# The Canonical Tensor Decomposition and Its Applications to Social Network Analysis

Evrim Acar, Tamara G. Kolda and Daniel M. Dunlavy
Sandia National Labs

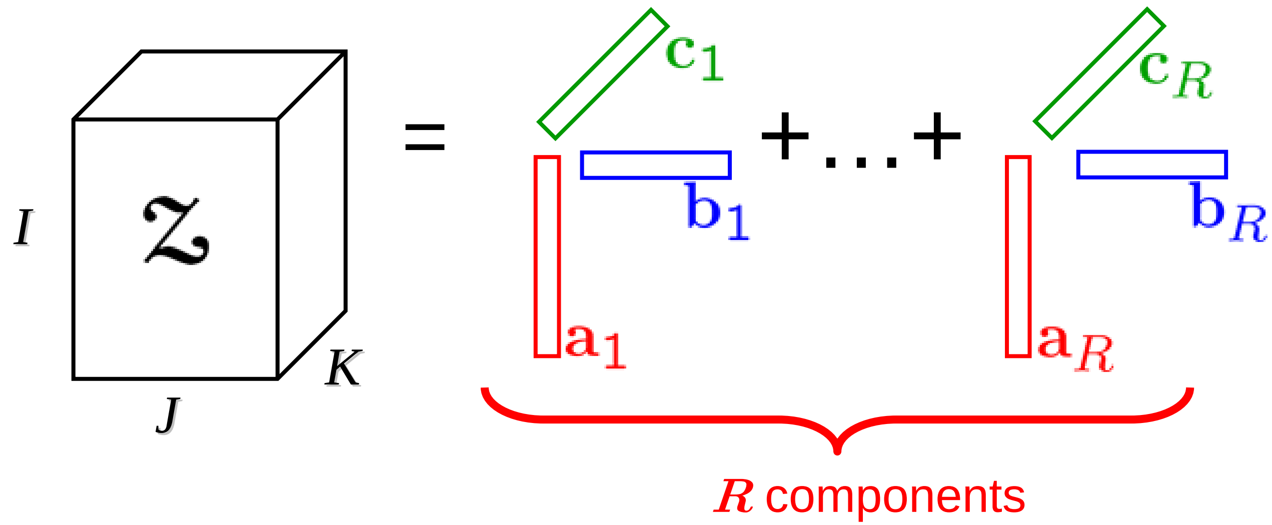
LDRD
LABORATORY DIRECTED RESEARCH & DEVELOPMENT

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

# What is Canonical Tensor Decomposition?

CANDECOMP/PARAFAC (CP) model [Hitchcock'27, Harshman'70, Carroll & Chang'70]




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

$$= [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!]$$

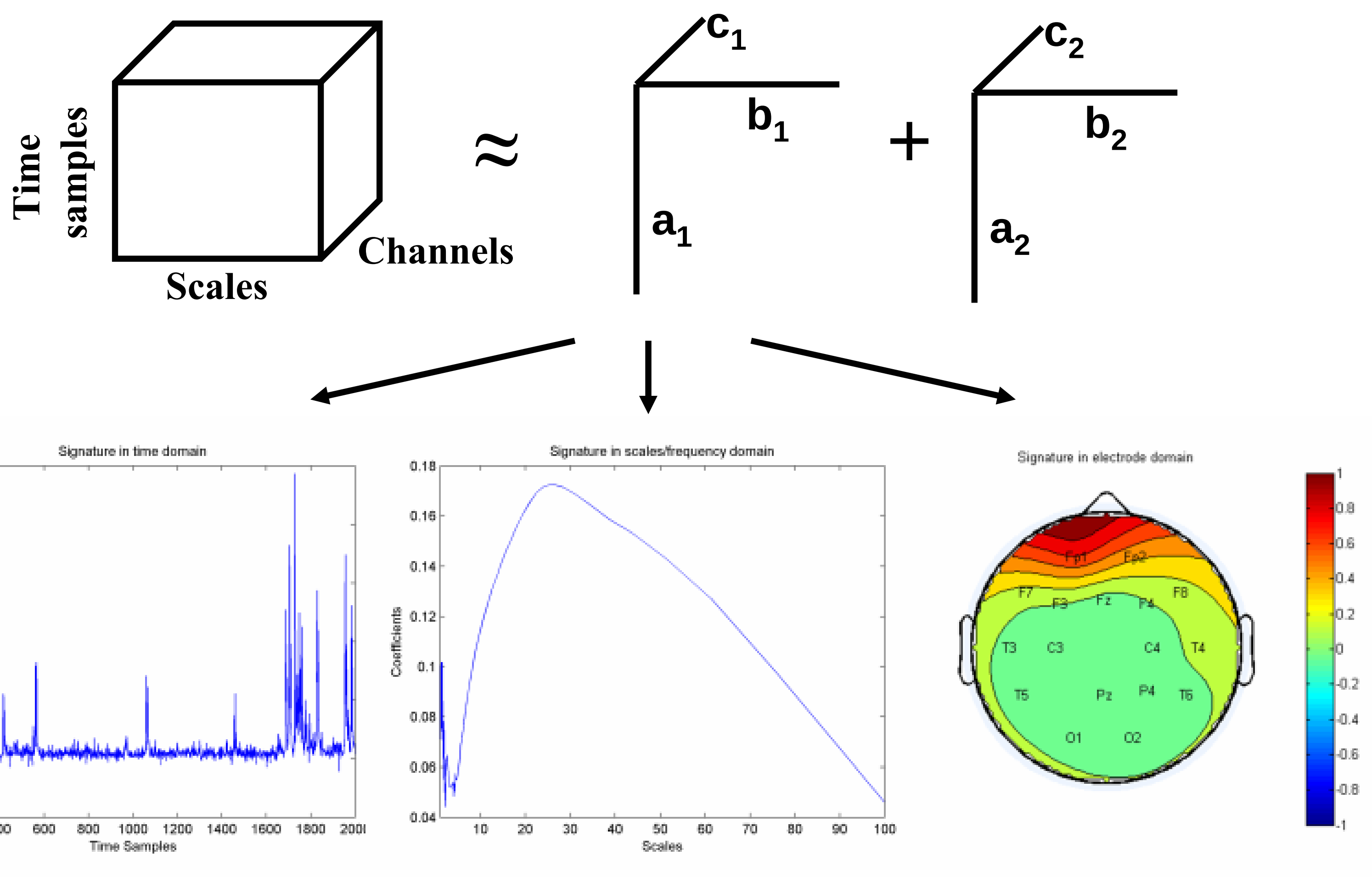
$$\mathbf{A} \in \mathbb{R}^{I \times R} = \begin{bmatrix} \mathbf{a}_1 & \cdots & \mathbf{a}_R \end{bmatrix}$$

$$\mathbf{B} \in \mathbb{R}^{J \times R} = \begin{bmatrix} \mathbf{b}_1 & \cdots & \mathbf{b}_R \end{bmatrix}$$

$$\mathbf{C} \in \mathbb{R}^{K \times R} = \begin{bmatrix} \mathbf{c}_1 & \cdots & \mathbf{c}_R \end{bmatrix}$$

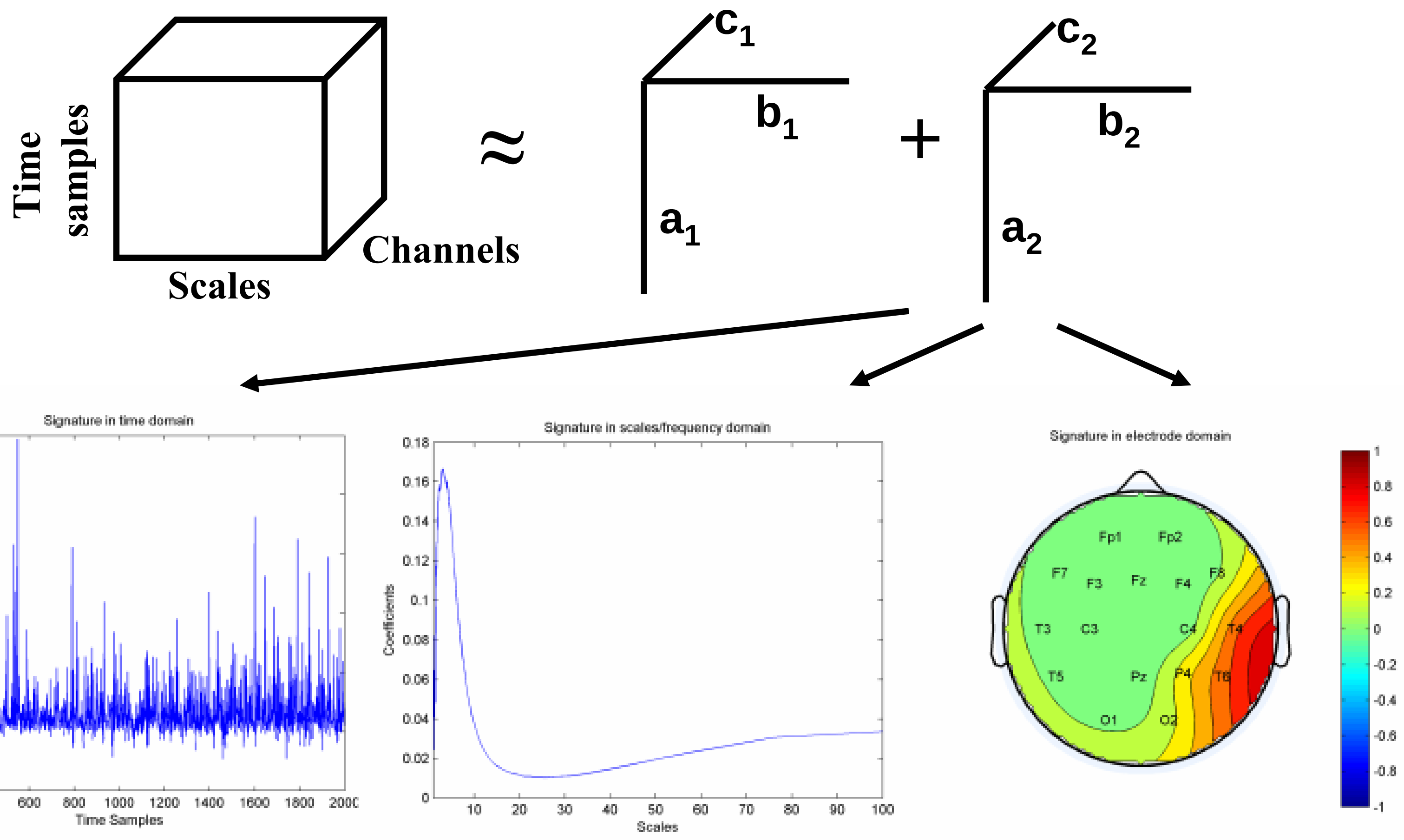
# CP Application: Neuroscience

**Epileptic Seizure Localization:**



# CP Application: Neuroscience

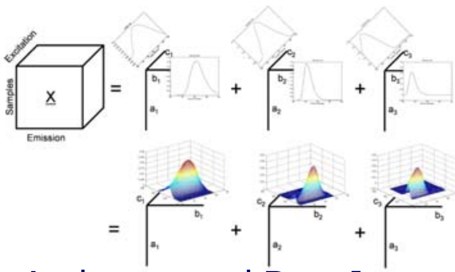
**Epileptic Seizure Localization:**



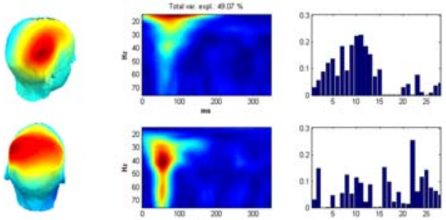

Acar et al, 2007, De Vos et al, 2007

# CP has Numerous Applications!

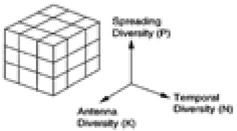
- **Chemometrics**
  - Fluorescence Spectroscopy
  - Chromatographic Data Analysis
- **Neuroscience**
  - Epileptic Seizure Localization
  - Analysis of EEG and ERP
- **Signal Processing**
- **Computer Vision**
  - Image compression, classification
  - Texture analysis
- **Social Network Analysis**
  - Web link analysis
  - Conversation detection in emails
  - Text analysis
- **Approximation of PDEs**



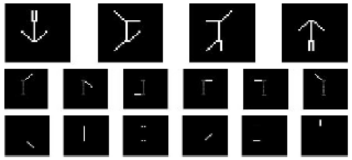
Andersen and Bro, *Journal of Chemometrics*, 2003.



Mørup, Hansen and Arnfred, *Journal of Neuroscience Methods*, 2007.



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



Hazan, Polak and Shashua, *ICCV 2005*.


