Illustration by Chris Brigman

Tensor Decomposition for Bernoulli Data

Tamara G. Kolda
Sandia National Laboratories
Livermore, CA

SIAM Conf. on Computational Science & Engineering (CSE17)
MS236: Tensor Decompositions: Applications and Efficient Algorithms
March 2, 2017

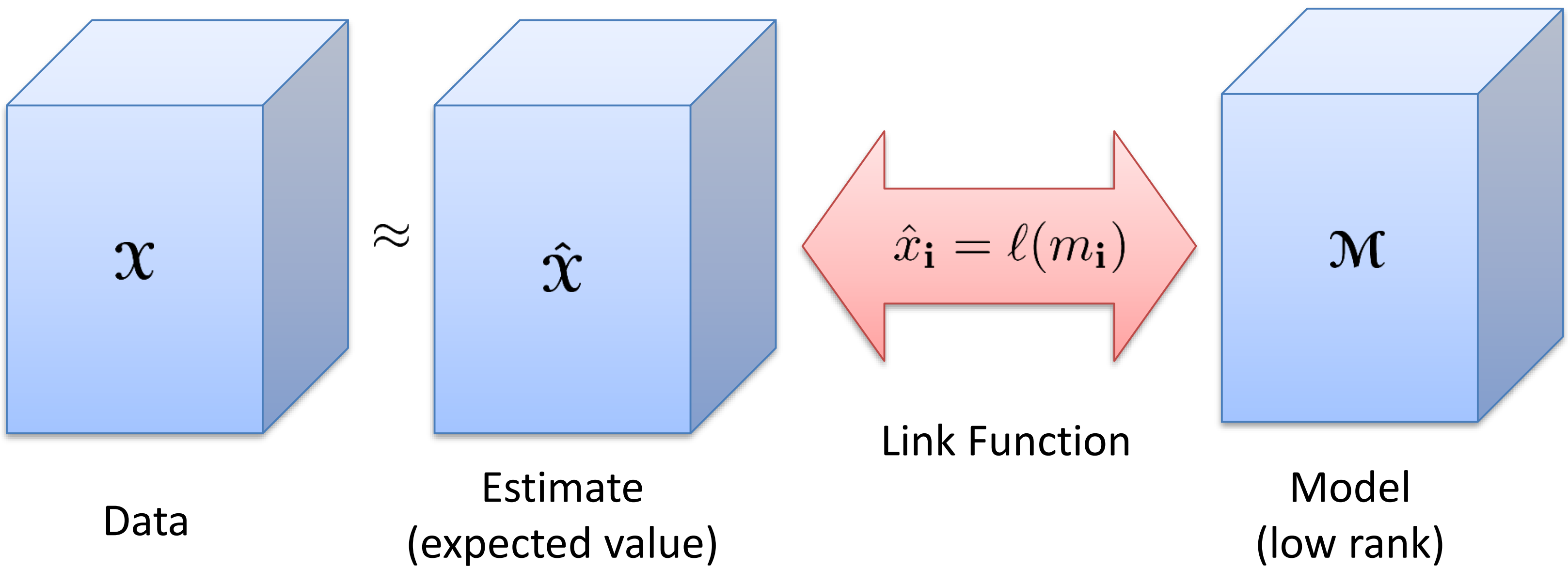
Sandia National Laboratories is a multi-mission laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Data, Estimates, Models, and Loss Functions

Sandia National Laboratories

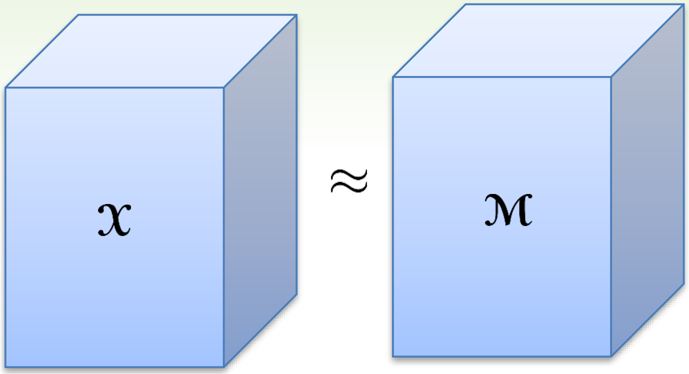
WLOG, all data are tensors of size $n_1 \times \cdots \times n_d$

Let $\mathcal{I} = \{ \mathbf{i} = (i_1, \ldots, i_d) \}$ denote the set of all indices



Loss function: $F(\mathcal{X}, \mathcal{M}) = \sum_{\mathbf{i} \in \mathcal{I}} f(x_{\mathbf{i}}, m_{\mathbf{i}})$ (sum of elementwise functions)

Sum of Square Error Assumes Normally Distributed Data



Data

Model
(low rank)

Typically: Consider data to be low-rank plus "white noise"

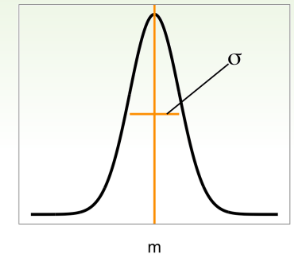
$$x_{\mathbf{i}} \sim m_{\mathbf{i}} + \mathcal{N}(0, \sigma)$$

