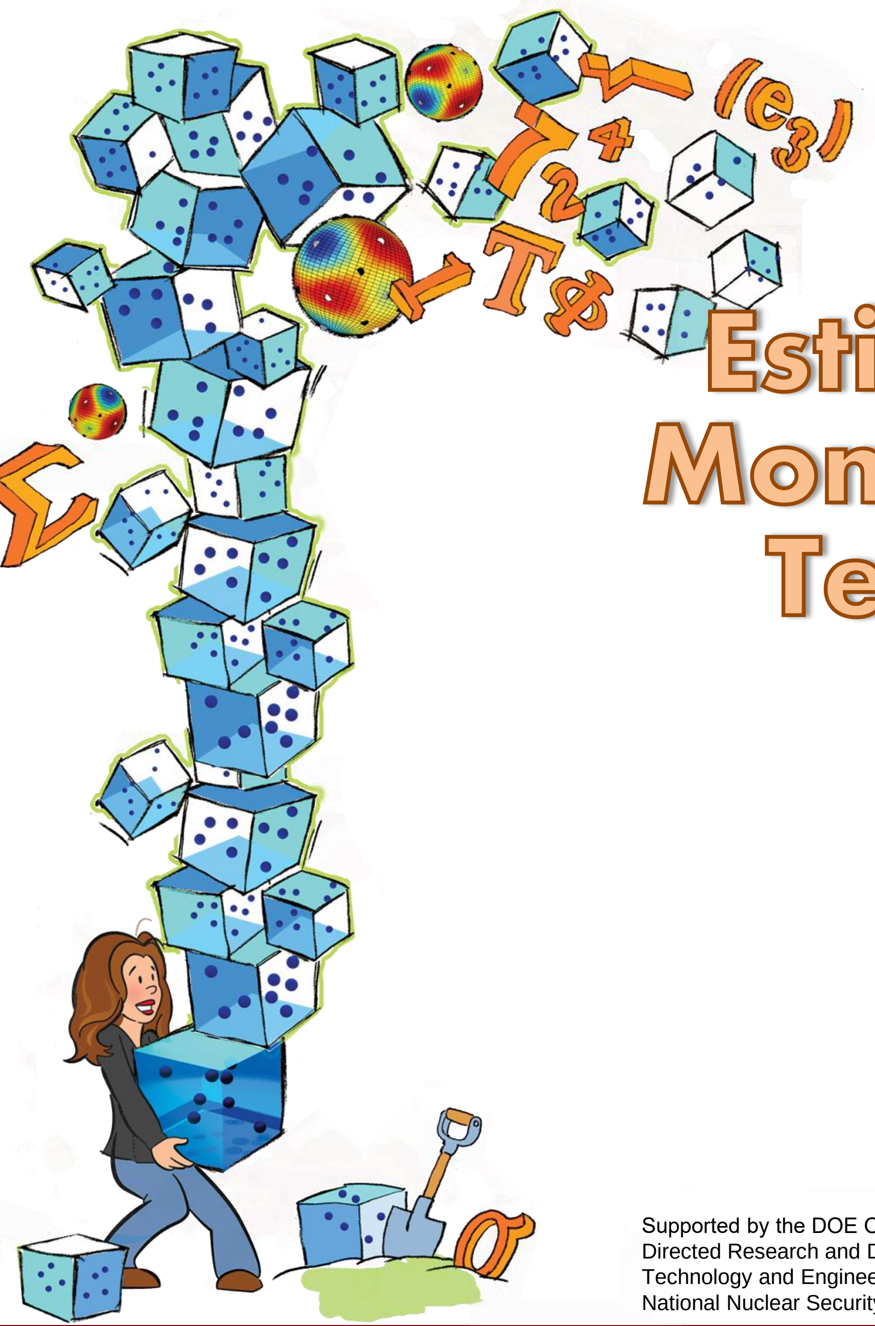
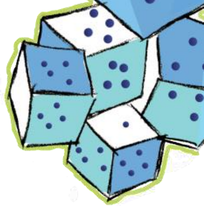


Illustration by Chris Brigman

Sandia National Laboratories



Estimating Higher-Order Moments Using Symmetric Tensor Decomposition

Tamara G. Kolda

Sandia National Labs, Livermore, CA

www.kolda.net

Joint work with

Samantha Sherman

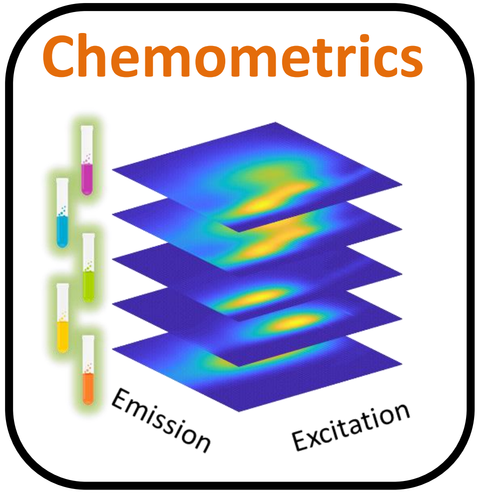
University of Notre Dame, South Bend, IN

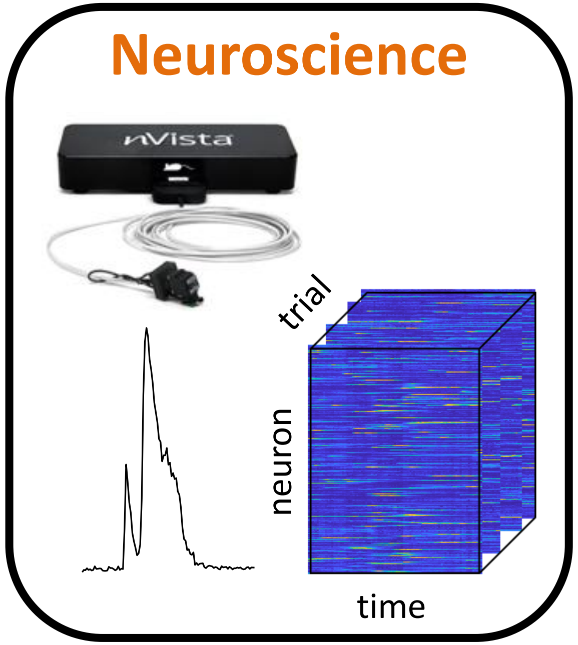
Supported by the DOE Office of Science Advanced Scientific Computing Research (ASCR) Applied Mathematics program and Sandia's Laboratory Directed Research and Development (LDRD program). Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA-0003525.

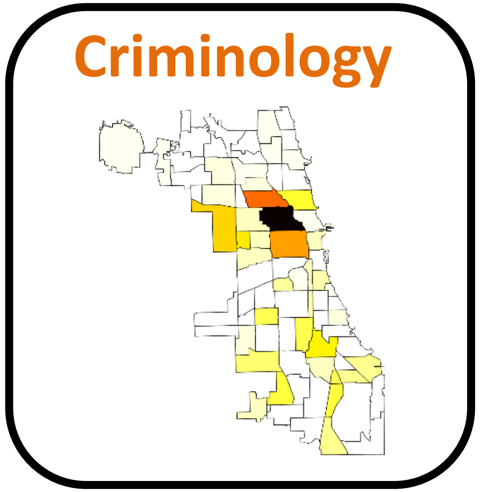
U.S. DEPARTMENT OF ENERGY

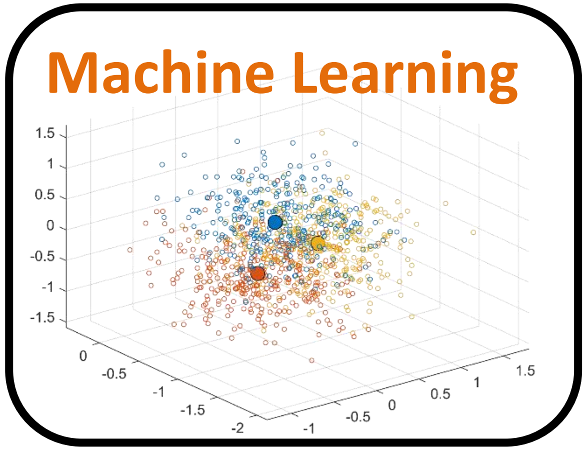
Tensors Come From Many Applications

- **Chemometrics:** Emission x Excitation x Samples (Fluorescence Spectroscopy)
- **Neuroscience:** Neuron x Time x Trial
- **Criminology:** Day x Hour x Location x Crime (Chicago Crime Reports)
- **Symmetric Higher-order Empirical Moments:** Multivariate Gaussian Distributions in Machine Learning
- **Transportation:** Pickup x Dropoff x Time (Taxis)
- **Sports:** Player x Statistic x Season (Basketball)
- **Cyber-Traffic:** IP x IP x Port x Time
- **Social Network:** Person x Person x Time x Interaction-Type
- **Symmetric Higher-order Derivatives:** From Optimization

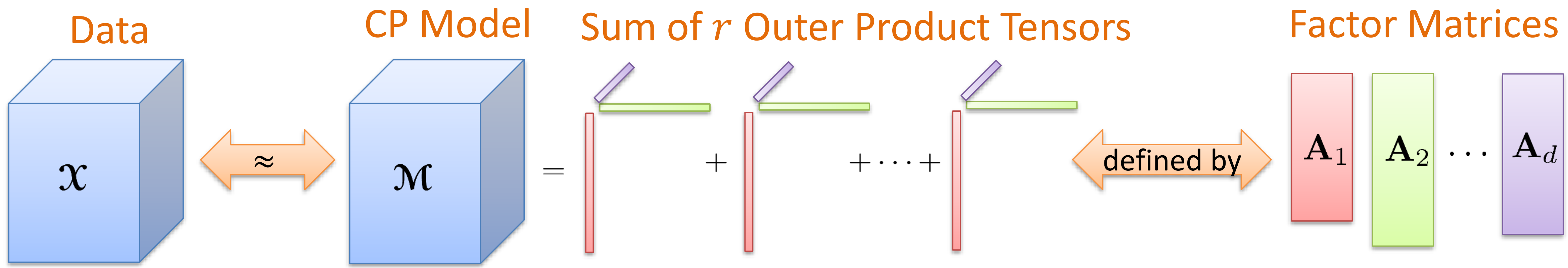








CP Tensor Decomposition Identifies Factors



$$\mathcal{X} \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$$

$$\mathcal{M} = [\![\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_d]\!] \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_d}$$

$$m_{i_1 i_2 \ldots i_d} = \sum_{j=1}^{r} \prod_{k=1}^{d} a_k(i_k, j)$$

$$\mathbf{A}_k \in \mathbb{R}^{n_k \times r}$$

Model Rank

Optimization Formulation

$$\min_{\mathbf{A}_1, \ldots, \mathbf{A}_d} \|\mathcal{X} - \mathcal{M}\|^2 = \sum_{i_1=1}^{n_1} \cdots \sum_{i_d=1}^{n_d} (x_{i_1 i_2 \ldots i_d} - m_{i_1 i_2 \ldots i_d})^2$$

CP First Invented in 1927

Frank Lauren Hitchcock
MIT Professor
(1875–1957)

THE EXPRESSION OF A TENSOR OR A POLYADIC AS A SUM OF PRODUCTS

By Frank L. Hitchcock

1. Addition and Multiplication.

Tensors are *added* by adding corresponding components. The *product* of a covariant tensor $A_{i_1 \cdots i_p}$ of order p *into* a covariant tensor $B_{i_{p+1} \cdots i_{p+q}}$ of order q is defined by writing

$$A_{i_1 \cdots i_p} B_{i_{p+1} \cdots i_{p+q}} = C_{i_1 \cdots i_{p+q}} \tag{1}$$

where the product $C_{i_1 \cdots i_{p+q}}$ is a covariant tensor of order $p+q$. When no confusion results indices may be omitted giving

$$\mathbf{AB} = \mathbf{C} \tag{1_a}$$

equivalent to the n^{p+q} equations (1). Boldface type is convenient for indicating that the letters do not denote merely numbers or scalars. Products of contravariant and of mixed tensors may be similarly defined.

A partial statement of the problem to be considered is as follows: to find under what conditions a given tensor can be expressed as a sum of products of assigned form. A more general statement of the problem will be given below.

2. Polyadic form of a tensor.

Any covariant tensor $A_{i_1 \cdots i_p}$ can be expressed as the sum of a finite number of tensors each of which is the product of p covariant vectors,

$$A_{i_1 \cdots i_p} = \sum_{j=1}^{j=h} a_{1j,i_1} a_{2j,i_2} \cdots a_{pj,i_p} \tag{2}$$

where a_{1j,i_1}, etc., are a set of hp covariant vectors. When the indices $i_1 \cdots i_p$ can be omitted this may be written

$$\mathbf{A} = \sum_{j=1}^{j=h} \mathbf{a}_{1j} \mathbf{a}_{2j} \cdots \mathbf{a}_{pj}. \tag{2_a}$$

The right member is now identical in appearance with a Gibbs

F. L. Hitchcock, ***The Expression of a Tensor or a Polyadic as a Sum of Products***, Journal of Mathematics and Physics, 1927

2. Polyadic form of a tensor.

Any covariant tensor $A_{i_1 \cdots i_p}$ can be expressed as the sum of a finite number of tensors each of which is the product of p covariant vectors,

$$A_{i_1 \cdots i_p} = \sum_{j=1}^{j=h} a_{1j,i_1} a_{2j,i_2} \cdots a_{pj,i_p} \tag{2}$$

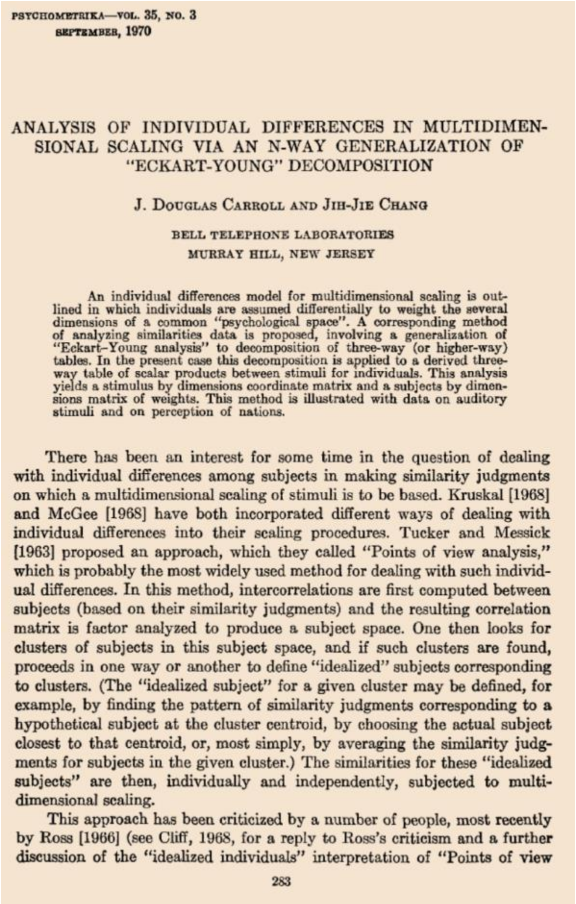
where a_{1j,i_1}, etc., are a set of hp covariant vectors. When the indices $i_1 \cdots i_p$ can be omitted this may be written

$$\mathbf{A} = \sum_{j=1}^{j=h} \mathbf{a}_{1j} \mathbf{a}_{2j} \cdots \mathbf{a}_{pj}. \tag{2_a}$$

CP Independently Reinvented (twice) in 1970

Sandia National Laboratories

CANDECOMP: Canonical Decomposition



PSYCHOMETRIKA—VOL. 35, NO. 3
SEPTEMBER, 1970

ANALYSIS OF INDIVIDUAL DIFFERENCES IN MULTIDIMENSIONAL SCALING VIA AN N-WAY GENERALIZATION OF "ECKART-YOUNG" DECOMPOSITION

J. DOUGLAS CARROLL AND JIH-JIE CHANG

BELL TELEPHONE LABORATORIES
MURRAY HILL, NEW JERSEY

An individual differences model for multidimensional scaling is outlined in which individuals are assumed differentially to weight the several dimensions of a common "psychological space". A corresponding method of analyzing similarities data is proposed, involving a generalization of "Eckart–Young analysis" to decomposition of three-way (or higher-way) tables. In the present case this decomposition is applied to a derived three-way table of scalar products between stimuli for individuals. This analysis yields a stimulus by dimensions coordinate matrix and a subjects by dimensions matrix of weights. This method is illustrated with data on auditory stimuli and on perception of nations.

There has been an interest for some time in the question of dealing with individual differences among subjects in making similarity judgments on which a multidimensional scaling of stimuli is to be based. Kruskal [1968] and McGee [1968] have both incorporated different ways of dealing with individual differences into their scaling procedures. Tucker and Messick [1963] proposed an approach, which they called "Points of view analysis," which is probably the most widely used method for dealing with such individual differences. In this method, intercorrelations are first computed between subjects (based on their similarity judgments) and the resulting correlation matrix is factor analyzed to produce a subject space. One then looks for clusters of subjects in this subject space, and if such clusters are found, proceeds in one way or another to define "idealized" subjects corresponding to clusters. (The "idealized subject" for a given cluster may be defined, for example, by finding the pattern of similarity judgments corresponding to a hypothetical subject at the cluster centroid, by choosing the actual subject closest to that centroid, or, most simply, by averaging the similarity judgments for subjects in the given cluster.) The similarities for these "idealized subjects" are then, individually and independently, subjected to multidimensional scaling.

This approach has been criticized by a number of people, most recently by Ross [1966] (see Cliff, 1968, for a reply to Ross's criticism and a further discussion of the "idealized individuals" interpretation of "Points of view

283

J. Douglas Carroll
Bell Labs
(1939-2011)

Jih-Jie Chang
Bell Labs
(1927-2007)

Richard A. Harshman
Univ. Ontario
(1943-2008)

PARAFAC: Parallel Factors

NOTE: This manuscript was originally published in 1970 and is reproduced here to make it more accessible to interested scholars. The original reference is
Harshman, R. A. (1970). Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics, 16*, 1-84. (University Microfilms, Ann Arbor, Michigan, No. 10,085).

FOUNDATIONS OF THE PARAFAC PROCEDURE: MODELS AND CONDITIONS FOR AN "EXPLANATORY" MULTIMODAL FACTOR ANALYSIS

by

Richard A. Harshman

UCLA

Working Papers in Phonetics

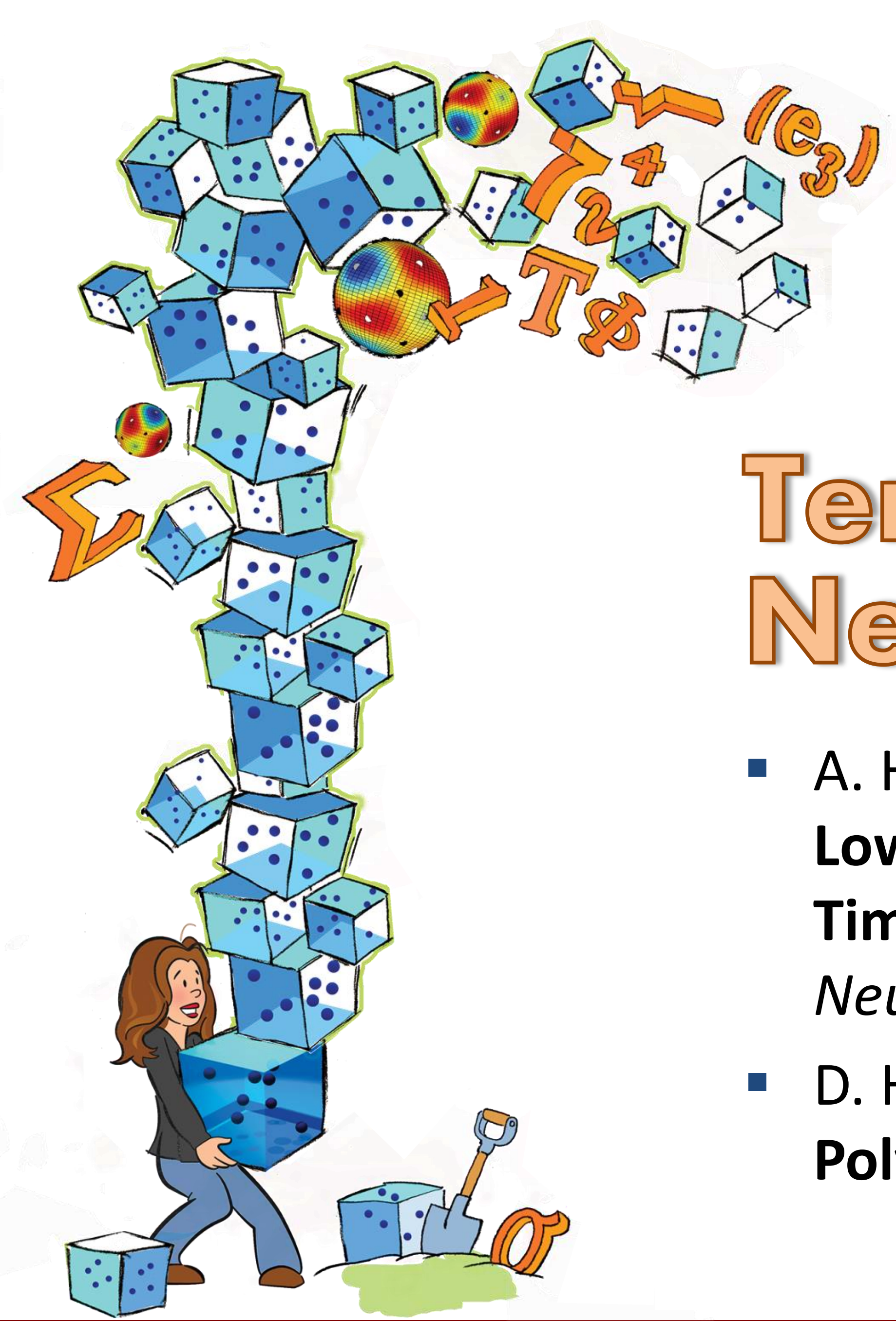
16

December, 1970

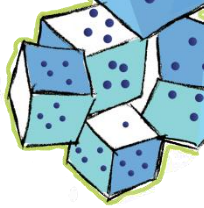
CP: CANDECOMP/PARAFAC — In 2000, Henk Kiers proposed this *compromise* name

CP: Canonical Polyadic — 2010: Pierre Comon, Lieven DeLathauwer, and others reverse-engineered CP, revising some of Hitchcock's terminology

Many thanks to the following persons for helping me learn about Jih-Jie Chang: Fan Chung, Ron Graham, Shen Lin (husband), May Chang (niece), Lili Bruer (daughter).