Bader, Berry, Browne, *Survey of Text Mining: Clustering, Classification, and Retrieval*, 2nd Ed., 2007.

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

Doostan and Iaccarino, *Journal of Computational Physics*, 2009.

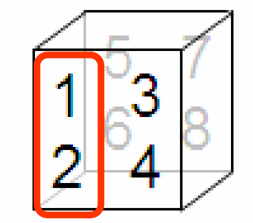
# Algorithms: How Can We Compute CP?

# Mathematical Details for CP

**Unfolding (Matricization)**

Columns: mode-1 fibers

$\mathcal{X} =$



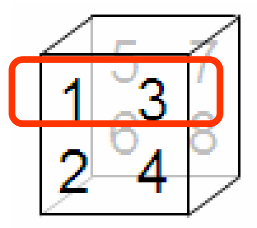

$$X_{(1)} = \begin{bmatrix} 1 & 3 & 5 & 7 \\ 2 & 4 & 6 & 8 \end{bmatrix}$$

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

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

# Mathematical Details for CP

**Unfolding (Matricization)**

$\mathcal{X} =$ 


Row: mode-2 fibers

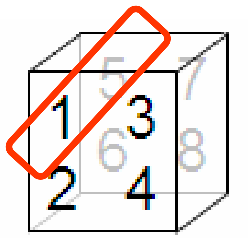
$$X_{(1)} = \begin{bmatrix} 1 & 3 & 5 & 7 \\ 2 & 4 & 6 & 8 \end{bmatrix}$$

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

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

# Mathematical Details for CP

**Unfolding (Matricization)**

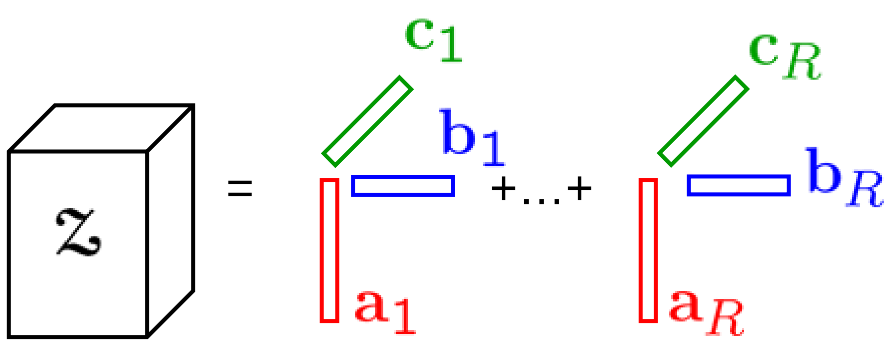
$\mathcal{X} =$ 


Tube: mode-3 fibers

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

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

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




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

$$\mathcal{Z} = [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!]$$

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

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

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

**Matrix Khatri-Rao Product**

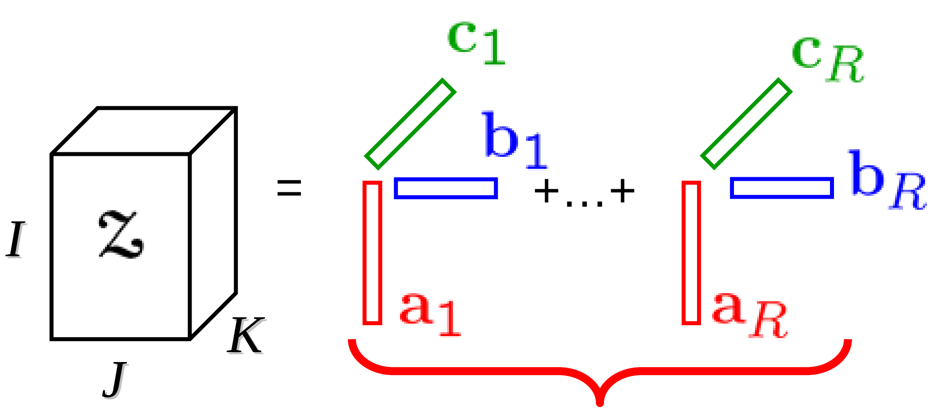
$$\mathbf{U} \odot \mathbf{V} = \begin{bmatrix} \mathbf{u}_1 \otimes \mathbf{v}_1 & \cdots & \mathbf{u}_R \otimes \mathbf{v}_R \end{bmatrix}$$

# CP is a Nonlinear Optimization Problem

Given tensor $\mathcal{Z}$ and $R$ (# of components), find matrices **A**, **B**, **C** that solve the following problem:

**Optimization Problem**

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \| \mathcal{Z} - [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!] \|^2$$




$R(I+J+K)$
variables

**Objective Function**

$$f(\mathbf{x}) = \frac{1}{2} \| \mathcal{Z} - [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!] \|^2$$

where the vector **x** comprises the entries of A, B, and C stacked column-wise:

$$\mathbf{x} = \begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_R \\ \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_R \\ \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_R \end{bmatrix}$$

# Traditional Approach: CPALS

CPALS dating back to Harshman'70 and Carroll & Chang'70 solves for one factor matrix at a time.

**Optimization Problem**

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \| \mathfrak{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2$$

**Alternating Algorithm**

for k = 1,…

$$\min_{\mathbf{A}} \| \mathfrak{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2$$

$$\min_{\mathbf{B}} \| \mathfrak{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2$$

$$\min_{\mathbf{C}} \| \mathfrak{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2$$

end

Each step can be converted to a matrix least squares problem:

$$\min_{\mathbf{A}} \left\| \mathbf{Z}_{(1)} - \mathbf{A}(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}} \right\|^2$$

$$\mathbf{A} = \mathbf{Z}_{(1)} \left( (\mathbf{C} \odot \mathbf{B})^{\mathsf{T}} \right)^{\dagger}$$

$\mathbf{Z}_{(1)}$: *I* x *JK*; $(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$: *JK* x *R*

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

$\mathbf{A}$: *I* x *R*; $\mathbf{Z}_{(1)}$: *I* x *JK*; $(\mathbf{C} \odot \mathbf{B})$: *JK* x *R*; $(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{B}^{\mathsf{T}}\mathbf{B})^{\dagger}$: *R* x *R* matrix

# Traditional Approach: CPALS

**Optimization Problem**

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \| \mathcal{Z} - [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!] \|^2$$

Repeat the following steps until “convergence”:

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

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

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

*Very fast, but not always accurate.*

*Not guaranteed to converge to a stationary point.*

*Other issues, e.g., cannot exploit symmetry.*

# Our Approach: CPOPT

Unlike CPALS, CPOPT solves for all factor matrices simultaneously using a gradient based optimization.

**Optimization Problem**

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \| \mathcal{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2$$

Define the objective function:

$$f(\mathbf{x}) = \frac{1}{2} \| \mathcal{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2$$

$$\mathbf{x} = \begin{bmatrix} \mathbf{a}_1 \\ \vdots \\ \mathbf{a}_R \\ \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_R \\ \mathbf{c}_1 \\ \vdots \\ \mathbf{c}_R \end{bmatrix} \qquad f : \mathbb{R}^{(I+J+K)R} \mapsto \mathbb{R}$$

$$\Rightarrow \quad \nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f}{\partial \mathbf{a}_1} \\ \vdots \\ \frac{\partial f}{\partial \mathbf{a}_R} \\ \frac{\partial f}{\partial \mathbf{b}_1} \\ \vdots \\ \frac{\partial f}{\partial \mathbf{b}_R} \\ \frac{\partial f}{\partial \mathbf{c}_1} \\ \vdots \\ \frac{\partial f}{\partial \mathbf{c}_R} \end{bmatrix}$$

# Rewriting the Objective Function

$$f(\mathbf{x}) = \frac{1}{2} \| \mathcal{Z} - [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \|^2$$

$$f(\mathbf{x}) = \underbrace{\frac{1}{2} \| \mathcal{Z} \|^2}_{f_1(\mathbf{x})} - \underbrace{\langle \mathcal{Z}, [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \rangle}_{f_2(\mathbf{x})} + \underbrace{\frac{1}{2} \| [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \|^2}_{f_3(\mathbf{x})}$$

$$\nabla f_1(\mathbf{x}) = 0$$

**Inner Product**

$$\langle \mathcal{U}, \mathcal{V} \rangle = \sum_{i,j,k} u_{ijk} v_{ijk}$$

**Norm**

$$\| \mathcal{U} \|^2 = \langle \mathcal{U}, \mathcal{U} \rangle$$

# Derivative of 2nd Summand

$$f(\mathbf{x}) = \underbrace{\frac{1}{2}\| \mathfrak{Z} \|^2}_{f_1(\mathbf{x})} - \underbrace{\langle \mathfrak{Z}, [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \rangle}_{f_2(\mathbf{x})} + \underbrace{\frac{1}{2}\| [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2}_{f_3(\mathbf{x})}$$

**Tensor-Vector Multiplication**

$\mathfrak{X} \in \mathbb{R}^{I \times J \times K}, \mathbf{u} \in \mathbb{R}^{I}, \mathbf{v} \in \mathbb{R}^{J}, \mathbf{w} \in \mathbb{R}^{K}$

$$\mathfrak{X} \times_1 \mathbf{u} \times_2 \mathbf{v} \times_3 \mathbf{w} = \sum_i \sum_j \sum_k x_{ijk}\, u_i\, v_j\, w_k \in \mathbb{R}$$

$$f_2(\mathbf{x}) = \sum_{\ell=1}^{R} \mathfrak{Z} \times_1 \mathbf{a}_\ell \times_2 \mathbf{b}_\ell \times_3 \mathbf{c}_\ell = \sum_{\ell=1}^{R} (\mathfrak{Z} \times_2 \mathbf{b}_\ell \times_3 \mathbf{c}_\ell)^{\mathsf{T}} \mathbf{a}_\ell.$$

$$\frac{\partial f_2}{\partial \mathbf{a}_r}(\mathbf{x}) = \mathfrak{Z} \times_2 \mathbf{b}_r \times_3 \mathbf{c}_r \in \mathbb{R}^{I}$$

*Analogous formulas exist for partials w.r.t. columns of B and C.*

# Derivative of 3rd Summand

$$f(\mathbf{x}) = \underbrace{\frac{1}{2}\| \mathcal{Z} \|^2}_{f_1(\mathbf{x})} - \underbrace{\langle \mathcal{Z}, [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \rangle}_{f_2(\mathbf{x})} + \underbrace{\frac{1}{2}\| [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2}_{f_3(\mathbf{x})}$$

$$f_3(\mathbf{x}) = \sum_{k=1}^{R}\sum_{\ell=1}^{R} \mathbf{a}_k^{\mathsf{T}}\mathbf{a}_\ell \mathbf{b}_k^{\mathsf{T}}\mathbf{b}_\ell \mathbf{c}_k^{\mathsf{T}}\mathbf{c}_\ell$$

$$= \left(\mathbf{b}_r^{\mathsf{T}}\mathbf{b}_r\mathbf{c}_r^{\mathsf{T}}\mathbf{c}_r\right)\mathbf{a}_r^{\mathsf{T}}\mathbf{a}_r + 2\sum_{k\neq r}\left(\mathbf{b}_r^{\mathsf{T}}\mathbf{b}_k\mathbf{c}_r^{\mathsf{T}}\mathbf{c}_k\right)\mathbf{a}_r^{\mathsf{T}}\mathbf{a}_k + \sum_{k\neq r}\sum_{\ell\neq r}\mathbf{a}_k^{\mathsf{T}}\mathbf{a}_\ell \mathbf{b}_k^{\mathsf{T}}\mathbf{b}_\ell \mathbf{c}_k^{\mathsf{T}}\mathbf{c}_\ell$$

$$\frac{\partial f_3}{\partial \mathbf{a}_r}(\mathbf{x}) = 2\left(\mathbf{b}_r^{\mathsf{T}}\mathbf{b}_r\mathbf{c}_r^{\mathsf{T}}\mathbf{c}_r\right)\mathbf{a}_r + 2\sum_{k\neq r}\left(\mathbf{b}_r^{\mathsf{T}}\mathbf{b}_k\mathbf{c}_r^{\mathsf{T}}\mathbf{c}_k\right)\mathbf{a}_k$$

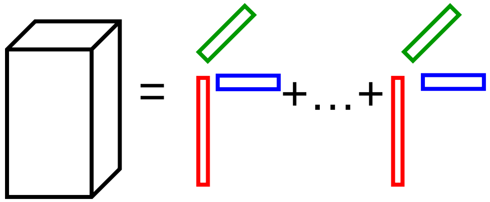
$$= 2\sum_{k=1}^{R}\left(\mathbf{b}_r^{\mathsf{T}}\mathbf{b}_k\mathbf{c}_r^{\mathsf{T}}\mathbf{c}_k\right)\mathbf{a}_k \in \mathbb{R}^I$$