Equivalently, Gaussian with mean $m_{\mathbf{i}}$

$$x_{\mathbf{i}} \sim \mathcal{N}(m_{\mathbf{i}}, \sigma)$$

Gaussian Probability Density Function (PDF)

$$\frac{e^{-(x-m)^2/2\sigma^2}}{\sqrt{2\pi\sigma^2}}$$



Want to maximize likelihood of model:

$$L(\mathcal{M}) = \prod_{\mathbf{i} \in \mathcal{I}} \frac{e^{-(x_{\mathbf{i}}-m_{\mathbf{i}})^2/2\sigma^2}}{\sqrt{2\pi\sigma^2}}$$

Equivalent to minimizing negative log likelihood:

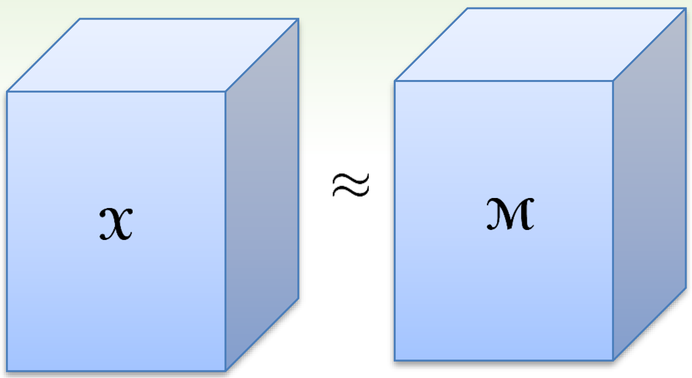
$$-\log(L(\mathcal{M})) = \sum_{\mathbf{i} \in \mathcal{I}} \frac{(x_{\mathbf{i}} - m_{\mathbf{i}})^2}{2\sigma^2} + \frac{1}{2}\log(2\pi\sigma^2)$$

Assume σ is constant, so left with SSE!

$$F(\mathcal{X}, \mathcal{M}) = \sum_{\mathbf{i} \in \mathcal{I}} \underbrace{(x_{\mathbf{i}} - m_{\mathbf{i}})^2}_{f(x_{\mathbf{i}}, m_{\mathbf{i}})} \sim \sum_{\mathbf{i} \in \mathcal{I}} \frac{1}{2} m_{\mathbf{i}}^2 - x_{\mathbf{i}} m_{\mathbf{i}}$$

KL Divergence Assumes Poisson Distributed Data

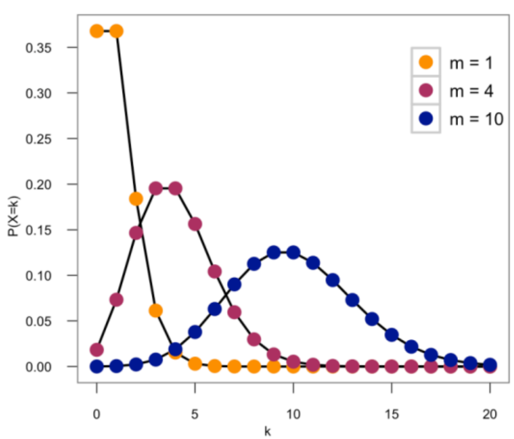
Sandia National Laboratories



Data

Model (low rank)

$$x_{\mathbf{i}} \sim \text{Poisson}(m_{\mathbf{i}}),\ m_{\mathbf{i}} \geq 0$$



Poisson Probability Mass Function (PMF) $\dfrac{e^{-m}\, m^{x}}{x!}$

Want to maximize likelihood of model:

$$L(\mathcal{M}) = \prod_{\mathbf{i} \in \mathcal{I}} \frac{e^{-m_{\mathbf{i}}} m_{\mathbf{i}}^{x_{\mathbf{i}}}}{x_{\mathbf{i}}!}$$

Equivalent to minimizing negative log likelihood:

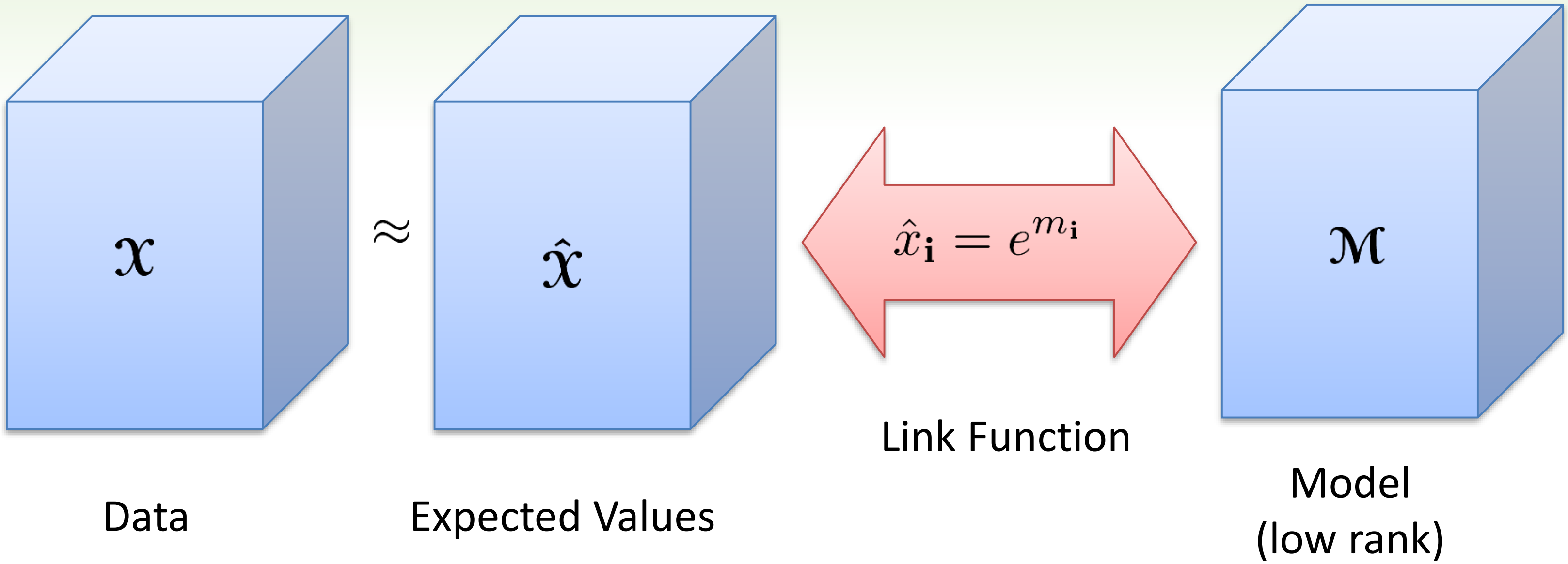
$$-\log(L(\mathcal{M})) = \sum_{\mathbf{i} \in \mathcal{I}} m_{\mathbf{i}} - x_{\mathbf{i}} \log m_{\mathbf{i}} + \log(x_{\mathbf{i}}!)$$

