15-826: Multimedia Databases and Data Mining

Lecture #21: Tensor decompositions
C. Faloutsos

Must-read Material

Outline
Goal: ‘Find similar / interesting things’
- Intro to DB
- Indexing - similarity search
- Data Mining
Indexing - Detailed outline

- primary key indexing
- secondary key / multi-key indexing
- spatial access methods
- fractals
- text
- Singular Value Decomposition (SVD)
  - Tensors
- multimedia
- ...

Most of foils by

- Dr. Tamara Kolda (Sandia N.L.)
  - cmu.ca.sandia.gov/~tgkolda
- Dr. Jimeng Sun (CMU -> IBM)
  - www.cs.cmu.edu/~jimeng

3h tutorial: www.cs.cmu.edu/~christos/TALKS/SDM-tut-07/

Outline

- Motivation - Definitions
- Tensor tools
- Case studies
Motivation 0: Why “matrix”?

• Why matrices are important?

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

<table>
<thead>
<tr>
<th>milk</th>
<th>bread</th>
<th>choc.</th>
<th>wine</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>13</td>
<td>11</td>
<td>22</td>
</tr>
<tr>
<td>Peter</td>
<td>5</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Mary</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Nick</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Examples of Matrices:

Documents and terms

<table>
<thead>
<tr>
<th>data</th>
<th>mining</th>
<th>classif.</th>
<th>tree</th>
</tr>
</thead>
<tbody>
<tr>
<td>Paper#1</td>
<td>13</td>
<td>11</td>
<td>22</td>
</tr>
<tr>
<td>Paper#2</td>
<td>5</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Paper#3</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Paper#4</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Examples of Matrices:

Authors and terms

<table>
<thead>
<tr>
<th>data</th>
<th>mining</th>
<th>classif.</th>
<th>tree</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>13</td>
<td>11</td>
<td>22</td>
</tr>
<tr>
<td>Peter</td>
<td>5</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Mary</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Nick</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
**Examples of Matrices:**

sensor-ids and time-ticks

<table>
<thead>
<tr>
<th></th>
<th>temp1</th>
<th>temp2</th>
<th>humid</th>
<th>pressure</th>
</tr>
</thead>
<tbody>
<tr>
<td>t1</td>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55 ...</td>
</tr>
<tr>
<td>t2</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7 ...</td>
</tr>
<tr>
<td>t3</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>t4</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Motivation: Why tensors?**

- **Q:** what is a tensor?

**Motivation 2: Why tensor?**

- **A:** N-D generalization of matrix:
Motivation 2: Why tensor?

- A: N-D generalization of matrix:

<table>
<thead>
<tr>
<th>Mode#3</th>
<th>data</th>
<th>mining</th>
<th>classif.</th>
<th>tree</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>KDD'05</td>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55</td>
<td>...</td>
</tr>
<tr>
<td>KDD'06</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>...</td>
</tr>
<tr>
<td>KDD'07</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>John</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Peter</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Mary</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Nick</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Tensors are useful for 3 or more modes

Terminology: ‘mode’ (or ‘aspect’):

<table>
<thead>
<tr>
<th>Mode#2</th>
<th>data</th>
<th>mining</th>
<th>classif.</th>
<th>tree</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode#3</td>
<td>13</td>
<td>11</td>
<td>22</td>
<td>55</td>
<td>...</td>
</tr>
<tr>
<td>KDD'05</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>...</td>
</tr>
<tr>
<td>KDD'06</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>KDD'07</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>John</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Peter</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Mary</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>Nick</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Motivating Applications

- Why matrices are important?
- Why tensors are useful?
  - P1: environmental sensors
  - P2: social networks
  - P3: web mining

16

17

18
P1: Environmental sensor monitoring

Data in three aspects (time, location, type)

Temperature

Light

Humidity

Voltage

P2: 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

P3: 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
Outline

- Motivation – Definitions
- Tensor tools
- Case studies
  - Tensor Basics
  - Tucker
  - PARAFAC

Tensor Basics

Reminder: SVD

\[ A \approx U\Sigma V^T = \sum_i \sigma_i u_i \circ v_i \]

