

Tensor Decomposition: A Mathematical Tool for Data Analysis

NC State
RED Talks
*Where Data
Meets Science*

October 29, 2018

Tamara G. Kolda

Sandia National Labs, Livermore, CA
www.kolda.net

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-NA0003525.

Acknowledgement of Collaborators

- **D. Hong**, T. G. Kolda, J. A. Duersch. **Generalized Canonical Polyadic Tensor Decomposition**. arXiv, 2018. <http://arxiv.org/abs/1808.07452>
- **A. H. Williams**, T. H. Kim, F. Wang, S. Vyas, S. I. Ryu, K. V. Shenoy, M. Schnitzer, T. G. Kolda, S. Ganguli. **Unsupervised Discovery of Demixed, Low-dimensional Neural Dynamics across Multiple Timescales through Tensor Components Analysis**. *Neuron*, 98(6):1099-1115, 2018. <https://doi.org/10.1016/j.neuron.2018.05.015>
- Older relevant work
 - E. C. Chi, T. G. Kolda. **On Tensors, Sparsity, and Nonnegative Factorizations**. *SIAM Journal on Matrix Analysis and Applications*, 33(4):1272-1299, 2013. <https://doi.org/10.1137/110859063>
 - E. Acar, D. M. Dunlavy, T. G. Kolda. **A Scalable Optimization Approach for Fitting Canonical Tensor Decompositions**. *Journal of Chemometrics* 25(2):67-86, 2011. <https://doi.org/10.1002/cem.1335>
 - T. G. Kolda, B. W. Bader. **Tensor Decompositions and Applications**. *SIAM Review* 51(3):455-500, 2009. <https://doi.org/10.1137/07070111X>
- Plus thanks to other collaborators including Cliff Anderson-Bergman, Casey Battaglino, Grey Ballard, Karen Devine, Brett Larsen, Eric Phipps, Sam Sherman, Kina Winoto, and many more

Recent Student Interns



David Hong
(Michigan)



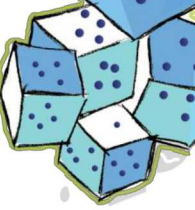
Brett Larsen
(Stanford)



Sam Sherman
(Notre Dame)



Alex Williams
(Stanford)



Tensor = Multi-way Array

Vector
Size: 3

1
2
3

$\mathbf{x}$

$x(2)$

Matrix
Size: 3×4

1	4	7	10
2	5	8	11
3	6	9	12

$\mathbf{X}$

$x(2,3)$

3-way Tensor
Size: $3 \times 4 \times 3$

25	28	31	34
13	16	19	22
1	4	7	10
⋮	⋮	⋮	⋮
26	29	32	35
14	17	20	23
2	5	8	11
⋮	⋮	⋮	⋮
27	30	33	36
15	18	21	24
3	6	9	12

$\mathbf{x}$

$x(1,3,2)$

4-way Tensor
Size: $3 \times 4 \times 3 \times 2$

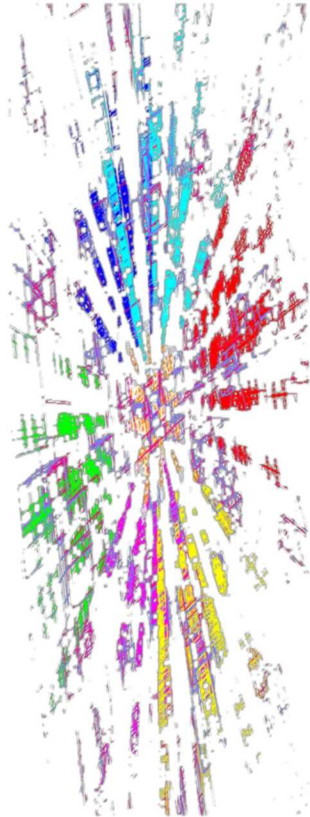
24	27	30	33
12	15	18	21
0	3	6	9
⋮	⋮	⋮	⋮
25	28	31	34
13	16	19	22
1	4	7	10
⋮	⋮	⋮	⋮
26	29	32	35
14	17	20	23
2	5	8	11