Remove constant term and left with KL divergence!

$$F(\mathcal{X}, \mathcal{M}) = \sum_{\mathbf{i} \in \mathcal{I}} \underbrace{m_{\mathbf{i}} - x_{\mathbf{i}} \log m_{\mathbf{i}}}_{f(x_{\mathbf{i}}, m_{\mathbf{i}})}$$

Chi & Kolda, SIMAX 2012

Alternative for Poisson Distributed Data: Log-Poisson



$$x_{\mathbf{i}} \sim \text{Poisson}(e^{m_{\mathbf{i}}}), \; m_{\mathbf{i}} \in \mathbb{R}$$

$$F(\mathcal{X}, \mathcal{M}) = \sum_{\mathbf{i} \in \mathcal{I}} \underbrace{e^{m_{\mathbf{i}}} - x_{\mathbf{i}} m_{\mathbf{i}}}_{f(x_{\mathbf{i}}, m_{\mathbf{i}})}$$

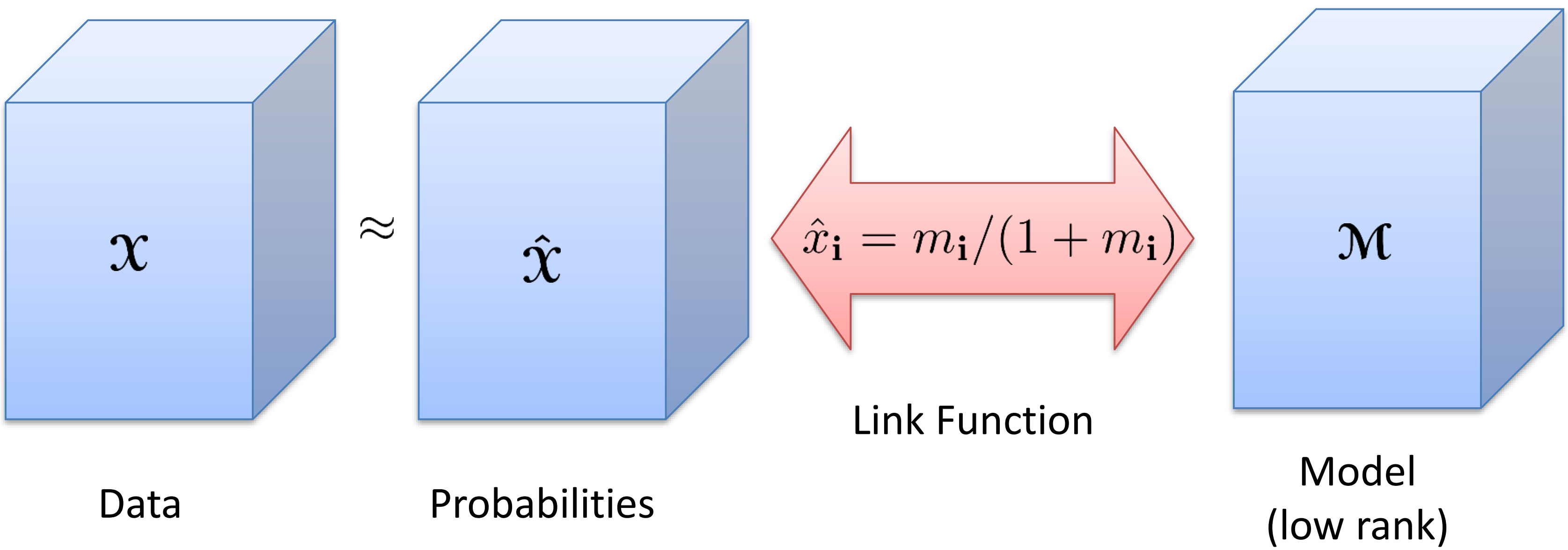
Contrast with previous slide

$$F(\mathcal{X}, \mathcal{M}) = \sum_{\mathbf{i} \in \mathcal{I}} m_{\mathbf{i}} - x_{\mathbf{i}} \log m_{\mathbf{i}}$$

