

Introduction to Tensor Decomposition Methods

Ryota Tomioka

Department of Mathematical Informatics,
The University of Tokyo

Includes joint work with Kohei Hayashi, Taiji Suzuki,
and Hisashi Kashima

Who are we?



Ryota Tomioka

Ph.D. from University of Tokyo (2008)

Machine Learning, Kernel methods, Matrix and
Tensor decomposition methods

Conferences: NIPS, ICML, IBIS etc.



Intro to tensor decomposition (**low-rank**)



Kohei Hayashi

Ph.D. from NAIST (2012)

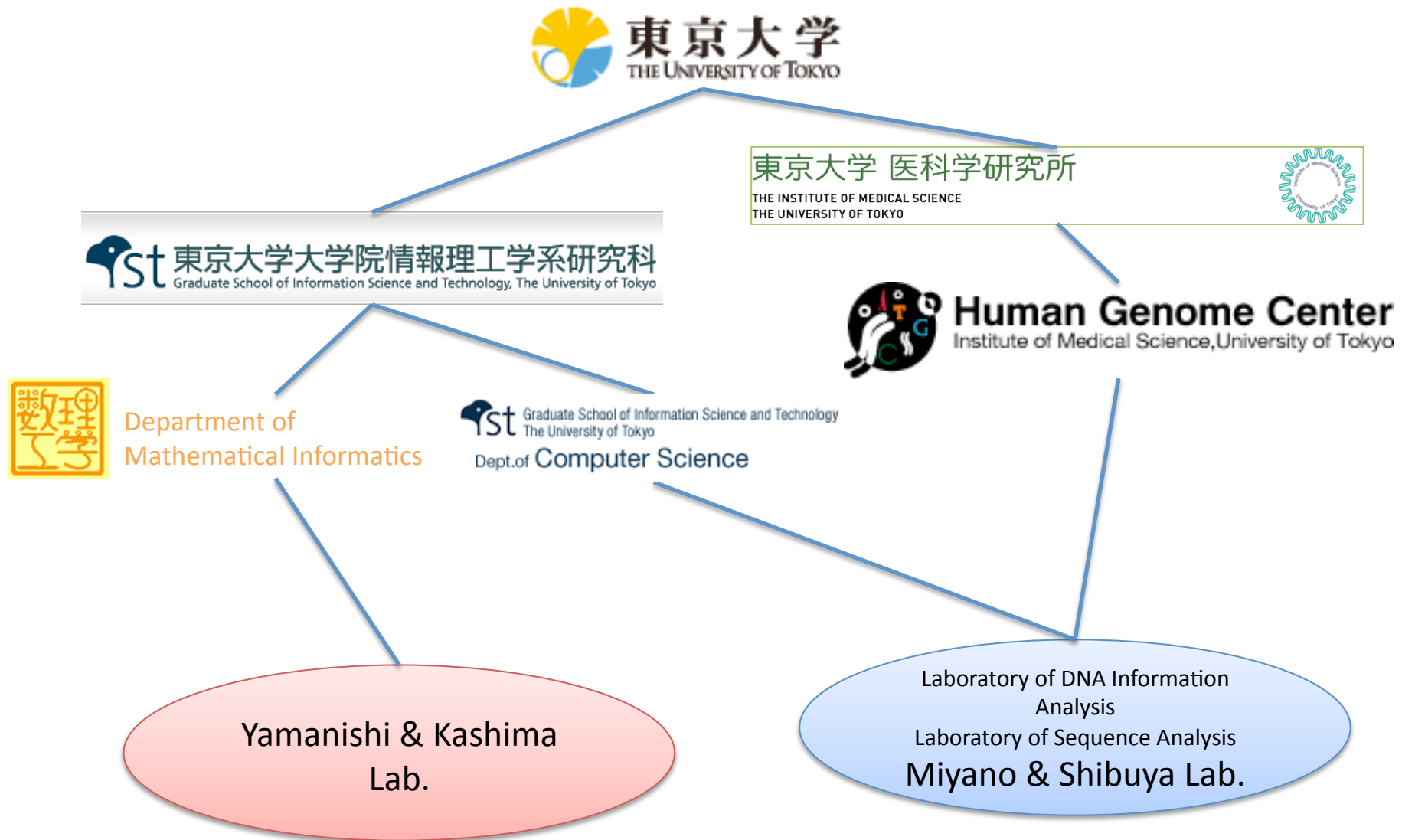
Machine Learning, Data-mining, Tensor models

Conferences: ICDM, ICML, IBIS etc.



Non low-rank matrix/tensor completion

Where are we from?



