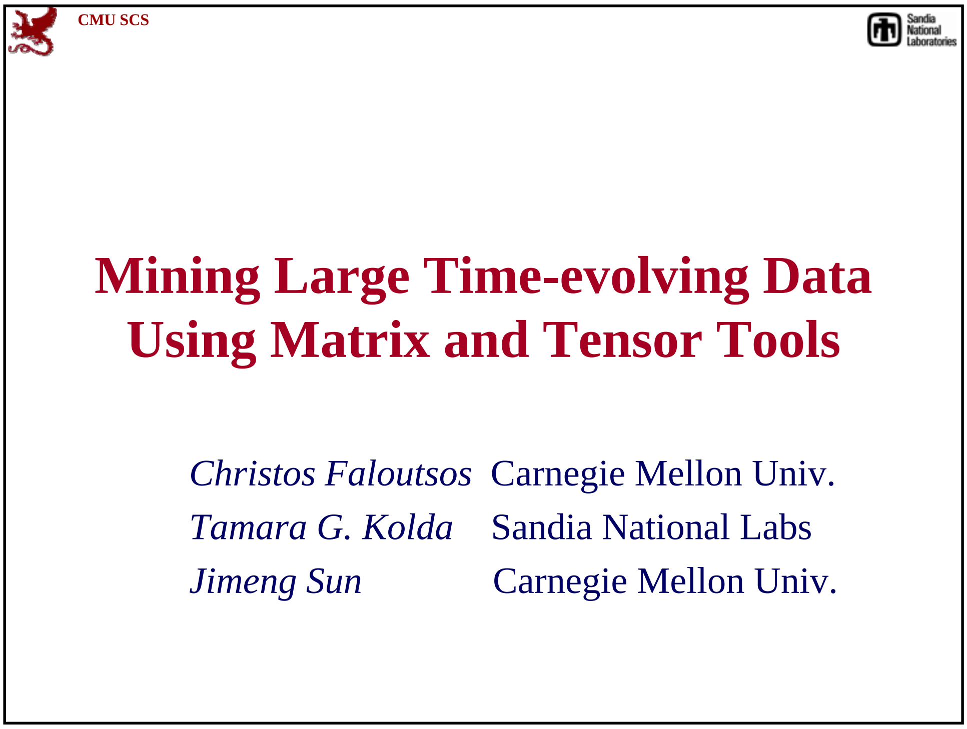
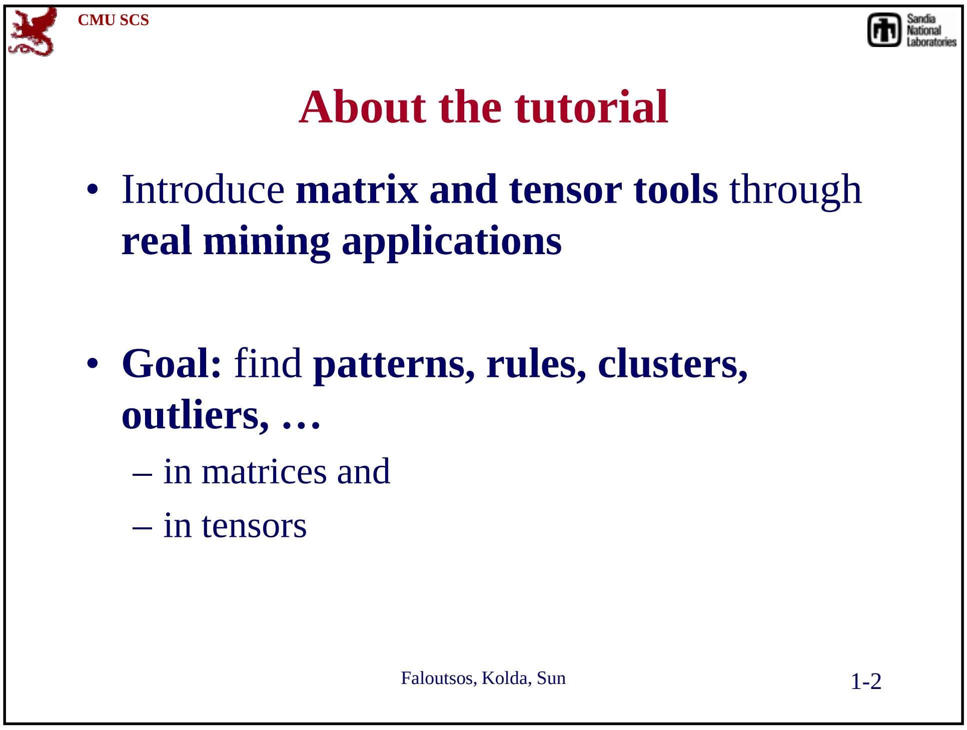


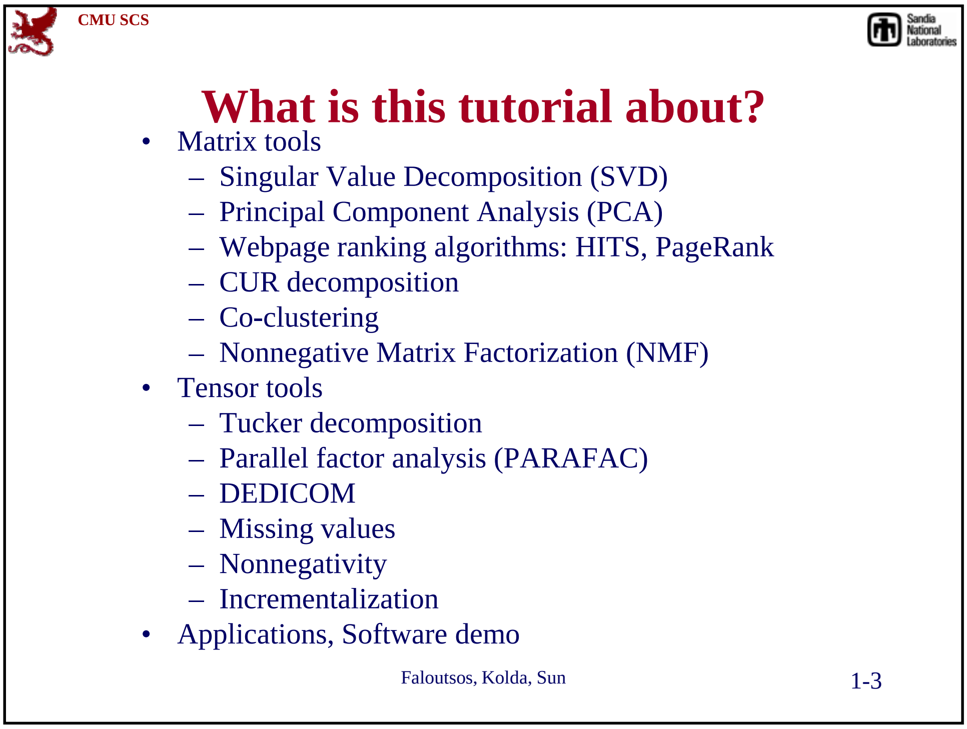
Mining Large Time-evolving Data Using Matrix and Tensor Tools

Christos Faloutsos Carnegie Mellon Univ.
Tamara G. Kolda Sandia National Labs
Jimeng Sun Carnegie Mellon Univ.



About the tutorial

- Introduce **matrix and tensor tools** through **real mining applications**
- **Goal:** find **patterns, rules, clusters, outliers, …**
  - in matrices and
  - in tensors

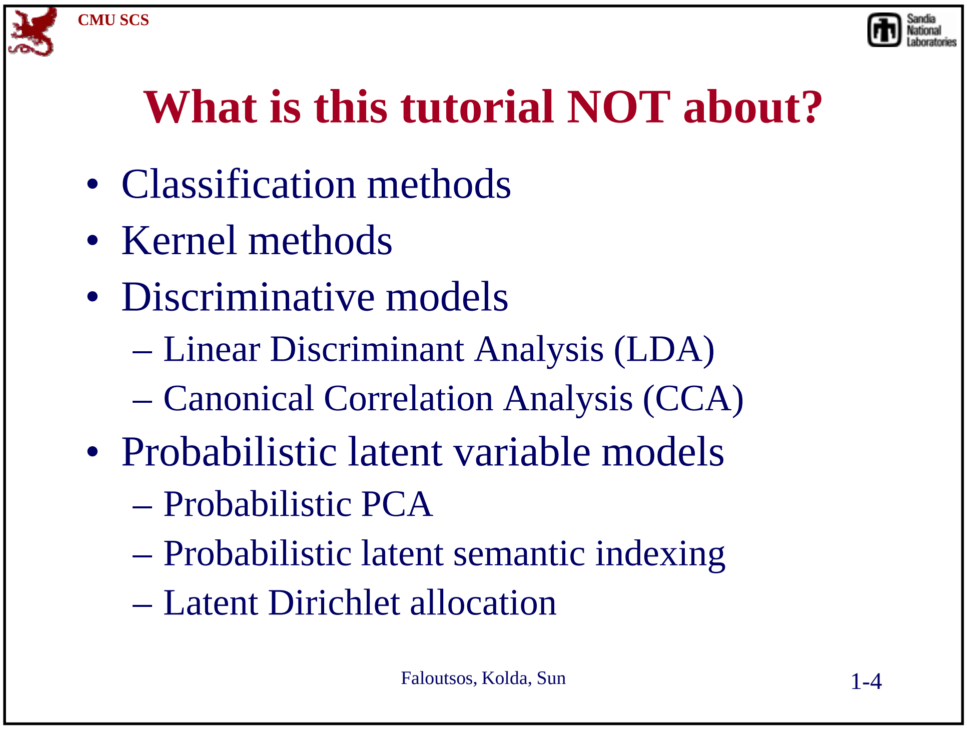


CMU SCS

What is this tutorial about?

- Matrix tools
  - Singular Value Decomposition (SVD)
  - Principal Component Analysis (PCA)
  - Webpage ranking algorithms: HITS, PageRank
  - CUR decomposition
  - Co-clustering
  - Nonnegative Matrix Factorization (NMF)
- Tensor tools
  - Tucker decomposition
  - Parallel factor analysis (PARAFAC)
  - DEDICOM
  - Missing values
  - Nonnegativity
  - Incrementalization
- Applications, Software demo

Faloutsos, Kolda, Sun 1-3

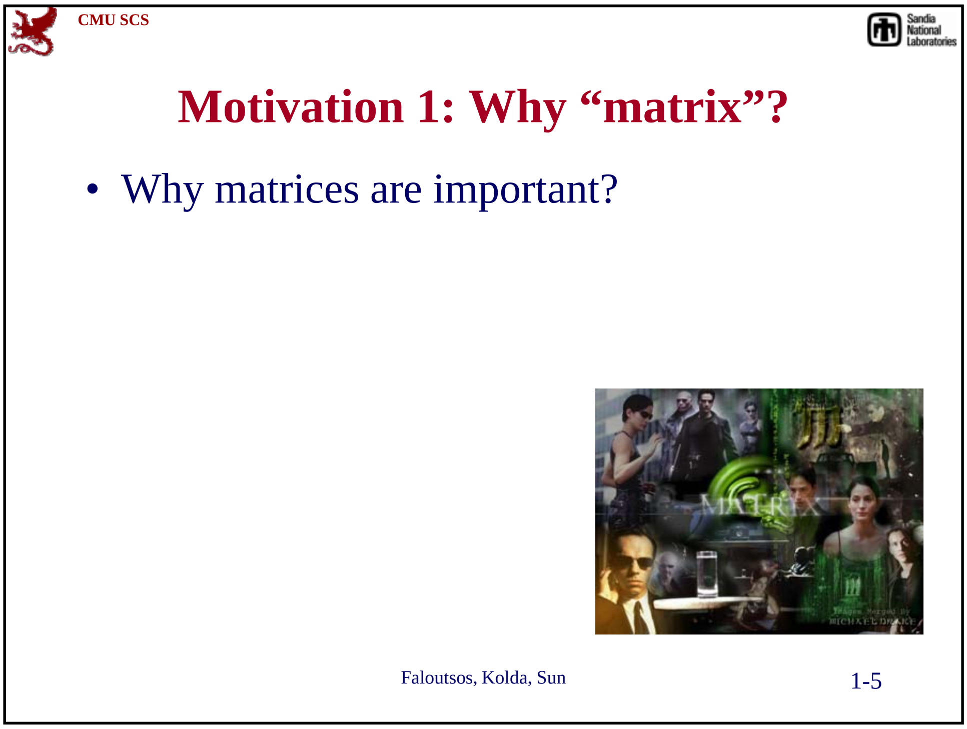


CMU SCS

What is this tutorial NOT about?

- Classification methods
- Kernel methods
- Discriminative models
  - Linear Discriminant Analysis (LDA)
  - Canonical Correlation Analysis (CCA)
- Probabilistic latent variable models
  - Probabilistic PCA
  - Probabilistic latent semantic indexing
  - Latent Dirichlet allocation

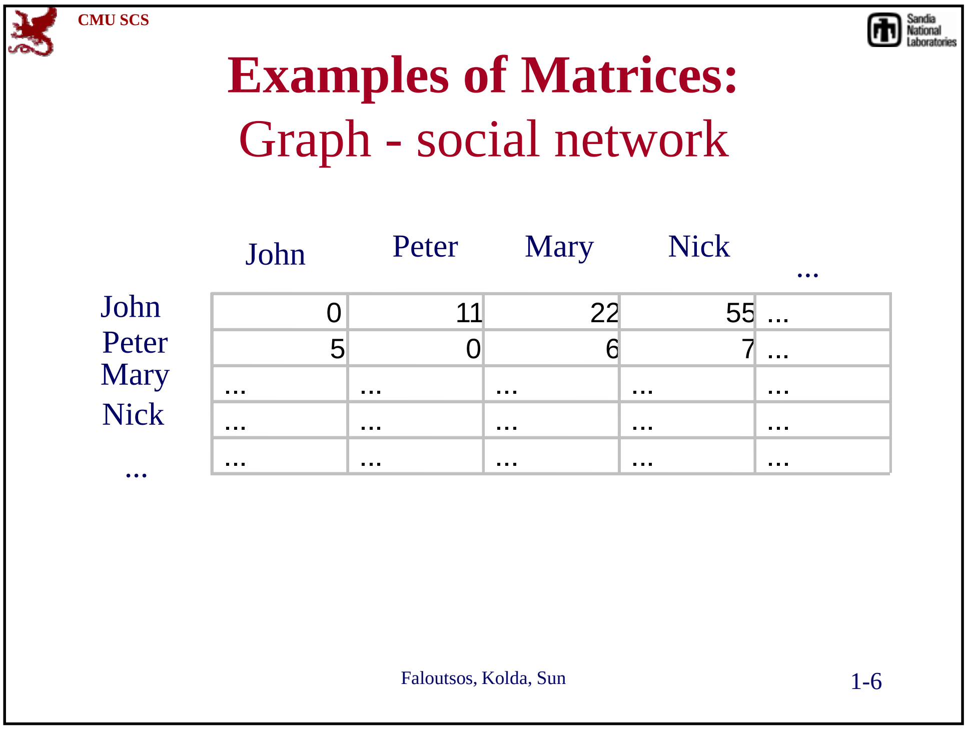
Faloutsos, Kolda, Sun 1-4



CMU SCS

Motivation 1: Why “matrix”?

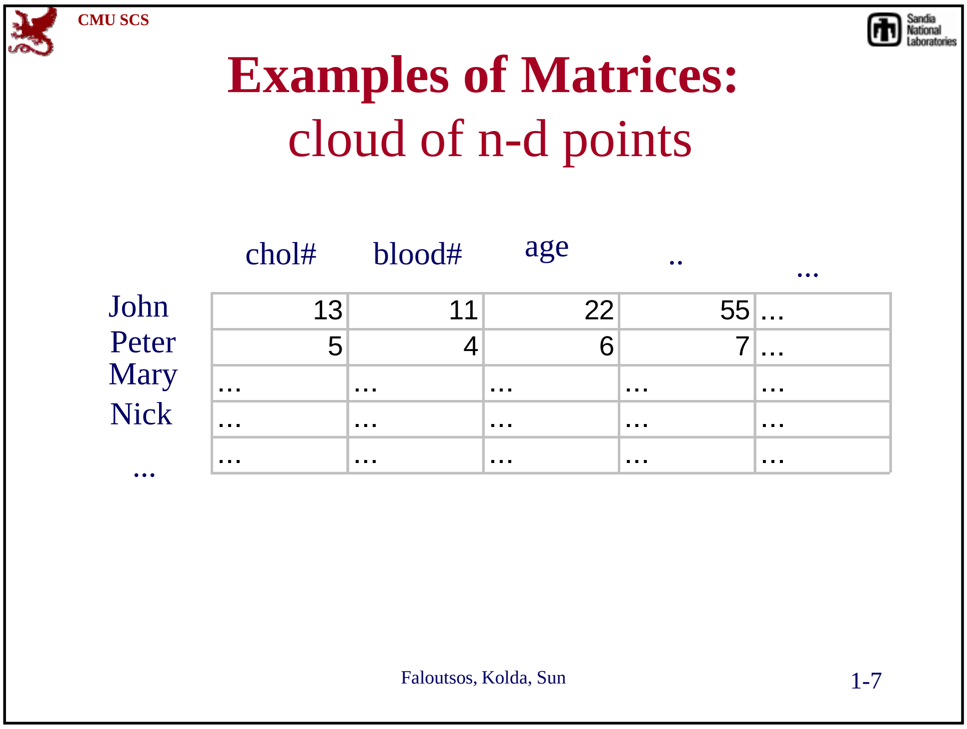
- Why matrices are important?



CMU SCS

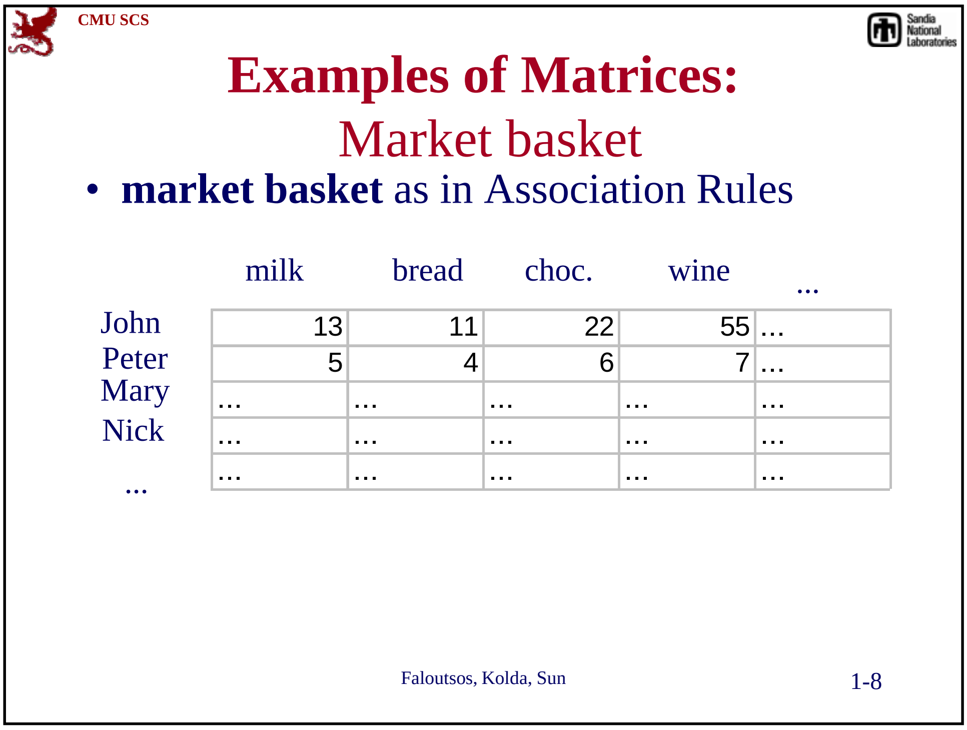
Examples of Matrices: Graph - social network

	John	Peter	Mary	Nick	...
John	0	11	22	55	...
Peter	5	0	6	7	...
Mary	...	...	...	...	...
Nick	...	...	...	...	...
...	...	...	...	...	...



Examples of Matrices:
cloud of n-d points

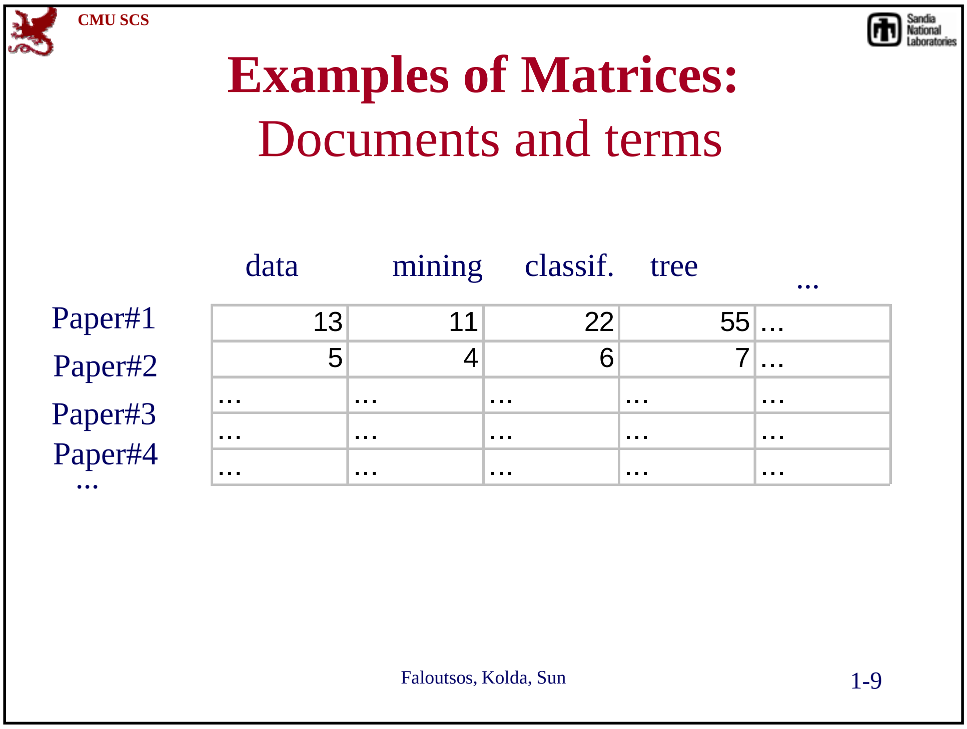
	chol#	blood#	age	..	...
John	13	11	22	55	...
Peter	5	4	6	7	...
Mary	...	...	...	...	...
Nick	...	...	...	...	...
...	...	...	...	...	...



Examples of Matrices:
Market basket

- **market basket** as in Association Rules

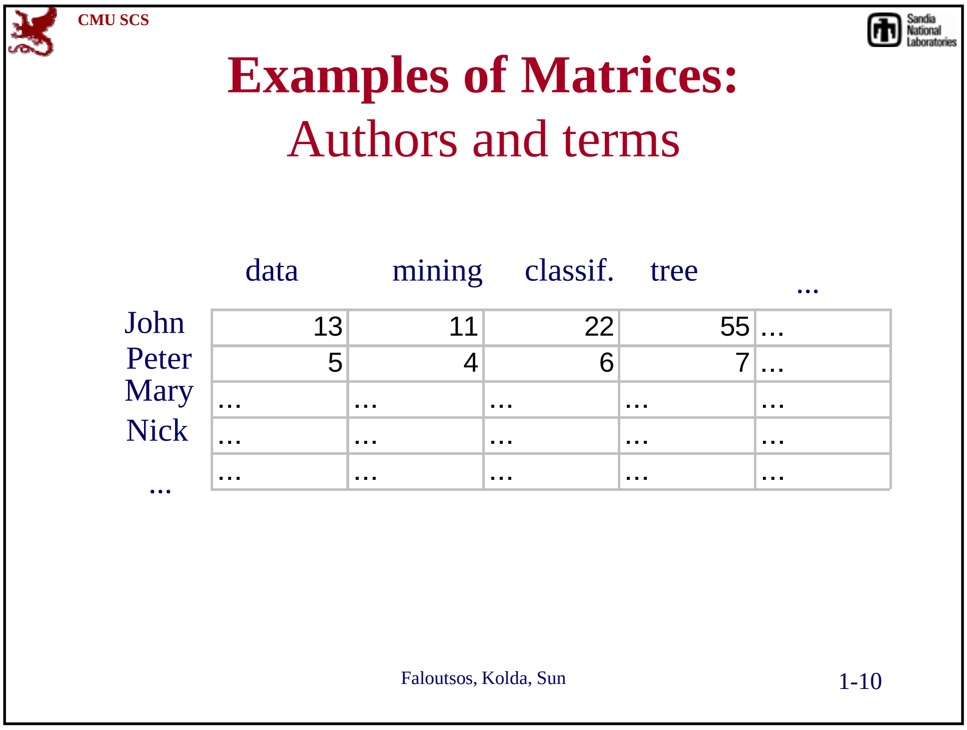
	milk	bread	choc.	wine	...
John	13	11	22	55	...
Peter	5	4	6	7	...
Mary	...	...	...	...	...
Nick	...	...	...	...	...
...	...	...	...	...	...



CMU SCS

Examples of Matrices:
Documents and terms

	data	mining	classif.	tree	...
Paper#1	13	11	22	55	...
Paper#2	5	4	6	7	...
Paper#3	...	...	...	...	...
Paper#4	...	...	...	...	...
...	...	...	...	...	...



CMU SCS

Examples of Matrices:
Authors and terms

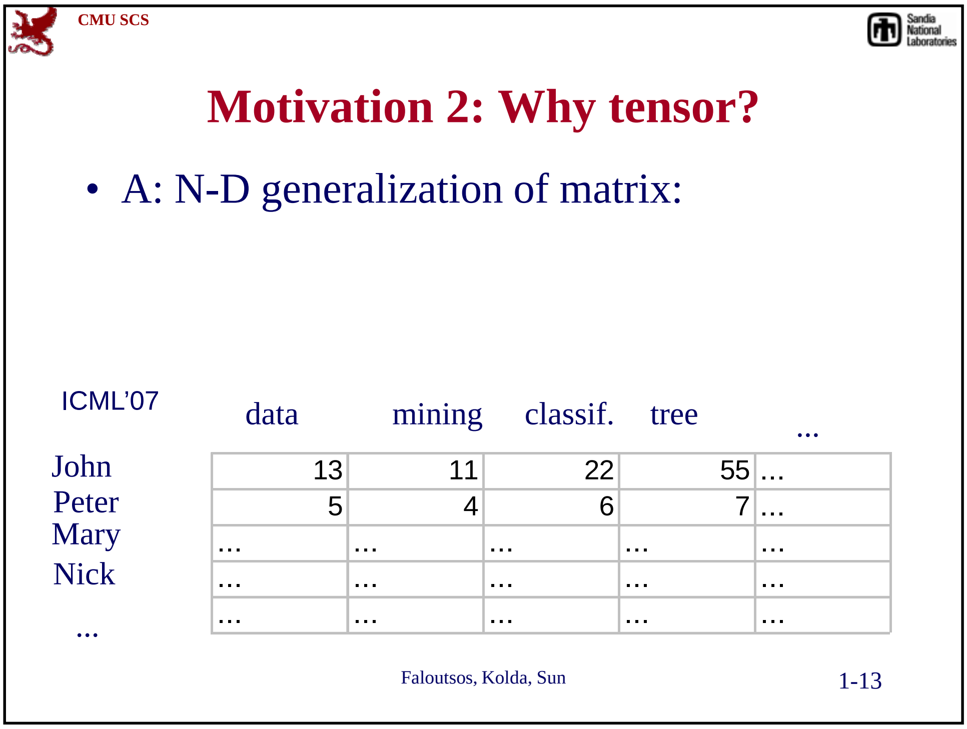
	data	mining	classif.	tree	...
John	13	11	22	55	...
Peter	5	4	6	7	...
Mary	...	...	...	...	...
Nick	...	...	...	...	...
...	...	...	...	...	...

Examples of Matrices:
sensor-ids and time-ticks

	temp1	temp2	humid.	pressure	...
t1	13	11	22	55	...
t2	5	4	6	7	...
t3	...	...	...	...	...
t4	...	...	...	...	...
...	...	...	...	...	...

Motivation 2: Why tensor?

- Q: what is a tensor?



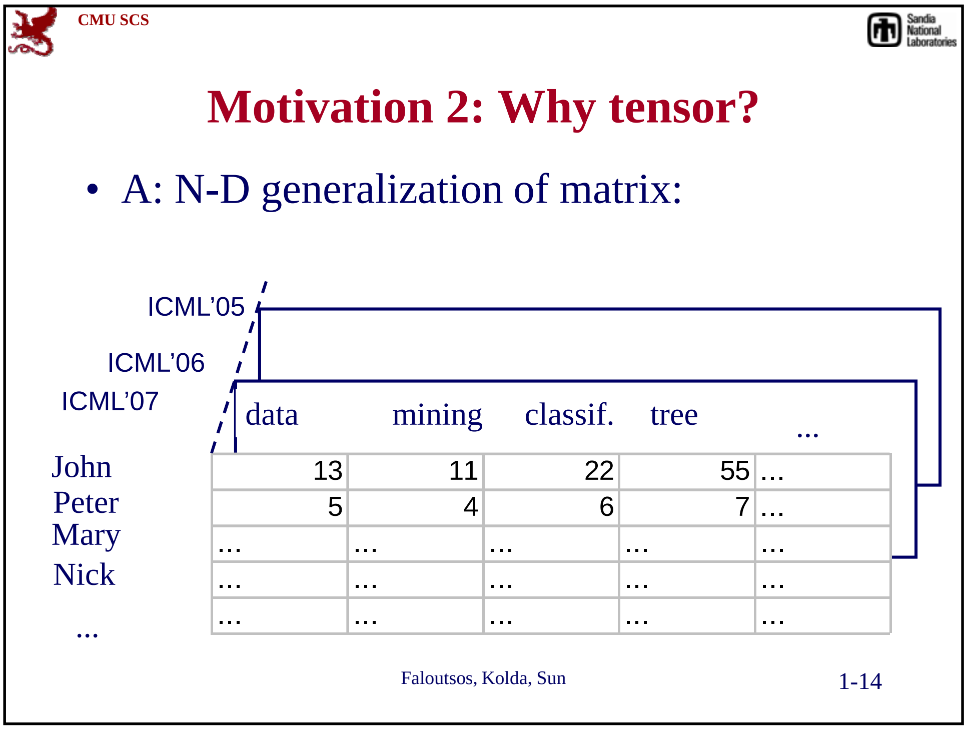
CMU SCS

Sandia National Laboratories

Motivation 2: Why tensor?

- A: N-D generalization of matrix:

ICML'07	data	mining	classif.	tree	...
John	13	11	22	55	...
Peter	5	4	6	7	...
Mary	...	...	...	...	...
Nick	...	...	...	...	...
...	...	...	...	...	...



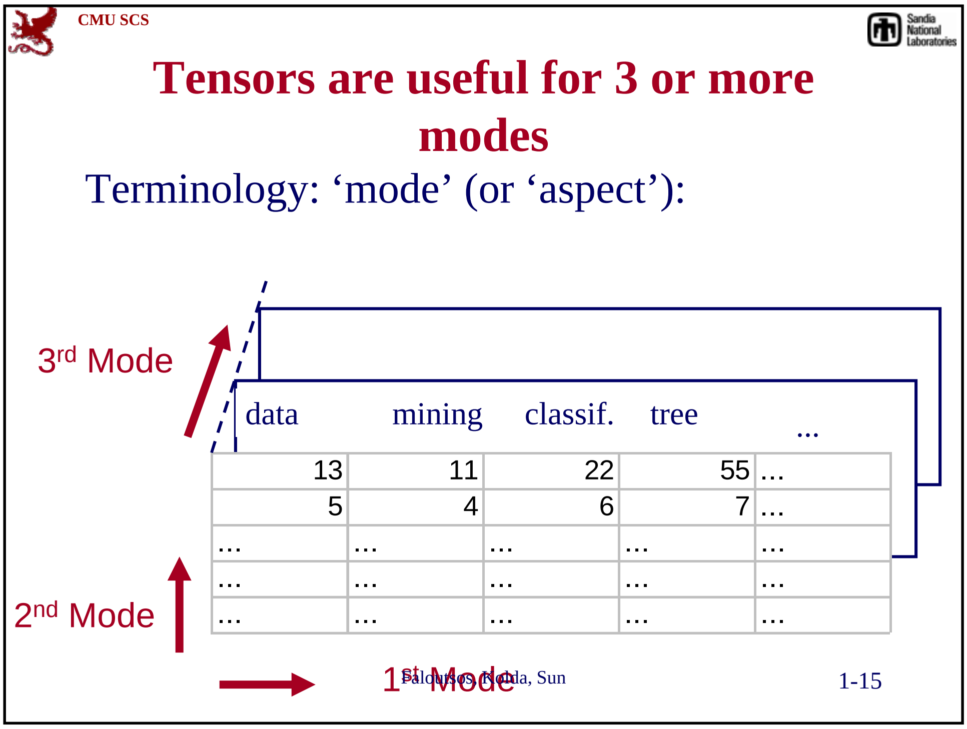
CMU SCS

Sandia National Laboratories

Motivation 2: Why tensor?

- A: N-D generalization of matrix:

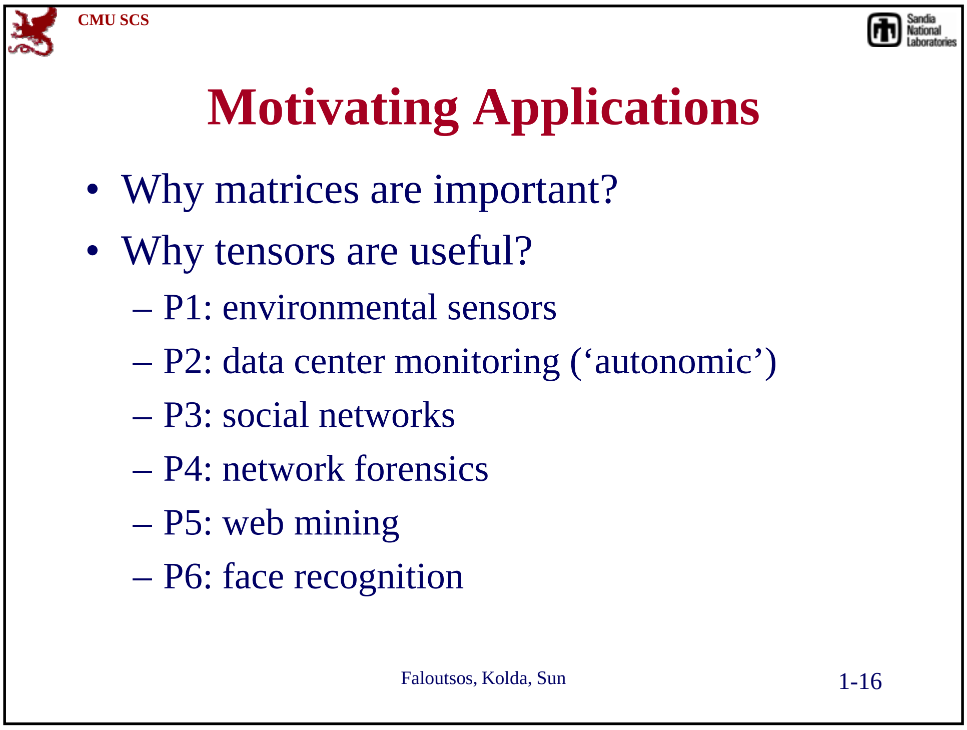
ICML'07	data	mining	classif.	tree	...
John	13	11	22	55	...
Peter	5	4	6	7	...
Mary	...	...	...	...	...
Nick	...	...	...	...	...
...	...	...	...	...	...



Tensors are useful for 3 or more modes

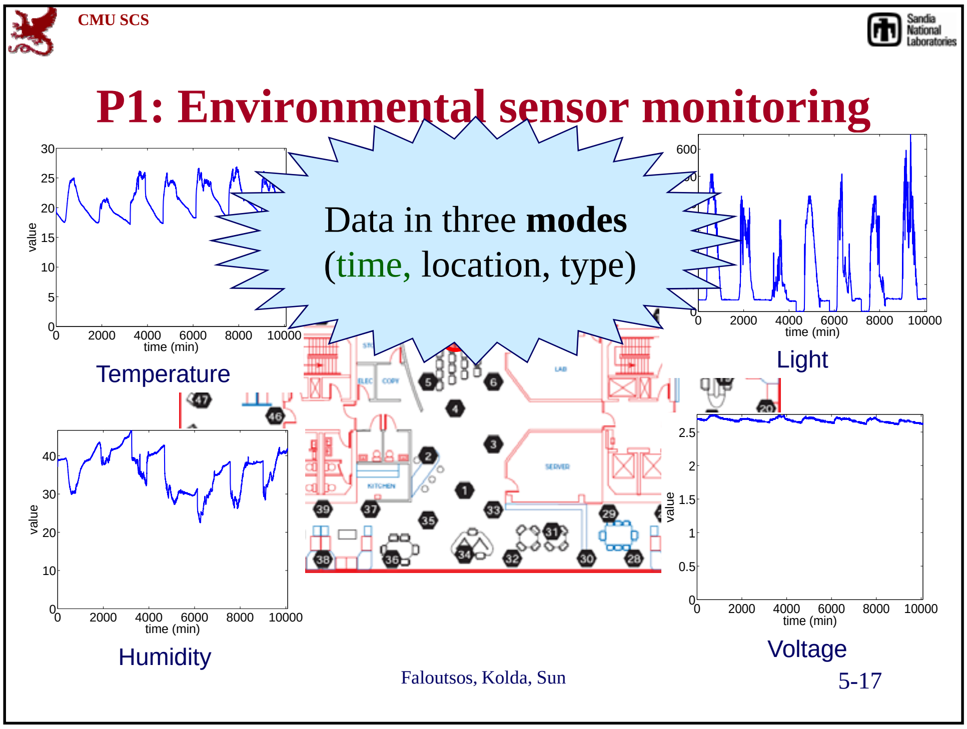
Terminology: ‘mode’ (or ‘aspect’):

data	mining	classif.	tree	...
13	11	22	55	...
5	4	6	7	...
...	...	...	...	...
...	...	...	...	...
...	...	...	...	...