Loss Function for Bernoulli Data

Bernoulli Probability Mass Function (PMF)

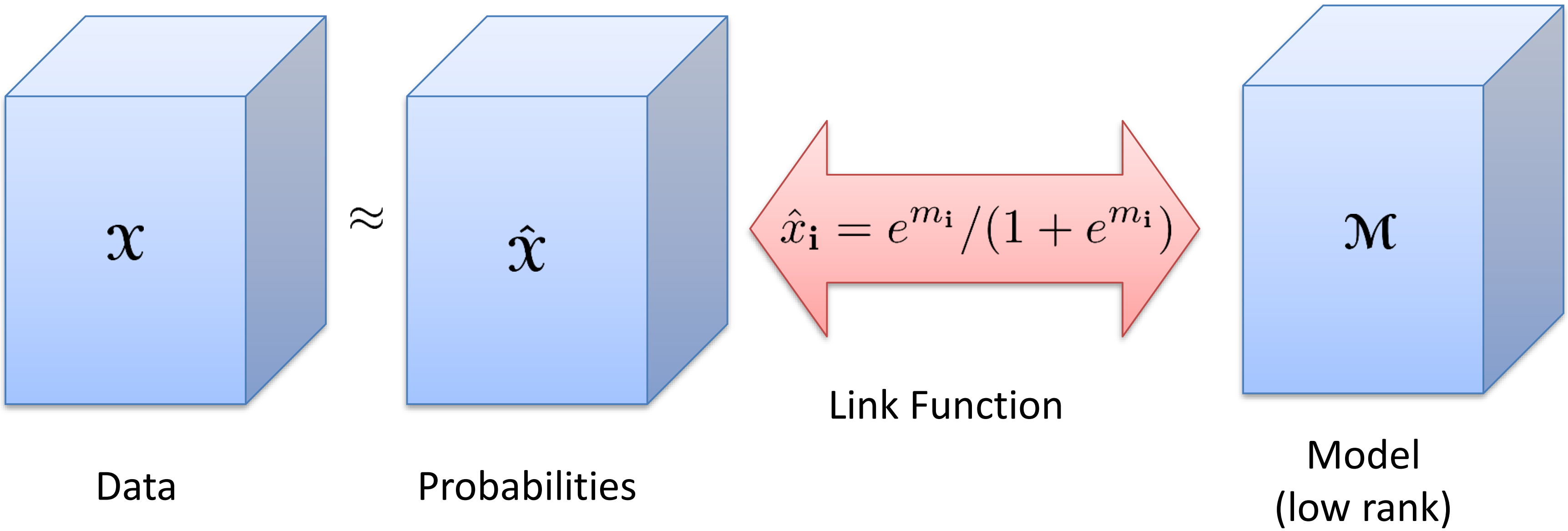
$$p^x(1-p)^{(1-x)},\ p \in (0,1)$$



$$x_{\mathbf{i}} \sim \text{Bernoulli}(m_{\mathbf{i}}/(1+m_{\mathbf{i}})),\ \ m_{\mathbf{i}} \geq 0$$

Alternative Loss Function for Bernoulli Data (Logit)