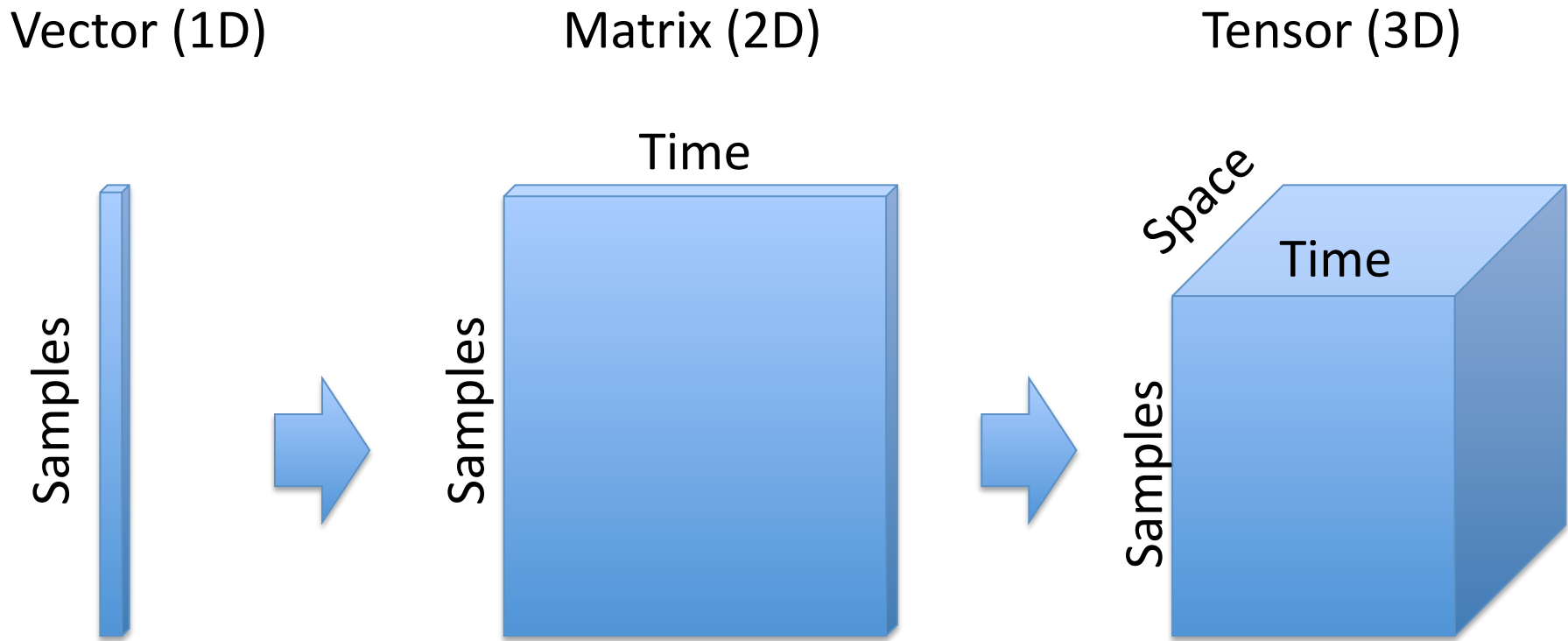


Tensor?

- Two ways to look at it:
 - As a representation of objects
 - As a representation of relations

Tensor represents objects

- Can be regarded as generalization of matrices



More dimensions: multiple features, conditions, etc.

Is this a matrix (tensor)?

Images

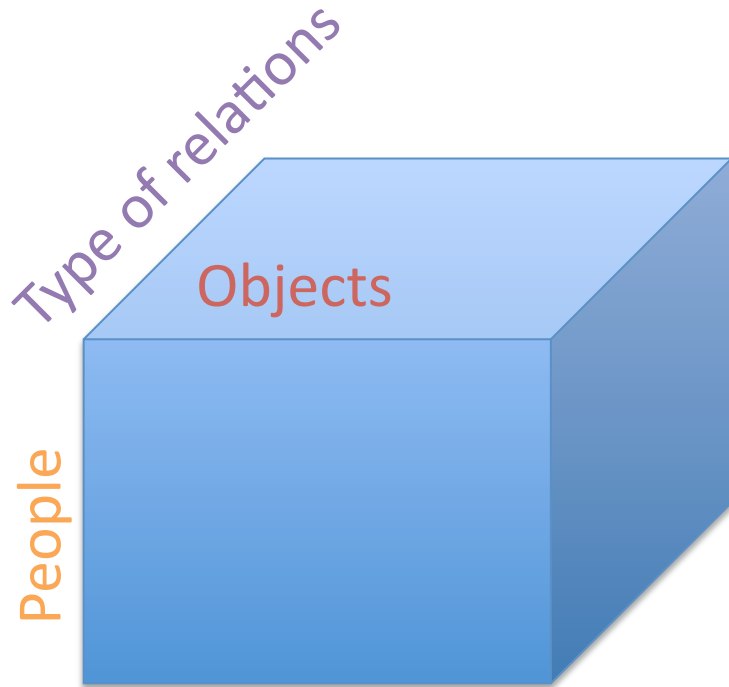


Not obvious if a given data is *naturally* a tensor or not.

We should ask whether tensor representation *helps* or not.

