijk} m_{ijk} - a_{ijk} \log m_{ijk} \quad \text{subject to} \quad m_{ijk} = \sum_r \lambda_r \, x_{ir} \, y_{jr} \, z_{kr}}$$

- Highly nonconvex problem!
  - Assume R is given
- Alternating Poisson regression
  - Assume (d-1) factor matrices are known and solve for the remaining one
  - Multiplicative updates Typically assume data tensor A is sparse and have special methods for this
  - Newton or Quasi-Newton method

Chi & Kolda, SIMAX 2012; Hansen, Plantenga, & Kolda OMS 2015

PTF for Time-Evolving Social Network

Enron email data from FERC investigation.



Data	8540 Email Messages
# Months	28 (Dec'99 – Mar'02)
# Senders/Recipients	108 (>10 messages each)
Links	8500 (3% dense)

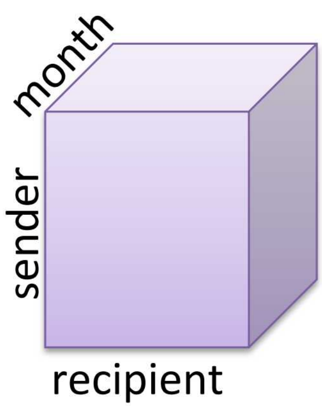
a_{ijk} = # emails from sender i to recipient j in month k

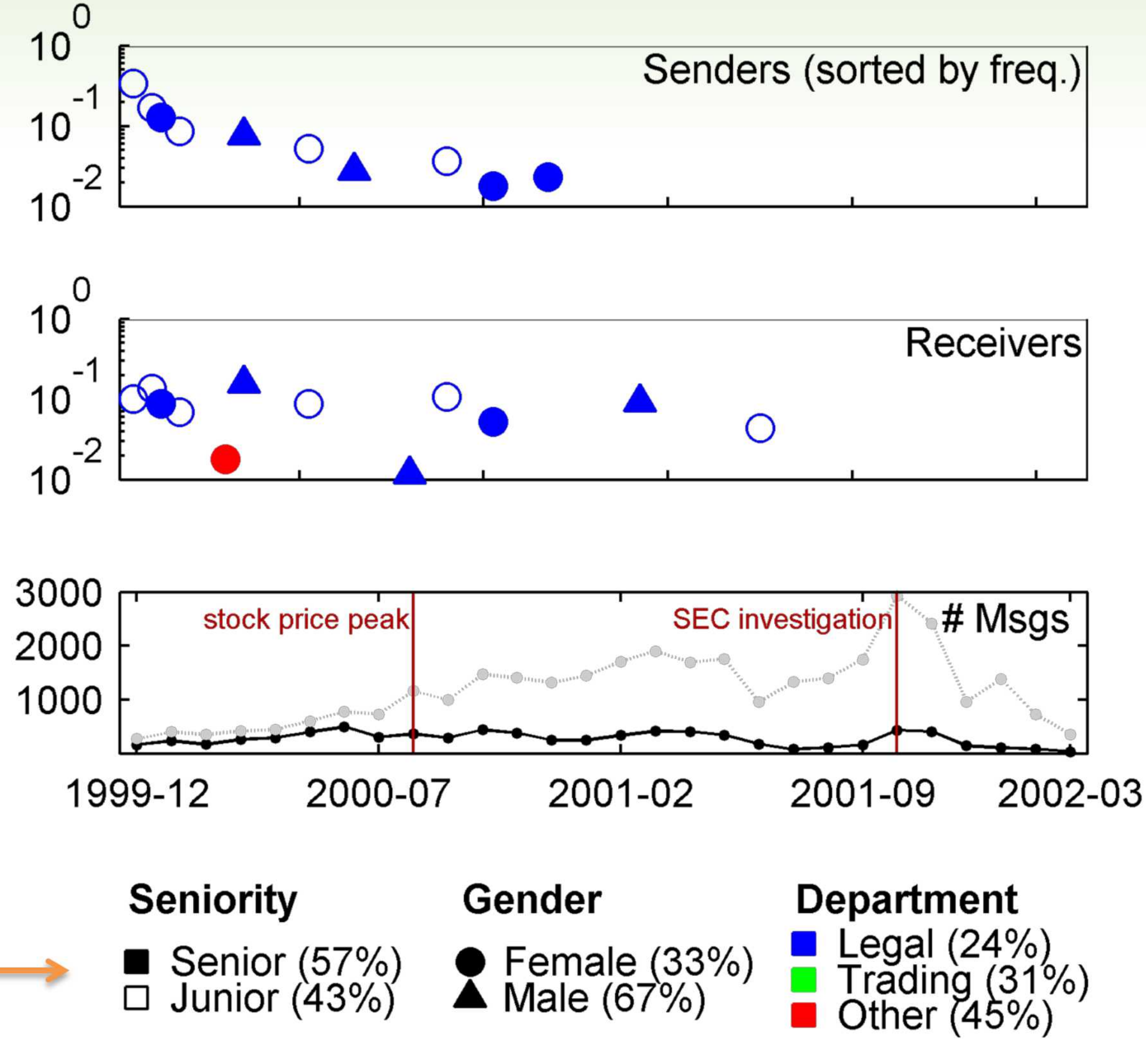
Let's look at some components from a 10-component (R=10) factorization, sorted by size…