Motivating Applications

- Why matrices are important?
- Why tensors are useful?
  - P1: environmental sensors
  - P2: data center monitoring (‘autonomic’)
  - P3: social networks
  - P4: network forensics
  - P5: web mining
  - P6: face recognition



CMU SCS

P1: Environmental sensor monitoring

Data in three **modes**
(time, location, type)

Temperature

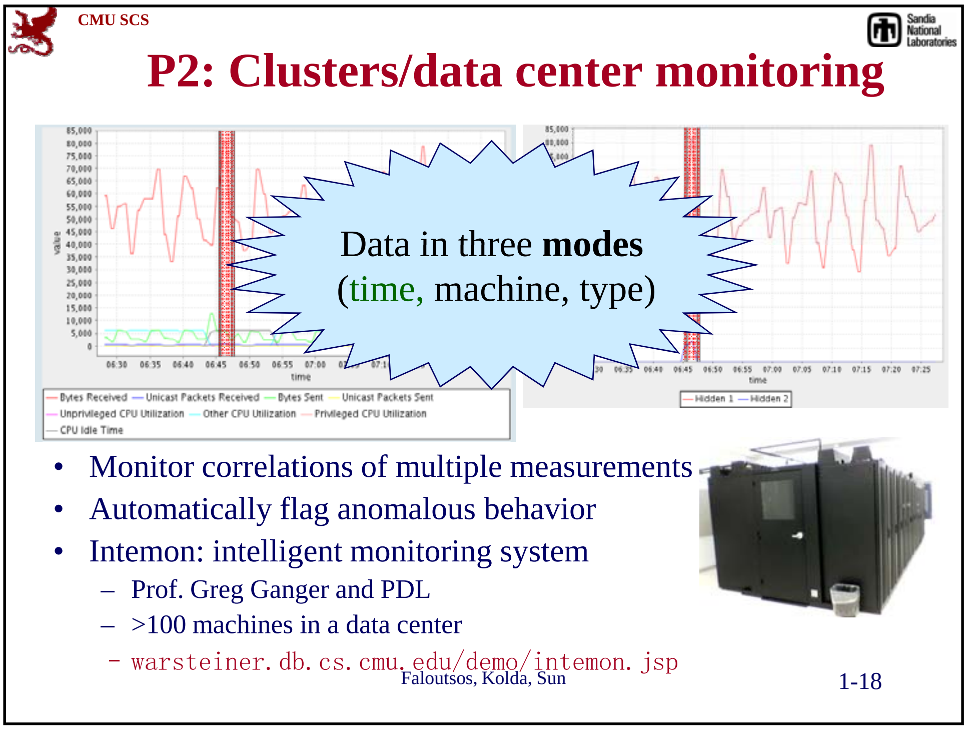
Light

Humidity

Voltage

Faloutsos, Kolda, Sun

5-17



CMU SCS

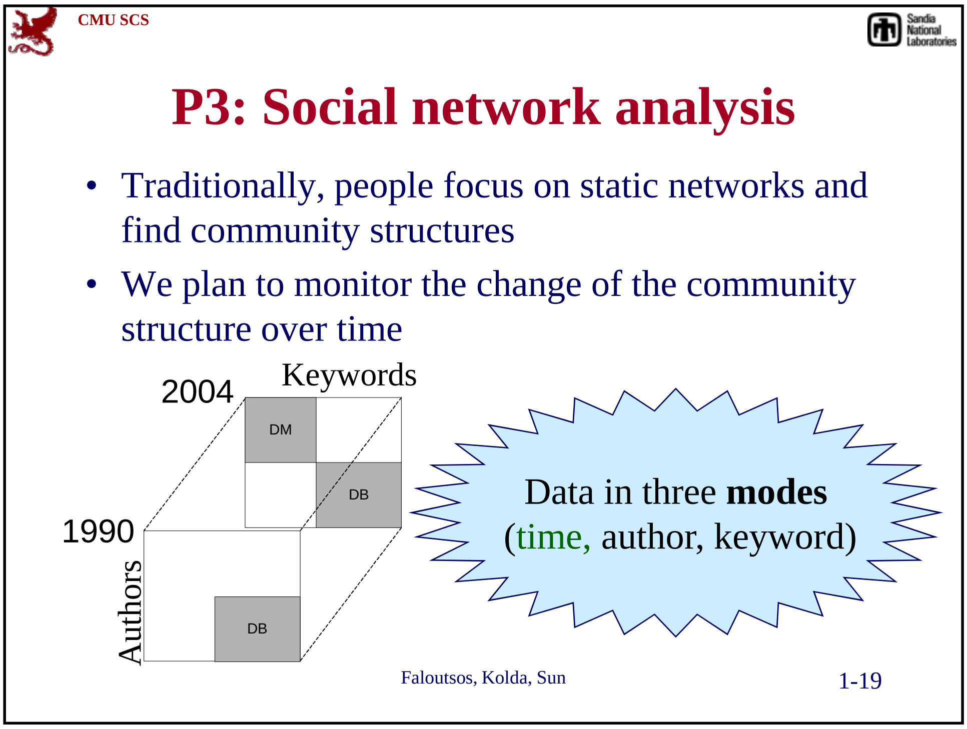
P2: Clusters/data center monitoring

Data in three **modes**
(time, machine, type)

- Monitor correlations of multiple measurements
- Automatically flag anomalous behavior
- Intemon: intelligent monitoring system
  - Prof. Greg Ganger and PDL
  - >100 machines in a data center
  - warsteiner.db.cs.cmu.edu/demo/intemon.jsp

Faloutsos, Kolda, Sun

1-18



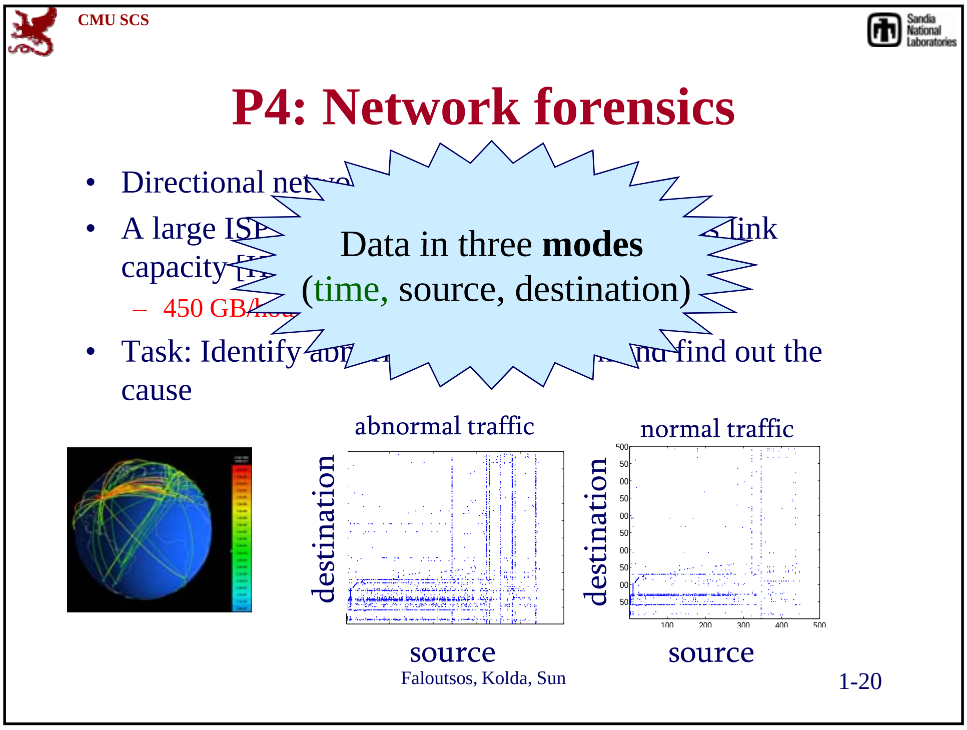
CMU SCS

P3: Social network analysis

- Traditionally, people focus on static networks and find community structures
- We plan to monitor the change of the community structure over time

Data in three **modes** (time, author, keyword)

Faloutsos, Kolda, Sun 1-19



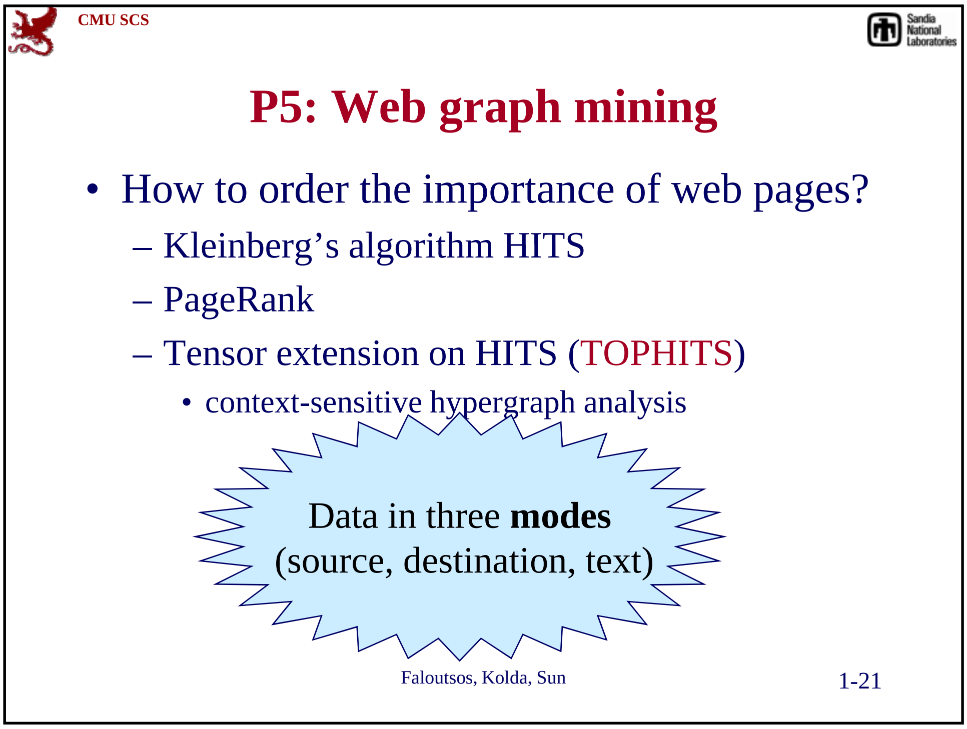
CMU SCS

P4: Network forensics

- Directional net
- A large ISP link capacity [I
  - 450 GB/
- Task: Identify ab nd find out the cause

Data in three **modes** (time, source, destination)

Faloutsos, Kolda, Sun 1-20



CMU SCS

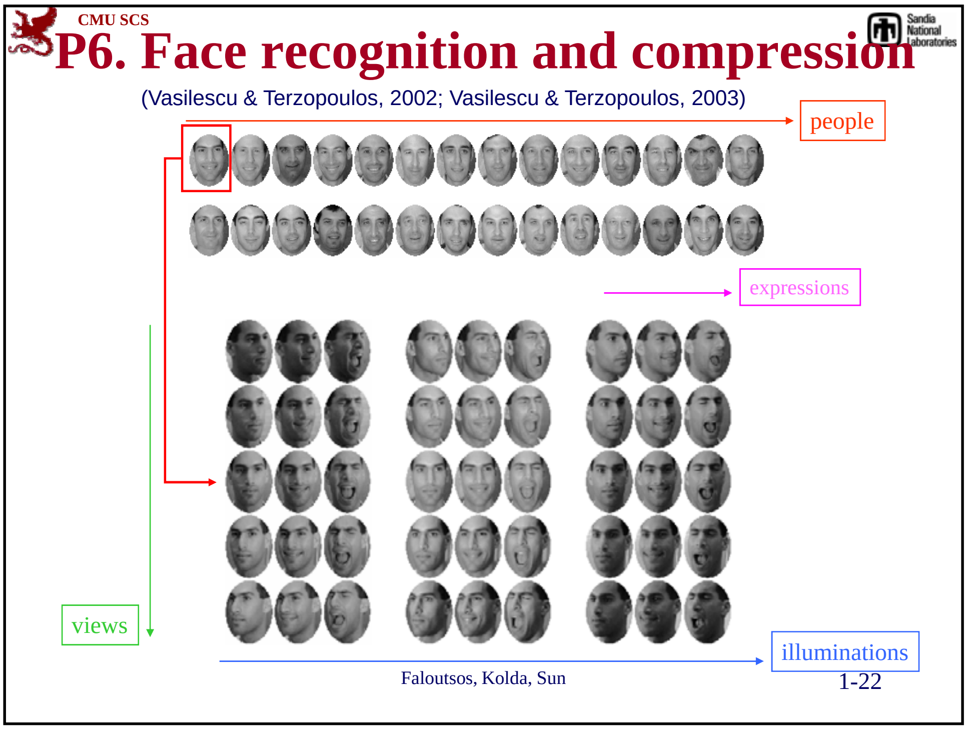
Sandia National Laboratories

P5: Web graph mining

- How to order the importance of web pages?
  - – Kleinberg's algorithm HITS
  - – PageRank
  - – Tensor extension on HITS (TOPHITS)
    - context-sensitive hypergraph analysis

Data in three **modes**
(source, destination, text)

Faloutsos, Kolda, Sun 1-21



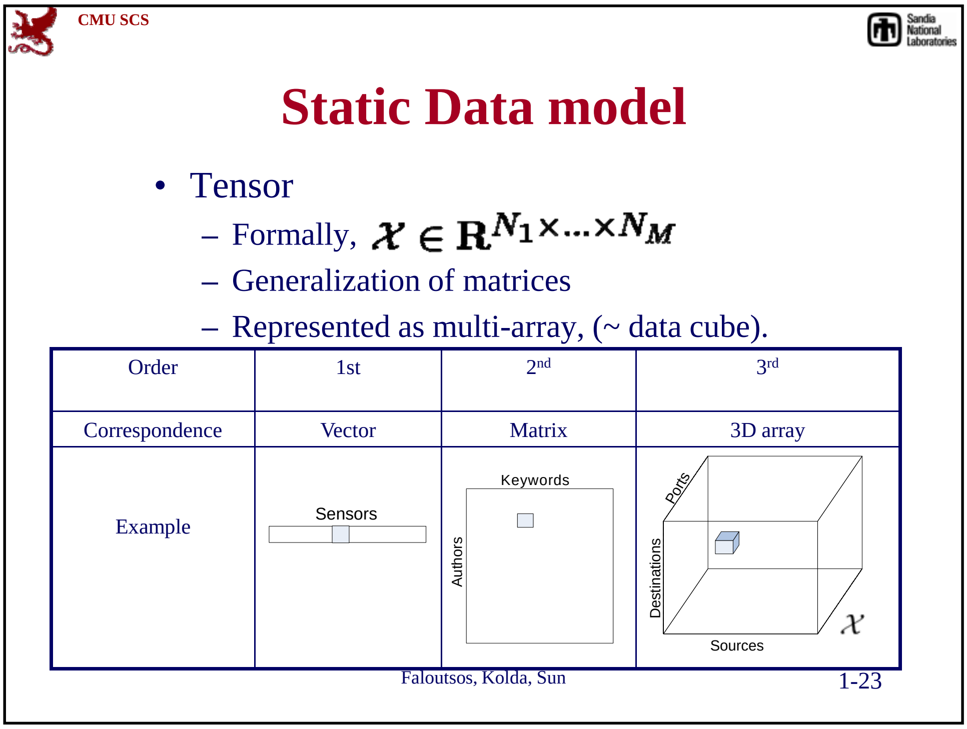
CMU SCS

Sandia National Laboratories

P6. Face recognition and compression

(Vasilescu & Terzopoulos, 2002; Vasilescu & Terzopoulos, 2003)

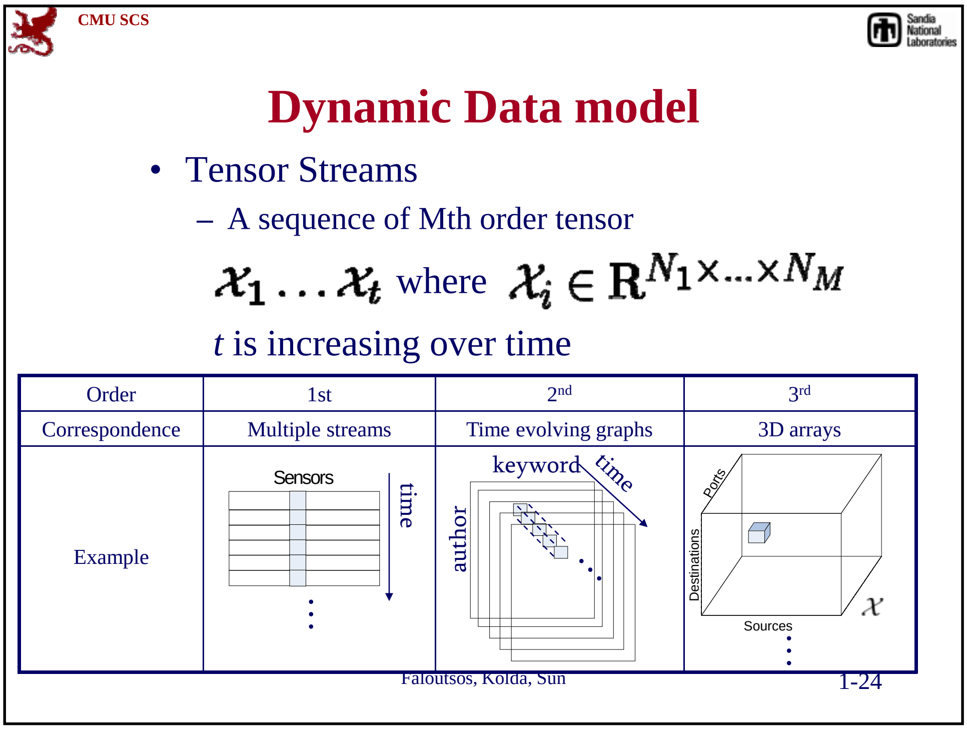
Faloutsos, Kolda, Sun 1-22



Static Data model

- Tensor
  - Formally, $\mathcal{X} \in \mathbf{R}^{N_1 \times \ldots \times N_M}$
  - Generalization of matrices
  - Represented as multi-array, (~ data cube).

Order	1st	2nd	3rd
Correspondence	Vector	Matrix	3D array
Example	Sensors	Keywords Authors	Ports Destinations Sources $\mathcal{X}$



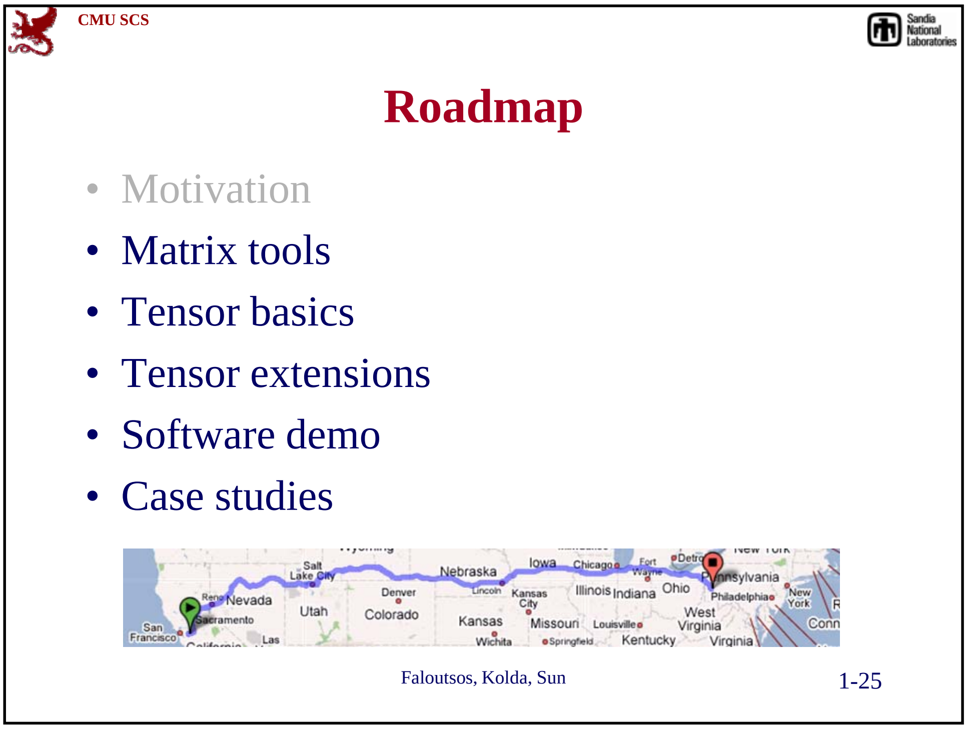
Dynamic Data model

- Tensor Streams
  - A sequence of Mth order tensor

$\mathcal{X}_1 \ldots \mathcal{X}_t$ where $\mathcal{X}_i \in \mathbf{R}^{N_1 \times \ldots \times N_M}$

t is increasing over time

Order	1st	2nd	3rd
Correspondence	Multiple streams	Time evolving graphs	3D arrays
Example	Sensors time	keyword time author	Ports Destinations Sources $\mathcal{X}$

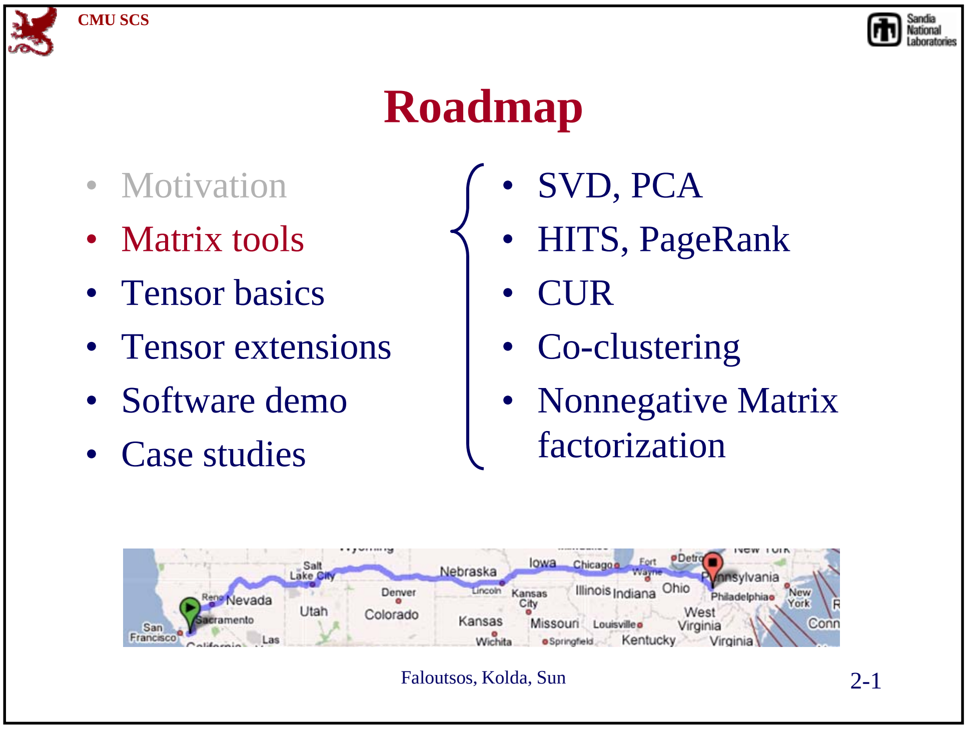


CMU SCS

Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

San Francisco
Sacramento
Reno
Nevada
Salt Lake City
Utah
Denver
Colorado
Nebraska
Lincoln
Kansas City
Kansas
Wichita
Iowa
Chicago
Illinois
Missouri
Springfield
Indiana
Fort Wayne
Louisville
Kentucky
Ohio
Detroit
Pennsylvania
Philadelphia
West Virginia
Virginia
New York
Conn



CMU SCS

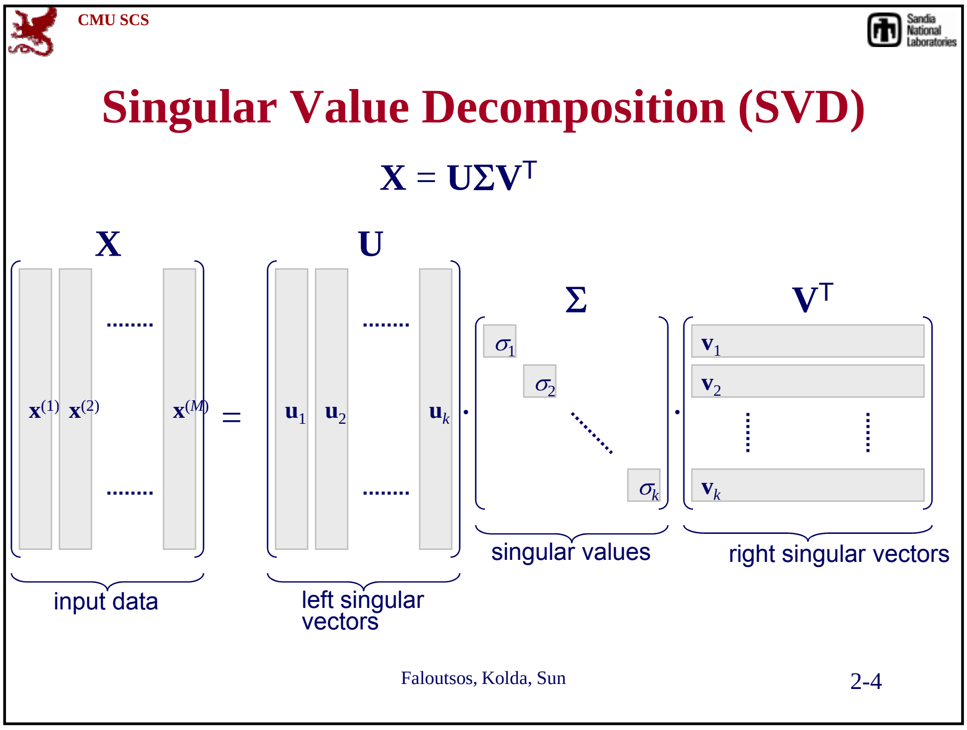
Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

- SVD, PCA
- HITS, PageRank
- CUR
- Co-clustering
- Nonnegative Matrix factorization

Salt Lake City
Nebraska
Iowa
Chicago
Fort Wayne
Detroit
Pennsylvania
Reno
Nevada
Denver
Lincoln
Kansas City
Illinois
Indiana
Ohio
Philadelphia
New York
Sacramento
Utah
Colorado
Kansas
Missouri
Louisville
West Virginia
San Francisco
Wichita
Springfield
Kentucky
Virginia
Conn

Faloutsos, Kolda, Sun 2-1



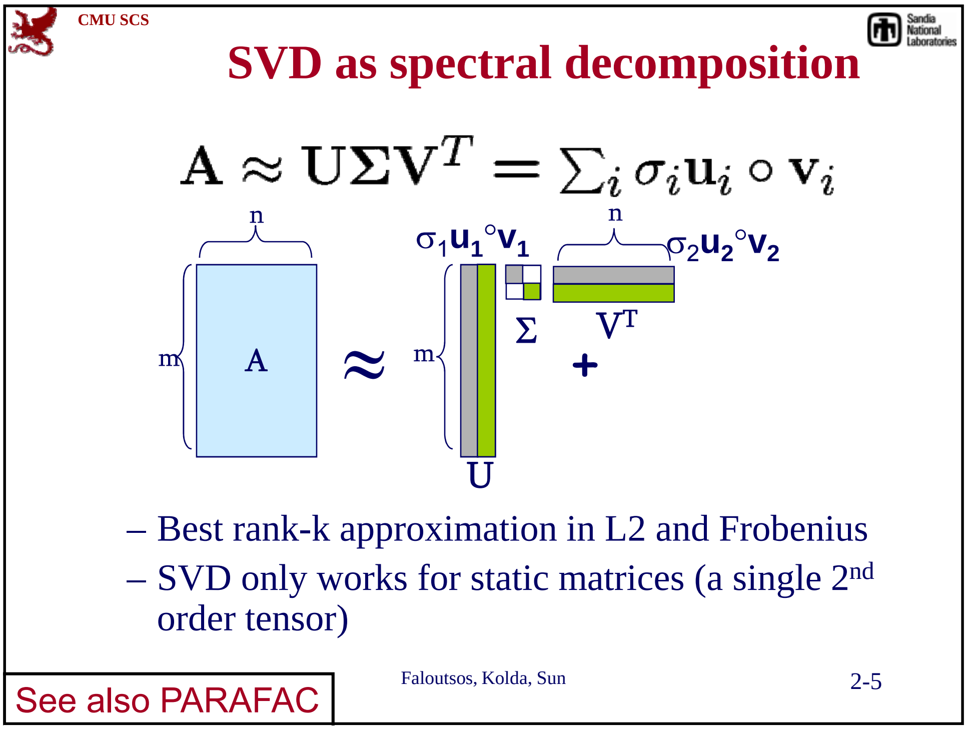
CMU SCS

Singular Value Decomposition (SVD)

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

$$\underbrace{\left[\mathbf{x}^{(1)}\ \mathbf{x}^{(2)}\ \cdots\ \mathbf{x}^{(M)}\right]}_{\text{input data}} = \underbrace{\left[\mathbf{u}_1\ \mathbf{u}_2\ \cdots\ \mathbf{u}_k\right]}_{\text{left singular vectors}} \cdot \underbrace{\begin{bmatrix}\sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_k\end{bmatrix}}_{\text{singular values}} \cdot \underbrace{\begin{bmatrix}\mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_k\end{bmatrix}}_{\text{right singular vectors}}$$

Faloutsos, Kolda, Sun 2-4



CMU SCS

SVD as spectral decomposition

$$\mathbf{A} \approx \mathbf{U}\Sigma\mathbf{V}^T = \sum_i \sigma_i \mathbf{u}_i \circ \mathbf{v}_i$$

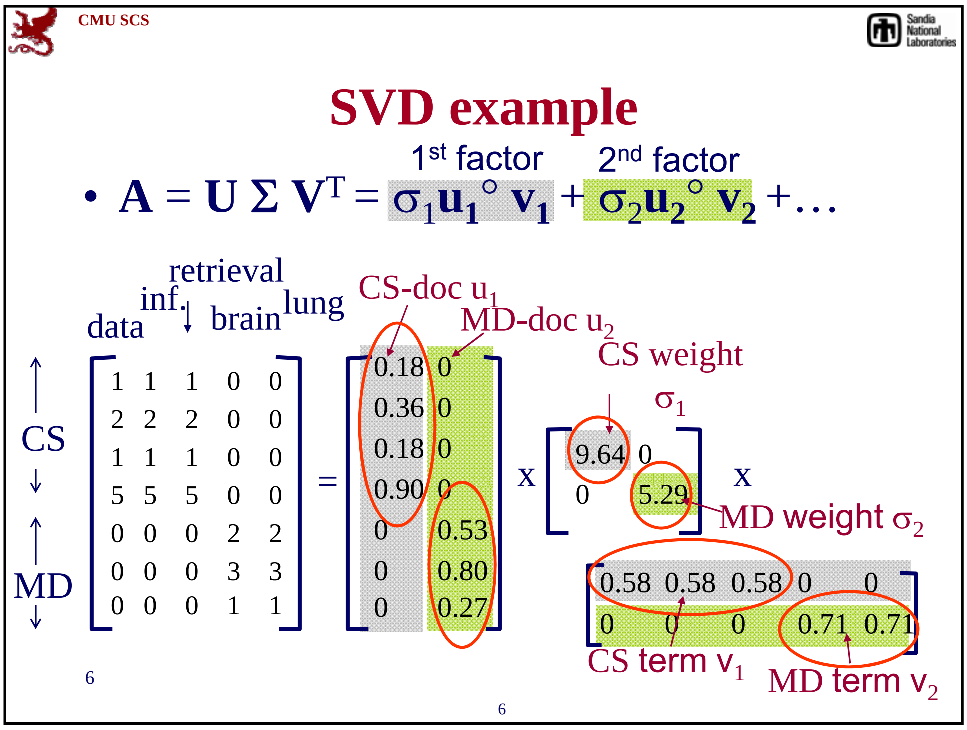
n
m
A
≈
$\sigma_1 \mathbf{u}_1 \circ \mathbf{v}_1$
n
$\sigma_2 \mathbf{u}_2 \circ \mathbf{v}_2$
m
Σ
V^T
+
U

– Best rank-k approximation in L2 and Frobenius

– SVD only works for static matrices (a single 2nd order tensor)

Faloutsos, Kolda, Sun 2-5

See also PARAFAC



CMU SCS

SVD example

1st factor 2nd factor

- $\mathbf{A} = \mathbf{U}\,\Sigma\,\mathbf{V}^T = \sigma_1 \mathbf{u}_1 \circ \mathbf{v}_1 + \sigma_2 \mathbf{u}_2 \circ \mathbf{v}_2 + \ldots$

$$\begin{array}{c|ccccc} & \text{data} & \text{inf. retrieval} & \text{brain} & \text{lung} & \\ \text{CS} & 1 & 1 & 1 & 0 & 0 \\ & 2 & 2 & 2 & 0 & 0 \\ & 1 & 1 & 1 & 0 & 0 \\ & 5 & 5 & 5 & 0 & 0 \\ \text{MD} & 0 & 0 & 0 & 2 & 2 \\ & 0 & 0 & 0 & 3 & 3 \\ & 0 & 0 & 0 & 1 & 1 \end{array} = \begin{bmatrix} 0.18 & 0 \\ 0.36 & 0 \\ 0.18 & 0 \\ 0.90 & 0 \\ 0 & 0.53 \\ 0 & 0.80 \\ 0 & 0.27 \end{bmatrix} \times \begin{bmatrix} 9.64 & 0 \\ 0 & 5.29 \end{bmatrix} \times \begin{bmatrix} 0.58 & 0.58 & 0.58 & 0 & 0 \\ 0 & 0 & 0 & 0.71 & 0.71 \end{bmatrix}$$

CS-doc u_1

MD-doc u_2

CS weight σ_1

MD weight σ_2

CS term v_1

MD term v_2

6

6

SVD properties

- **V** are the eigenvectors of the *covariance matrix* **X**^{T}**X**, since

$$\mathbf{X}^{\mathsf{T}}\mathbf{X} = \left(\mathbf{U}\Sigma\mathbf{V}^{\mathsf{T}}\right)^{\mathsf{T}}\left(\mathbf{U}\Sigma\mathbf{V}^{\mathsf{T}}\right) = \mathbf{V}\Sigma^{2}\mathbf{V}^{\mathsf{T}}$$

- **U** are the eigenvectors of the *Gram (inner-product) matrix* **XX**^{T}, since

$$\mathbf{X}\mathbf{X}^{\mathsf{T}} = \left(\mathbf{U}\Sigma\mathbf{V}^{\mathsf{T}}\right)\left(\mathbf{U}\Sigma\mathbf{V}^{\mathsf{T}}\right)^{\mathsf{T}} = \mathbf{U}\Sigma^{2}\mathbf{U}^{\mathsf{T}}$$

Further reading:
1. Ian T. Jolliffe, *Principal Component Analysis* (2nd ed), Springer, 2002.
2. Gilbert Strang, *Linear Algebra and Its Applications* (4th ed), Brooks Cole, 2005.

SVD - Interpretation

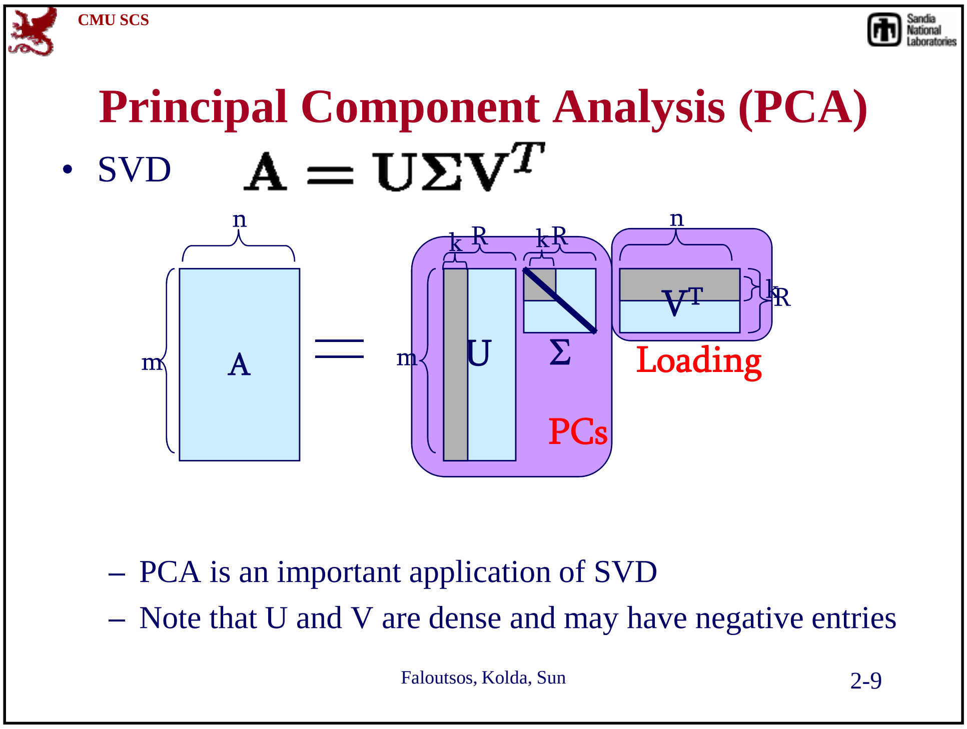
'documents', 'terms' and 'concepts':

Q**:** if **A** is the document-to-term matrix, what is **A**^{T} **A**?

A: term-to-term ([m x m]) similarity matrix

Q: **A A**^{T} ?

A: document-to-document ([n x n]) similarity matrix



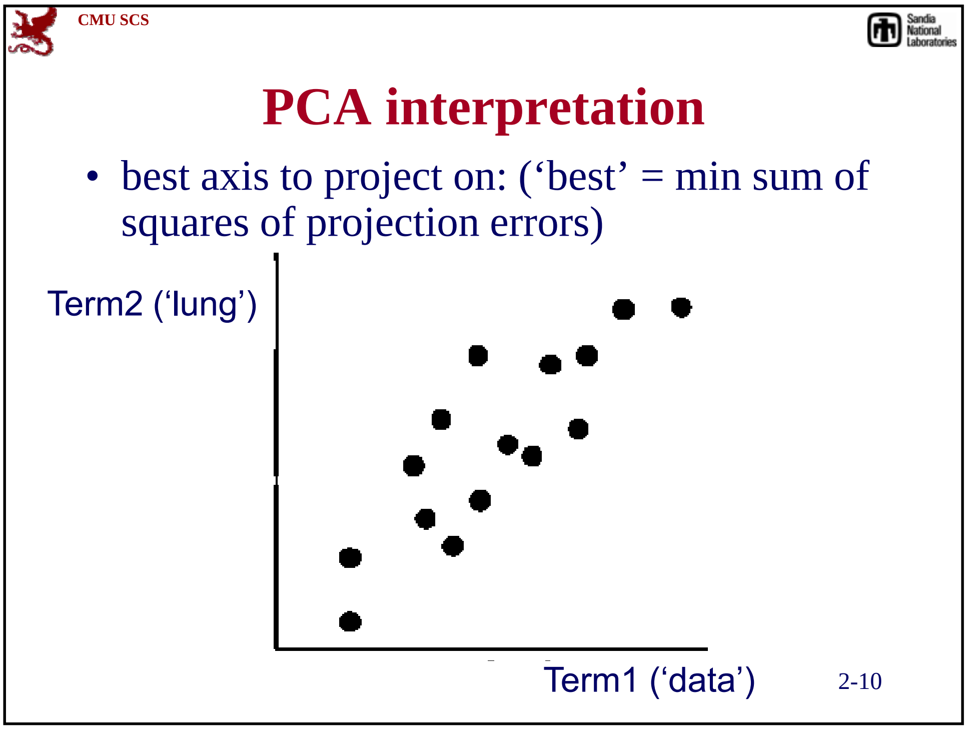
CMU SCS

Principal Component Analysis (PCA)

- SVD $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$

- PCA is an important application of SVD
- Note that U and V are dense and may have negative entries

Faloutsos, Kolda, Sun 2-9

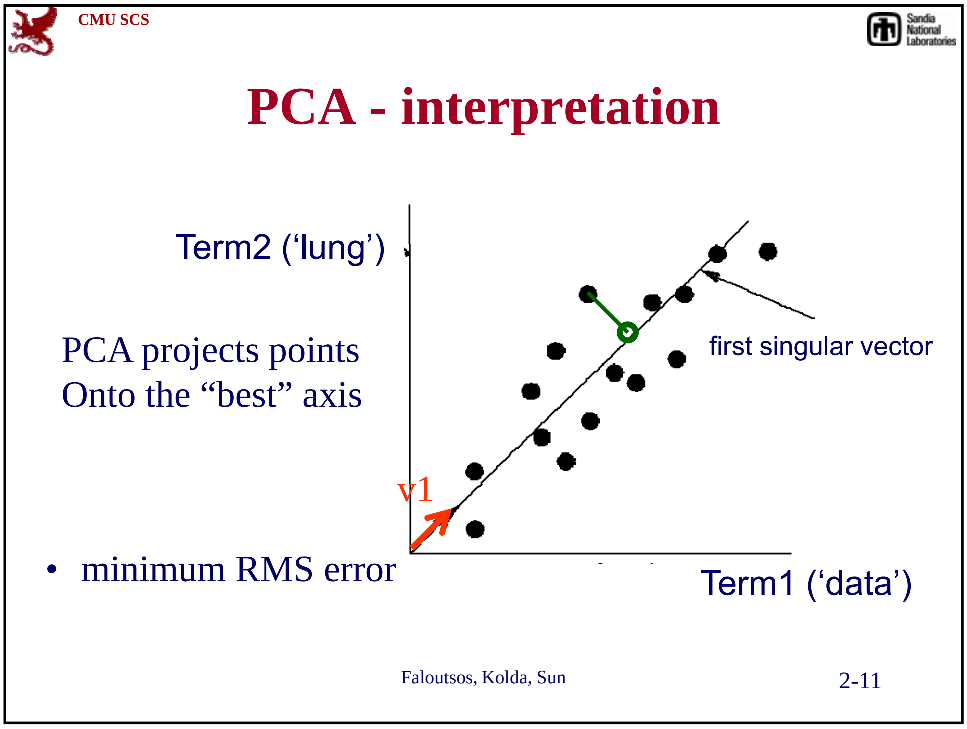


CMU SCS

PCA interpretation

- best axis to project on: ('best' = min sum of squares of projection errors)

2-10

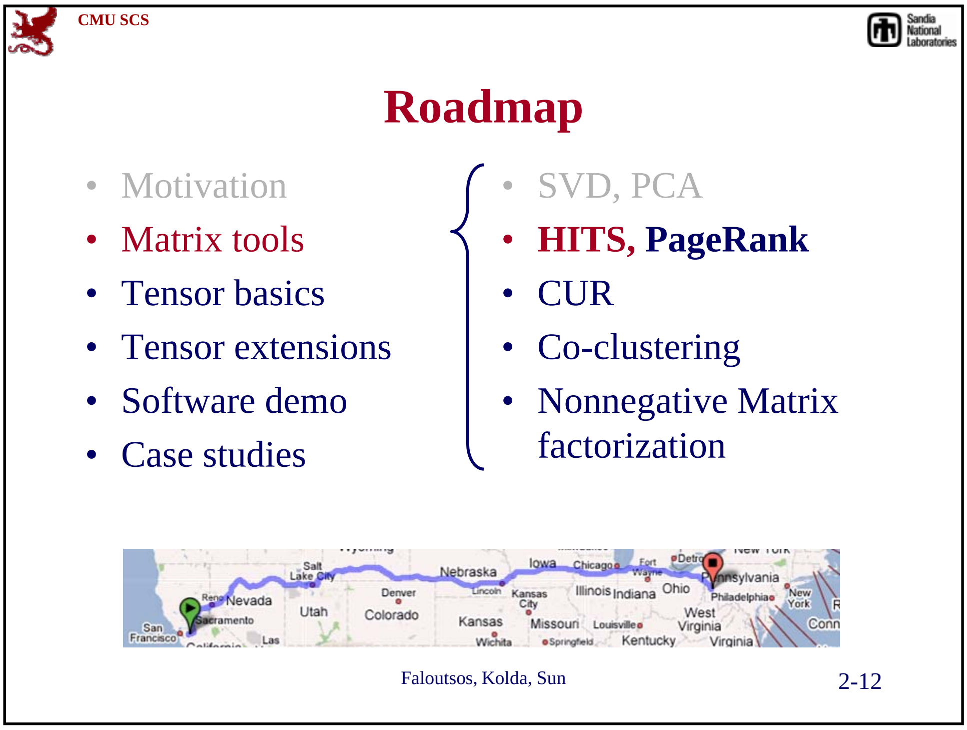


CMU SCS

PCA - interpretation

PCA projects points
Onto the “best” axis

- minimum RMS error



CMU SCS

Roadmap

- Motivation
- Matrix tools
  - SVD, PCA
  - **HITS, PageRank**
  - CUR
  - Co-clustering
  - Nonnegative Matrix factorization
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

San Francisco
Sacramento
Reno
Nevada
Salt Lake City
Utah
Denver
Colorado
Nebraska
Lincoln
Kansas
Kansas City
Wichita
Iowa
Chicago
Missouri
Springfield
Illinois
Indiana
Louisville
Kentucky
Ohio
Fort Wayne
West Virginia
Virginia
Pennsylvania
Philadelphia
New York
Conn

CMU SCS

Kleinberg's algorithm HITS

- Problem dfn: given the web and a query
- find the most 'authoritative' web pages for this query

Step 0: find all pages containing the query terms

Step 1: expand by one move forward and backward

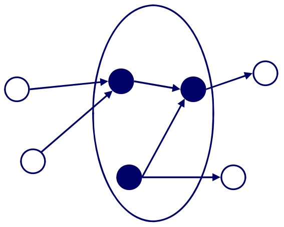
Further reading:

1. J. Kleinberg. Authoritative sources in a hyperlinked environment. SODA 1998

CMU SCS

Kleinberg's algorithm HITS

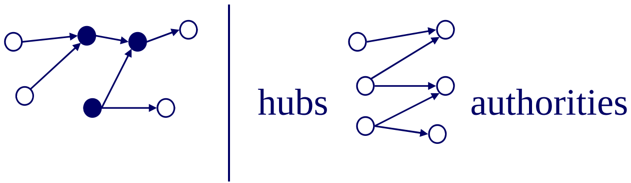
- Step 1: expand by one move forward and backward



CMU SCS

Kleinberg's algorithm HITS

- on the resulting graph, give high score (= 'authorities') to nodes that many important nodes point to
- give high importance score ('hubs') to nodes that point to good 'authorities'



hubs authorities

Faloutsos, Kolda, Sun 2-15

CMU SCS

Kleinberg's algorithm HITS

observations

- recursive definition!
- each node (say, '*i*'-th node) has both an authoritativeness score a_i and a hubness score h_i

Faloutsos, Kolda, Sun 2-16

Kleinberg's algorithm: HITS

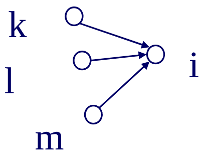
Let **A** be the adjacency matrix:

the (i,j) entry is 1 if the edge from i to j exists

Let **h** and **a** be [n x 1] vectors with the 'hubness' and 'authoritativiness' scores.