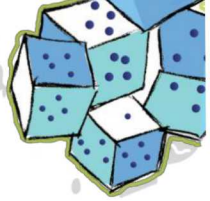
$\mathbf{x}(:, :, :, 1)$

60	63	66	69
48	51	54	57
36	39	42	45
⋮	⋮	⋮	⋮
61	64	67	70
49	52	55	58
37	40	43	46
⋮	⋮	⋮	⋮
62	65	68	71
50	53	56	59
38	41	44	47

$\mathbf{x}(:, :, :, 2)$

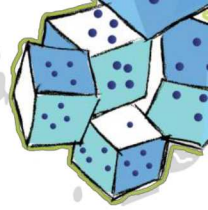
$x(2,3,1,2)$





Where do Data Tensors Come From?

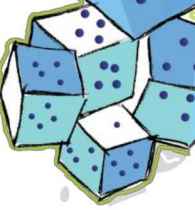




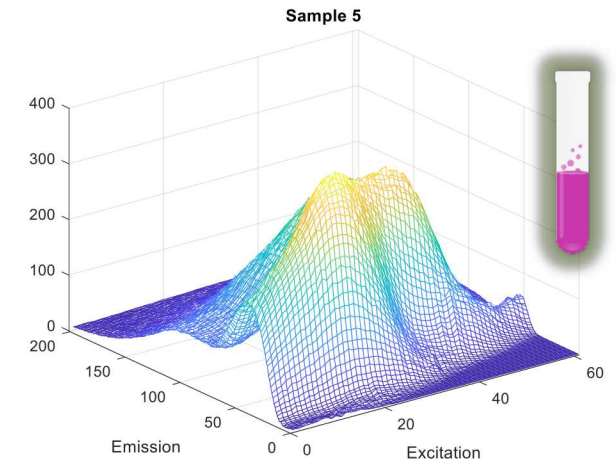
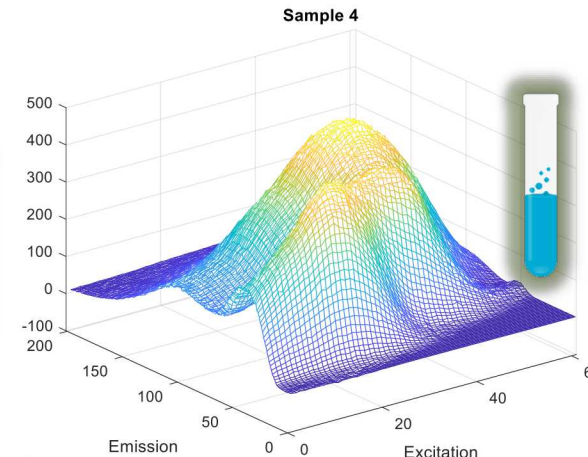
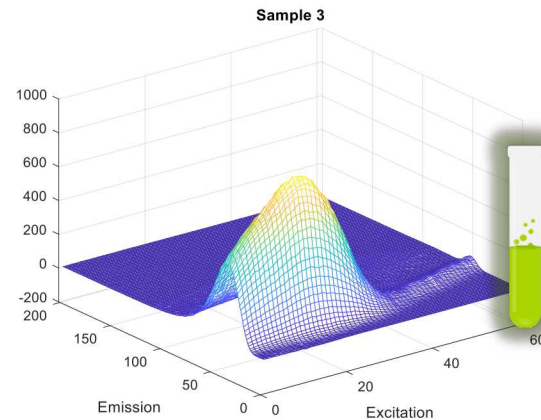
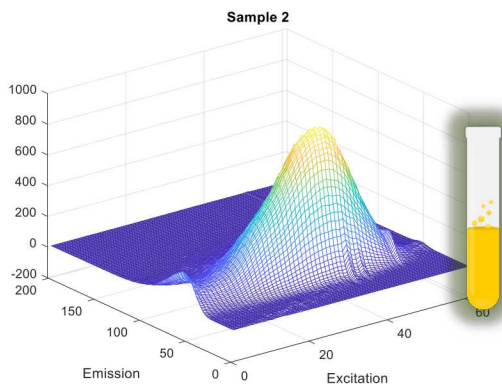
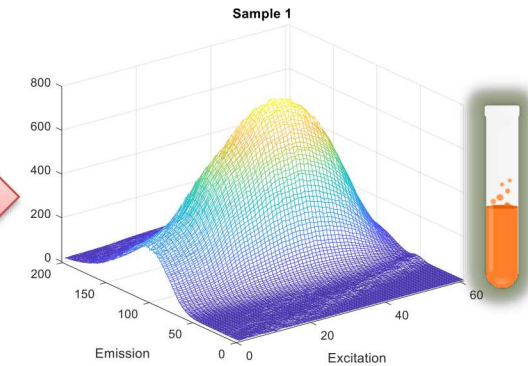
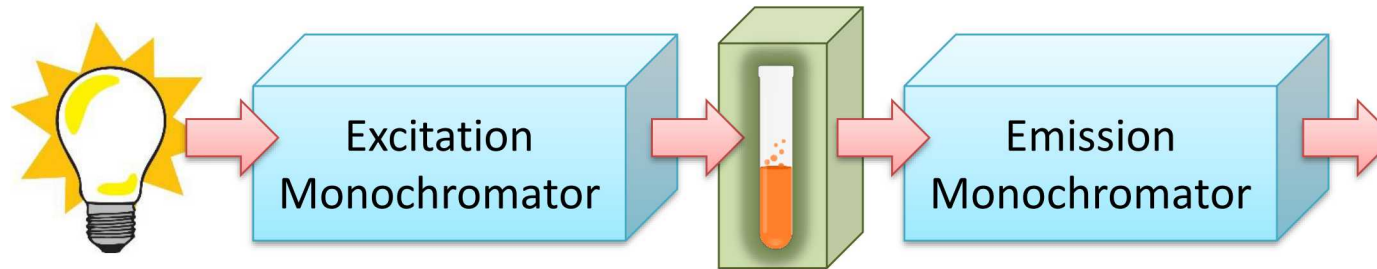
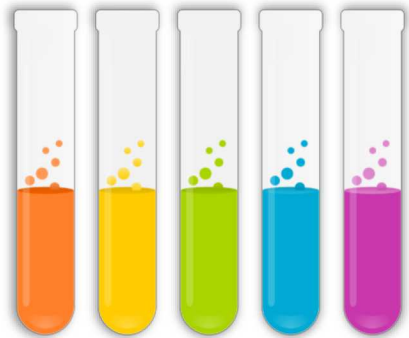
Example Tensor from Chemistry