Chi & Kolda, On Tensors, Sparsity, and Nonnegative Factorizations, SIMAX 2012

Enron Email Data (Component 1)

Legal Dept;
Mostly Female



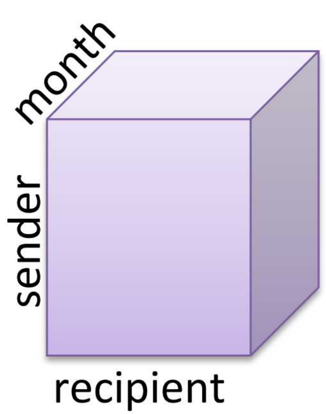


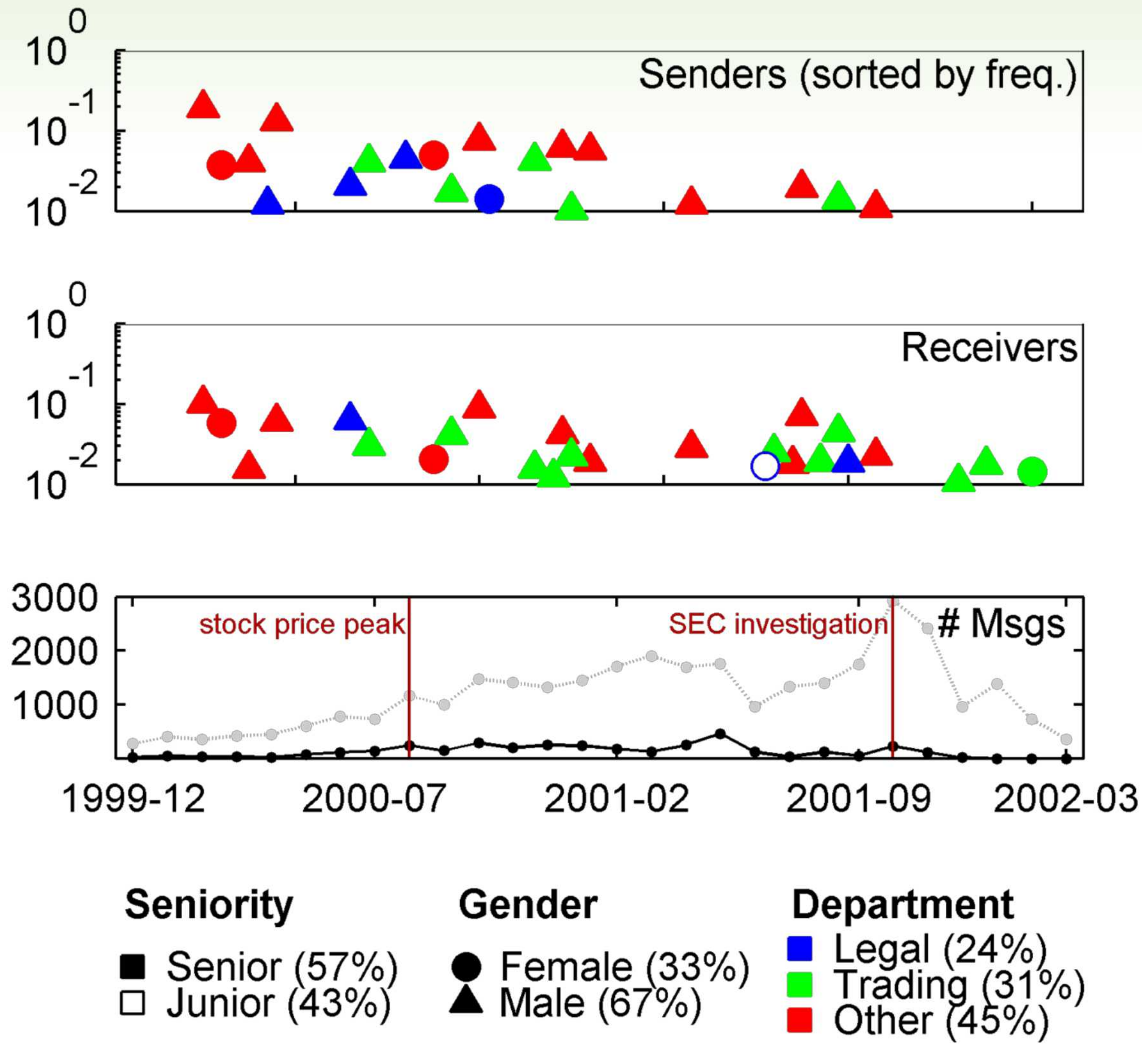
Each person labeled by Zhou et al. (2007)