Then:

Kleinberg's algorithm: HITS



Then:

$$a_i = h_k + h_l + h_m$$

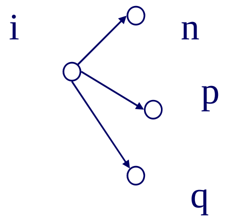
that is

a_i = Sum (h_j) over all j that (j,i) edge exists

or

$\mathbf{a} = \mathbf{A}^T \mathbf{h}$

Kleinberg's algorithm: HITS



symmetrically, for the 'hubness':

$$h_i = a_n + a_p + a_q$$

that is

h_i = Sum (q_j) over all j that (i,j) edge exists

or

$\mathbf{h} = \mathbf{A}\ \mathbf{a}$

Kleinberg's algorithm: HITS

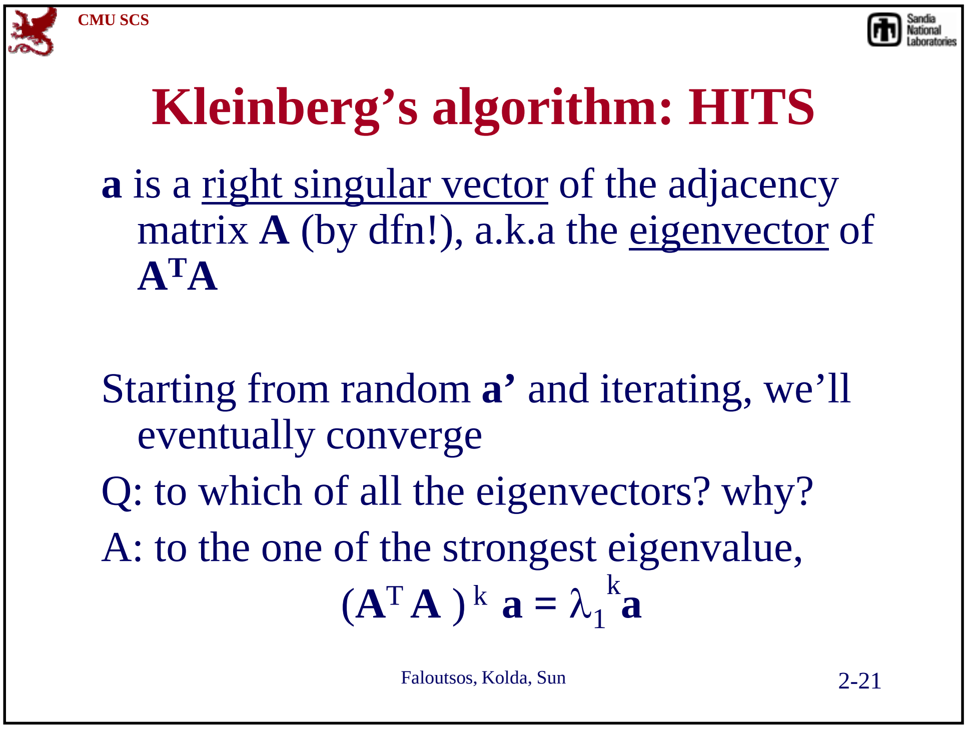
In conclusion, we want vectors **h** and **a** such that:

$$\mathbf{h} = \mathbf{A}\ \mathbf{a}$$

$$\mathbf{a} = \mathbf{A}^T\ \mathbf{h}$$

That is:

$$\mathbf{a} = \mathbf{A}^T\mathbf{A}\ \mathbf{a}$$



CMU SCS

Kleinberg's algorithm: HITS

a is a right singular vector of the adjacency matrix **A** (by dfn!), a.k.a the eigenvector of **A**$^{\mathbf{T}}$**A**

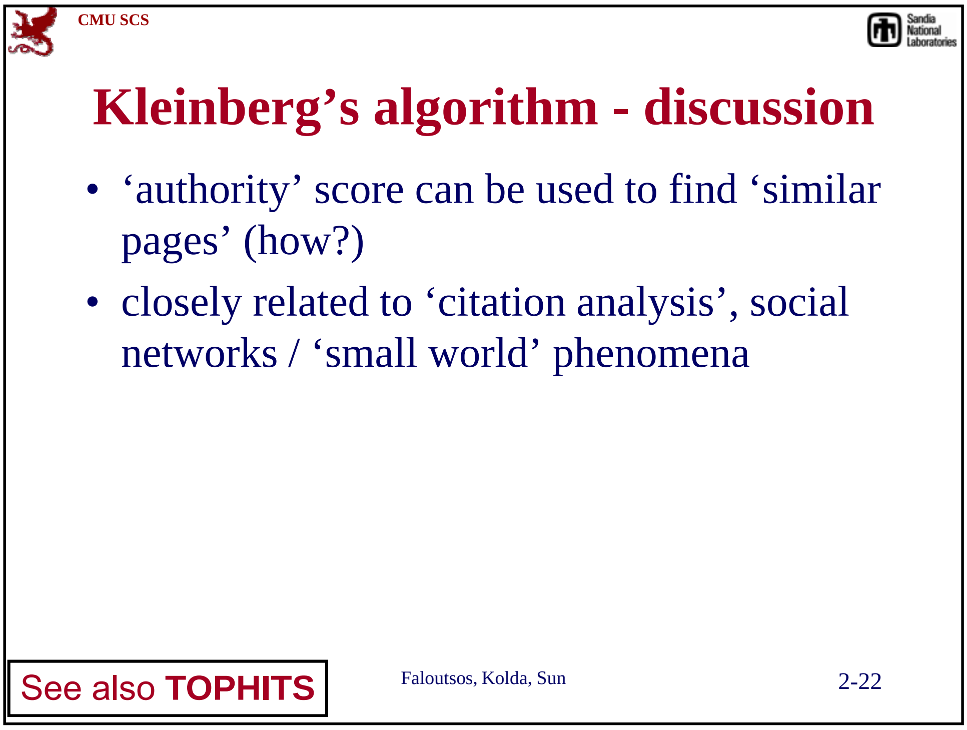
Starting from random **a'** and iterating, we'll eventually converge

Q: to which of all the eigenvectors? why?

A: to the one of the strongest eigenvalue,

$$(\mathbf{A}^{T}\mathbf{A})^{k}\ \mathbf{a} = \lambda_1^{k}\mathbf{a}$$

Faloutsos, Kolda, Sun 2-21



CMU SCS

Kleinberg's algorithm - discussion

- 'authority' score can be used to find 'similar pages' (how?)
- closely related to 'citation analysis', social networks / 'small world' phenomena

See also **TOPHITS**

Faloutsos, Kolda, Sun 2-22

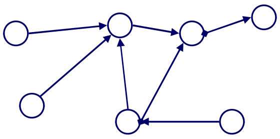
Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

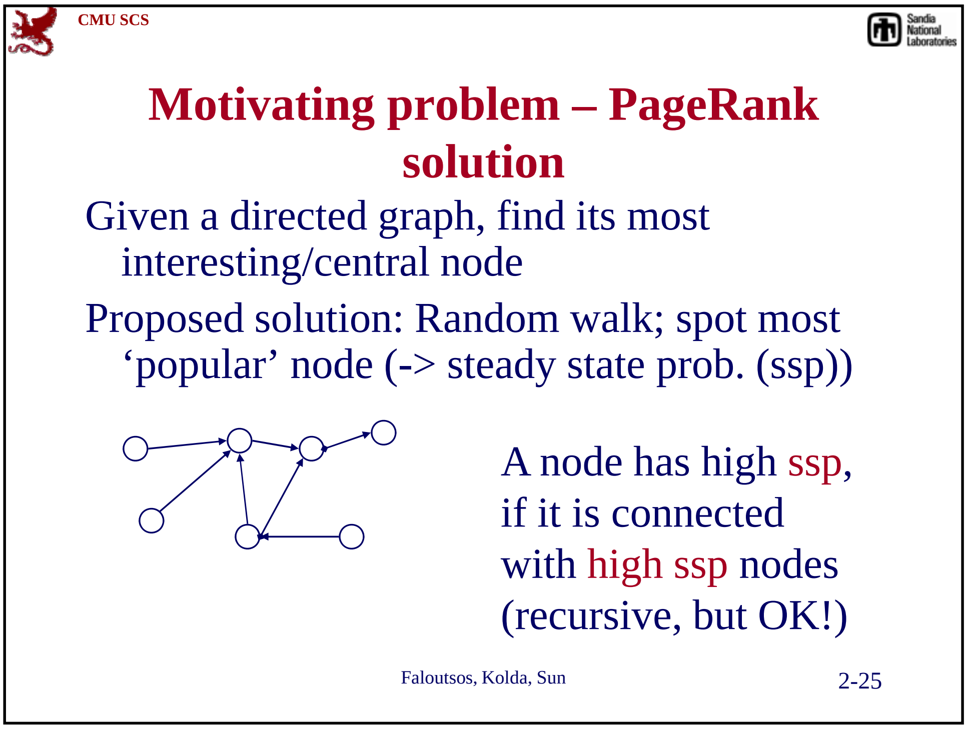
- SVD, PCA
- HITS, **PageRank**
- CUR
- Co-clustering
- Nonnegative Matrix factorization

Motivating problem: PageRank

Given a directed graph, find its most interesting/central node



A node is important, if it is connected with important nodes (recursive, but OK!)

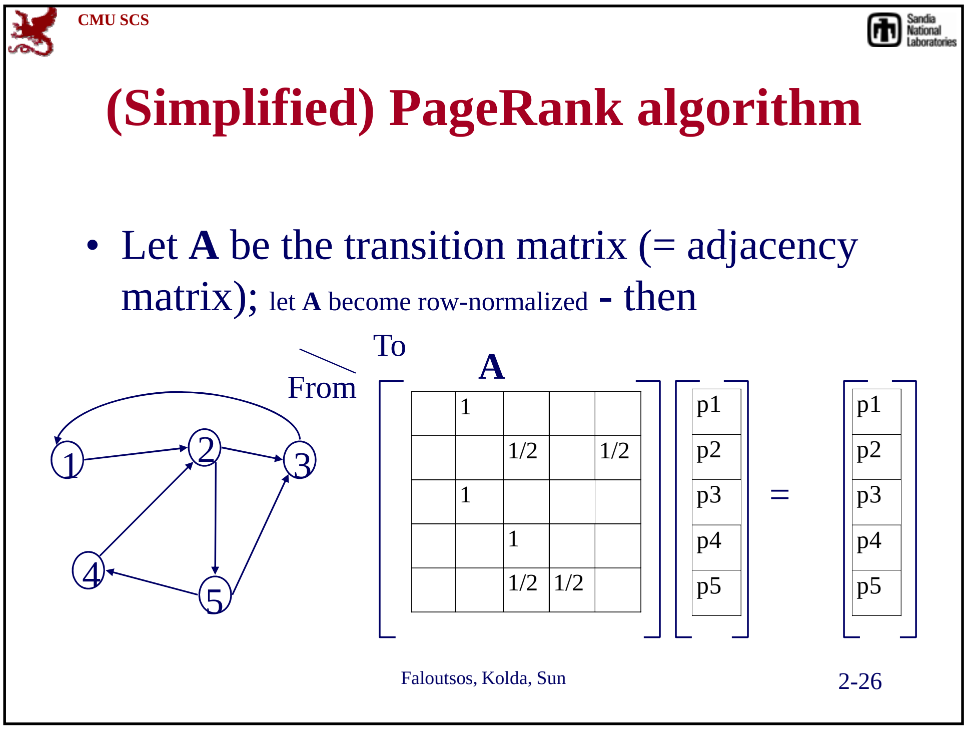


Motivating problem – PageRank solution

Given a directed graph, find its most interesting/central node

Proposed solution: Random walk; spot most 'popular' node (-> steady state prob. (ssp))

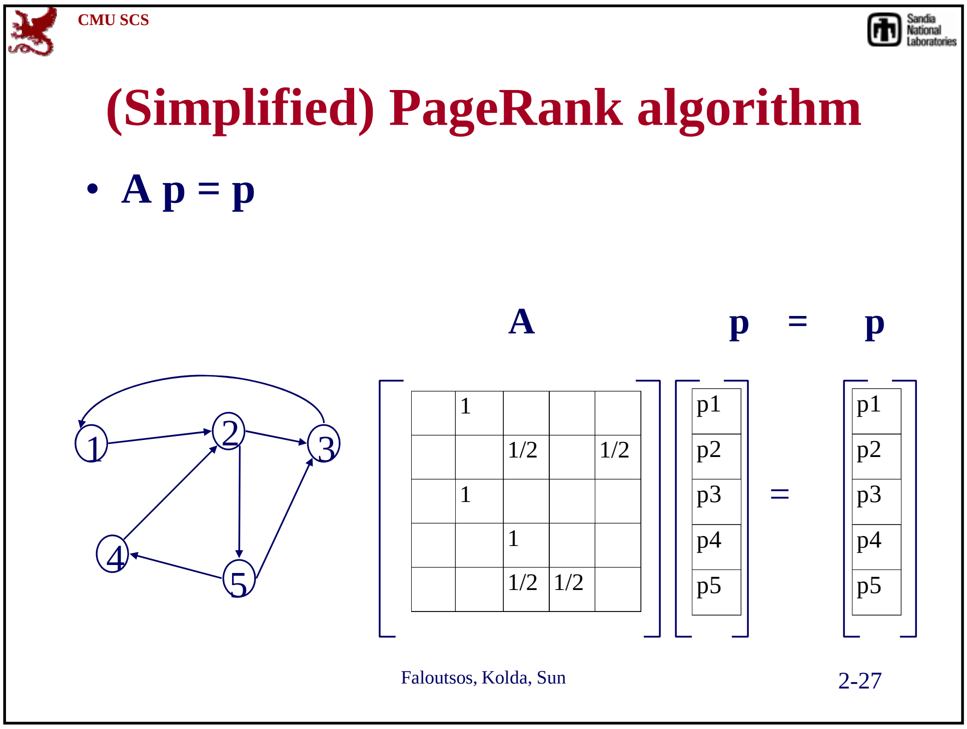
A node has high ssp, if it is connected with high ssp nodes (recursive, but OK!)



(Simplified) PageRank algorithm

- Let **A** be the transition matrix (= adjacency matrix); let **A** become row-normalized - then

$$\overset{\mathbf{A}}{\begin{bmatrix} & 1 & & & \\ & & 1/2 & & 1/2 \\ & 1 & & & \\ & & 1 & & \\ & & 1/2 & 1/2 & \end{bmatrix}}\begin{bmatrix}p1\\p2\\p3\\p4\\p5\end{bmatrix}=\begin{bmatrix}p1\\p2\\p3\\p4\\p5\end{bmatrix}$$

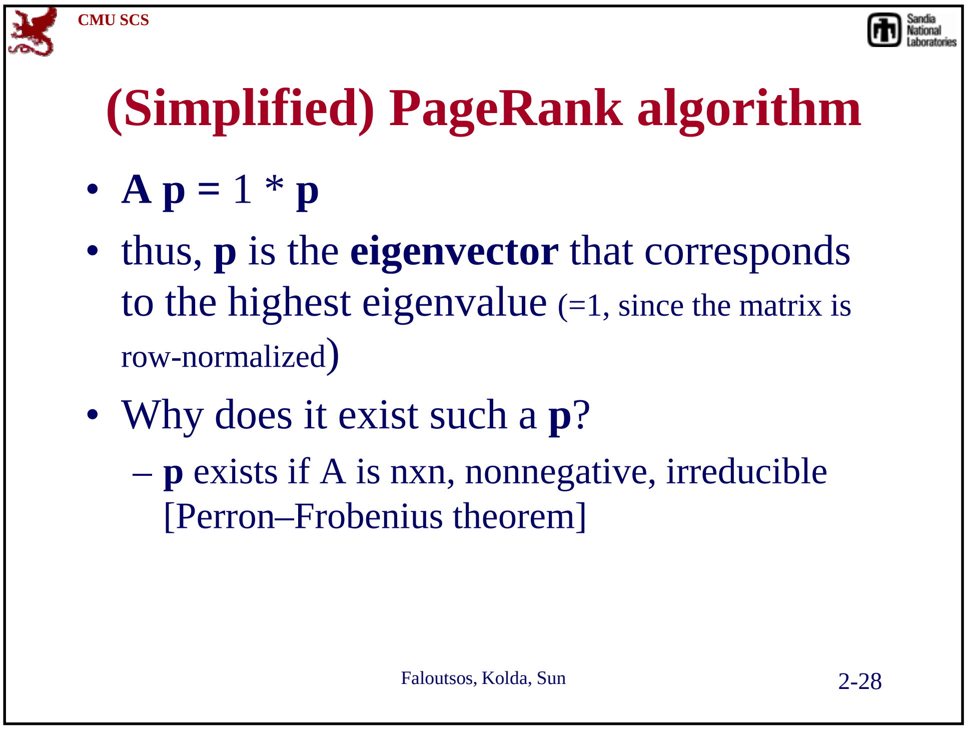


CMU SCS

(Simplified) PageRank algorithm

- **A p = p**

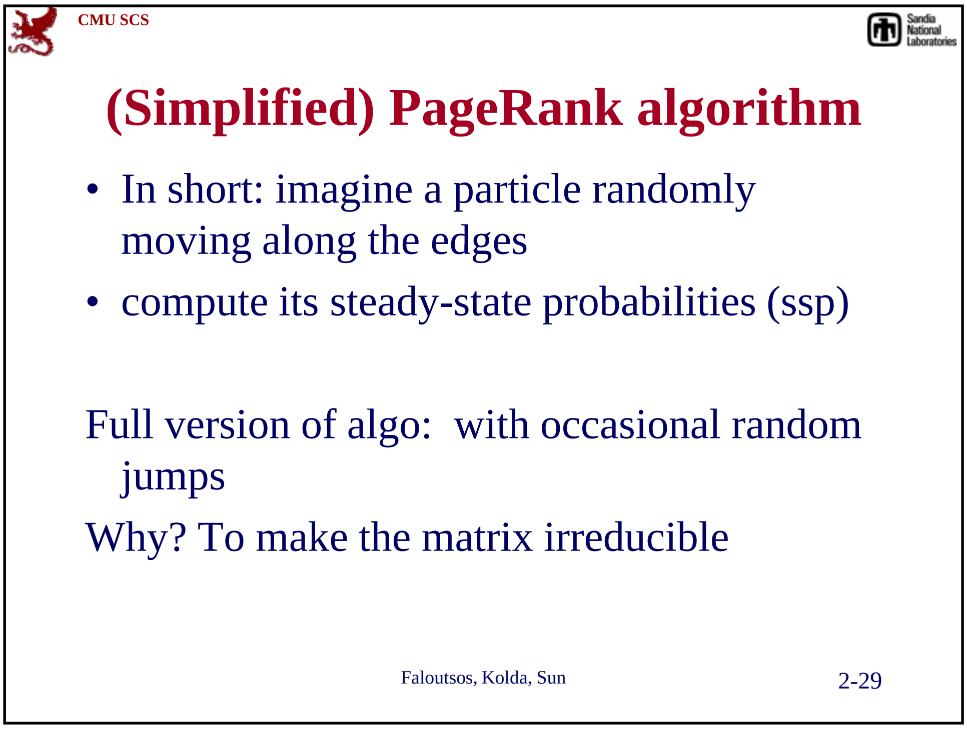
		A			p	=	p
	1				p1		p1
		1/2		1/2	p2		p2
	1				p3	=	p3
		1			p4		p4
		1/2	1/2		p5		p5



CMU SCS

(Simplified) PageRank algorithm

- **A p** = 1 * **p**
- thus, **p** is the **eigenvector** that corresponds to the highest eigenvalue (=1, since the matrix is row-normalized)
- Why does it exist such a **p**?
  - **p** exists if A is nxn, nonnegative, irreducible [Perron–Frobenius theorem]



CMU SCS

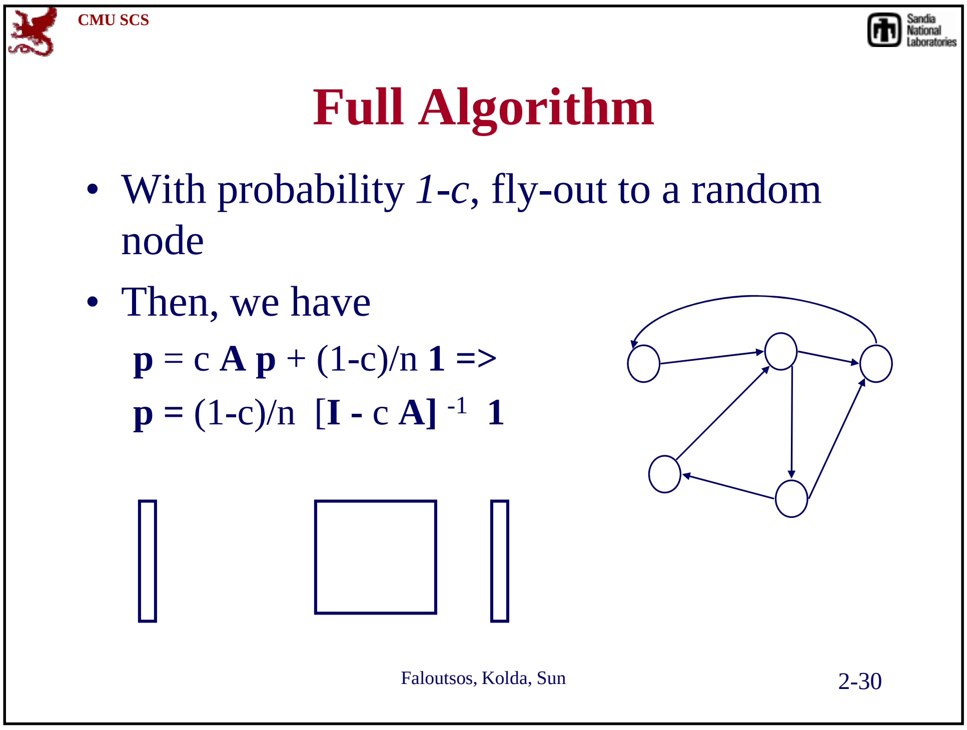
(Simplified) PageRank algorithm

- In short: imagine a particle randomly moving along the edges
- compute its steady-state probabilities (ssp)

Full version of algo: with occasional random jumps

Why? To make the matrix irreducible

Faloutsos, Kolda, Sun 2-29



CMU SCS

Full Algorithm

- With probability *1-c*, fly-out to a random node
- Then, we have

$\mathbf{p} = c\,\mathbf{A}\,\mathbf{p} + (1\text{-}c)/n\ \mathbf{1} \Rightarrow$

$\mathbf{p} = (1\text{-}c)/n\ [\mathbf{I} - c\,\mathbf{A}]^{-1}\ \mathbf{1}$

Faloutsos, Kolda, Sun 2-30

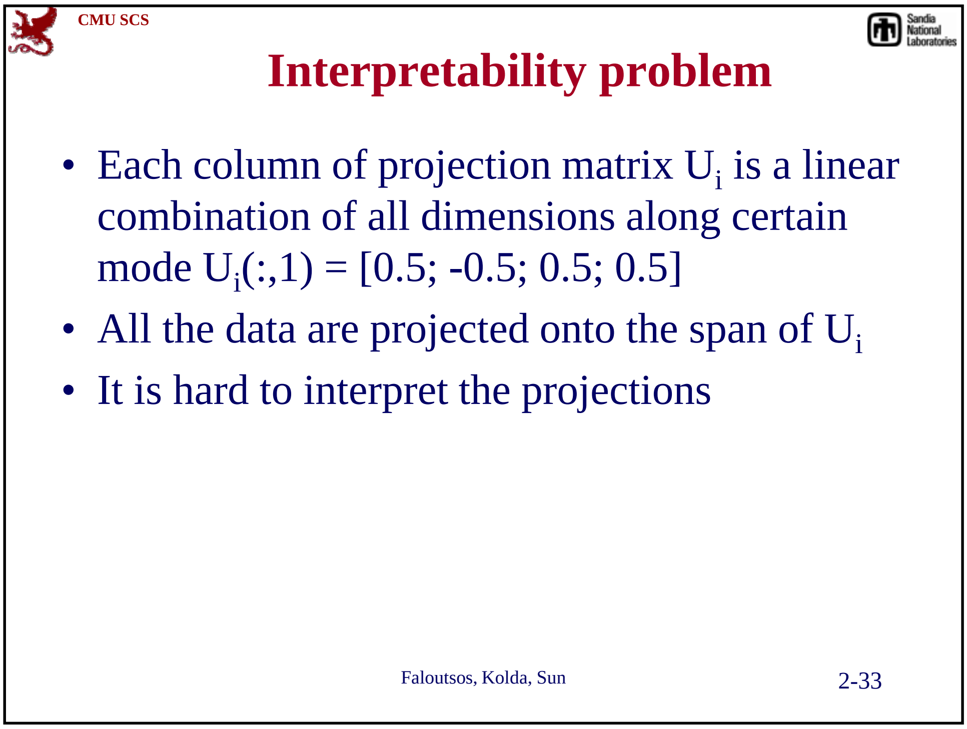
Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

- SVD, PCA
- HITS, PageRank
- **CUR**
- Co-clustering
- Nonnegative Matrix factorization

Motivation of CUR or CMD

- SVD, PCA all transform data into some abstract space (specified by a set basis)
  – Interpretability problem
  – Loss of sparsity

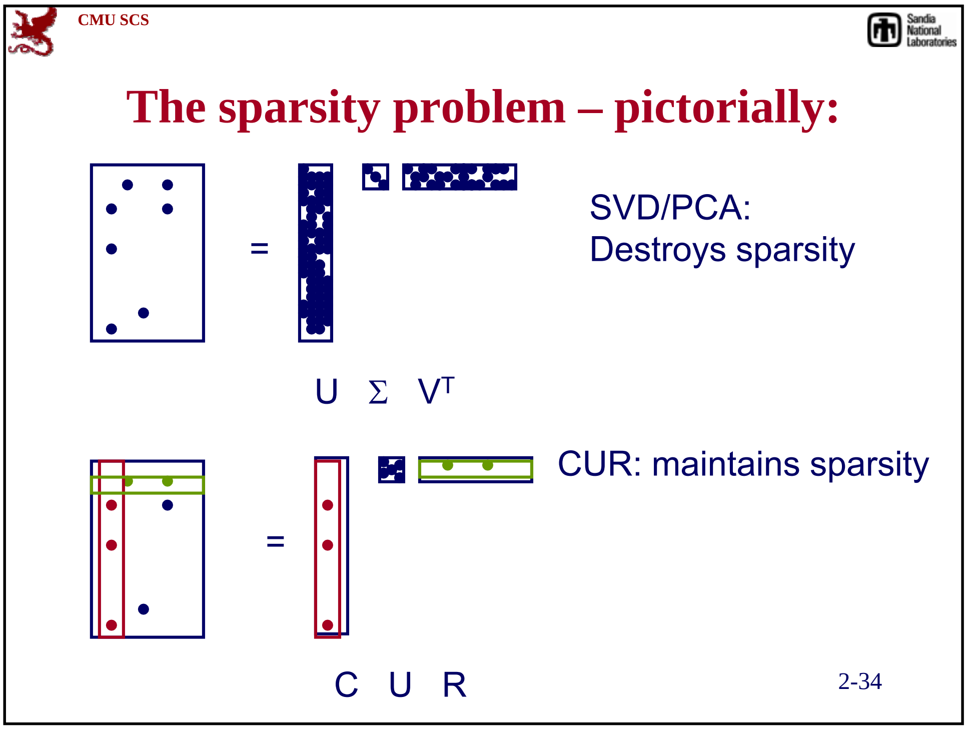


CMU SCS

Interpretability problem

- Each column of projection matrix U_i is a linear combination of all dimensions along certain mode $U_i(:,1) = [0.5; -0.5; 0.5; 0.5]$
- All the data are projected onto the span of U_i
- It is hard to interpret the projections

Faloutsos, Kolda, Sun 2-33



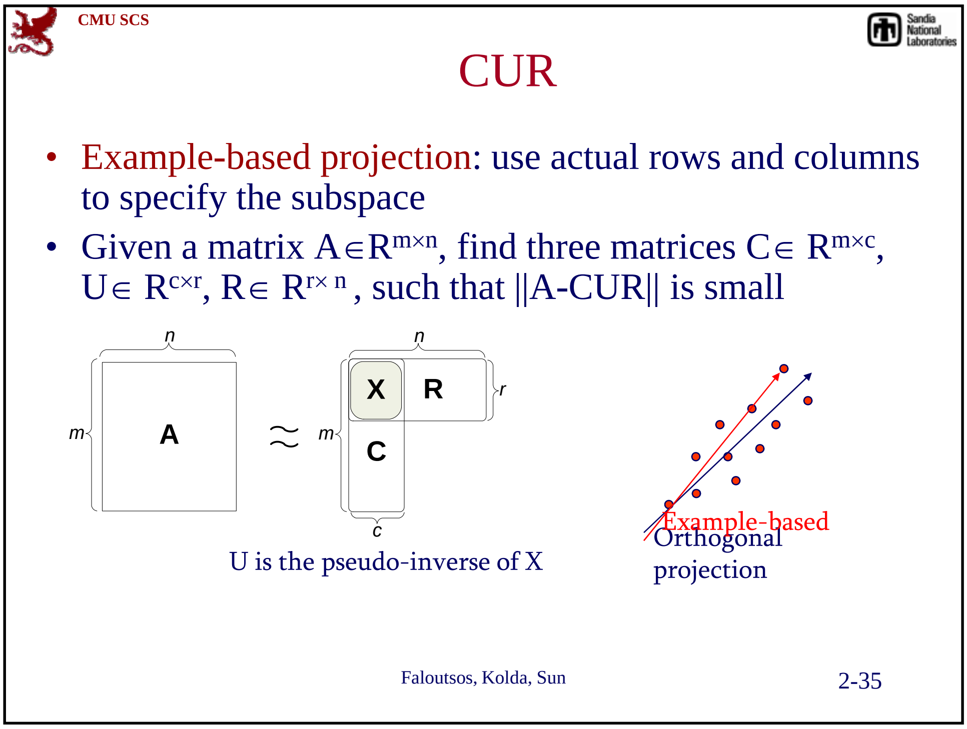
CMU SCS

The sparsity problem – pictorially:

SVD/PCA: Destroys sparsity

CUR: maintains sparsity

2-34



CMU SCS

CUR

- Example-based projection: use actual rows and columns to specify the subspace
- Given a matrix $A \in R^{m\times n}$, find three matrices $C \in R^{m\times c}$, $U \in R^{c\times r}$, $R \in R^{r\times n}$, such that $\|A-CUR\|$ is small

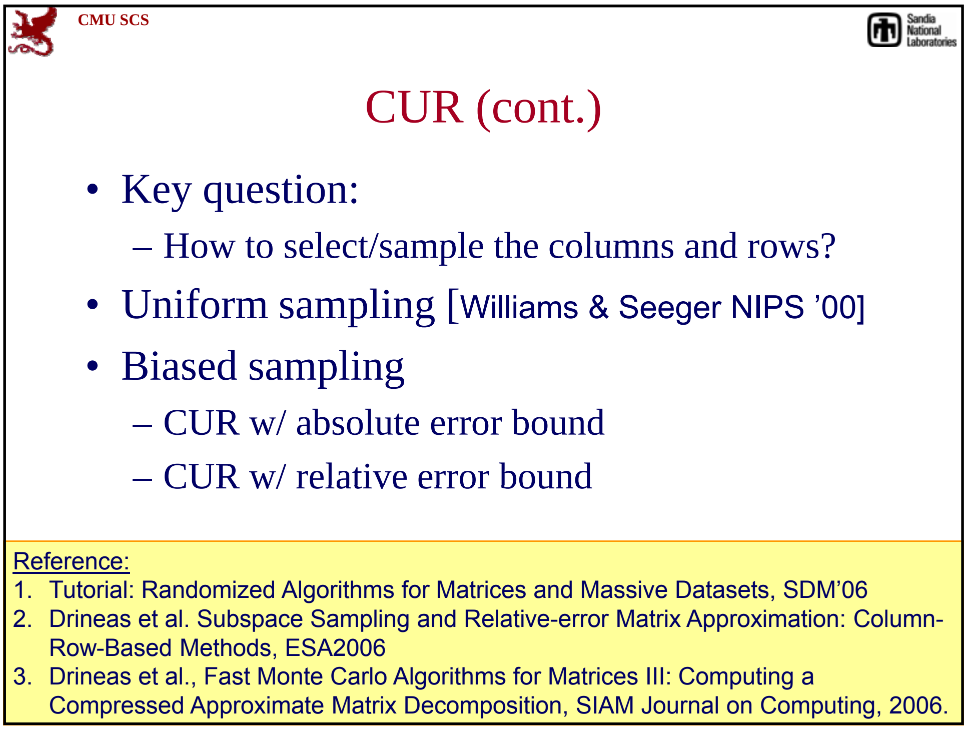
n
m
A
≈
n
X
R
r
m
C
c

U is the pseudo-inverse of X

Example-based

Orthogonal projection

Faloutsos, Kolda, Sun 2-35

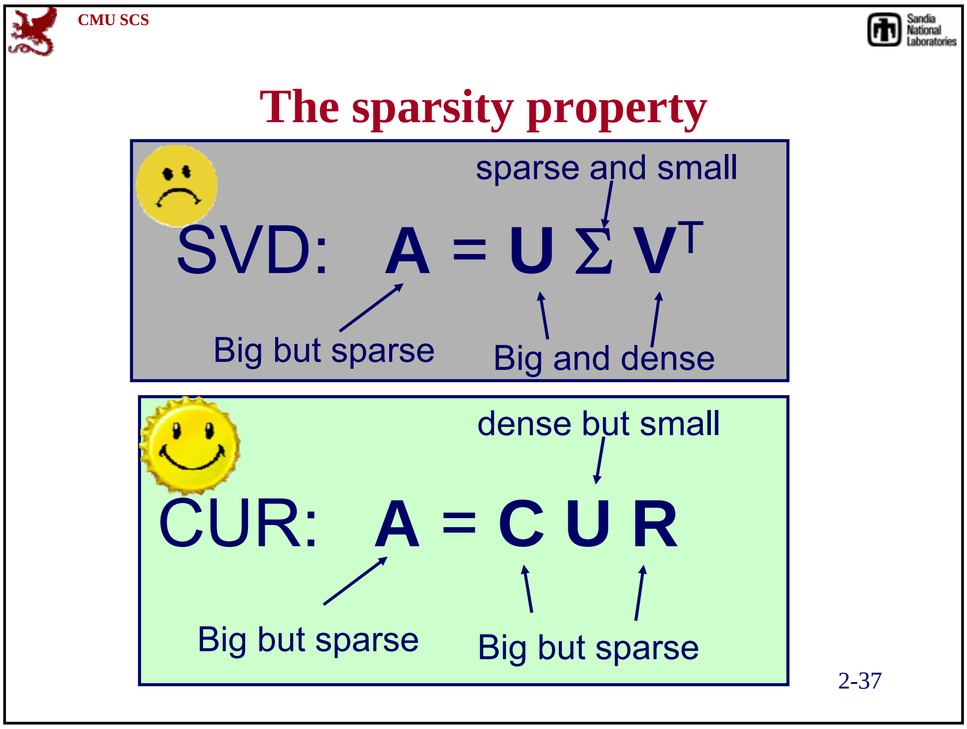


CMU SCS

CUR (cont.)

- Key question:
  - How to select/sample the columns and rows?
- Uniform sampling [Williams & Seeger NIPS '00]
- Biased sampling
  - CUR w/ absolute error bound
  - CUR w/ relative error bound

Reference:
1. Tutorial: Randomized Algorithms for Matrices and Massive Datasets, SDM'06
2. Drineas et al. Subspace Sampling and Relative-error Matrix Approximation: Column-Row-Based Methods, ESA2006
3. Drineas et al., Fast Monte Carlo Algorithms for Matrices III: Computing a Compressed Approximate Matrix Decomposition, SIAM Journal on Computing, 2006.



CMU SCS

The sparsity property

sparse and small

SVD: $\mathbf{A} = \mathbf{U}\,\Sigma\,\mathbf{V}^T$

Big but sparse

Big and dense

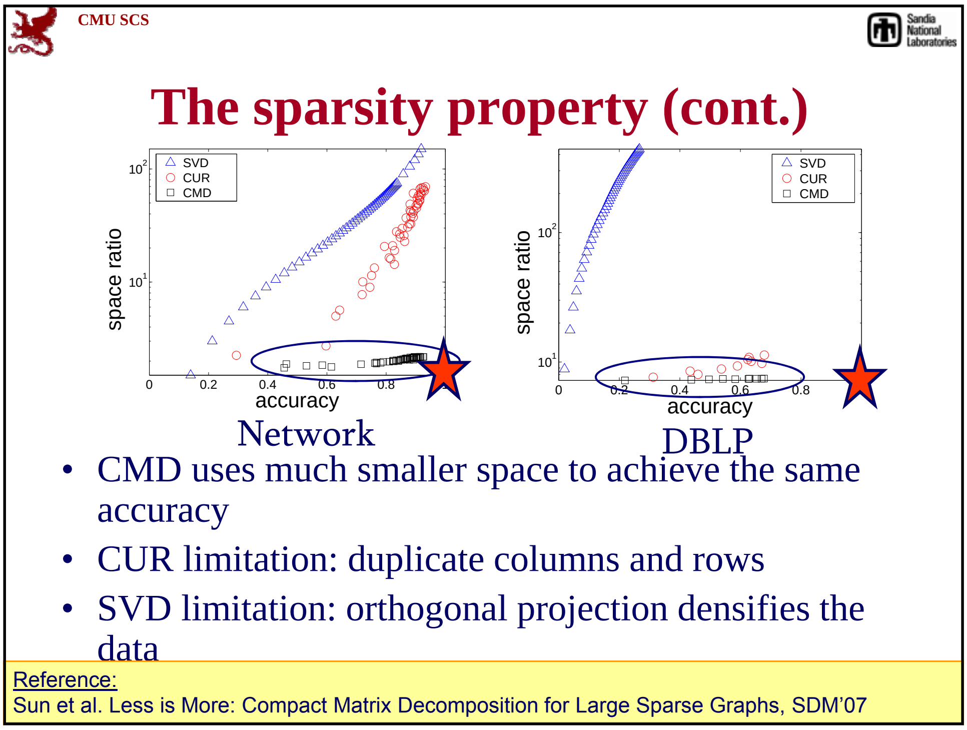
dense but small

CUR: $\mathbf{A} = \mathbf{C}\,\mathbf{U}\,\mathbf{R}$

Big but sparse

Big but sparse

2-37



CMU SCS

The sparsity property (cont.)

Network

DBLP

- CMD uses much smaller space to achieve the same accuracy
- CUR limitation: duplicate columns and rows
- SVD limitation: orthogonal projection densifies the data

Reference:
Sun et al. Less is More: Compact Matrix Decomposition for Large Sparse Graphs, SDM'07

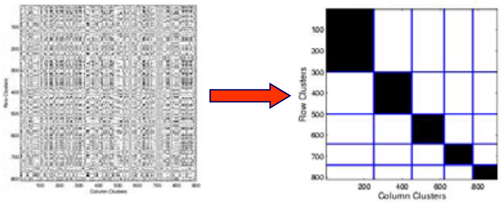
Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

- SVD, PCA
- HITS, PageRank
- CUR
- **Co-clustering etc**
- Nonnegative Matrix factorization

Co-clustering

- Given data matrix and the number of row and column groups *k* and *l*
- Simultaneously
  - Cluster rows of *p(X, Y)* into *k* disjoint groups
  - Cluster columns of *p(X, Y)* into *l* disjoint groups



CMU SCS

Co-clustering

- Let *X* and *Y* be discrete random variables
  - *X* and *Y* take values in *{1, 2, …, m}* and *{1, 2, …, n}*
  - *p(X, Y)* denotes the joint probability distribution—if not known, it is often estimated based on co-occurrence data
  - Application areas: text mining, market-basket analysis, analysis of browsing behavior, etc.
- Key Obstacles in Clustering Contingency Tables
  - High Dimensionality, Sparsity, Noise
  - Need for robust and scalable algorithms

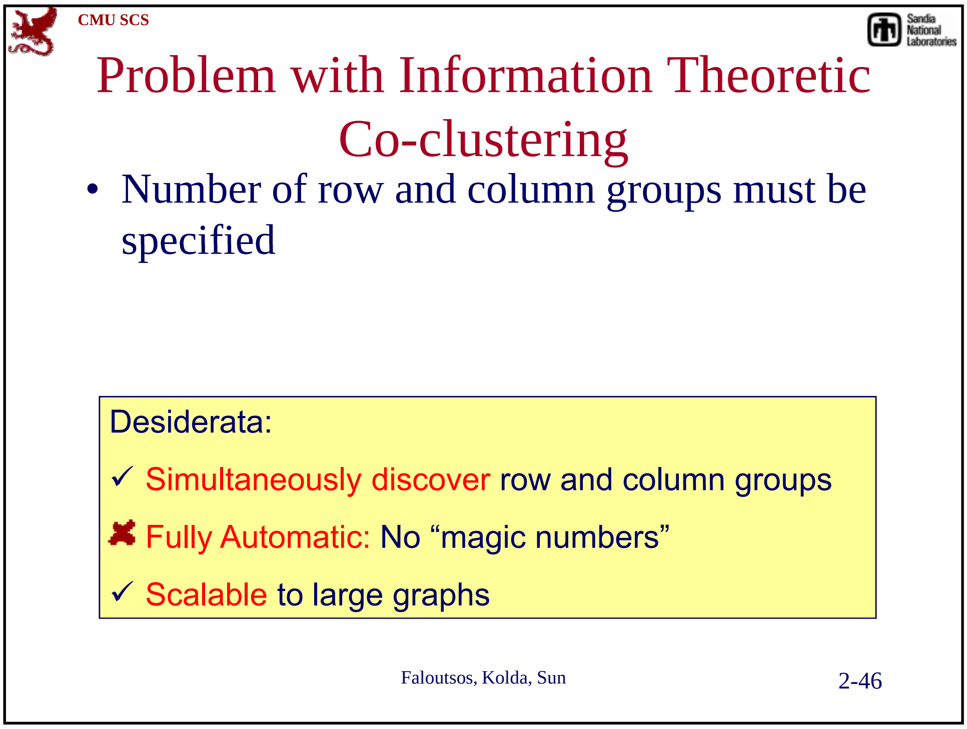
Reference:
1. Dhillon et al. Information-Theoretic Co-clustering, KDD’03

CMU SCS

$$p(x,y) = \begin{array}{cc} & n \\ m & \begin{bmatrix} .05 & .05 & .05 & 0 & 0 & 0 \\ .05 & .05 & .05 & 0 & 0 & 0 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ 0 & 0 & 0 & .05 & .05 & .05 \\ .04 & .04 & 0 & .04 & .04 & .04 \\ .04 & .04 & .04 & 0 & .04 & .04 \end{bmatrix} \end{array}$$

$$\underset{p(x|\hat{x})}{\begin{array}{cc} & k \\ m & \begin{bmatrix} .5 & 0 & 0 \\ .5 & 0 & 0 \\ 0 & .5 & 0 \\ 0 & .5 & 0 \\ 0 & 0 & .5 \\ 0 & 0 & .5 \end{bmatrix} \end{array}} \underset{p(\hat{x},\hat{y})}{\begin{array}{cc} & l \\ k & \begin{bmatrix} .3 & 0 \\ 0 & .3 \\ .2 & .2 \end{bmatrix} \end{array}} \underset{p(y|\hat{y})}{\begin{array}{cc} & n \\ l & \begin{bmatrix} .36 & .36 & .28 & 0 & 0 & 0 \\ 0 & 0 & 0 & .28 & .36 & .36 \end{bmatrix} \end{array}} = \underset{q(x,y)}{\left[\begin{array}{ccc|ccc} .054 & .054 & .042 & 0 & 0 & 0 \\ .054 & .054 & .042 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & .042 & .054 & .054 \\ 0 & 0 & 0 & .042 & .054 & .054 \\ \hline .036 & .036 & 028 & .028 & .036 & .036 \\ .036 & .036 & .028 & .028 & .036 & .036 \end{array}\right]}$$

#parameters that determine *q(x,y)* are: $(m-k)+(kl-1)+(n-l)$



CMU SCS

Problem with Information Theoretic Co-clustering

- Number of row and column groups must be specified

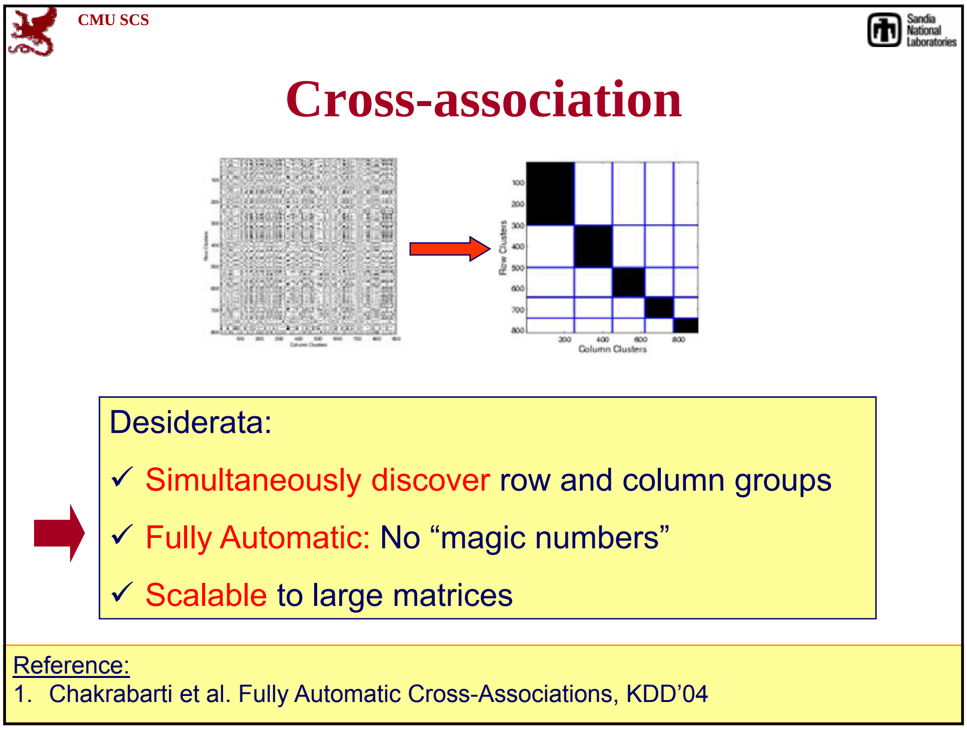
Desiderata:

✓ Simultaneously discover row and column groups

✗ Fully Automatic: No “magic numbers”

✓ Scalable to large graphs

Faloutsos, Kolda, Sun 2-46



CMU SCS

Cross-association

Row Clusters
100
200
300
400
500
600
700
800
200
400
600
800
Column Clusters

Desiderata:

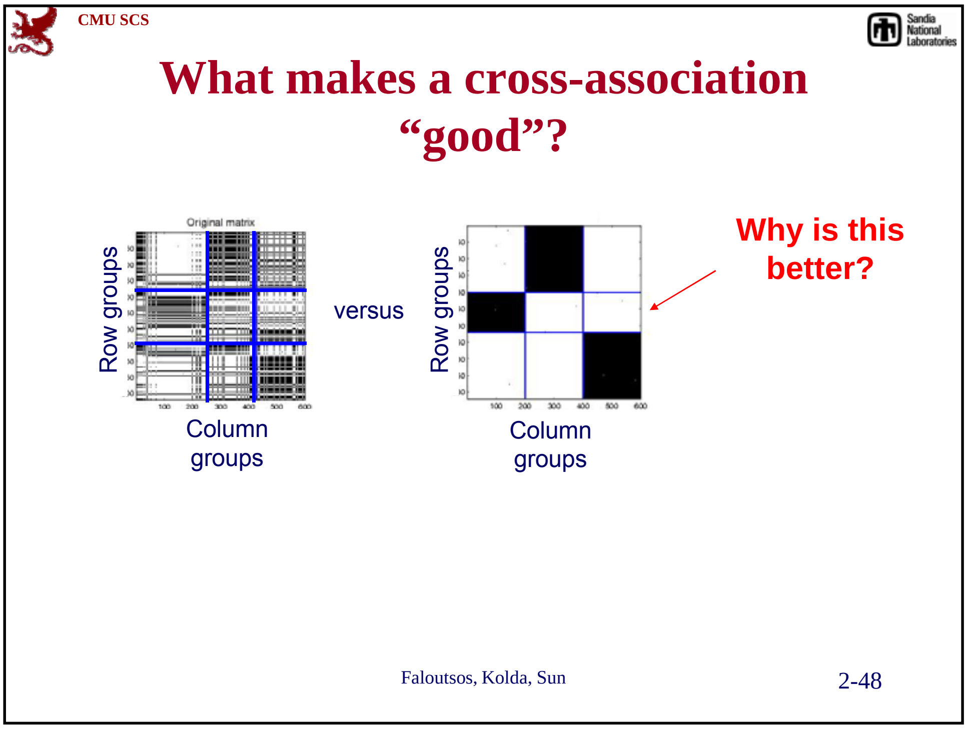
✓ Simultaneously discover row and column groups

✓ Fully Automatic: No “magic numbers”

✓ Scalable to large matrices

Reference:

1. Chakrabarti et al. Fully Automatic Cross-Associations, KDD’04



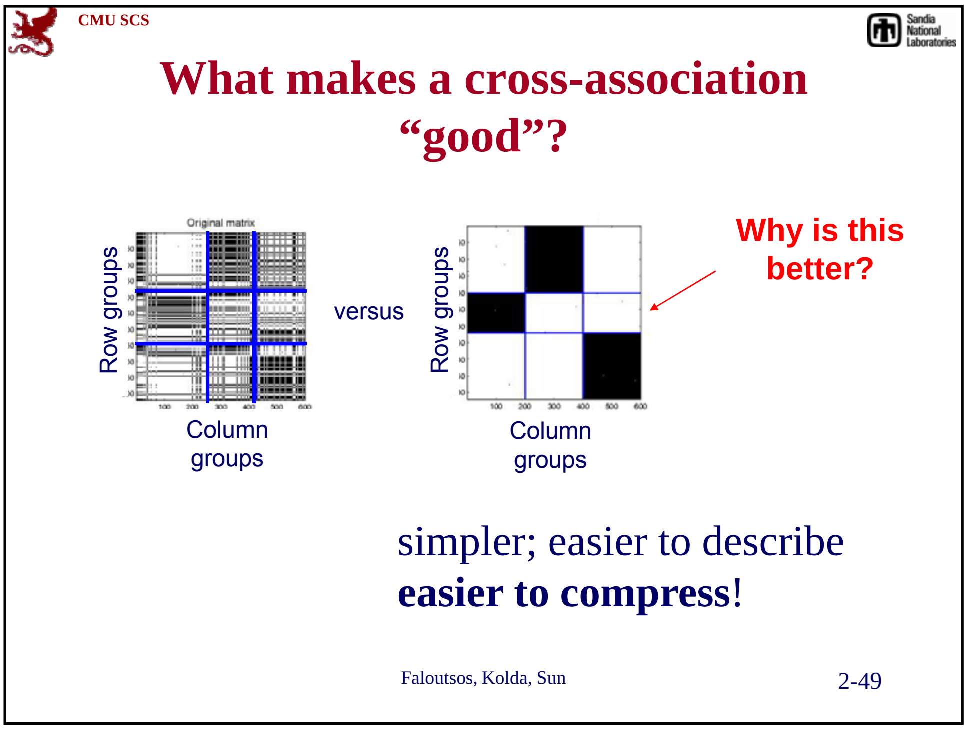
CMU SCS

What makes a cross-association "good"?

