15-826: Multimedia Databases and Data Mining

Lecture #19: SVD – part II – applications

C. Faloutsos

Must-read Material

- MM Textbook Appendix D

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)
- multimedia
- ...
**SVD - Detailed outline**

- Motivation
- Definition - properties
- Interpretation
- Complexity
- Case studies
- SVD properties
- Conclusions

**SVD - Case studies**

- multi-lingual IR; LSI queries
- compression
- PCA - ‘ratio rules’
- Karhunen-Lowe transform
- query feedbacks
- google/Kleinberg algorithms

**Case study - LSI**

Q1: How to do queries with LSI?
Q2: multi-lingual IR (english query, on spanish text?)

**Case study - LSI**

Q1: How to do queries with LSI?
Problem: Eg., find documents with ‘data’

\[
\begin{bmatrix}
\text{data} \\
\text{CS} \\
\text{MD}
\end{bmatrix}
= \begin{bmatrix}
0.18 & 0.36 & 0.18 & 0.90 \\
0.18 & 0.18 & 0.36 & 0.53 \\
0.90 & 0.53 & 0.80 & 0.27 \\
0.53 & 0.36 & 0.53 & 0.80
\end{bmatrix}
\begin{bmatrix}
9.64 \\
0 \\
0
\end{bmatrix}
\]

\[
= \begin{bmatrix}
0.58 & 0.58 & 0.58 & 0 \\
0.58 & 0.58 & 0.58 & 0.71 \\
0.58 & 0.58 & 0.58 & 0.71
\end{bmatrix}
\]
Case study - LSI

Q1: How to do queries with LSI?
A: map query vectors into ‘concept space’ – how?

A: inner product (cosine similarity) with each ‘concept’ vector $v_i$
Case study - LSI

compactly, we have:

\[ q^T V = q_{\text{concept}} \]

Eg:

\[
\begin{bmatrix}
data & \text{inf} & \text{retrieval} & \text{brain} & \text{lung} \\
0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]

\[
\begin{bmatrix}
0.58 & 0 \\
0.58 & 0 \\
0.58 & 0 \\
0 & 0.71 \\
0 & 0.71
\end{bmatrix}
\]

\[ \text{CS-concept} = \begin{bmatrix} 1.16 & 0 \end{bmatrix} \]

Drill: how would the document (‘information’, ‘retrieval’) be handled by LSI?

Observation: document (‘information’, ‘retrieval’) will be retrieved by query (‘data’), although it does not contain ‘data’!!

\[
\begin{bmatrix}
data & \text{inf} & \text{retrieval} & \text{brain} & \text{lung} \\
0 & 1 & 1 & 0 & 0
\end{bmatrix}
\]

\[
\begin{bmatrix}
0.58 & 0 \\
0 & 0.71 \\
0 & 0.71
\end{bmatrix}
\]

\[ \text{CS-concept} = \begin{bmatrix} 0.58 & 0 \end{bmatrix} \]
Case study - LSI

Q1: How to do queries with LSI?

Q2: multi-lingual IR (english query, on spanish text?)

• Problem:
  – given many documents, translated to both languages (eg., English and Spanish)
  – answer queries across languages

Solution: ~ LSI

data retrieval informacion
lung

SVD - Case studies

• multi-lingual IR; LSI queries
• compression
• PCA - ‘ratio rules’ & visualization
• Karhunen-Lowe transform
• query feedbacks
• google/Kleinberg algorithms
Case study: compression

[Korn+97]

Problem:
- given a matrix
- compress it, but maintain ‘random access’
(surprisingly, its solution leads to data mining and visualization...)

Problem - specs
- ~10**6 rows; ~10**3 columns; no updates;
- random access to any cell(s); small error: OK

<table>
<thead>
<tr>
<th>customer</th>
<th>We</th>
<th>Th</th>
<th>Fr</th>
<th>Sa</th>
<th>Su</th>
<th>Su</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABC Inc.</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DEF Ltd.</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>GHI Inc.</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>KLM Co.</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Smiths</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Johnson</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Thompson</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Idea

SVD - reminder
- space savings: 2:1
- minimum RMS error

first singular vector
Case study: compression

outliers?
A: treat separately
(SVD with ‘Deltas’)

first singular
vector

day 1

Compression - Performance

• 3 pass algo (→ scalability) (HOW?)
• random cell(s) reconstruction
• 10:1 compression with < 2% error