- Best rank-k approximation in L2

See also PARAFAC
Reminder: SVD

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

– Best rank-k approximation in L2

See also PARAFAC

Goal: extension to \(\geq 3\) modes

\[ \mathbf{X} \approx \left[ \lambda; \mathbf{A}_1, \mathbf{B}_1, \mathbf{C}_1 \right] = \sum_r \lambda_r \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \]

Main points:

- 2 major types of tensor decompositions: PARAFAC and Tucker
- both can be solved with "alternating least squares" (ALS)
- Details follow – we start with terminology:
A tensor is a multidimensional array

Matrization: Converting a Tensor to a Matrix

Tensor Mode-n Multiplication
Pictorial View of Mode-n Matrix Multiplication

Mode-1 multiplication (frontal slices)
\[ Y = X \times_1 A \]
\[ Y_{ij0} = X_{ij0} A^T \]

Mode-2 multiplication (lateral slices)
\[ Y = X \times_2 B \]
\[ Y_{ij0} = X_{ij0} B^T \]

Mode-3 multiplication (horizontal slices)
\[ Y = X \times_3 C \]
\[ Y_{ijk} = X_{ijk} C^T \]

Mode-n product Example

• Tensor times a matrix

Time Location

\[ X_{Time} \times \] Location

Time

\[ = \]

Clusters


• Tensor times a vector

Time Location

\[ X_{Time} \times \] Location

Time

\[ = \]

Clusters
Observe: For two vectors \(a\) and \(b\), \(a \circ b\) and \(a \otimes b\) have the same elements, but one is shaped into a matrix and the other into a vector.

Specially Structured Tensors

- **Tucker Tensor**
  \[
  X - \sum_{i,j,k}^{I,J,K} x_{ij,k} u_i v_j w_k = [u; v, w]
  \]

- **Kruskal Tensor**
  \[
  X - \sum_{i,j,k}^{I,J,K} x_{ij,k} s_i w_j = [s; u, v, w]
  \]
Specially Structured Tensors

- Tucker Tensor
  \[ X = \sum_{j=1}^{J} \sum_{k=1}^{K} \sum_{i=1}^{I} A_{ijk} V_i \otimes W_j \otimes U_k \]
  In matrix form:
  \[ \mathbf{X} = \mathbf{A} \otimes \mathbf{W} \otimes \mathbf{U} \]

- Kruskal Tensor
  \[ X = \sum_{j=1}^{J} \sum_{k=1}^{K} \sum_{i=1}^{I} B_{ijk} V_i \otimes W_j \otimes U_k \]
  In matrix form:
  \[ \mathbf{X} = \mathbf{B} \otimes \mathbf{W} \otimes \mathbf{U} \]

Tensor Decompositions

Tucker Decomposition - intuition

- author x keyword x conference
- A: author x author-group
- B: keyword x keyword-group
- C: conf. x conf-group
- S: how groups relate to each other
Intuition behind core tensor

- 2-d case: co-clustering
- [Dhillon et al. Information-Theoretic Co-clustering, KDD'03]
Tucker Decomposition

\[ \mathbf{X} \approx \langle \mathbf{G} ; \mathbf{A}, \mathbf{B}, \mathbf{C} \rangle \]

Given \( \mathbf{A}, \mathbf{B}, \mathbf{C} \), the optimal core is:

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

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

Recall the equations for converting a tensor to a matrix:

\[
\begin{align*}
\mathbf{X}_{(1)} &= \mathbf{G}_{(1)} \odot \mathbf{A}^T \\
\mathbf{X}_{(2)} &= \mathbf{G}_{(2)} \odot \mathbf{B}^T \\
\mathbf{X}_{(3)} &= \mathbf{G}_{(3)} \odot \mathbf{C}^T \\
\mathbf{X}_{(X)} &= (\mathbf{G} \odot \mathbf{A}) \mathbf{B}^T \\
\end{align*}
\]

Solving for Tucker

\[ \mathbf{X} \approx \langle \mathbf{G} ; \mathbf{A}, \mathbf{B}, \mathbf{C} \rangle \]

Given \( \mathbf{A}, \mathbf{B}, \mathbf{C} \) orthonormal, the optimal core is:

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

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

\[ \| \mathbf{X} - \langle \mathbf{G} ; \mathbf{A}, \mathbf{B}, \mathbf{C} \rangle \|^2 = \| \mathbf{X} \|^2 - 2 \langle \mathbf{G} ; \mathbf{A}, \mathbf{B}, \mathbf{C} \rangle + \| \mathbf{G} \|^2 \]

Eliminate the core to get:

\[ \| \mathbf{X} \|^2 - 2 \langle \mathbf{X} ; \mathbf{A}, \mathbf{B}, \mathbf{C} \rangle + \| \mathbf{G} \|^2 = 0 \]

Minimize s.t. \( \mathbf{A}, \mathbf{B}, \mathbf{C} \) orthonormal

If \( \mathbf{B} \) and \( \mathbf{C} \) are fixed, then we can solve for \( \mathbf{A} \) as follows:

\[ \| \mathbf{X} ; \mathbf{A}^T, \mathbf{B}^T, \mathbf{C}^T \| = \| \mathbf{A}^T \mathbf{X} (\mathbf{C} \odot \mathbf{B}) \| \]

Optimal \( \mathbf{A} \) is \( R \) left leading singular vectors for \( \mathbf{X}_{(1)} \mathbf{C} \odot \mathbf{B} \)
Higher Order SVD (HO-SVD)

\[ A = \text{leading } R \text{ left singular vectors of } X^{(1)} \]
\[ B = \text{leading } S \text{ left singular vectors of } X^{(2)} \]
\[ C = \text{leading } T \text{ left singular vectors of } X^{(3)} \]
\[ g = [x; A^T, B^T, C^T] \]

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

Tucker-Alternating Least Squares (ALS)

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

\[ \begin{align*}
\text{Initialize} \\
&\quad \text{Choose } R, S, T \\
&\quad \text{Calculate } A, B, C \text{ via HO-SVD} \\
\text{Until converged do} \\
&\quad \text{A = R leading left singular vectors of } X^{(1)}(C \otimes B) \\
&\quad \text{B = S leading left singular vectors of } X^{(2)}(C \otimes A) \\
&\quad \text{C = T leading left singular vectors of } X^{(3)}(B \otimes A) \\
&\quad \text{Solve for core:} \\
&\quad g = [x; A^T, B^T, C^T] \\
\end{align*} \]

Kroonenberg & De Leeuw, Psychometrika, 1980

Tucker in Not Unique

Tucker decomposition is not unique. Let \( Y \) be an \( R \times R \) orthogonal matrix. Then...

\[ X = g \times_1 A \times_2 B \times_3 C = ([1 \times_1 Y^T] \times_1 (A^T X) \times_2 B \times_3 C) \]
\[ X_{0Y} = AC(C \otimes D)^T - AYY^TC(C \otimes D)^T \]

Kroonenberg & De Leeuw, Psychometrika, 1980
Outline

- Motivation – Definitions
- Tensor tools
- Case studies

Tensor tools

- Tensor Basics
- Tucker
- PARAFAC

Definitions

- PARAFAC
- CANDECOMP
- CANDECOMP/PARAFAC

Properties

- CANDECOMP = Canonical Decomposition (Carroll & Chang, 1970)
- PARAFAC = Parallel Factors (Harshman, 1970)
- Core is diagonal (specified by the vector $\lambda$)
- 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 $\text{rank}(\mathbf{X}) > \min\{I, J, K\}$

Decomposition

\[
\mathbf{X} = \sum_{r} \lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r
\]

- CANDECOMP/PARAFAC

PARAFAC Alternating Least Squares (ALS)

Succesively solve for each component ($A, B, C$).

\[
\mathbf{X} = \sum_{r} \lambda_r \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r
\]

- CANDECOMP
- Column-wise Kronecker product

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

\[
A = \mathbf{X}_{ij} (C \otimes B) (C^T \otimes B^T) \mathbf{A}^{-1}
\]

Repeat for $B, C$, etc.
PARAFAC is often unique

\[
\mathbf{X} = [\mathbf{A} \odot \mathbf{B} \odot \mathbf{C}] = \sum_{r=1}^{R} \mathbf{a}_r \mathbf{b}_r \mathbf{c}_r
\]