Chi & Kolda, On Tensors, Sparsity, and Nonnegative Factorizations, SIMAX 2012

Enron Email Data (Component 3)

Senior;
Mostly Male

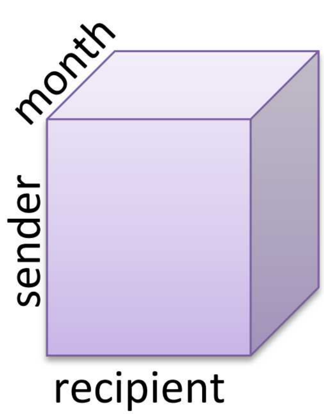


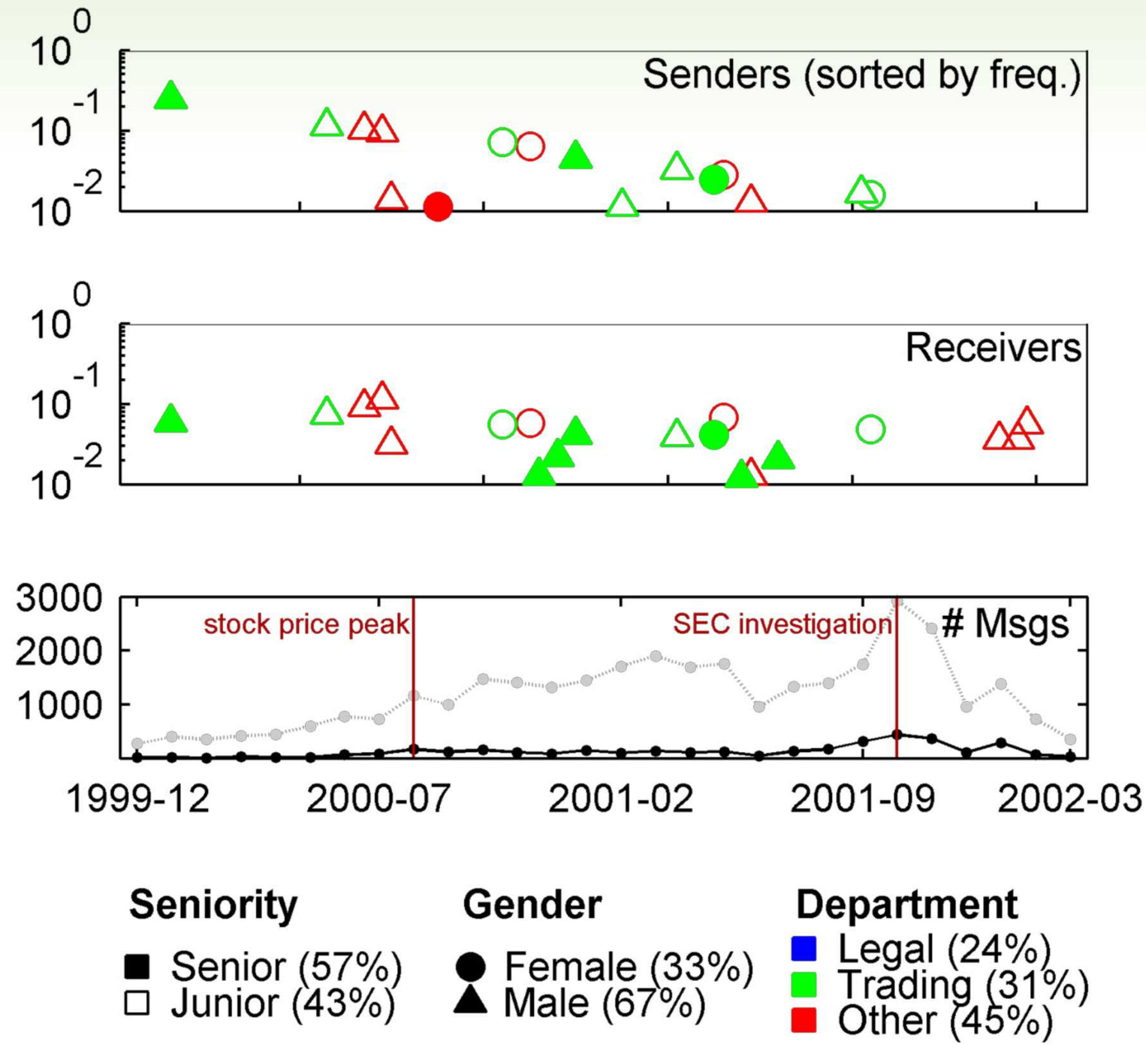


Chi & Kolda, On Tensors, Sparsity, and Nonnegative Factorizations, SIMAX 2012

Enron Email Data (Component 4)