—Tensor is a special matrix, matrix is a special vector

Another view: tensor represents relations



Examples:

- John likes Starwars
- Steve is the CEO of Microsoft
- Drug A binds to protein B

Typical questions

- Missing data imputation (→ Hayashi-san's talk)
 - Some sensors broken.
 - Predict the relation between objects (drug-target interaction, recommendation, etc.)
- **Uncover latent low-dimensional structure**
 - Multi-linear generalization of singular value decomposition
 - What are the hidden components?



Tucker decomposition (HOSVD) / CP decomposition

Recap: Singular value decomposition

$$\begin{matrix} & n \\ m & \boxed{X} \end{matrix} = \begin{matrix} & r \\ m & \boxed{U} \end{matrix} \begin{matrix} & r \\ r & \boxed{\Sigma} \end{matrix} \begin{matrix} & n \\ r & \boxed{V^T} \end{matrix}$$

where U, V : Orthogonal ($U^T U = I, V^T V = I$)

$$\Sigma = \begin{pmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \sigma_r \end{pmatrix}$$

σ_j : j th largest singular value
 r : rank (number of non-zero singular values)

- Note: $r \leq \min(m, n)$
- Can be computed efficiently even for very large matrices (see Mark Tygert's `pca.m`)

Tucker decomposition [Tucker 66]

Factors

$$X = r_1 \text{Core} \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)}$$

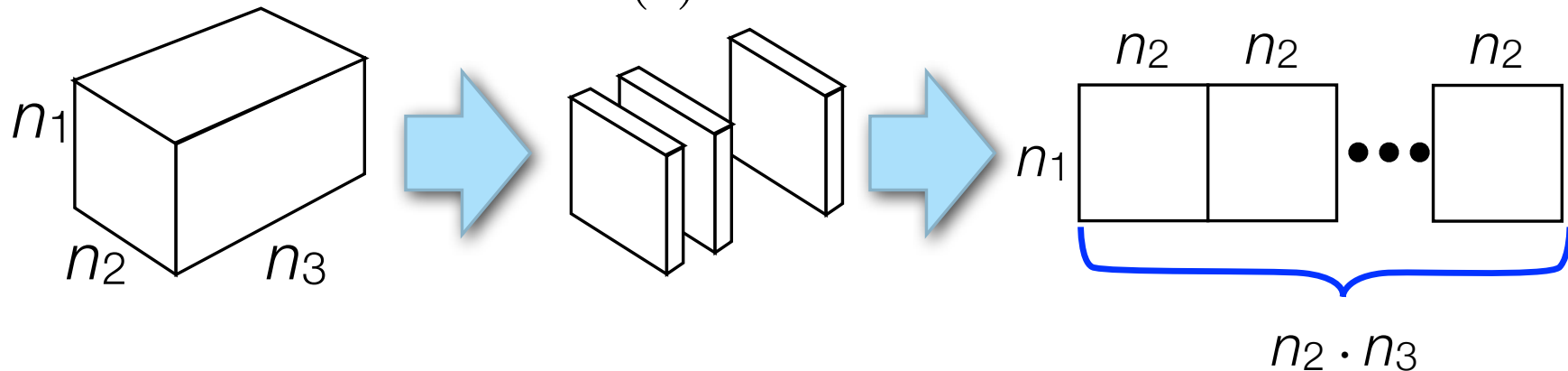
$$\left(X_{ijk} = \sum_{a=1}^{r_1} \sum_{b=1}^{r_2} \sum_{c=1}^{r_3} C_{abc} U_{ia}^{(1)} U_{jb}^{(2)} U_{kc}^{(3)} \right)$$

- rank is defined for each mode (dimension)
- core is not diagonal!
- Orthogonal ambiguity