*Analogous formulas exist for partial w.r.t. columns of B and C.*

# Objective and Gradient

**Objective Function**

$$f(\mathbf{x}) = \frac{1}{2} \| \mathcal{Z} - [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!] \|^2$$




**Gradient** (for $r = 1,\ldots,R$)

$$\frac{\partial f}{\partial \mathbf{a}_r}(\mathbf{x}) = -\mathcal{Z} \times_2 \mathbf{b}_r \times_3 \mathbf{c}_r + \sum_{k=1}^{R} \left( \mathbf{b}_r^{\mathsf{T}} \mathbf{b}_k \mathbf{c}_r^{\mathsf{T}} \mathbf{c}_k \right) \mathbf{a}_k$$

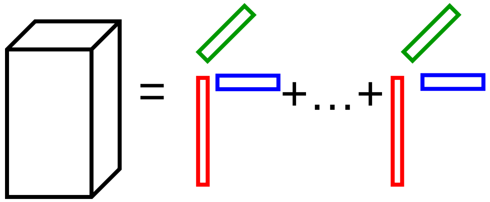
$$\frac{\partial f}{\partial \mathbf{b}_r}(\mathbf{x}) = -\mathcal{Z} \times_1 \mathbf{a}_r \times_3 \mathbf{c}_r + \sum_{k=1}^{R} \left( \mathbf{a}_r^{\mathsf{T}} \mathbf{a}_k \mathbf{c}_r^{\mathsf{T}} \mathbf{c}_k \right) \mathbf{b}_k$$

$$\frac{\partial f}{\partial \mathbf{c}_r}(\mathbf{x}) = -\mathcal{Z} \times_1 \mathbf{a}_r \times_2 \mathbf{b}_r + \sum_{k=1}^{R} \left( \mathbf{a}_r^{\mathsf{T}} \mathbf{a}_k \mathbf{b}_r^{\mathsf{T}} \mathbf{b}_k \right) \mathbf{c}_k$$

# Gradient in Matrix Form

**Objective Function**

$$f(\mathbf{x}) = \frac{1}{2} \| \mathcal{Z} - [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!] \|^2$$




**Gradient**

$$\frac{\partial f}{\partial \mathbf{A}}(\mathbf{x}) = -\mathbf{Z}_{(1)}(\mathbf{C} \odot \mathbf{B}) + \mathbf{A}(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{B}^{\mathsf{T}}\mathbf{B})$$

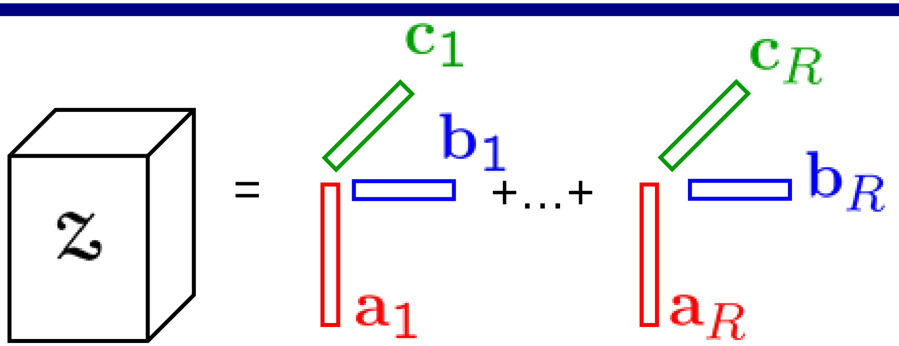
$$\frac{\partial f}{\partial \mathbf{B}}(\mathbf{x}) = -\mathbf{Z}_{(2)}(\mathbf{C} \odot \mathbf{A}) + \mathbf{B}(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{A}^{\mathsf{T}}\mathbf{A})$$

$$\frac{\partial f}{\partial \mathbf{C}}(\mathbf{x}) = -\mathbf{Z}_{(3)}(\mathbf{B} \odot \mathbf{A}) + \mathbf{C}(\mathbf{B}^{\mathsf{T}}\mathbf{B} * \mathbf{A}^{\mathsf{T}}\mathbf{A})$$

*Note that this formulation can be used to derive the ALS approach!*

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

# Indeterminacies of CP




- CP is often unique.
- However, CP has two fundamental indeterminacies
  - **Permutation** – The components can be reordered
    - Swap $\mathbf{a}_1$, $\mathbf{b}_1$, $\mathbf{c}_1$ with $\mathbf{a}_3$, $\mathbf{b}_3$, $\mathbf{c}_3$

    ← *Not a big deal. Leads to multiple, but separated, minima.*
  - **Scaling** – The vectors comprising a single rank-one factor can be scaled
    - Replace $\mathbf{a}_1$ and $\mathbf{b}_1$ with 2 $\mathbf{a}_1$ and ½ $\mathbf{b}_1$

    ← *This leads to a continuous space of equivalent solutions.*

Sandia National Laboratories

# Adding Regularization

**Objective Function**

$$f(\mathbf{x}) = \frac{1}{2} \| \mathcal{Z} - [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!] \|^2 + \frac{\lambda}{2} \left( \| \mathbf{A} \|_F^2 + \| \mathbf{B} \|_F^2 + \| \mathbf{C} \|_F^2 \right)$$

**Gradient**

$$\frac{\partial f}{\partial \mathbf{A}}(\mathbf{x}) = -\mathbf{Z}_{(1)}(\mathbf{C} \odot \mathbf{B}) + \mathbf{A}(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{B}^{\mathsf{T}}\mathbf{B}) + \lambda \mathbf{A}$$

$$\frac{\partial f}{\partial \mathbf{B}}(\mathbf{x}) = -\mathbf{Z}_{(2)}(\mathbf{C} \odot \mathbf{A}) + \mathbf{B}(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{A}^{\mathsf{T}}\mathbf{A}) + \lambda \mathbf{B}$$

$$\frac{\partial f}{\partial \mathbf{C}}(\mathbf{x}) = -\mathbf{Z}_{(3)}(\mathbf{B} \odot \mathbf{A}) + \mathbf{C}(\mathbf{B}^{\mathsf{T}}\mathbf{B} * \mathbf{A}^{\mathsf{T}}\mathbf{A}) + \lambda \mathbf{C}$$

# Our methods: CPOPT & CPOPTR

**CPOPT**: Apply derivative-based optimization method to the following objective function:

$$f(\mathbf{x}) = \frac{1}{2} \| \mathfrak{Z} - [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \|^2$$

**CPOPTR**: Apply derivative-based optimization method to the following regularized objective function:

$$f(\mathbf{x}) = \frac{1}{2} \| \mathfrak{Z} - [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \|^2 + \frac{\lambda}{2} \left( \| \mathbf{A} \|_F^2 + \| \mathbf{B} \|_F^2 + \| \mathbf{C} \|_F^2 \right)$$

# Another competing method: CPNLS

**CPNLS**: Apply nonlinear least squares solver to the following equations:

$$F(\mathbf{x}) = \text{vec}\left(\mathcal{Z} - [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]\right)$$

⬇

$$F : \mathbb{R}^{(I+J+K)R} \mapsto \mathbb{R}^{IJK}$$

⬇

Jacobian is of size $IJK \times (I + J + K)R$.

Proposed by **Paatero'**97 and also **Tomasi and Bro**'05.

# Experimental Set-Up [Tomasi&Bro'06]

*20 triplets*

**Step 2**: Construct tensor from factor matrices and add noise. All combinations of:

- Homoscedastic: 1%, 5%, 10%
- Heteroscedastic: 0%, 1%, 5%

$$\mathcal{Z} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] + \mathcal{N}$$

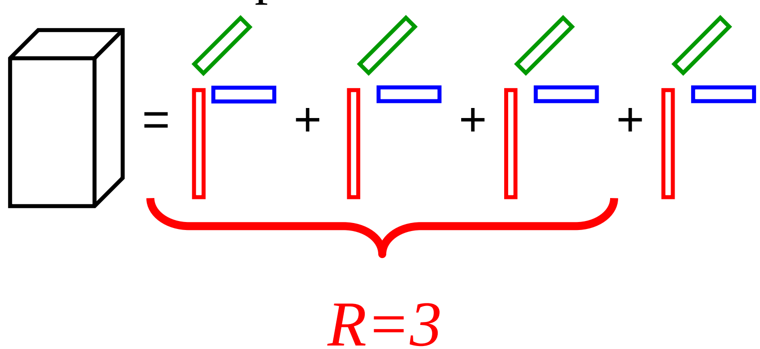
**Step 1**: Generate random factor matrices A, B, C with $R_{true}$ = 3 or 5 columns each and collinearity set to 0.5,

i.e., $\mathbf{a}_r^{\mathsf{T}} \mathbf{a}_s = 0.5$

*180 tensors*

**Step 3**: Use algorithm to extract factors, using $R_{true}$ and $R_{true}+1$ factors. Compare against factors in Step 1.

*360 tests*



# Implementation Details

- All experiments were performed in MATLAB on a Linux workstation (Quad-Core Intel Xeon 2.50GHz, 9 GB RAM).

- Methods
  - **CPALS** – Alternating least squares. Used **parafac_als** in the Tensor Toolbox (Bader & Kolda)
  - **CPNLS** – Nonlinear least squares. Used **PARAFAC3W**, which implements Levenberg-Marquadt (necessary due to scaling ambiguity), by Tomasi and Bro.
  - **CPOPT** – Optimization. Used routines in the **Tensor Toolbox** in calculation of function values and gradients. Optimization via Nonlinear Conjugate Gradient (NCG) method with Hestenes-Stiefel update, using **Poblano** (in-house code to be released soon).
  - **CPOPTR** – Optimization with regularization. Same as above. (Regularization parameter = 0.02.)

# CPOPT is Fast and Accurate

Generated 360 dense test problems (with ranks 3 and 5) and factorized with *R* as the correct number of components and one more than that. Total of 720 tests for each entry below.

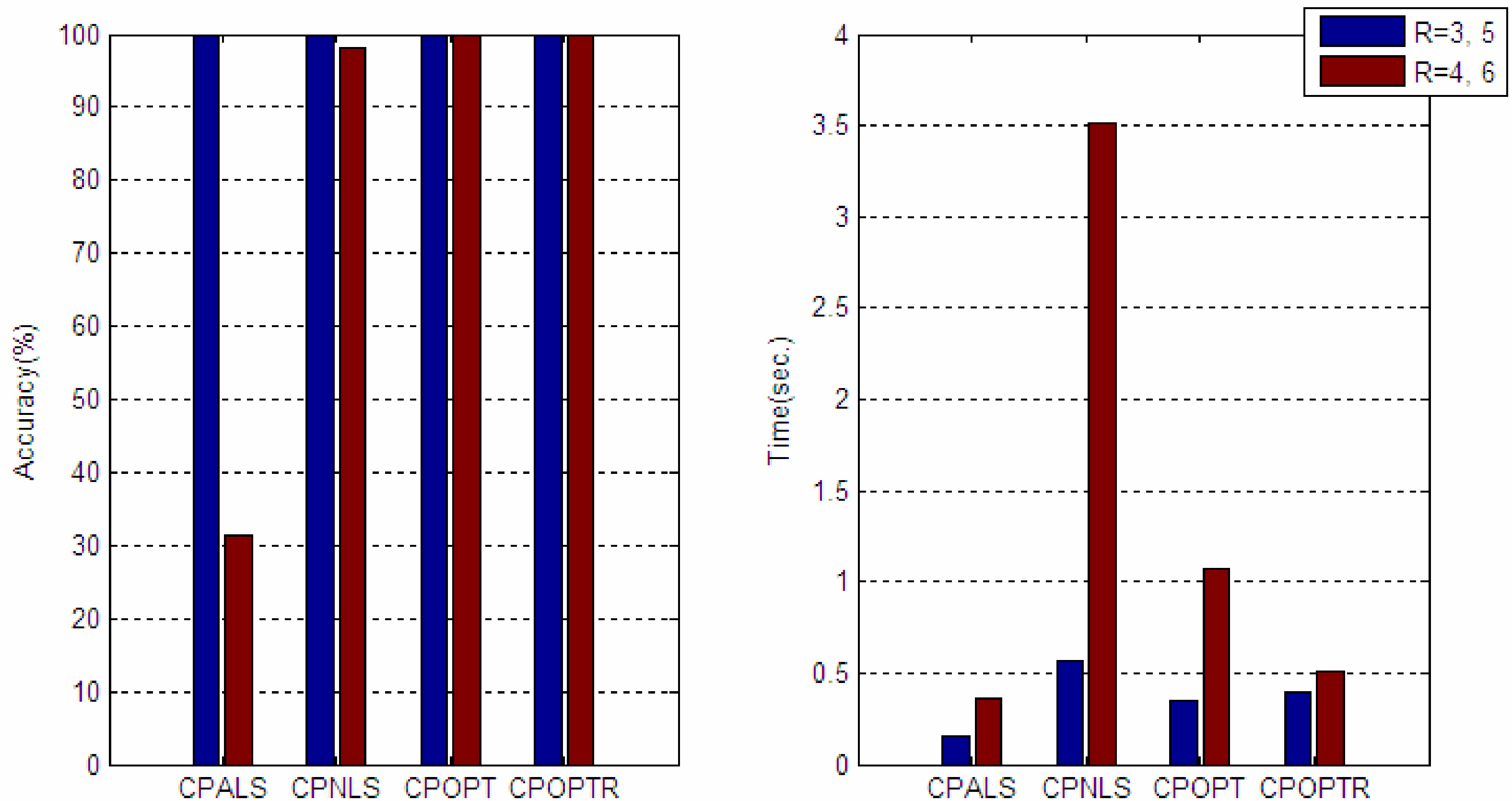
| | Time (sec) | | | |
|---|---|---|---|---|
| Size | CPALS | CPNLS | CPOPT | CPOPTR |
| $20 \times 20 \times 20$ | $0.5 \pm 1.0$ | $0.3 \pm 0.3$ | $0.3 \pm 0.2$ | $0.2 \pm 0.1$ |
| $50 \times 50 \times 50$ | $0.3 \pm 0.3$ | $2.0 \pm 2.6$ | $0.7 \pm 0.5$ | $0.5 \pm 0.1$ |
| $100 \times 100 \times 100$ | $1.7 \pm 1.1$ | $11.5 \pm 11.5$ | $5.6 \pm 3.6$ | $4.3 \pm 1.3$ |
| $250 \times 250 \times 250$ | $26.6 \pm 9.1$ | $143.9 \pm 125.0$ | $83.5 \pm 35.2$ | $81.9 \pm 22.8$ |