Not Legal

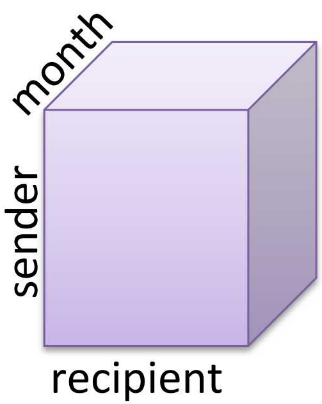


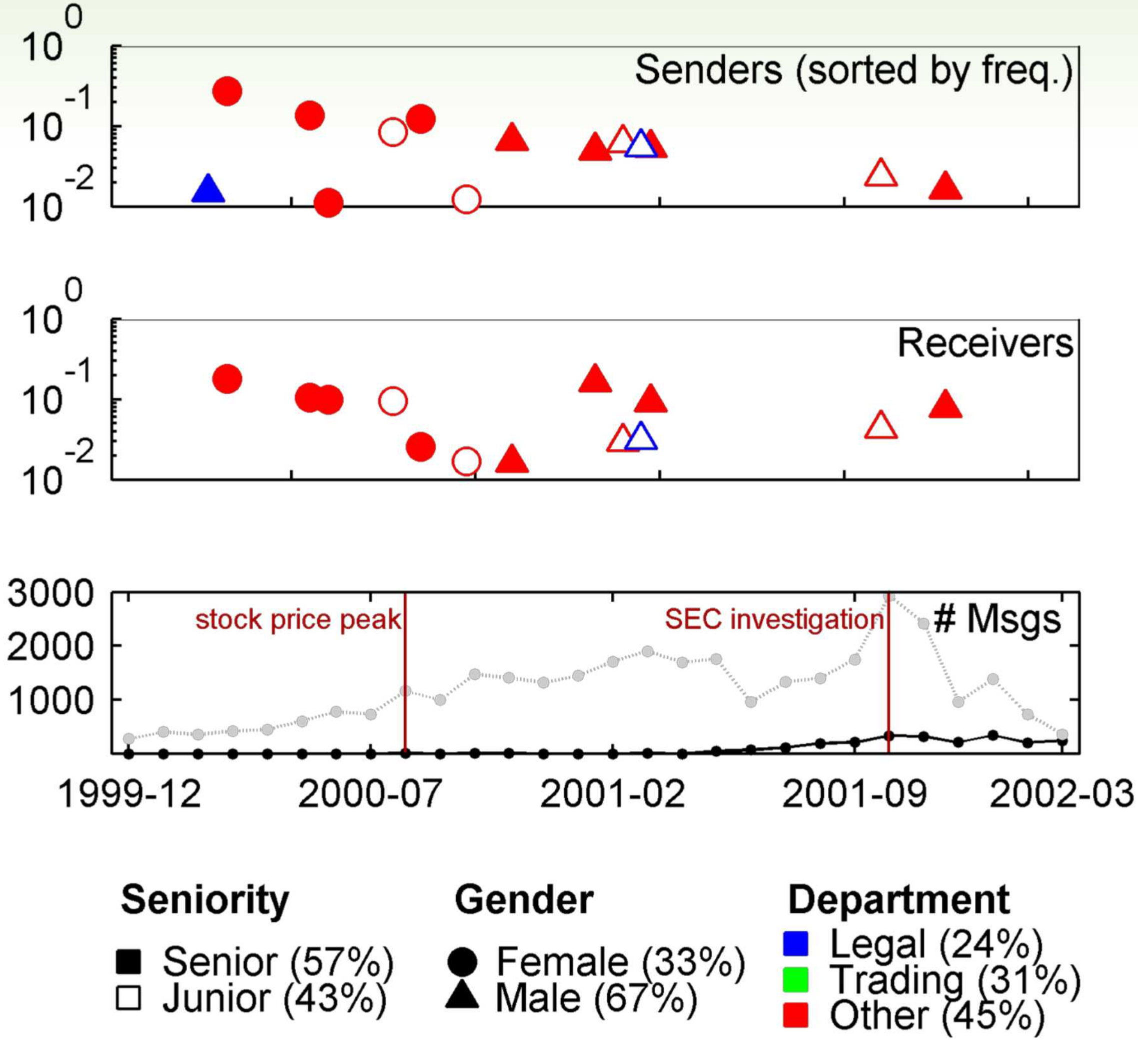


Chi & Kolda, On Tensors, Sparsity, and Nonnegative Factorizations, SIMAX 2012

Enron Email Data (Component 5)

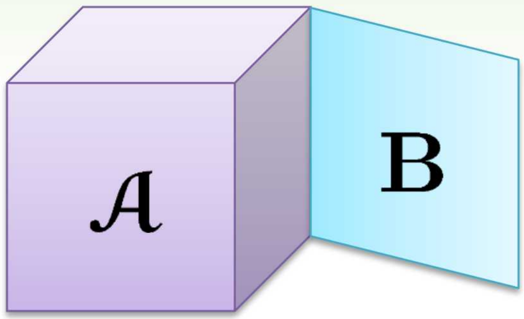
Other;
Mostly Female





Chi & Kolda, On Tensors, Sparsity, and Nonnegative Factorizations, SIMAX 2012

Coupled Factorizations



$$\mathcal{M} \approx \sum_r \lambda_r \, \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$$
$$\mathbf{B} \approx \mathbf{X}\mathbf{W}^{\mathsf{T}}$$

- Applications
  - Biology
    - Gene x Expression x Time
    - Gene x Function
  - Consumer information
    - Consumer x Purchase x Season
    - Consumer x Zip Code
- CMTF Toolbox (uses Tensor Toolbox)
  - Can do ALS or all-at-once optimization