Computing Tucker decomposition

1. Unfolding (matricization)

Mode-1 unfolding $\mathbf{X}_{(1)}$



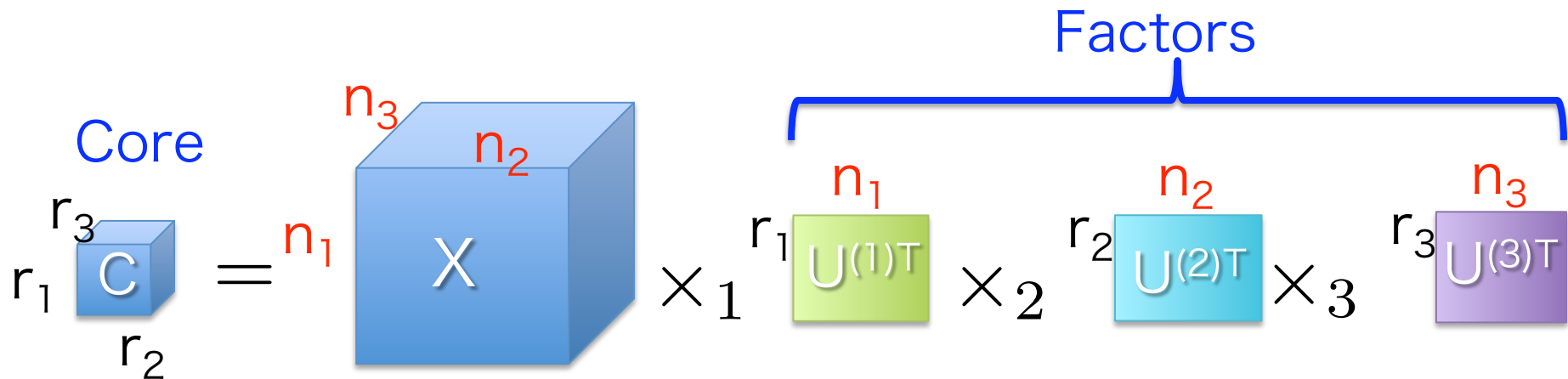
2. Compute SVD

$$\begin{matrix} & n_2 n_3 \\ n_1 & \mathbf{X}_{(1)} \end{matrix} = \begin{matrix} r_1 \\ n_1 & \mathbf{U}^{(1)} \end{matrix} \begin{matrix} r_1 \\ r_1 & \mathbf{\Sigma}_1 \end{matrix} \begin{matrix} n_2 n_3 \\ & \mathbf{V}_1^T \end{matrix}$$

3. Repeat for every mode.

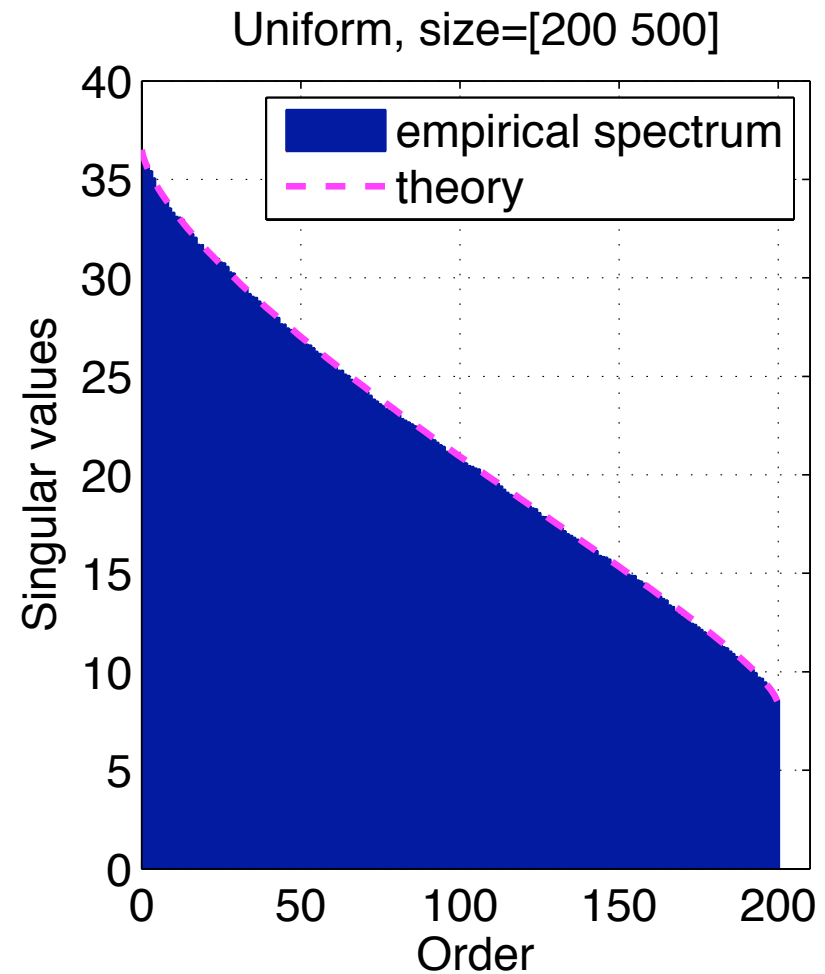
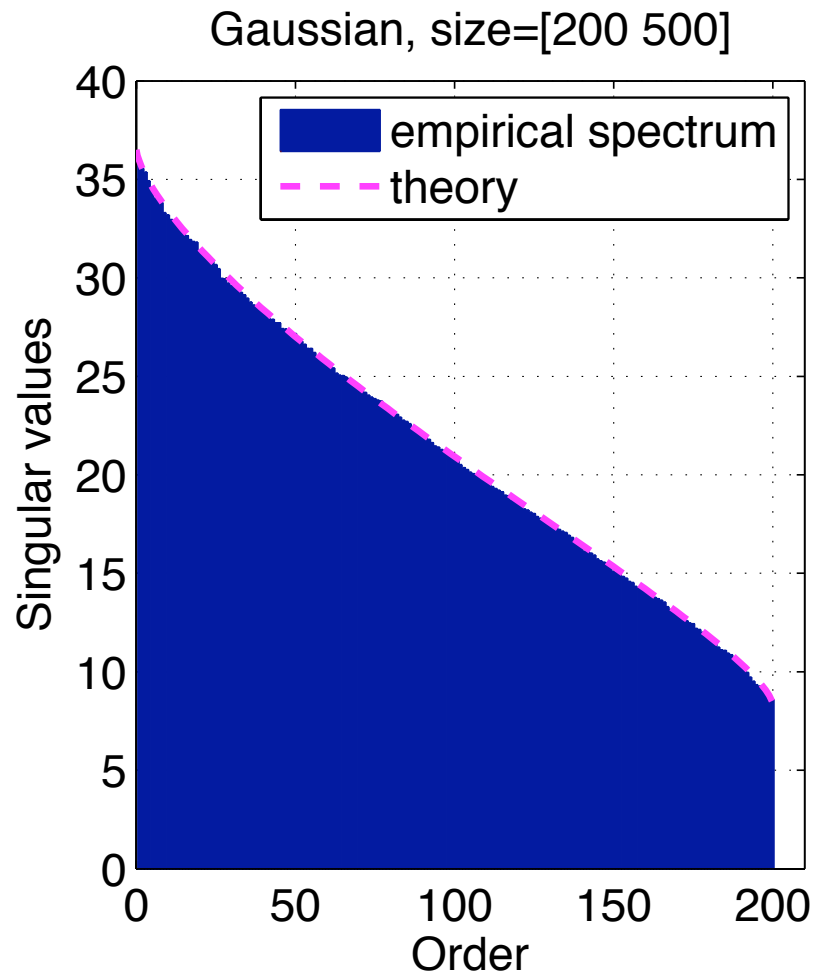
Computing the core