Bernoulli Probability Mass Function (PMF) $p^x(1-p)^{(1-x)},\ p \in (0,1)$

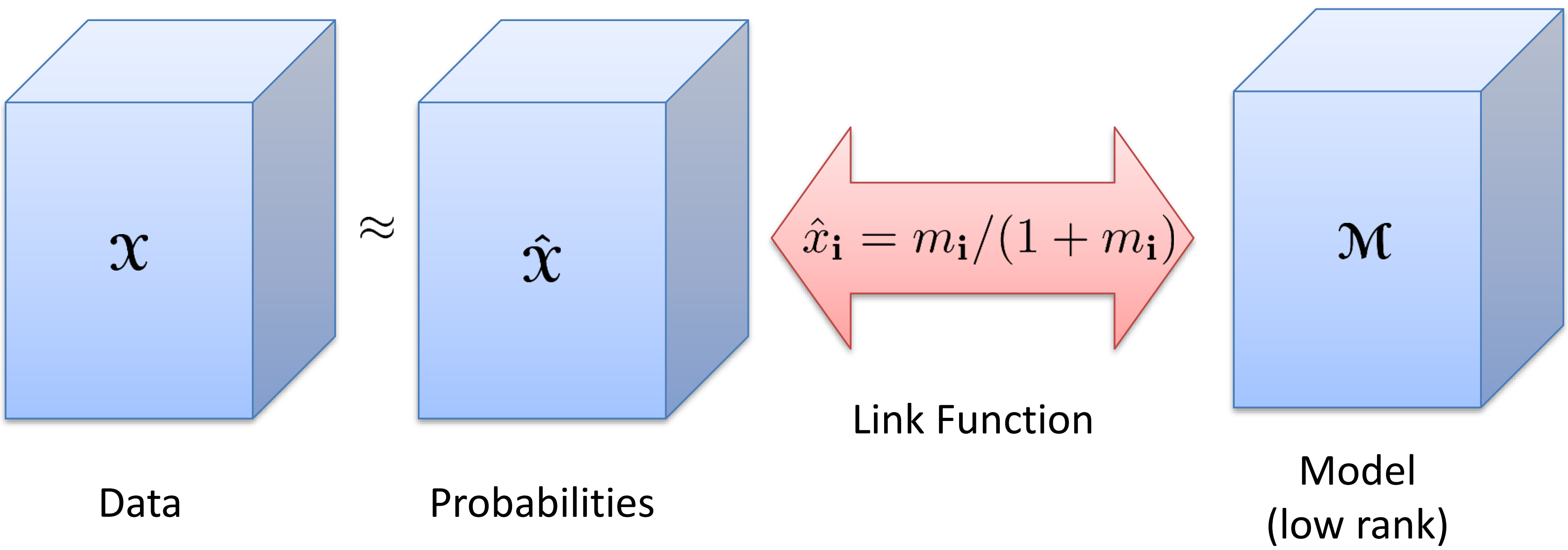


$$x_{\mathbf{i}} \sim \text{Bernoulli}(e^{m_{\mathbf{i}}}/(1+e^{m_{\mathbf{i}}})),\ m_{\mathbf{i}} \in \mathbb{R}$$

Loss Function for Bernoulli Data

Bernoulli Probability Mass Function (PMF)

$$p^x(1-p)^{(1-x)},\ p \in (0,1)$$



$$x_{\mathbf{i}} \sim \text{Bernoulli}(m_{\mathbf{i}}/(1+m_{\mathbf{i}})),\ \ m_{\mathbf{i}} \geq 0$$

Log-likelihood for Bernoulli

Bernoulli Probability Mass Function (PMF)

$$p^x(1-p)^{(1-x)},\; p \in (0,1)$$

$$x_{\mathbf{i}} \sim \text{Bernoulli}(m_{\mathbf{i}}/(1+m_{\mathbf{i}})),\;\; m_{\mathbf{i}} \geq 0$$

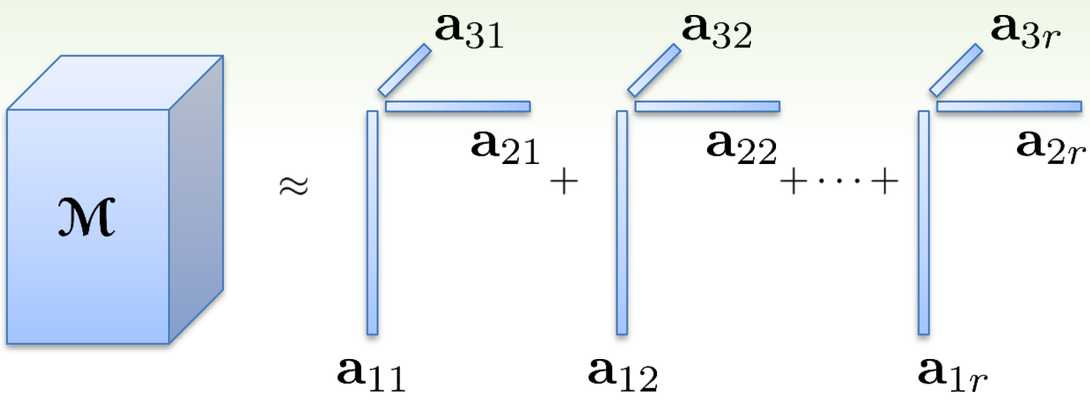
$$L(\mathcal{M}) = \prod_{\mathbf{i}\in\mathcal{I}} \left(\frac{m_{\mathbf{i}}}{1+m_{\mathbf{i}}}\right)^{x_{\mathbf{i}}} \left(1 - \frac{m_{\mathbf{i}}}{1+m_{\mathbf{i}}}\right)^{(1-x_{\mathbf{i}})}$$

$$-\log(L(\mathcal{M})) = \sum_{\mathbf{i}\in\mathcal{I}} -x_{\mathbf{i}} \log\left(\frac{m_{\mathbf{i}}}{1+m_{\mathbf{i}}}\right) - (1-x_{\mathbf{i}})\log\left(1 - \frac{m_{\mathbf{i}}}{1+m_{\mathbf{i}}}\right)$$

$$F(\mathcal{X},\mathcal{M}) = \sum_{\mathbf{i}\in\mathcal{I}} \underbrace{\log(m_{\mathbf{i}}+1) - x_{\mathbf{i}}\log m_{\mathbf{i}}}_{f(x_{\mathbf{i}},\, m_{\mathbf{i}})}$$

Low-Rank Multiway Model

Assume model has CP structure



Defined by d factor matrices: $\mathbf{A}_k = [\mathbf{a}_{k1} \cdots \mathbf{a}_{kr}] \in \mathbb{R}^{n_k \times r}$

Outer product expression: $\mathcal{M} = \sum_{j=1}^{r} \mathbf{a}_{1j} \circ \cdots \circ \mathbf{a}_{dj}$

Elementwise expression: $m_{\mathbf{i}} = \sum_{j=1}^{r} \mathbf{a}_{1j}(i_1) \cdots \mathbf{a}_{dj}(i_d)$

Shorthand: $\mathcal{M} = [\![\mathbf{A}_1, \ldots, \mathbf{A}_d]\!]$

Generalized Formulation

Minimize $F(\mathfrak{X}, \mathfrak{M}) = \sum_{\mathbf{i} \in \mathcal{I}} f(x_{\mathbf{i}}, m_{\mathbf{i}})$ subject to $\mathfrak{M} = [\![\mathbf{A}_1, \dots, \mathbf{A}_d]\!]$