  - Handles missing data

$$f(\mathbf{W}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}) = \frac{1}{2}\left\| \mathcal{A} - \sum_r \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r \right\|^2 + \frac{1}{2}\left\| \mathbf{B} - \mathbf{X}\mathbf{W}^{\mathsf{T}} \right\|^2$$

Acar, Dunlavy, Kolda, MLG'11; Acar et al., IEEE EMBC, 2013; Acar et al., BMC Bioinformatics, 2014

Symmetric Tensor Factorization

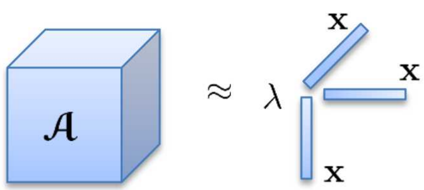
- d = number of modes or ways, N = size of each mode
- symmetric = entries invariant to permutation of indices

Symmetry for 3-way tensor (d = 3)

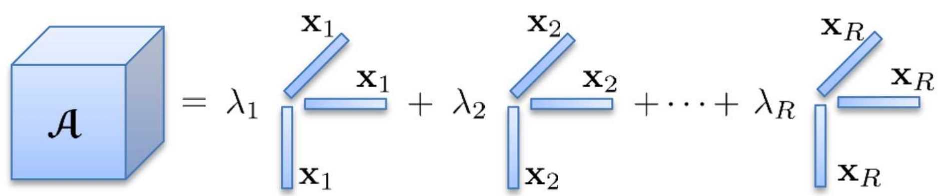
$$a_{ijk} = a_{ikj} = a_{jik} = a_{kij} = a_{jki} = a_{kji}$$
$$\text{for all } i, j, k \in \{1, 2, \ldots, N\}$$

