DYSCO course on low-rank approximation and its applications

Introduction

Ivan Markovsky

Vrije Universiteit Brussel
Outline

About the course

Historical review

Applications

Demos

Exercises
Outline

About the course

Historical review

Applications

Demos

Exercises
Subject areas

"You can learn only what you have already half known."  R. Vaccaro

- numerical linear algebra
  - (generalized) least norm and least squares
  - structured (Hankel/Toeplitz) matrices
  - variable projections method

- optimization
  - penalty methods for nonlinear optimization
  - optimization on a manifold
  - convex relaxations
- statistics
  - errors-in-variables models
  - maximum likelihood
  - bias correction

- dynamical system
  - realization theory, system identification
  - behavioral approach

- computer algebra
  - approximate common divisors
  - polynomial factorizations

- computer vision
  - image deblurring (blind deconvolution)
  - image compression
"If you try to say everything, you end up saying nothing."

P. Stewart

- **Aim**
  - main goal: recognize and exploit common features, methods, and algorithms across different applications
  - low-rank approx. is a unifying problem; related to
    - total least squares (numerical linear algebra)
    - principal component analysis (statistics)
    - factor analysis (psychometrics)
    - latent semantic analysis (natural language proc.)
    - ...
Plan

1. Introduction (this lecture)
2. Computational tools (QR, SVD, LS, TLS)
3. Behavioral approach (TLS $\rightarrow$ LRA)
4. System identification (modeling from data)
5. Subspace methods (exact modeling)
6. Generalizations (missing data, ...)
Exercises and evaluation

"I hear, I forget;
I see, I remember;
I do, I understand." Chinese philosopher

- analytic/computer exercises are part of the course
  - bring a laptop
  - try all problems

- need evaluation? (contact me in the break)
  - work on an individual project, related to the course and feasible to complete in two weeks
  - submit < 10 pages report by 21 March and give a 10-minutes presentation on 21 March
Materials

- books

- lecture slides available from after the lectures
  
  http://homepages.vub.ac.be/~imarkovs/dysco

- references to the literature
Outline

About the course

Historical review

Applications

Demos

Exercises
"There are repeated patterns in the history of science that teach us how to overcome modern problems. Those who are not aware of the history are missing much."  

P. Stewart

Eugenio Beltrami (1835–1900)

- considered bilinear forms: \( f(x, y) = x^\top Ay \)

- **problem:** represent \( f \) as a sum of squares via orthogonal transformations \( \tilde{x} = U^\top x \) and \( \tilde{y} = V^\top y \), i.e.,

  \[
  f(x, y) = x^\top Ay = \tilde{x}^\top U^\top AV\tilde{y} = \tilde{x}^\top \Sigma \tilde{y}, \text{ with } \Sigma \text{ diagonal}
  \]

- equivalent problem is

  \[
  A = U\Sigma V^\top, \text{ with } U, \ V \text{ orthogonal and } \Sigma \text{ diagonal}
  \]

  \( \leadsto \) singular value decomposition \( \approx \) low-rank approx.
Camille Jordan (1838–1922)

- problem:

  maximize $x^\top Ay$ subject to $x^\top x = 1$ and $y^\top y = 1$

- the solution is given by the extreme singular values and corresponding singular vectors of $A$

- generalization to infinite dimensional spaces (integrals rather than sums)

- "the fundamental theorem"

\[
\text{For any } \hat{A} \text{ with } \text{rank}(\hat{A}) \leq r, \|A - \hat{A}\|_2 \geq \sigma_{r+1}(A).
\]
principal component analysis problem (1933)

If $x$ is a random vector with zero mean and dispersion $D$, with eigenvalue decomposition $D = V \Sigma^2 V^\top$, the components of $V^\top x$ are uncorrelated with variances $\sigma_i^2$. Then the $\hat{V}$ factor, obtained from the SVD of $X$, is an estimate of $V$. 
G. Golub (1932–2007) and W. Kahan (1933–)

- computation of the SVD by a two step procedure:
  1. reduction to a bidiagonal matrix (in $O(mn^2)$ for $m > n$)
  2. compute the SVD of the bidiagonal matrix (by a variant of the QR algorithm for EVD)

- step 2 requires iterative algorithms

- convergence to machine precision is fast

- in fact, the first step is more expensive
Rudolf Kalman (1930–) and others

- realization theory

\[
\text{rank(Hankel matrix)} = \text{order of a minimal realization}
\]


- nuclear norm heuristic for rank minimization problems
Outline

About the course

Historical review

Applications

Demos

Exercises
"For applicable control engineering research, three things need to be present:

1. a real and pressing set of problems,
2. intuitively graspable theoretical approaches to design, which can be underpinned by sound mathematics, and
3. good interactive software which can be used to turn designs into practical applications."