Sandia National Laboratories



Tensor Decomposition in Neuroscience

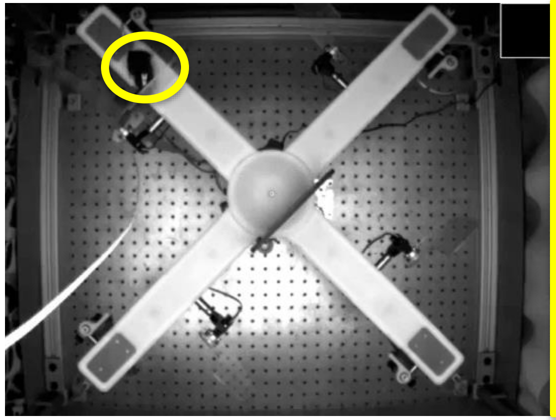
- A. H. Williams et al. **Unsupervised Discovery of Demixed, Low-dimensional Neural Dynamics across Multiple Timescales through Tensor Components Analysis**. *Neuron*, 2018
- D. Hong, T. G. Kolda, J. A. Duersch. **Generalized Canonical Polyadic Tensor Decomposition**. *SIAM Review*, 2020

Activity of Single Neuron Measured Over Time Produces Vector Data

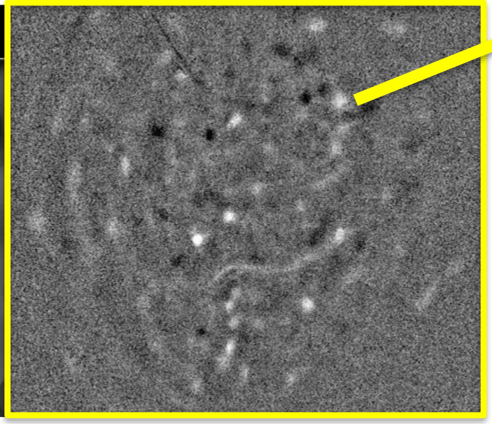
Thanks to Schnitzer Group @ Stanford
Mark Schnitzer, Fori Wang, Tony Kim

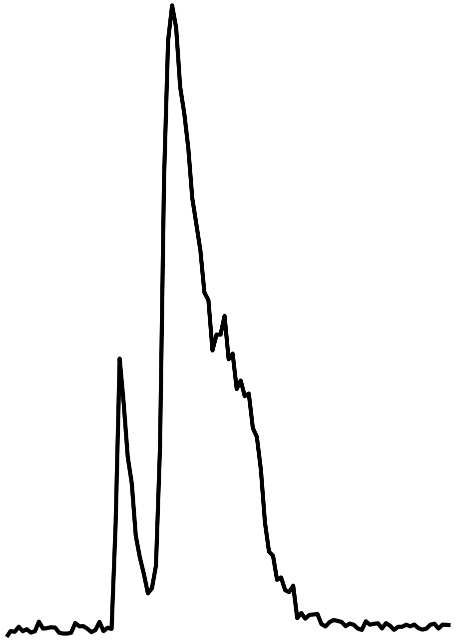
Microscope by Inscopix

mouse in maze



neural activity via calcium imaging





Williams et al., Neuron, 2018

Activity of Single Neuron Measured Over Time Produces Vector Data

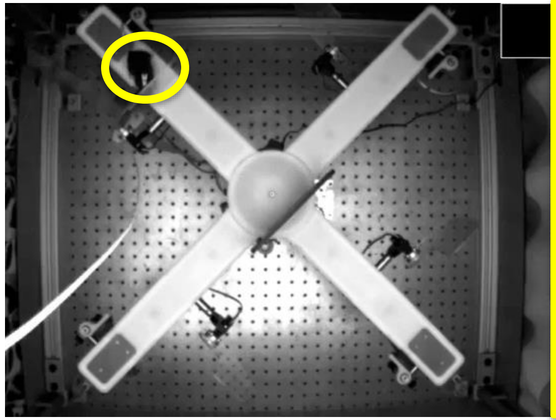
Sandia National Laboratories

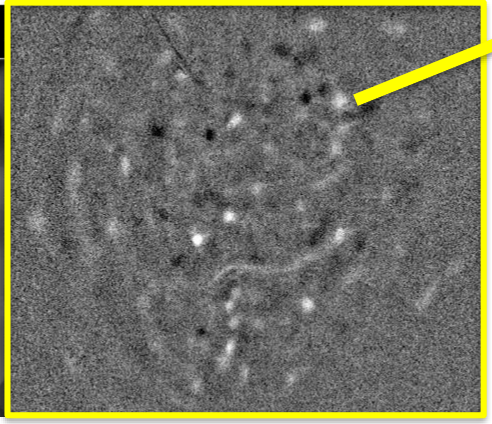
Thanks to Schnitzer Group @ Stanford
Mark Schnitzer, Fori Wang, Tony Kim

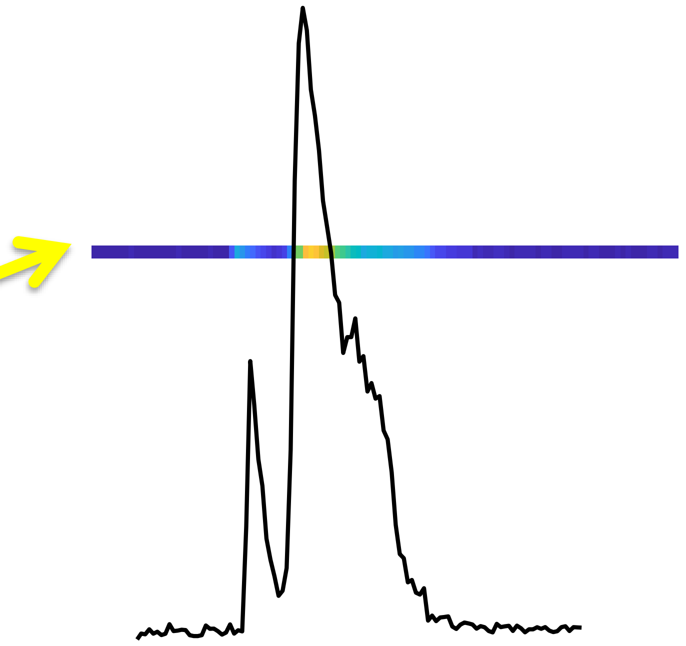
Microscope by Inscopix

mouse in maze

neural activity via calcium imaging







Williams et al., Neuron, 2018

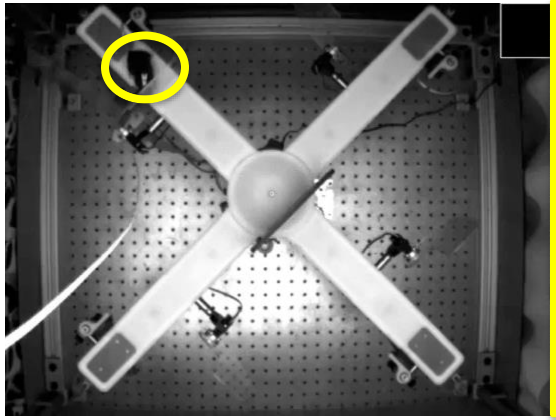
Multiple Neurons Measured Over Time Produces Matrix

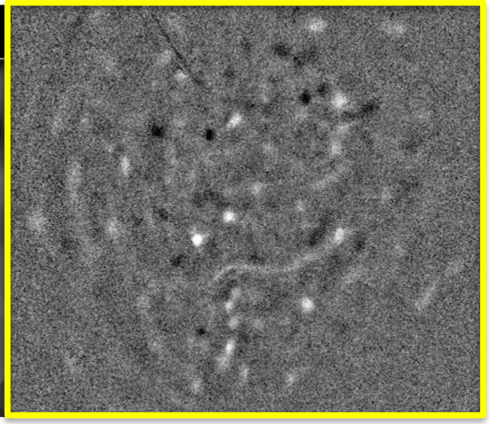
Thanks to Schnitzer Group @ Stanford
Mark Schnitzer, Fori Wang, Tony Kim

Microscope by Inscopix

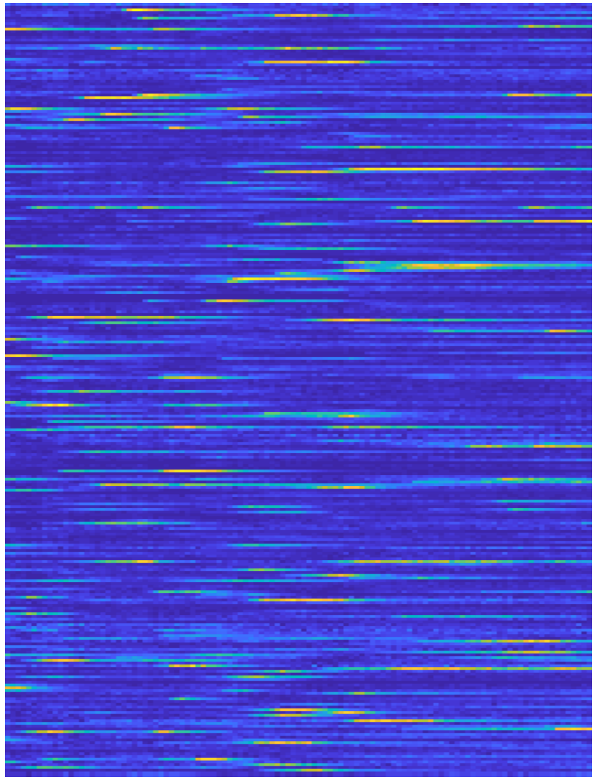
mouse
in “maze”