- After obtaining $U^{(1)}$, $U^{(2)}$, $U^{(3)}$



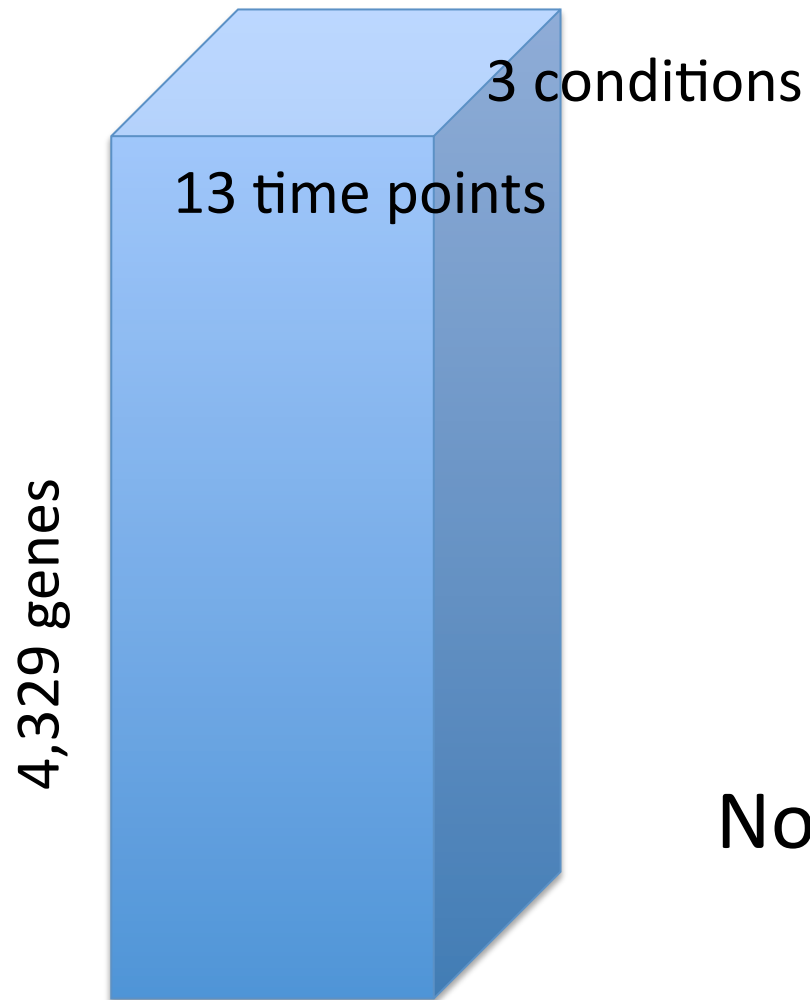
$$C_{abc} = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} X_{ijk} U_{ia}^{(1)} U_{jb}^{(2)} U_{kc}^{(3)}$$

Singular-values of random matrices: Marchenko-Pastur distribution



No similar result known for Tucker decomposition.

