Analyzing Data ‘Boxes’: Multi-way linear algebra and its applications in signal processing and communications

Nikos Sidiropoulos

Dept. ECE, TUC-Greece
nikos@telecom.tuc.gr
Dedication

In memory of Richard Harshman († Jan. 10, 2008), who co-founded three-way analysis, and fathered PARAFAC in the early 70’s.

Richard was a true gentleman.
Acknowledgments

- **3-way Students**: X. Liu, T. Jiang

- **3-way Collaborators**: R. Bro (Denmark), J. ten Berge, A. Stegeman (Netherlands), D. Nion (ENSEA-France & TUC-Greece)

- **Sponsors**: NSF CCR 9733540, 0096165, 9979295, 0096164; ONR N/N00014-99-1-0693; DARPA/ATO MDA 972-01-0056; ARL C & N CTA DADD19-01-2-0011
Contents

- 3-way arrays: similarities and differences with matrices
- Rank, and low-rank decomposition
- 3-way notation, using CDMA as example
- Uniqueness
- Algorithms
- Performance
- Application: blind speech separation
- What lies ahead & wrap-up
Applications

- CDMA intercept / signal intelligence
- Sensor array processing
- Multi-dimensional harmonic retrieval
- Radar
- Speech separation
- Multimedia data mining
- Chemistry
- Psychology
- Chromatography, spectroscopy, magnetic resonance ...
Matrices, Factor Analysis, Rotational Indeterminacy

\[
X = AB^T = a_1 b_1^T + \cdots + a_r b_r^T \quad (r := \text{rank}(X))
\]

\[
x_{i,j} = \sum_{k=1}^{r} a_{i,k} b_{j,k}
\]

\[
X = AB^T = A M M^{-1} B^T
\]
Reverse engineering of soup?

Can only guess recipe: $a_1 b_1^T + \cdots + a_r b_r^T$
Sample from two or more Cooks!

Left: $\mathbf{a}_1 \mathbf{b}_1^T + \cdots + \mathbf{a}_r \mathbf{b}_r^T$; right: $1.2 \times \mathbf{a}_1 \mathbf{b}_1^T + \cdots + 0.87 \times \mathbf{a}_r \mathbf{b}_r^T$

Same ingredients, different proportions $\rightarrow$ recipe!
Three-Way Arrays

- Two-way arrays, AKA matrices: $X := [x_{i,j}] : (I \times J)$
- Three-way arrays: $[x_{i,j,k}] : (I \times J \times K)$
- CDMA w/ Rx Ant array @ baseband: chip $\times$ symbol $\times$ antenna
Fact 1: Low-rank matrix (2-way array) decomposition not unique for rank > 1

Fact 2: Low-rank 3- and higher-way array decomposition (PARAFAC) is unique under certain conditions
Three-Way vs Two-Way Arrays - Similarities

- Rank := smallest number of rank-one “factors” (“terms” is probably better) for exact additive decomposition (same concept for both 2-way and 3-way)

- Two-way rank-one factor: rank-one MATRIX outer product of 2 vectors (containing all double products)

- Three-way rank-one factor: rank-one 3-WAY ARRAY outer product of 3 vectors (containing all triple products) - same concept
Three-Way vs Two-Way Arrays - Differences

- Two-way \((I \times J)\): row-rank = column-rank = rank \(\leq\) \(\min(I, J)\);

- Three-way: row-rank \(\neq\) column-rank \(\neq\) “tube”-rank \(\neq\) rank

- Two-way: \(\text{rank}(\text{randn}(I,J))=\min(I,J)\) w.p. 1;

- Three-way: \(\text{rank}(\text{randn}(2,2,2))\) is a RV (2 w.p. 0.3, 3 w.p. 0.7)

- 2-way: rank insensitive to whether or not underlying field is open or closed \((\mathbb{R} \text{ versus } \mathbb{C})\); 3-way: rank sensitive to \(\mathbb{R} \text{ versus } \mathbb{C}\)

- 3-way: Except for loose bounds and special cases [Kruskal; J.M.F. ten Berge], general results for maximal rank and typical rank sorely missing for decomposition over \(\mathbb{R}\); theory more developed for decomposition over \(\mathbb{C}\) [Burgisser, Clausen, Shokrollahi, *Algebraic complexity theory*, Springer, Berlin, 1997]
Column-wise Kronecker Product:

\[
A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 5 & 10 \\ 15 & 20 \\ 25 & 30 \end{bmatrix}, \quad A \odot B = \begin{bmatrix} 5 & 20 \\ 15 & 40 \\ 25 & 60 \\ 15 & 40 \\ 45 & 80 \\ 75 & 120 \end{bmatrix}
\]

\[
\text{vec}(AB^T) = (B \odot A)d(D)
\]

\[
A \odot (B \odot C) = (A \odot B) \odot C
\]
LRD of Three-Way Arrays: Notation