$\Rightarrow$ N^d elements but only $N^d/d! + O(N^{d-1})$ **distinct** elements

Best rank-1 approximation

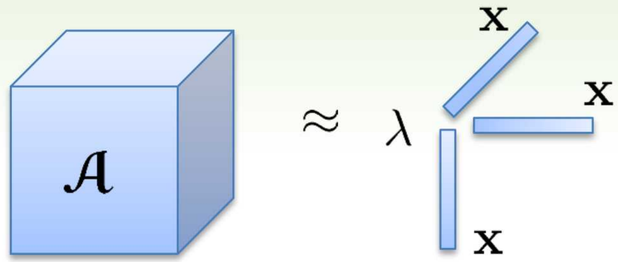


Rank-R factorization



Applications of symmetric tensors: diffusion tensor imaging (DTI/HARDI), higher-order statistics, higher-order derivatives, relativity, signal processing, etc.

Best Symmetric Rank-1 Approximation



Data   Model

$$\mathcal{M} = \lambda \mathbf{x} \circ \mathbf{x} \circ \mathbf{x}$$

$$\min_{\lambda, \mathbf{x}} \sum_{ijk} (a_{ijk} - \lambda x_i x_j x_k)^2$$

Eliminate λ:

$$\lambda = \sum_{ijk} a_{ijk} \, x_i x_j x_k$$

$$\max_{\mathbf{x}} \mathcal{A}\mathbf{x}^d \equiv \sum_{ijk} a_{ijk} \, x_i x_j x_k$$

Nonlinear Program

$$\max_{\mathbf{x}} f(\mathbf{x}) \equiv \frac{\mathcal{A}\mathbf{x}^d}{\mathcal{B}\mathbf{x}^d} \|\mathbf{x}\|^d$$

$$\text{subject to } \|\mathbf{x}\| = 1$$

FYI: Generalized Eigenpair

(Chang, Pearson, Zhang 2009)

$$\mathcal{A}\mathbf{x}^{d-1} = \lambda \mathcal{B}\mathbf{x}^{d-1}$$

$$\text{subject to } (\lambda, \mathbf{x}) \in \mathbb{R} \times \mathbb{R}^N$$

$$(\mathcal{A}\mathbf{x}^{d-1})_i \equiv \sum_{ijk} a_{ijk} \, x_j x_k \text{ for } i = 1, \ldots, N$$

$\mathcal{B} =$
- "identity" tensor $\Rightarrow$ Z-eigenproblem
- "diagonal ones" tensor $\Rightarrow$ H-eigenproblem

Qi 2005; Lim 2005; Chang, Pearson, & Zhang 2009

Adaptive Shifted Power Method: Special Optimization on a Sphere

Theorem

Assume $\mathbf{w} \in \{\, \mathbf{x} \mid \|\mathbf{x}\| = 1 \,\}$,

$\Omega =$ open nbhd of $\mathbf{w}$,

$\hat{f}$ convex and C^1 on Ω

Let $\mathbf{v} = \nabla \hat{f}(\mathbf{w}) / \|\nabla \hat{f}(\mathbf{w})\|$.

If $\mathbf{v} \in \Omega$ and $\mathbf{v} \neq \mathbf{w}$,

then $\hat{f}(\mathbf{v}) > \hat{f}(\mathbf{w})$

Simple fixed point iteration is monotonically convergent:

$$\mathbf{x}_{k+1} \leftarrow \frac{\nabla \hat{f}(\mathbf{x}_k)}{\|\nabla \hat{f}(\mathbf{x}_k)\|}$$

Creating local convexity on a sphere:

$$\hat{f}(\mathbf{x}) = f(\mathbf{x}) + \alpha \|\mathbf{x}\|^d$$

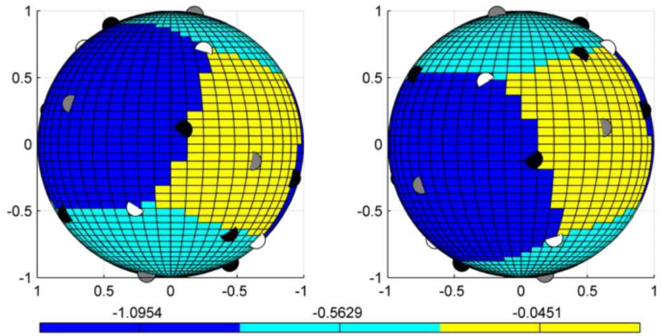
For $\mathbf{x} \in \{\, \mathbf{x} \mid \|\mathbf{x}\| = 1 \,\}$:

$$\hat{\mathbf{g}}(\mathbf{x}) = \mathbf{g}(\mathbf{x}) + \alpha d \mathbf{x},$$

$$\hat{\mathbf{H}}(\mathbf{x}) = \mathbf{H}(\mathbf{x}) + \alpha d \mathbf{I} + \alpha d(d-2)\mathbf{x}\mathbf{x}^{\mathsf{T}}$$