Source: Bro, R, Multi-way Analysis in the Food Industry. Models, Algorithms, and Applications. 1998. Ph.D. Thesis, University of Amsterdam. Download from “amino acids” at <http://www.models.life.ku.dk/nwaydata1>.

Emission Fluorescence Spectroscopy for Single Sample Creates a Matrix of Measurements

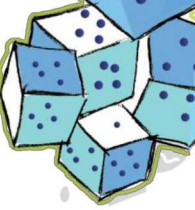


Goal: Figure out what's in a given set of chemical samples

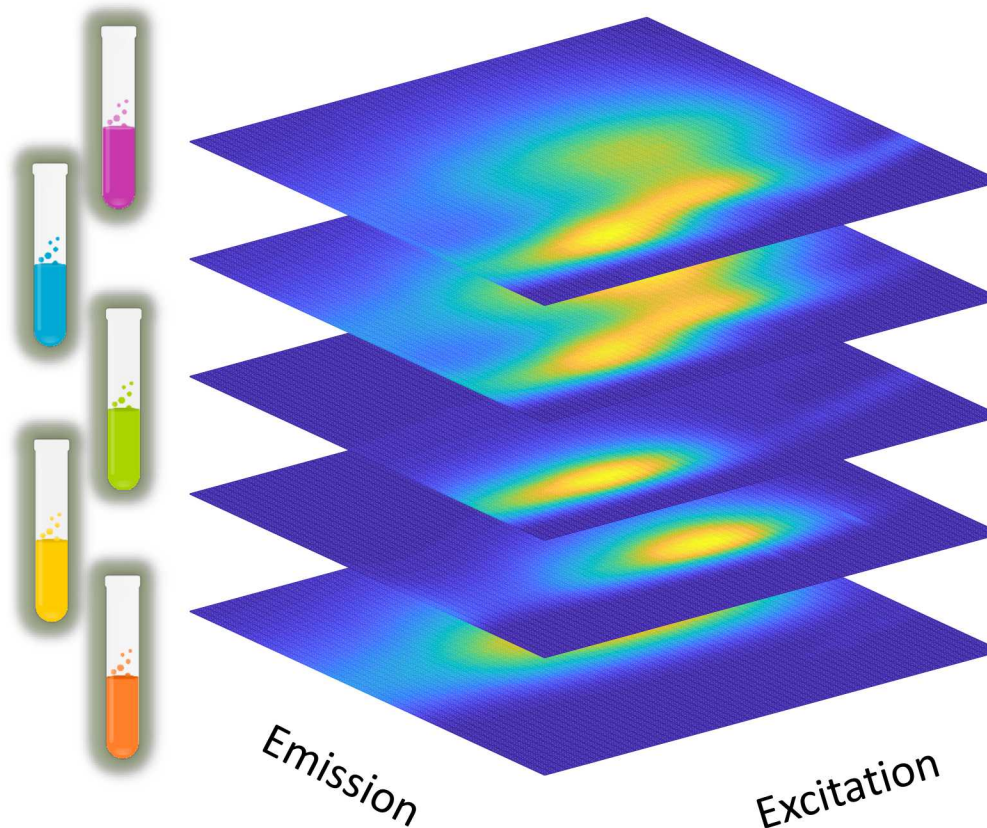
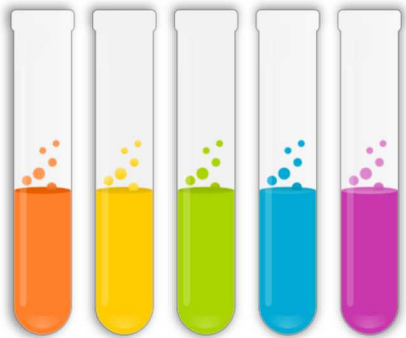


Bro, PhD Thesis, 1998.

Emission Fluorescence Spectroscopy for Multiple Samples Creates a 3-way Tensor

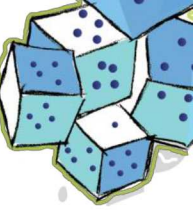


Goal: Figure out what's in a given set of chemical samples



- 3-way Tensor
- Size: 5 x 201 x 61
 - 5 samples
 - 201 emissions
 - 61 excitations

Bro, PhD Thesis, 1998.