Sufficient condition for uniqueness (Kruskal, 1977):

\[
2k + 2 \leq k_A + k_B + k_C
\]

\(k_A = k\)-rank of \(\mathbf{A}\) = max number \(k\) such that every set of \(k\) columns of \(\mathbf{A}\) is linearly independent

---

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

---

Tensor tools - summary

- Two main tools
  - PARAFAC
  - Tucker

- Both find row-, column-, tube-groups
  - but in PARAFAC the three groups are identical

- To solve: Alternating Least Squares

- Toolbox: from Tamara Kolda:
  http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/
Outline

- Motivation - Definitions
- Tensor tools
- Case studies
  - Sensors
  - Social networks
  - Web mining

P1: Environmental sensor monitoring

- Temperature
- Humidity
- Voltage
- Light

1st factor
- Daily periodicity on time
- Uniform on all locations
- Temp, Light and Volt are positively correlated while negatively correlated with Humid

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.

P2: Social network analysis

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

<table>
<thead>
<tr>
<th>Authors</th>
<th>Keywords</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hector Garcia-Molina</td>
<td>Database, storage, network, query</td>
<td>1995</td>
</tr>
<tr>
<td>Philip Yu</td>
<td>Database, storage, query</td>
<td>2004</td>
</tr>
<tr>
<td>Michael Stonebreaker</td>
<td>Database, storage, query</td>
<td>2004</td>
</tr>
</tbody>
</table>

- Two groups are correctly identified: Databases and Data mining
- People and concepts are drifting over time
### P3: Web graph mining

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

### Kleinberg’s Hubs and Authorities (the HITS method)

![Sparse adjacency matrix and its SVD](https://example.com/sparse_matrix.png)

- Authority scores for 1st topic
- Authority scores for 2nd topic
- Hub scores for 1st topic
- Hub scores for 2nd topic

Kleinberg, JACM, 1999

### HITS Authorities on Sample Data

<table>
<thead>
<tr>
<th>Page 1 Authority</th>
<th>Page 2 Authority</th>
<th>Page 3 Authority</th>
</tr>
</thead>
<tbody>
<tr>
<td>site1</td>
<td>site2</td>
<td>site3</td>
</tr>
<tr>
<td>site4</td>
<td>site5</td>
<td>site6</td>
</tr>
<tr>
<td>site7</td>
<td>site8</td>
<td>site9</td>
</tr>
</tbody>
</table>

We started our crawl from http://www-news.mcs.anl.gov/heise, and crawled 4700 pages, resulting in 560 cross-linked hosts.
Observe that this tensor is very sparse!

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

$$X = \sum_{j=1}^{R} \lambda_j x_j \circ x_j$$

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

$$X = \sum_{n=1}^{R} \mu_n b_n \circ a_n$$
## Summary

<table>
<thead>
<tr>
<th>Methods</th>
<th>Pros</th>
<th>Cons</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVD, PCA</td>
<td>Optimal in L2 and Frobenius norm</td>
<td>Dense representation, Negative entries</td>
<td>LSI, PageRank, HITS</td>
</tr>
<tr>
<td>Co-clustering</td>
<td>Interpretability</td>
<td>Local minimum</td>
<td>Social networks, microarray data</td>
</tr>
<tr>
<td>Tucker</td>
<td>Flexible representation</td>
<td>Interpretability, non-uniqueness, dense core</td>
<td>TensorFaces</td>
</tr>
<tr>
<td>PARAFAC</td>
<td>Interpretability, efficient parse computation</td>
<td>Slow convergence</td>
<td>TPHISTS</td>
</tr>
</tbody>
</table>

## Conclusions

- Real data are often in high dimensions with multiple aspects (modes)
- Matrices and tensors provide elegant theory and algorithms
References

• Inderjit S. Dhillon, Subramanyam Mallela, Dharmendra S. Modha: Information-theoretic co-clustering. KDD 2003: 89-98


• Jimeng Sun, Spiros Papadimitriou, Philip Yu. Window-based Tensor Analysis on High-dimensional and Multi-aspect Streams, Proc. of the Int. Conf. on Data Mining (ICDM), Hong Kong, China, Dec 2006