| | Accuracy (%) | | | |
|---|---|---|---|---|
| Size | CPALS | CPNLS | CPOPT | CPOPTR |
| $20 \times 20 \times 20$ | 78.8 | 99.0 | 99.9 | 100.0 |
| $50 \times 50 \times 50$ | 65.7 | 99.0 | 100.0 | 100.0 |
| $100 \times 100 \times 100$ | 63.5 | 97.9 | 100.0 | 100.0 |
| $250 \times 250 \times 250$ | 62.2 | 99.0 | 100.0 | 100.0 |

| $K \times K \times K$<br>$R$ = # components | $O(RK^3)$ | $O(R^3K^3)$ | $O(RK^3)$ | $O(RK^3)$ |
|---|---|---|---|---|

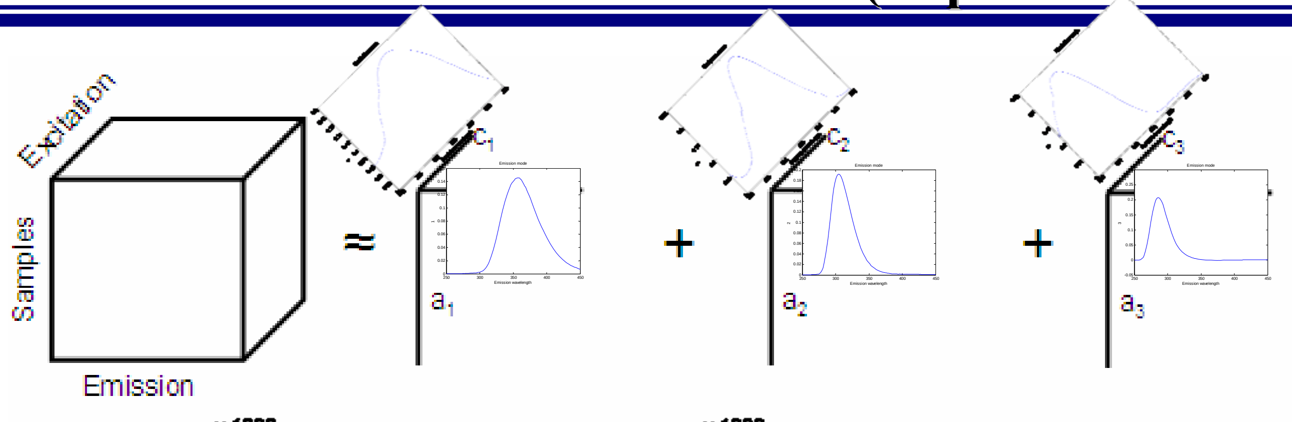
# Overfactoring has a significant impact

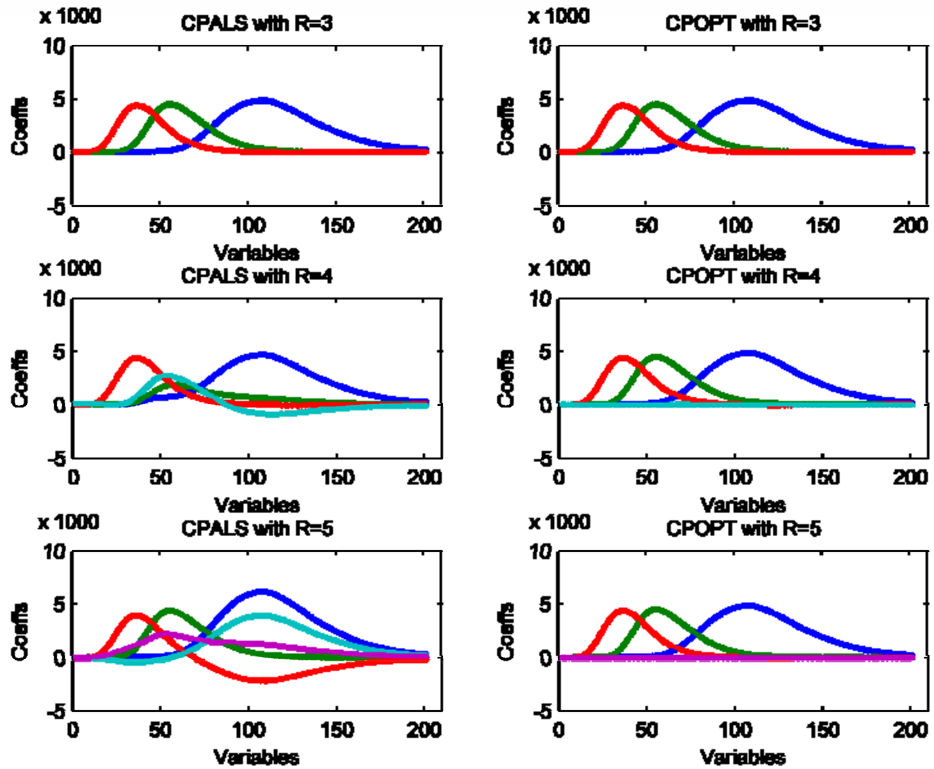
$50 \times 50 \times 50$



# CPOPT is robust to overfactoring

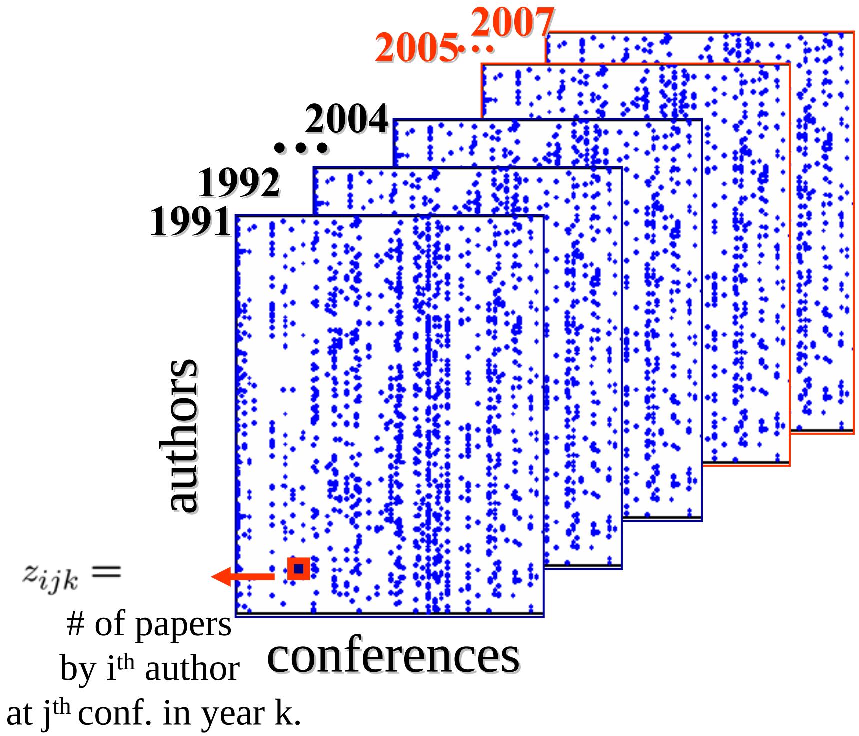
**Amino (http://www.models.life.ku.dk/)**



# Application: Link Prediction

# Link Prediction on Bibliometric Data

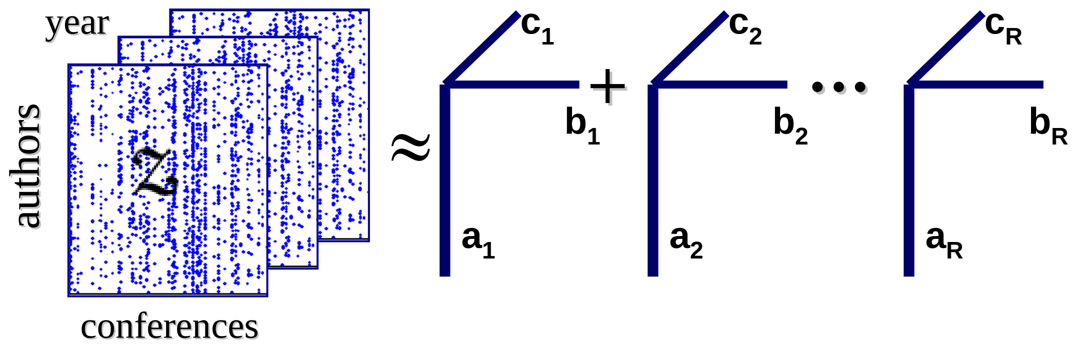



Question1: Can we use tensor decompositions to model the data and extract meaningful factors?

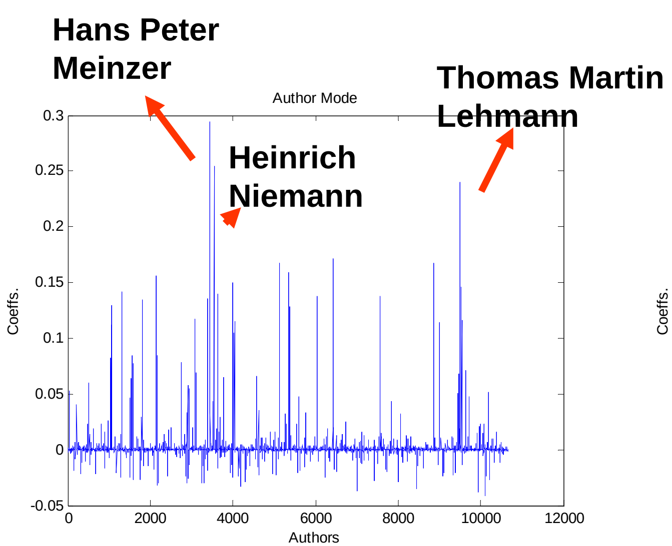
Question2: Can we predict who is going to publish at which conferences in future?

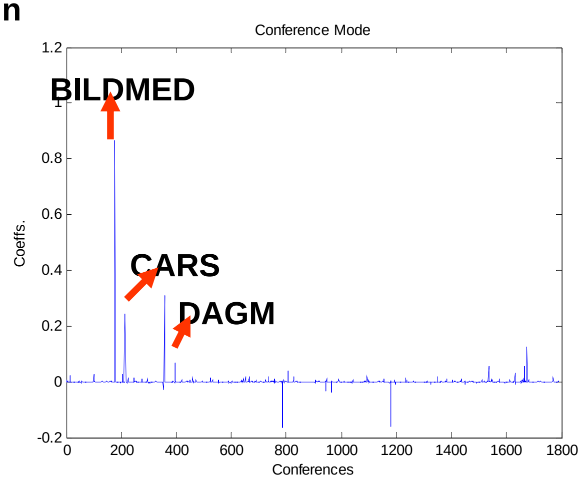
# Components make sense!