neural activity



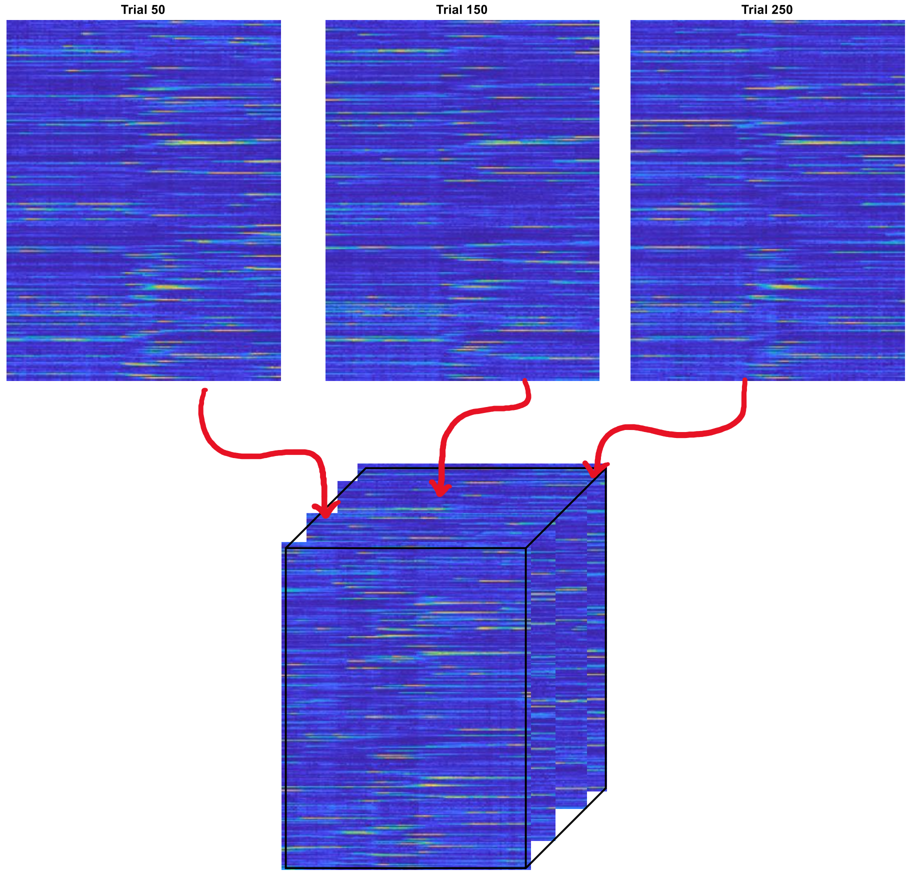


282 neurons × 111 time bins

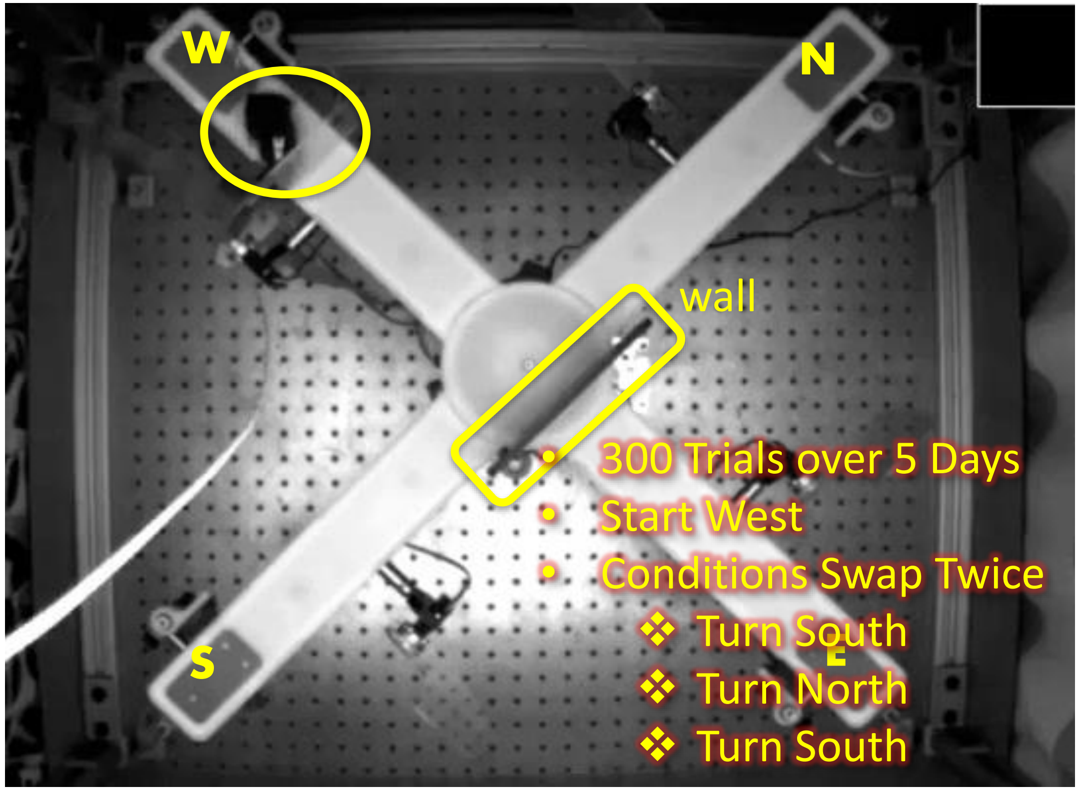


Williams et al., Neuron, 2018

Multiple Trials Produces 3-way Tensor



282 neurons × 111 time bins × 300 trials

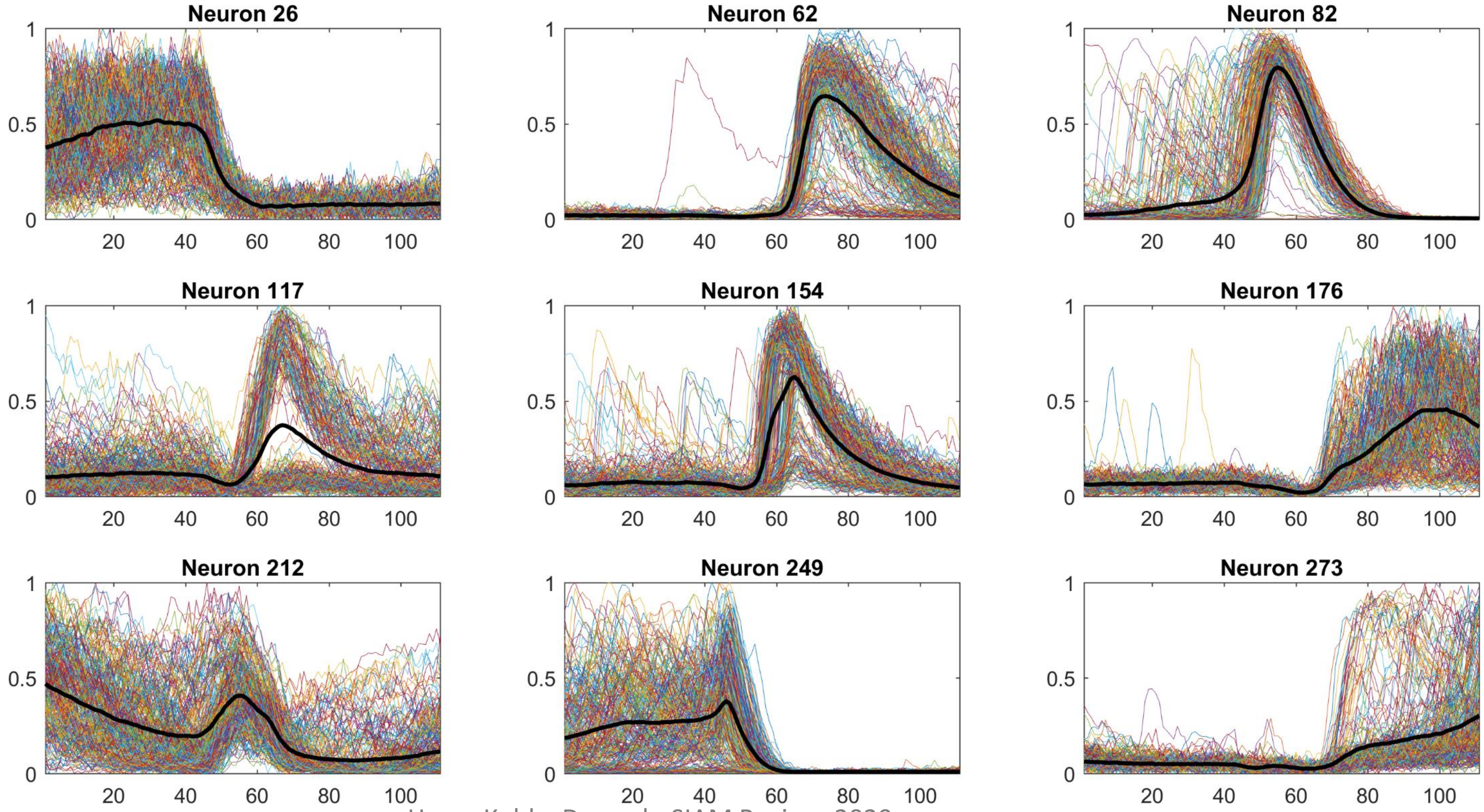


Williams et al., Neuron, 2018

Example Neuron Activity

Thin lines show 300 individual trials

Thick line is average



Hong, Kolda, Duersch, SIAM Review, 2020

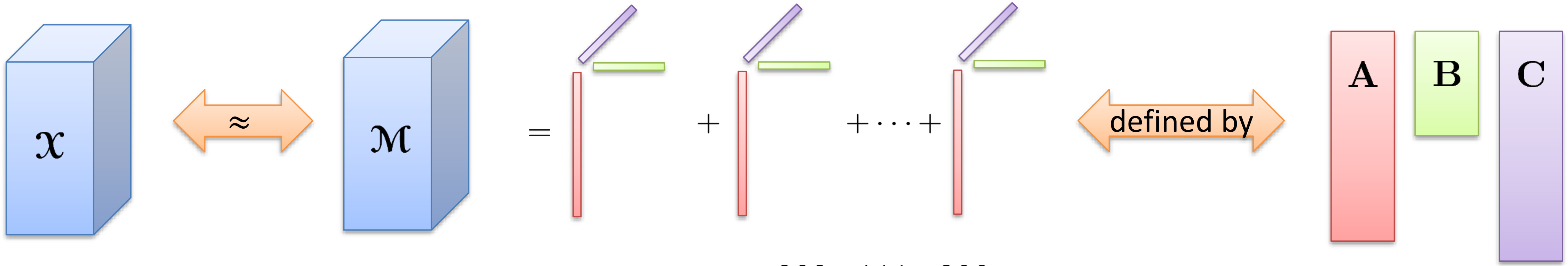
CP Tensor for Neuron Activity Data

Data

CP Model

Sum of r Outer Product Tensors

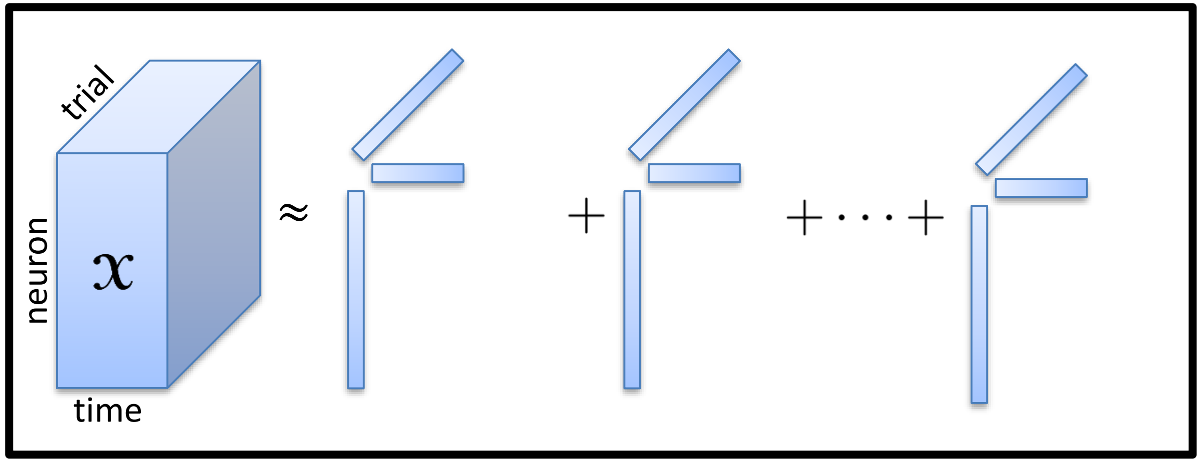
Factor Matrices



$\mathcal{X} \in \mathbb{R}^{282 \times 111 \times 300}$

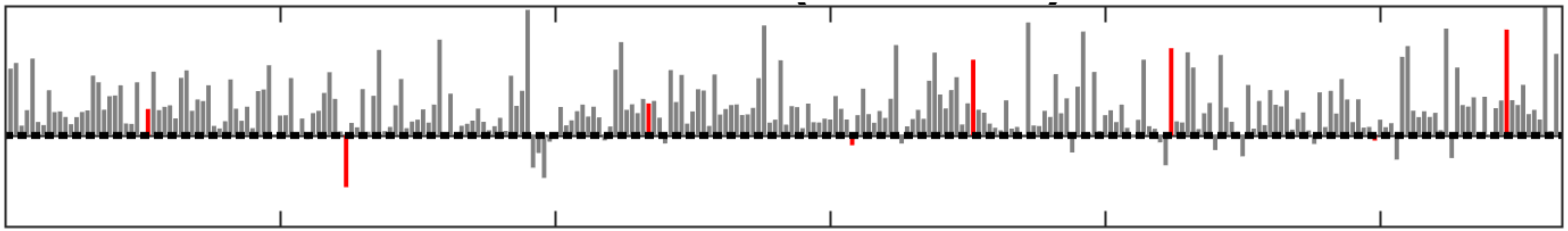
$\mathcal{M} = [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \in \mathbb{R}^{282 \times 111 \times 300}$

Neuron Factor Vector Visualized as Bar Chart

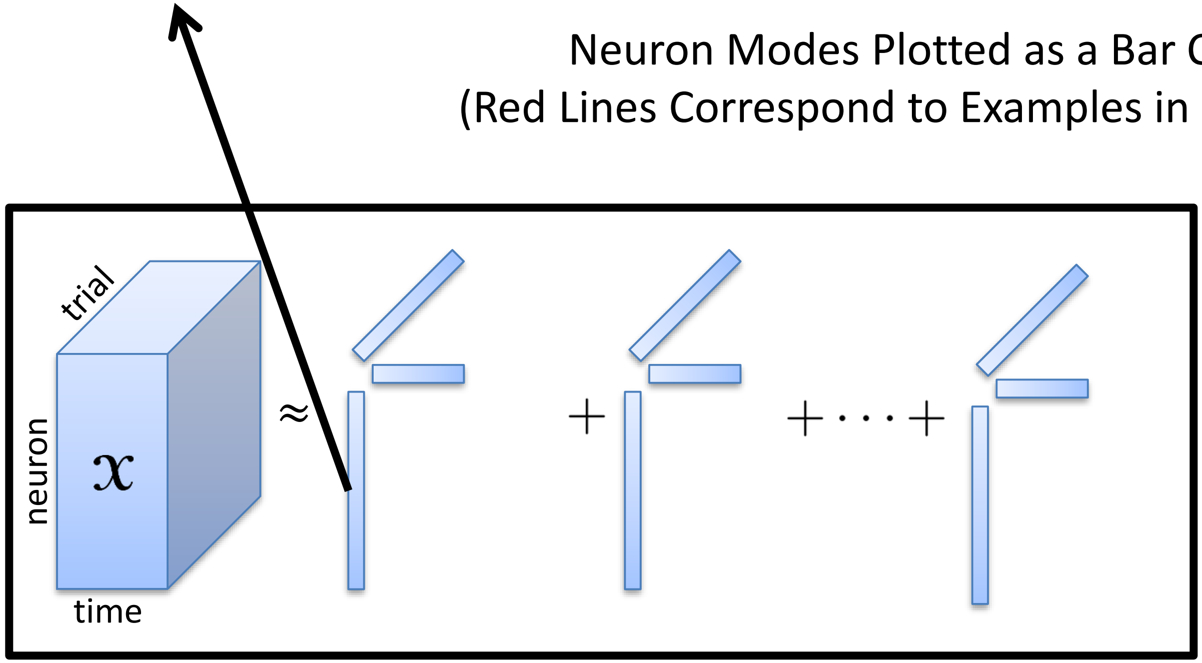


Hong, Kolda, Duersch, SIAM Review, 2020

Neuron Factor Vector Visualized as Bar Chart

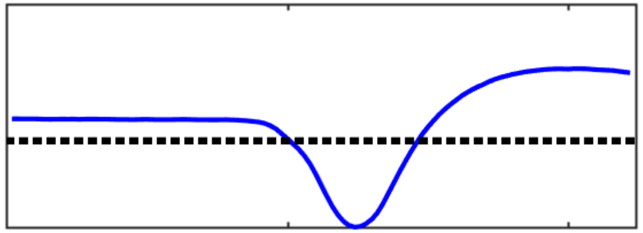


Neuron Modes Plotted as a Bar Chart
(Red Lines Correspond to Examples in Prior Slide)

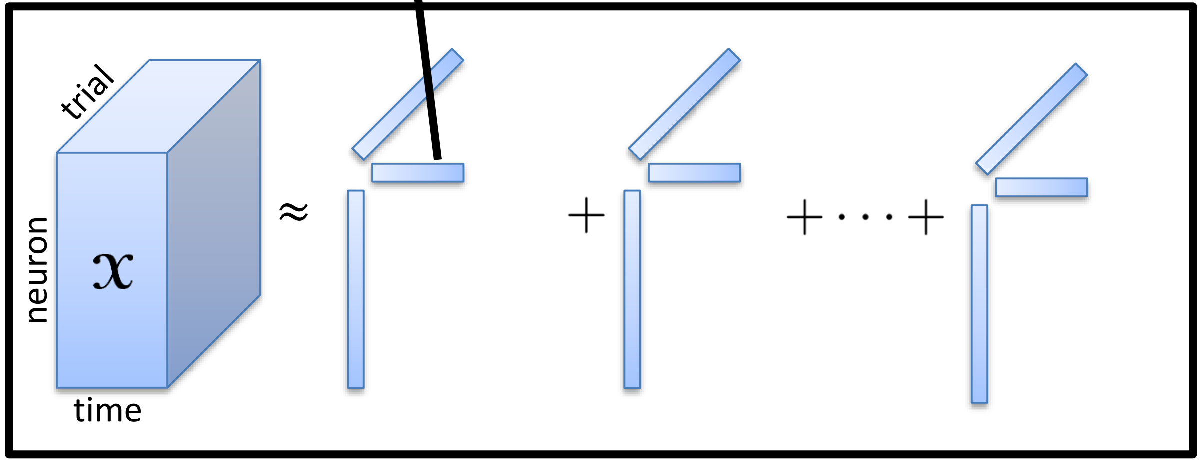


Hong, Kolda, Duersch, SIAM Review, 2020

Time Factor Vector Visualized as Line

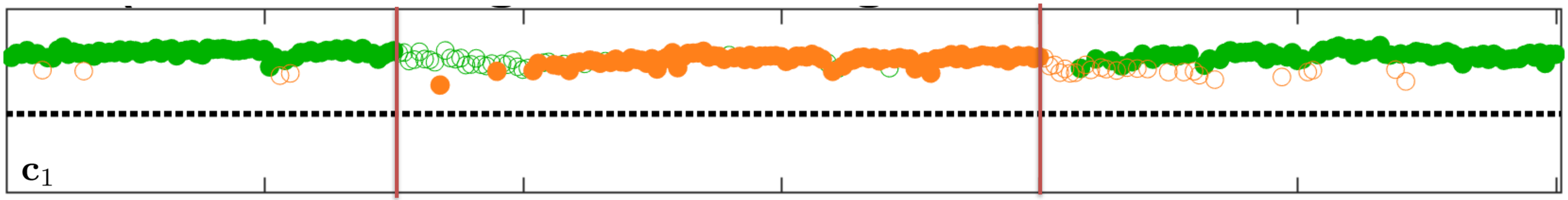


Time (within trial) Plotted as a Line
(Dashed Line is Zero)

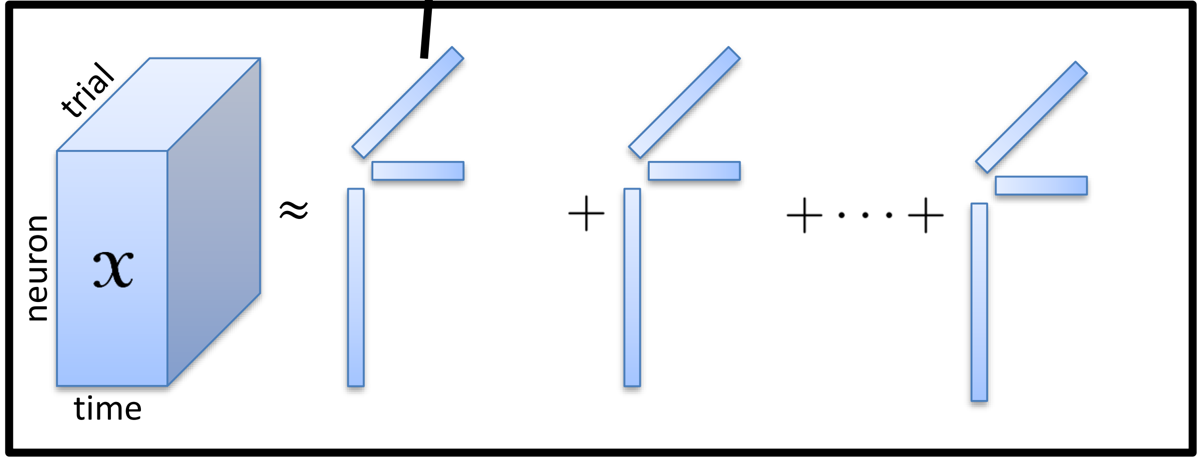


Hong, Kolda, Duersch, SIAM Review, 2020

Trial Factor Vector Visualized as Color-Coded Scatter Plot

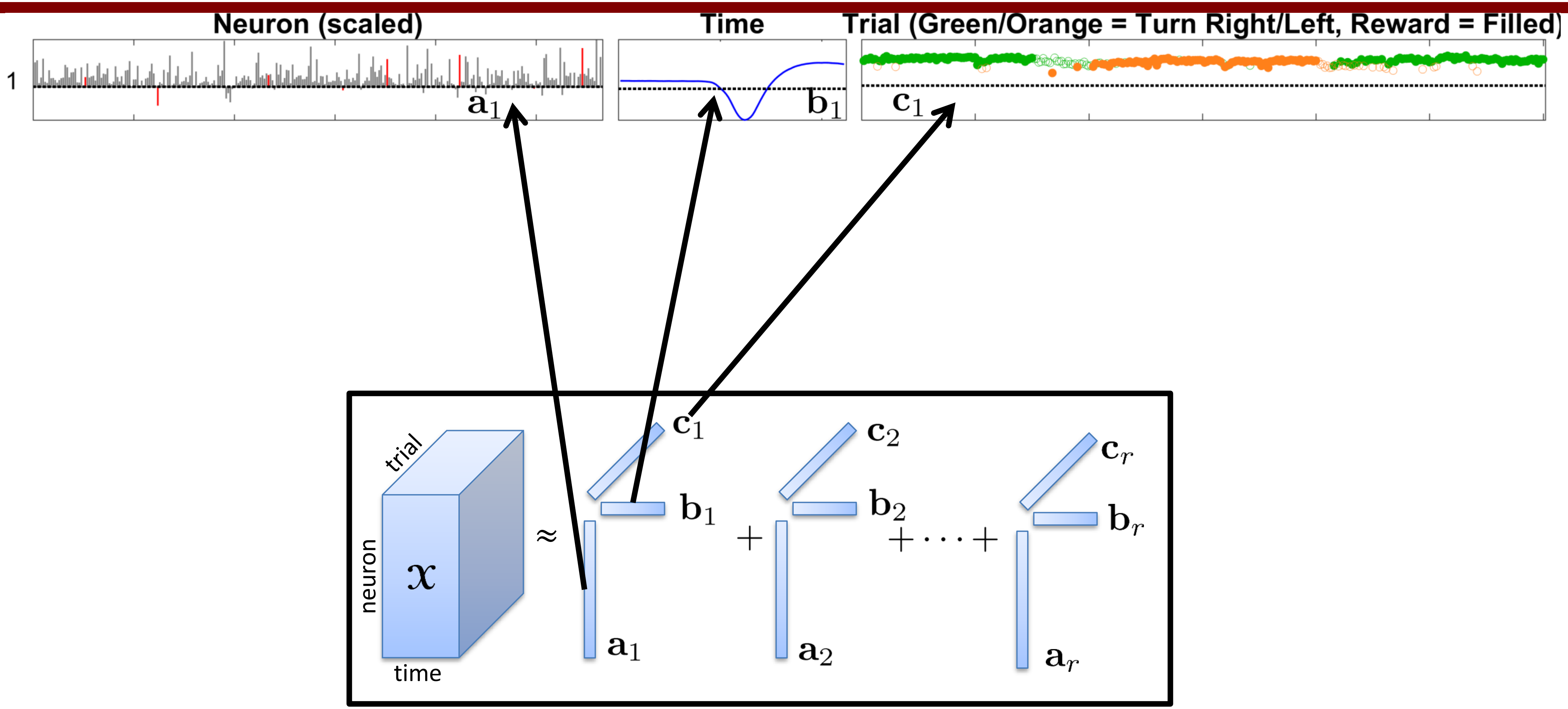


Trial Plotted as Scatter Graph
Right turn = Green
Left turn = Orange
Filled = Reward



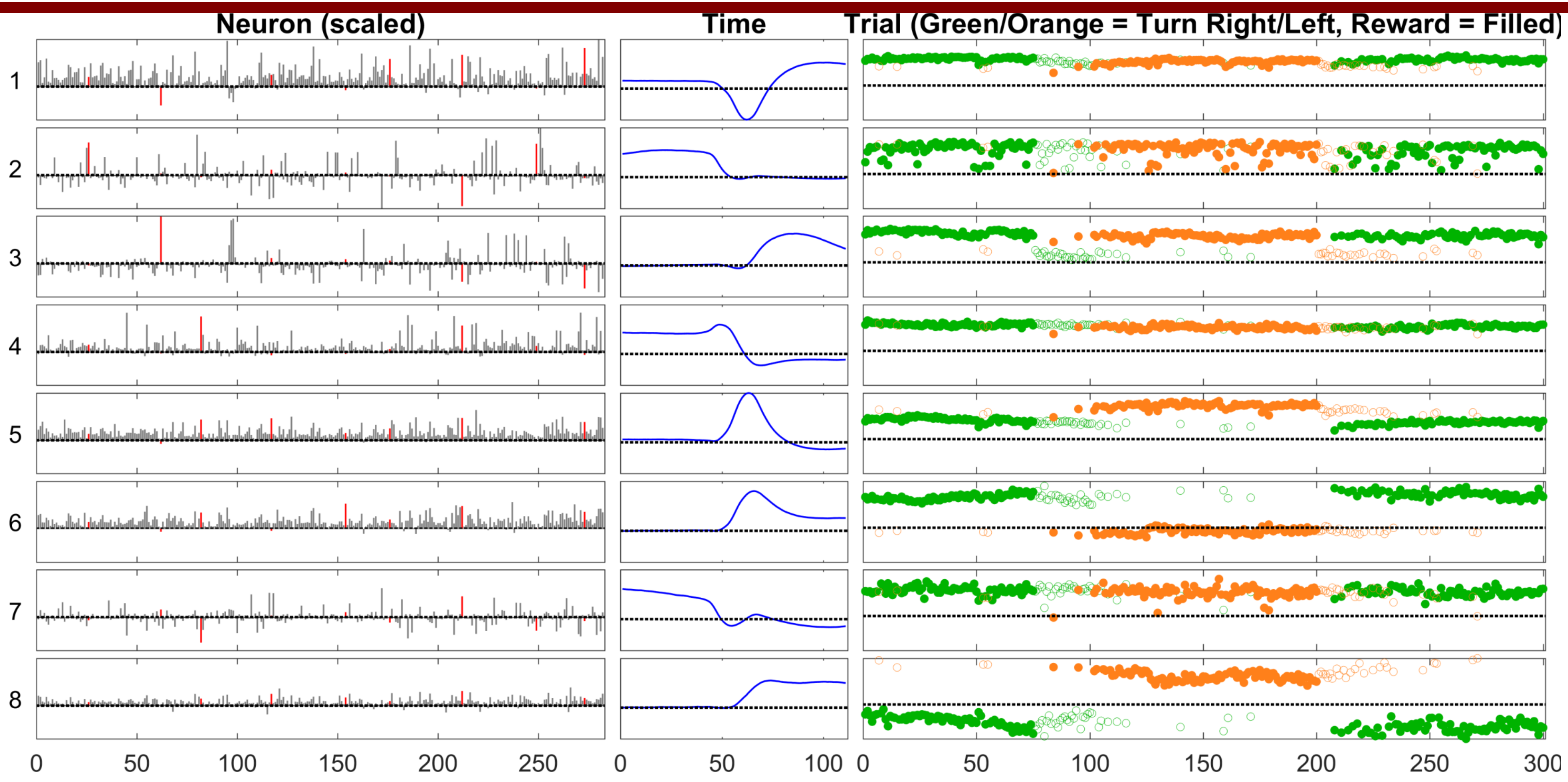
Hong, Kolda, Duersch, SIAM Review, 2020

Visualization of CP Tensor Decomposition Shows the Factors (Vectors)