Use Weyl's inequality to choose α

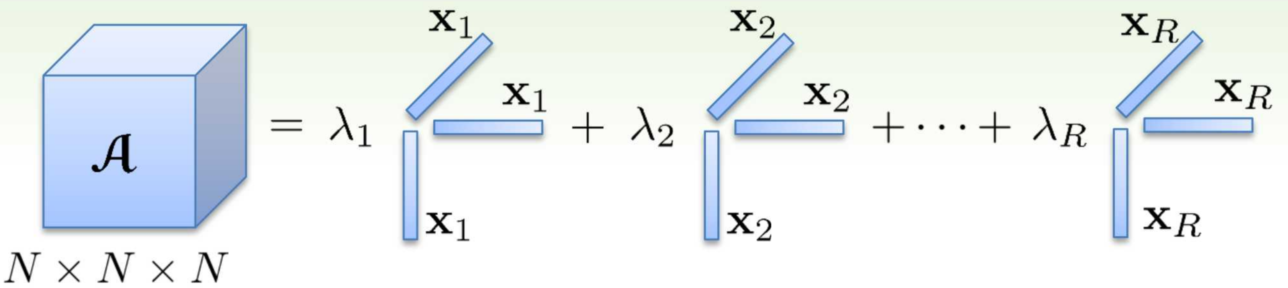
Positive Stable Basins of Attraction for 3x3x3x3 Tensors



Regalia & Kofidis 2002 & 2003; Kolda & Mayo 2012 & 2014

Han (2012): Optimization formulation; Cui, Dai, Nie (2014): SDP formulation

Optimization for Symmetric CP Tensor Decomposition



variables = $R(N + 1)$
data points = $N^d/d!$

Option 1: Standard least squares

Exact penalty to remove scaling ambiguity

$$\min_{\mathcal{M}} \sum_{ijk} (a_{ijk} - m_{ijk})^2 + \gamma \sum_r (\|\mathbf{x}_r\|^2 - 1)^2 \text{ s.t. } \mathcal{M} = \sum_r \lambda_r \, \mathbf{x}_r^d$$

Option 2: Distinct elements only ⇒ *Overall best option for time and accuracy*

$$\min_{\mathcal{M}} \sum_{i \le j \le k} (a_{ijk} - m_{ijk})^2 + \gamma \sum_r (\|\mathbf{x}_r\|^2 - 1)^2 \text{ s.t. } \mathcal{M} = \sum_r \lambda_r \, \mathbf{x}_r^d$$