Performance - scaleup

e-space
time

Compression - Visualization

• no Gaussian clusters; Zipf-like distribution
SVD - Case studies

• multi-lingual IR; LSI queries
• compression
• PCA - ‘ratio rules’
• Karhunen-Lowe transform
• query feedbacks
• google/Kleinberg algorithms

PCA - ‘Ratio Rules’

[Korn+00]
Typically: ‘Association Rules’ (eg.,
{bread, milk} -> {butter}
But, can we discover more details? like:
$-bread : $-milk : $-butter ~ $2 : $4 : $3

PCA - ‘Ratio Rules’

Idea: try to find ‘concepts’:
• singular vectors dictate rules about ratios:
bread:milk:butter = 2:4:3

\begin{align*}
    \text{bread} : \text{milk} : \text{butter} &= 2 : 4 : 3 \\
    \$ \text{on butter} &\rightarrow \$ \text{on milk} \\
    &\rightarrow \$ \text{on bread}
\end{align*}

PCA - ‘Ratio Rules’

Identical to PCA = Principal Components Analysis
✓ – Q1: which set of rules is ‘better’?
✓ – Q2: how to reconstruct missing/corrupted values?
✓ – Q3: is there need for binary/bucketized values? NO
✓ – Q4: how to interpret the rules (= ‘principal components’)?
NBA dataset
~500 players;
~30 attributes

PCA: get singular vectors \( v_1, v_2, \ldots \)

ignore entries with small abs. value

try to interpret the rest

NBA dataset - \( V \) matrix (term to ‘concept’ similarities)

<table>
<thead>
<tr>
<th>field</th>
<th>( P_{R_1} )</th>
<th>( P_{R_2} )</th>
<th>( P_{R_3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>minutes played</td>
<td>.808</td>
<td>-.4</td>
<td></td>
</tr>
<tr>
<td>field goals</td>
<td>.106</td>
<td>.190</td>
<td></td>
</tr>
<tr>
<td>goal attempts</td>
<td></td>
<td>.459</td>
<td>.602</td>
</tr>
<tr>
<td>points</td>
<td></td>
<td></td>
<td>-.07</td>
</tr>
<tr>
<td>total rebounds</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>assists</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>steals</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Ratio Rules - example

RR1: minutes:points = 2:1

corresponding concept?

v1
Ratio Rules - example

- RR1: minutes:points = 2:1
- corresponding concept?
- A: ‘goodness’ of player

<table>
<thead>
<tr>
<th>field</th>
<th>RR2</th>
<th>RR3</th>
<th>RR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>minutes played</td>
<td>.808</td>
<td>- .4</td>
<td></td>
</tr>
<tr>
<td>field goals</td>
<td>.406</td>
<td>.199</td>
<td></td>
</tr>
<tr>
<td>goal attempts</td>
<td>.406</td>
<td>.199</td>
<td>.602</td>
</tr>
<tr>
<td>points</td>
<td>.406</td>
<td>.199</td>
<td>.602</td>
</tr>
<tr>
<td>total rebounds</td>
<td>- .489</td>
<td>- .486</td>
<td>- .486</td>
</tr>
<tr>
<td>assists</td>
<td>- .486</td>
<td>- .486</td>
<td>- .486</td>
</tr>
<tr>
<td>steals</td>
<td>- .486</td>
<td>- .486</td>
<td>- .486</td>
</tr>
</tbody>
</table>
Ratio Rules - example

• RR2: points: rebounds negatively correlated(!) - concept?

A subtle point:
SVD will give vectors that go through the origin

SVD - Case studies

• multi-lingual IR; LSI queries
• compression
• PCA - ‘ratio rules’ & visualization
• Karhunen-Lowe transform
• query feedbacks
• google/Kleinberg algorithms

K-L transform

[Duda & Hart]; [Fukunaga]

A subtle point:
SVD will give vectors that go through the origin
K-L transform

A subtle point: SVD will give vectors that go through the origin
Q: how to find $v_1'$?

A: ‘centered’ PCA, ie., move the origin to center of gravity

• How to ‘center’ a set of vectors (= data matrix)?
• What is the covariance matrix?
• A: see textbook
• (‘whitening transformation’)

A: see textbook
Conclusions

• SVD: popular for dimensionality reduction / compression
• SVD is the ‘engine under the hood’ for PCA (principal component analysis)
• ... as well as the Karhunen-Lowe transform
• (and there is more to come ...)

References


References