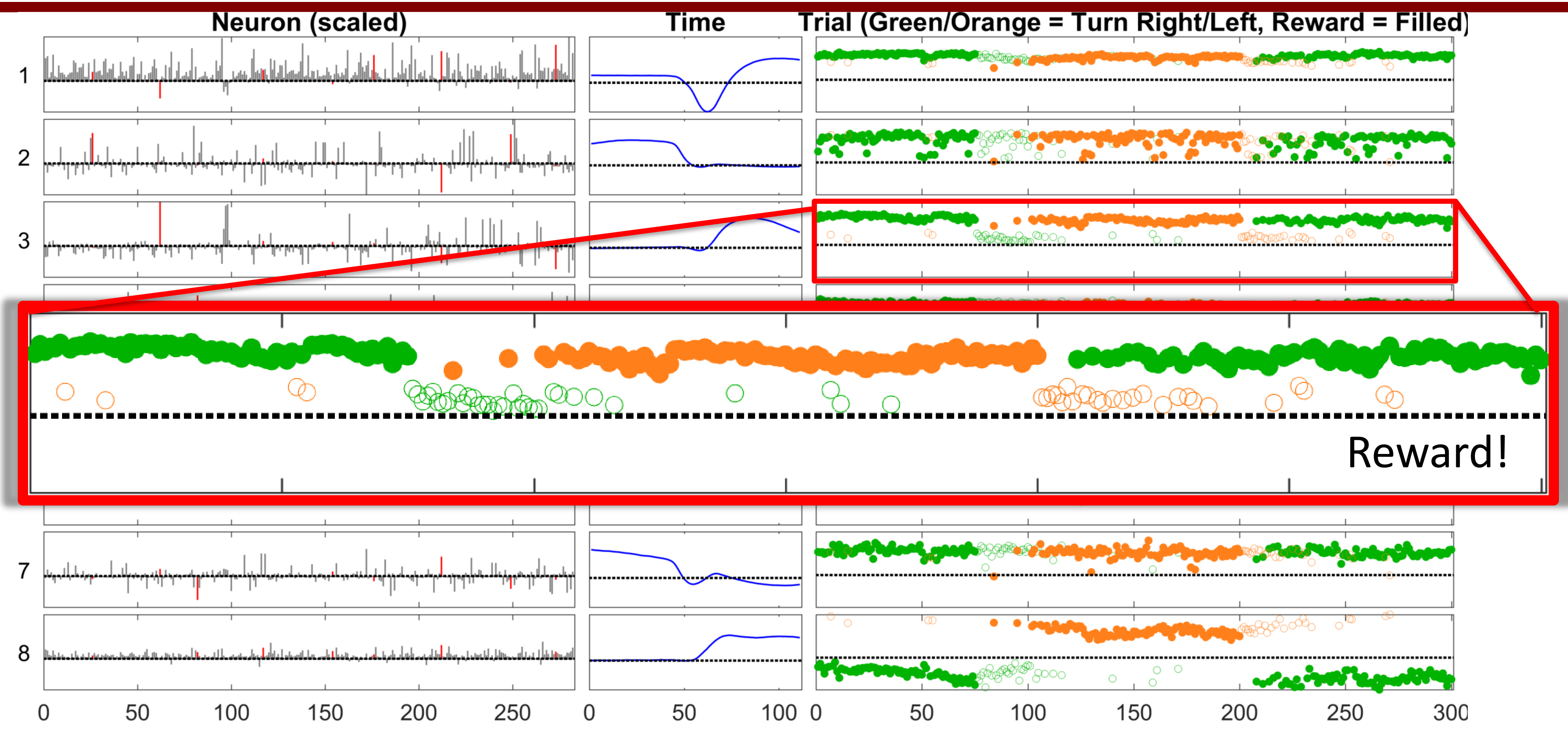


Hong, Kolda, Duersch, SIAM Review, 2020

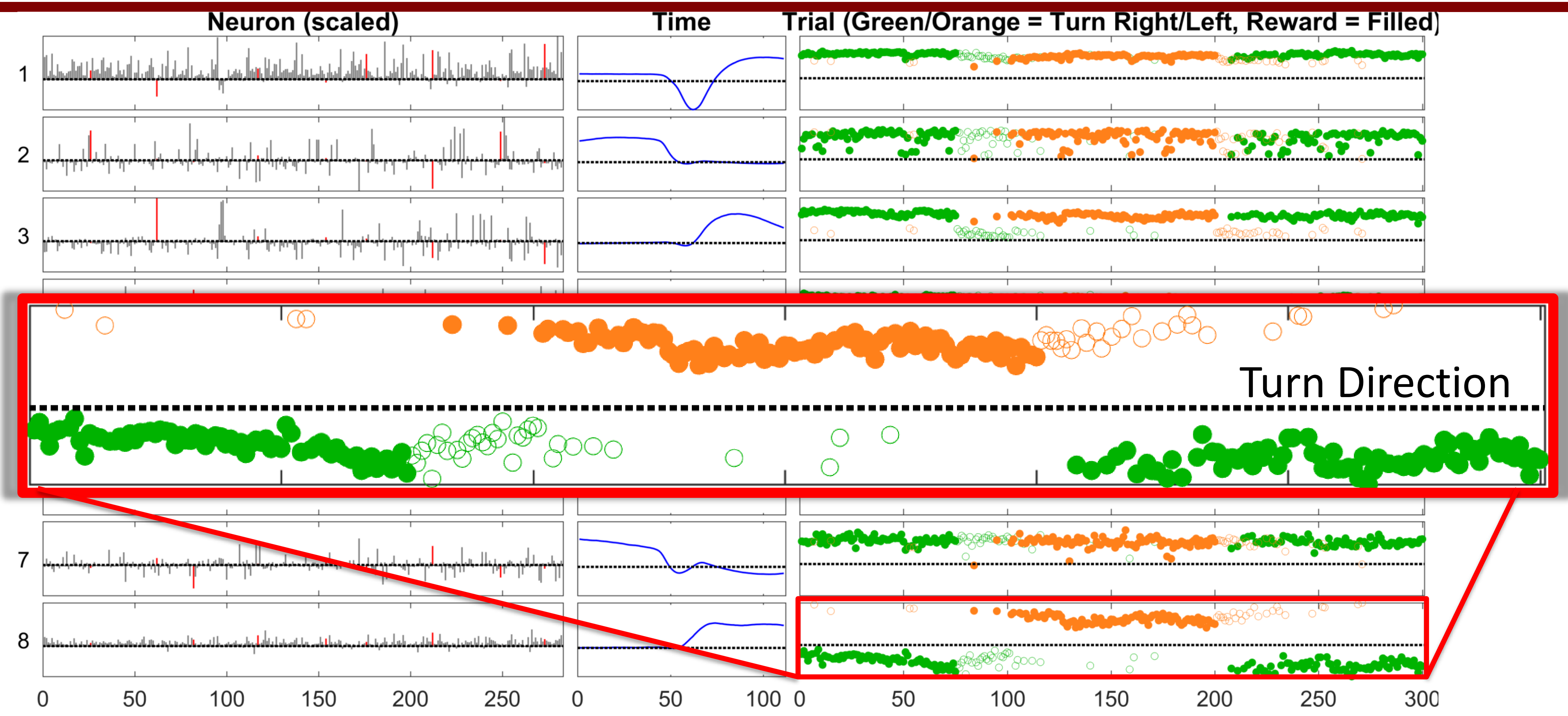
CP Decomposition of Mouse Data



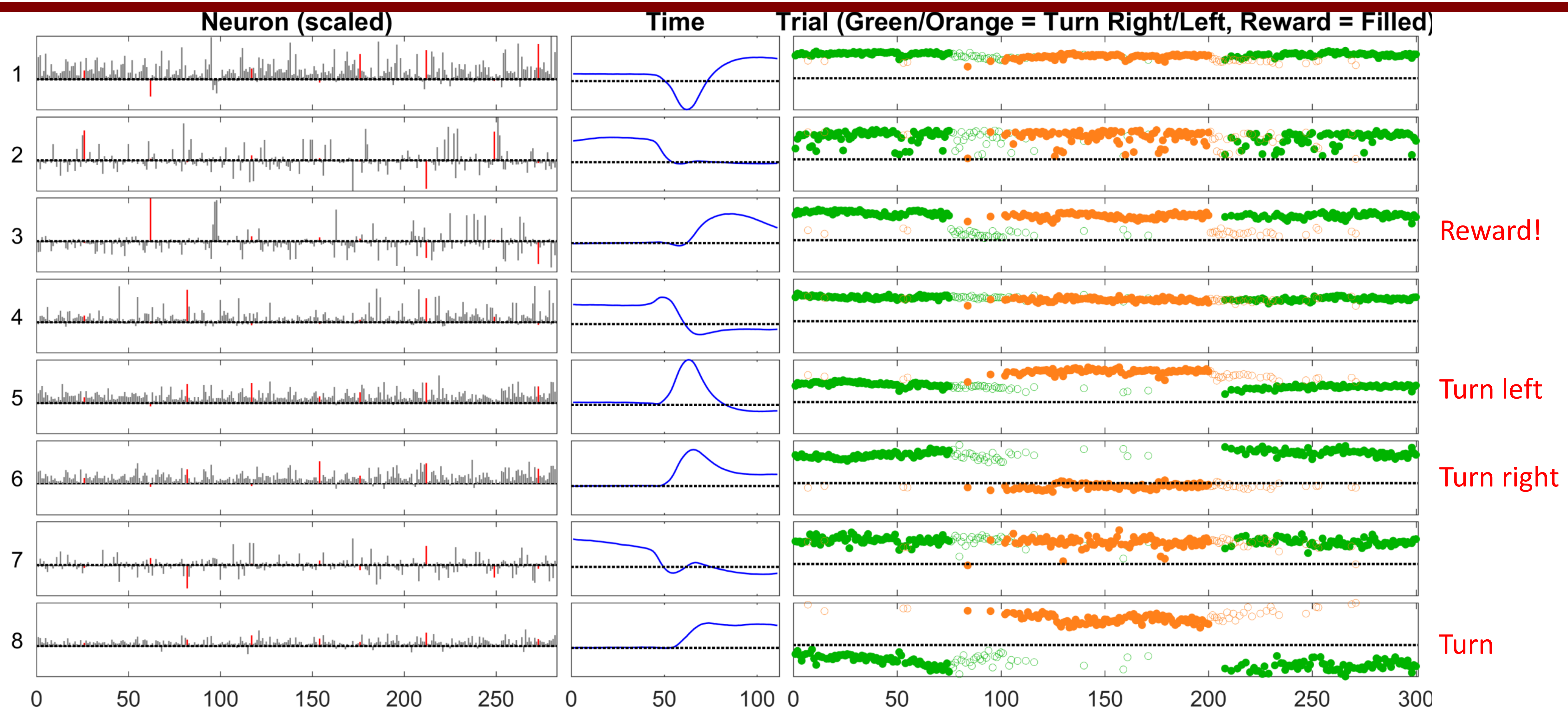
CP Tensor Decomposition "Sees" Reward

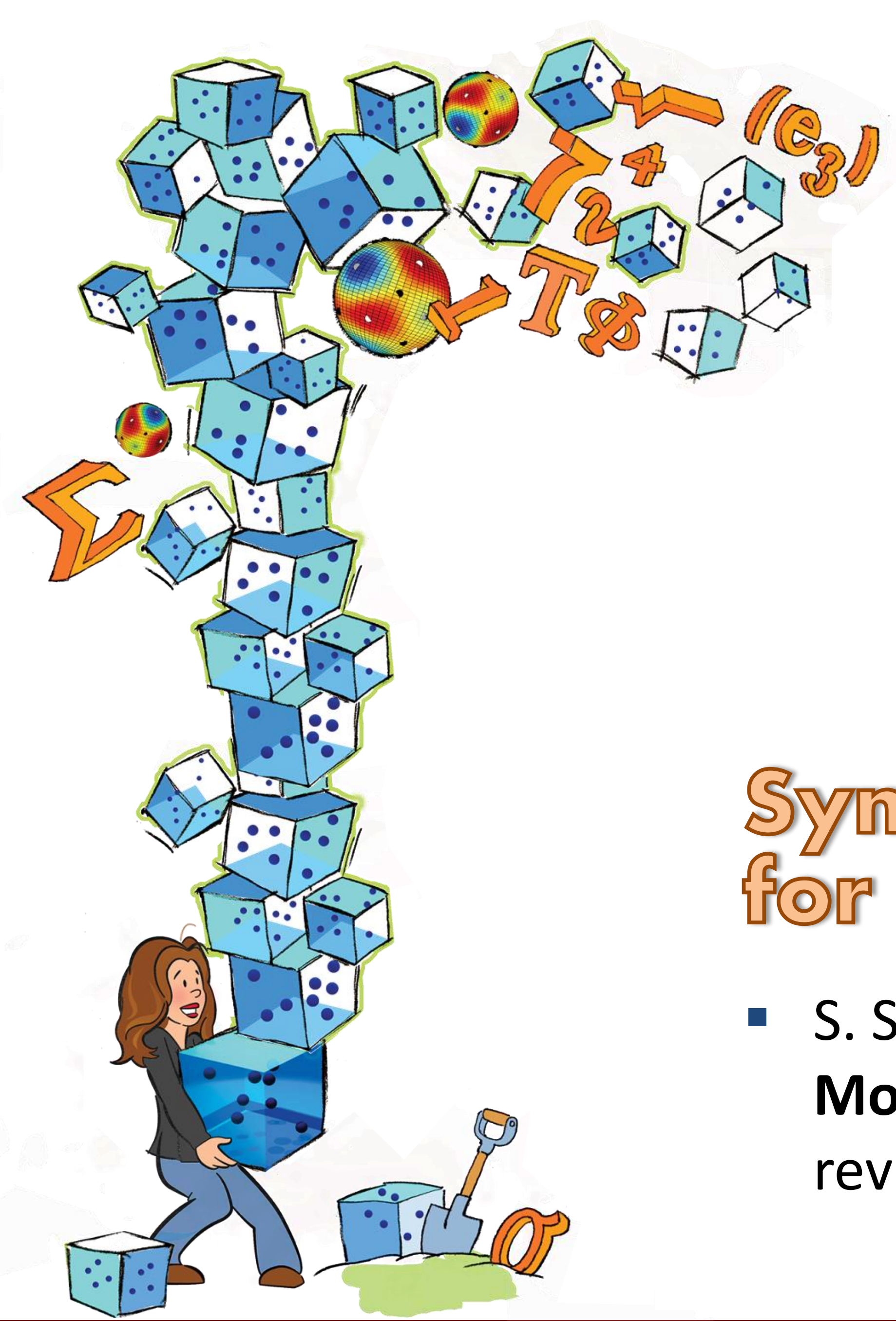


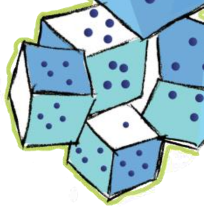
CP Tensor Decomposition "Sees" Turn Direction



CP Tensor Decomposition Yields Interpretation of a Complex Dataset







Sam Sherman
Notre Dame

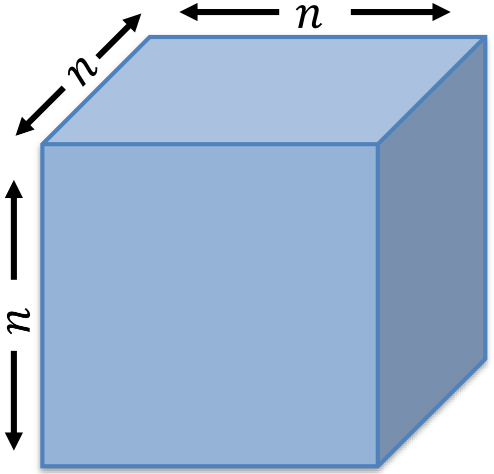
Tammy Kolda
Sandia

Symmetric CP Tensor Factorization for (Symmetric) Moment Tensors

- S. Sherman, T. G. Kolda. **Estimating Higher-Order Moments Using Symmetric Tensor Decomposition**, revised April 2020, http://arxiv.org/abs/1911.03813

Symmetric Tensor Entries Invariant Under Permutation of Indices

A tensor is <u>symmetric</u> if its entries are invariant under permutation of the indices



For d-way tensor, of dimension n, number of unique entries is:

$$\binom{n+d-1}{d} \approx \frac{n^d}{d!}$$

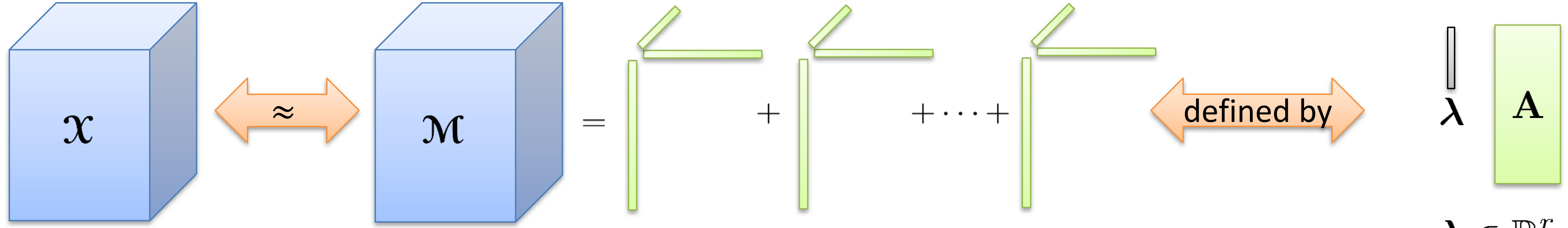
Example 1.2 from Nie (2014)
$3 \times 3 \times 3$ symmetric tensor (10 distinct entries)

$$\mathcal{X} = \left(\begin{array}{rrr|rrr|rrr} 7 & -3 & 9 & -3 & 13 & 20 & 9 & 20 & 19 \\ -3 & 13 & 20 & 13 & -27 & 6 & 20 & 6 & 6 \\ 9 & 20 & 19 & 20 & 6 & 6 & 19 & 6 & 45 \end{array} \right)$$

$$\begin{aligned} x(1,1,1) &= 7 & x(1,3,3) &= 19 \\ x(1,1,2) &= -3 & x(2,2,2) &= -27 \\ x(1,1,3) &= 9 & x(2,2,3) &= 6 \\ x(1,2,2) &= 13 & x(2,3,3) &= 6 \\ x(1,2,3) &= 20 & x(3,3,3) &= 45 \end{aligned}$$

Symmetric CP Tensor Decomposition Has Single Factor Matrix

Symmetric Data | Symmetric CP Model | Sum of r Symmetric Outer Product Tensors | Single Factor Matrix



$\mathcal{X} \in \mathbb{R}^{\overbrace{n \times n \times \cdots \times n}_{\text{order } d}}$