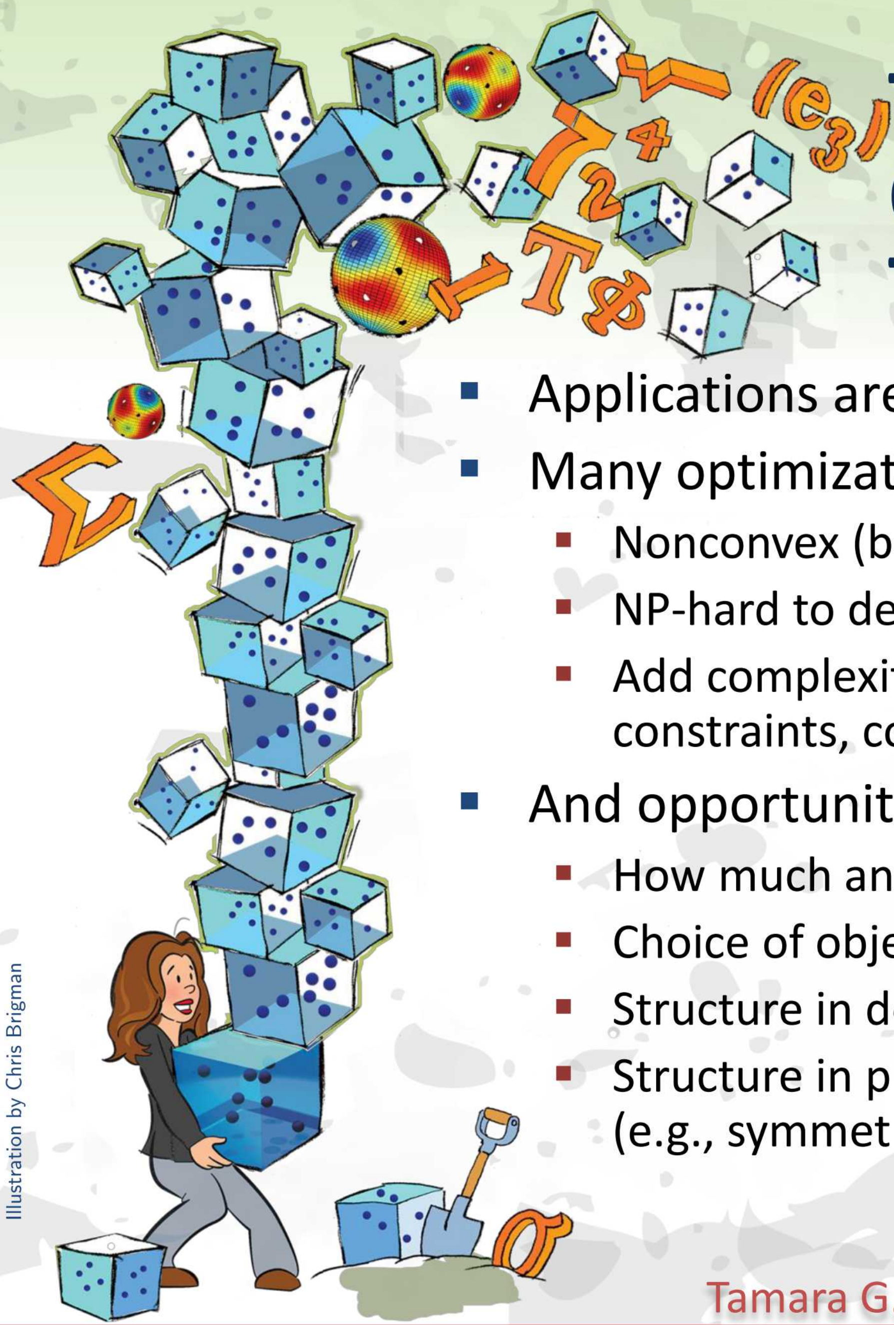
Option 3: Ignore symmetry ⇒ *2-100 times faster when it works*

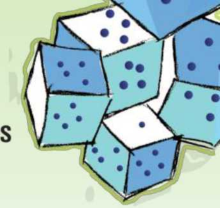
$$\text{Uniqueness: } 2R + (d-1) \le d \cdot \text{k-rank}(\mathbf{X})$$

$$\min_{\mathcal{M}} \sum_{ijk} (a_{ijk} - m_{ijk})^2 \text{ s.t. } \mathcal{M} = \sum_r \lambda_r \, \mathbf{x}_r \circ \mathbf{y}_r \circ \mathbf{z}_r$$

Orthogonal symmetric CP is equivalent to symmetric EVD.
(Kolda 2015)

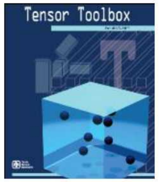
Kolda, Math Prog B, 2015; Algebraic geometry: Brachat et al. (2010), Oeding & Ottaviani (2011); Complex: Nie 2015





Sandia National Laboratories

Takeaways: Optimization for Tensor Decomposition



- Applications are ubiquitous in data analysis
- Many optimization challenges…
  - Nonconvex (but one example of eliminating this)
  - NP-hard to determine complexity (i.e., choice of R)
  - Add complexity for higher order, higher dimension, constraints, coupled problems
- And opportunities…
  - How much and which data do we need?
  - Choice of objective function
  - Structure in derivatives
  - Structure in problems (e.g., symmetry)

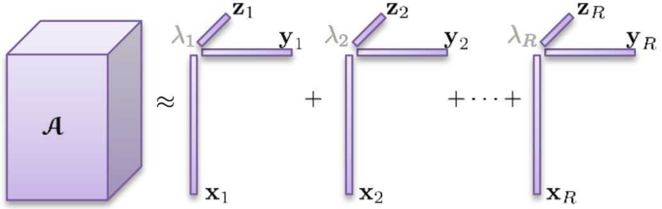


Illustration by Chris Brigman

Tamara G. Kolda: http://www.sandia.gov/~tgkolda/