“A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies” Omberg et al. PNAS 2007

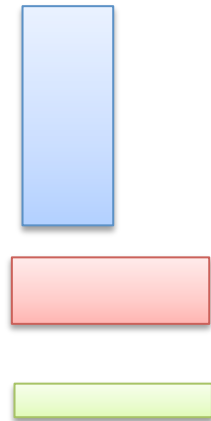


Full-rank Tucker decomposition

Mode-1 unfolding = $4,329 \times 39$

Mode-2 unfolding = $13 \times 12,987$

Mode-3 unfolding = $3 \times 56,277$



Note: maximum rank = $39 \times 13 \times 3$

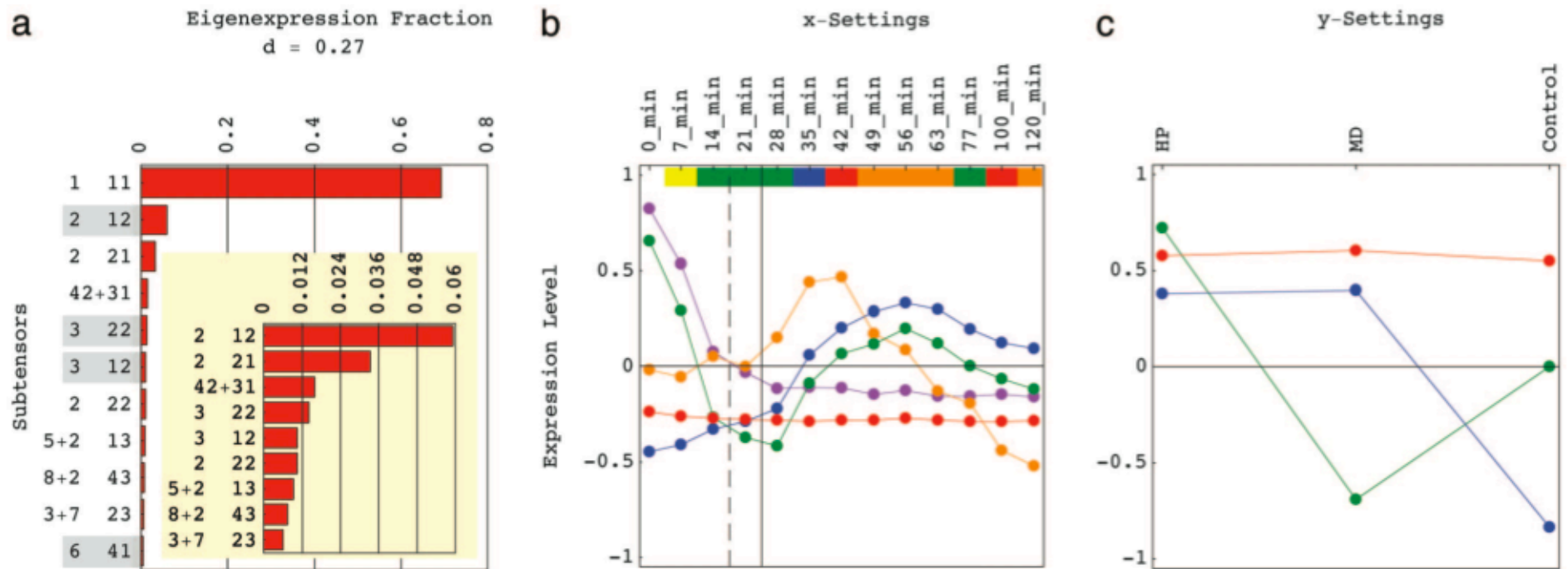


Fig. 1. Significant HOSVD subtensors, after rotation of the approximately degenerate subtensor spaces $\mathcal{S}(4, 2+3, 1)$, $\mathcal{S}(5+2, 1, 3)$, $\mathcal{S}(8+2, 4, 3)$, and $\mathcal{S}(3+7, 2, 3)$. (a) Bar chart of the fractions of the 11 most significant subtensors. The higher-order singular values corresponding to subtensors highlighted in gray are < 0 . The entropy of the data tensor is 0.27. (b) Line-joined graphs of the first (red), second (blue), third (green), and fourth (orange) x-eigenvalues and the superposition of the second and third x-eigenvalues (violet), which define the expression variation across time in these subtensors. The time points are color-coded according to their cell cycle classification in the control time course: M/G₁ (yellow), G₁ (green), S (blue), S/G₂ (red), and G₂/M (orange). The grid lines mark the dissipation of the response to α -factor in the control time course (dashed) and the start of exposure to either HP or MD, at ≈ 20 and 25 min, respectively. (c) Line-joined graphs of the first y-eigenvalue (red), and the second (blue) and third (green) rotated y-eigenvalues, which define the expression variation across the oxidative stress conditions.