- **Scalar:** [CDMA: \((i, j, k, f): (\text{Rx}, \text{symbol}, \text{chip}, \text{user})\)]
  \[
  x_{i,j,k} = \sum_{f=1}^{F} a_{i,f} b_{j,f} c_{k,f}, \quad i = 1, \ldots, I, \ j = 1, \ldots, J, \ k = 1, \ldots, K
  \]

- **Slabs:**
  \[
  X_k = AD_k(C)B^T, \quad k = 1, \ldots, K
  \]

- **Matrix:**
  \[
  X^{(KJ \times I)} = (B \odot C)A^T
  \]

- **Vector:**
  \[
  x^{(KJI)} := vec \left( X^{(KJ \times I)} \right) = (A \odot (B \odot C))1_{F \times 1} = (A \odot B \odot C)1_{F \times 1}
  \]
LRD of N-Way Arrays: Notation

- **Scalar:**
  \[ x_{i_1, \ldots, i_N} = \sum_{f=1}^{F} \prod_{n=1}^{N} a_{i_n, f}^{(n)} \]

- **Matrix:**
  \[ X^{(I_1I_2\ldots I_{N-1} \times I_N)} = \left( A^{(N-1)} \odot A^{(N-2)} \odot \ldots \odot A^{(1)} \right) \left( A^{(N)} \right)^T \]

- **Vector:**
  \[ x^{(I_1\ldots I_N)} := \text{vec} \left( X^{(I_1I_2\ldots I_{N-1} \times I_N)} \right) = \left( A^{(N)} \odot A^{(N-1)} \odot A^{(N-2)} \odot \ldots \odot A^{(1)} \right) 1_{F \times 1} \]
Uniqueness

\[ X = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{C} \end{pmatrix} \]

\[ k_{\mathbf{A}} + k_{\mathbf{B}} + k_{\mathbf{C}} \geq 2F + 2 \]

k-rank = maximum \( r \) such that every \( r \) columns are linearly independent (\( \leq \) rank)

\[ \textbf{[Kruskal, 1977], } N = 3, \mathbb{R}: \quad k_{\mathbf{A}} + k_{\mathbf{B}} + k_{\mathbf{C}} \geq 2F + 2 \]

\[ \textbf{[Sidiropoulos et al, IEEE TSP, 2000]: } N = 3, \mathbb{C} \]

\[ \textbf{[Sidiropoulos & Bro, J. Chem., 2000]: any } N, \mathbb{C}: \quad \sum_{n=1}^{N} k - \text{ranks} \geq 2F + (N - 1) \]
Kruskal’s Permutation Lemma [Kruskal, 1977]: Consider $A (I \times F)$ having no zero column, and $\tilde{A} (I \times \tilde{F})$. Let $w(\cdot)$ be the weight (# of nonzero elements) of its argument. If for any vector $x$ such that

$$w(x^H \tilde{A}) \leq F - r_{\tilde{A}} + 1,$$

we have

$$w(x^H A) \leq w(x^H \tilde{A}),$$

then $F \leq \tilde{F}$; if also $F \geq \tilde{F}$, then $F = \tilde{F}$, and there exist a permutation matrix $P$ and a non-singular diagonal matrix $D$ such that $A = \tilde{A}PD$.

Easy to show for a pair of square nonsingular matrices (use rows of pinv); but the result is very deep and difficult for fat matrices - see [Jiang & Sidiropoulos, TSP:04], [Stegeman & Sidiropoulos, LAA:07]
Property: [Sidiropoulos & Liu, 1999; Sidiropoulos & Bro, 2000]

If $k_A \geq 1$ and $k_B \geq 1$, then it holds that

$$k_B \odot A \geq \min(k_A + k_B - 1, F),$$

whereas if $k_A = 0$ or $k_B = 0$

$$k_B \odot A = 0$$
Is Kruskal’s Condition Necessary?

- Long-held conjecture (Kruskal’89): Yes

- ten Berge & Sidiropoulos, *Psychometrika*, 2002: Yes for $F \in \{2, 3\}$, no for $F > 3$

- Jiang & Sidiropoulos ’03: new insights that explain part of the puzzle: E.g., for $r_C = F$, the following condition has been proven to be necessary and sufficient:

  No linear combination of two or more columns of $A \odot B$
  
  can be written as KRP of two vectors
P-a.s. uniqueness results

- de Lathauwer ’03 - SIAM JMAA ’06 (cf. Jiang & Sidiropoulos ’03): Decomposition is a.s. unique provided

\[ \min(K, IJ) \geq F \quad \text{and} \quad F(F - 1) \leq \frac{1}{2}I(I - 1)J(J - 1) \]

- Far better than previously known in many cases of practical interest
Algorithms

- SVD/EVD or TLS 2-slab solution (similar to ESPRIT) in some cases (but conditions for this to work are restrictive)
- Workhorse: ALS [Harshman, 1970]: LS-driven (ML for AWGN), iterative, initialized using 2-slab solution or multiple random cold starts
- ALS → monotone convergence, usually to global minimum (uniqueness), close to CRB for $F << IJK$
ALS is based on matrix view:

\[ X^{(KJ \times I)} = (B \odot C)A^T \]