A. MacFarlane


- next set of problems (Section 1.3 of the book)
Applications

1. Direction of arrival estimation (signal processing)
2. Latent semantic analysis (language processing)
3. Recommender systems (machine learning)
4. Multidimensional scaling (computer vision)
5. Conic section fitting (computer vision)
6. System realization (systems and control)
7. System identification (systems and control)
8. Greatest common divisor (computer algebra)
Direction of arrival estimation

- setup: \( q \) antennas and \( m < q \) distant sources

- \( \ell_k \) — source intensity (a function of time)

- \( w(t) = p_k \ell_k(t - \tau_k) \) — array's response to \( k \)th source

- \( \tau_k \) — pure delay
$p_k$ depends on the array geometry and the source locations (assumed constant in time)

assuming that the array responds linearly to a mixture of sources, we have

\[
D = \begin{bmatrix} w(1) & \cdots & w(T) \end{bmatrix}
\]

\[
= \sum_{k=1}^{m} p_k \begin{bmatrix} \ell_k(1 - \tau_k) & \cdots & \ell_k(T - \tau_k) \end{bmatrix} = PL
\]

where $P := \begin{bmatrix} p_1 & \cdots & p_m \end{bmatrix}$ and $L := \begin{bmatrix} \ell_1 \\ \vdots \\ \ell_m \end{bmatrix}$

$\text{rank}(D) = \# \text{ of sources}$
Computational problem

- with exact data $D$, the direction of arrival problem is
  
  \emph{rank revealing factorization} $PL$ of $D$

- $P, L$ carry information about the source locations

- in practice, $D$ is full rank and we aim to
  
  approximate $D$ by $\hat{D}$ of rank $\leq m < \max(q, N)$

- this is
  
  \emph{unstructured low-rank approximation problem}
the rank constraint $m$ is a hyper parameter

determining its value is part of the problem

from $\hat{D}$, we need to obtain $P, L$, such that $\hat{D} = PL$

this is the (simple) problem of exact modeling (rank-revealing factorization)

some algorithms return $P, L$ as a byproduct

we separate the issues of

1. solution methods (optimization algorithms)
2. problem formulation (low-rank approximation)
Latent semantic analysis

- $N$ documents involve $q$ terms and $m$ concepts

- $p_k$ — term frequencies related to the $k$th concept

- $\ell_{kj}$ — relevance of the $k$th concept to the $j$th document

- the term frequencies related to the documents are

$$D = \sum_{k=1}^{m} p_k \begin{bmatrix} \ell_{k1} & \cdots & \ell_{kN} \end{bmatrix} = [p_1 \cdots p_m] \begin{bmatrix} \ell_1 \\ \vdots \\ \ell_m \end{bmatrix} = PL$$

- $\text{rank}(D) = \# \text{ of concepts}$
Recommender systems

- $q$ items are rated by $N$ users

- $d_{ij}$ — rating of the $i$th item by the $j$th user

- not all ratings are available $\leadsto$ missing data in $D$

- assumption: $m$ “typical” users, where $m \ll \min(q, N)$

- $p_k$ — ratings of the items by the $k$th typical user
the $j$th user is a linear combination of typical users

$$d_j = \sum_{k=1}^{m} p_k \ell_{kj}$$

$$\ell_k := [\ell_{k1} \cdots \ell_{kN}]$$ — weights for the $j$th user

model for the ratings

$$D = \sum_{k=1}^{m} p_k \ell_k = PL$$

rank($D$) = number of “typical” users
Matrix completion problems

- exact matrix completion

  minimize over $\hat{D}$ $\text{rank}(\hat{D})$
  subject to $\hat{D}_{ij} = D_{ij}$ for all $(i,j)$, where $D_{ij}$ is given

- approximate matrix completion

  minimize over $\hat{D}$ and $\Delta D$ $\text{rank}(\hat{D}) + \lambda \|\Delta D\|_F$
  subject to $\hat{D}_{ij} = D_{ij} + \Delta D_{ij}$ for all $(i,j)$, where $D_{ij}$ is given
Multidimensional scaling

- consider $N$ points: $\mathcal{X} := \{x_1, \ldots, x_N\} \subset \mathbb{R}^2$