Why is this better?

Faloutsos, Kolda, Sun

2-48



CMU SCS

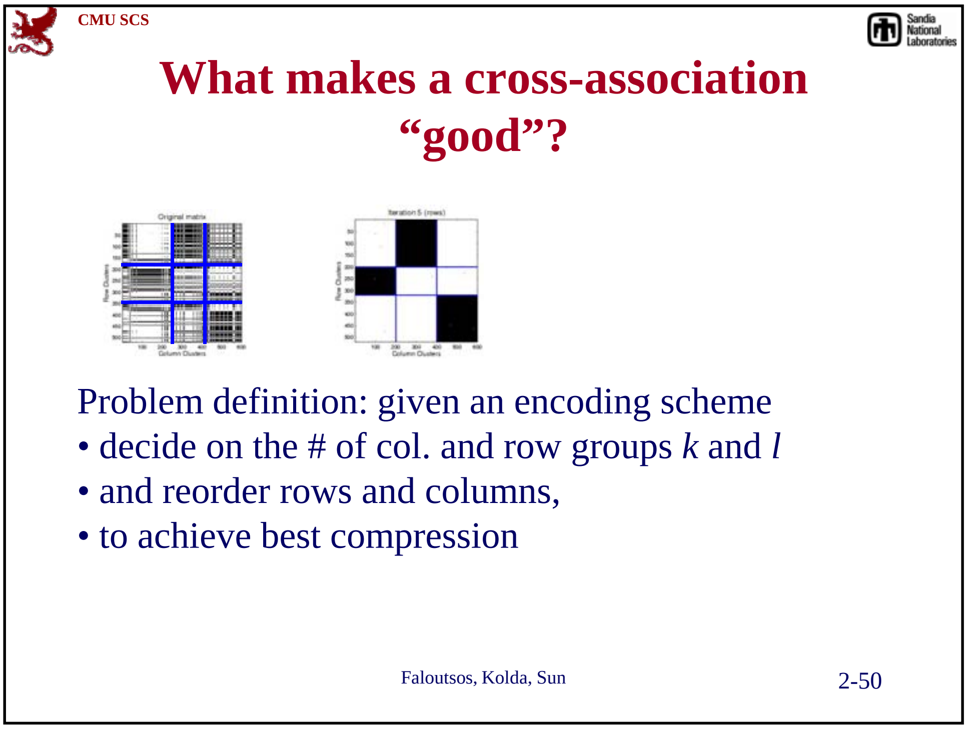
What makes a cross-association "good"?

Why is this better?

simpler; easier to describe
easier to compress!

Faloutsos, Kolda, Sun

2-49

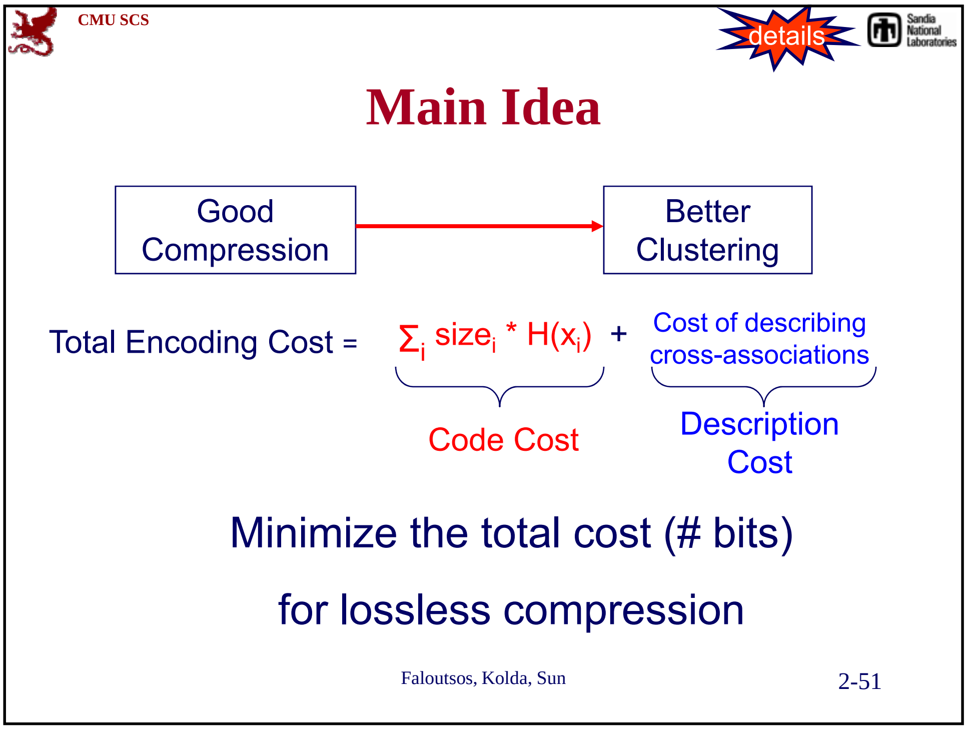


CMU SCS

What makes a cross-association "good"?

Original matrix
Iteration 5 (rows)

Problem definition: given an encoding scheme
- decide on the # of col. and row groups *k* and *l*
- and reorder rows and columns,
- to achieve best compression



CMU SCS

details

Main Idea

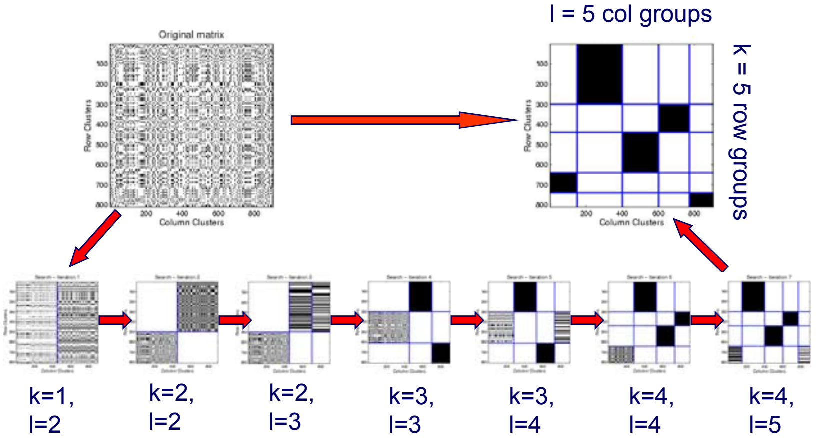
Good Compression → Better Clustering

$$\text{Total Encoding Cost} = \underbrace{\Sigma_i \, \text{size}_i * H(x_i)}_{\text{Code Cost}} + \underbrace{\text{Cost of describing cross-associations}}_{\text{Description Cost}}$$

Minimize the total cost (# bits)

for lossless compression

Algorithm



Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

- SVD, PCA
- HITS, PageRank
- CUR
- Co-clustering, etc
- **Nonnegative Matrix factorization**

CMU SCS

Nonnegative Matrix Factorization

- Coming up soon with **nonnegative tensor factorization**

Faloutsos, Kolda, Sun 2-56

CMU SCS

Roadmap

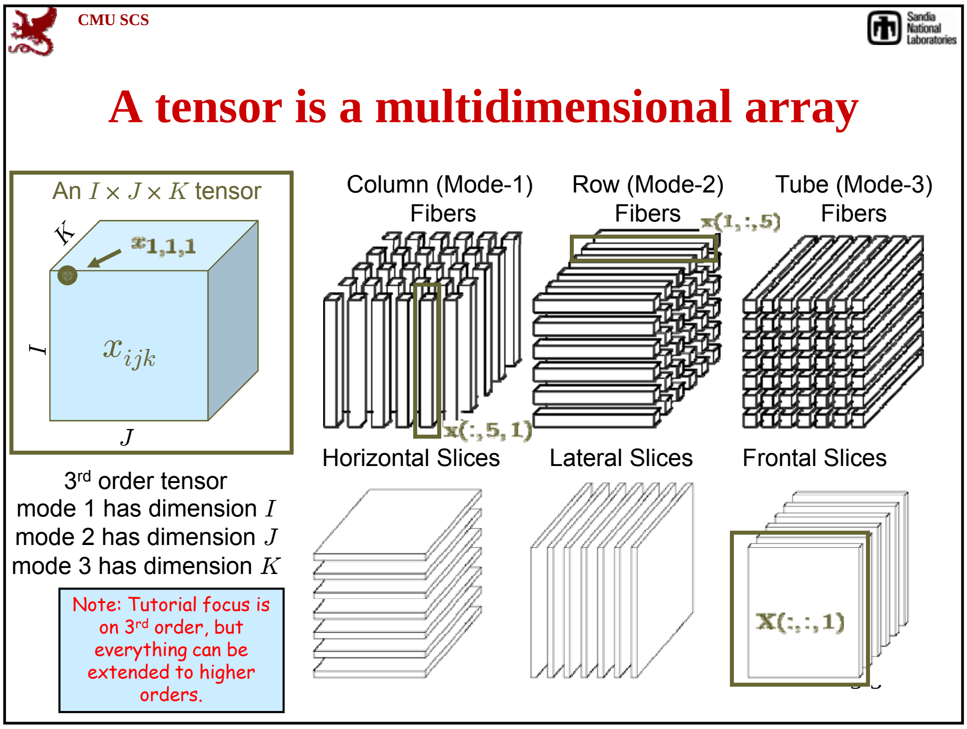
- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

- Tensor Basics
- Tucker
  - Tucker 1
  - Tucker 2
  - Tucker 3
- PARAFAC

3-1

CMU SCS

Tensor Basics



CMU SCS

A tensor is a multidimensional array

An $I \times J \times K$ tensor

3rd order tensor
mode 1 has dimension I
mode 2 has dimension J
mode 3 has dimension K

Note: Tutorial focus is on 3rd order, but everything can be extended to higher orders.

Column (Mode-1) Fibers

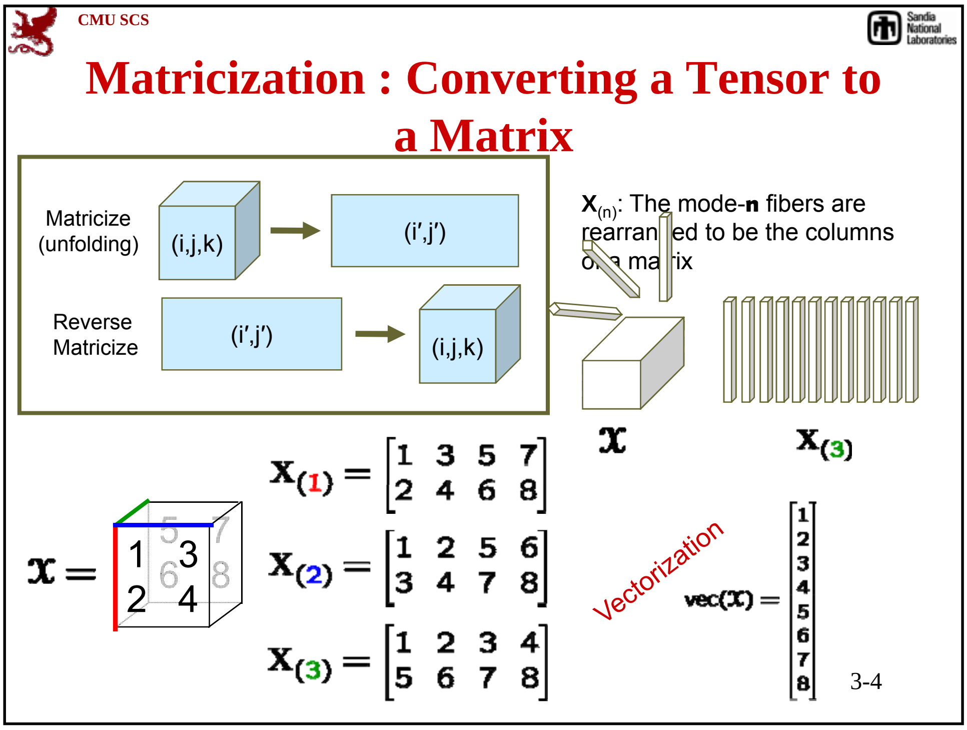
Row (Mode-2) Fibers

Tube (Mode-3) Fibers

Horizontal Slices

Lateral Slices

Frontal Slices



CMU SCS

Matricization : Converting a Tensor to a Matrix

Matricize (unfolding)

Reverse Matricize

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

$$\mathcal{X} = $$

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

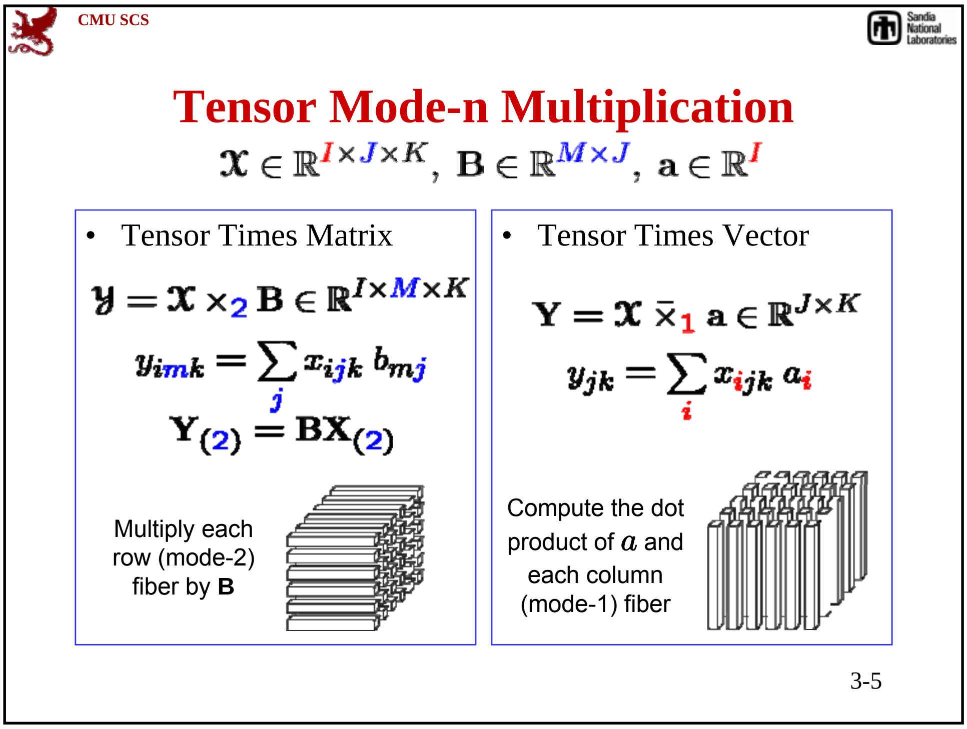
$$\mathbf{X}_{(2)} = \begin{bmatrix} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \end{bmatrix}$$

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

Vectorization

$$\mathrm{vec}(\mathcal{X}) = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{bmatrix}$$

3-4



CMU SCS

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 row (mode-2) fiber by **B**

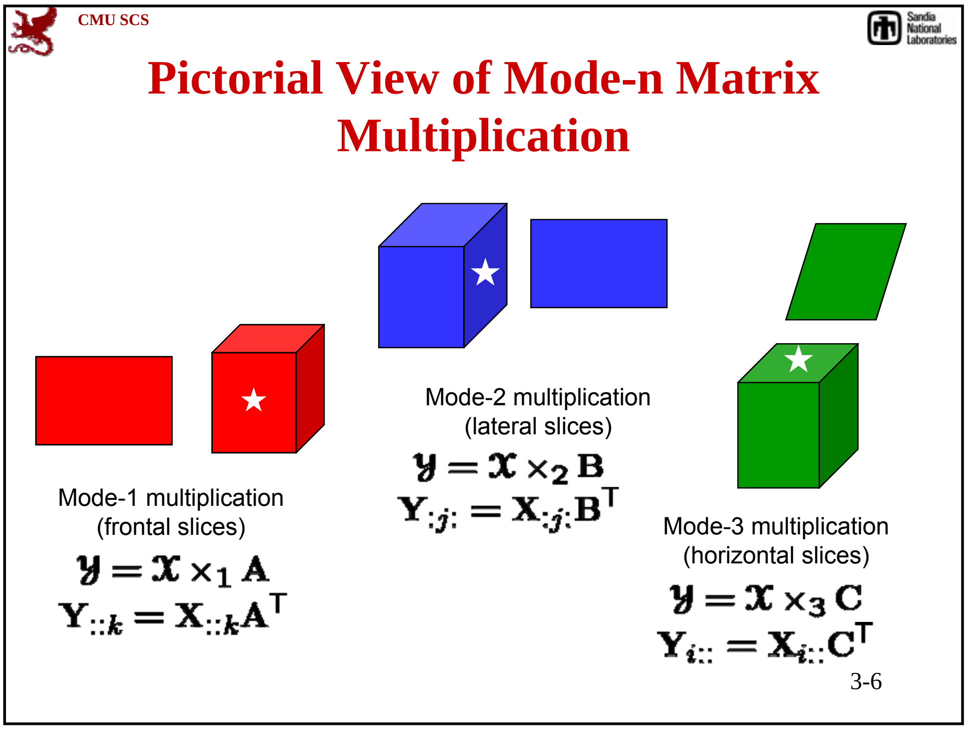
- Tensor Times Vector

$$\mathbf{Y} = \mathcal{X}\, \bar{\times}_1\, \mathbf{a} \in \mathbb{R}^{J\times K}$$

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

Compute the dot product of a and each column (mode-1) fiber

3-5



CMU SCS

Pictorial View of Mode-n Matrix Multiplication

Mode-1 multiplication (frontal slices)

$$\mathcal{Y} = \mathcal{X} \times_1 \mathbf{A}$$

$$\mathbf{Y}_{::k} = \mathbf{X}_{::k}\mathbf{A}^{\mathsf{T}}$$

Mode-2 multiplication (lateral slices)

$$\mathcal{Y} = \mathcal{X} \times_2 \mathbf{B}$$

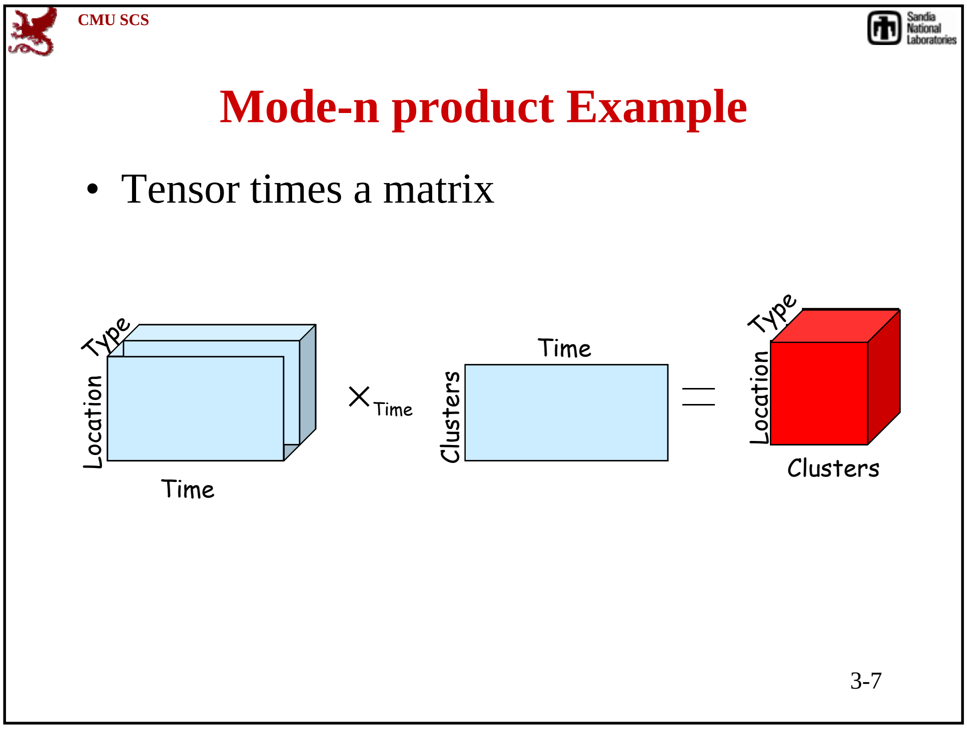
$$\mathbf{Y}_{:j:} = \mathbf{X}_{:j:}\mathbf{B}^{\mathsf{T}}$$

Mode-3 multiplication (horizontal slices)

$$\mathcal{Y} = \mathcal{X} \times_3 \mathbf{C}$$

$$\mathbf{Y}_{i::} = \mathbf{X}_{i::}\mathbf{C}^{\mathsf{T}}$$

3-6

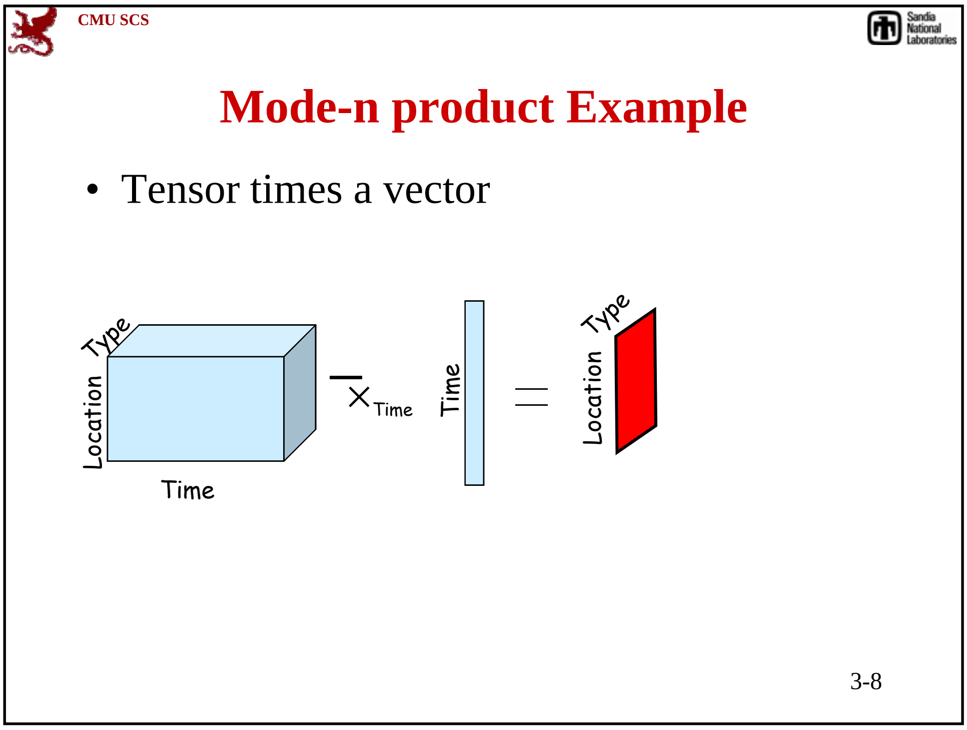


CMU SCS

Mode-n product Example

- Tensor times a matrix

3-7

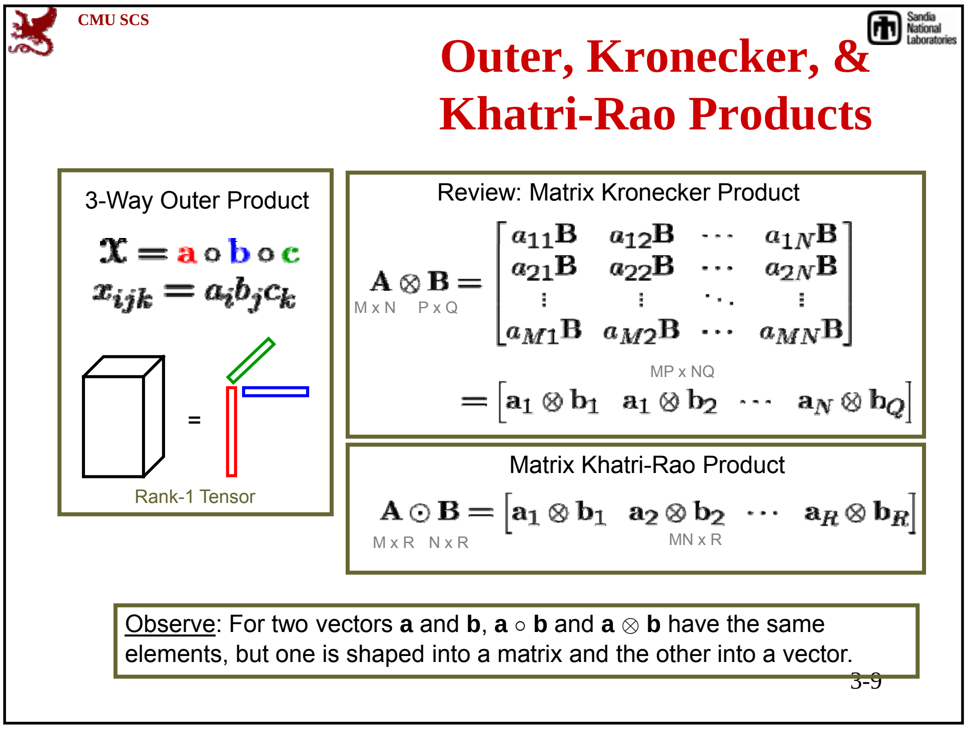


CMU SCS

Mode-n product Example

- Tensor times a vector

3-8



CMU SCS

Outer, Kronecker, & Khatri-Rao 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

Review: Matrix Kronecker Product

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

MP x NQ

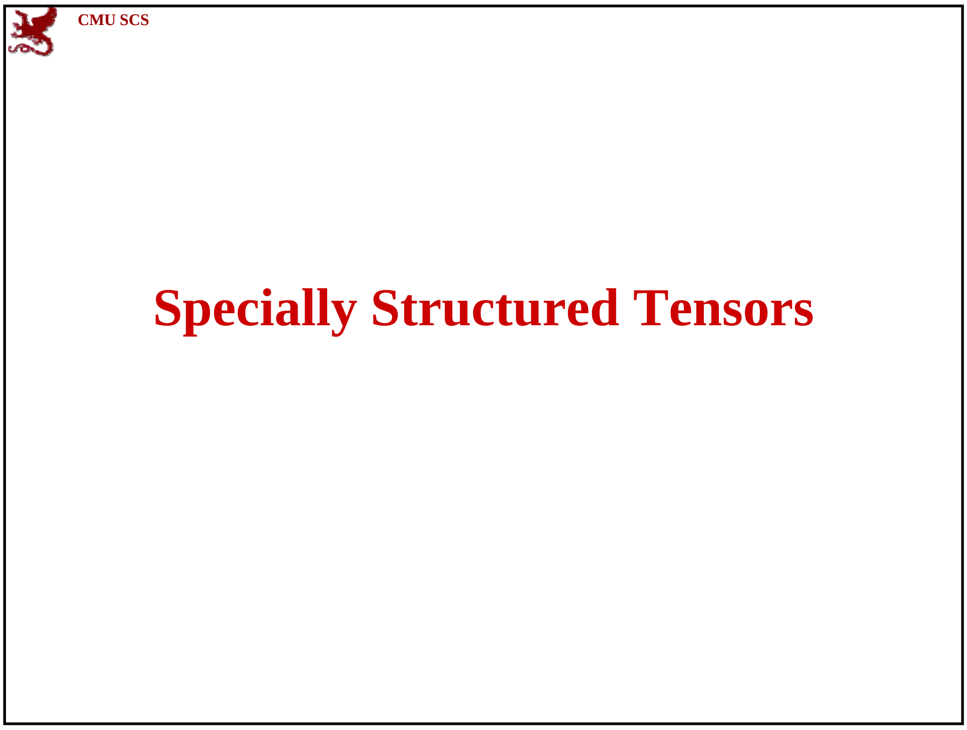
$$= \begin{bmatrix} \mathbf{a}_1 \otimes \mathbf{b}_1 & \mathbf{a}_1 \otimes \mathbf{b}_2 & \cdots & \mathbf{a}_N \otimes \mathbf{b}_Q \end{bmatrix}$$

Matrix Khatri-Rao Product

$$\underset{M \times R}{\mathbf{A}} \odot \underset{N \times R}{\mathbf{B}} = \underset{MN \times R}{\begin{bmatrix} \mathbf{a}_1 \otimes \mathbf{b}_1 & \mathbf{a}_2 \otimes \mathbf{b}_2 & \cdots & \mathbf{a}_R \otimes \mathbf{b}_R \end{bmatrix}}$$

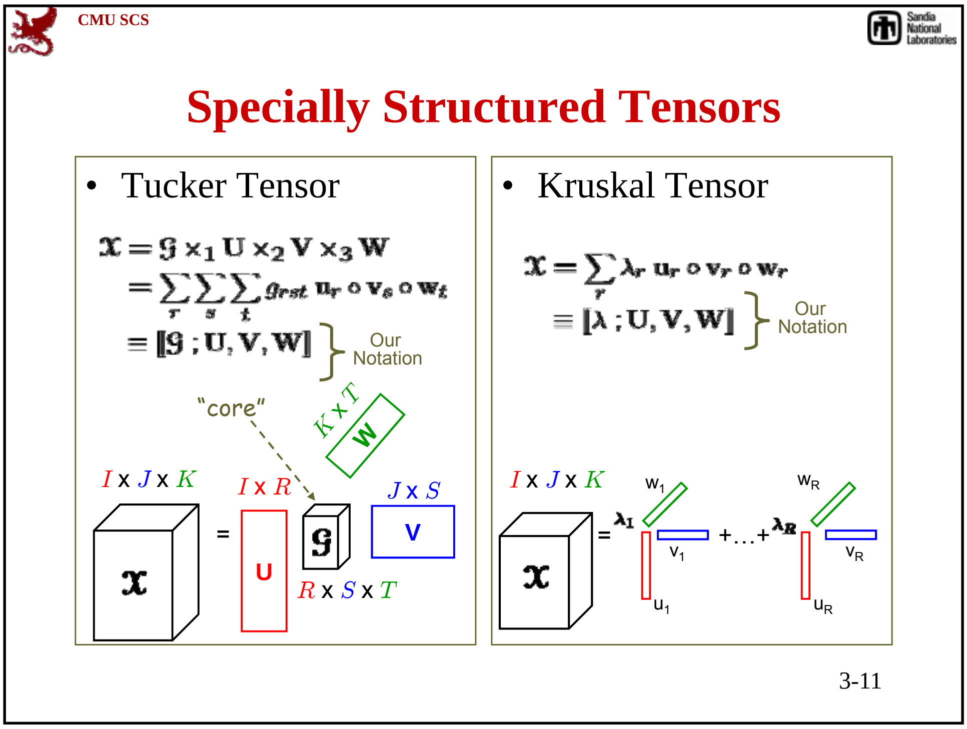
Observe: For two vectors **a** and **b**, **a** ○ **b** and **a** ⊗ **b** have the same elements, but one is shaped into a matrix and the other into a vector.

3-9



CMU SCS

Specially Structured Tensors



CMU SCS

Specially Structured Tensors

- Tucker Tensor

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W}$$
$$= \sum_r \sum_s \sum_t g_{rst}\, \mathbf{u}_r \circ \mathbf{v}_s \circ \mathbf{w}_t$$
$$\equiv [\![\mathcal{G} \,; \mathbf{U}, \mathbf{V}, \mathbf{W}]\!]$$

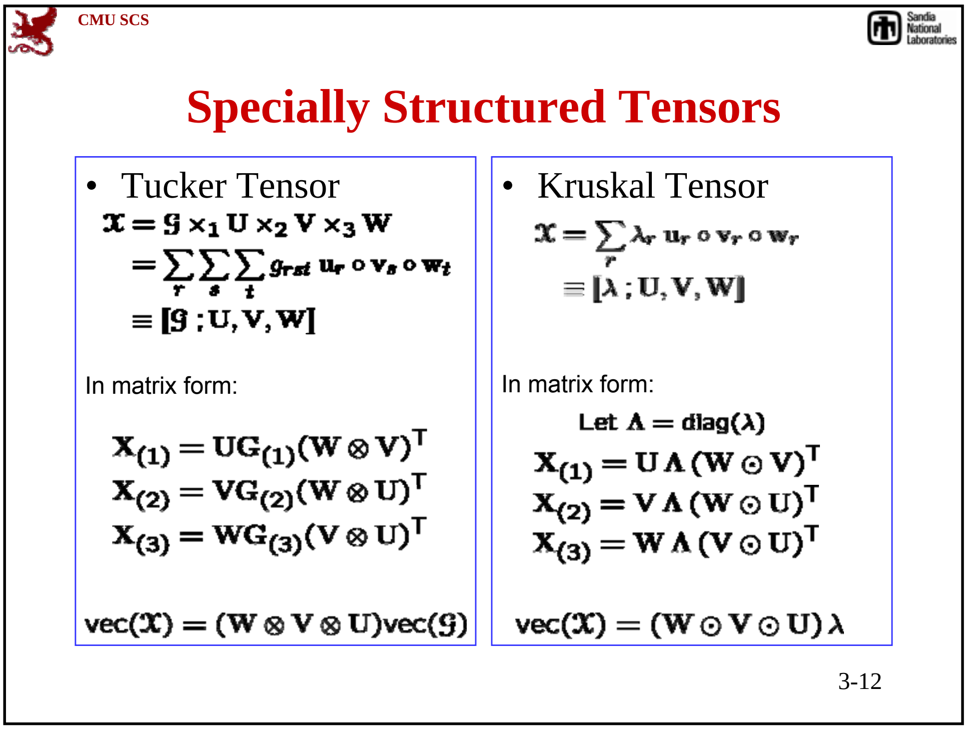
Our Notation

- Kruskal Tensor

$$\mathcal{X} = \sum_r \lambda_r\, \mathbf{u}_r \circ \mathbf{v}_r \circ \mathbf{w}_r$$
$$\equiv [\![\lambda \,; \mathbf{U}, \mathbf{V}, \mathbf{W}]\!]$$

Our Notation

3-11



CMU SCS

Specially Structured Tensors

- Tucker Tensor

$$\mathcal{X} = \mathcal{G} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W}$$
$$= \sum_r \sum_s \sum_t g_{rst}\, \mathbf{u}_r \circ \mathbf{v}_s \circ \mathbf{w}_t$$
$$\equiv [\![\mathcal{G} \,; \mathbf{U}, \mathbf{V}, \mathbf{W}]\!]$$

In matrix form:

$$\mathbf{X}_{(1)} = \mathbf{U}\mathbf{G}_{(1)}(\mathbf{W} \otimes \mathbf{V})^{\mathsf{T}}$$
$$\mathbf{X}_{(2)} = \mathbf{V}\mathbf{G}_{(2)}(\mathbf{W} \otimes \mathbf{U})^{\mathsf{T}}$$
$$\mathbf{X}_{(3)} = \mathbf{W}\mathbf{G}_{(3)}(\mathbf{V} \otimes \mathbf{U})^{\mathsf{T}}$$

$$\text{vec}(\mathcal{X}) = (\mathbf{W} \otimes \mathbf{V} \otimes \mathbf{U})\text{vec}(\mathcal{G})$$

- Kruskal Tensor

$$\mathcal{X} = \sum_r \lambda_r\, \mathbf{u}_r \circ \mathbf{v}_r \circ \mathbf{w}_r$$
$$\equiv [\![\lambda \,; \mathbf{U}, \mathbf{V}, \mathbf{W}]\!]$$

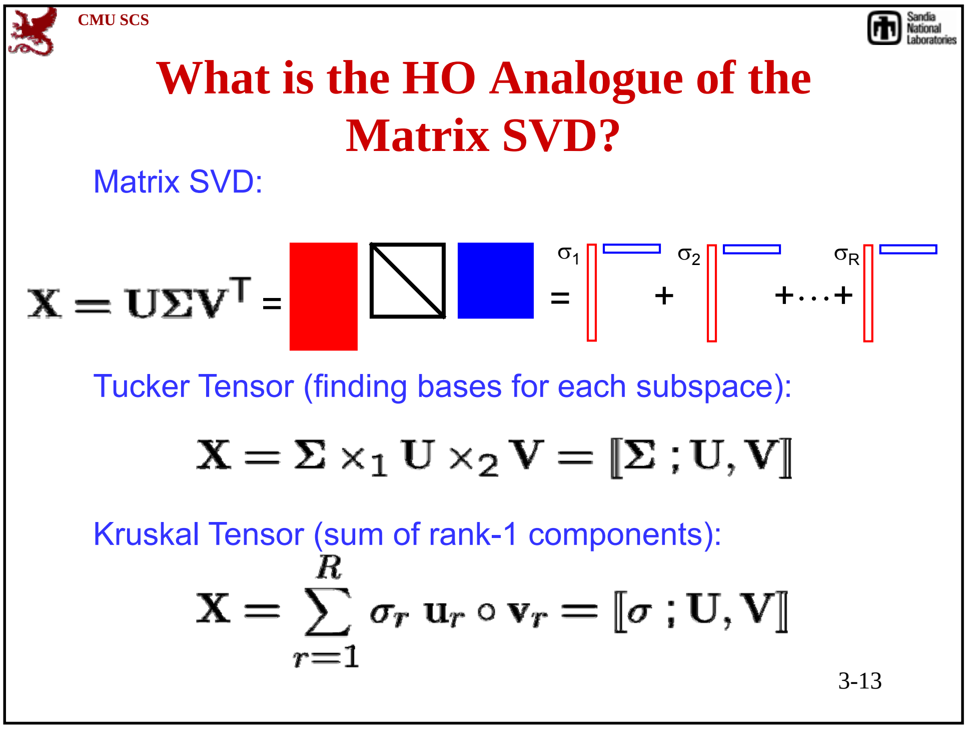
In matrix form:

Let $\mathbf{\Lambda} = \text{diag}(\lambda)$

$$\mathbf{X}_{(1)} = \mathbf{U}\,\mathbf{\Lambda}\,(\mathbf{W} \odot \mathbf{V})^{\mathsf{T}}$$
$$\mathbf{X}_{(2)} = \mathbf{V}\,\mathbf{\Lambda}\,(\mathbf{W} \odot \mathbf{U})^{\mathsf{T}}$$
$$\mathbf{X}_{(3)} = \mathbf{W}\,\mathbf{\Lambda}\,(\mathbf{V} \odot \mathbf{U})^{\mathsf{T}}$$

$$\text{vec}(\mathcal{X}) = (\mathbf{W} \odot \mathbf{V} \odot \mathbf{U})\,\lambda$$

3-12



CMU SCS

What is the HO Analogue of the Matrix SVD?