Given interim estimates of \( B, C \), solve for conditional LS update of \( A \):

\[ A_{CLS} = \left( (B \odot C)^+ X^{(KJ \times I)} \right)^T \]

Similarly for the CLS updates of \( B, C \) (symmetry); repeat in a circular fashion until convergence in fit (guaranteed)
**Algorithms**

- ALS initialization matters, not crucial for heavily over-determined problems
- Alt: rank-1 updates possible [Kroonenberg], but inferior
- COMFAC (Tucker3 compression), G-N, Levenberg, ATLD, DTLD, ESPRIT-like,...
- G-N converges faster than ALS, but it may fail
- In general, no “algebraic” solution like SVD
- Possible if e.g., a subset of columns in A is known [Jiang & Sidiropoulos, JASP 2003]; or under very restrictive rank conditions
Robust Regression Algorithms

- Laplacian, Cauchy-distributed errors, outliers

- Least Absolute Error (LAE) criterion: optimal (ML) for Laplacian, robust across $\alpha$-stable

- Similar to ALS, each conditional matrix update can be shown equivalent to a LP problem $\rightarrow$ alternating LP [Vorobyov, Rong, Sidiropoulos, Gershman, 2003]

- Alternatively, very simple element-wise updating using weighted median filtering [Vorobyov, Rong, Sidiropoulos, Gershman, 2003]

- Robust algorithms perform well for Laplacian, Cauchy, and not far from optimal in the Gaussian case
CRBs for the PARAFAC model

- Dependent on how scale-permutation ambiguity is resolved
- Real i.i.d. Gaussian, 3-way, Complex circularly symmetric i.i.d. Gaussian, 3-way & 4-way [Liu & Sidiropoulos, TSP 2001]
- Compact expressions for complex 3-way case & asymptotic CRB when one mode length goes to infinity [Jiang & Sidiropoulos, JASP/SMART:04]
- Laplacian, Cauchy [Vorobyov, Rong, Sidiropoulos, Gershman, TSP:04] - scaled versions of the Gaussian CRB; scaling parameter only dependent on noise pdf
Figure 1: RMSEs versus SNR: Gaussian noise, $8 \times 8 \times 20$, $F = 2$
Figure 2: RMSEs versus SNR: Cauchy noise, $8 \times 8 \times 20, F = 2$
ALS works well in AWGN because it is ML-driven, and with 3-way data it is easy to get to the large-samples regime: e.g., 
$10 \times 10 \times 10 = 1000$

Performance is worse (and further from the CRB) when operating close to the identifiability boundary; but ALS works under model identifiability conditions only, which means that at high SNR the parameter estimates are still accurate.

Main shortcoming of ALS and related algorithms is the high computational cost.

For difficult datasets, so-called swamps are possible: progress towards convergence becomes extremely slow.
Frequency-domain vs time-domain methods

Joint diagonalization (symmetric PARAFAC / INDSCAL) per frequency bin

Exploits time variation in speaker powers: $R_k(f) = A(f)D_kA^H(f)$

Frequency-dependent permutation problem is key

How to ensure consistency ("string together") across bins

Engineering! - not science ...

We now have very competitive solution

Joint work with D. Nion, K. Mokios, A. Potamianos

http://www.telecom.tuc.gr/~nikos/BSS_Nikos.html
Adaptive PARAFAC

Figure 3: Blind speaker separation and tracking
Adaptive PARAFAC

MIMO radar

Figure 4: Trajectory tracking
Adaptive PARAFAC

Complexity

![Graph showing execution time](image)

**Figure 5: Execution time**
Learn more - tutorials, bibliography, papers, software,...

- **Group homepage (Nikos Sidiropoulos):**
  
  www.telecom.tuc.gr/~nikos

- **3-way group at KVL/DK (Rasmus Bro):**
  
  http://www.models.kvl.dk/users/rasmus/ and
  http://www.models.kvl.dk/courses/

- **3-Mode Company (Peter Kroonenburg):**
  
  http://www.leidenuniv.nl/fsi/three-mode/3modecy.htm

- **Hard-to-find original papers (Richard Harshman):**
  
  http://publish.uwo.ca/~harshman/

- **3-way workshop: TRICAP** every 3 years, since ’97; 2006, Chania-Crete Greece; 2009, Pyrenees Spain.
Take home point: \( (N > 3) \)-way arrays are different; low-rank models unique, have many applications

Major challenges: Rank & uniqueness: i) rank detection; ii) necessary & sufficient conditions, esp. for higher-way models; iii) uniqueness under application-specific constraints

Major challenges: Algorithms: Faster at small performance loss; incorporation of application-specific constraints

New exciting applications: Yours!
Preaching the Gospel of 3-Way Analysis

Thank you!