$\mathcal{M} \in \mathbb{R}^{n \times n \times \cdots \times n}$

$\boldsymbol{\lambda} \in \mathbb{R}^{r}$

$\mathbf{A} \in \mathbb{R}^{n \times r}$

Model Rank

$$m_{i_1 i_2 \dots i_d} = \sum_{j=1}^{r} \lambda_j \prod_{k=1}^{d} a(i_k, j)$$

Symmetric Outer Product

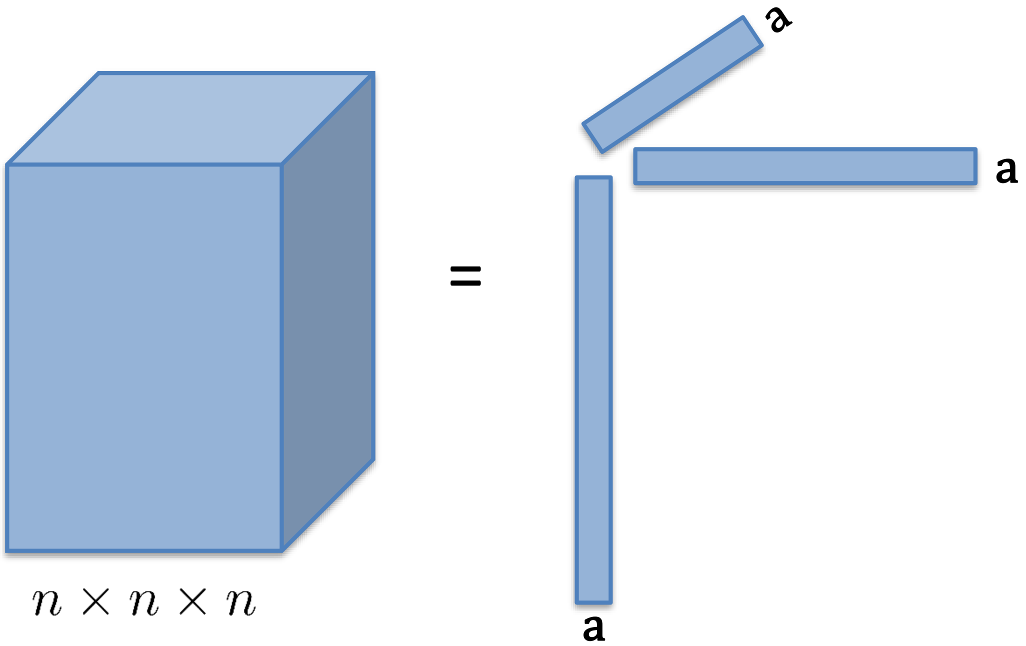
Given **a vector**:

$$\mathbf{a} \in \mathbb{R}^n$$

The **outer product** is

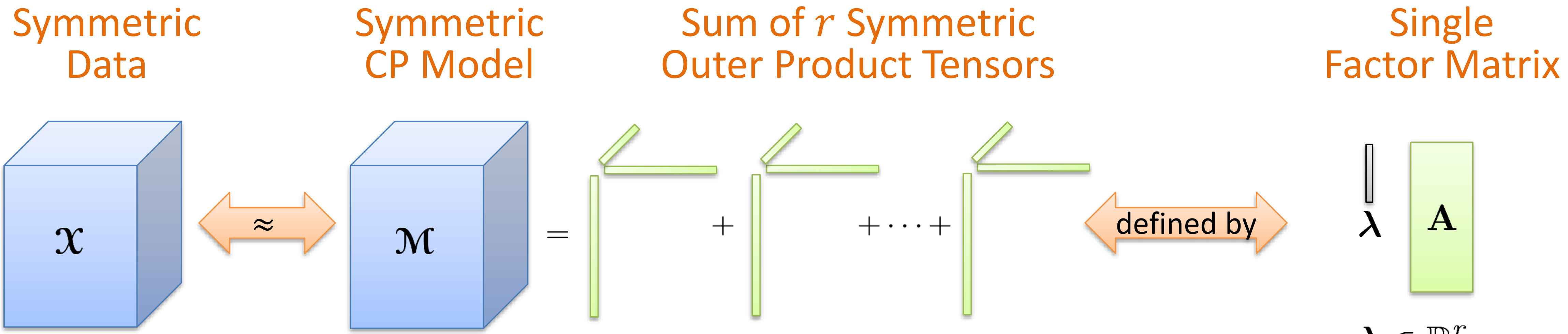
$$\mathcal{P} = \mathbf{a}^{\otimes d} \in \mathbb{R}^{n \times n \times \cdots \times n}$$

$$\mathbf{a}^{\otimes 3} \equiv \mathbf{a} \otimes \mathbf{a} \otimes \mathbf{a}$$



$$(\mathbf{a}^{\otimes 3})_{ijk} = a_i a_j a_k$$

Model Expressed as Sum of Symmetric Outer Products



$$\mathcal{M} = \sum_{j=1}^{r} \lambda_j \, \mathbf{a}_j^{\otimes d} \in \mathbb{R}^{n \times n \times \cdots \times n}$$

$\boldsymbol{\lambda} \in \mathbb{R}^r$

$\mathbf{A} \in \mathbb{R}^{n \times r}$

Model Rank

Symmetric Tensor Rank & Decomposition

Example 1.2 from Nie (2014)
3 × 3 × 3 symmetric tensor (10 distinct entries)

$$\mathcal{X} = \left(\begin{array}{ccc|ccc|ccc} 7 & -3 & 9 & -3 & 13 & 20 & 9 & 20 & 19 \\ -3 & 13 & 20 & 13 & -27 & 6 & 20 & 6 & 6 \\ 9 & 20 & 19 & 20 & 6 & 6 & 19 & 6 & 45 \end{array}\right)$$

$$\operatorname{rank}(\mathcal{X}) = \min \{ r \mid \mathcal{X} = \mathbf{a}_1^{\otimes d} + \cdots + \mathbf{a}_r^{\otimes d} \}$$

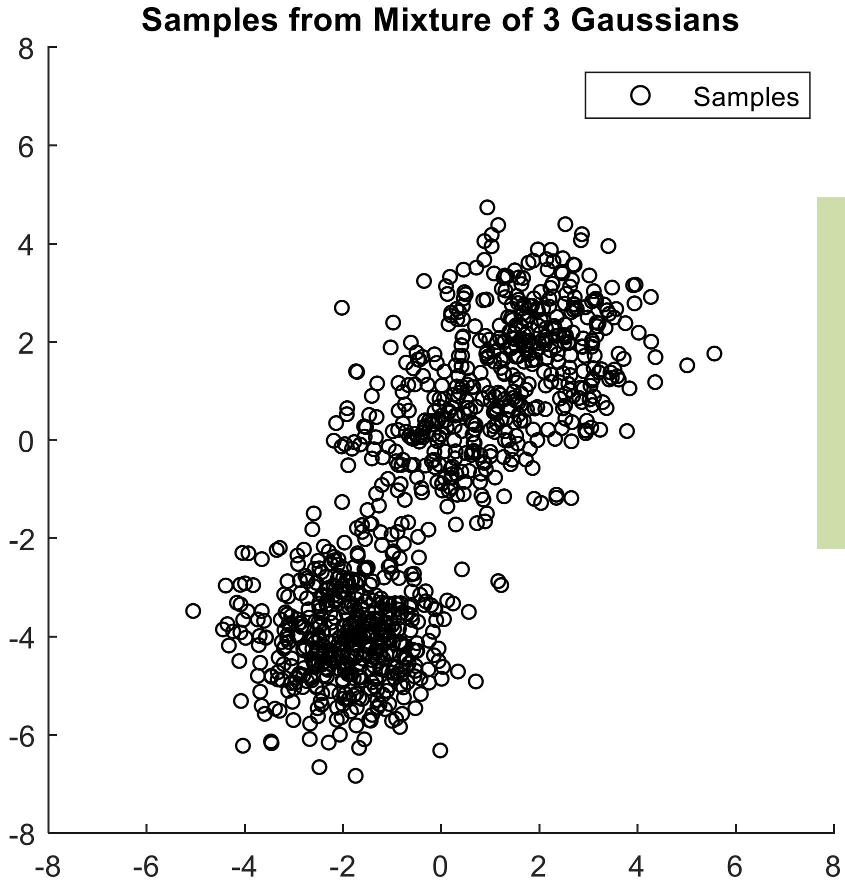
Rank decomposition

$$\mathcal{X} = 2 \cdot \begin{bmatrix} 1 \\ -2 \\ -1 \end{bmatrix}^{\otimes 3} + 5 \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}^{\otimes 3} - \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix}^{\otimes 3}$$

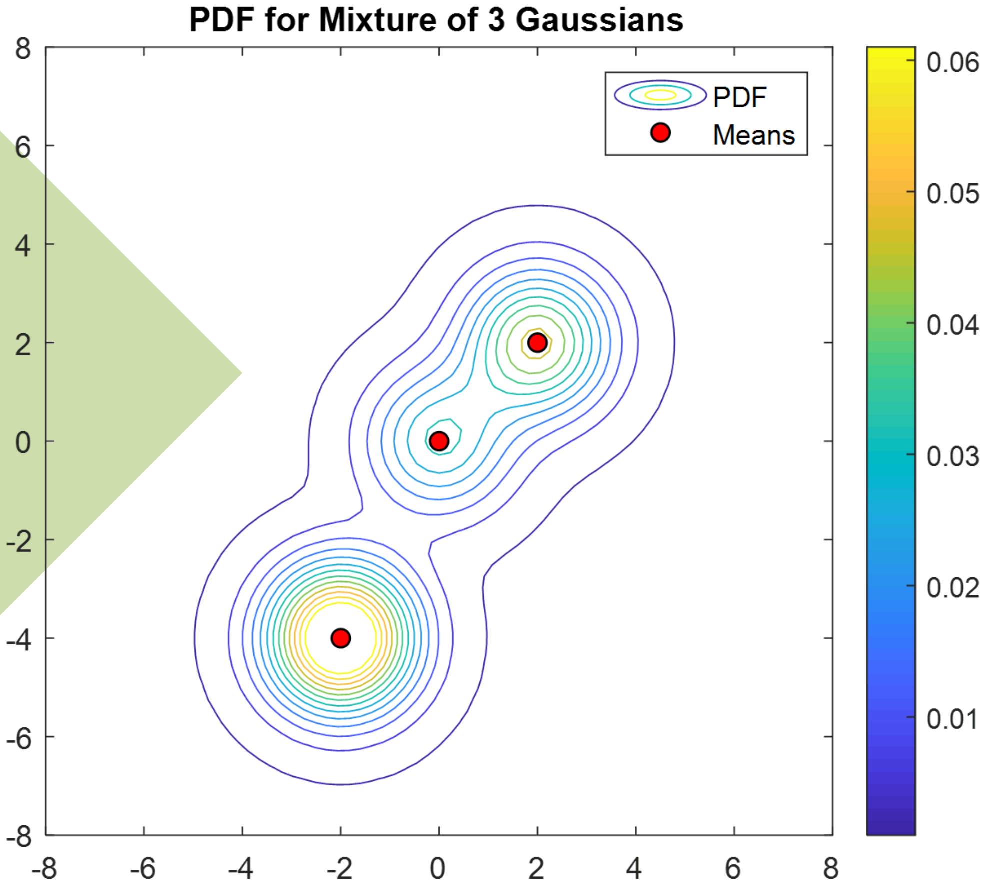
- Symmetric tensor rank
  - For any given tensor, NP-hard to compute its rank (Hillar & Lim, 2013)
  - Typical rank known over $\mathbb{C}$ (Comon, Golub, Lim, Mourraine, 2008)
  - In practice, trial and error!
- Symmetric tensor decomposition
  - Waring decomposition (Landsberg, 2012; Oeding & Ottaviani, 2013)
  - Gröbner bases algebraic methods or numerical root-finding method (Nie, 2014)
  - Direct optimization formulation (Kolda, 2015)
  - Subspace power method (Kileel & Pereira, 2019)

Moment Tensors Arise in Inference of Gaussian Mixture Models (GMMs)

For ease of illustration, we focus on $n = 2$ dimensions.
Generally interested in much higher dimensions, i.e, $n = 500$!



Given just the samples (point cloud), can we recover the means?

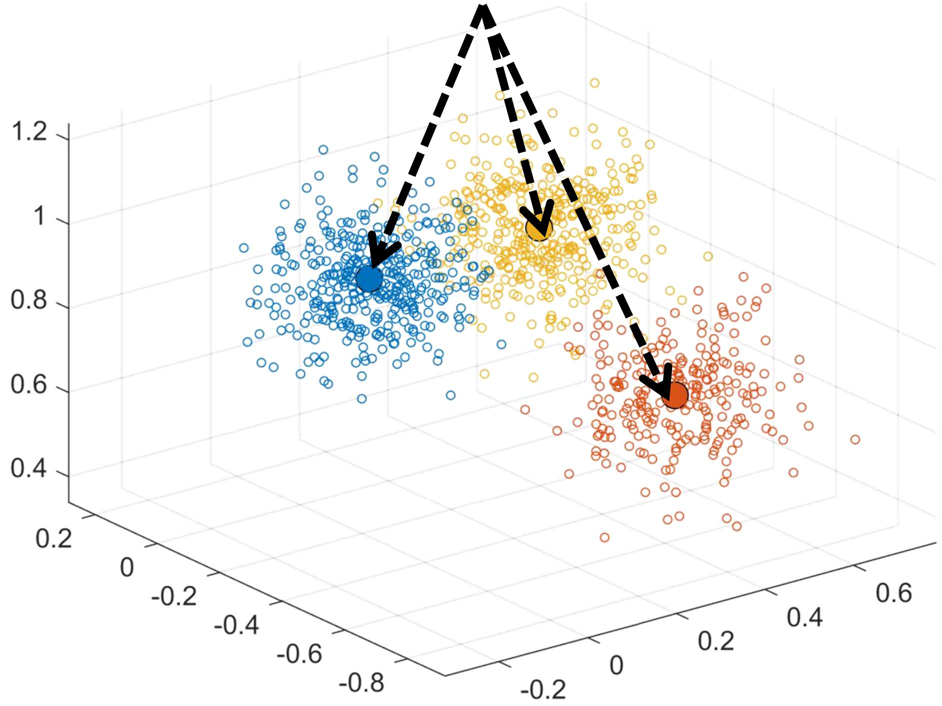


Machine Learning Motivation: Observations from Unknown Mixture of Gaussians

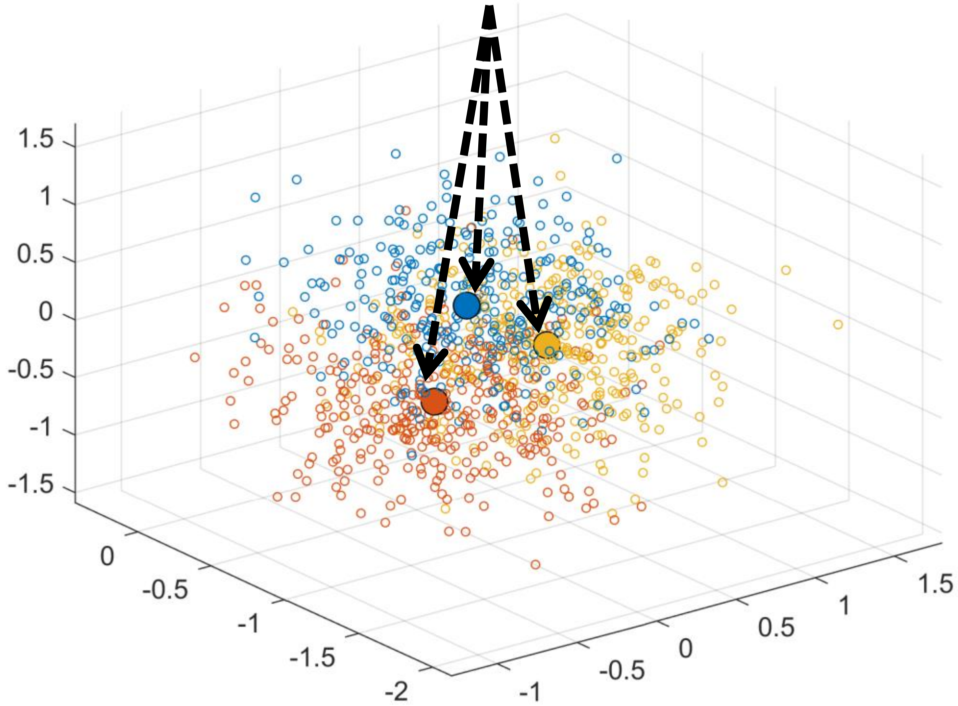
We observe p random vectors of length n coming from a mixture of r Gaussian distributions.

Can we recover the means of the Gaussians?

Easy: Means Well Separated



Hard: Means Close Together



For these pictures: $p = 1000, n = 3, r = 3$. Means shown as filled in larger circles. Samples as open circles.

We care about larger values of n!

Moment Structure for Spherical GMMs Corresponds to CP Model

Data Model: $V \sim \mathcal{N}(\boldsymbol{\mu}_\xi, \sigma^2 \mathbf{I}), \quad \xi \sim \textsc{Multi}(w_1, \ldots, w_r)$

Multivariate Normal

Probability to select jth center is w_j

3rd-order Moment: $\boxed{\mathbb{E}[V^{\otimes 3}]} + O(\sigma^2) = \boxed{\sum_{j=1}^{r} w_j \boldsymbol{\mu}_j^{\otimes 3}}$

Can also do higher - order moments

Calculate empirically from data

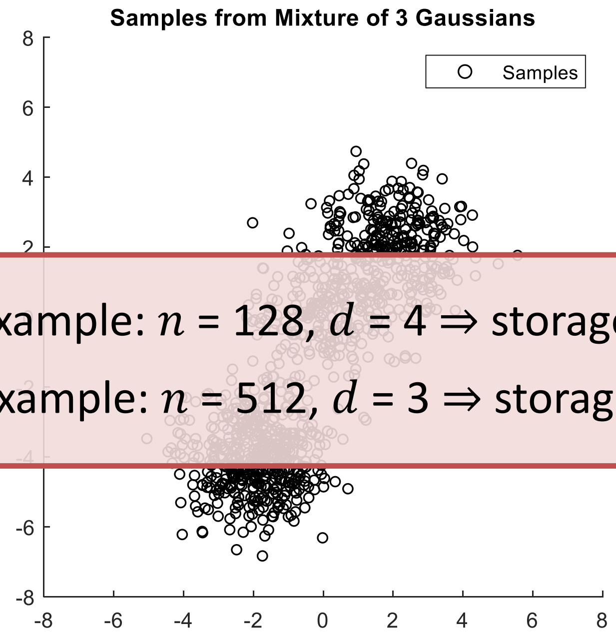
CP-like Model

$$\mathcal{X} = \frac{1}{p} \sum_{\ell=1}^{p} \mathbf{v}_\ell^{\otimes 3}$$

$$\mathcal{M} = \sum_{j=1}^{r} \lambda_j \, \mathbf{a}_j^{\otimes 3}$$

Bottlenecks:
$O(pn^d)$ to compute,
$O(n^d)$ to store

Hsu and Kakade, 2013



Example: n = 128, d = 4 ⇒ storage = 2 GB

Example: n = 512, d = 3 ⇒ storage = 1 GB

Simplifying assumptions for this work

$$\|\boldsymbol{\mu}_j\|_2 = 1 \; \forall j \in [r]$$

$$\omega_j = \frac{1}{r} \; \forall j \in [r]$$

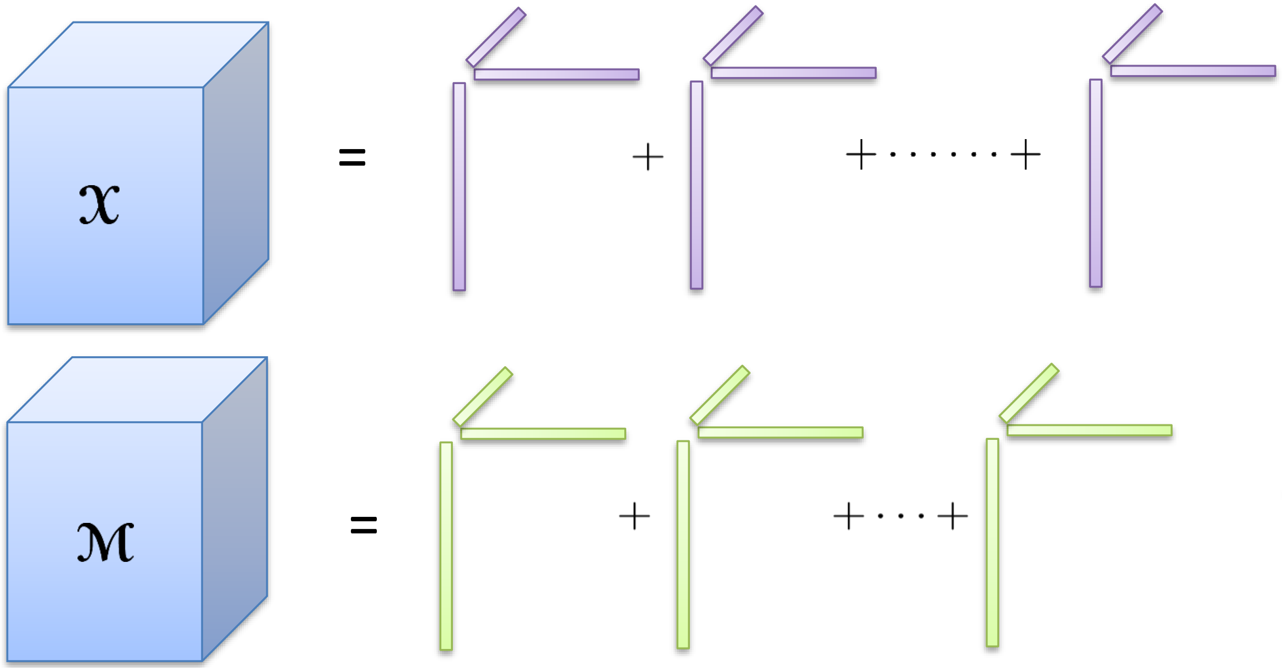
Our Focus Today: Accelerating Computation for Special Case of Moment Tensors

Sandia National Laboratories

$$\mathcal{X} = \frac{1}{p} \sum_{\ell=1}^{p} \mathbf{v}_\ell^{\otimes d}$$

Symmetric Data

Symmetric CP Model



$$\mathcal{M} = \sum_{j=1}^{r} \lambda_j \, \mathbf{a}_j^{\otimes d}$$

defined by

defined by

Given Observations

$\mathbf{V} \in \mathbb{R}^{n \times p}$

Want to Find Compact Representation

$\mathbf{A} \in \mathbb{R}^{n \times r}$

$r \ll p$

Optimization Approach for Symmetric CP of Symmetric Tensor Requires TTSV

Optimization Problem

$$\min_{\boldsymbol{\lambda},\mathbf{A}} F(\mathcal{X},\mathcal{M}) \equiv \frac{1}{2}\|\mathcal{X}-\mathcal{M}\|^2 \text{ where } \mathcal{M} = \sum_{j=1}^{r} \lambda_j\, \mathbf{a}_j^{\otimes d}$$

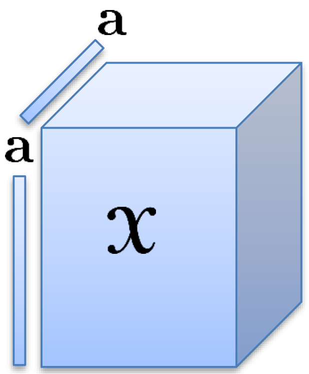
Gradients $\forall j \in [r]$

$$\frac{\partial F}{\partial \mathbf{a}_j} = -d\lambda_j \boxed{\mathcal{X}\mathbf{a}_j^{d-1}} + d\lambda_j \sum_{k=1}^{r} \lambda_k \langle \mathbf{a}_j, \mathbf{a}_k\rangle^{d-1} \mathbf{a}_k$$

$$\frac{\partial F}{\partial \lambda_j} = -\mathcal{X}\mathbf{a}_j^{d} + \sum_{k=1}^{r} \lambda_k \langle \mathbf{a}_j, \mathbf{a}_k\rangle^{d}$$

Plug function and gradient into favorite optimization method. My favorite: L-BFGS.