[Omberg, Golub, Alter 2007]

CP decomposition

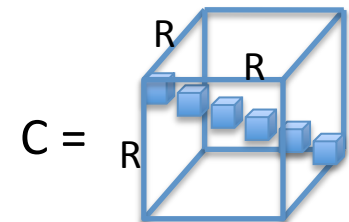
- CP = CANDECAMP [Carroll & Chang 70] / PARAFAC [Harshman 70]

$$X = \sum_{r=1}^R a_r b_r c_r$$

$$\left(X_{ijk} = \sum_{r=1}^R a_{ir} b_{jr} c_{kr} \right)$$

minimal R
is called
the **rank**

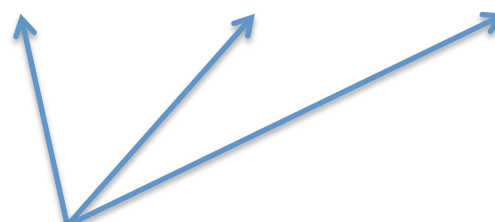
Special case of Tucker decomp. with diagonal core



Properties of CP decomposition

- Tensor rank is **NP complete** [Håstad 90]
 - computing the minimal decomposition is NP hard.
 - In practice, alternate least squares is used.
- CP is **unique** up to permutation and scaling if

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

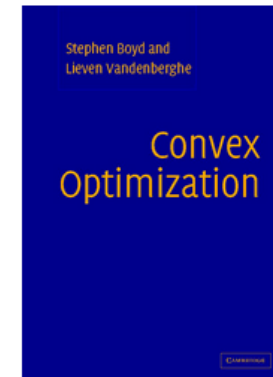
k-rank: maximum number s.t. any k columns of a given matrix is linearly independent.

Issues

- How do we deal with noise and missing data?
 - SVD cannot be used.
 - alternate least squares can over-fit.
- How do we choose the rank?
 - Tucker: three numbers r_1, r_2, r_3
 - CP: single number R

Our recent work: “Statistical Performance of Convex Tensor Decomposition”

- Formulate Tucker decomposition as a **convex optimization** problem



- Convex optimization is fast and reliable
- Instead of choosing ranks (r_1, r_2, r_3) choose the regularization constant λ (like in SVM)
- Statistical analysis of the performance

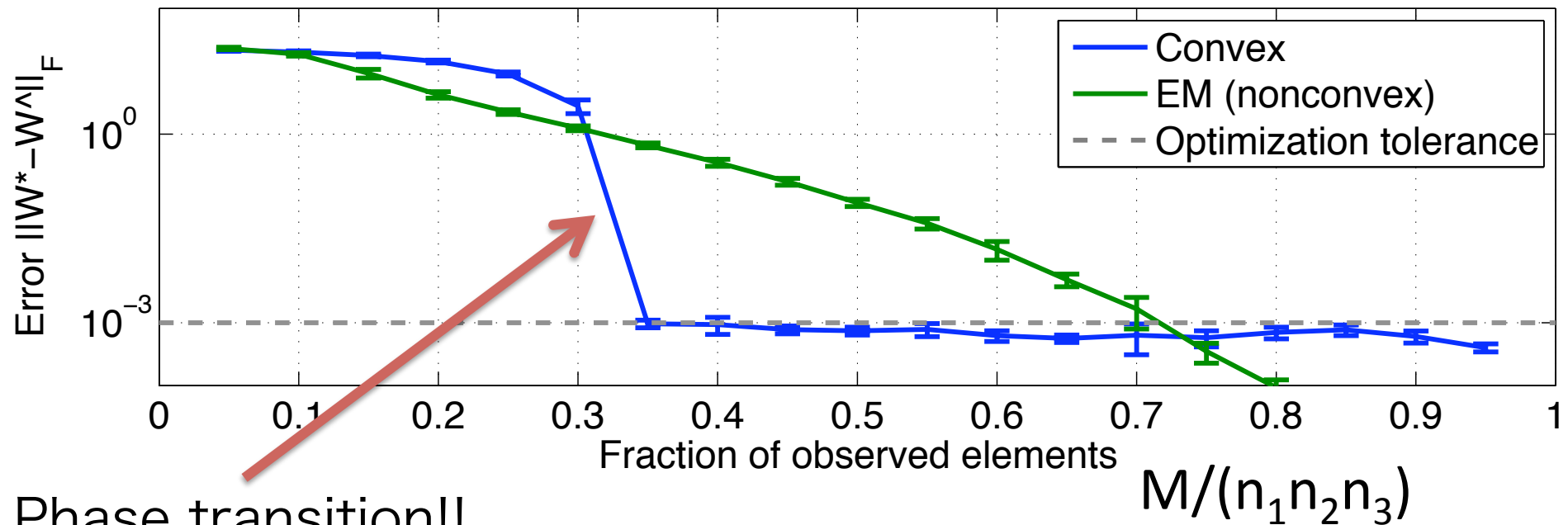
[Tomioka, Suzuki, Hayashi, Kashima 2011]

Empirical Performance

Tensor completion result [Tomioka+ 10]

$$\begin{aligned} \min \quad & \|\mathcal{W}\|_{S_1} \\ \text{s.t.} \quad & \mathcal{W}_{ijk} = \mathcal{Y}_{ijk} \quad ((ijk) \in \Omega) \end{aligned}$$

size=50x50x20, rank=7x8x9 (No noise)



Phase transition!!