- $d_{ij} := \|x_i - x_j\|_2^2$ — squared distance from $x_i$ to $x_j$

- distance matrix: $D = [d_{ij}]$ of the pair-wise distances

- rank($D$) $\leq$ 4, indeed

$$d_{ij} = (x_i - x_j)^\top (x_i - x_j) = x_i^\top x_i - 2x_i^\top x_j + x_j^\top x_j$$
\[ d_{ij} = (x_i - x_j)^\top (x_i - x_j) = x_i^\top x_i - 2x_i^\top x_j + x_j^\top x_j \]

\[ D = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \begin{bmatrix} x_1^\top x_1 & \cdots & x_N^\top x_N \end{bmatrix} - 2 \begin{bmatrix} x_1^\top \\ \vdots \\ x_N^\top \end{bmatrix} \begin{bmatrix} x_1 & \cdots & x_N \end{bmatrix} \]

\[ \text{rank} \leq 1 \quad \text{and} \quad \text{rank} \leq 2 \]

\[ + \begin{bmatrix} x_1^\top x_1 \\ \vdots \\ x_N^\top x_N \end{bmatrix} \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix} \]

\[ \text{rank} \leq 1 \]

- approximate modeling:

\textit{bilinearly structured low-rank approximation}
Conic section fitting

- **data:**
  
  \[ \{ d_1, \ldots, d_N \} \subset \mathbb{R}^2, \quad \text{where} \quad d_j = \begin{bmatrix} a_j \\ b_j \end{bmatrix} \]

- **model:**
  
  \[ \mathcal{B}(S, u, v) := \{ d \in \mathbb{R}^2 \mid d^\top S d + u^\top d + v = 0 \} \]

- **linear relation in the model parameters**

  \[ d^\top S d + u^\top d + v = \begin{bmatrix} s_{11} & 2s_{12} & u_1 & s_{22} & u_2 & v \end{bmatrix} \begin{bmatrix} a^2 \\ ab \\ a \\ b^2 \\ b \\ 1 \end{bmatrix} \]
- parameter vector

\[ \theta := [s_{11} \quad 2s_{12} \quad u_1 \quad s_{22} \quad u_2 \quad v] \]

- extended data vector (feature map)

\[ d_{\text{ext}} := [a^2 \quad ab \quad a \quad b^2 \quad b \quad 1]^\top \]

- exact modeling

\[ d \in \mathcal{B}(\theta) = \mathcal{B}(S, u, v) \iff \theta d_{\text{ext}} = 0 \]

- approximate modeling:

quadratically structured low-rank approximation
System realization

- **problem:**
  
  \[ \text{impulse response} \rightarrow \text{state space representation} \]

- let \( H \) be an impulse response of \( n \)th order discrete-time linear time-invariant system

- then

  \[
  \begin{bmatrix}
  H(1) & H(2) & H(3) & \cdots \\
  H(2) & H(3) & \ddots & \\
  H(3) & \ddots & & \\
  \vdots & & & \\
  \end{bmatrix}
  \]

  \[ \text{rank } H(H) = n \]

- partial realization problem
Stochastic realization

- stochastic system: white noise $\rightarrow$ deterministic system $\rightarrow$ $y$

- data: $R(\tau) := \mathbb{E}(y(t)y^\top(t-\tau))$ autocorrelation

- problem:
  
  autocorrelation $R \mapsto$ state space representation

- main result:
  
  $\text{rank}(\mathcal{H}(R)) = \text{order of minimal realization of } R$
System identification

- **problem:**

  general trajectory $\leftrightarrow$ representation of the system

- **data:**

  \[
  w = \begin{bmatrix} u \\ y \end{bmatrix}, \quad u = (u(1), \ldots, u(T)) \quad \text{— input}
  \\
  y = (y(1), \ldots, y(T)) \quad \text{— output}
  \]

- **link to low-rank approximation**

  \[
  \text{rank}(\mathcal{H}_{n_{\text{max}}+1}(w)) \leq \text{rank}(\mathcal{H}_{n_{\text{max}}+1}(u)) + \text{order of the system}
  \]

- **persistency of excitation:** $\mathcal{H}(u)$ is full row rank
Greatest common divisor

- the GCD of the polynomials

\[
p(z) = p_0 + p_1 z + \cdots + p_n z^n \\
q(z) = q_0 + q_1 z + \cdots + q_m z^m
\]

is polynomial \( c \) of maximal degree dividing \( p \) and \( q \)

\[
p = rc \quad \text{and} \quad q = sc
\]