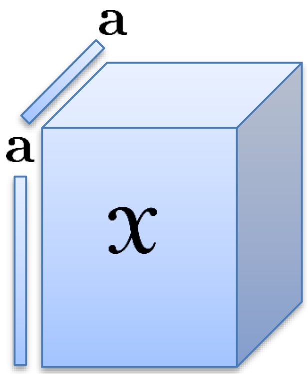
Bottleneck is TTSV which costs $O(n^d)$



Key Kernel: Tensor Times Single Vector (TTSV)

$$\left(\mathcal{X}\mathbf{a}^{d-1}\right)_{i_1} = \sum_{i_2=1}^{n} \cdots \sum_{i_d=1}^{n} \left(x_{i_1 i_2 \dots i_d} \prod_{k=2}^{d} a_{i_k} \right) \quad \forall i_1 \in [n]$$

Key Result: Implicit Computation of TTSV



TTSV Definition: $$\left(\mathcal{X}\mathbf{a}^{d-1}\right)_{i_1} = \sum_{i_2=1}^{n} \cdots \sum_{i_d=1}^{n} \left(x_{i_1 i_2 \dots i_d} \prod_{k=2}^{d} a_{i_k} \right) \quad \forall i_1 \in [n]$$

Lemma. Let $\mathcal{X} = \frac{1}{p} \sum_{\ell=1}^{p} \mathbf{v}_\ell^{\otimes d}$ and $\mathbf{V} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 & \cdots & \mathbf{v}_p \end{bmatrix}$, then

$$\mathcal{X}\mathbf{a}^{d-1} = \frac{1}{p} \mathbf{V} \left[\mathbf{V}^\intercal \mathbf{a} \right]^{d-1}$$

Entry-wise Power

$O(n^d)$

$O(pn)$

Minimal Change in Function/Gradient Calculation Replaces Expensive TTSV

function $\textsc{FG\_Explicit}(\mathcal{X}, \boldsymbol{\lambda}, \mathbf{A}, \alpha)$
  for $j = 1, \dots, r$, **do** $\mathbf{y}_j = \mathcal{X}\mathbf{a}_j^{d-1}$, **end**
  for $j = 1, \dots, r$, **do** $w_j = \mathbf{a}_j^T \mathbf{y}_j$, **end**
  $\mathbf{B} = \mathbf{A}^T \mathbf{A}$
  $\mathbf{C} = [\mathbf{B}]^{d-1}$
  $\mathbf{u} = (\mathbf{B} * \mathbf{C})\boldsymbol{\lambda}$
  $f = \alpha + \boldsymbol{\lambda}^T \mathbf{u} - 2\mathbf{w}^T \boldsymbol{\lambda}$
  $\mathbf{g}_{\boldsymbol{\lambda}} = -2(\mathbf{w} - \mathbf{u})$
  $\mathbf{G}_{\mathbf{A}} = -2d(\mathbf{Y} - \mathbf{A}\mathbf{D}_{\boldsymbol{\lambda}}\mathbf{C})\mathbf{D}_{\boldsymbol{\lambda}}$
  return $f, \mathbf{g}_{\boldsymbol{\lambda}}, \mathbf{G}_{\mathbf{A}}$
end function

function $\textsc{FG\_Implicit}(\mathbf{V}, \boldsymbol{\lambda}, \mathbf{A}, \alpha)$
  $\mathbf{Y} = \frac{1}{p}\mathbf{V}[\mathbf{V}^{\intercal}\mathbf{A}]^{d-1}$
  for $j = 1, \dots, r$, **do** $w_j = \mathbf{a}_j^T \mathbf{y}_j$, **end**
  $\mathbf{B} = \mathbf{A}^T \mathbf{A}$
  $\mathbf{C} = [\mathbf{B}]^{d-1}$
  $\mathbf{u} = (\mathbf{B} * \mathbf{C})\boldsymbol{\lambda}$
  $f = \alpha + \boldsymbol{\lambda}^T \mathbf{u} - 2\mathbf{w}^T \boldsymbol{\lambda}$
  $\mathbf{g}_{\boldsymbol{\lambda}} = -2(\mathbf{w} - \mathbf{u})$
  $\mathbf{G}_{\mathbf{A}} = -2d(\mathbf{Y} - \mathbf{A}\mathbf{D}_{\boldsymbol{\lambda}}\mathbf{C})\mathbf{D}_{\boldsymbol{\lambda}}$
  return $f, \mathbf{g}_{\boldsymbol{\lambda}}, \mathbf{G}_{\mathbf{A}}$
end function

Experimental Difference in Per-Iteration Cost of Implicit versus Explicit

Rank-r Symmetric CP Tensor Factorization
for d-way tensor of size n

$$r < n < p$$

Method	Storage	Per-Iteration
Explicit	$O(n^d)$	$O(rn^d)$
Implicit	$O(pn)$	$O(pnr)$

Implicit cheaper if $p < O(n^{d-1})$

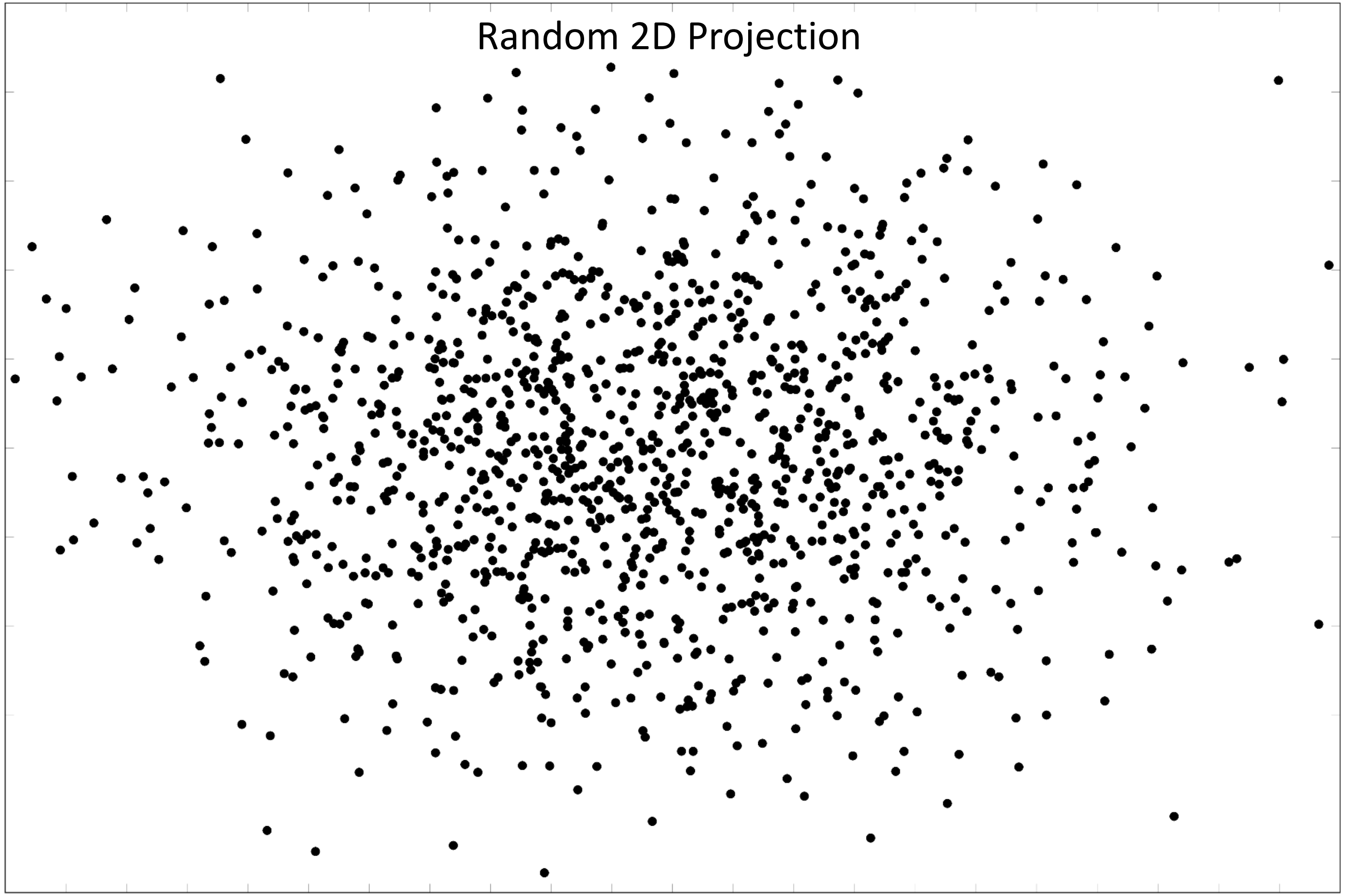
Average cost per iteration for $r = 5$ over 10 runs

d	n	p	n^{d-1}	Explicit	Implicit
3	75	3750	5625	5e-4	8e-4
3	375	3750	140625	2e-2	5e-3
4	75	3750	421875	1e-2	9e-4

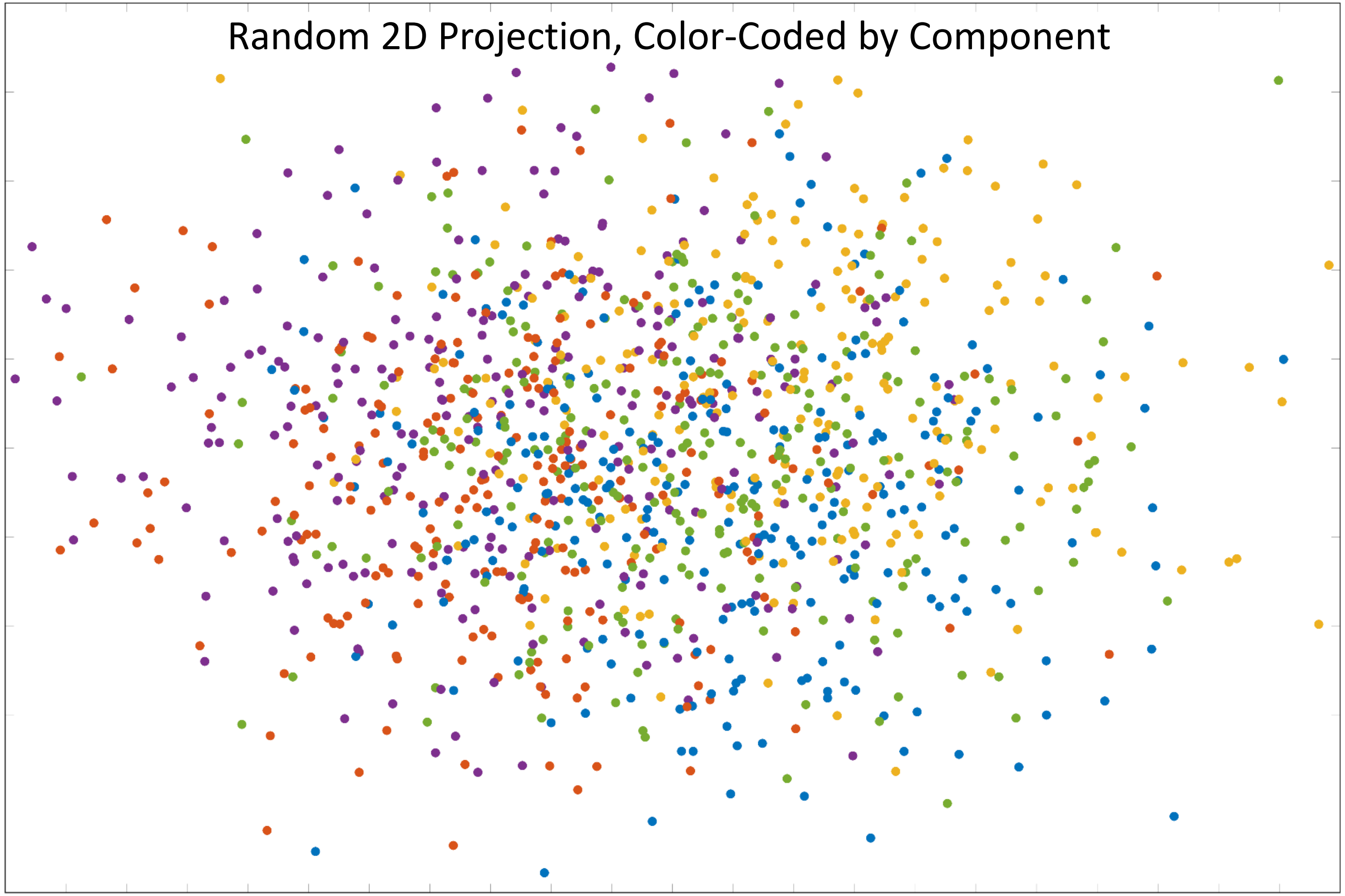
GMM Example with r=5 (mixtures), n=500 (dimension) and p=750 (observations)

For $d = 3$, explicit method requires 1 GB storage

For d =4, explicit method requires 500 GB storage



GMM Example with r=5 (mixtures), n=500 (dimension) and p=750 (observations)



GMM Example with r=5 (mixtures), n=500 (dimension) and p=750 (observations)

$$\boldsymbol{\mu}_j \in \mathbb{R}^{500}$$

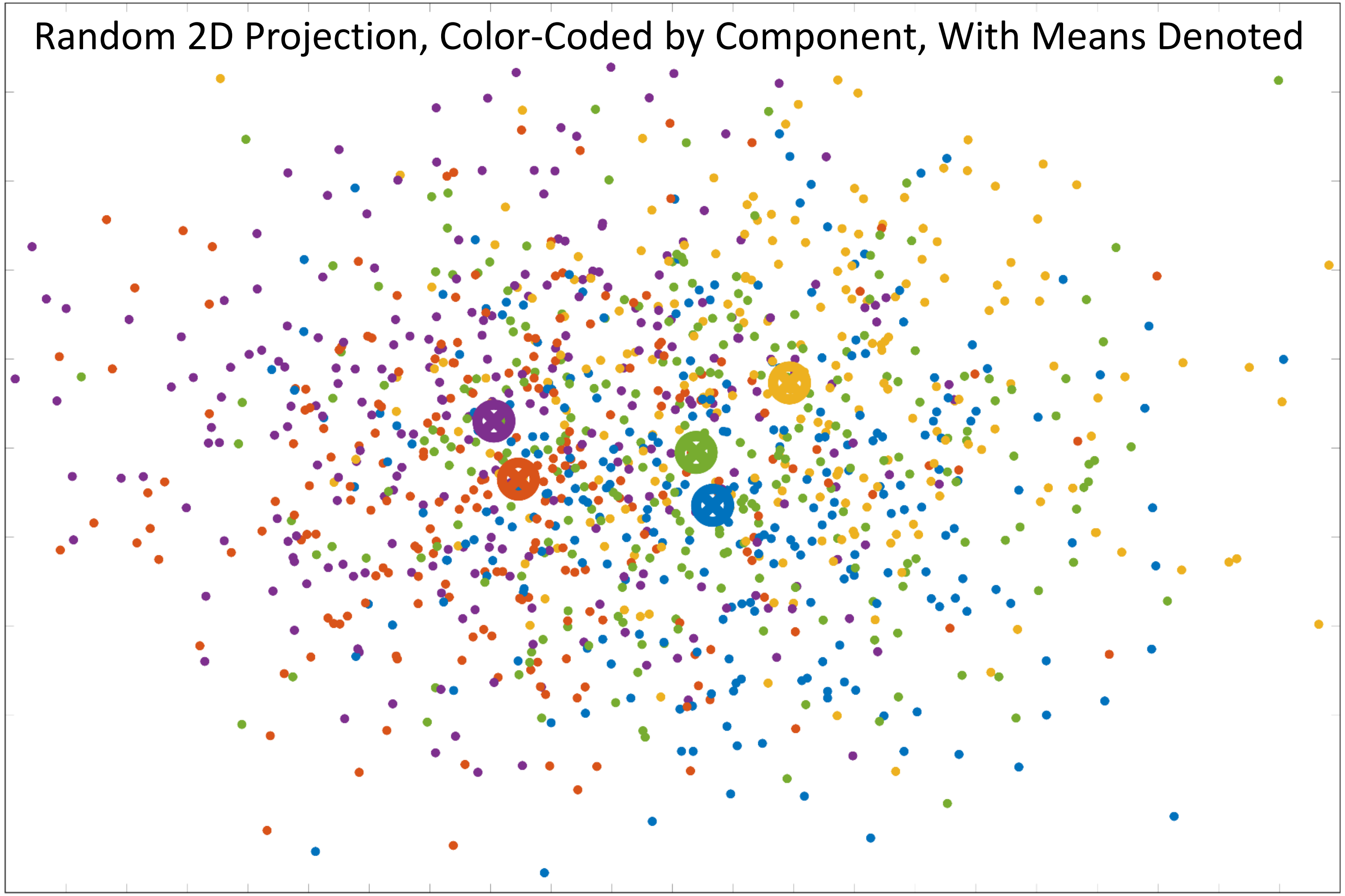
$$\|\boldsymbol{\mu}_j\|_2 = 1$$

$$\forall j \in [r]$$

$$\boldsymbol{\mu}_j^T \boldsymbol{\mu}_k = 0.5$$

$$\forall j \neq k$$

Shown here:
$\sigma = 0.1$



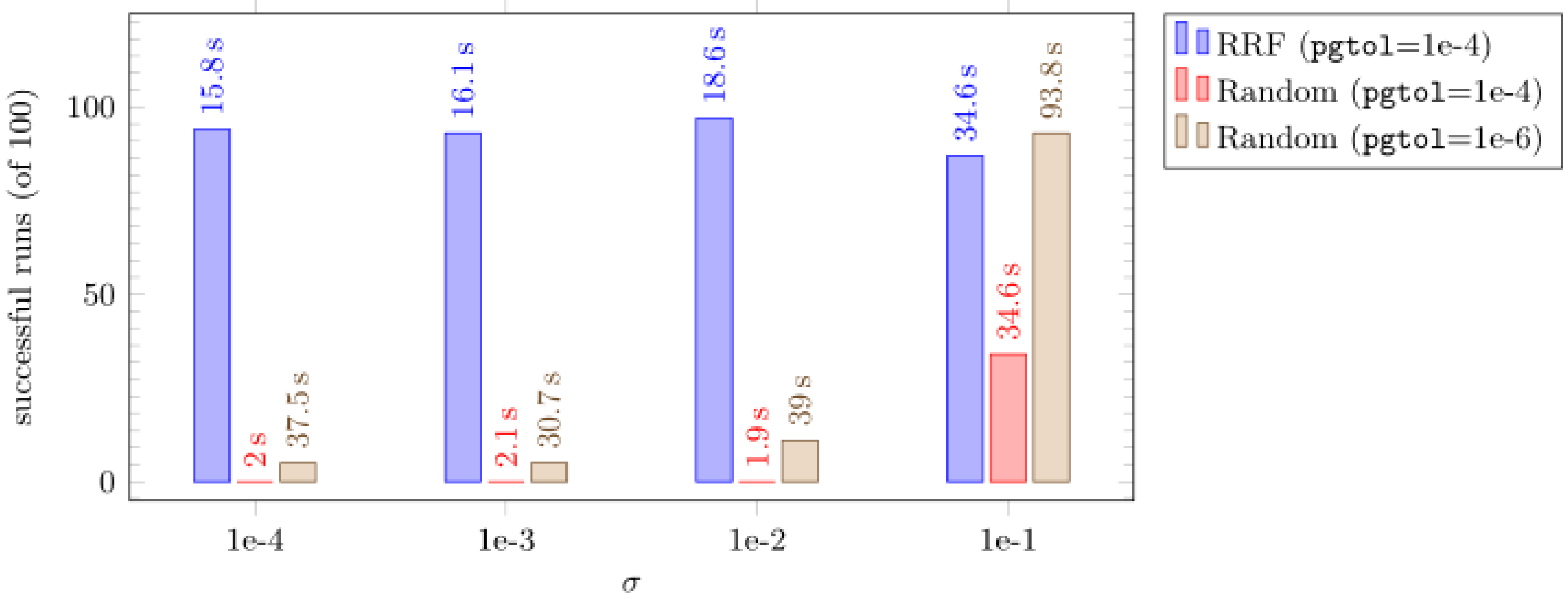
Choosing Starting Guess Within Range of Observations is Key!