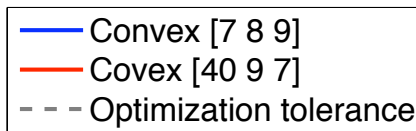
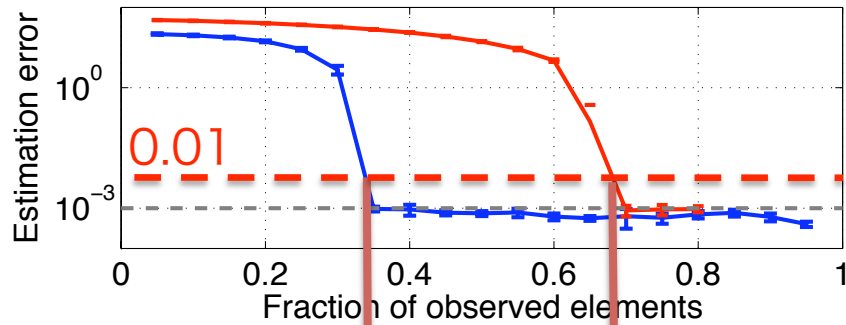
Can we predict this theoretically?

Theoretical analysis

- **Normalized rank** predicts the empirical scaling

size = 50x50x20 true rank 7x8x9 or 40x9x7

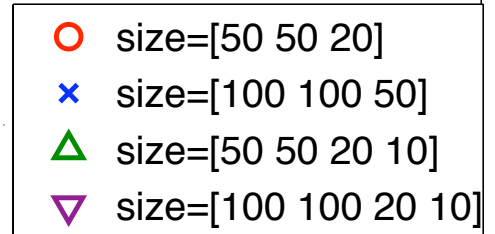
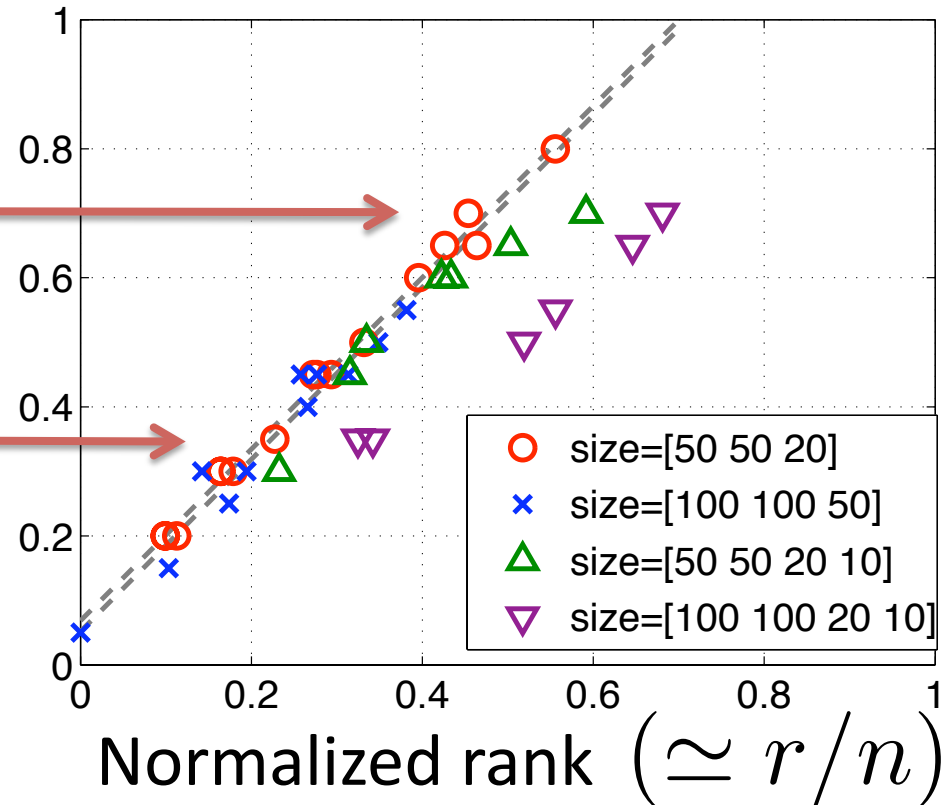
behavior well



$\frac{\text{\#samples } (M)}{\text{\#variables } (N)}$

Fraction M/N at

error ≤ 0.01



Summary

- Two views for tensor data
 - as a representation of objects
 - as a representation of relations
- Two tensor decomposition methods
 - Tucker decomposition
 - CP decomposition
- Convex optimization based tensor decomposition with performance guarantee.

References

- Kolda & Bader (2009) Tensor Decompositions and Applications. SIAM Review.
- M. Mørup (2011) Applications of tensor (multiway array) factorizations and decompositions in data mining. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 1(1).
- Halko, Martinsson, Tropp (2011) Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. SIAM Review, 53 (2).
- Omberg, Golub, Alter (2007) A tensor higher-order singular value decomposition for integrative analysis of DNA microarray data from different studies. PNAS, 104(47).
- Tomioka, Suzuki, Hayashi, Kashima (2011) Statistical Performance of Convex Tensor Decomposition. In Advances in NIPS 24.