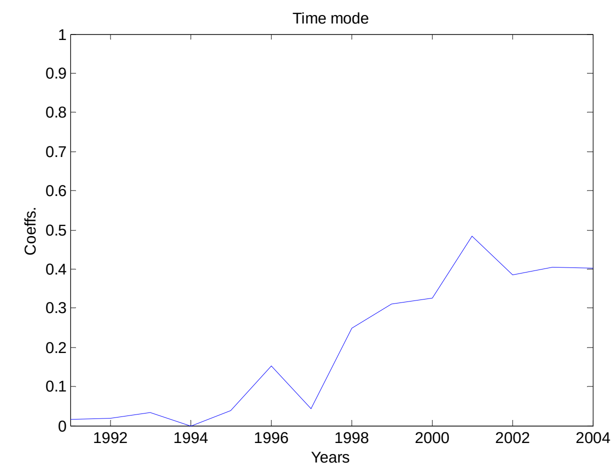
**DBLP**



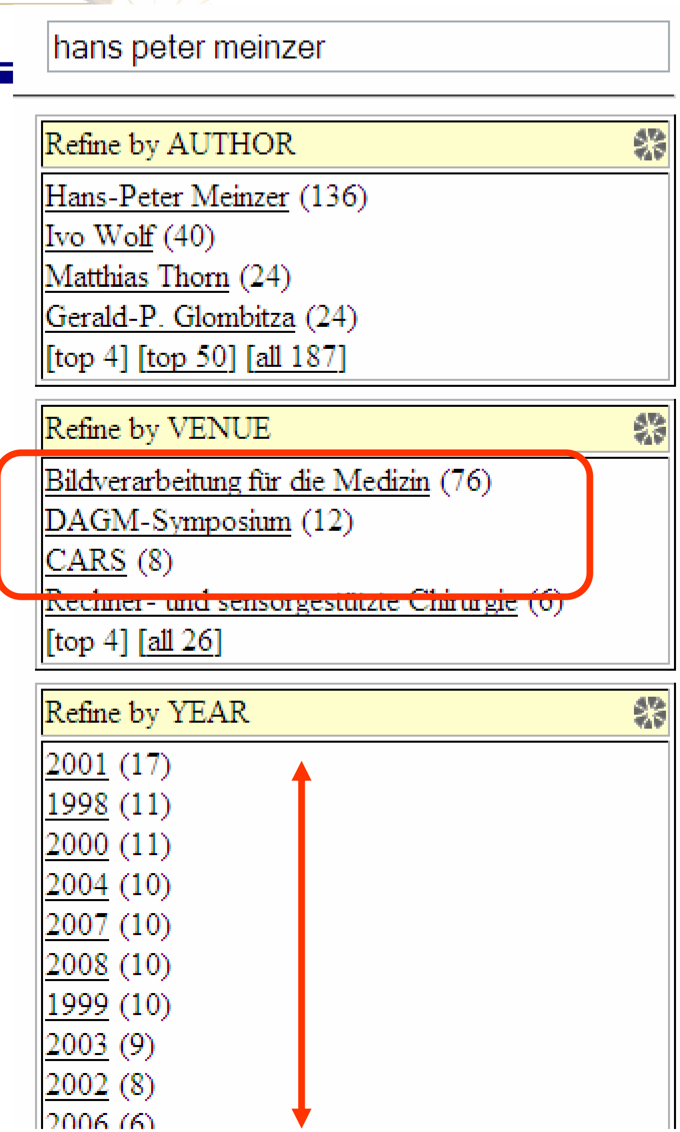

$\mathbf{a}_r$ $\mathbf{b}_r$ $\mathbf{c}_r$

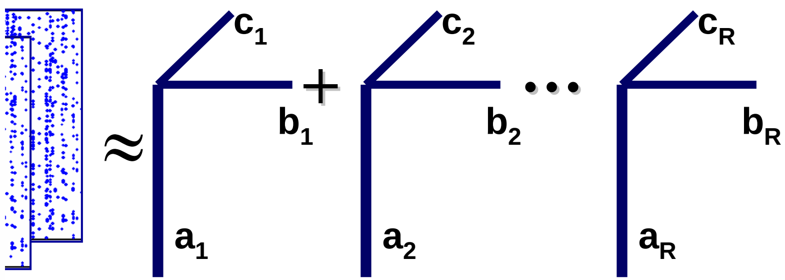


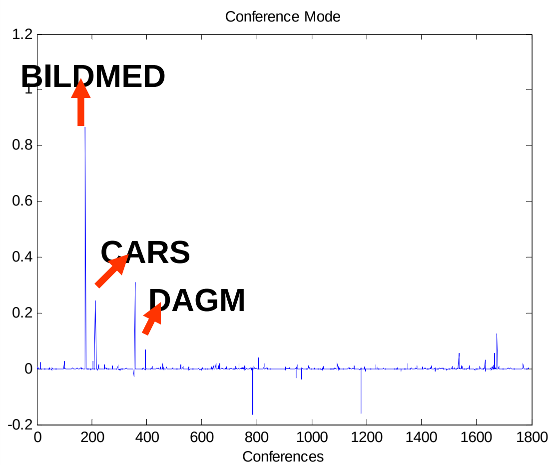
# Components make sense!

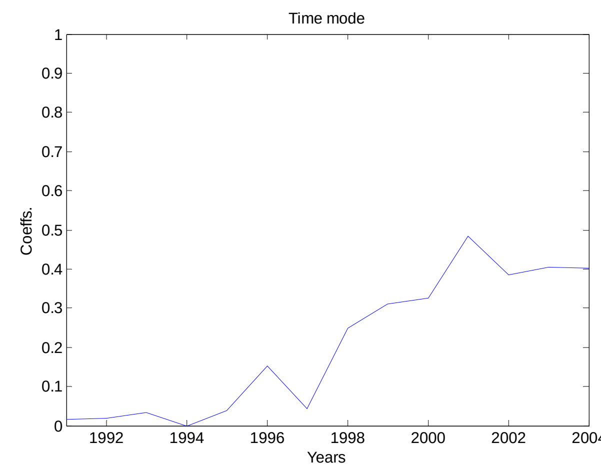


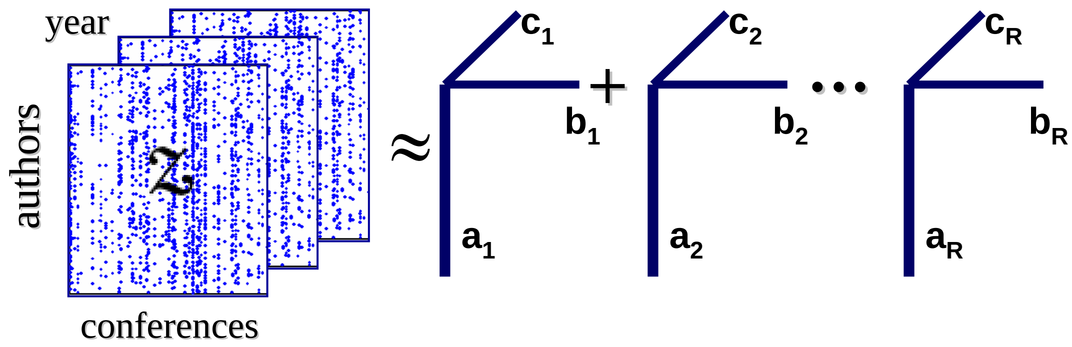

$b_r$

$c_r$

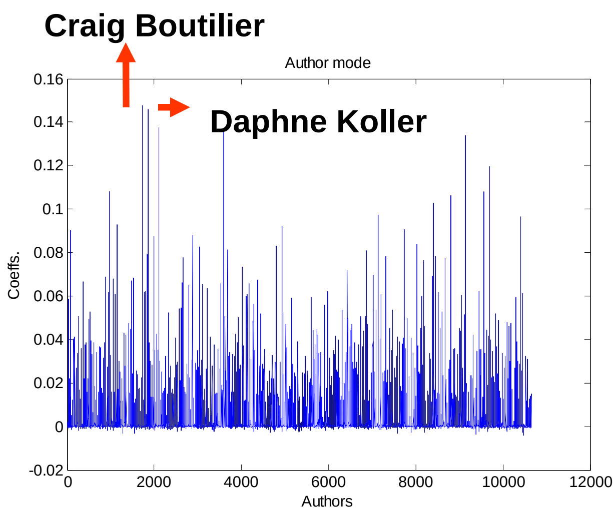


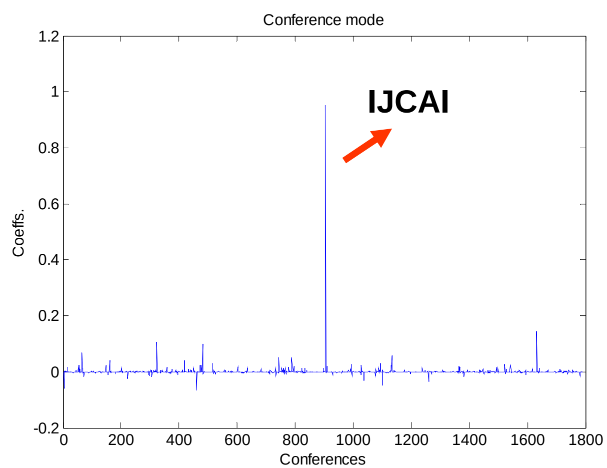



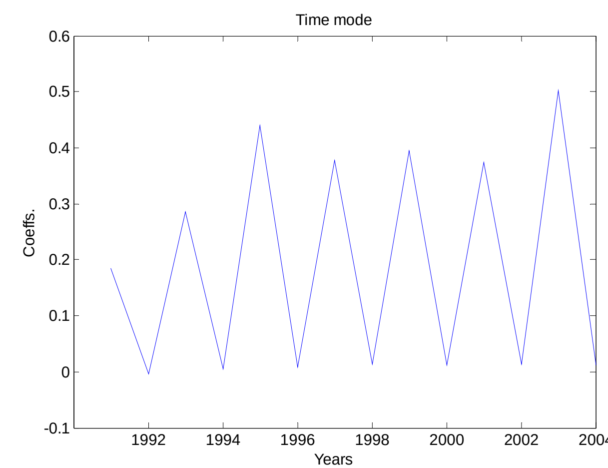
# Components make sense!