Randomized Range Finder (RRF): $\mathbf{A}_0 = \mathbf{V}\boldsymbol{\Omega},\ \boldsymbol{\Omega} \sim \mathcal{N}(0,1)^{p\times\hat{r}}$

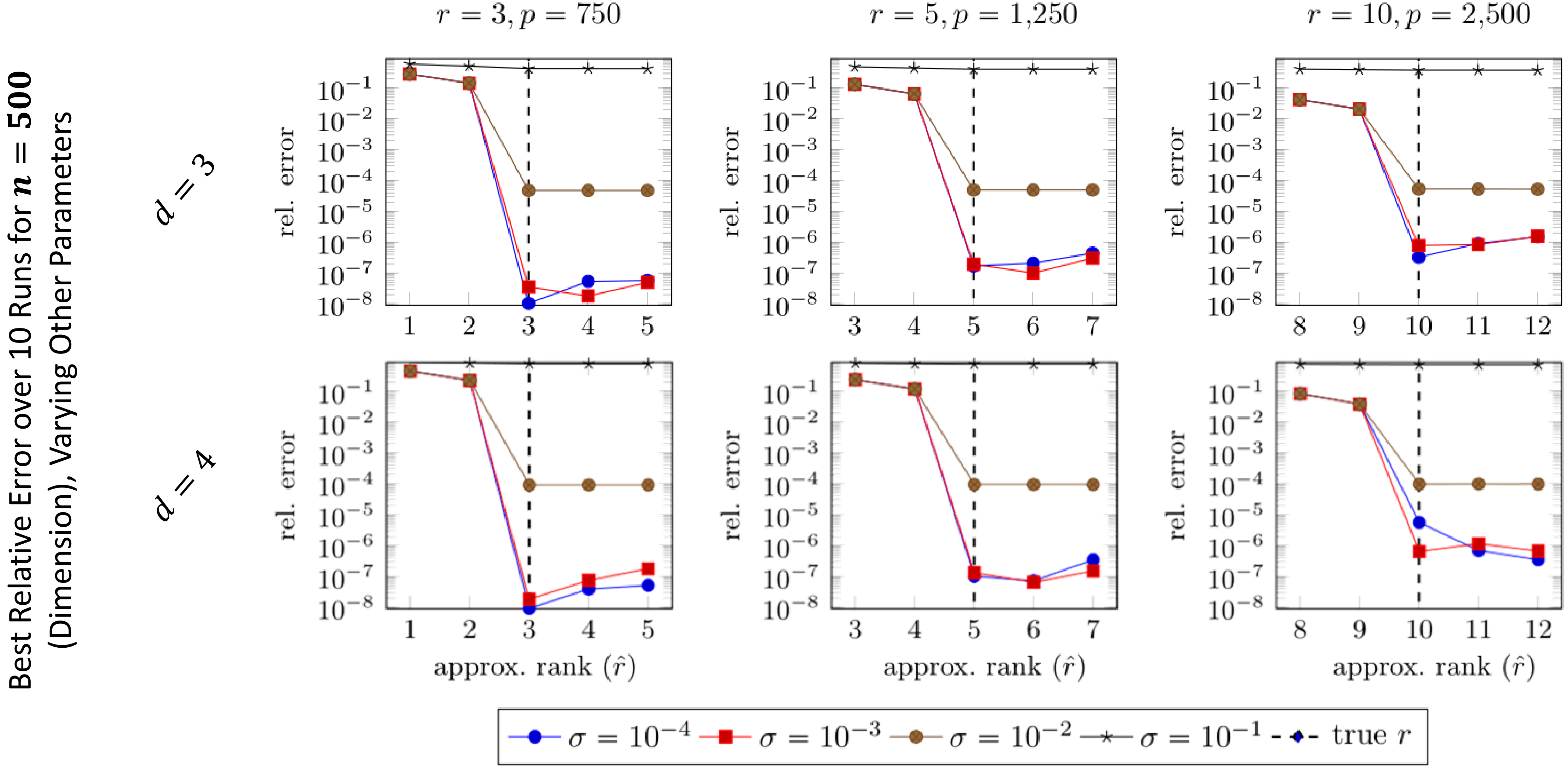
Random: $\mathbf{A}_0 \sim \mathcal{N}(0,1)^{n\times\hat{r}}$

[with columns normalized in both cases]

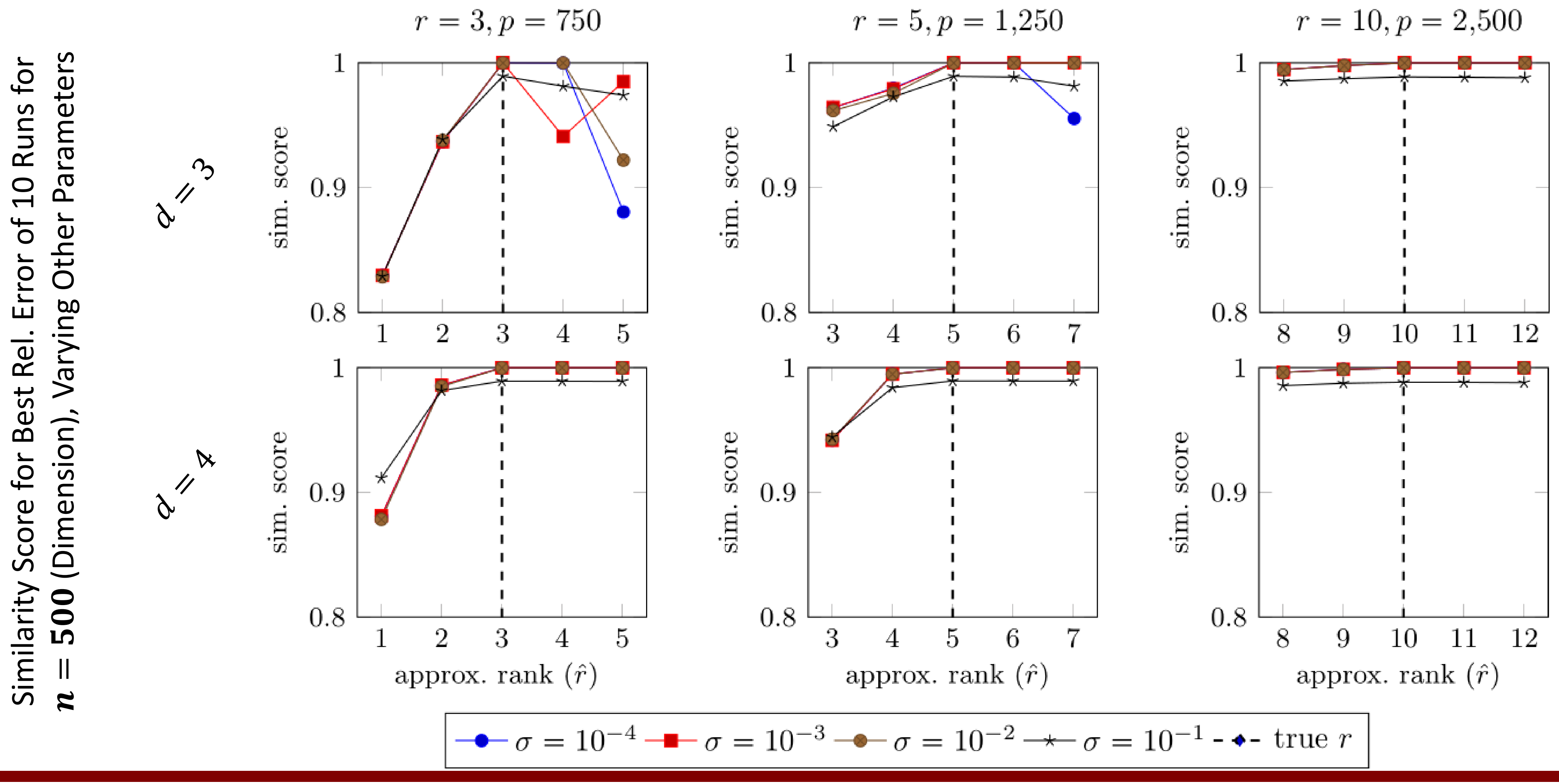
Results of computing $\hat{r} = 3$ approximation for moment tensor of order $d = 3$, with $r = 3$ components, $n = 500$ dimensions, and $p = 750$ observations



Minimum Generally Achieved at "True" Rank



Identifies "True" Solution Even When Relative Error Poor



Identified Factors for $\hat{r}$=5 with $r = 5, p = 750, n = 500, \sigma = 0.1$

$\boldsymbol{\mu}_j \in \mathbb{R}^{500}$

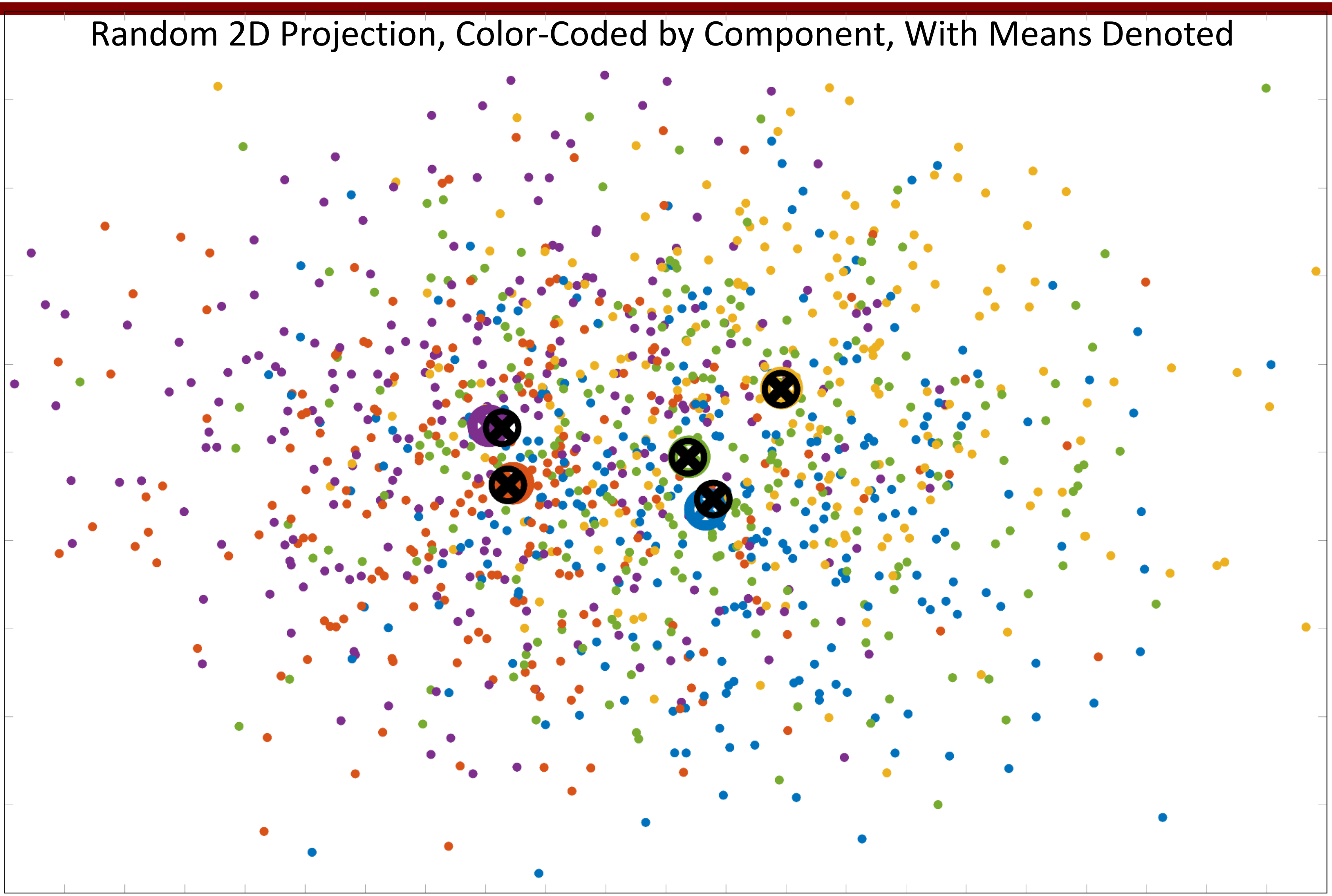
$\|\boldsymbol{\mu}_j\|_2 = 1$

$\forall j \in [r]$

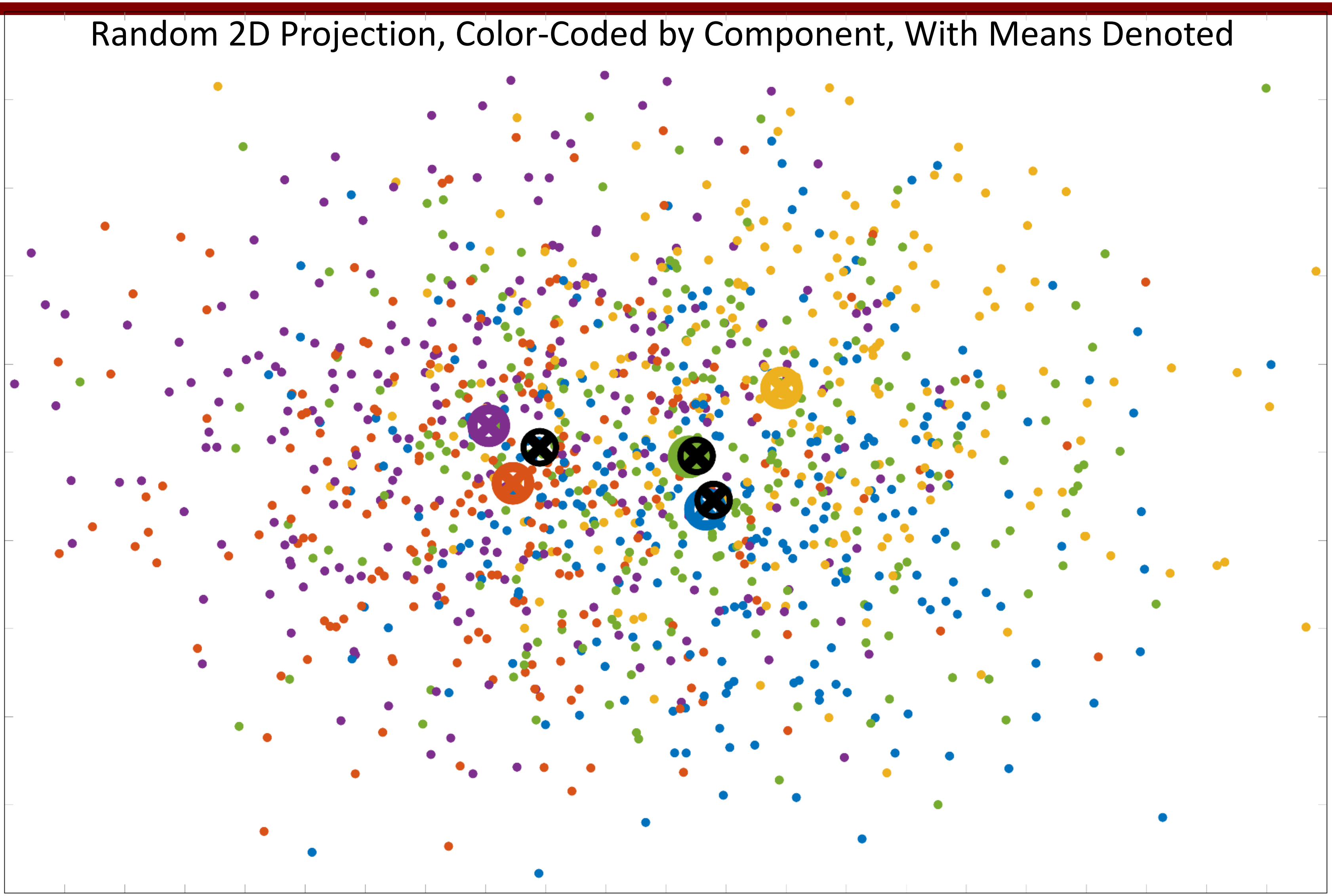
$\boldsymbol{\mu}_j^T \boldsymbol{\mu}_k = 0.5$