Theorem: The partial derivative of F w.r.t. $\mathbf{A}_k$ is given by

$$\frac{\partial F}{\partial \mathbf{A}_k} = \mathbf{G}_{(k)} \left(\mathbf{A}_d \odot \cdots \odot \mathbf{A}_{k+1} \odot \mathbf{A}_{k-1} \odot \cdots \odot \mathbf{A}_1 \right)$$

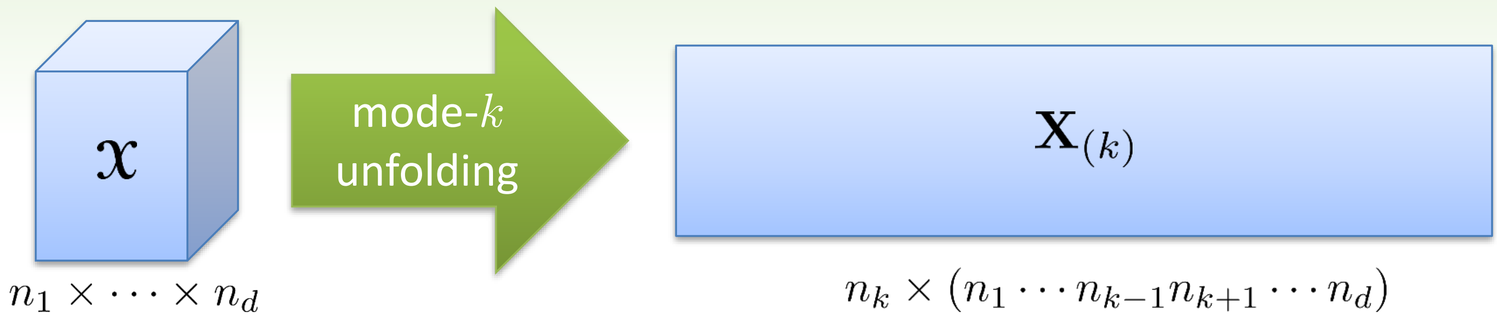
MTTRKP

where $\mathbf{G}_{(k)}$ is the mode-k unfolding of a tensor defined by elementwise by

$$g_{\mathbf{i}} = \frac{\partial f}{\partial m}(x_{\mathbf{i}}, m_{\mathbf{i}})$$

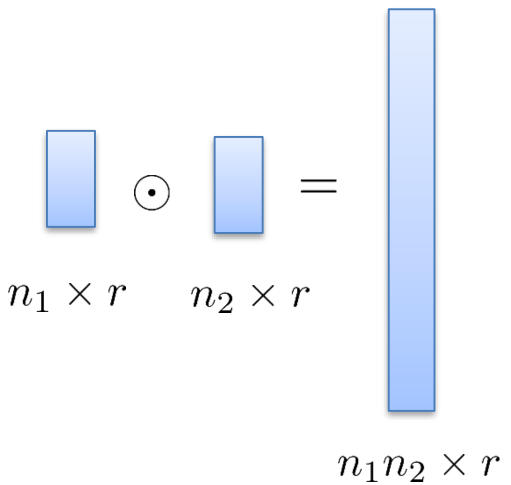
Easily extensible to the case of incomplete data, i.e., using a weight tensor.

Notation: Mode-k Unfolding, Khatri-Rao Product, MTTKRP



Khatri-Rao Product

Columnwise Kronecker Product



MTTKRP: matricized tensor times Khatri-Rao product

$$\mathbf{B} = \mathbf{X}_{(k)}(\mathbf{A}_d \odot \cdots \odot \mathbf{A}_{k+1} \odot \mathbf{A}_{k-1} \odot \cdots \odot \mathbf{A}_1)$$

Can exploit special structure in this computation, especially if the tensor is sparse.

Generalized Formulation with Missing Values

Minimize $F(\mathcal{X}, \mathcal{M}) = \sum_{\mathbf{i} \in \mathcal{I}} f(x_{\mathbf{i}}, m_{\mathbf{i}})$ subject to $\mathcal{M} = [\![\mathbf{A}_1, \dots, \mathbf{A}_d]\!]$

<u>Theorem</u>: The partial derivative of F w.r.t. $\mathbf{A}_k$ is given by

$$\frac{\partial F}{\partial \mathbf{A}_k} = \mathbf{G}_{(k)} \left(\mathbf{A}_d \odot \cdots \odot \mathbf{A}_{k+1} \odot \mathbf{A}_{k-1} \odot \cdots \odot \mathbf{A}_1 \right)$$

where $\mathbf{G}_{(k)}$ is the mode-k unfolding of a tensor defined by elementwise by

$$g_{\mathbf{i}} = \frac{\partial f}{\partial m}(x_{\mathbf{i}}, m_{\mathbf{i}})$$

Generalized Formulation with Missing Values

Minimize $F(\mathcal{X}, \mathcal{M}) = \sum_{\mathbf{i} \in \Omega \subseteq \mathcal{I}} f(x_{\mathbf{i}}, m_{\mathbf{i}})$ subject to $\mathcal{M} = [\![\mathbf{A}_1, \dots, \mathbf{A}_d]\!]$