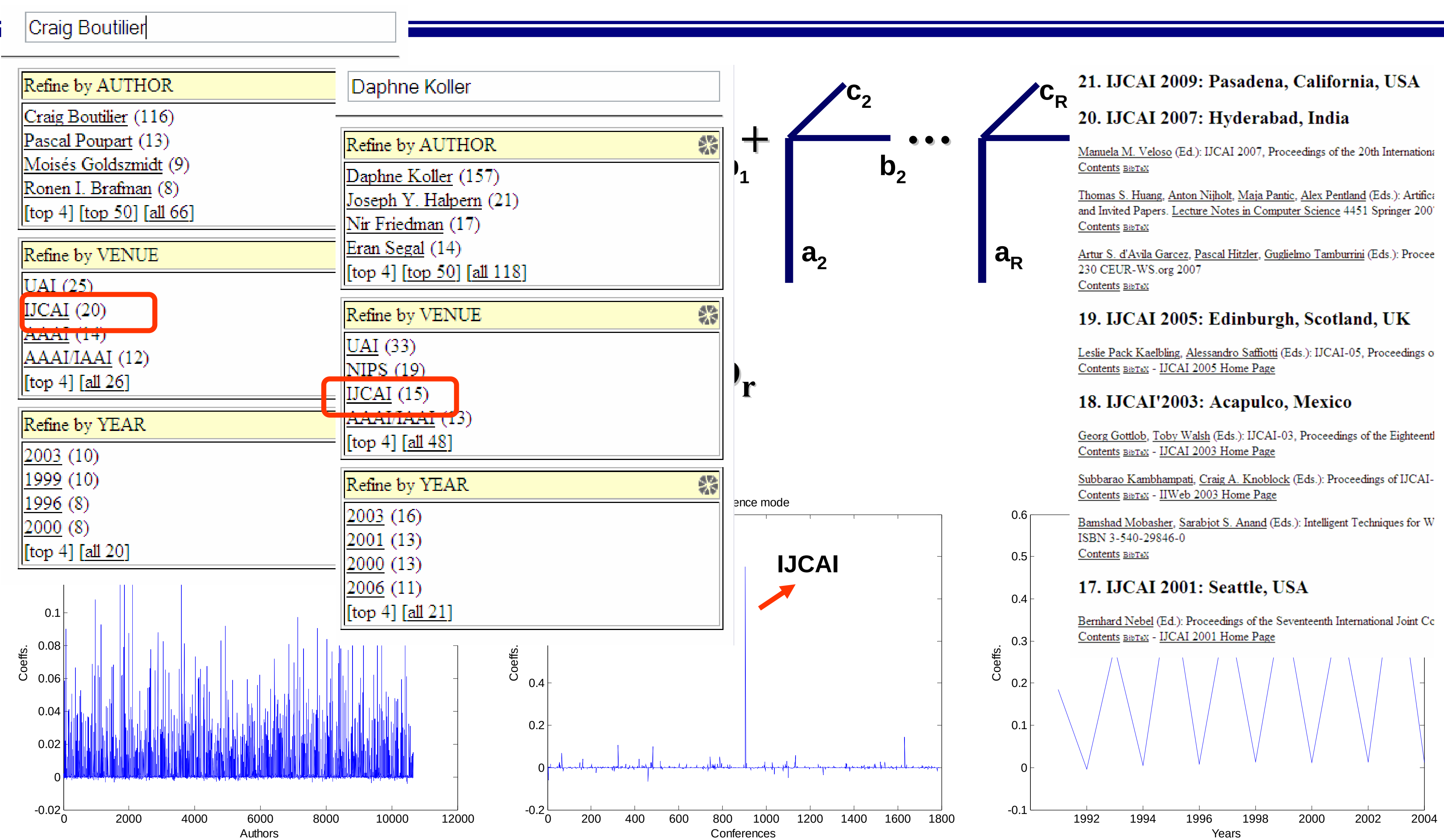

$\mathbf{a}_r$ $\mathbf{b}_r$ $\mathbf{c}_r$



# Components make sense!

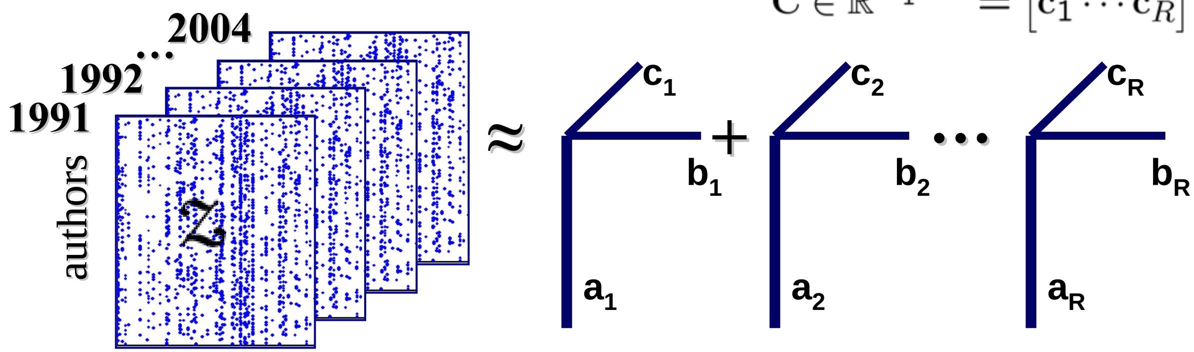


# Link Prediction Problem

**TRAIN:**

$\mathcal{Z} \in \mathbb{R}^{I \times J \times K}$

$\mathbf{C} \in \mathbb{R}^{K_1 \times R} = [\mathbf{c}_1 \cdots \mathbf{c}_R]$

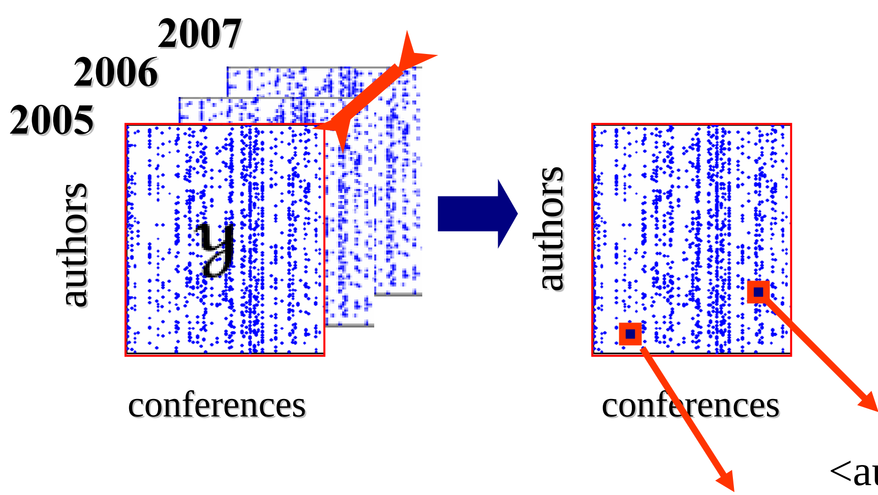



$\mathbf{B} \in \mathbb{R}^{J \times R} = [\mathbf{b}_1 \cdots \mathbf{b}_R]$

$\mathbf{A} \in \mathbb{R}^{I \times R} = [\mathbf{a}_1 \cdots \mathbf{a}_R]$

**TEST:**

$\mathcal{Y} \in \mathbb{R}^{I \times J \times \hat{K}}$




~ 60K links out of 19 million possible <author, conf> pairs

~ 0.3% dense

~ 32K previously unseen links in the training set

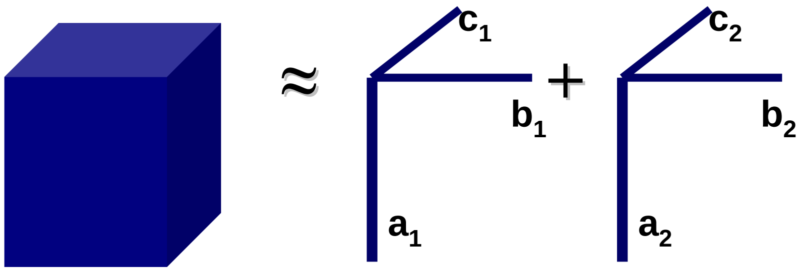
$\langle author_i, conf_j \rangle = 0$

$\langle author_i, conf_j \rangle = 1$

if $i^{th}$ author publishes at $j^{th}$ conf.

# Score for <author$_i$, conf$_j$>

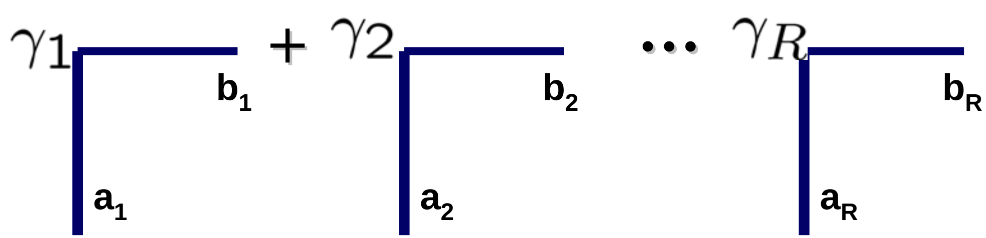
- Sign ambiguity:

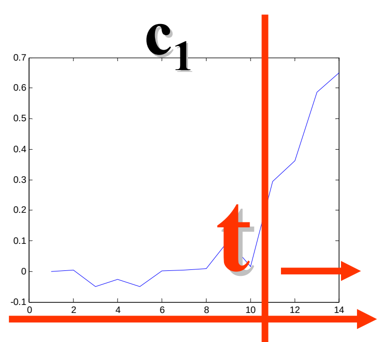



$$\begin{aligned}\mathcal{Z} &\approx \sum_{r=1}^{2} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \\ &\approx (-\mathbf{a}_1) \circ (-\mathbf{b}_1) \circ \mathbf{c}_1 + \mathbf{a}_2 \circ (-\mathbf{b}_2) \circ (-\mathbf{c}_2) \\ &\approx (-\mathbf{a}_1) \circ \mathbf{b}_1 \circ (-\mathbf{c}_1) + (-\mathbf{a}_2) \circ (-\mathbf{b}_2) \circ \mathbf{c}_2\end{aligned}$$

- Fix signs using the signs of the maximum magnitude entries and then compute a score for each author-conference pair using the information from the time domain:

$$\text{SCORE}_{ij} = \sum_{r=1}^{R} \gamma_r \mathbf{a}_{ir} \mathbf{b}_{jr}$$

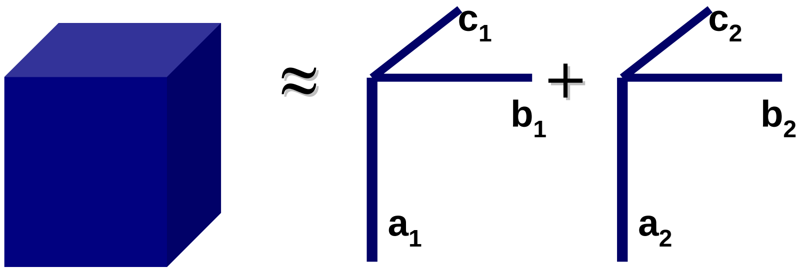



$$\gamma_1 = \sum_{k=t}^{K} c_{k1}$$

# Score for <author$_i$, conf$_j$>

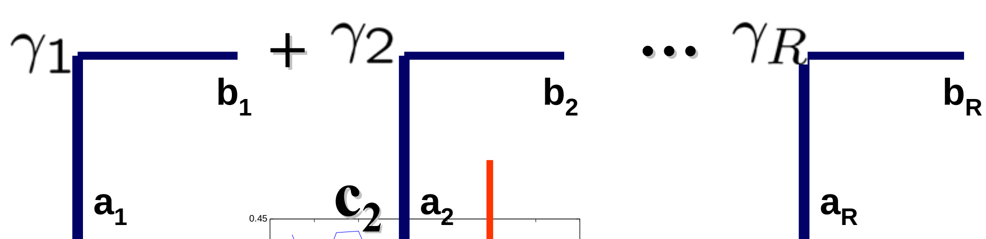
- Sign ambiguity:

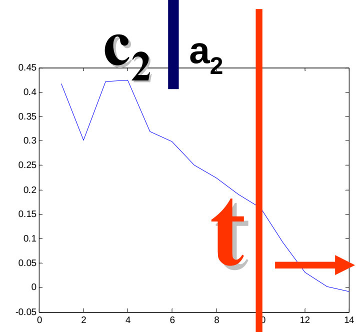



$$\mathcal{Z} \approx \sum_{r=1}^{2} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$
$$\approx (-\mathbf{a}_1) \circ (-\mathbf{b}_1) \circ \mathbf{c}_1 + \mathbf{a}_2 \circ (-\mathbf{b}_2) \circ (-\mathbf{c}_2)$$
$$\approx (-\mathbf{a}_1) \circ \mathbf{b}_1 \circ (-\mathbf{c}_1) + (-\mathbf{a}_2) \circ (-\mathbf{b}_2) \circ \mathbf{c}_2$$

- Fix signs using the signs of the maximum magnitude entries and then compute a score for each author-conference pair using the information from the time domain:

$$\text{SCORE}_{ij} = \sum_{r=1}^{R} \gamma_r \mathbf{a}_{ir} \mathbf{b}_{jr}$$

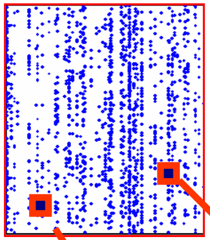



$$\gamma_2 = \sum_{k=t}^{K} c_{k2}$$

# Performance Measure: AUC

s: contains the scores for all possible pairs, e.g., ~19 million




<author$_i$, conf$_j$> = 0

<author$_i$, conf$_j$> = 1
if i$^{th}$ author publishes at j$^{th}$ conf.