- main result:

\[
\deg(c) = n + m - \text{rank } (\mathcal{S}(p, q))
\]

\[
\mathcal{S}(p, q) \leftarrow (n + m) \times (n + m) \text{ Sylvester matrix}
\]
The Sylvester matrix of \( p \) and \( q \)

\[
\mathcal{S}(p, q) := \begin{bmatrix}
p_0 & q_0 \\
p_1 & p_0 & q_1 & q_0 \\
\vdots & \ddots & \ddots & \ddots & \ddots \ \\
p_n & \ldots & p_0 & q_m & \ldots & q_0 \\
p_n & p_1 & \ldots & q_m & \ldots & q_1 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots \ \\
p_n & \ldots & \ldots & \ldots & \ldots & \ldots & q_m
\end{bmatrix}
\]

an \((m + n) \times (m + n)\) structured matrix
Other applications

- Factor analysis (psychometrics)
- Multivariate calibration (chemometrics)
- Microarray data analysis (bioinformatics)
- Fundamental matrix estimation (computer vision)
- Factorizability of multivariable polynomials
One problem, many applications

- systems and control
  - model reduction
  - system identification

- signal processing
  - spectral estimation
  - image deblurring

- computational mathematics
  - approx. GCD
  - approx. factorization

- machine learning
  - dim. reduction
  - clustering

structured low-rank approximation
Outline

About the course

Historical review

Applications

Demos

Exercises
IQ test

- extend the sequence: 0, 1, 1, 2, 3, 5, 8, ...

- extend the sequence: 0, 1, 1, 2, 5, 9, 18, ...

- more interesting is to find a systematic solution

- the key ingredient is rank deficiency of a matrix

"Behind every data modeling problem there is a (hidden) low-rank approximation problem: the model imposes relations on the data which render a matrix constructed from exact data rank deficient."
Time series interpolation

- from extrapolation to interpolation

- data: classic Box & Jenkins airline data
  monthly airline passenger numbers 1949–1960

- aim: estimate missing values
  - missing values in "the future": extrapolation
  - other missing values: interpolation
  - take into account the time series nature of the data
Autonomous LTI model

- using all 144 data points to identify a model

- solid line — data, dashed — fit by 6th order model
Missing data estimation

- [5:10 20:30 50:70 100:140] are missing

- piecewise cubic interpolation, 6th order LTI model
Modeling as data compression

- the model is a concise representation of the data
- exact model $\leftrightarrow$ lossless compression (e.g., zip)
- approximate model $\leftrightarrow$ lossy compression (e.g., mp3)
Example: compression of a random vector

- data: $1 \times n$ vector, generated by \texttt{randn}
- compression in \texttt{mat} format

<table>
<thead>
<tr>
<th>length $n$</th>
<th>1</th>
<th>223</th>
<th>334</th>
<th>556</th>
<th>667</th>
<th>1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. original size</td>
<td>8</td>
<td>1784</td>
<td>2672</td>
<td>4448</td>
<td>5336</td>
<td>8000</td>
</tr>
<tr>
<td>2. mat file size</td>
<td>178</td>
<td>1945</td>
<td>2798</td>
<td>4490</td>
<td>5341</td>
<td>7893</td>
</tr>
</tbody>
</table>
Example: low-rank matrix compression

- data: random $100 \times 100$ matrix $D$ of rank 5
- stored in four different ways

<table>
<thead>
<tr>
<th>representation</th>
<th>size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. all elements of $D$</td>
<td>80000</td>
</tr>
<tr>
<td>2. $D$ in mat format</td>
<td>75882</td>
</tr>
<tr>
<td>3. all elements of $P$ and $L$</td>
<td>8024</td>
</tr>
<tr>
<td>4. $P$ and $L$ in mat format</td>
<td>7767</td>
</tr>
</tbody>
</table>

- in 2 and 4, we compute a rank revealing factorization

$$D = PL$$

- can we do better than storing $P$ and $L$ (compressed)?
Example: trajectory of an LTI system

- data: impulse response of a random 3rd order system
- stored in four different ways

<table>
<thead>
<tr>
<th>representation</th>
<th>size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. impulse response $h$</td>
<td>192</td>
</tr>
<tr>
<td>2. $h$ in mat format</td>
<td>377</td>
</tr>
<tr>
<td>3. model parameters $\theta$</td>
<td>56</td>
</tr>
<tr>
<td>4. $\theta$ in mat format</td>
<td>233</td>
</tr>
</tbody>
</table>

- in 3 and 4, we have parameterized the system
Low-rank approximation of images

- an image is a matrix of gray values (integers 0–255)

- typical singular values plot:

- an image can be approximate by lower rank

- the basis of many methods for image processing

- note that SVD does not respect the 0–255 bounds
Original $512 \times 512$ image
Rank 100 approximation
Rank 80 approximation
Rank 60 approximation
Rank 40 approximation
Outline

About the course

Historical review

Applications

Demos

Exercises