Theorem: The partial derivative of F w.r.t. $\mathbf{A}_k$ is given by

$$\frac{\partial F}{\partial \mathbf{A}_k} = \mathbf{G}_{(k)} \left(\mathbf{A}_d \odot \cdots \odot \mathbf{A}_{k+1} \odot \mathbf{A}_{k-1} \odot \cdots \odot \mathbf{A}_1 \right)$$

where $\mathbf{G}_{(k)}$ is the mode-k unfolding of a tensor defined by elementwise by

$$g_{\mathbf{i}} = \begin{cases} \frac{\partial f}{\partial m}(x_{\mathbf{i}}, m_{\mathbf{i}}) & \text{if } \mathbf{i} \in \Omega \\ 0 & \text{if } \mathbf{i} \notin \Omega \end{cases}$$

Bernoulli Tensor Factorization

Original Equations

$$f(x,m) = \log(m+1) - x \log m$$

$$\frac{\partial f}{\partial m}(x,m) = 1/(m+1) - x/m$$

Adjustments to Prevent Numerical Issues

$$f(x,m) = \log(m+1) - x \log(m+\xi)$$

$$\frac{\partial f}{\partial m}(x,m) = 1/(m+1) - x/(m+\xi)$$

$$\xi = 10^{-7}$$

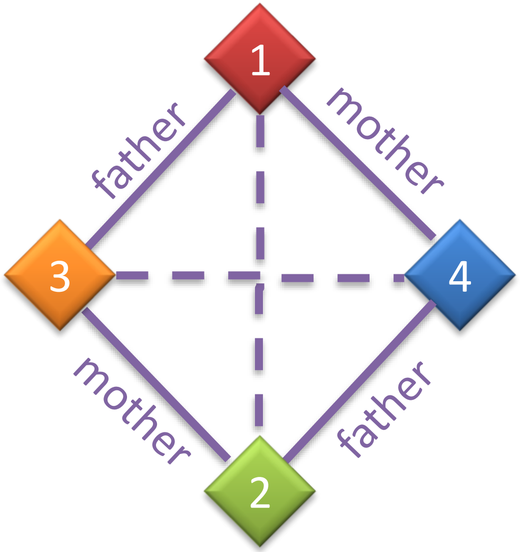
$$\min_{\mathbf{A}^{(1)},\ldots,\mathbf{A}^{(d)}} F(\mathcal{X},\mathcal{M}) = \sum_{\mathbf{i}\in\mathcal{I}} f(x_{\mathbf{i}}, m_{\mathbf{i}})$$

s.t. $\mathcal{M} = [\![\mathbf{A}_1,\ldots,\mathbf{A}_d]\!]$ and $\mathbf{A}_k \geq 0 \ \forall k \in [d]$

$$\frac{\partial F}{\partial \mathbf{A}_k} = \mathbf{G}_{(k)} \left(\mathbf{A}_d \odot \cdots \odot \mathbf{A}_{k+1} \odot \mathbf{A}_{k-1} \odot \cdots \odot \mathbf{A}_1\right) \qquad g_{\mathbf{i}} = \frac{\partial f}{\partial m}(x_{\mathbf{i}}, m_{\mathbf{i}})$$