**scores** | **sorted scores** | **labels**

$$\begin{bmatrix} s_{11} \\ s_{12} \\ \cdots \\ s_{ij} \\ \cdots \\ s_{IJ} \end{bmatrix} \xrightarrow{\textbf{sort}} \begin{bmatrix} s_{95} \\ s_{23} \\ \cdots \\ \cdots \\ \cdots \\ s_{67} \end{bmatrix} \quad \begin{bmatrix} 1 \\ 0 \\ \cdots \\ 1 \\ \cdots \\ 0 \end{bmatrix}$$

*N*: number of 1's
*M*: number of 0's

# Performance Measure: AUC

s: contains the scores for all possible pairs, e.g., ~19 million

**scores**

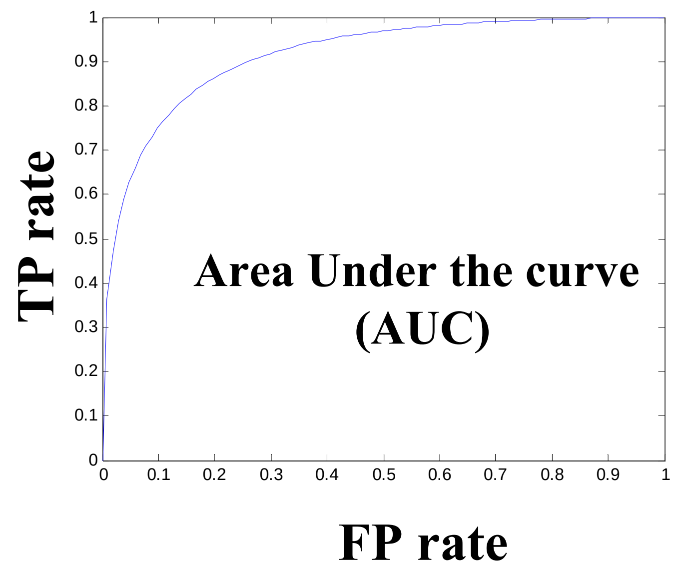
$$\begin{bmatrix} s_{11} \\ s_{12} \\ \cdots \\ s_{ij} \\ \cdots \\ s_{IJ} \end{bmatrix}$$

***sort*** →

| sorted scores | labels | TP rate | FP rate |
|---|---|---|---|
| $s_{95}$ | 1 | 1/N | 0 |
| $s_{23}$ | 0 | 1/N | 1/M |
| …. | …. | ⋮ | ⋮ |
| …. | 1 | | |
| …. | …. | | |
| $s_{67}$ | 0 | 1 | 1 |

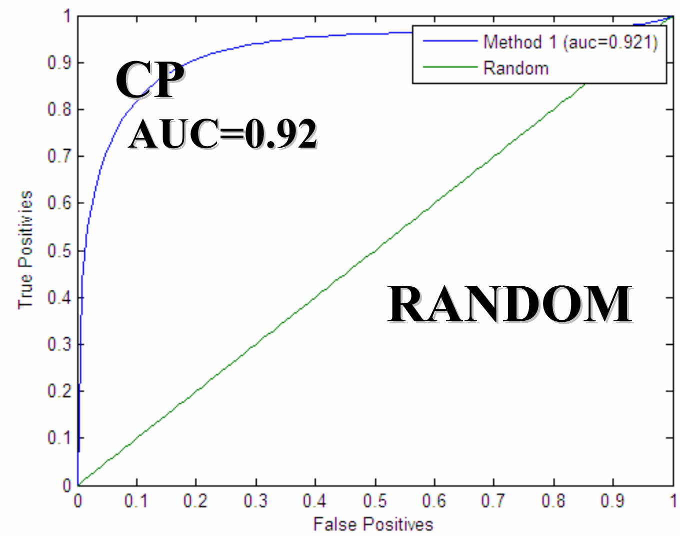
$N$: number of 1's
$M$: number of 0's

**Receiver Operating Characteristic (ROC) Curve**

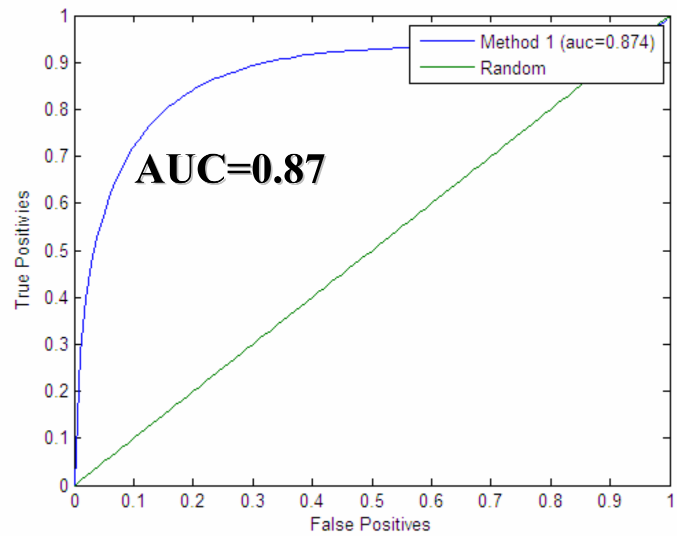


# Performance Evaluation

Predicting Links
for 2005 - 2007 (~ 60K):




Predicting **Previously Unseen Links**
for 2005 - 2007(~ 32K):



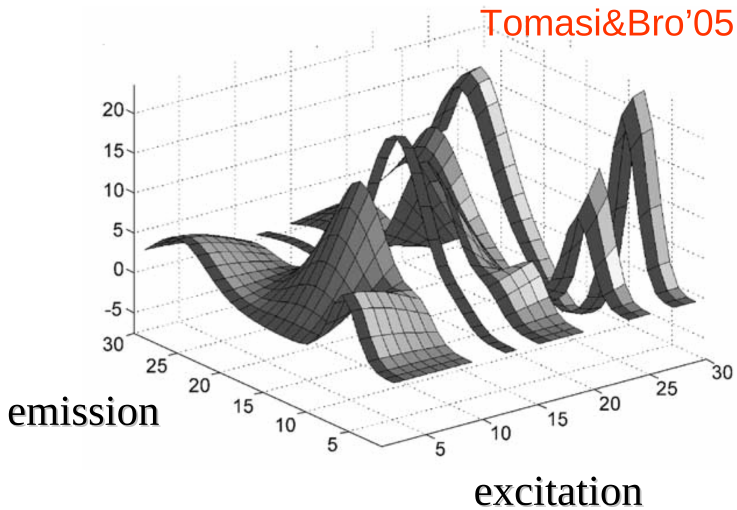
# CP-WOPT: Handling Missing Data

# Missing Data Examples

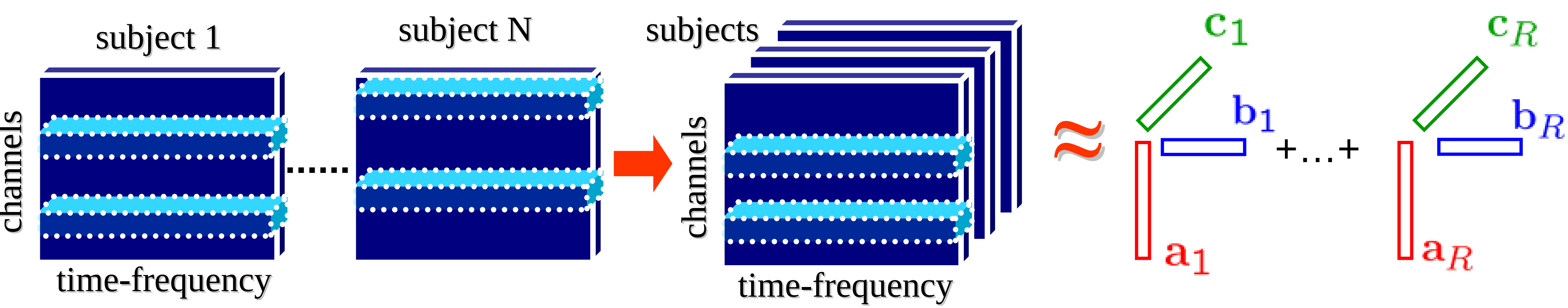
Missing data in different disciplines due to loss of information, machine failures, different sampling frequencies or experimental-set ups.

- Chemometrics
- Biomedical signal processing (e.g., EEG)
- Network traffic analysis (e.g., packet drops)
- Computer vision (e.g., occlusions)
- …

**CHEMISTRY**




**EEG**



# Modify the objective for CP

**FOR HANDLING MISSING DATA**

**NO MISSING DATA**

**Optimization Problem**

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \| \mathcal{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!] \|^2$$

**Optimization Problem**

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} \| \mathcal{W} * (\mathcal{Z} - [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!]) \|^2$$

$$w_{ijk} = \begin{cases} 1 & \text{if } z_{ijk} \text{ is known,} \\ 0 & \text{if } z_{ijk} \text{ is missing.} \end{cases}$$

**Objective Function**

$$f_{\mathcal{W}}(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \left\{ w_{ijk} \left( z_{ijk} - \sum_{r=1}^{R} a_{ir} b_{jr} c_{kr} \right) \right\}^2$$

# Our approach: CP-WOPT

**Objective Function**

$$f_{\mathcal{W}}(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\left\{w_{ijk}\left(z_{ijk} - \sum_{r=1}^{R} a_{ir}b_{jr}c_{kr}\right)\right\}^2$$

$$f_{\mathcal{W}} : \mathbb{R}^{(I+J+K)R} \mapsto \mathbb{R}$$

$$\mathbf{x} = \begin{bmatrix} a_{11} \\ \vdots \\ a_{IR} \\ b_{11} \\ \vdots \\ b_{JR} \\ c_{11} \\ \vdots \\ c_{KR} \end{bmatrix}$$

$$\nabla f_{\mathcal{W}}(\mathbf{x}) = \begin{bmatrix} \frac{\partial f_{\mathcal{W}}}{\partial a_{11}} \\ \vdots \\ \frac{\partial f_{\mathcal{W}}}{\partial a_{IR}} \\ \frac{\partial f_{\mathcal{W}}}{\partial b_{11}} \\ \vdots \\ \frac{\partial f_{\mathcal{W}}}{\partial b_{JR}} \\ \frac{\partial f_{\mathcal{W}}}{\partial c_{11}} \\ \vdots \\ \frac{\partial f_{\mathcal{W}}}{\partial c_{KR}} \end{bmatrix}$$

# Objective and Gradient

**Objective Function**

$$f_{\mathcal{W}}(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\left\{w_{ijk}\left(z_{ijk}-\sum_{r=1}^{R}a_{ir}b_{jr}c_{kr}\right)\right\}^2$$

**Gradient** (for *r* = *1,…,R; i=1,…I; j=1,..J; k=1,..K* )

$$\frac{\partial f_{\mathcal{W}}}{\partial a_{ir}} = -2\sum_{k=1}^{K}\sum_{j=1}^{J} w_{ijk}z_{ijk}b_{jr}c_{kr} + 2\sum_{k=1}^{K}\sum_{j=1}^{J} w_{ijk}\left(\sum_{l=1}^{R} a_{il}b_{jl}c_{kl}\right)b_{jr}c_{kr}$$

$$\frac{\partial f_{\mathcal{W}}}{\partial b_{jr}} = -2\sum_{k=1}^{K}\sum_{i=1}^{I} w_{ijk}z_{ijk}a_{ir}c_{kr} + 2\sum_{k=1}^{K}\sum_{i=1}^{I} w_{ijk}\left(\sum_{l=1}^{R} a_{il}b_{jl}c_{kl}\right)a_{ir}c_{kr}$$

$$\frac{\partial f_{\mathcal{W}}}{\partial c_{kr}} = -2\sum_{j=1}^{J}\sum_{i=1}^{I} w_{ijk}z_{ijk}a_{ir}b_{jr} + 2\sum_{j=1}^{J}\sum_{i=1}^{I} w_{ijk}\left(\sum_{l=1}^{R} a_{il}b_{jl}c_{kl}\right)a_{ir}b_{jr}$$

# Gradient in Matrix Form

**Objective Function**

$$f_{\mathcal{W}}(\mathbf{A},\mathbf{B},\mathbf{C}) = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{k=1}^{K}\left\{w_{ijk}\left(z_{ijk} - \sum_{r=1}^{R} a_{ir}b_{jr}c_{kr}\right)\right\}^2$$

$$\mathcal{X} = \mathcal{W} * [\![\mathbf{A},\mathbf{B},\mathbf{C}]\!], \mathcal{Y} = \mathcal{W} * \mathcal{Z}$$

**Gradient**

$$\frac{\partial f_{\mathcal{W}}}{\partial \mathbf{A}} = 2(\mathbf{X}_{(1)} - \mathbf{Y}_{(1)})(\mathbf{C} \odot \mathbf{B})$$

$$\frac{\partial f_{\mathcal{W}}}{\partial \mathbf{B}} = 2(\mathbf{X}_{(2)} - \mathbf{Y}_{(2)})(\mathbf{C} \odot \mathbf{A})$$

$$\frac{\partial f_{\mathcal{W}}}{\partial \mathbf{C}} = 2(\mathbf{X}_{(3)} - \mathbf{Y}_{(3)})(\mathbf{B} \odot \mathbf{A})$$