Matrix SVD:

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

=
σ_1
+
σ_2
+…+
σ_R

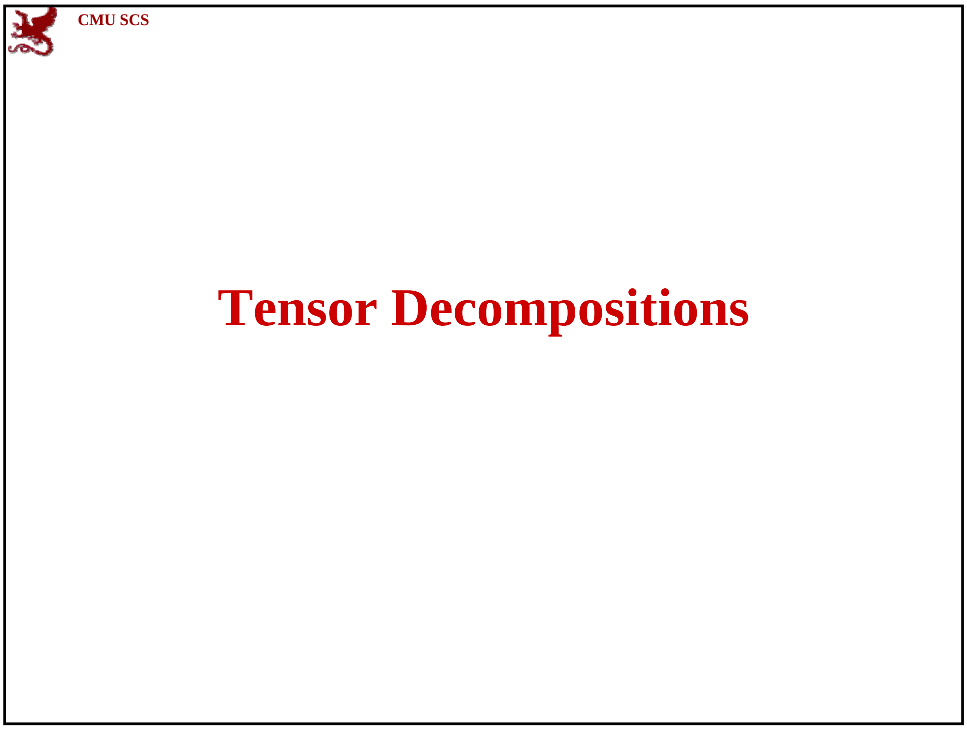
Tucker Tensor (finding bases for each subspace):

$$\mathbf{X} = \mathbf{\Sigma} \times_1 \mathbf{U} \times_2 \mathbf{V} = [\![\mathbf{\Sigma} \, ; \mathbf{U}, \mathbf{V}]\!]$$

Kruskal Tensor (sum of rank-1 components):

$$\mathbf{X} = \sum_{r=1}^{R} \sigma_r \, \mathbf{u}_r \circ \mathbf{v}_r = [\![\sigma \, ; \mathbf{U}, \mathbf{V}]\!]$$

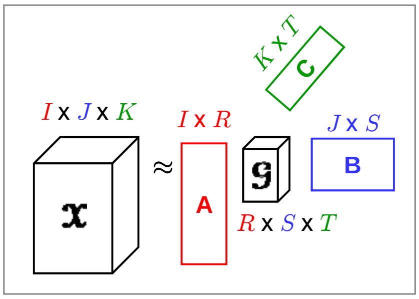
3-13



CMU SCS

Tensor Decompositions

Tucker Decomposition



$$\mathcal{X} \approx [\![\mathcal{G}\,; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$$

Given A, B, C, the optimal core is:

$$\mathcal{G} = [\![\mathcal{X}\,; \mathbf{A}^\dagger, \mathbf{B}^\dagger, \mathbf{C}^\dagger]\!]$$

- Proposed by Tucker (1966)
- AKA: Three-mode factor analysis, three-mode PCA, orthogonal array decomposition
- **A**, **B**, and **C** generally assumed to be orthonormal (generally assume they have full column rank)
- $\mathcal{G}$ is not diagonal
- Not unique

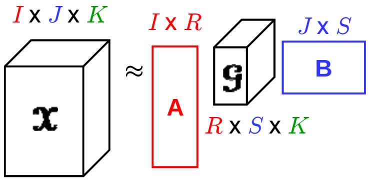
Recall the equations for converting a tensor to a matrix

$$\mathbf{X}_{(1)} = \mathbf{A}\mathbf{G}_{(1)}(\mathbf{C} \otimes \mathbf{B})^{\mathsf{T}}$$
$$\mathbf{X}_{(2)} = \mathbf{B}\mathbf{G}_{(2)}(\mathbf{C} \otimes \mathbf{A})^{\mathsf{T}}$$
$$\mathbf{X}_{(3)} = \mathbf{C}\mathbf{G}_{(3)}(\mathbf{B} \otimes \mathbf{A})^{\mathsf{T}}$$
$$\mathrm{vec}(\mathcal{X}) = (\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A})\mathrm{vec}(\mathcal{G})$$

Tucker Variations

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

- Tucker2

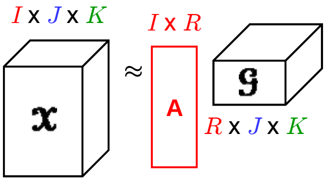


Identity Matrix

$$\mathcal{X} \approx [\![\mathcal{G}\,; \mathbf{A}, \mathbf{B}, \mathbf{I}]\!]$$

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

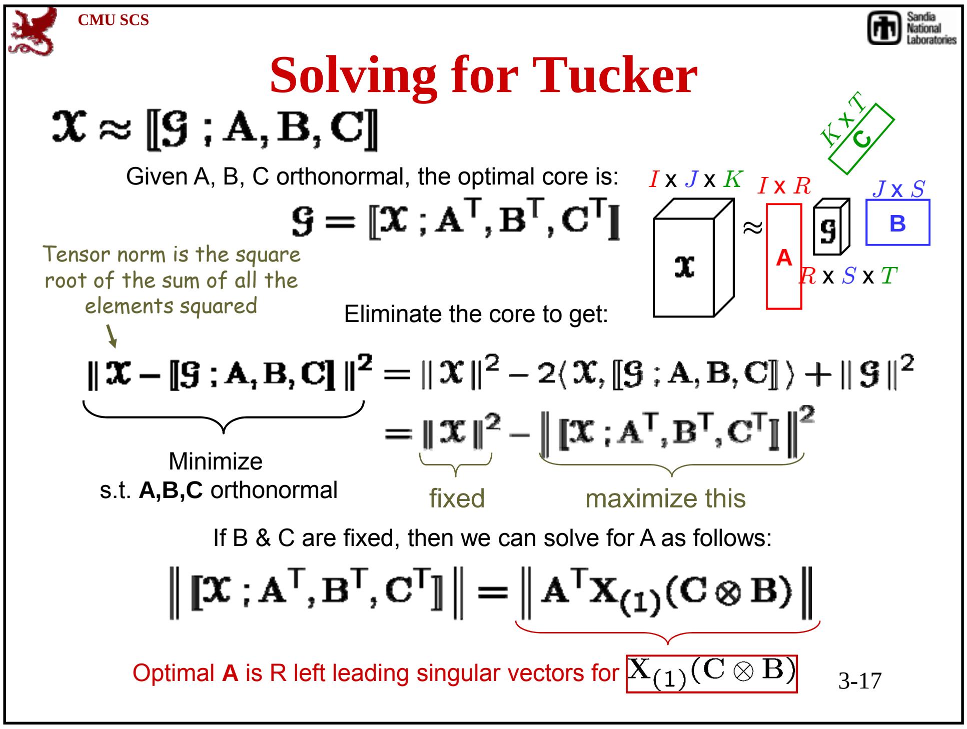
- Tucker1



$$\mathcal{X} \approx [\![\mathcal{G}\,; \mathbf{A}, \mathbf{I}, \mathbf{I}]\!]$$

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

Finding principal components in only mode 1 can be solved via rank-R matrix SVD



CMU SCS

Solving for Tucker

$$\mathcal{X} \approx [\![\mathcal{G} \,;\, \mathbf{A},\mathbf{B},\mathbf{C}]\!]$$

Given A, B, C orthonormal, the optimal core is:

$$\mathcal{G} = [\![\mathcal{X} \,;\, \mathbf{A}^{\mathsf{T}},\mathbf{B}^{\mathsf{T}},\mathbf{C}^{\mathsf{T}}]\!]$$

K x T
C
I x J x K
I x R
J x S
B
$\mathcal{X}$
≈
A
$\mathcal{G}$
R x S x T

Tensor norm is the square root of the sum of all the elements squared

Eliminate the core to get:

$$\underbrace{\|\mathcal{X} - [\![\mathcal{G} \,;\, \mathbf{A},\mathbf{B},\mathbf{C}]\!]\|^2}_{\text{Minimize s.t. } \mathbf{A,B,C} \text{ orthonormal}} = \|\mathcal{X}\|^2 - 2\langle \mathcal{X}, [\![\mathcal{G} \,;\, \mathbf{A},\mathbf{B},\mathbf{C}]\!] \rangle + \|\mathcal{G}\|^2$$

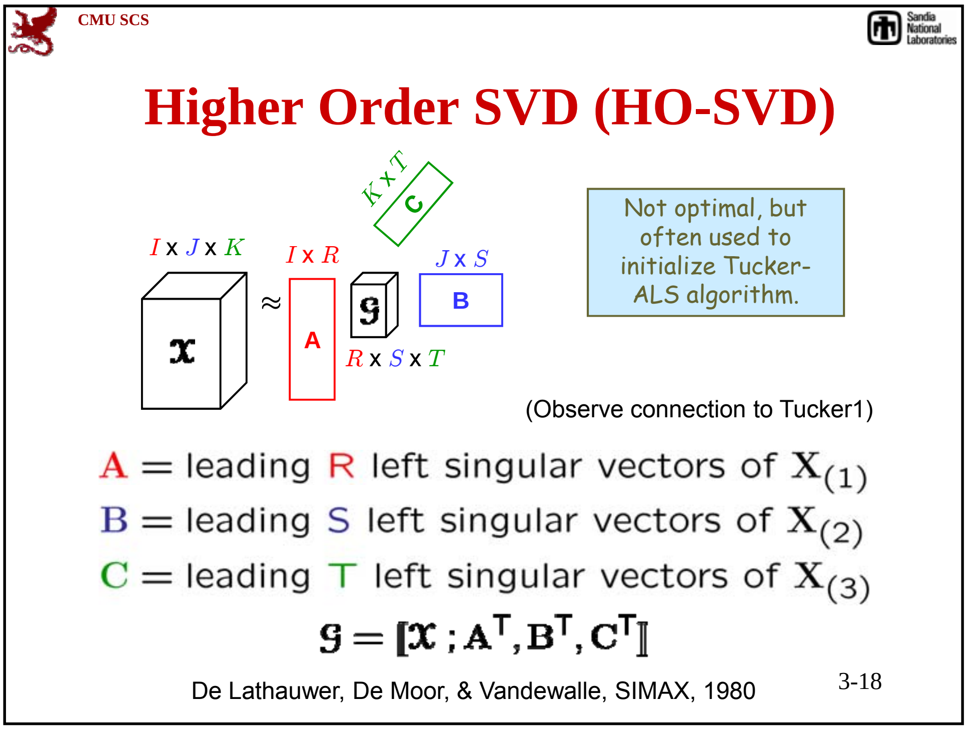
$$= \underbrace{\|\mathcal{X}\|^2}_{\text{fixed}} - \underbrace{\left\| [\![\mathcal{X} \,;\, \mathbf{A}^{\mathsf{T}},\mathbf{B}^{\mathsf{T}},\mathbf{C}^{\mathsf{T}}]\!] \right\|^2}_{\text{maximize this}}$$

If B & C are fixed, then we can solve for A as follows:

$$\left\| [\![\mathcal{X} \,;\, \mathbf{A}^{\mathsf{T}},\mathbf{B}^{\mathsf{T}},\mathbf{C}^{\mathsf{T}}]\!] \right\| = \underbrace{\left\| \mathbf{A}^{\mathsf{T}} \mathbf{X}_{(1)} (\mathbf{C} \otimes \mathbf{B}) \right\|}$$

Optimal **A** is R left leading singular vectors for $\boxed{\mathbf{X}_{(1)}(\mathbf{C} \otimes \mathbf{B})}$

3-17



CMU SCS

Higher Order SVD (HO-SVD)

K x T
C
I x J x K
I x R
J x S
$\mathcal{X}$
≈
A
$\mathcal{G}$
B
R x S x T

Not optimal, but often used to initialize Tucker-ALS algorithm.

(Observe connection to Tucker1)

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

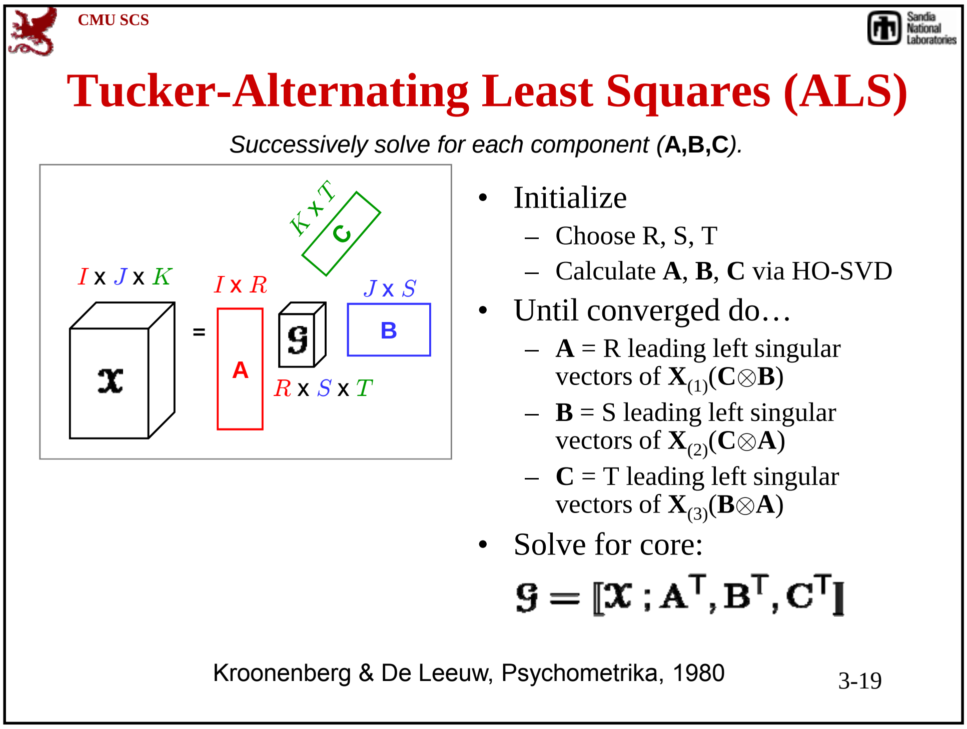
$\mathbf{B}$ = leading S left singular vectors of $\mathbf{X}_{(2)}$

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

$$\mathcal{G} = [\![\mathcal{X} \,;\, \mathbf{A}^{\mathsf{T}},\mathbf{B}^{\mathsf{T}},\mathbf{C}^{\mathsf{T}}]\!]$$

De Lathauwer, De Moor, & Vandewalle, SIMAX, 1980

3-18


I x J x K
=
𝒳
I x R
A
K x T
C
𝒢
J x S
B
R x S x T

CMU SCS

Tucker-Alternating Least Squares (ALS)

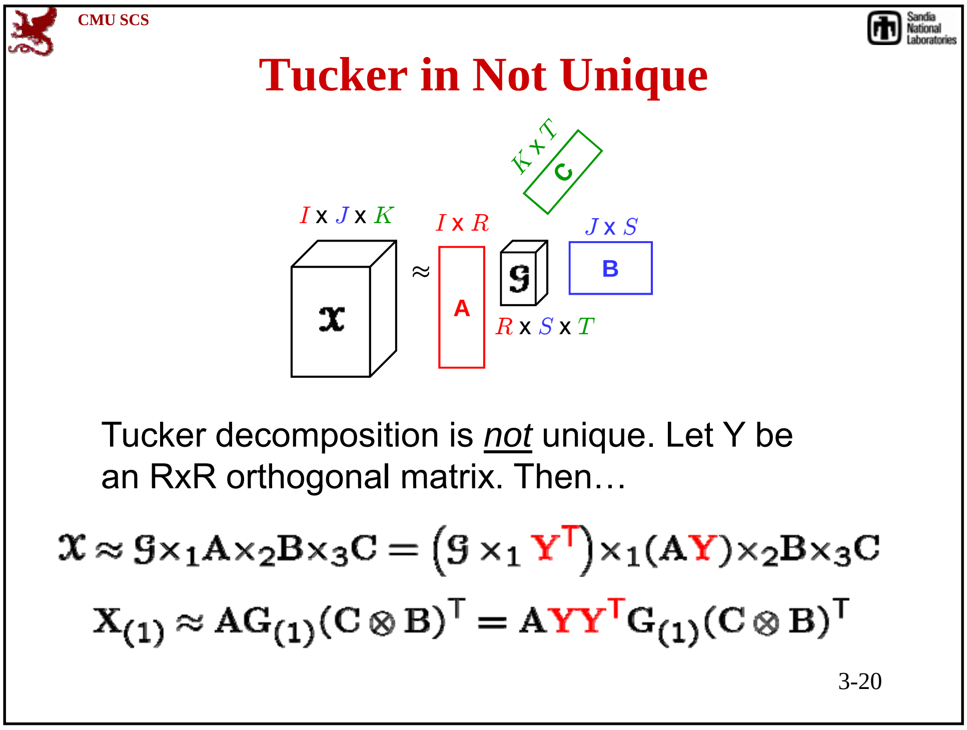
Successively solve for each component (**A,B,C**)*.*

- Initialize
  - Choose R, S, T
  - Calculate **A**, **B**, **C** via HO-SVD
- Until converged do…
  - **A** = R leading left singular vectors of $\mathbf{X}_{(1)}(\mathbf{C}\otimes\mathbf{B})$
  - **B** = S leading left singular vectors of $\mathbf{X}_{(2)}(\mathbf{C}\otimes\mathbf{A})$
  - **C** = T leading left singular vectors of $\mathbf{X}_{(3)}(\mathbf{B}\otimes\mathbf{A})$
- Solve for core:

$$\mathcal{G} = [\![\mathcal{X} ; \mathbf{A}^\mathsf{T}, \mathbf{B}^\mathsf{T}, \mathbf{C}^\mathsf{T}]\!]$$

Kroonenberg & De Leeuw, Psychometrika, 1980

3-19


I x J x K
≈
𝒳
I x R
A
K x T
C
𝒢
J x S
B
R x S x T

CMU SCS

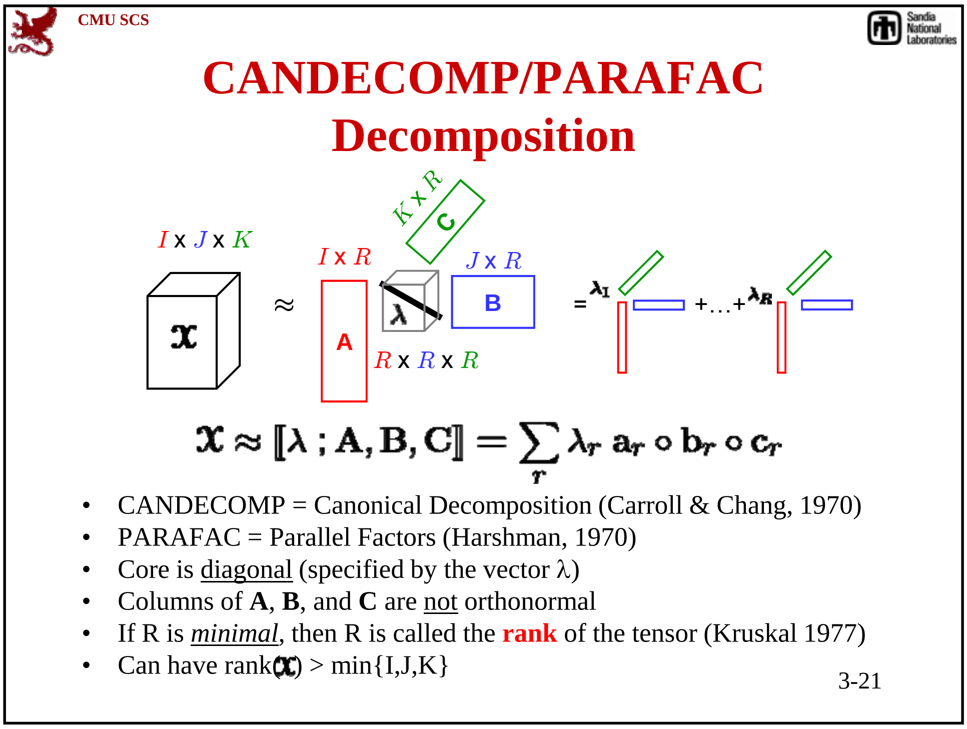
Tucker in Not Unique

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

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

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

3-20



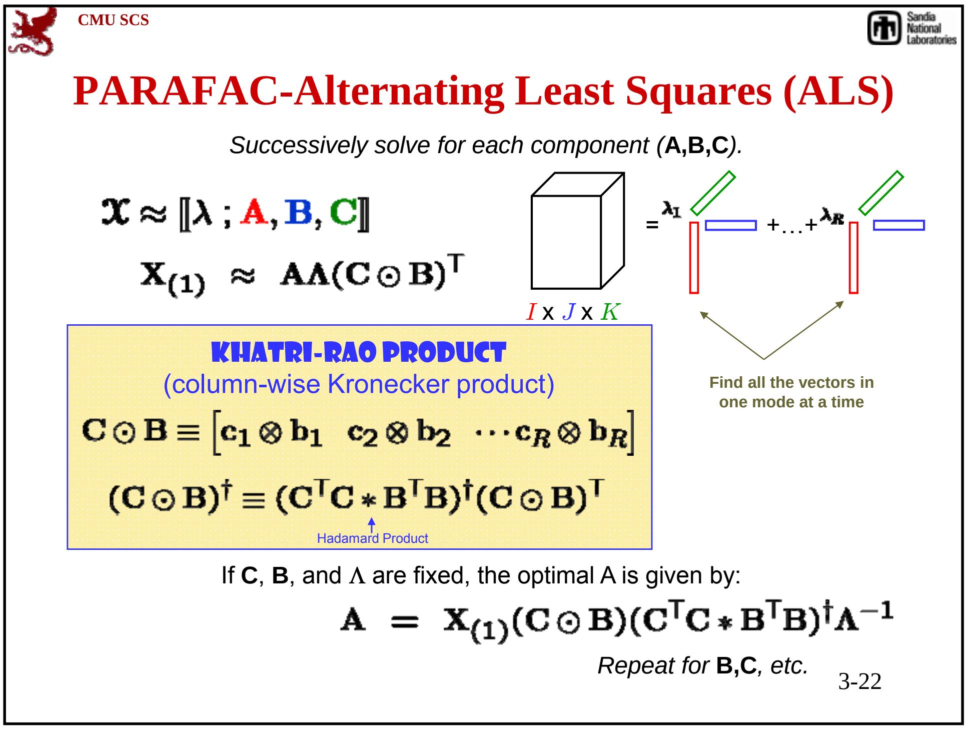
CMU SCS

CANDECOMP/PARAFAC Decomposition

$$\mathcal{X} \approx [\![\lambda \,;\, \mathbf{A},\mathbf{B},\mathbf{C}]\!] = \sum_r \lambda_r \, \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

- CANDECOMP = Canonical Decomposition (Carroll & Chang, 1970)
- PARAFAC = Parallel Factors (Harshman, 1970)
- Core is diagonal (specified by the vector λ)
- Columns of **A**, **B**, and **C** are not orthonormal
- If R is *minimal*, then R is called the **rank** of the tensor (Kruskal 1977)
- Can have rank($\mathcal{X}$) > min{I,J,K}

3-21



CMU SCS

PARAFAC-Alternating Least Squares (ALS)

*Successively solve for each component (***A,B,C***).*

$$\mathcal{X} \approx [\![\lambda \,;\, \mathbf{A},\mathbf{B},\mathbf{C}]\!]$$

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

Find all the vectors in one mode at a time

KHATRI-RAO PRODUCT

(column-wise Kronecker product)

$$\mathbf{C}\odot\mathbf{B} \equiv \left[\mathbf{c}_1\otimes\mathbf{b}_1 \;\; \mathbf{c}_2\otimes\mathbf{b}_2 \;\; \cdots \mathbf{c}_R\otimes\mathbf{b}_R\right]$$

$$(\mathbf{C}\odot\mathbf{B})^{\dagger} \equiv (\mathbf{C}^{\mathsf{T}}\mathbf{C} \underset{\uparrow}{*} \mathbf{B}^{\mathsf{T}}\mathbf{B})^{\dagger}(\mathbf{C}\odot\mathbf{B})^{\mathsf{T}}$$

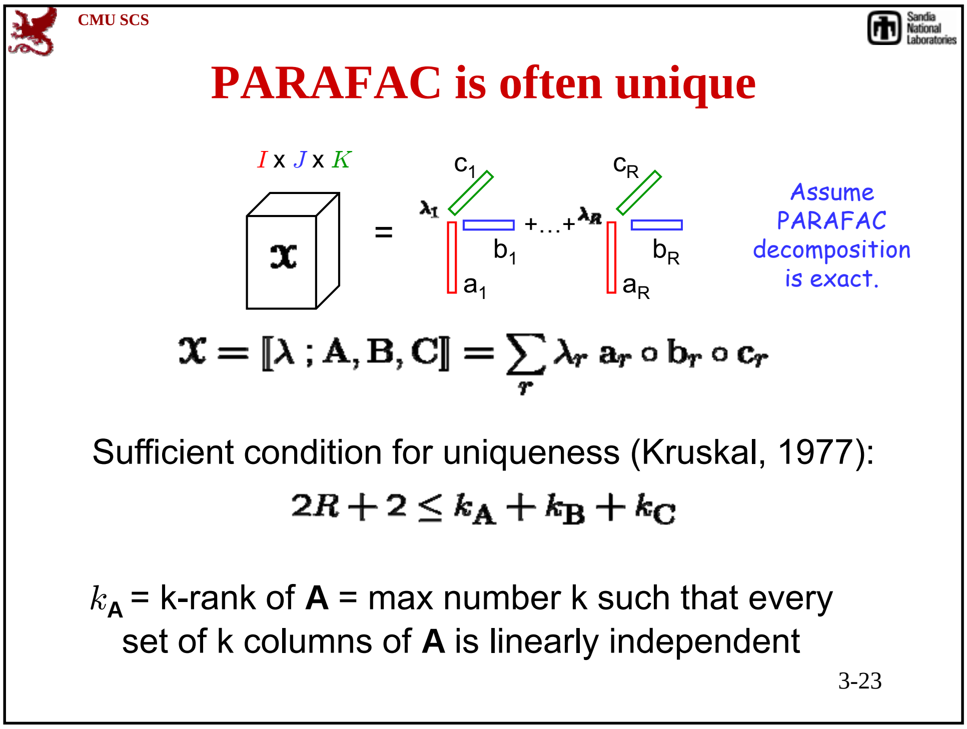
Hadamard Product

If **C**, **B**, and Λ are fixed, the optimal A is given by:

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

Repeat for **B,C**, *etc.*

3-22



CMU SCS

PARAFAC is often unique

Assume PARAFAC decomposition is exact.

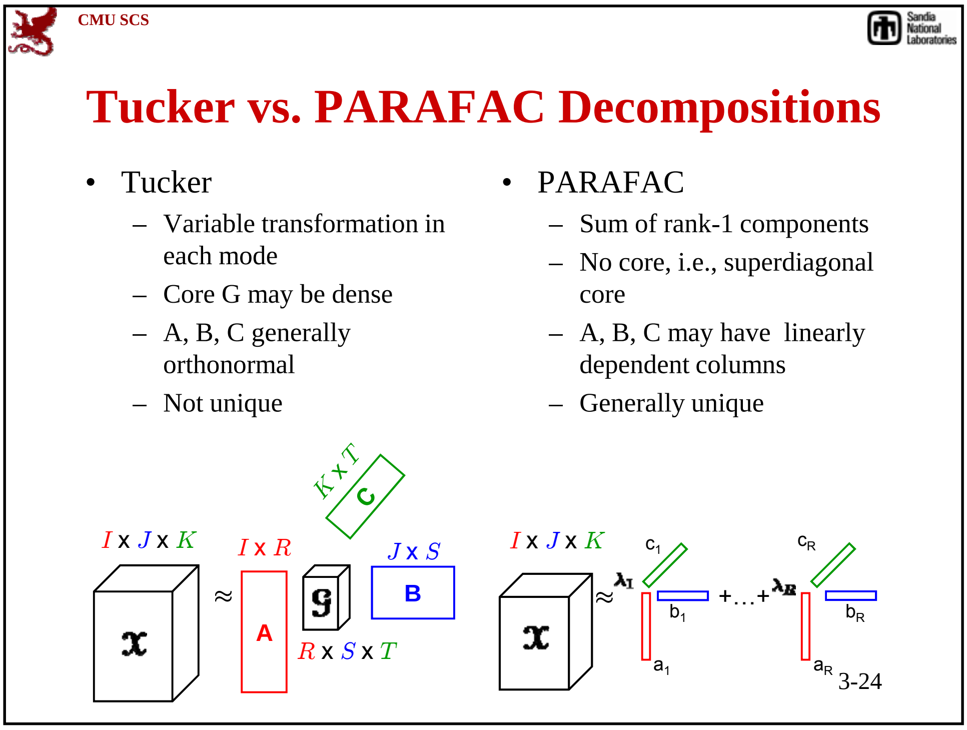
$$\mathcal{X} = [\![\lambda\, ; \mathbf{A},\mathbf{B},\mathbf{C}]\!] = \sum_r \lambda_r \, \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

Sufficient condition for uniqueness (Kruskal, 1977):

$$2R + 2 \leq k_{\mathbf{A}} + k_{\mathbf{B}} + k_{\mathbf{C}}$$

$k_{\mathbf{A}}$ = k-rank of **A** = max number k such that every set of k columns of **A** is linearly independent

3-23

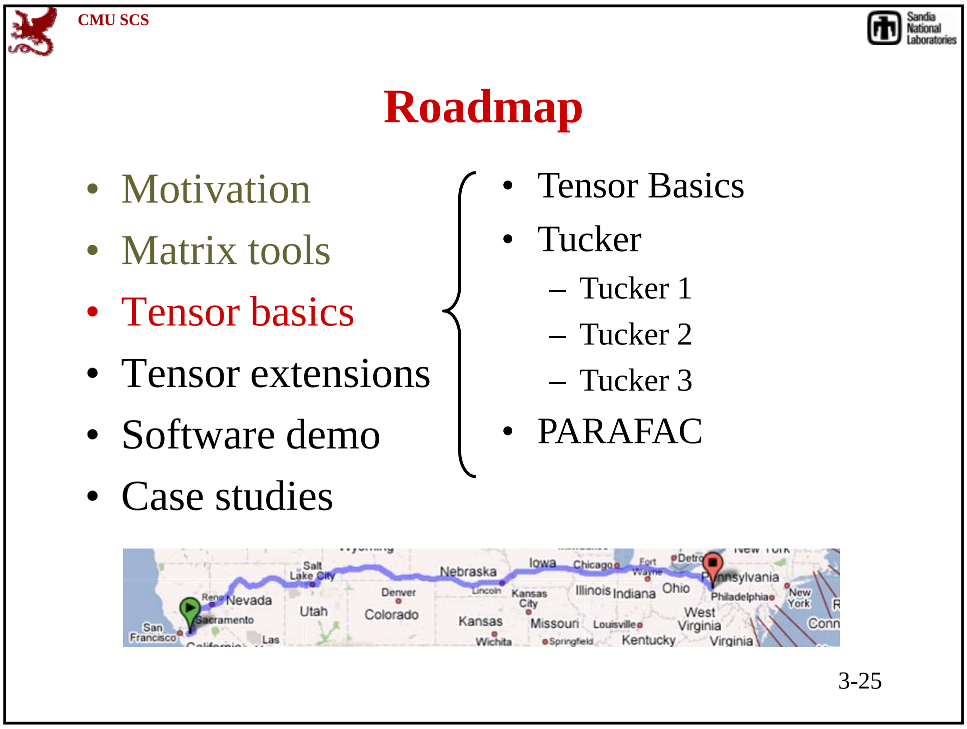


CMU SCS

Tucker vs. PARAFAC Decompositions

- Tucker
  - Variable transformation in each mode
  - Core G may be dense
  - A, B, C generally orthonormal
  - Not unique
- PARAFAC
  - Sum of rank-1 components
  - No core, i.e., superdiagonal core
  - A, B, C may have linearly dependent columns
  - Generally unique

3-24



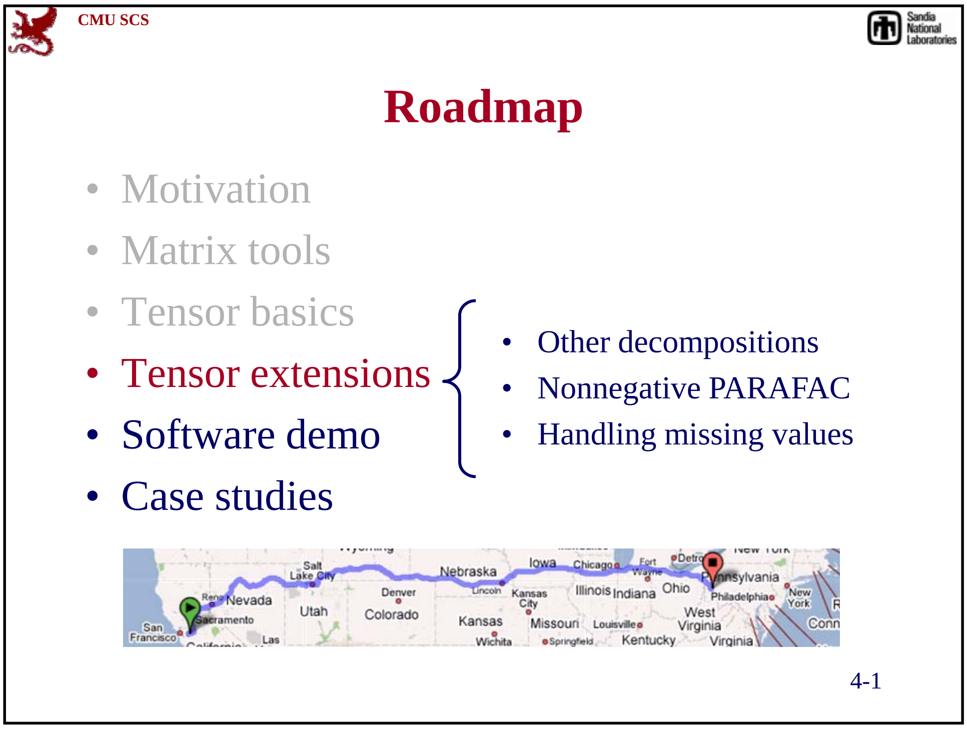
CMU SCS

Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

- Tensor Basics
- Tucker
  - Tucker 1
  - Tucker 2
  - Tucker 3
- PARAFAC

3-25

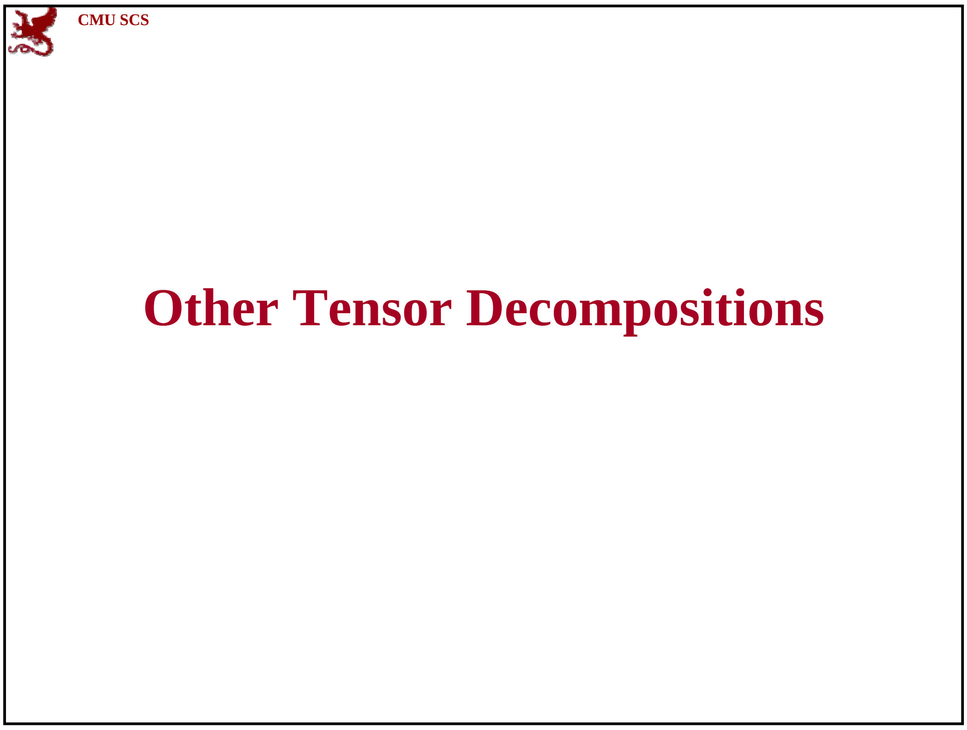


CMU SCS

Roadmap

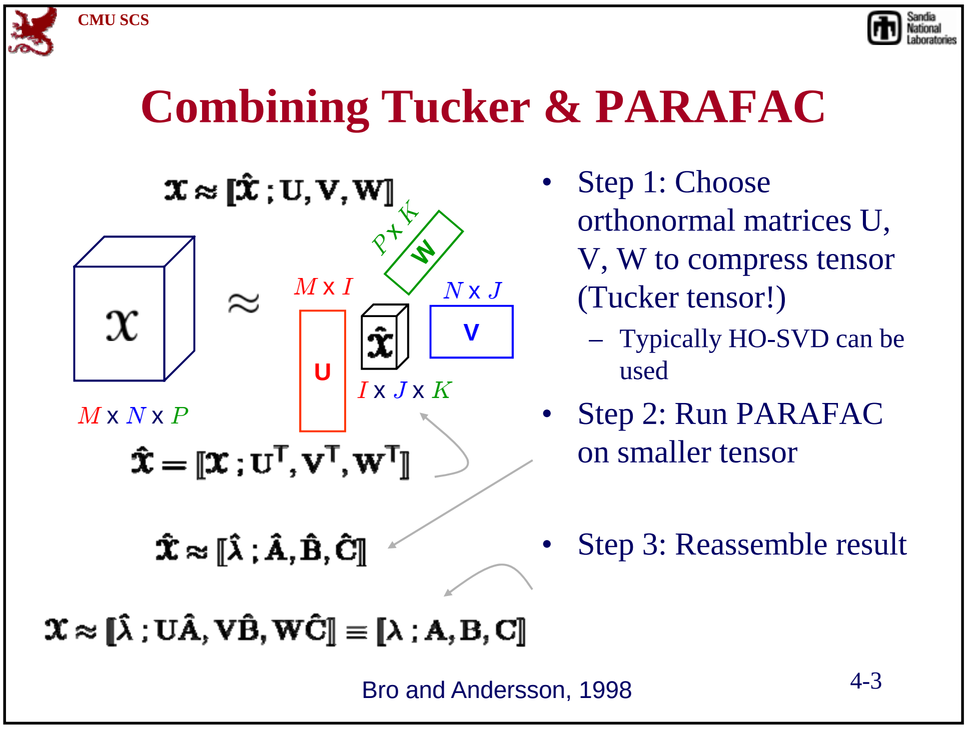
- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
  - Other decompositions
  - Nonnegative PARAFAC
  - Handling missing values
- Software demo
- Case studies

4-1



CMU SCS

Other Tensor Decompositions



CMU SCS

Combining Tucker & PARAFAC

$$\mathcal{X} \approx [\![\hat{\mathcal{X}} \, ; \mathbf{U}, \mathbf{V}, \mathbf{W}]\!]$$

$$\hat{\mathcal{X}} = [\![\mathcal{X} \, ; \mathbf{U}^\mathsf{T}, \mathbf{V}^\mathsf{T}, \mathbf{W}^\mathsf{T}]\!]$$

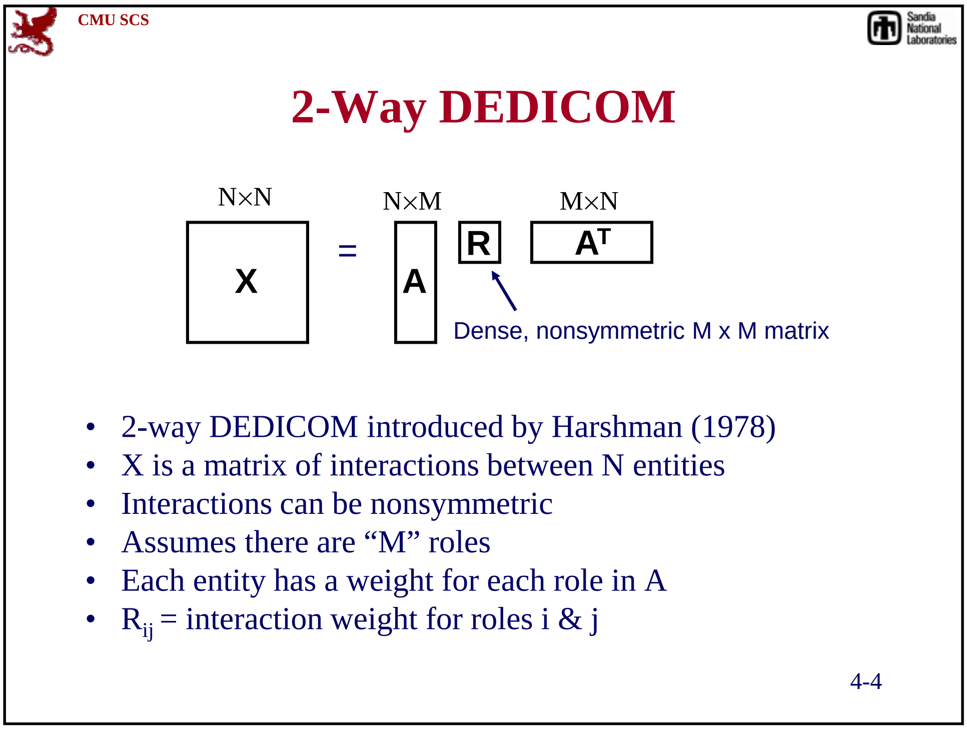
$$\hat{\mathcal{X}} \approx [\![\hat{\lambda} \, ; \hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}]\!]$$

$$\mathcal{X} \approx [\![\hat{\lambda} \, ; \mathbf{U}\hat{\mathbf{A}}, \mathbf{V}\hat{\mathbf{B}}, \mathbf{W}\hat{\mathbf{C}}]\!] \equiv [\![\lambda \, ; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$$

- Step 1: Choose orthonormal matrices U, V, W to compress tensor (Tucker tensor!)
  - Typically HO-SVD can be used
- Step 2: Run PARAFAC on smaller tensor
- Step 3: Reassemble result

Bro and Andersson, 1998

4-3



CMU SCS

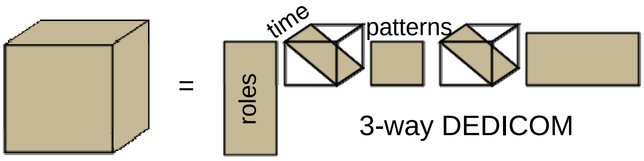
2-Way DEDICOM

Dense, nonsymmetric M x M matrix

- 2-way DEDICOM introduced by Harshman (1978)
- X is a matrix of interactions between N entities
- Interactions can be nonsymmetric
- Assumes there are “M” roles
- Each entity has a weight for each role in A
- R_{ij} = interaction weight for roles i & j

4-4

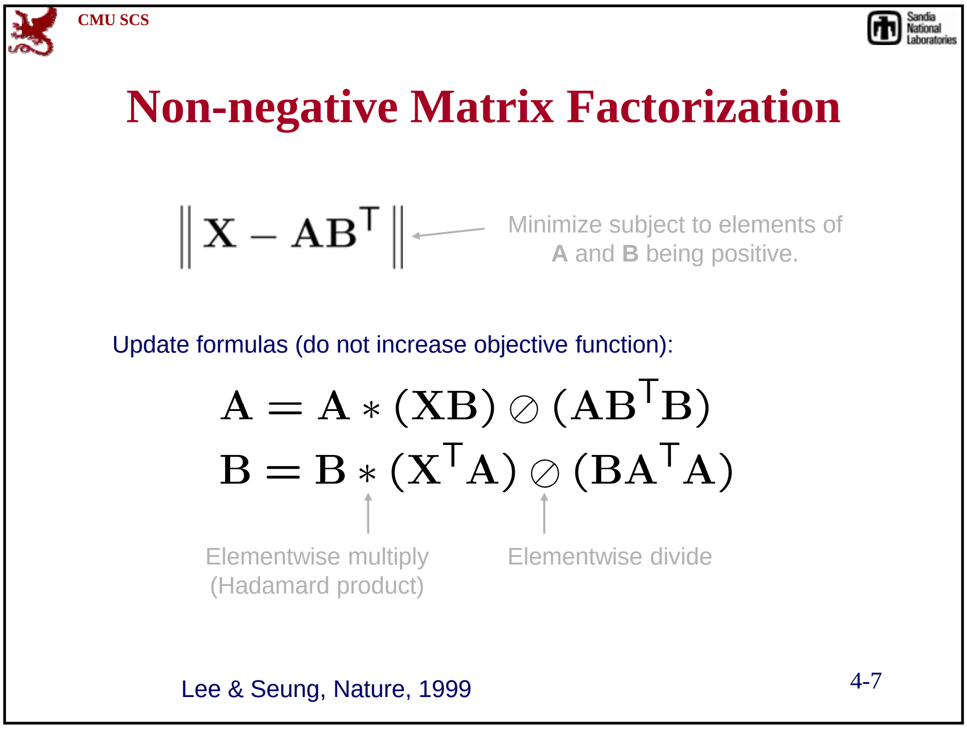
3-Way DEDICOM



$$\mathbf{X}_{::k} = \mathbf{A}\,\mathbf{D}_{::k}\,\mathbf{R}\,\mathbf{D}_{::k}\,\mathbf{A}^{\mathsf{T}}$$

- 3-way DEDICOM due to Kiers (1993)
- Once again, X captures interactions among entities
- Third dimension can correspond to time
- Diagonal slices capture participation of each role at each time
- See Bader et al., SAND2006-7744 , for application to Enron email data

Nonnegativity



CMU SCS

Non-negative Matrix Factorization

$$\left\| \mathbf{X} - \mathbf{A}\mathbf{B}^{\mathsf{T}} \right\|$$

Minimize subject to elements of **A** and **B** being positive.