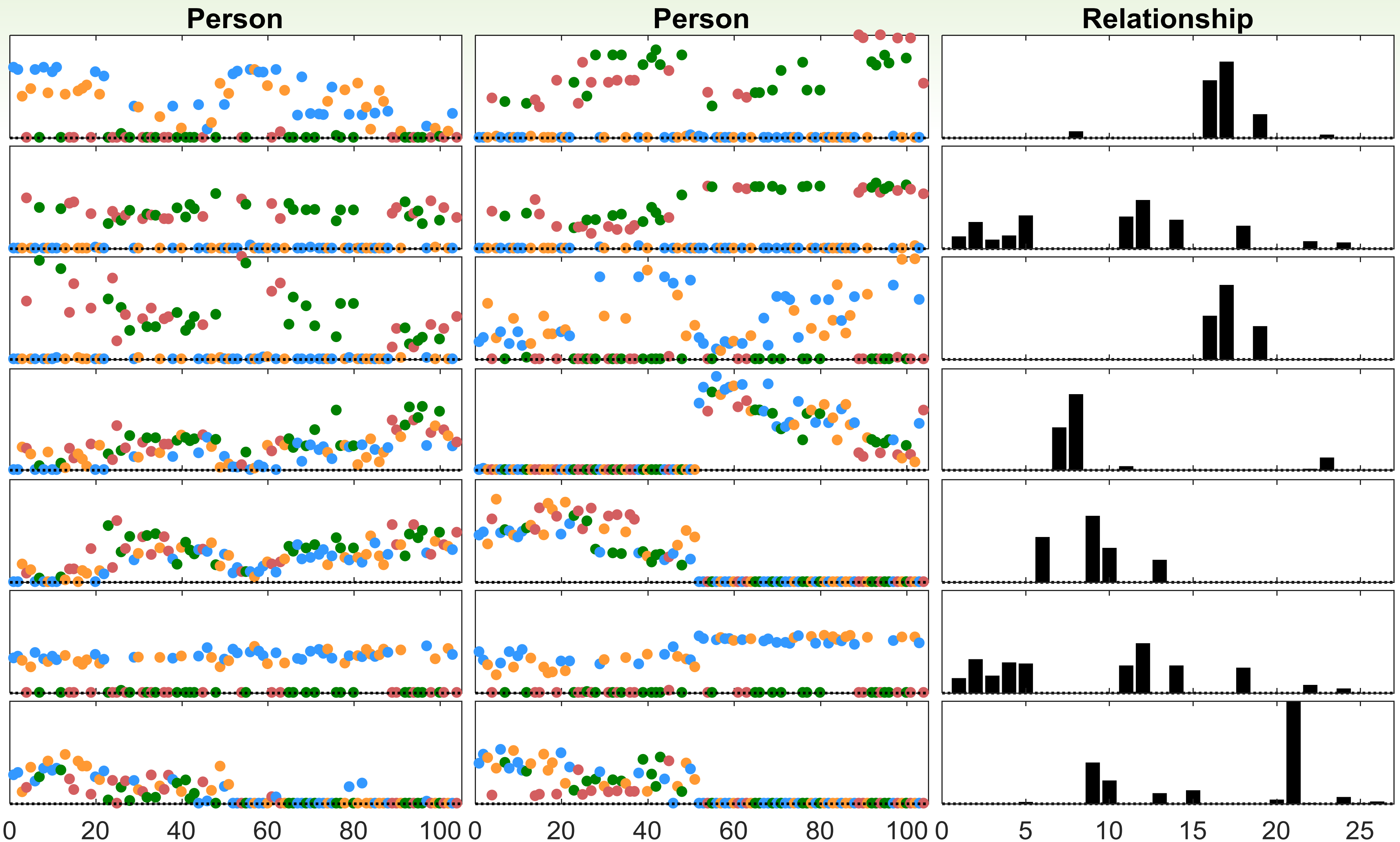
Preliminary Analysis: Kinship Data

- Australian tribe
- 104 persons
- 4 sections
- 26 kinship terms



- Kinship Terms
  - Complex relationships having to do with sections, gender, and age
  - Example: *Adiadya* – Younger person in same section
- Citations
  - Denham, PhD Thesis, 1973
  - Kemp, Tenenbaum, Griffiths, Yamada, Ueda, *Learning Systems of Concepts with an Infinite Relational Model*, AAAI-06, 2006
  - Nickel, Tresp and Kriegel, *A three-way model for collective learning on multi-relational data,* ICML-11, 2011

7-Component Results



Scaling Bernoulli Tensor Factorization

- Expectation of dense tensors
  - Even if data is sparse, gradient 'G' tensor is dense
  - If data is sparse, may be dealing with zero inflation
- No clear way to maintain sparsity
  - Is possible in Gaussian & Poisson cases with special handling
- Instead, can use variant of stochastic gradient descent
  - Sparsify function tensor
  - Sparsify gradient tensor

Bernoulli Equations

$$f(x,m) = \log(m+1) - x \log m$$

$$\frac{\partial f}{\partial m}(x,m) = 1/(m+1) - x/m$$

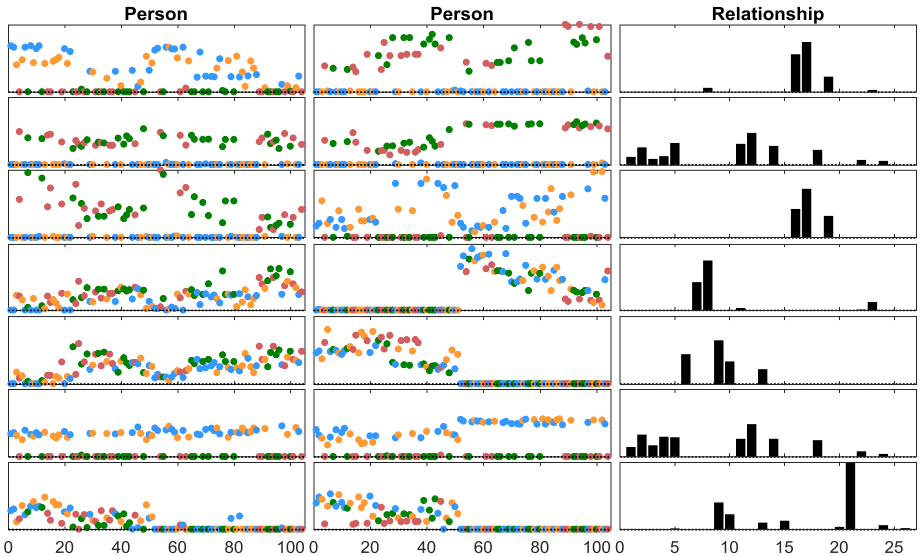
$$F(\mathcal{X},\mathcal{M}) = \sum_{\mathbf{i} \in \Omega \subseteq \mathcal{I}} f(x_{\mathbf{i}}, m_{\mathbf{i}})$$

$$g_{\mathbf{i}} = \begin{cases} \frac{\partial f}{\partial m}(x_{\mathbf{i}}, m_{\mathbf{i}}) & \text{if } \mathbf{i} \in \Omega \\ 0 & \text{if } \mathbf{i} \notin \Omega \end{cases}$$

Bernoulli Tensor Factorization

- Consider data types in formulation of loss function
- General formulation of tensor factorization
  - Accommodates any loss function
  - Accounts for missing data
  - Can be adapted for randomized optimization
- Applied to Bernoulli tensor factorization
- Preliminary results on “kinship” data

$$F(\mathcal{X},\mathcal{M}) = \sum_{\mathbf{i}\in\mathcal{I}} \log(m_{\mathbf{i}}+1) - x_{\mathbf{i}} \log m_{\mathbf{i}}$$



More Info: Tammy Kolda
tgkolda@sandia.gov