# Experimental Set-Up [Tomasi&Bro'05]

*20 triplets*

**Step 1**: Generate random factor matrices A, B, C with $R$ = 5 or 10 columns each and collinearity set to 0.5.

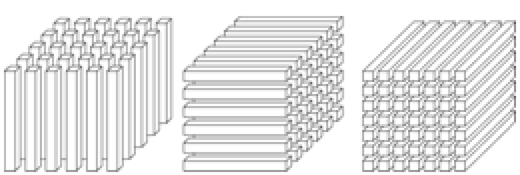
**Step 2**: Construct tensor from factor matrices and add noise ( 2% homoscedastic noise)

$$\mathcal{Z} = [\![ \mathbf{A}, \mathbf{B}, \mathbf{C} ]\!] + \mathcal{N}$$

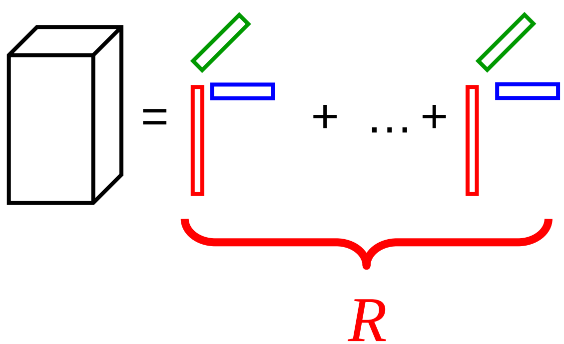
**Step 3**: Set some entries to missing

- Percentage of Missing Data: 10%, 40%, 70%

*Missing: **entries, fibers***



**Step 4**: Use algorithm to extract $R$ factors. Compare against factors in Step 1.



# CP-WOPT is Accurate!

Generated 40 test problems (with ranks 5 and 10) and factorized with an *R*-component CP model. Each entry corresponds to the percentage of correctly recovered solutions.

| | Accuracy (Randomly Missing Entries) | | | | | |
|---|---|---|---|---|---|---|
| | $M = 10\%$ | | $M = 40\%$ | | $M = 70\%$ | |
| **Size** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** |
| $50 \times 50 \times 50$ | 100.0 | 100.0 | 100.0 | 100.0 | 90.0 | 100.0 |
| $150 \times 150 \times 150$ | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| | **Accuracy (Randomly Missing Fibers)** | | | | | |
| | $M = 10\%$ | | $M = 40\%$ | | $M = 70\%$ | |
| **Size** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** |
| $50 \times 50 \times 50$ | 100.0 | 100.0 | 92.5 | 100.0 | 22.5 | 82.5 |
| $150 \times 150 \times 150$ | 100.0 | 100.0 | 100.0 | 100.0 | 75.0 | 100.0 |

$$\frac{\#\ \text{known data entries}}{\#\ \text{variables}}$$

# CP-WOPT is Accurate!

Generated 40 test problems (with ranks 5 and 10) and factorized with an *R*-component CP model. Each entry corresponds to the percentage of correctly recovered solutions.

| | Accuracy (Randomly Missing Entries) | | | | | |
|---|---|---|---|---|---|---|
| | $M = 10\%$ | | $M = 40\%$ | | $M = 70\%$ | |
| **Size** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** |
| $50 \times 50 \times 50$ | 100.0 | 100.0 | 100.0 | 100.0 | 90.0 | 100.0 |
| $150 \times 150 \times 150$ | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| | **Accuracy (Randomly Missing Fibers)** | | | | | |
| | $M = 10\%$ | | $M = 40\%$ | | $M = 70\%$ | |
| **Size** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** |
| $50 \times 50 \times 50$ | 100.0 | 100.0 | 92.5 | 100.0 | 22.5 | 82.5 |
| $150 \times 150 \times 150$ | 100.0 | 100.0 | 100.0 | 100.0 | 75.0 | 100.0 |

**CPNLS** : Nonlinear least squares. Used INDAFAC, which implements Levenberg-Marquadt [Tomasi and Bro'05].

Other alternative: **ALS-based imputation** (For comparisons, see Tomasi and Bro'05).

# CP-WOPT is Fast!

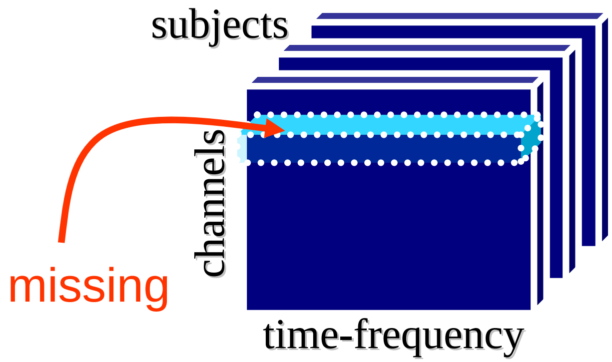
Generated 60 test problems (with *M* =10%, 40% and 70%) and factorized with an *R*-component CP model. Each entry corresponds to the average/std of the CP models, which successfully recover the underlying factors.

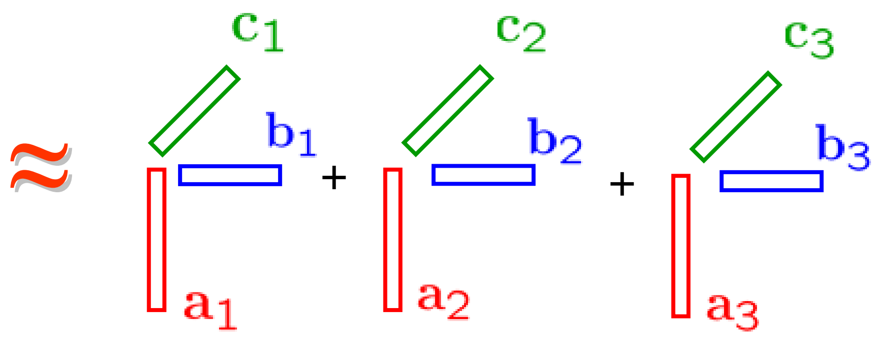
| | Time (sec) (Randomly Missing Entries) | | | |
|---|---|---|---|---|
| | $R = 5$ | | $R = 10$ | |
| **Size** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** |
| $50 \times 50 \times 50$ | $6.1 \pm 1.3$ | $2.8 \pm 0.6$ | $21.7 \pm 4.5$ | $7.0 \pm 1.4$ |
| $150 \times 150 \times 150$ | $218.9 \pm 42.7$ | $107.6 \pm 20.3$ | $808.1 \pm 153.4$ | $290.5 \pm 46.3$ |
| | **Time (sec) (Randomly Missing Fibers)** | | | |
| | $R = 5$ | | $R = 10$ | |
| **Size** | **CPNLS** | **CP-WOPT** | **CPNLS** | **CP-WOPT** |
| $50 \times 50 \times 50$ | $5.9 \pm 1.8$ | $3.0 \pm 1.1$ | $20.4 \pm 3.8$ | $7.4 \pm 2.0$ |
| $150 \times 150 \times 150$ | $216.6 \pm 43.1$ | $94.7 \pm 18.9$ | $720.2 \pm 156.8$ | $265.4 \pm 47.1$ |

# CP-WOPT is useful for real data!

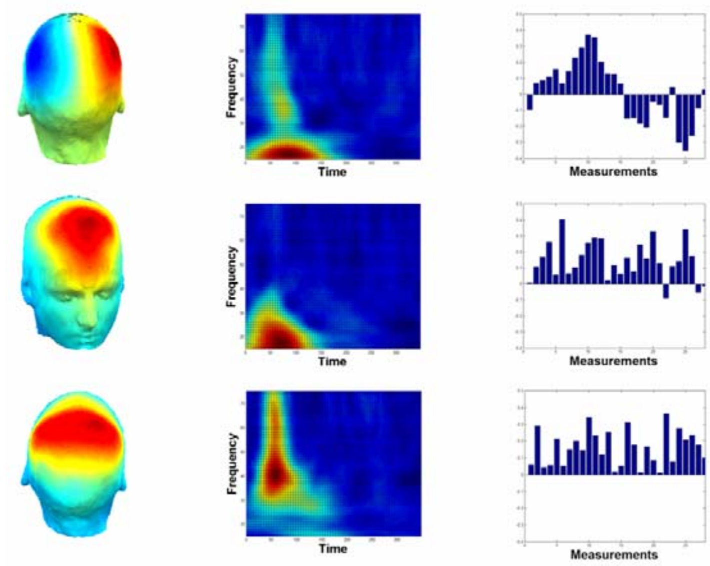
Thanks to Morten Mørup!

**GOAL:** To differentiate between left and right hand stimulation

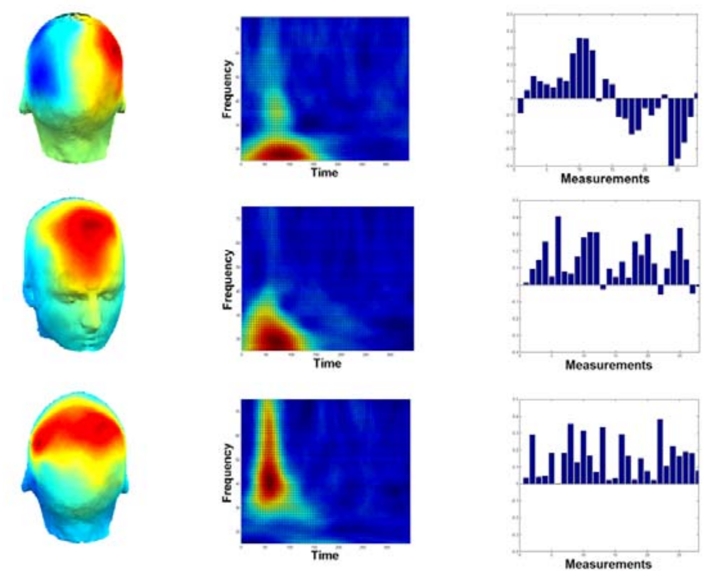



**COMPLETE DATA**

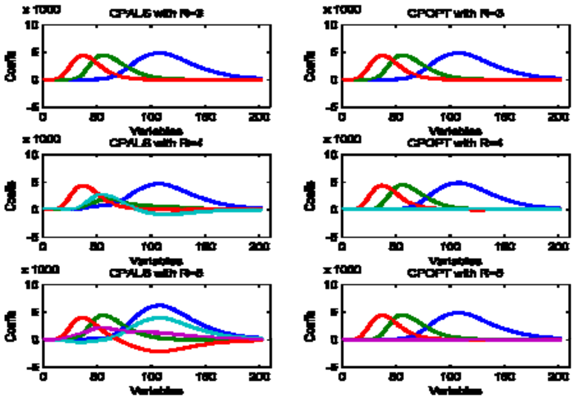


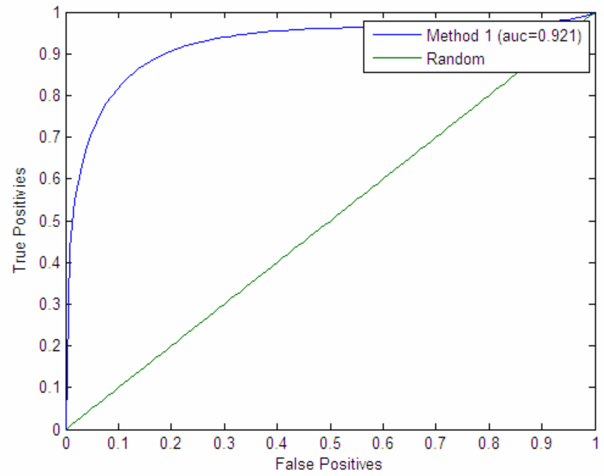

**INCOMPLETE DATA**



# Summary & Future Work

- New CPOPT method
  - Accurate & scalable
- Extend CPOPT to CP-WOPT to handle missing data
  - Accurate & scalable
- More open questions…
  - Starting point?
  - Tuning the optimization
  - Regularization
  - Exploiting sparsity
  - Nonnegativity
- Application to link prediction
  - On-going work comparing to other methods



# Thank you!

- **More on tensors and tensor models:**
  - **Survey** : E. Acar and B. Yener, Unsupervised Multiway Data Analysis: A Literature Survey, *IEEE Transactions on Knowledge and Data Engineering*, 21(1): 6-20, 2009.
  - **CPOPT** : E. Acar, T. G. Kolda and D. M. Dunlavy, An Optimization Approach for Fitting Canonical Tensor Decompositions, *Submitted for publication.*
  - **CP-WOPT :** E. Acar, T.G. Kolda, D. M. Dunlavy and M. Mørup, Tensor Factorizations with Missing Data, *Submitted for publication*.
  - **Link Prediction:** E. Acar, T.G. Kolda and D. M. Dunlavy, Link Prediction on Evolving Data, *in preparation.*
- **Contact:**

  - Evrim Acar, **eacarat@sandia.gov**
  - Tamara G. Kolda, **tgkolda@sandia.gov**
  - Daniel M. Dunlavy, **dmdunla@sandia.gov**


2009 SIAM Annual Meeting
July 6-10, 2009
Sheraton Denver Downtown Hotel
Denver, Colorado

**Minisymposia on**
**Tensors and Tensor-based Computations**