Update formulas (do not increase objective function):

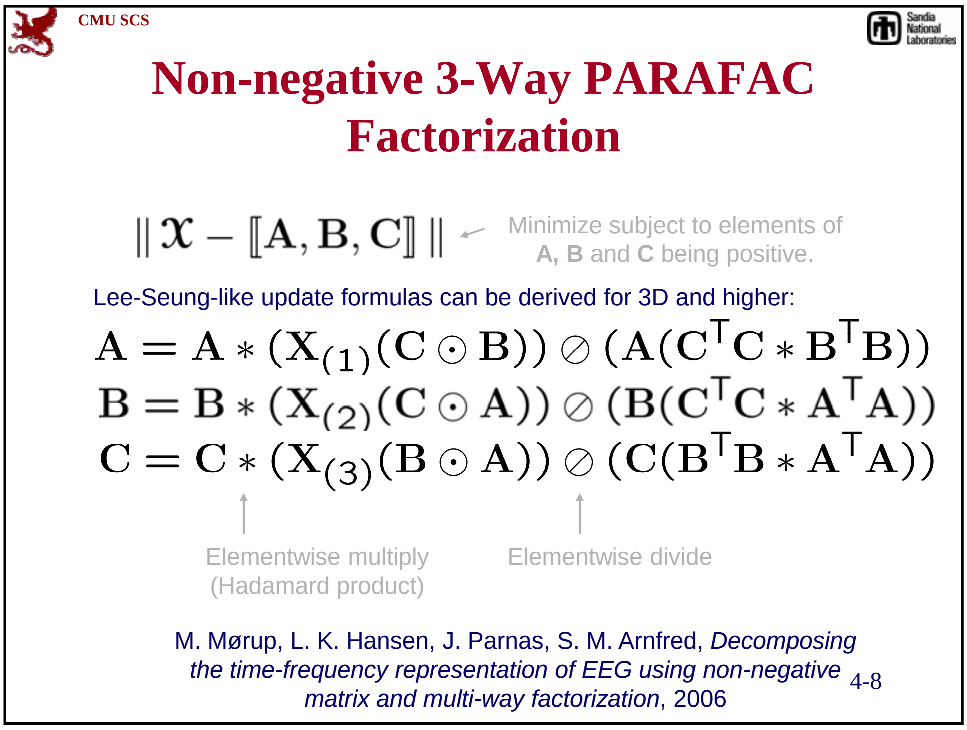
$$\mathbf{A} = \mathbf{A} * (\mathbf{X}\mathbf{B}) \oslash (\mathbf{A}\mathbf{B}^{\mathsf{T}}\mathbf{B})$$
$$\mathbf{B} = \mathbf{B} * (\mathbf{X}^{\mathsf{T}}\mathbf{A}) \oslash (\mathbf{B}\mathbf{A}^{\mathsf{T}}\mathbf{A})$$

Elementwise multiply (Hadamard product)

Elementwise divide

Lee & Seung, Nature, 1999

4-7



CMU SCS

Non-negative 3-Way PARAFAC Factorization

$$\| \mathfrak{X} - [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \|$$

Minimize subject to elements of **A, B** and **C** being positive.

Lee-Seung-like update formulas can be derived for 3D and higher:

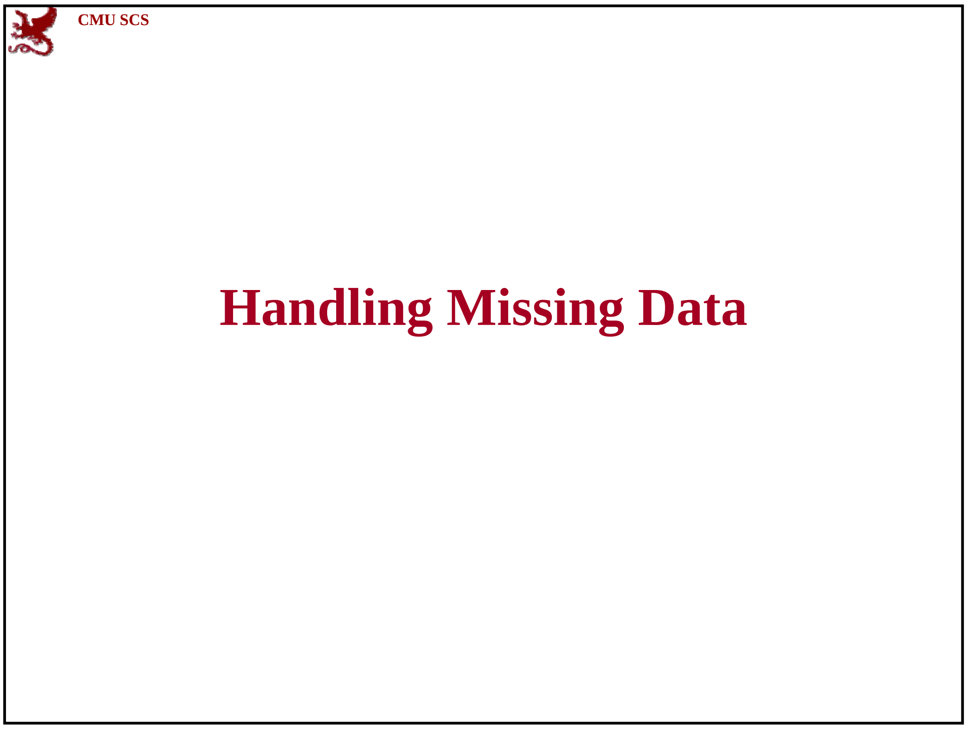
$$\mathbf{A} = \mathbf{A} * (\mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B})) \oslash (\mathbf{A}(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{B}^{\mathsf{T}}\mathbf{B}))$$
$$\mathbf{B} = \mathbf{B} * (\mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A})) \oslash (\mathbf{B}(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{A}^{\mathsf{T}}\mathbf{A}))$$
$$\mathbf{C} = \mathbf{C} * (\mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A})) \oslash (\mathbf{C}(\mathbf{B}^{\mathsf{T}}\mathbf{B} * \mathbf{A}^{\mathsf{T}}\mathbf{A}))$$

Elementwise multiply (Hadamard product)

Elementwise divide

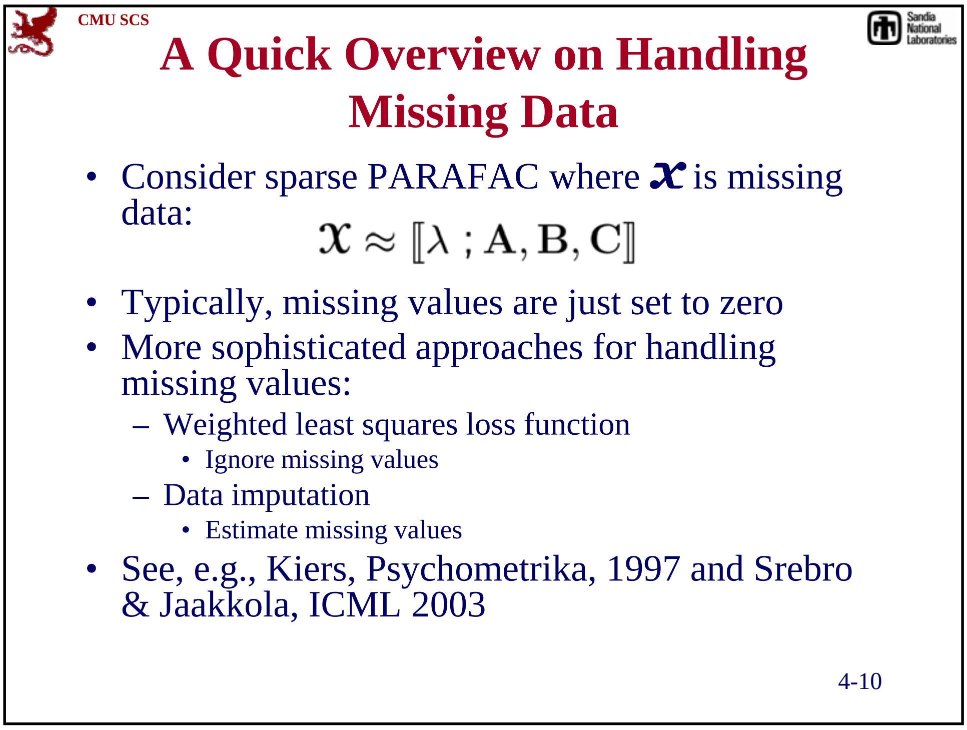
M. Mørup, L. K. Hansen, J. Parnas, S. M. Arnfred, *Decomposing the time-frequency representation of EEG using non-negative matrix and multi-way factorization*, 2006

4-8



CMU SCS

Handling Missing Data



CMU SCS

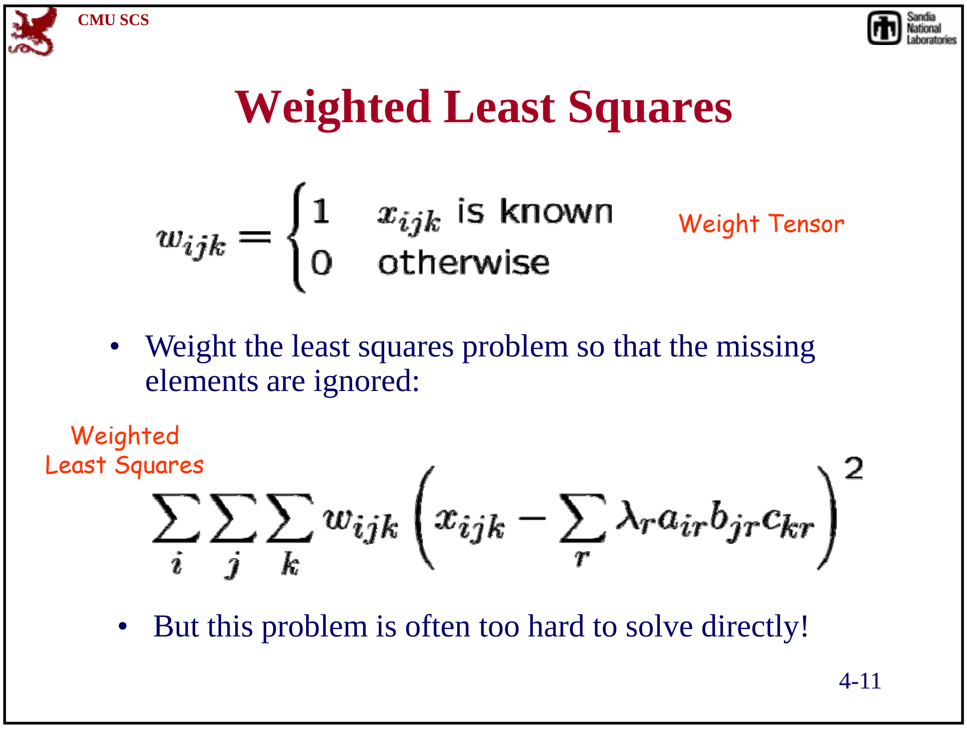
A Quick Overview on Handling Missing Data

- Consider sparse PARAFAC where $\boldsymbol{\mathcal{X}}$ is missing data:

$$\mathcal{X} \approx [\![\lambda\,;\mathbf{A},\mathbf{B},\mathbf{C}]\!]$$

- Typically, missing values are just set to zero
- More sophisticated approaches for handling missing values:
  - Weighted least squares loss function
    - Ignore missing values
  - Data imputation
    - Estimate missing values
- See, e.g., Kiers, Psychometrika, 1997 and Srebro & Jaakkola, ICML 2003

4-10



CMU SCS

Weighted Least Squares

$$w_{ijk} = \begin{cases} 1 & x_{ijk} \text{ is known} \\ 0 & \text{otherwise} \end{cases}$$

Weight Tensor

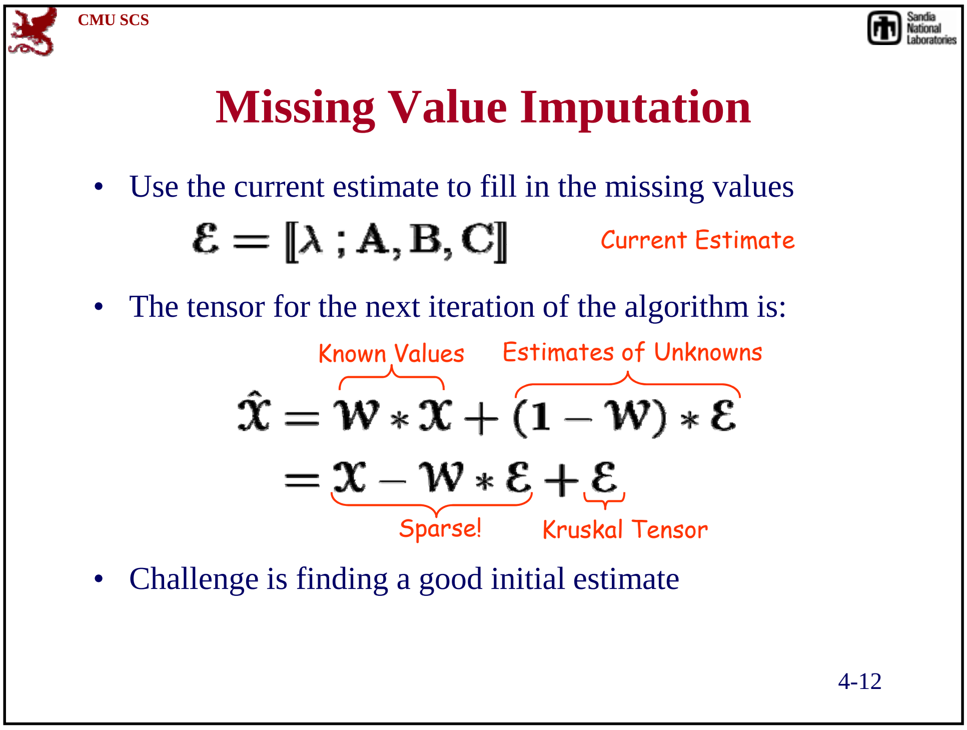
- Weight the least squares problem so that the missing elements are ignored:

Weighted Least Squares

$$\sum_i \sum_j \sum_k w_{ijk} \left(x_{ijk} - \sum_r \lambda_r a_{ir} b_{jr} c_{kr} \right)^2$$

- But this problem is often too hard to solve directly!

4-11



CMU SCS

Missing Value Imputation

- Use the current estimate to fill in the missing values

$$\mathcal{E} = [\![\lambda \,; \mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$$

Current Estimate

- The tensor for the next iteration of the algorithm is:

$$\begin{aligned} \hat{\mathcal{X}} &= \overbrace{\mathcal{W} * \mathcal{X}}^{\text{Known Values}} + \overbrace{(\mathbf{1} - \mathcal{W}) * \mathcal{E}}^{\text{Estimates of Unknowns}} \\ &= \underbrace{\mathcal{X} - \mathcal{W} * \mathcal{E}}_{\text{Sparse!}} + \underbrace{\mathcal{E}}_{\text{Kruskal Tensor}} \end{aligned}$$

- Challenge is finding a good initial estimate

4-12

CMU SCS

Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

4-13

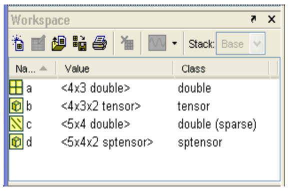
CMU SCS

Computations with Tensors

Tensor Toolbox for MATLAB

http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox

- Six object-oriented tensor classes
  - Working with tensors is easy
- Most comprehensive set of kernel operations in any language
  - E.g., arithmetic, logical, multiplication operations
- Sparse tensors are unique
  - Speed-ups of two orders of magnitude for smaller problems
  - Larger problems than ever before



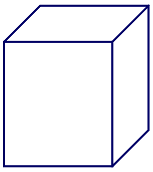
Na...	Value	Class
a	<4x3 double>	double
b	<4x3x2 tensor>	tensor
c	<5x4 double>	double (sparse)
d	<5x4x2 sptensor>	sptensor

- Free for research or evaluations purposes
- 297 unique registered users from all over the world (as of January 17, 2006)

Bader & Kolda, ACM TOMS 2006 & SAND2006-7592

Dense Tensors

- Largest tensor that can be stored on a laptop is 200 x 200 x 200
- Typically, tensor operations are reduced to matrix operations
  - Requires permuting and reshaping the tensor
- Example: Mode-n tensor-matrix multiply

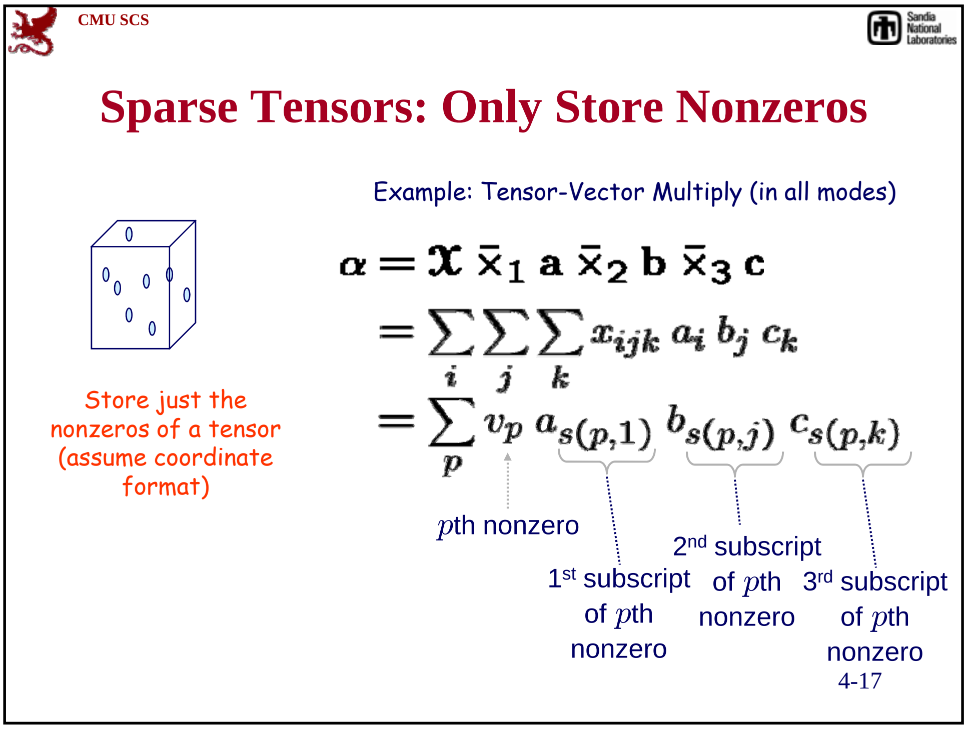


I x *J* x *K*

Example: Mode-1 Matrix Multiply

$$\underset{M \times J \times K}{\mathcal{Y}} = \underset{I \times J \times K}{\mathcal{X}} \times_1 \underset{M \times I}{\mathbf{U}}$$

$$\underset{M \times JK}{\mathbf{Y}_{(n)}} = \underset{M \times I}{\mathbf{U}}\, \underset{I \times JK}{\mathbf{X}_{(n)}}$$



CMU SCS

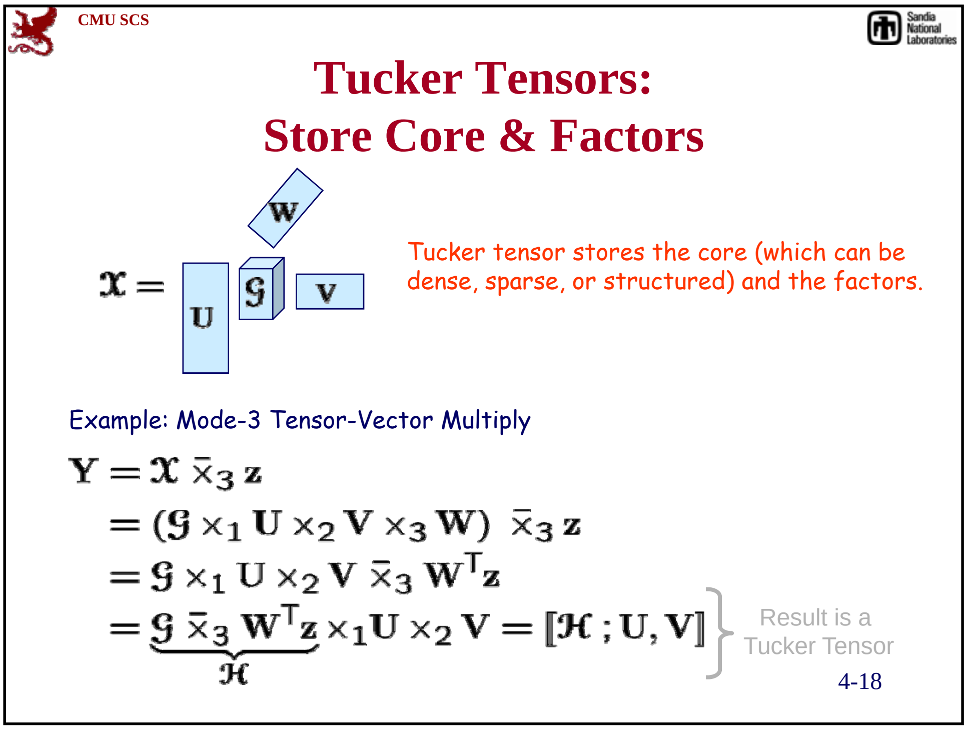
Sparse Tensors: Only Store Nonzeros

Example: Tensor-Vector Multiply (in all modes)

$$\begin{aligned}\alpha &= \mathfrak{X} \,\bar{\times}_1\, \mathbf{a} \,\bar{\times}_2\, \mathbf{b} \,\bar{\times}_3\, \mathbf{c} \\ &= \sum_i \sum_j \sum_k x_{ijk}\, a_i\, b_j\, c_k \\ &= \sum_p \underset{\uparrow\ p\text{th nonzero}}{v_p}\, \underbrace{a_{s(p,1)}}_{\text{1st subscript of } p\text{th nonzero}}\, \underbrace{b_{s(p,j)}}_{\text{2nd subscript of } p\text{th nonzero}}\, \underbrace{c_{s(p,k)}}_{\text{3rd subscript of } p\text{th nonzero}}\end{aligned}$$

Store just the nonzeros of a tensor (assume coordinate format)

4-17



CMU SCS

Tucker Tensors: Store Core & Factors

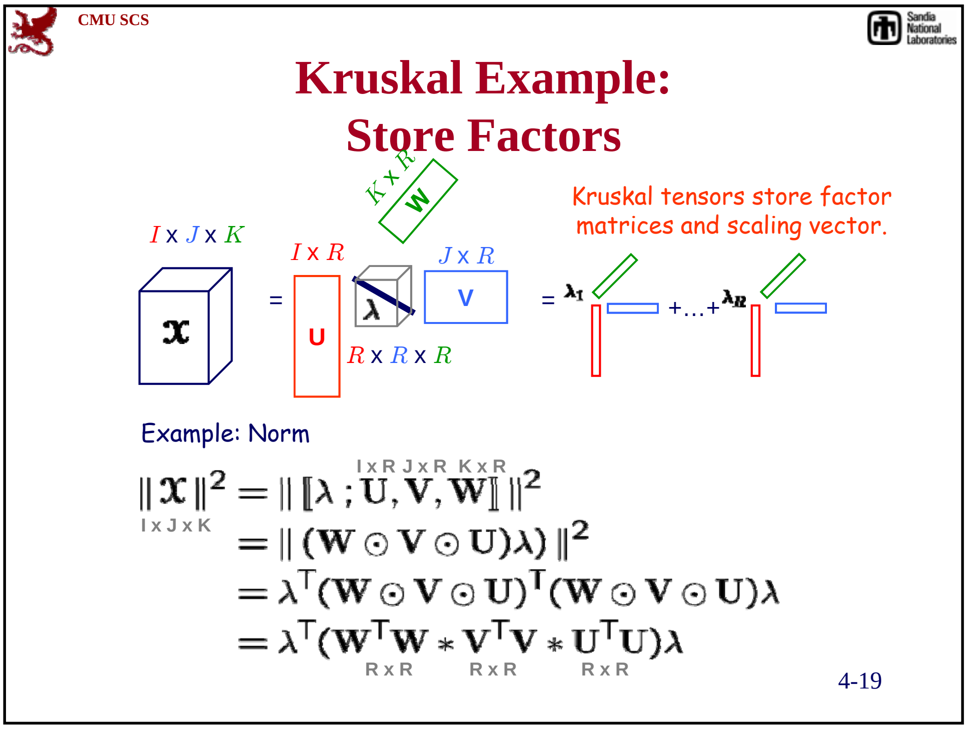
Tucker tensor stores the core (which can be dense, sparse, or structured) and the factors.

Example: Mode-3 Tensor-Vector Multiply

$$\begin{aligned}\mathbf{Y} &= \mathfrak{X} \,\bar{\times}_3\, \mathbf{z} \\ &= (\mathcal{G} \times_1 \mathbf{U} \times_2 \mathbf{V} \times_3 \mathbf{W}) \;\bar{\times}_3\, \mathbf{z} \\ &= \mathcal{G} \times_1 \mathbf{U} \times_2 \mathbf{V} \,\bar{\times}_3\, \mathbf{W}^{\mathsf{T}}\mathbf{z} \\ &= \underbrace{\mathcal{G} \,\bar{\times}_3\, \mathbf{W}^{\mathsf{T}}\mathbf{z}}_{\mathcal{H}} \times_1 \mathbf{U} \times_2 \mathbf{V} = [\![\mathcal{H}\,; \mathbf{U}, \mathbf{V}]\!]\end{aligned}\Bigg\}$$

Result is a Tucker Tensor

4-18



CMU SCS

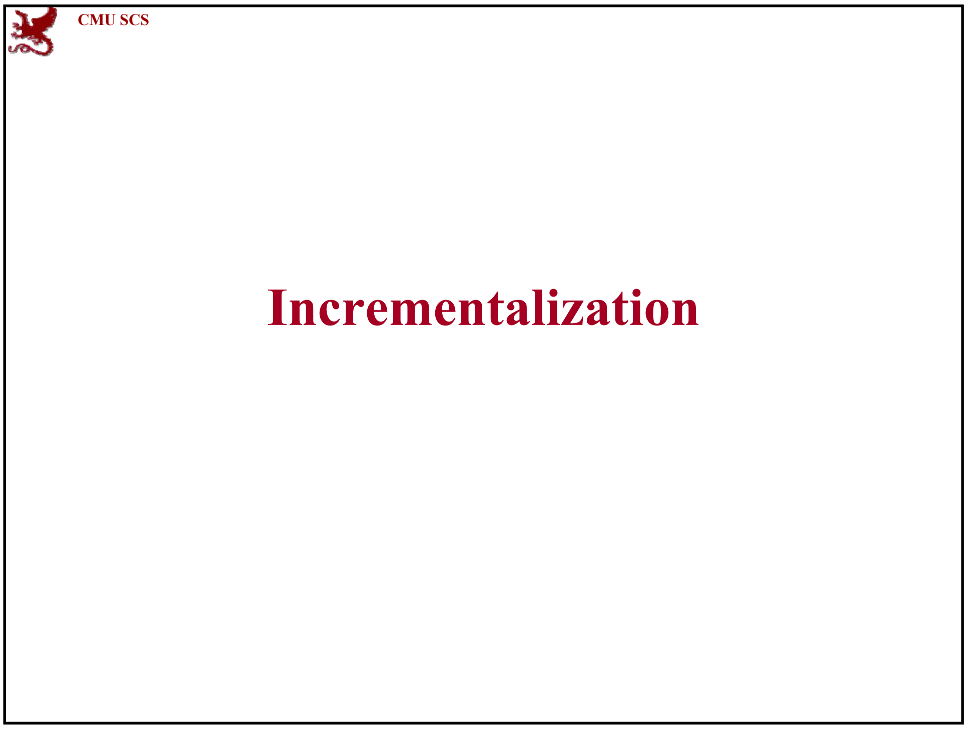
Kruskal Example: Store Factors

Kruskal tensors store factor matrices and scaling vector.

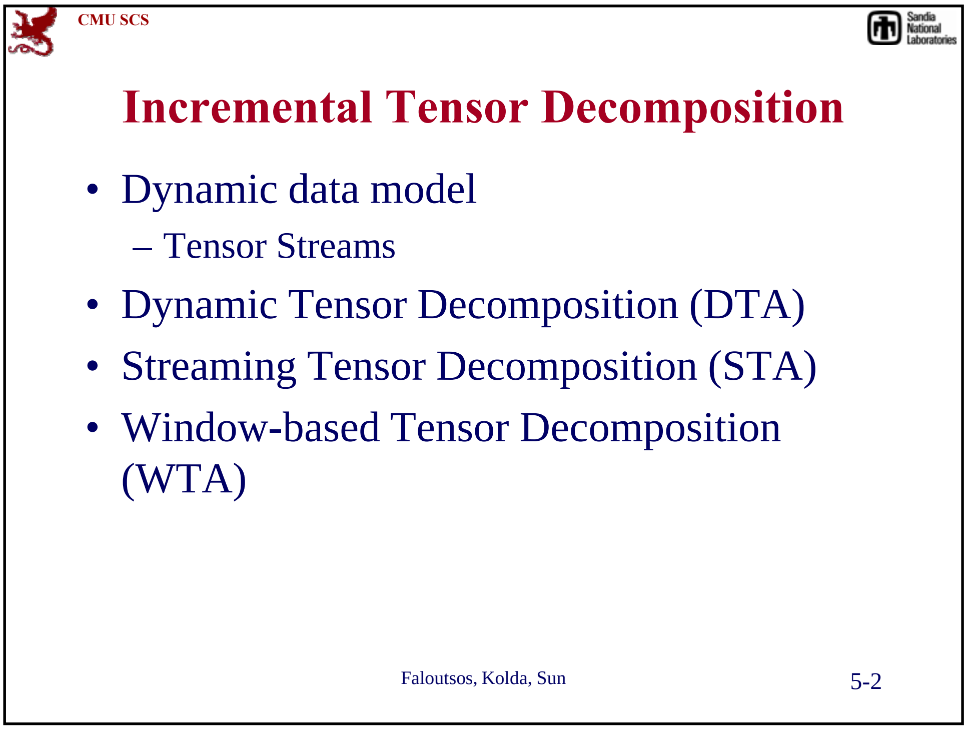
Example: Norm

$$
\begin{aligned}
\underset{\text{I x J x K}}{\|\mathcal{X}\|^2} &= \| [\![\lambda \, ; \overset{\text{I x R}}{\mathbf{U}}, \overset{\text{J x R}}{\mathbf{V}}, \overset{\text{K x R}}{\mathbf{W}}]\!] \|^2 \\
&= \| (\mathbf{W} \odot \mathbf{V} \odot \mathbf{U})\lambda) \|^2 \\
&= \lambda^{\mathsf{T}}(\mathbf{W} \odot \mathbf{V} \odot \mathbf{U})^{\mathsf{T}}(\mathbf{W} \odot \mathbf{V} \odot \mathbf{U})\lambda \\
&= \lambda^{\mathsf{T}}(\underset{\text{R x R}}{\mathbf{W}^{\mathsf{T}}\mathbf{W}} * \underset{\text{R x R}}{\mathbf{V}^{\mathsf{T}}\mathbf{V}} * \underset{\text{R x R}}{\mathbf{U}^{\mathsf{T}}\mathbf{U}})\lambda
\end{aligned}
$$

4-19

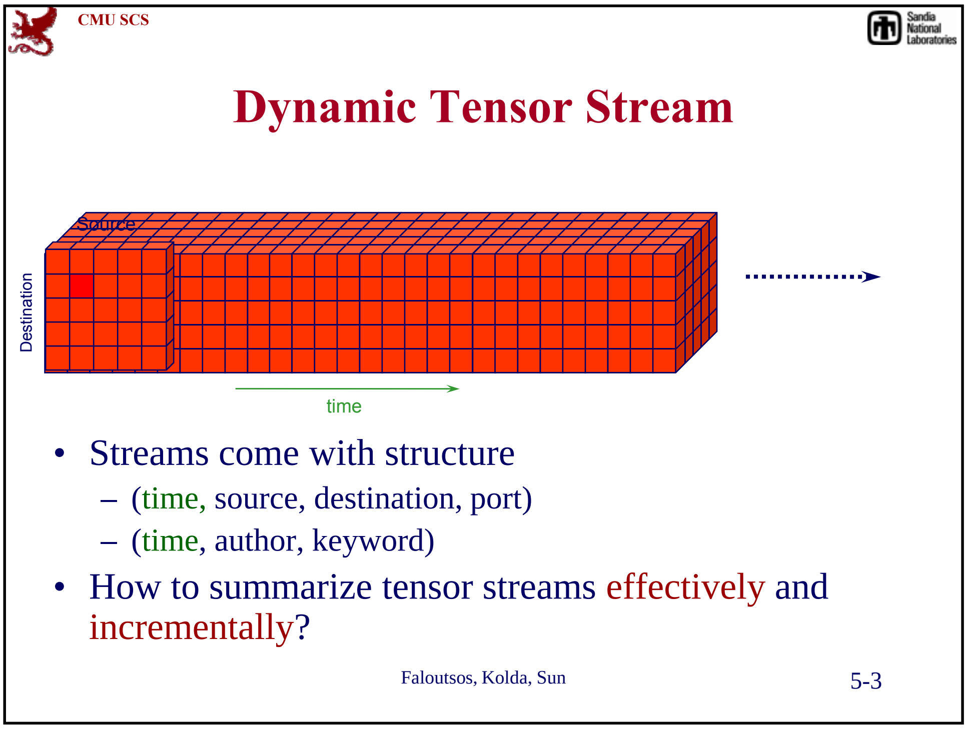


Incrementalization



Incremental Tensor Decomposition

- Dynamic data model
  - Tensor Streams
- Dynamic Tensor Decomposition (DTA)
- Streaming Tensor Decomposition (STA)
- Window-based Tensor Decomposition (WTA)

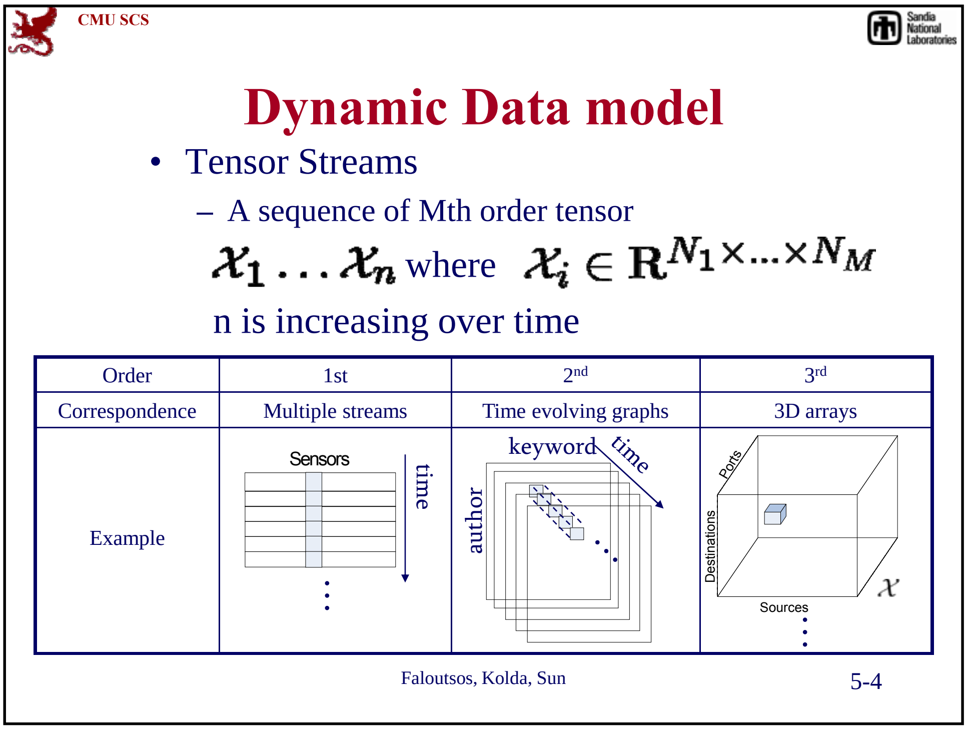


CMU SCS

Dynamic Tensor Stream

- Streams come with structure
  - (time, source, destination, port)
  - (time, author, keyword)
- How to summarize tensor streams effectively and incrementally?