$\forall j \neq k$

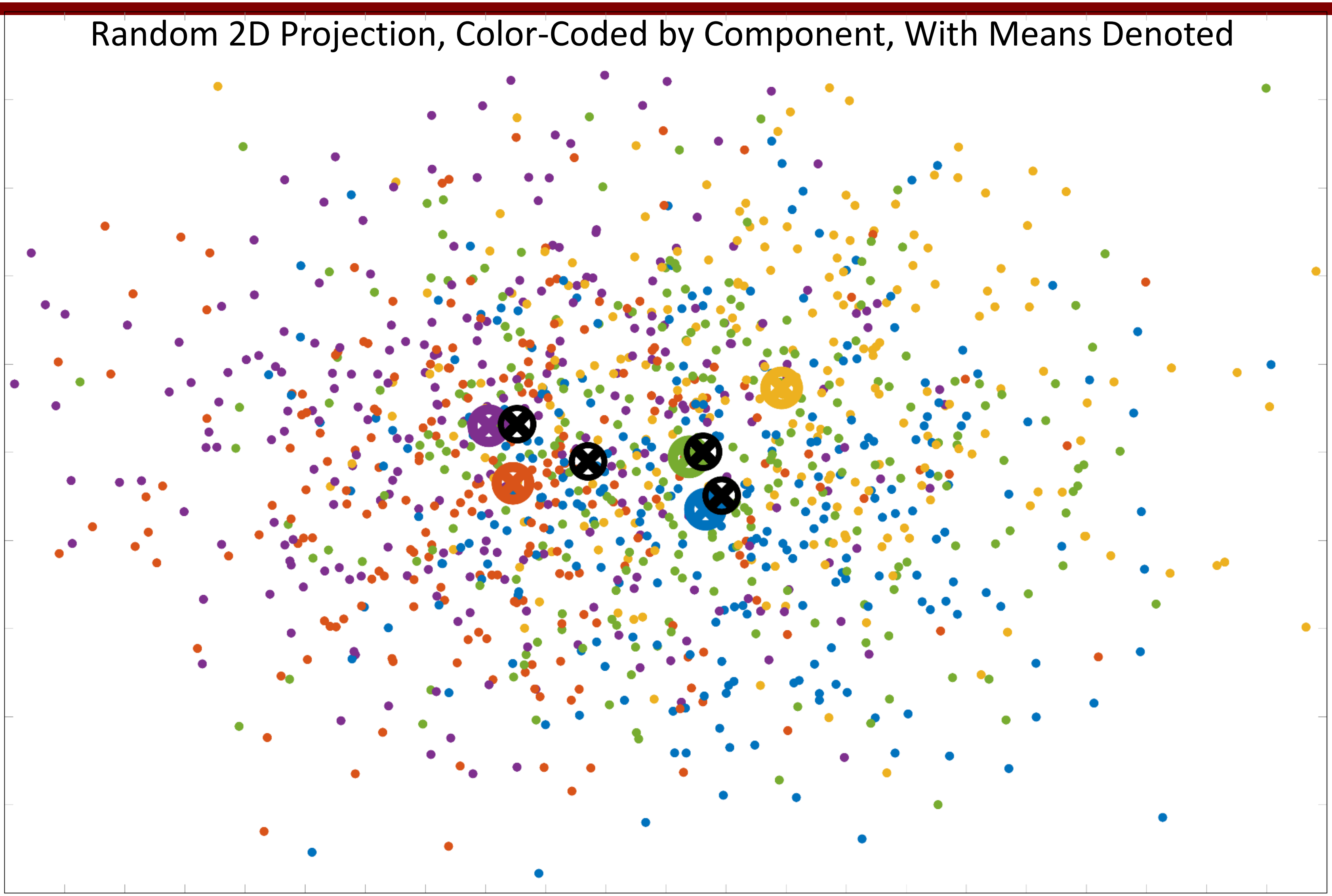
Shown here:
$\sigma = 0.1$



Identified Factors for $\hat{r}$=3 with $r = 5, p = 750, n = 500, \sigma = 0.1$

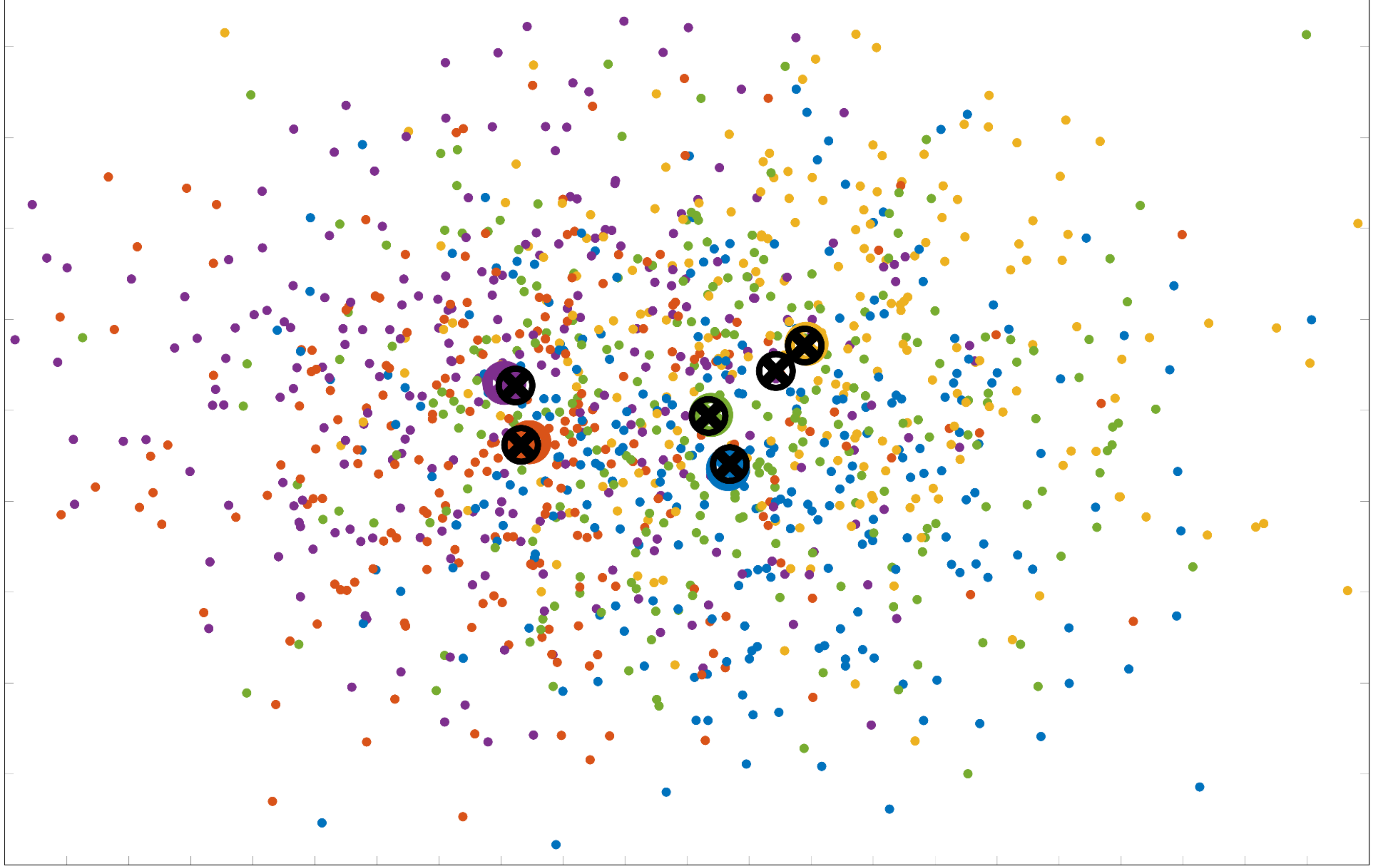


Identified Factors for $\hat{r}$=4 with $r = 5, p = 750, n = 500, \sigma = 0.1$

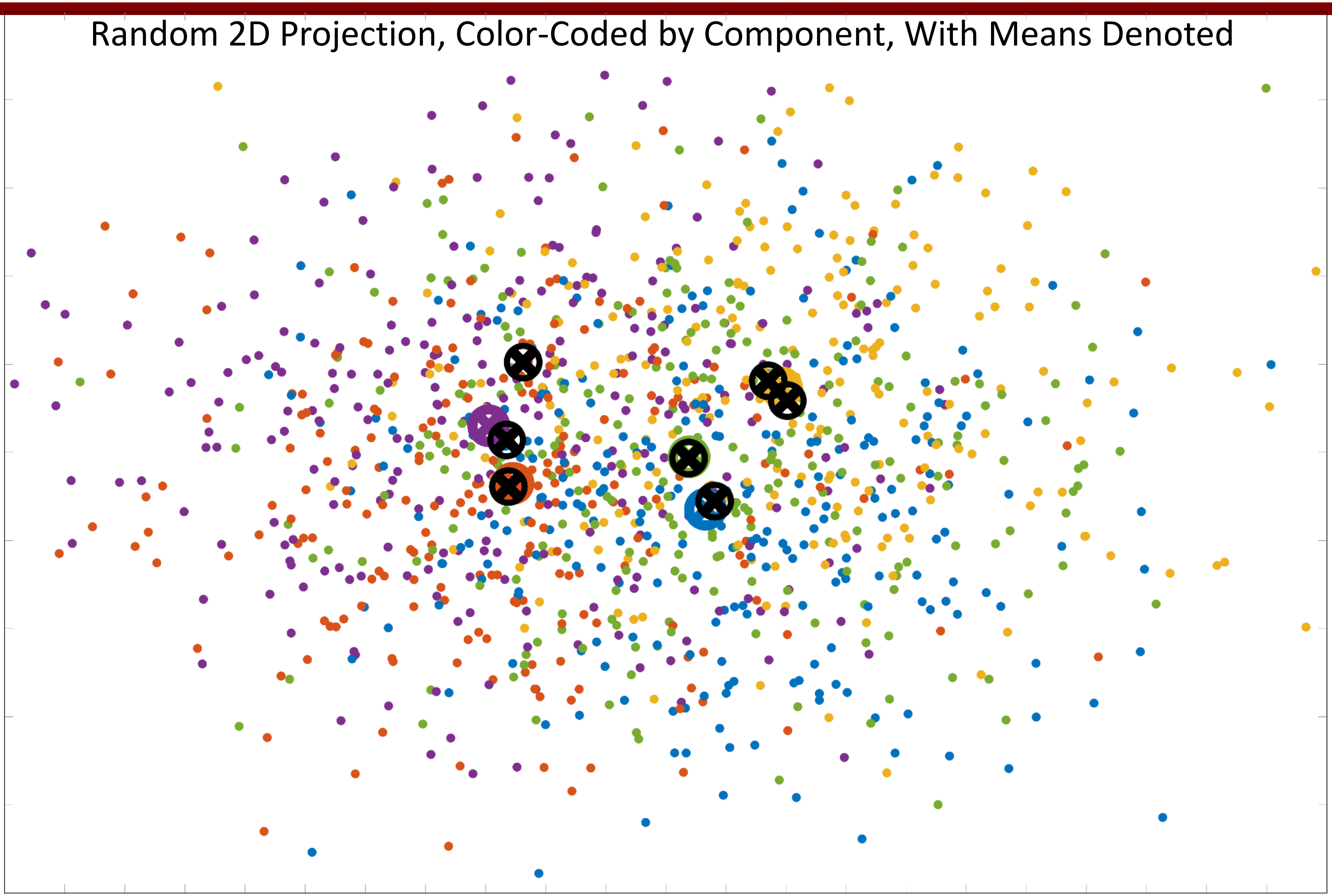


Identified Factors for $\hat{r}$=6 with $r = 5, p = 750, n = 500, \sigma = 0.1$

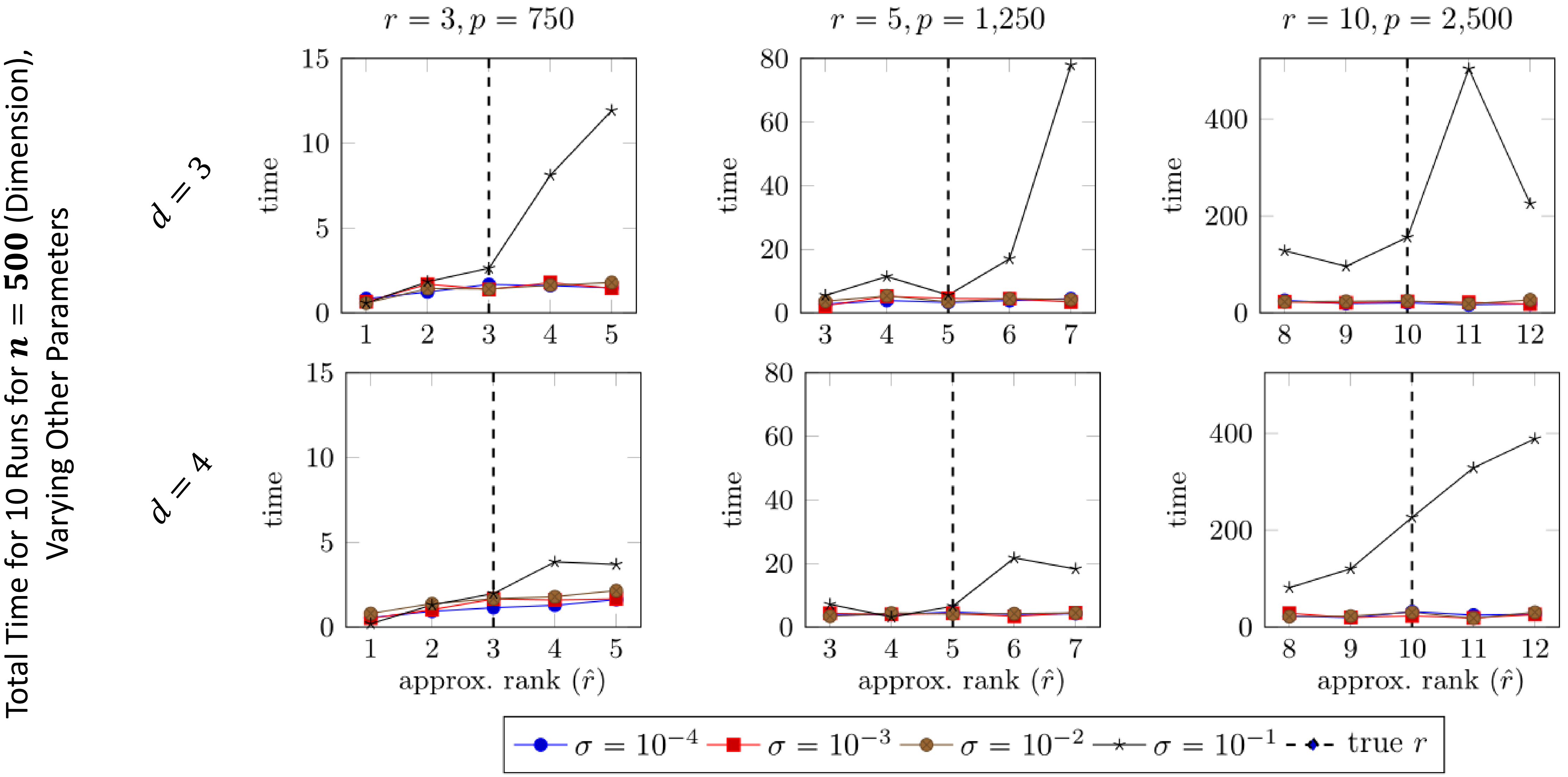
Random 2D Projection, Color-Coded by Component, With Means Denoted



Identified Factors for $\hat{r}$=7 with $r = 7, p = 750, n = 500, \sigma = 0.1$



Total Time is Very Modest and Oftentimes Less for Higher Order!



For Massive Numbers of Observations, Use Stochastic Variants

$\mathbf{V} \in \mathbb{R}^{n \times p}$

$$\mathcal{X} = \frac{1}{p} \sum_{\ell=1}^{p} \mathbf{v}_\ell^{\otimes d}$$

Sample columns with replacement

$\tilde{\mathbf{V}} \in \mathbb{R}^{n \times s}$

$$\tilde{\mathcal{X}} = \frac{1}{s} \sum_{\ell=1}^{s} \tilde{\mathbf{v}}_\ell^{\otimes d}$$

$$\Rightarrow \quad \mathbb{E}[\tilde{\mathcal{X}}\mathbf{a}^{d-1}] = \mathcal{X}\mathbf{a}^{d-1}$$

Example Results

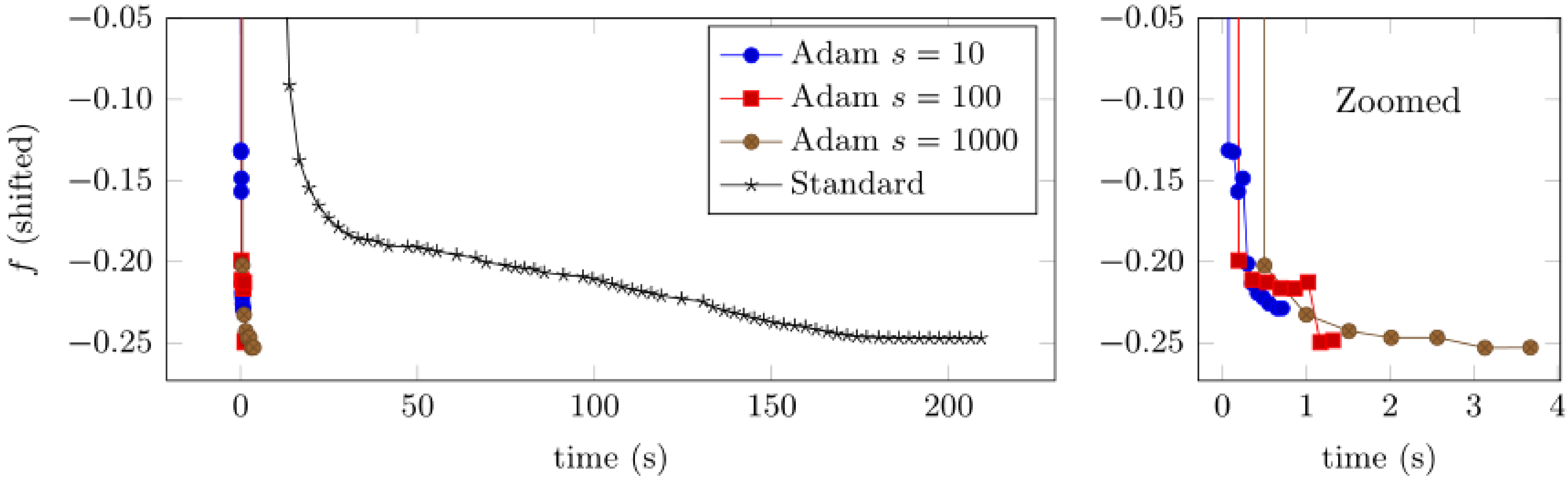
$\hat{r} = r = 10, n = 500,$
$\sigma = 0.1, d = 3$
$p = 100{,}000$

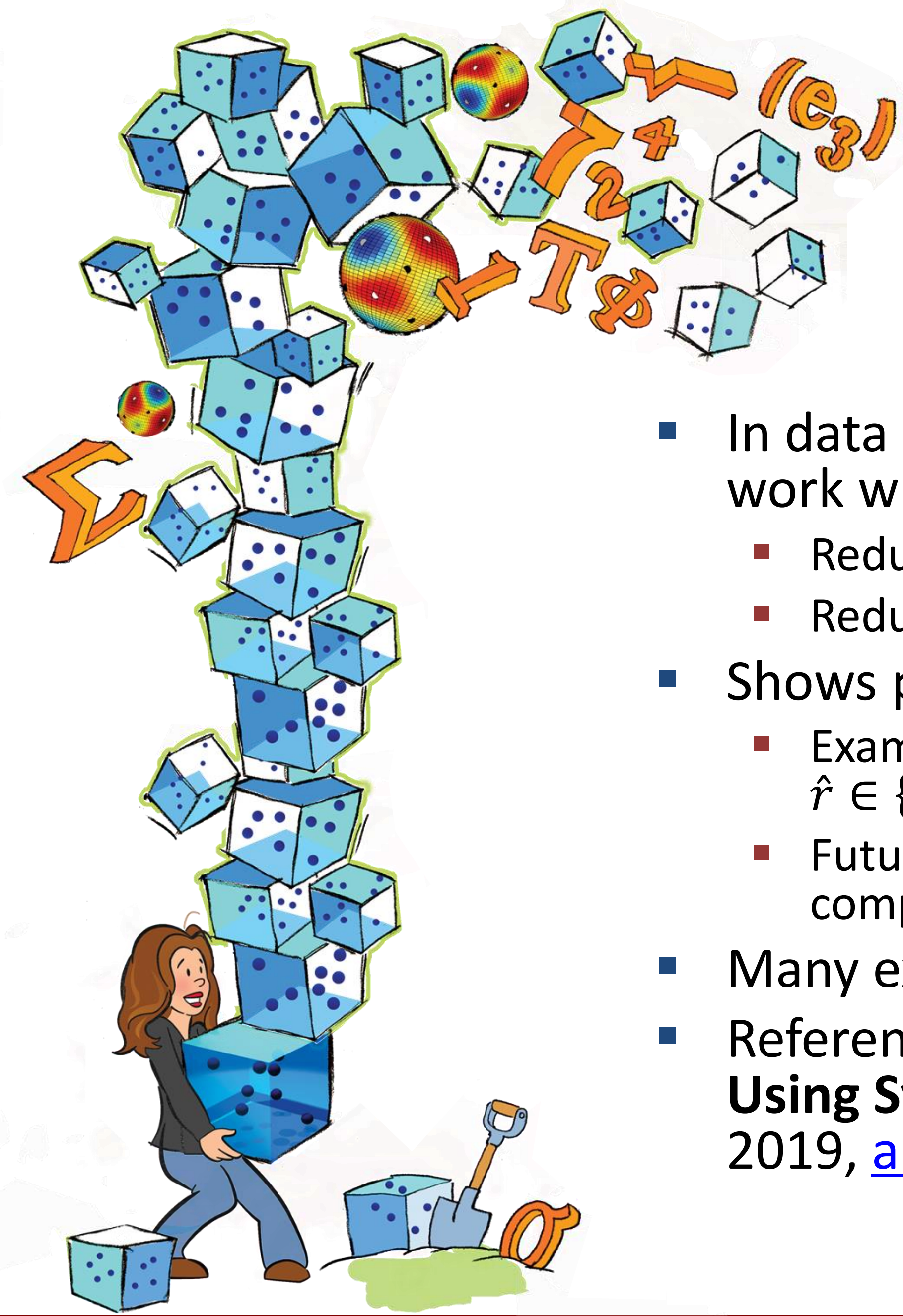
Method	Best f (shifted)	Sim. Score	Total Time (s)
standard	-0.2471	0.9998	2166.70
Adam, s=10	-0.2209	0.9225	8.03
Adam, s=100	-0.2427	0.9929	10.48
Adam, s=1000	-0.2464	0.9990	41.00

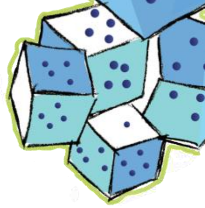
Speed Advantage for Stochastic Methods

Best Runs (of 10)

$\hat{r} = r = 10, n = 500, \sigma = 0.1, d = 3, p = 100{,}000$







Conclusions and Future Work

- In data analysis, dth-order moment is expensive to compute – instead work with implicit moment
  - Reduces storage from $O(n^d)$ to $O(np)$
  - Reduces computation per iteration from $O(rn^d)$ to $O(rnp)$
- Shows promise for fitting spherical GMMs
  - Example with $n = 500$ (dimension), $r \in \{3,5,10\}$ (components), $p = 250r$, $\hat{r} \in \{r-2, \dots, r+2\}$, and $d = 3,4$ (orders)
  - Future work will incorporate lower-order terms, different σ for each component, multiple values for d simultaneously, etc.
- Many extensions possible, e.g., for subspace power method
- Reference: S. Sherman, T. G. Kolda. **Estimating Higher-Order Moments Using Symmetric Tensor Decomposition**, submitted for publication, 2019, arXiv:1911.03813