Faloutsos, Kolda, Sun 5-3



CMU SCS

Dynamic Data model

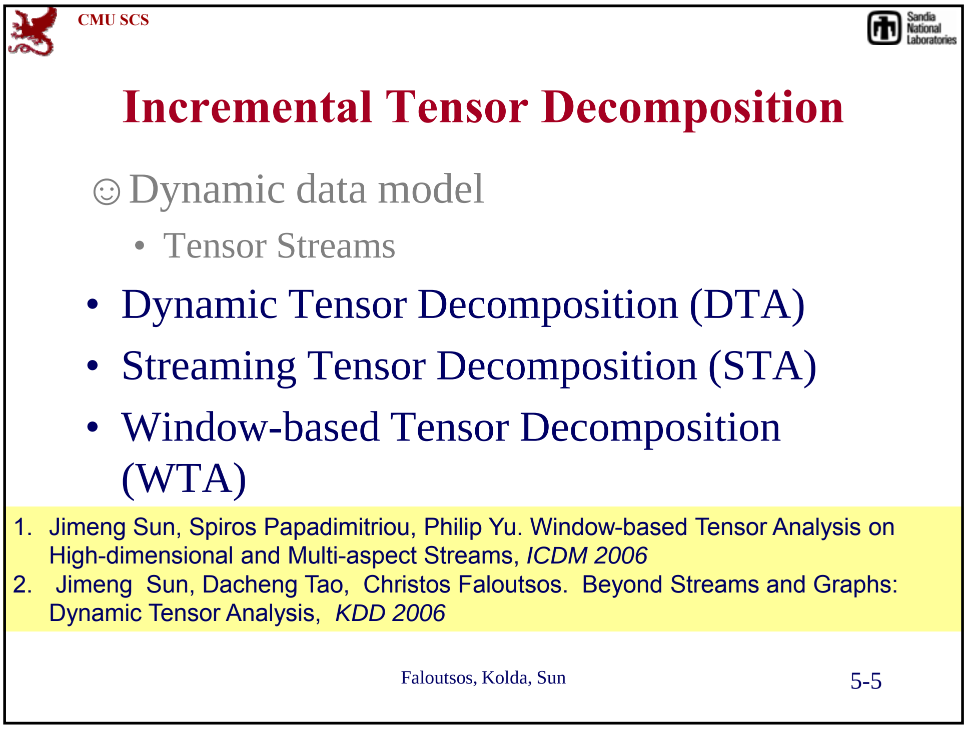
- Tensor Streams
  - A sequence of Mth order tensor

$\mathcal{X}_1 \ldots \mathcal{X}_n$ where $\mathcal{X}_i \in \mathbf{R}^{N_1 \times \ldots \times N_M}$

n is increasing over time

Order	1st	2nd	3rd
Correspondence	Multiple streams	Time evolving graphs	3D arrays
Example			

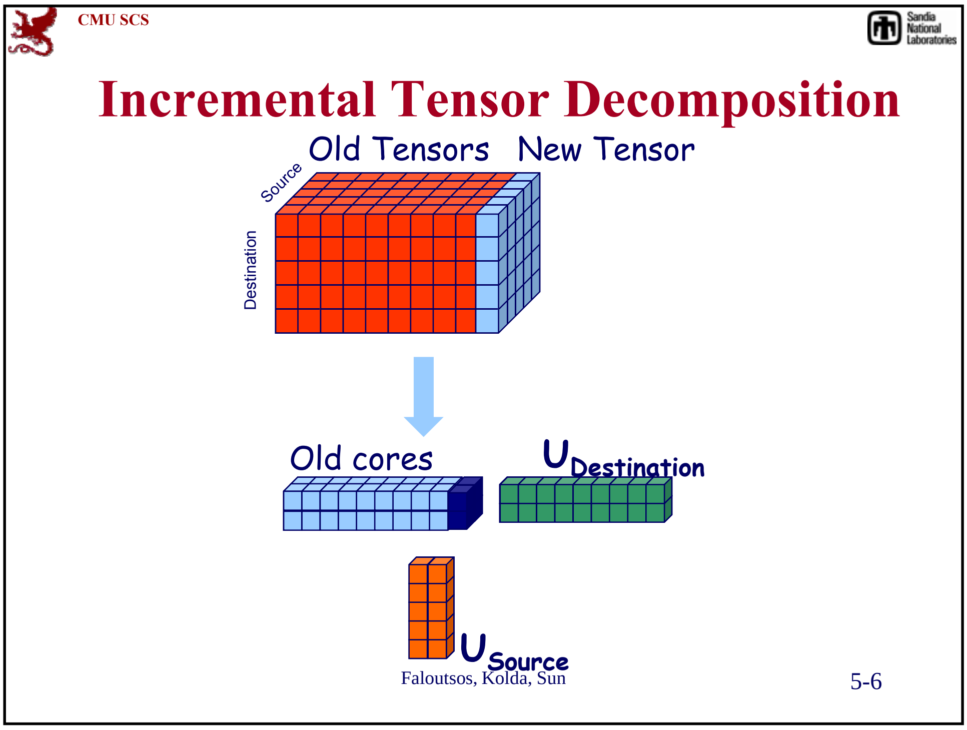
Faloutsos, Kolda, Sun 5-4



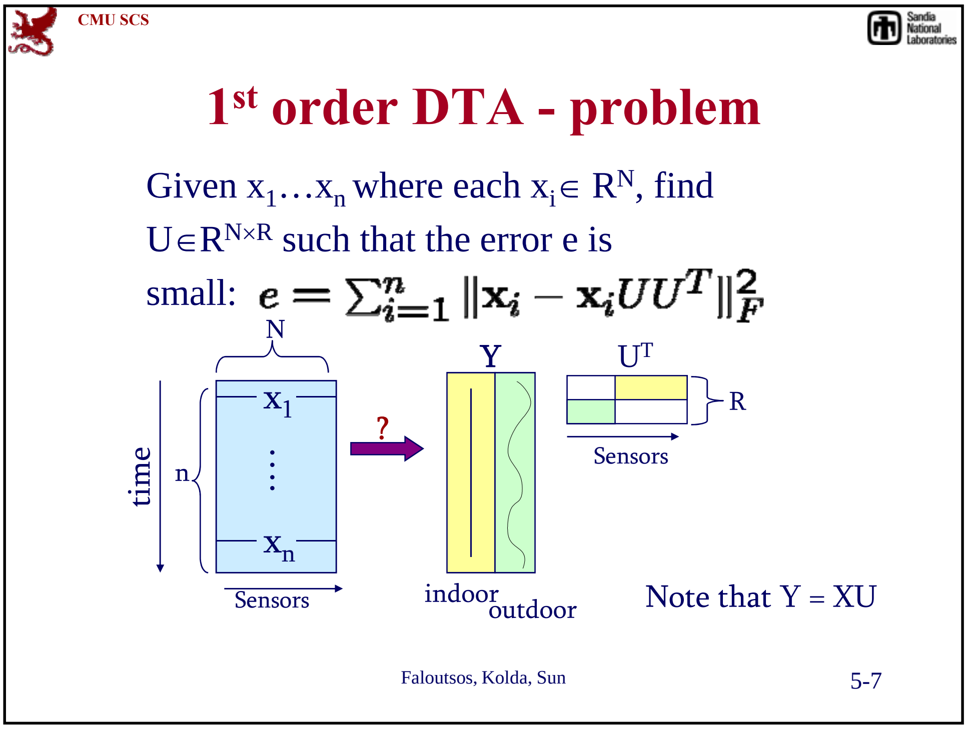
Incremental Tensor Decomposition

☺ Dynamic data model
  - Tensor Streams
- Dynamic Tensor Decomposition (DTA)
- Streaming Tensor Decomposition (STA)
- Window-based Tensor Decomposition (WTA)

1. Jimeng Sun, Spiros Papadimitriou, Philip Yu. Window-based Tensor Analysis on High-dimensional and Multi-aspect Streams, *ICDM 2006*
2. Jimeng Sun, Dacheng Tao, Christos Faloutsos. Beyond Streams and Graphs: Dynamic Tensor Analysis, *KDD 2006*



Incremental Tensor Decomposition

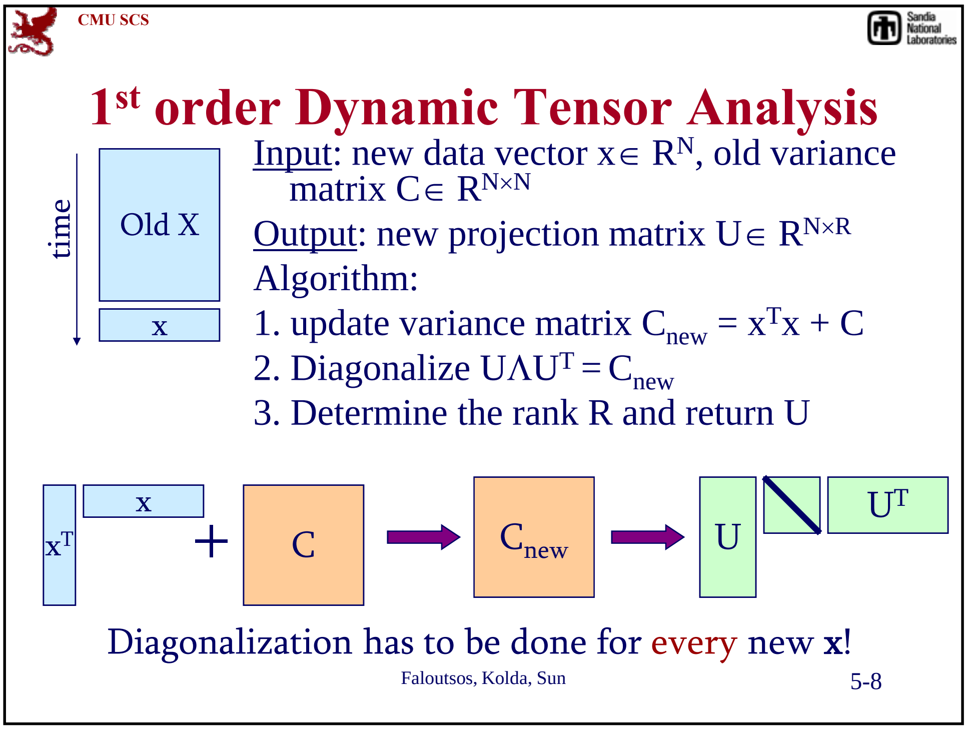


1st order DTA - problem

Given $x_1 \ldots x_n$ where each $x_i \in R^N$, find $U \in R^{N \times R}$ such that the error e is small: $e = \sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{x}_i U U^T\|_F^2$

N
time
n
x_1
⋮
x_n
Sensors
?
Y
indoor
outdoor
U^T
R
Sensors

Note that Y = XU



1st order Dynamic Tensor Analysis

time
Old X
x

Input: new data vector $x \in R^N$, old variance matrix $C \in R^{N \times N}$

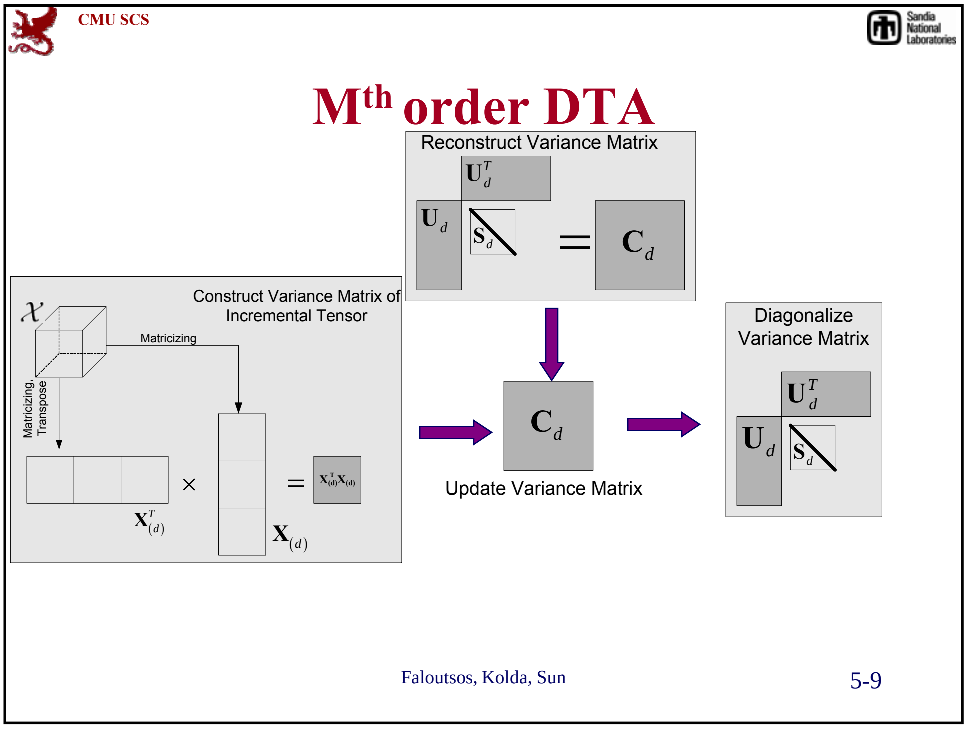
Output: new projection matrix $U \in R^{N \times R}$

Algorithm:

1. update variance matrix $C_{new} = x^T x + C$
2. Diagonalize $U \Lambda U^T = C_{new}$
3. Determine the rank R and return U

x^T
x
+
C
C_{new}
U
U^T

Diagonalization has to be done for every new **x**!



CMU SCS

M^{th} order DTA

Reconstruct Variance Matrix

$\mathbf{U}_d \, \mathbf{S}_d \, \mathbf{U}_d^T = \mathbf{C}_d$

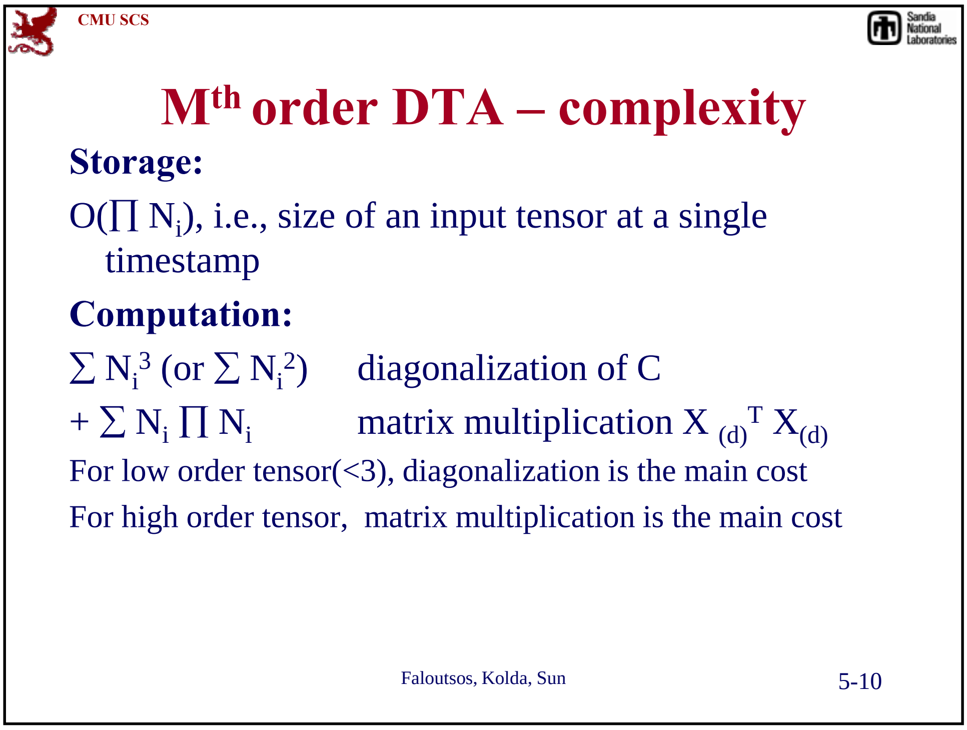
Construct Variance Matrix of Incremental Tensor

$\mathbf{X}_{(d)}^T \times \mathbf{X}_{(d)} = \mathbf{X}_{(d)}^T\mathbf{X}_{(d)}$

Update Variance Matrix

Diagonalize Variance Matrix

Faloutsos, Kolda, Sun 5-9



CMU SCS

M^{th} order DTA – complexity

Storage:

$O(\prod N_i)$, i.e., size of an input tensor at a single timestamp

Computation:

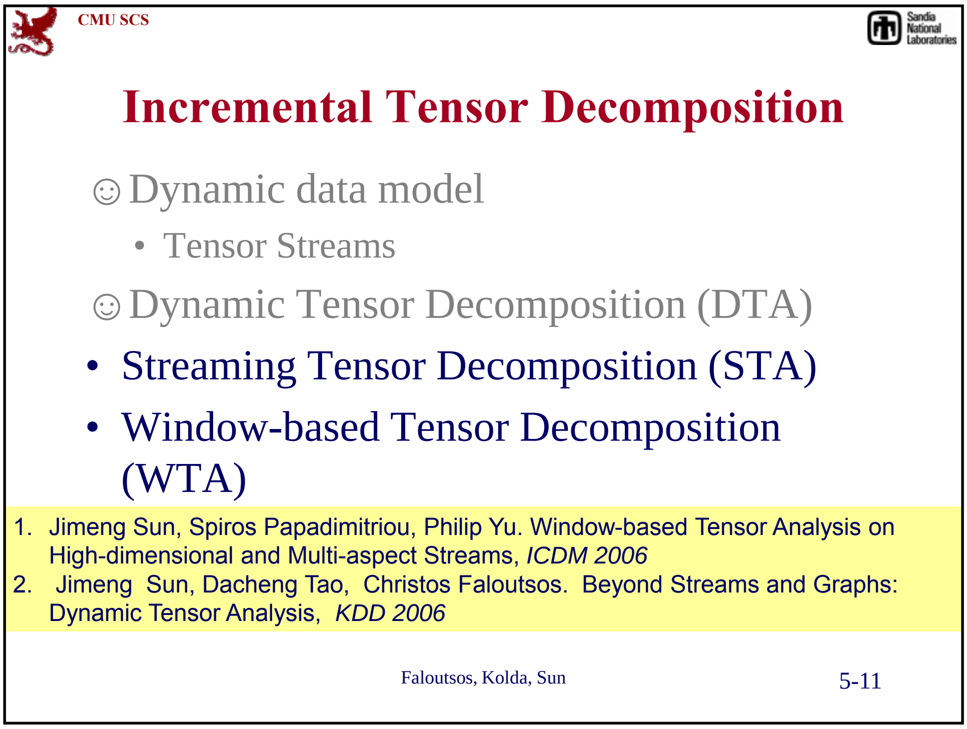
$\sum N_i^3$ (or $\sum N_i^2$) diagonalization of C

$+ \sum N_i \prod N_i$ matrix multiplication $X_{(d)}^T X_{(d)}$

For low order tensor(<3), diagonalization is the main cost

For high order tensor, matrix multiplication is the main cost

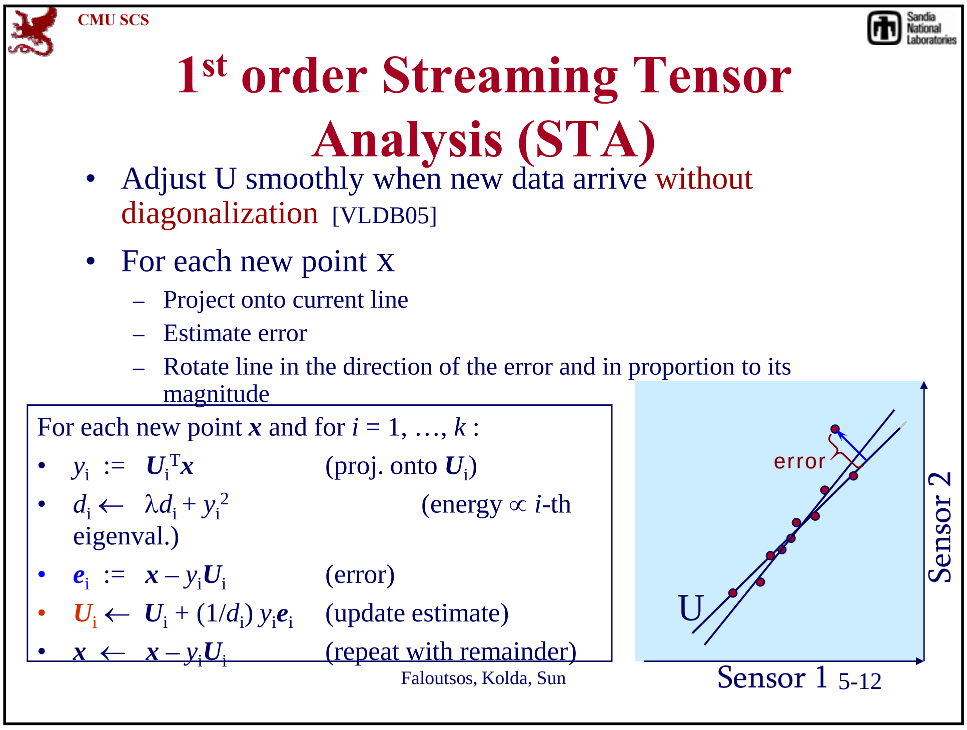
Faloutsos, Kolda, Sun 5-10



Incremental Tensor Decomposition

- ☺ Dynamic data model
  - Tensor Streams
- ☺ Dynamic Tensor Decomposition (DTA)
- Streaming Tensor Decomposition (STA)
- Window-based Tensor Decomposition (WTA)

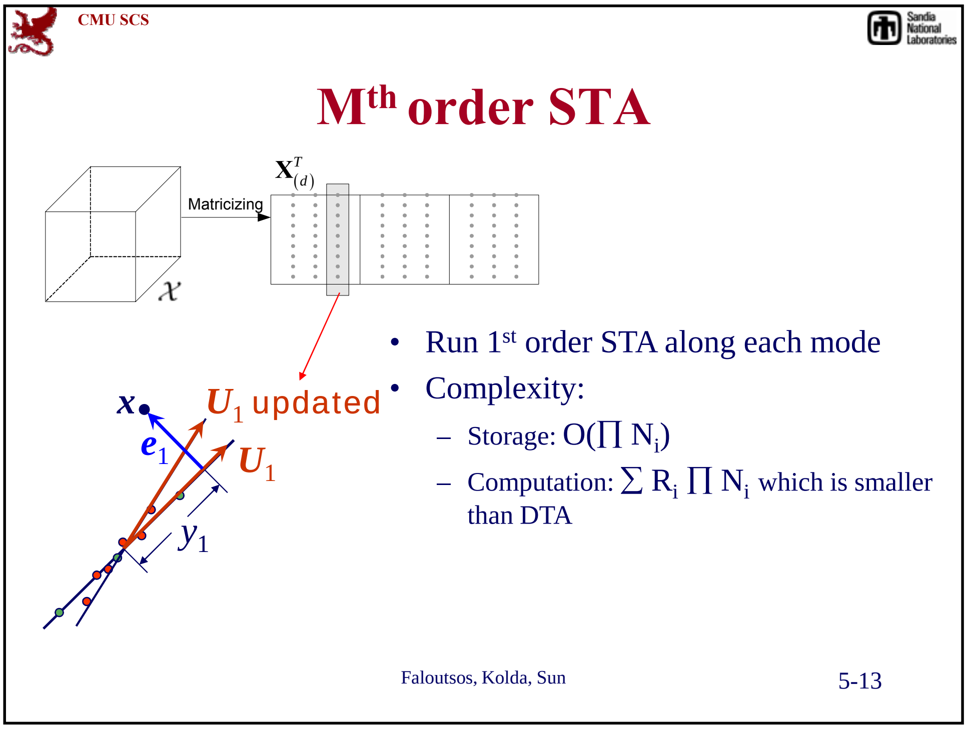
1. Jimeng Sun, Spiros Papadimitriou, Philip Yu. Window-based Tensor Analysis on High-dimensional and Multi-aspect Streams, *ICDM 2006*
2. Jimeng Sun, Dacheng Tao, Christos Faloutsos. Beyond Streams and Graphs: Dynamic Tensor Analysis, *KDD 2006*



1st order Streaming Tensor Analysis (STA)

- Adjust U smoothly when new data arrive without diagonalization [VLDB05]
- For each new point x
  - Project onto current line
  - Estimate error
  - Rotate line in the direction of the error and in proportion to its magnitude

```
For each new point $\boldsymbol{x}$ and for $i$ = 1, …, $k$ :
•  $y_i := \boldsymbol{U}_i^T \boldsymbol{x}$            (proj. onto $\boldsymbol{U}_i$)
•  $d_i \leftarrow \lambda d_i + y_i^2$            (energy ∝ $i$-th eigenval.)
•  $\boldsymbol{e}_i := \boldsymbol{x} - y_i \boldsymbol{U}_i$          (error)
•  $\boldsymbol{U}_i \leftarrow \boldsymbol{U}_i + (1/d_i)\, y_i \boldsymbol{e}_i$   (update estimate)
•  $\boldsymbol{x} \leftarrow \boldsymbol{x} - y_i \boldsymbol{U}_i$          (repeat with remainder)
```



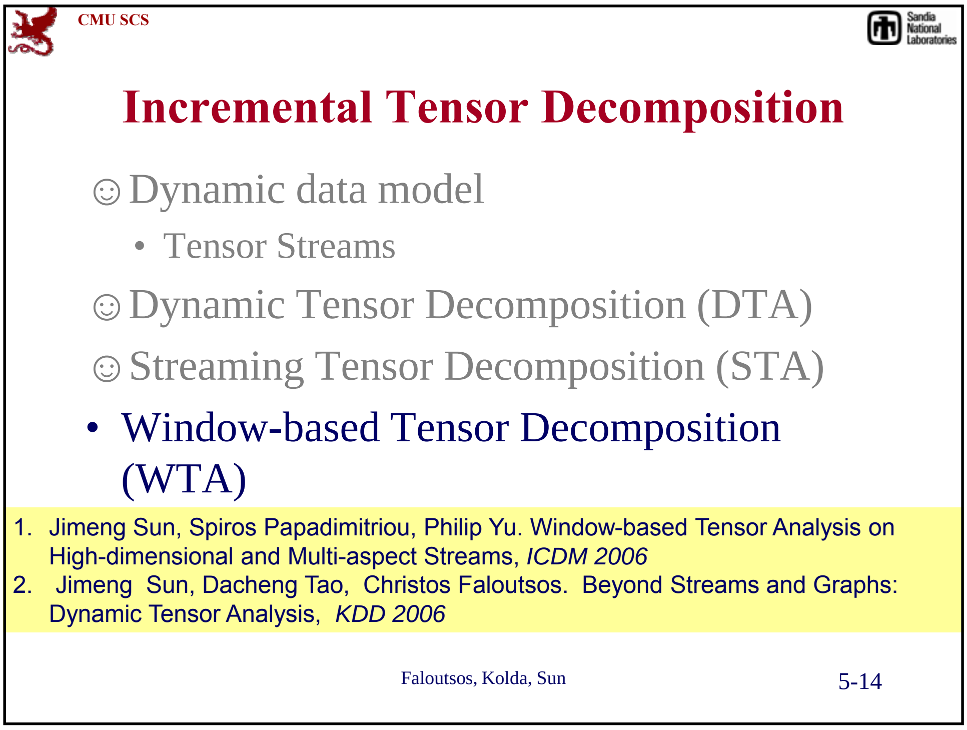
CMU SCS

Sandia National Laboratories

Mth order STA

- Run 1st order STA along each mode
- Complexity:
  - Storage: $O(\prod N_i)$
  - Computation: $\sum R_i \prod N_i$ which is smaller than DTA

Faloutsos, Kolda, Sun 5-13



CMU SCS

Sandia National Laboratories

Incremental Tensor Decomposition

☺ Dynamic data model

- Tensor Streams

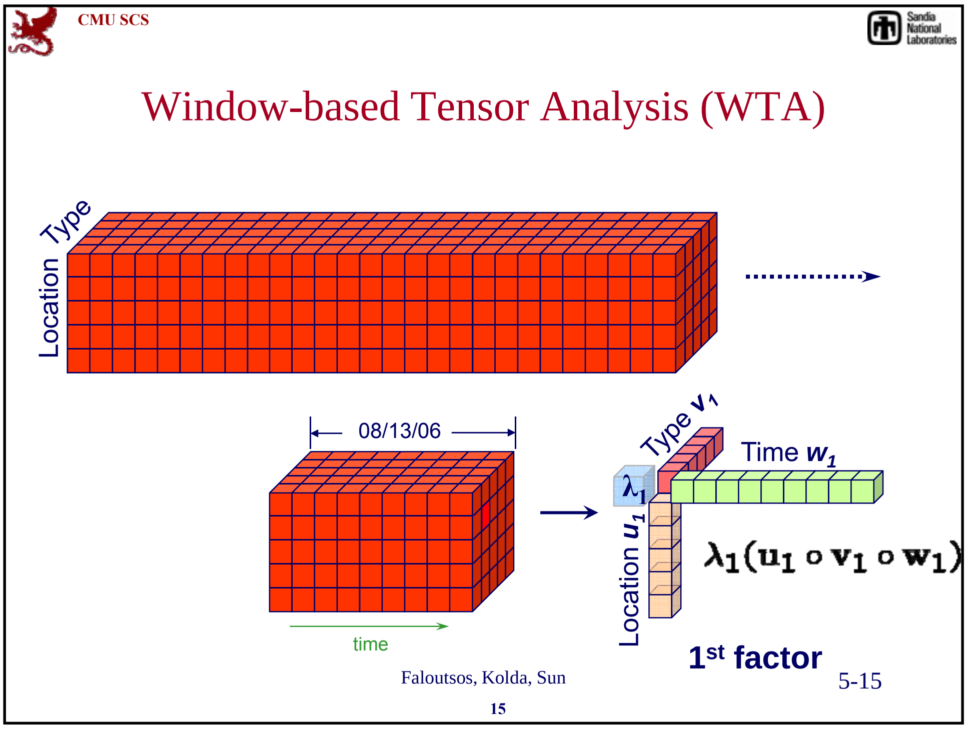
☺ Dynamic Tensor Decomposition (DTA)

☺ Streaming Tensor Decomposition (STA)

- Window-based Tensor Decomposition (WTA)

1. Jimeng Sun, Spiros Papadimitriou, Philip Yu. Window-based Tensor Analysis on High-dimensional and Multi-aspect Streams, *ICDM 2006*
2. Jimeng Sun, Dacheng Tao, Christos Faloutsos. Beyond Streams and Graphs: Dynamic Tensor Analysis, *KDD 2006*

Faloutsos, Kolda, Sun 5-14

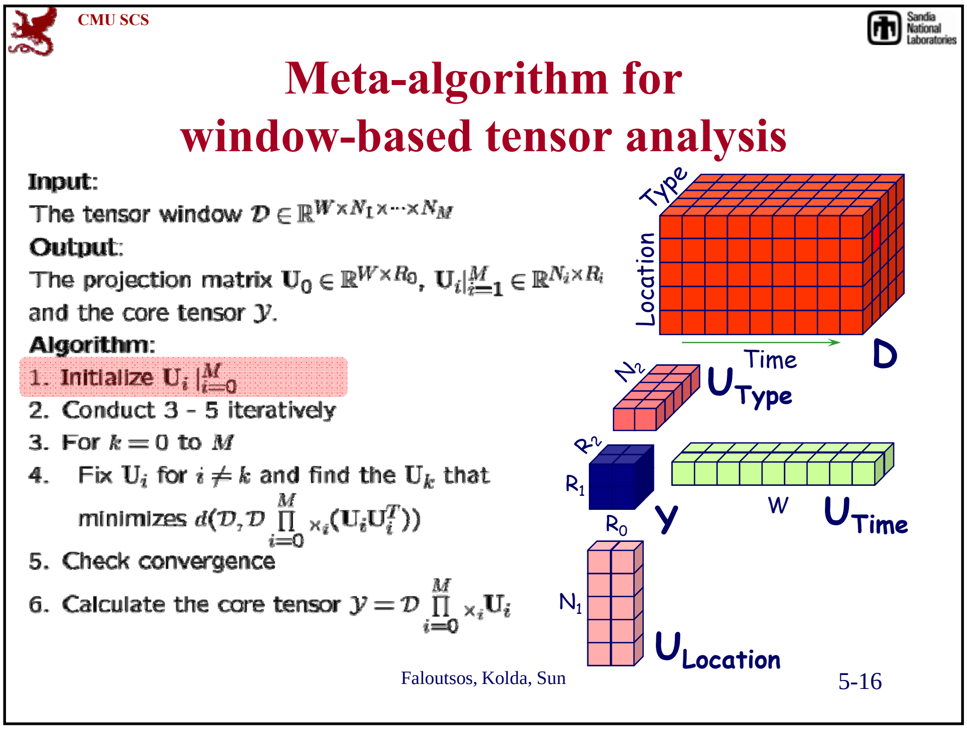


CMU SCS

Window-based Tensor Analysis (WTA)

$\lambda_1(\mathbf{u}_1 \circ \mathbf{v}_1 \circ \mathbf{w}_1)$

1st factor



CMU SCS

Meta-algorithm for window-based tensor analysis

Input:

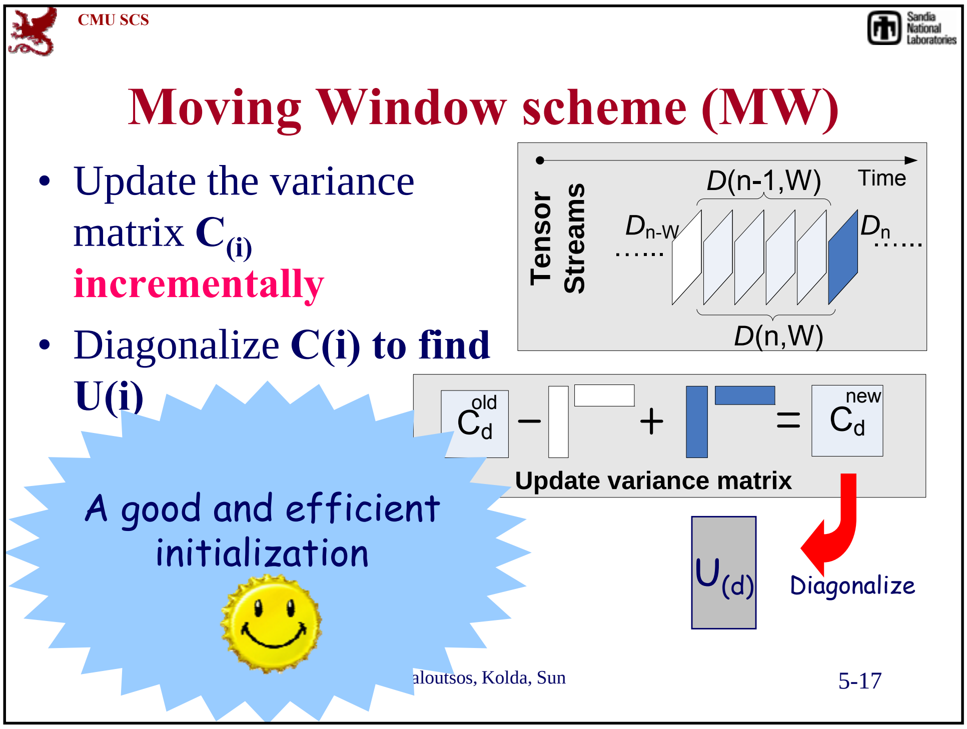
The tensor window $\mathcal{D} \in \mathbb{R}^{W \times N_1 \times \cdots \times N_M}$

Output:

The projection matrix $\mathbf{U}_0 \in \mathbb{R}^{W \times R_0}$, $\mathbf{U}_i|_{i=1}^{M} \in \mathbb{R}^{N_i \times R_i}$ and the core tensor $\mathcal{Y}$.

Algorithm:

```
1. Initialize $\mathbf{U}_i \,|_{i=0}^{M}$
2. Conduct 3 - 5 iteratively
3. For $k = 0$ to $M$
4.    Fix $\mathbf{U}_i$ for $i \neq k$ and find the $\mathbf{U}_k$ that
      minimizes $d(\mathcal{D}, \mathcal{D} \prod_{i=0}^{M} \times_i (\mathbf{U}_i \mathbf{U}_i^T))$
5. Check convergence
6. Calculate the core tensor $\mathcal{Y} = \mathcal{D} \prod_{i=0}^{M} \times_i \mathbf{U}_i$
```



CMU SCS

Moving Window scheme (MW)

- Update the variance matrix $C_{(i)}$ **incrementally**
- Diagonalize **C(i) to find U(i)**

A good and efficient initialization

aloutsos, Kolda, Sun 5-17

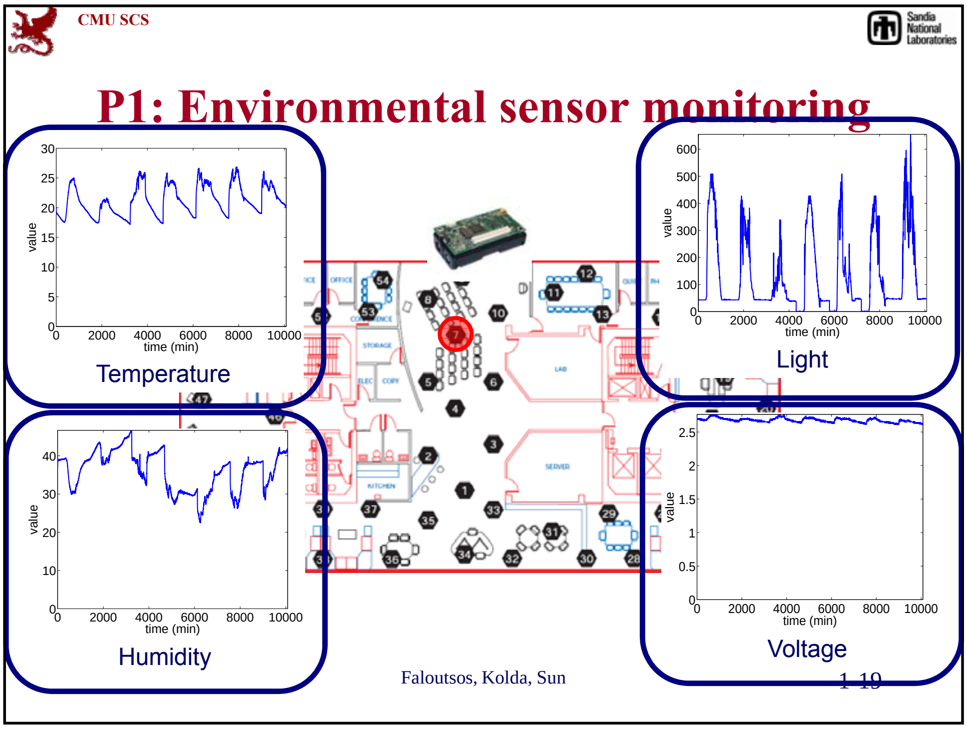
CMU SCS

Roadmap

- Motivation
- Matrix tools
- Tensor basics
- Tensor extensions
- Software demo
- Case studies

San Francisco
Sacramento
Reno Nevada
Salt Lake City
Utah
Denver
Colorado
Nebraska
Lincoln
Kansas
Kansas City
Wichita
Iowa
Chicago
Illinois
Indiana
Missouri
Springfield
Louisville
Kentucky
Fort Wayne
Ohio
West Virginia
Virginia
Pennsylvania
Philadelphia
New York
Conn

Faloutsos, Kolda, Sun 5-18

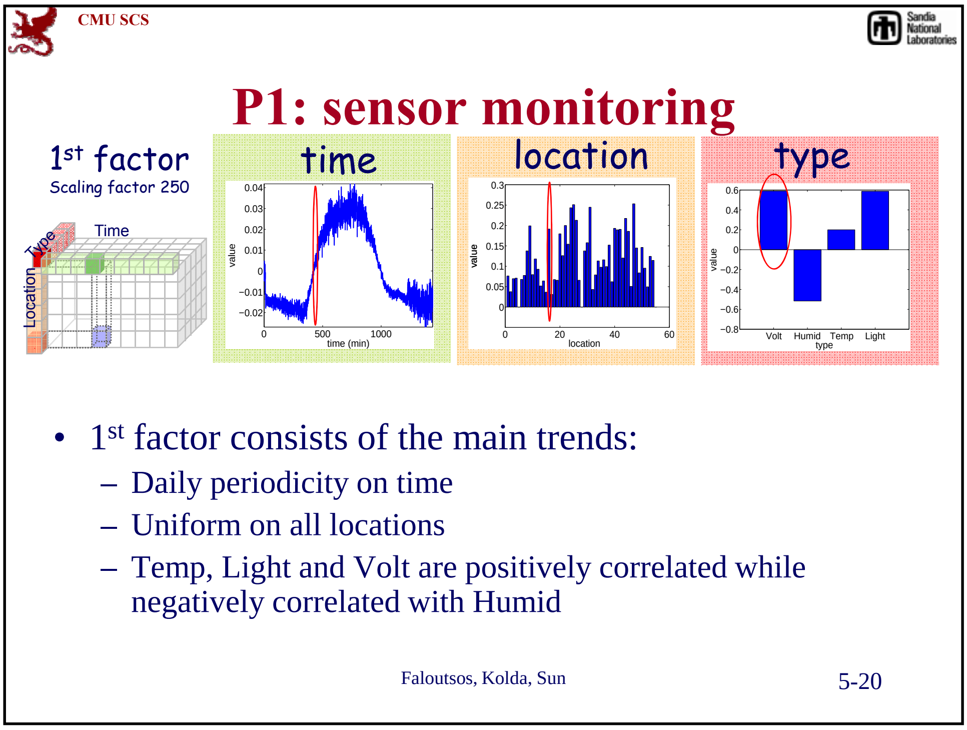


CMU SCS

P1: Environmental sensor monitoring

Faloutsos, Kolda, Sun

1-19



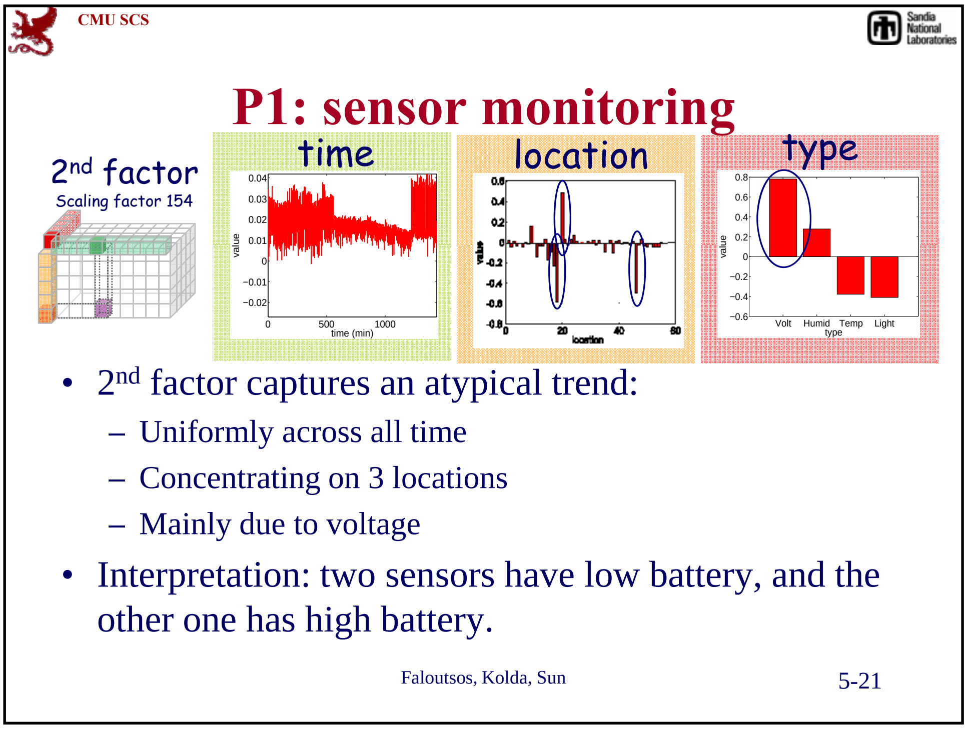
CMU SCS

P1: sensor monitoring

- 1^{st} factor consists of the main trends:
  - Daily periodicity on time
  - Uniform on all locations
  - Temp, Light and Volt are positively correlated while negatively correlated with Humid

Faloutsos, Kolda, Sun

5-20

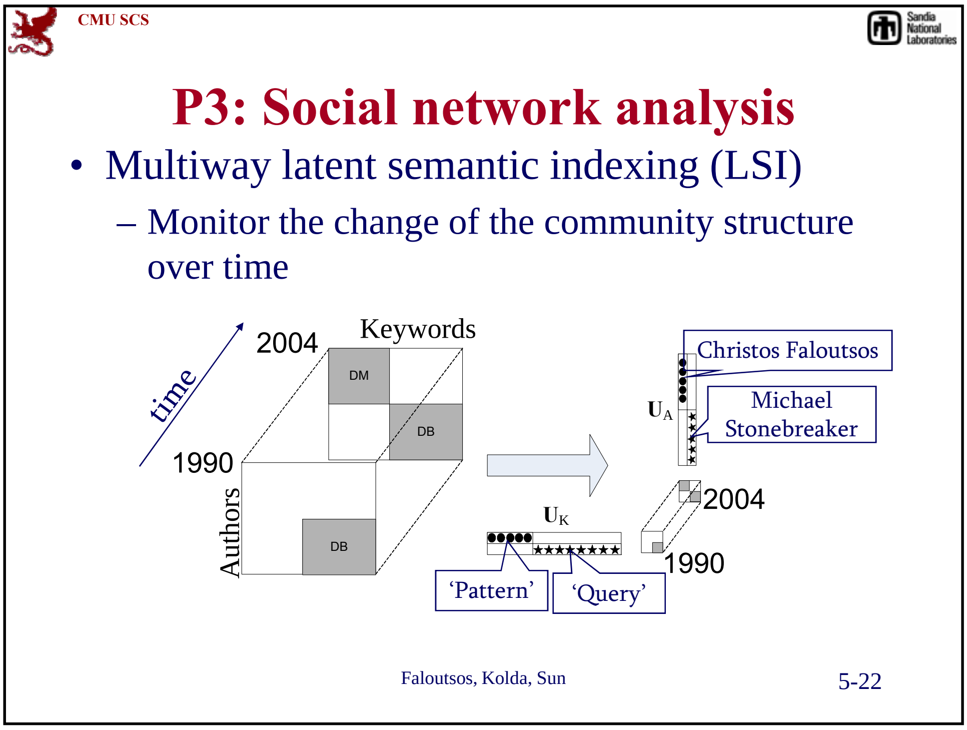


CMU SCS

P1: sensor monitoring

- 2nd factor captures an atypical trend:
  - Uniformly across all time
  - Concentrating on 3 locations
  - Mainly due to voltage
- Interpretation: two sensors have low battery, and the other one has high battery.

Faloutsos, Kolda, Sun 5-21

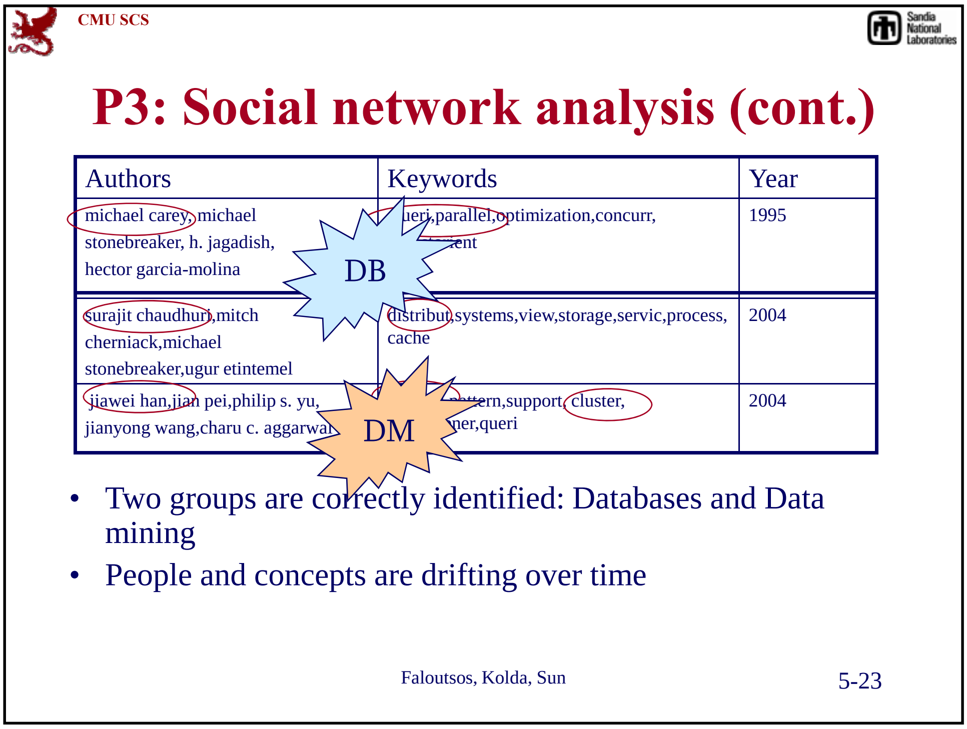


CMU SCS

P3: Social network analysis

- Multiway latent semantic indexing (LSI)
  - Monitor the change of the community structure over time

Faloutsos, Kolda, Sun 5-22



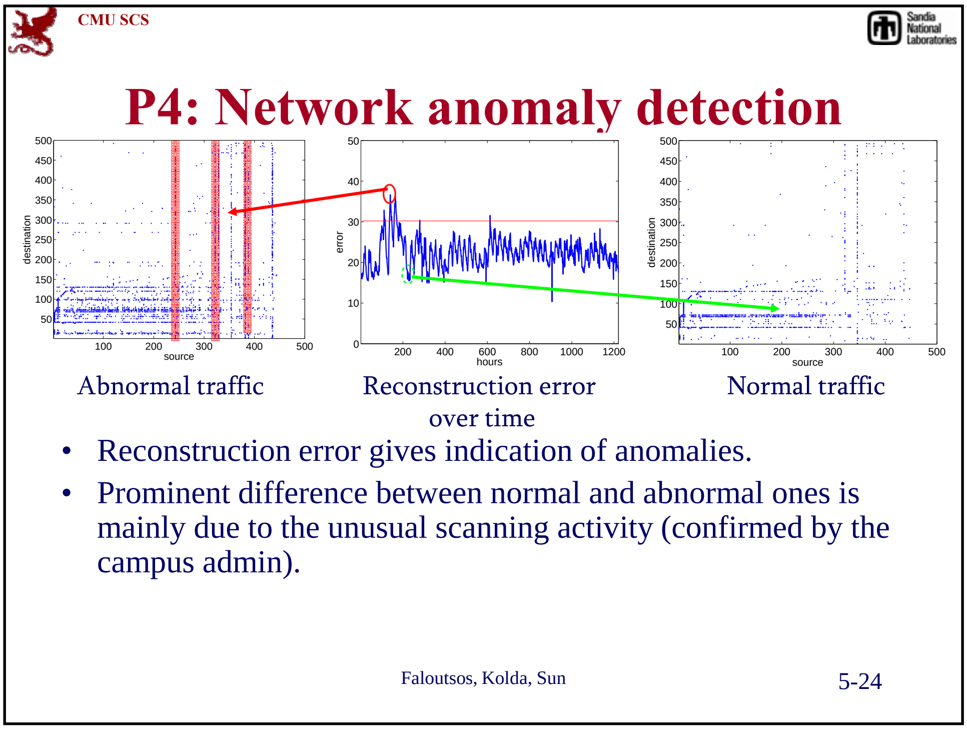
CMU SCS

P3: Social network analysis (cont.)

Authors	Keywords	Year
michael carey, michael stonebreaker, h. jagadish, hector garcia-molina	ueri,parallel,optimization,concurr, ent	1995
surajit chaudhuri,mitch cherniack,michael stonebreaker,ugur etintemel	distribut,systems,view,storage,servic,process, cache	2004
jiawei han,jian pei,philip s. yu, jianyong wang,charu c. aggarwa	ttern,support, cluster, ner,queri	2004

- Two groups are correctly identified: Databases and Data mining
- People and concepts are drifting over time

Faloutsos, Kolda, Sun 5-23



CMU SCS

P4: Network anomaly detection

Abnormal traffic

Reconstruction error over time

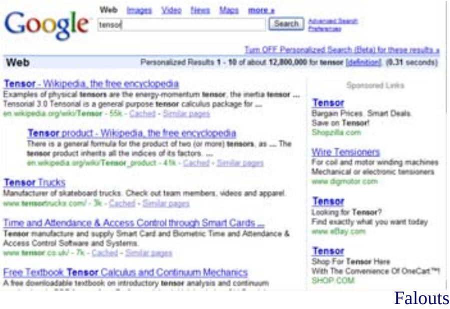
Normal traffic

- Reconstruction error gives indication of anomalies.
- Prominent difference between normal and abnormal ones is mainly due to the unusual scanning activity (confirmed by the campus admin).

Faloutsos, Kolda, Sun 5-24

P5: Web graph mining

- How to order the importance of web pages?
  - Kleinberg's algorithm HITS
  - PageRank
  - Tensor extension on HITS (TOPHITS)



Google

Web Images Video News Maps more »

tensor

Search

Advanced Search
Preferences

Turn OFF Personalized Search (Beta) for these results »

Web Personalized Results **1** - **10** of about **12,800,000** for **tensor** [definition]. (**0.31** seconds)

Tensor - Wikipedia, the free encyclopedia
Examples of physical **tensors** are the energy-momentum **tensor**, the inertia **tensor** ...
Tensorial 3.0 Tensorial is a general purpose **tensor** calculus package for ...
en.wikipedia.org/wiki/**Tensor** - 55k - Cached - Similar pages

Tensor product - Wikipedia, the free encyclopedia
There is a general formula for the product of two (or more) **tensors**, as ... The **tensor** product inherits all the indices of its factors. ...
en.wikipedia.org/wiki/**Tensor**_product - 41k - Cached - Similar pages

Tensor Trucks
Manufacturer of skateboard trucks. Check out team members, videos and apparel.
www.**tensor**trucks.com/ - 3k - Cached - Similar pages

Time and Attendance & Access Control through Smart Cards ...
Tensor manufacture and supply Smart Card and Biometric Time and Attendance & Access Control Software and Systems.
www.**tensor**.co.uk/ - 7k - Cached - Similar pages

Free Textbook **Tensor** Calculus and Continuum Mechanics
A free downloadable textbook on introductory **tensor** analysis and continuum

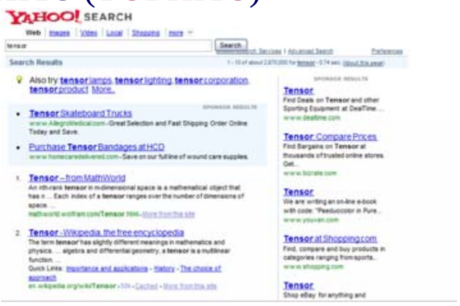
Sponsored Links

Tensor
Bargain Prices. Smart Deals.
Save on **Tensor**!
Shopzilla.com

Wire Tensioners
For coil and motor winding machines
Mechanical or electronic tensioners
www.digmotor.com

Tensor
Looking for **Tensor**?
Find exactly what you want today
www.eBay.com

Tensor
Shop For **Tensor** Here
With The Convenience Of OneCart™!
SHOP.COM



YAHOO! SEARCH

Web | Images | Video | Local | Shopping | more

tensor

Search

Yahoo! Search Services | Advanced Search Preferences

Search Results 1 - 10 of about 2,870,000 for tensor - 0.74 sec. (About this page)

Also try **tensor** lamps, **tensor** lighting, **tensor** corporation, **tensor** product More...

SPONSOR RESULTS

- **Tensor** Skateboard Trucks
  www.AllegroMedical.com - Great Selection and Fast Shipping Order Online Today and Save.
- Purchase **Tensor** Bandages at HCD
  www.homecaredelivered.com - Save on our full line of wound care supplies.

1. **Tensor** -- from MathWorld
   An nth-rank **tensor** in m-dimensional space is a mathematical object that has n ... Each index of a **tensor** ranges over the number of dimensions of space. ...
   mathworld.wolfram.com/**Tensor**.html - More from this site
2. **Tensor** - Wikipedia, the free encyclopedia
   The term **tensor** has slightly different meanings in mathematics and physics. ... algebra and differential geometry, a **tensor** is a multilinear function. ...
   Quick Links: Importance and applications - History - The choice of approach
   en.wikipedia.org/wiki/**Tensor** - 10k - Cached - More from this site

SPONSOR RESULTS

Tensor
Find Deals on **Tensor** and other Sporting Equipment at DealTime. ...
www.dealtime.com

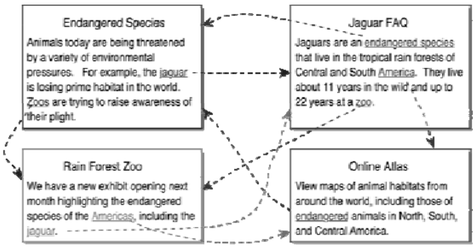
Tensor Compare Prices
Find Bargains on **Tensor** at thousands of trusted online stores. Get...
www.bizrate.com

Tensor
We are writing an on-line e-book with code: "Pseudocolor in Pure...
www.youvan.com

Tensor at Shopping.com
Find, compare and buy products in categories ranging from sports...
www.shopping.com

Tensor
Shop eBay for anything and

Kleinberg's Hubs and Authorities (the HITS method)

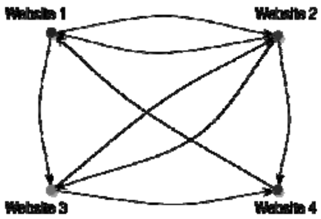


Endangered Species
Animals today are being threatened by a variety of environmental pressures. For example, the jaguar is losing prime habitat in the world. Zoos are trying to raise awareness of their plight.

Jaguar FAQ
Jaguars are an endangered species that live in the tropical rain forests of Central and South America. They live about 11 years in the wild and up to 22 years at a zoo.

Rain Forest Zoo
We have a new exhibit opening next month highlighting the endangered species of the Americas, including the jaguar.

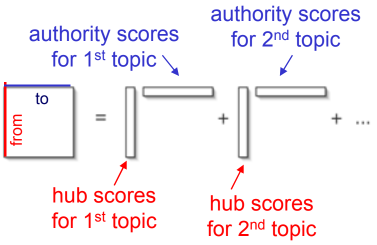
Online Atlas
View maps of animal habitats from around the world, including those of endangered animals in North, South, and Central America.



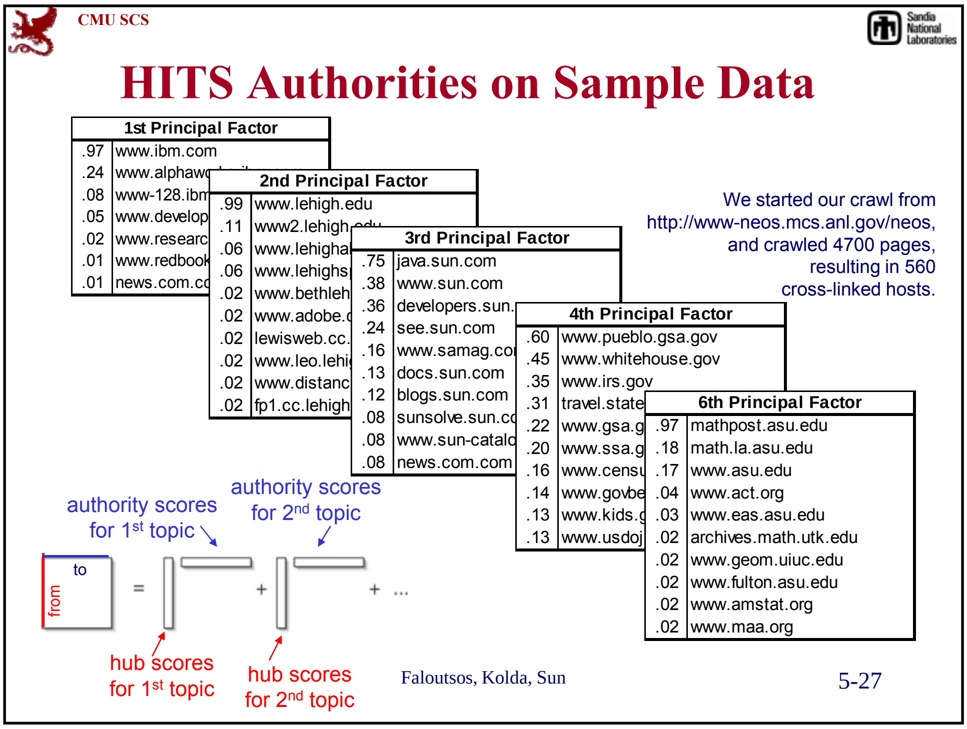
Sparse adjacency matrix and its SVD:

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

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



Kleinberg, JACM, 1999



CMU SCS

HITS Authorities on Sample Data

1st Principal Factor	
.97	www.ibm.com
.24	www.alphawo
.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.edu
.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

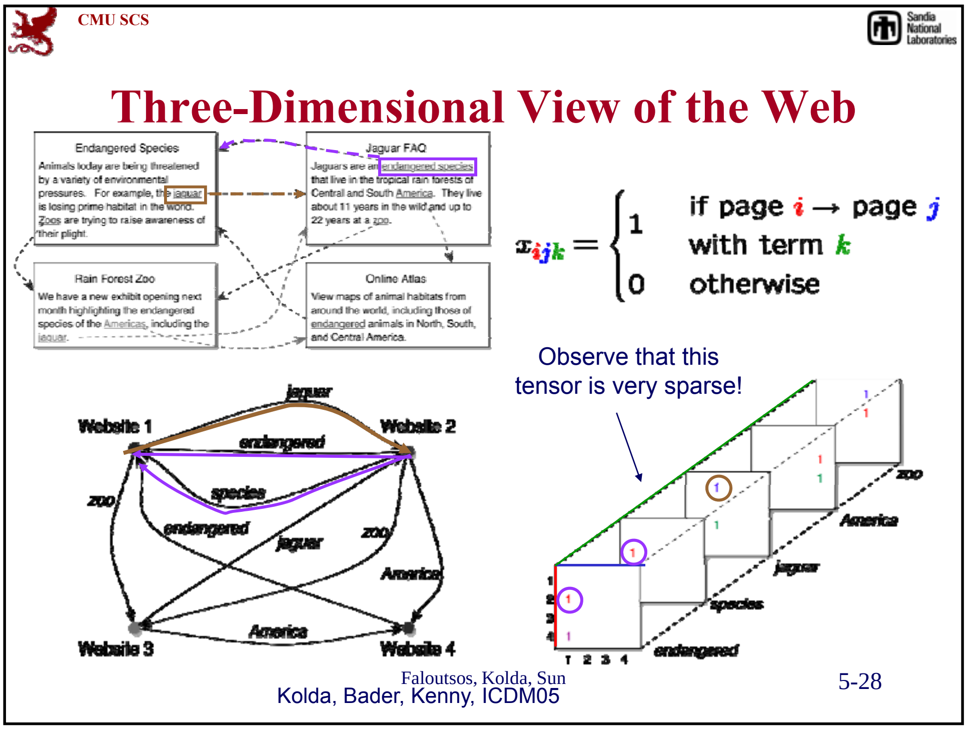
3rd Principal Factor	
.75	java.sun.com
.38	www.sun.com
.36	developers.sun.
.24	see.sun.com
.16	www.samag.co
.13	docs.sun.com
.12	blogs.sun.com
.08	sunsolve.sun.c
.08	www.sun-catalo
.08	news.com.com

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.

Faloutsos, Kolda, Sun 5-27



CMU SCS

Three-Dimensional View of the Web

Endangered Species
Animals today are being threatened by a variety of environmental pressures. For example, the jaguar is losing prime habitat in the world. Zoos are trying to raise awareness of their plight.
Jaguar FAQ
Jaguars are an endangered species that live in the tropical rain forests of Central and South America. They live about 11 years in the wild and up to 22 years at a zoo.
Rain Forest Zoo
We have a new exhibit opening next month highlighting the endangered species of the Americas, including the jaguar.
Online Atlas
View maps of animal habitats from around the world, including those of endangered animals in North, South, and Central America.
Website 1
Website 2
Website 3
Website 4
jaguar
endangered
species
zoo
endangered
jaguar
zoo
America
America
1 2 3 4
zoo
America
jaguar
species
endangered

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

Observe that this tensor is very sparse!

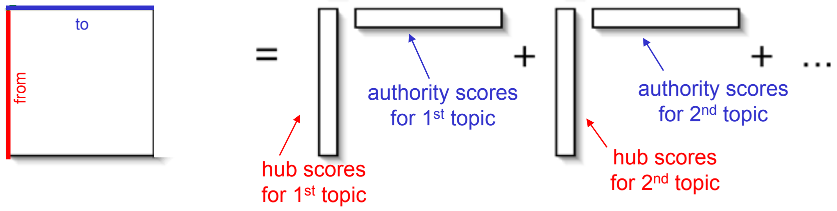
Faloutsos, Kolda, Sun 5-28

Kolda, Bader, Kenny, ICDM05

Topical HITS (TOPHITS)

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

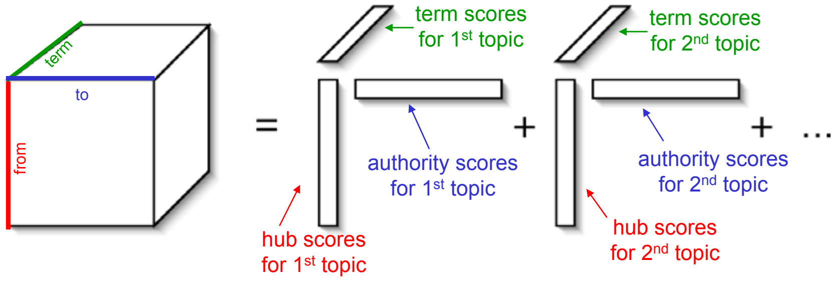
$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, \mathbf{h}_r \circ \mathbf{a}_r$$

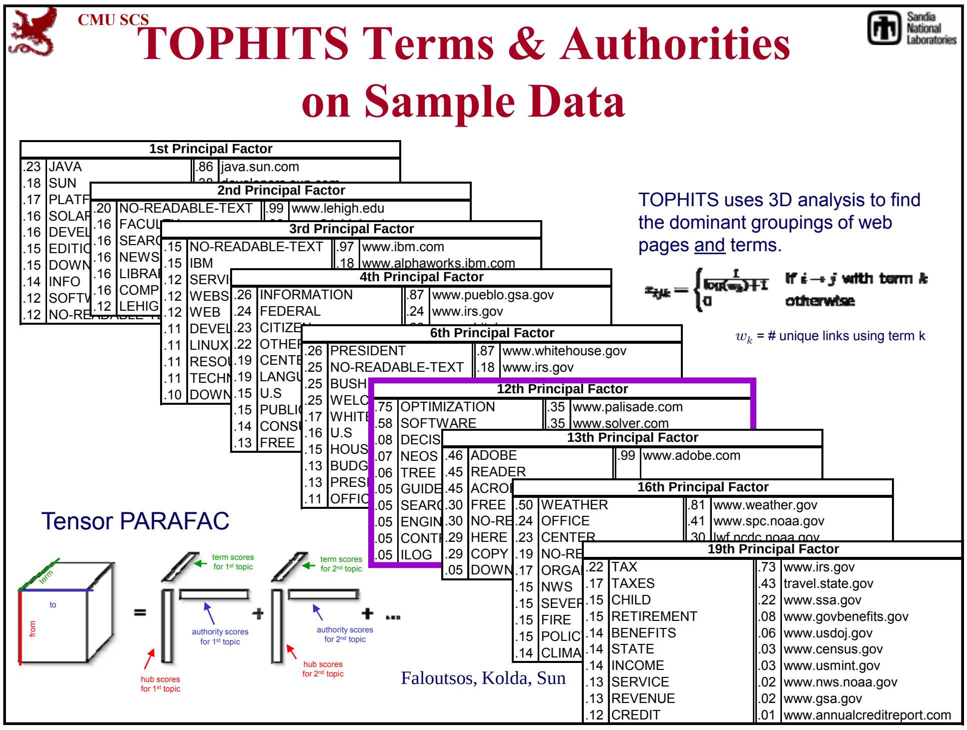

to
from
=
+
+ …
authority scores for 1st topic
authority scores for 2nd topic
hub scores for 1st topic
hub scores for 2nd topic

Topical HITS (TOPHITS)

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

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


term
to
from
=
term scores for 1st topic
term scores for 2nd topic
+
+ …
authority scores for 1st topic
authority scores for 2nd topic
hub scores for 1st topic
hub scores for 2nd topic



CMU SCS

TOPHITS Terms & Authorities on Sample Data

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-READABLE-TE		

2nd Principal Factor			
.20	NO-READABLE-TEXT	.99	www.lehigh.edu
.16	FACULTY		
.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	CONSU		
.13	FREE		

6th Principal Factor			
.26	PRESIDENT	.87	www.whitehouse.gov
.25	NO-READABLE-TEXT	.18	www.irs.gov
.25	BUSH		
.25	WELC		
.17	WHITE		
.16	U.S		
.15	HOUS		
.13	BUDG		
.13	PRESI		
.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

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

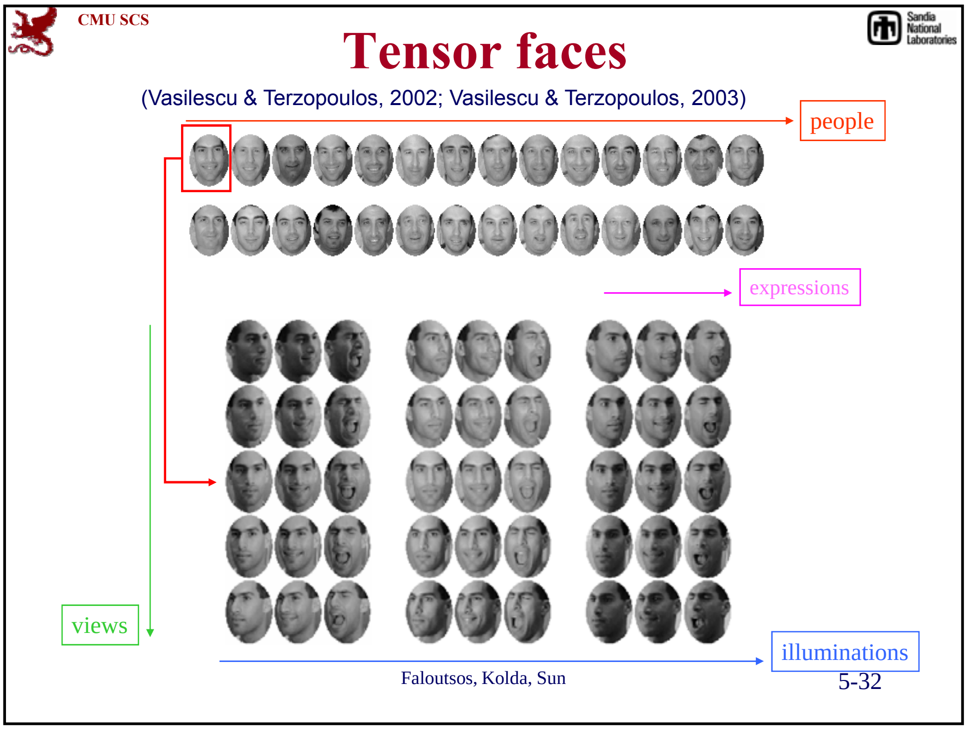
$$x_{ijk} = \begin{cases} \frac{1}{\log(w_k)+1} & \text{if } i \to j \text{ with term } k \\ 0 & \text{otherwise} \end{cases}$$

w_k = # unique links using term k

Tensor PARAFAC

term
to
from
=
term scores for 1st topic
authority scores for 1st topic
hub scores for 1st topic
+
term scores for 2nd topic
authority scores for 2nd topic
hub scores for 2nd topic
+ ...

Faloutsos, Kolda, Sun



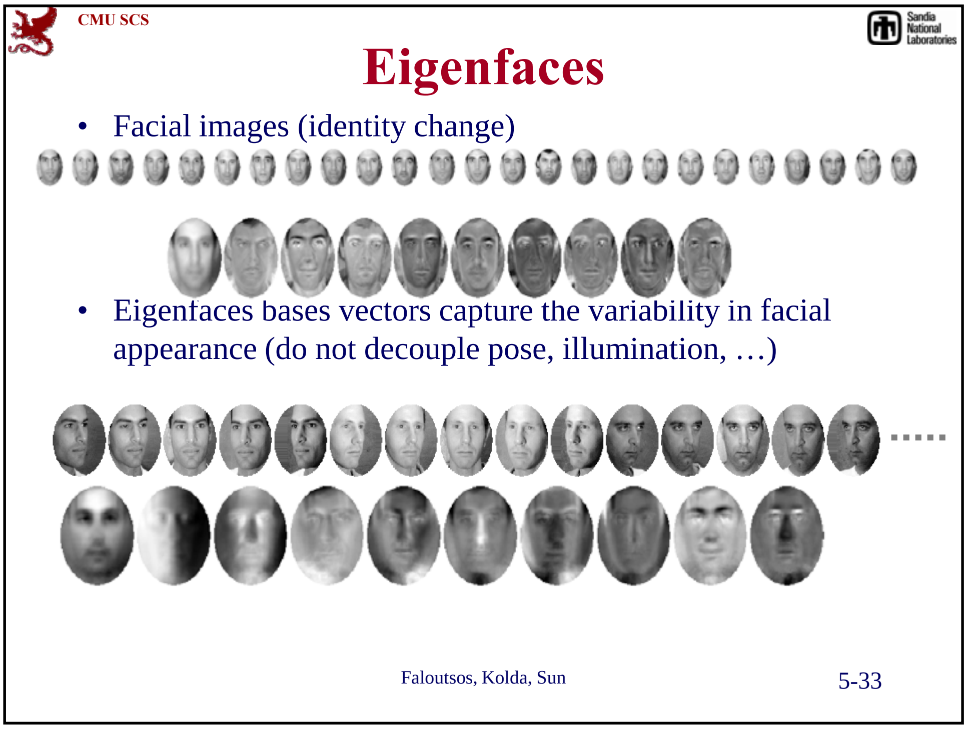
CMU SCS

Tensor faces

(Vasilescu & Terzopoulos, 2002; Vasilescu & Terzopoulos, 2003)

Faloutsos, Kolda, Sun

5-32



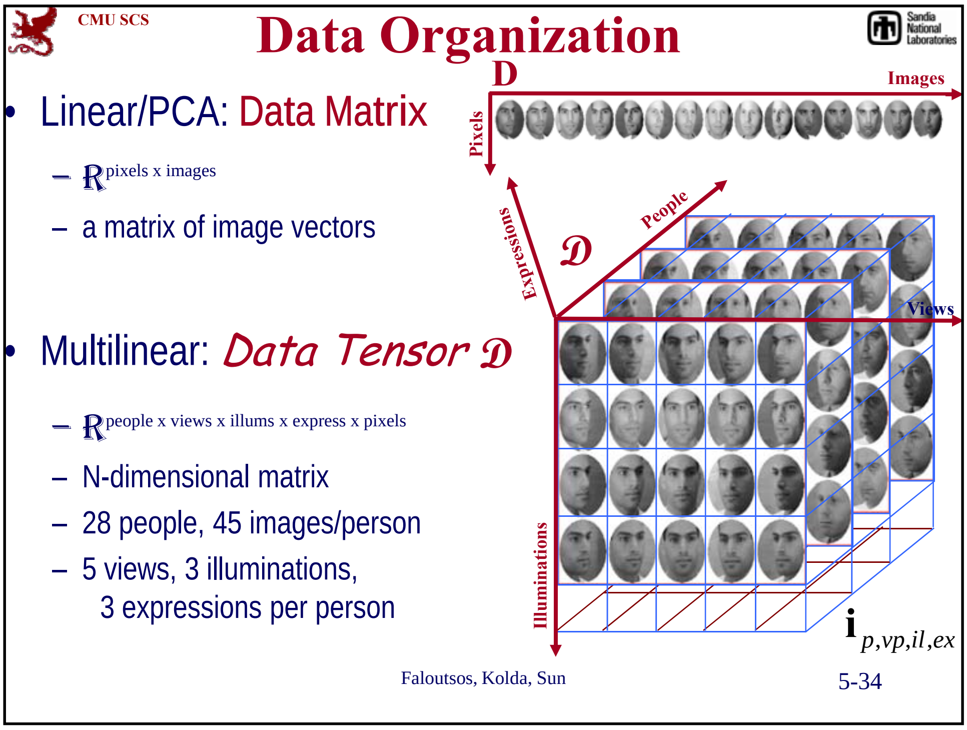
CMU SCS

Eigenfaces

- Facial images (identity change)
- Eigenfaces bases vectors capture the variability in facial appearance (do not decouple pose, illumination, …)

Faloutsos, Kolda, Sun

5-33



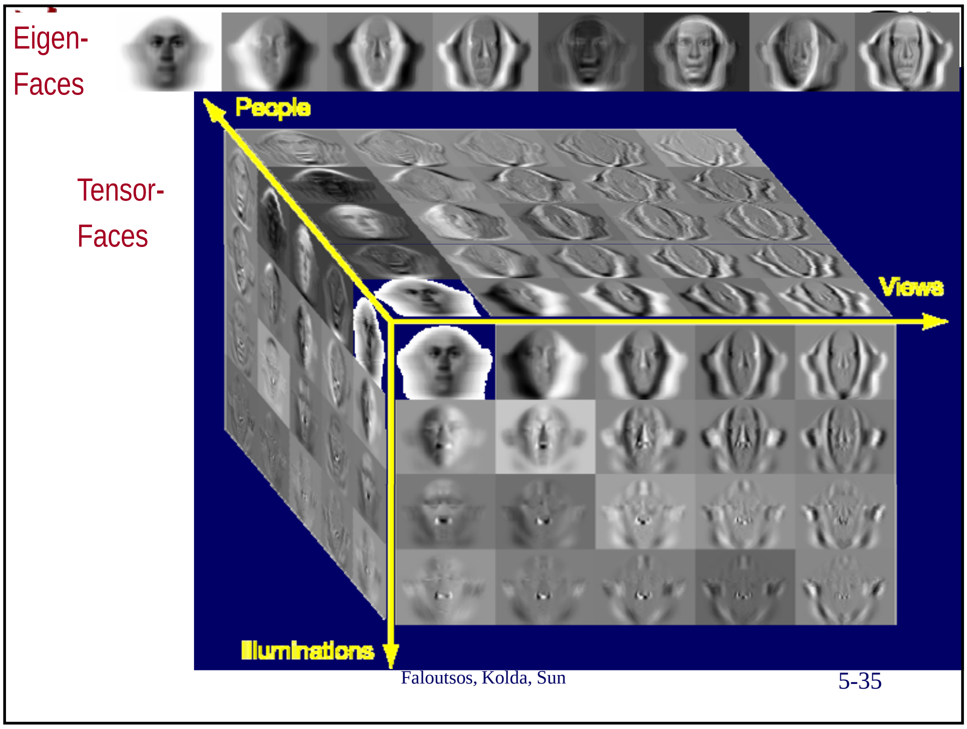
CMU SCS

Data Organization

- Linear/PCA: Data Matrix
  - $\mathbb{R}^{\text{pixels x images}}$
  - a matrix of image vectors
- Multilinear: *Data Tensor* $\mathcal{D}$
  - $\mathbb{R}^{\text{people x views x illums x express x pixels}}$
  - N-dimensional matrix
  - 28 people, 45 images/person
  - 5 views, 3 illuminations, 3 expressions per person

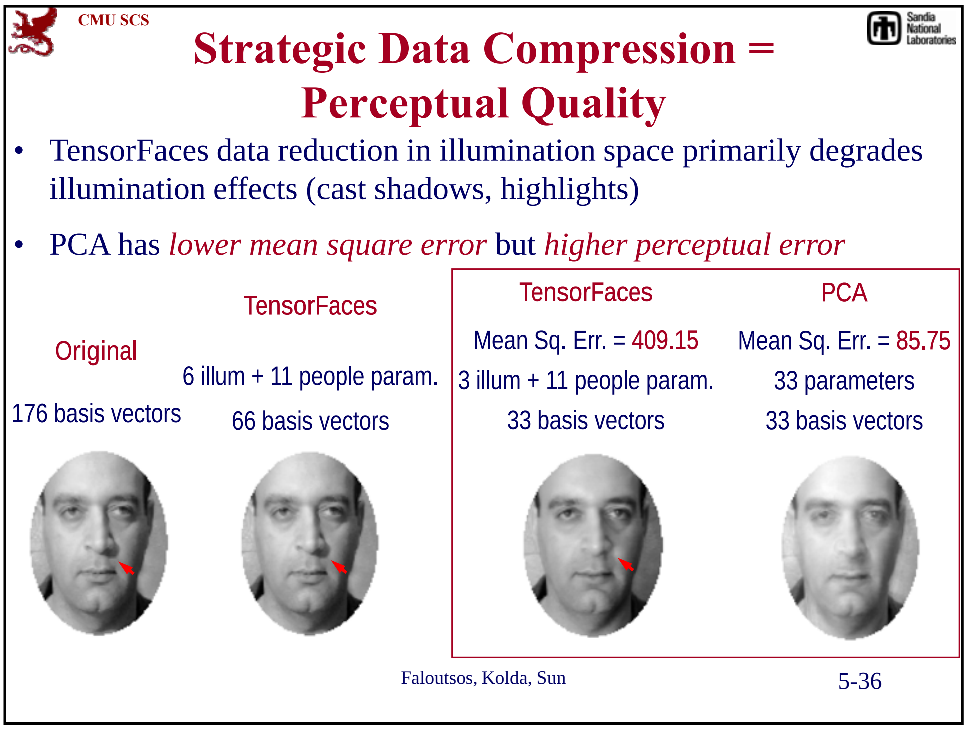
Faloutsos, Kolda, Sun

5-34



Eigen-Faces

Tensor-Faces



CMU SCS

Strategic Data Compression = Perceptual Quality

- TensorFaces data reduction in illumination space primarily degrades illumination effects (cast shadows, highlights)
- PCA has *lower mean square error* but *higher perceptual error*

Original	TensorFaces	TensorFaces	PCA
		Mean Sq. Err. = 409.15	Mean Sq. Err. = 85.75
	6 illum + 11 people param.	3 illum + 11 people param.	33 parameters
176 basis vectors	66 basis vectors	33 basis vectors	33 basis vectors

CMU SCS

TensorFaces: An Application of the Tucker Decomposition

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

- Example: 7942 pixels x 16 illuminations x 11 subjects
- PCA (eigenfaces): SVD of 7942 x 176 matrix
- Tensorfaces: Tucker decomposition of 7942 x 16 x 11 tensor

7942 x 33   176 x 33

$$\mathbf{X} \approx \mathbf{E} \times_2 \mathbf{V}$$

eigenfaces   loadings

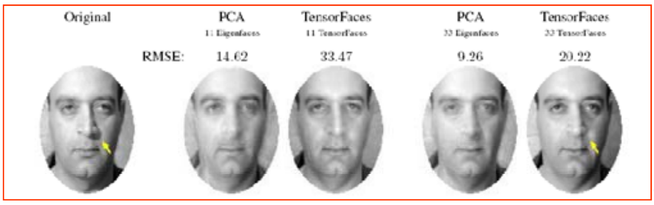
An image is represented by a linear combination of 33 eigenfaces.

7942 x 3 x 11   16 x 3   11 x 11

$$\mathcal{X} \approx \mathcal{T} \times_2 \mathbf{U}_{illum} \times_3 \mathbf{U}_{person}$$

tensorfaces   illumination   subjects

An image is represented by a multilinear combination of 33 tensorfaces using the outer product (or Kronecker product) of a length-3 illumination vector and a length-11 person vector.

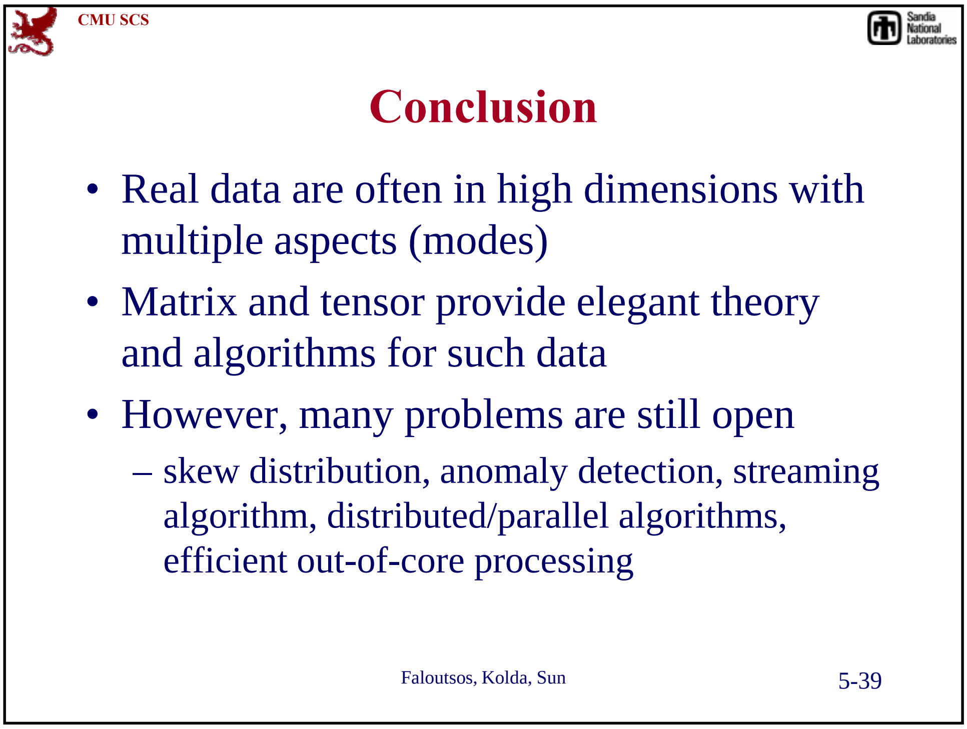


5-37

CMU SCS

Summary

Methods	Pros	Cons	Applications
SVD, PCA	Optimal in L2 and Frobenius	Dense representation, Negative entries	LSI, PageRank, HITS
CUR, CMD	Interpretability, sparse bases	Not optimal like SVD, dense core	DNA SNP data, network forensics
Co-clustering	Interpretability	Local minimum	Social networks, microarray data
Tucker	Flexible representation	Interpretability, non-uniqueness, dense core	TensorFaces
PARAFAC	Interpretability, efficient parse computation	Slow convergence	TOPHISTS
Incrementalization	Efficiency	Non-optimal	Tensor Streams
Nonnegativity	Interpretability, sparse results	Local minimum, non-uniqueness	Image segmentation

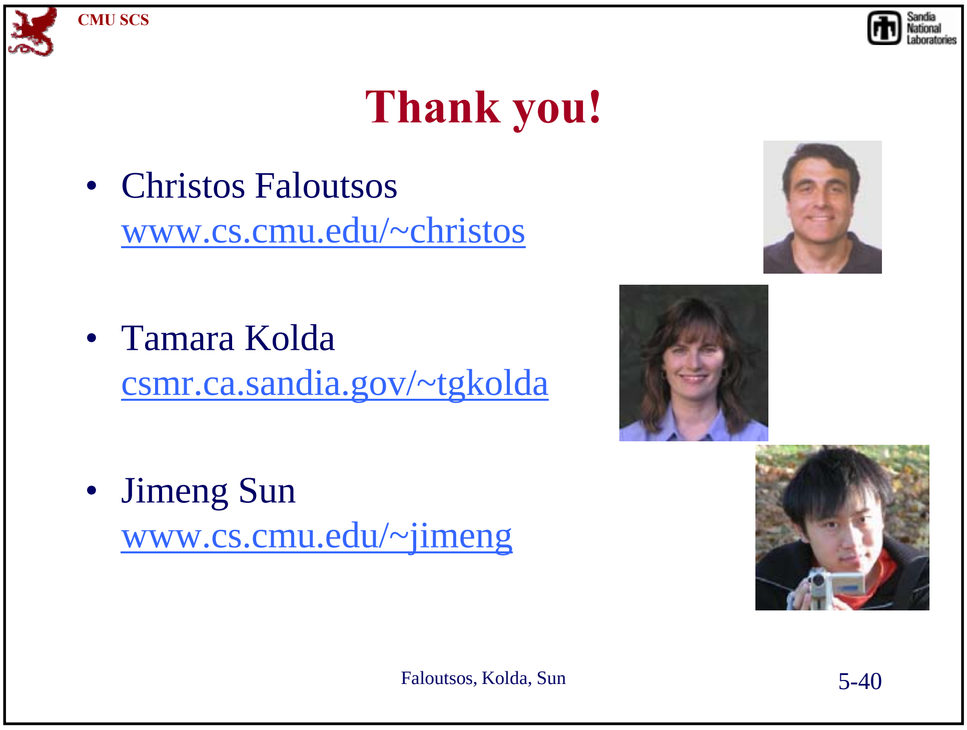


CMU SCS

Conclusion

- Real data are often in high dimensions with multiple aspects (modes)
- Matrix and tensor provide elegant theory and algorithms for such data
- However, many problems are still open
  - skew distribution, anomaly detection, streaming algorithm, distributed/parallel algorithms, efficient out-of-core processing

Faloutsos, Kolda, Sun 5-39



CMU SCS

Thank you!

- Christos Faloutsos
  www.cs.cmu.edu/~christos
- Tamara Kolda
  csmr.ca.sandia.gov/~tgkolda
- Jimeng Sun
  www.cs.cmu.edu/~jimeng

Faloutsos, Kolda, Sun 5-40