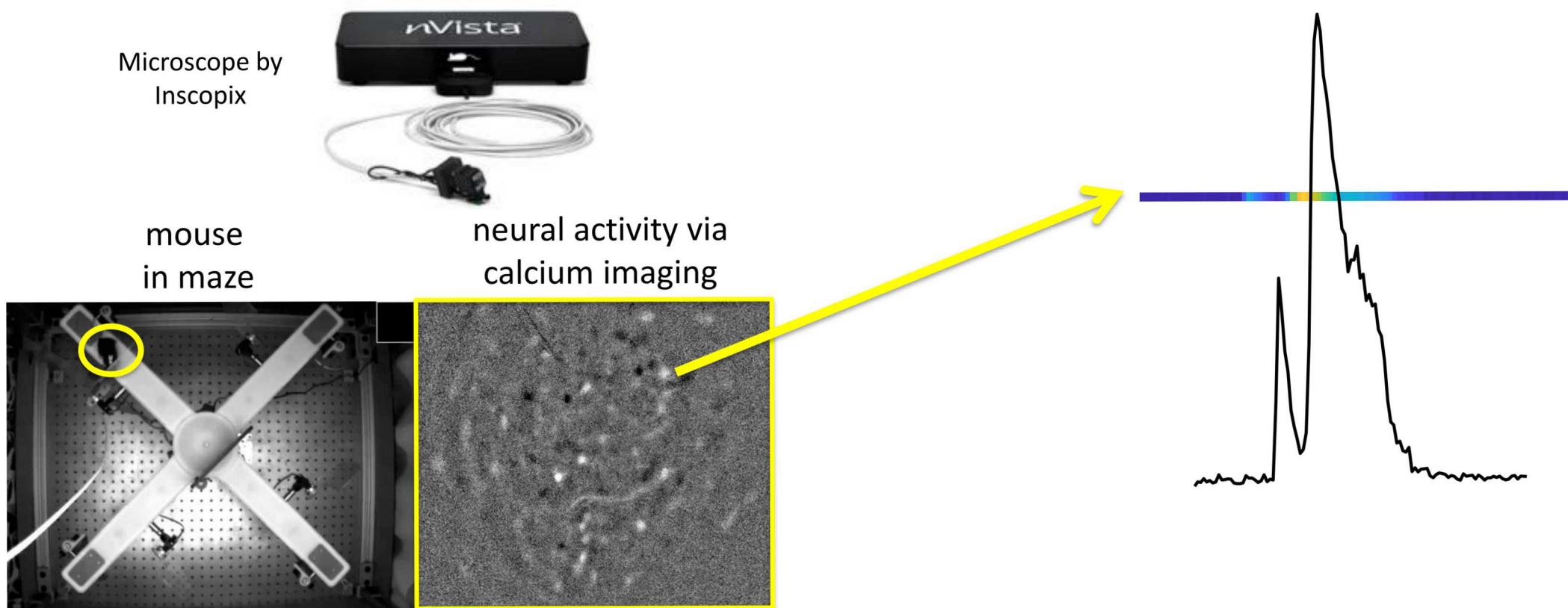


Example Tensor from Neuroscience

Source: Williams, et al. Unsupervised Discovery of Demixed, Low-dimensional Neural Dynamics across Multiple Timescales through Tensor Components Analysis. Neuron, 2018. <https://doi.org/10.1016/j.neuron.2018.05.015>

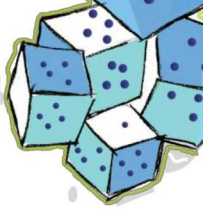
Activity of Single Neuron Measured Over Time Produces Vector Data

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



Williams et al., Neuron, 2018

Multiple Neurons Measured Over Time Produces Matrix

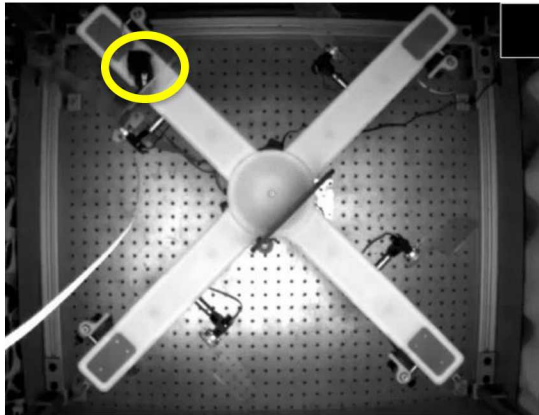


Thanks to Schnitzer Group @ Stanford
Mark Schnitzer, Fori Wang, Tony Kim

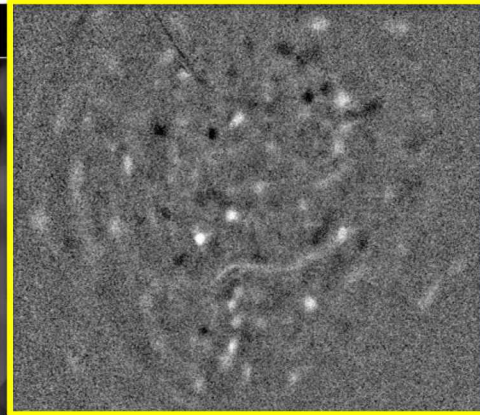
Microscope by
Inscopix



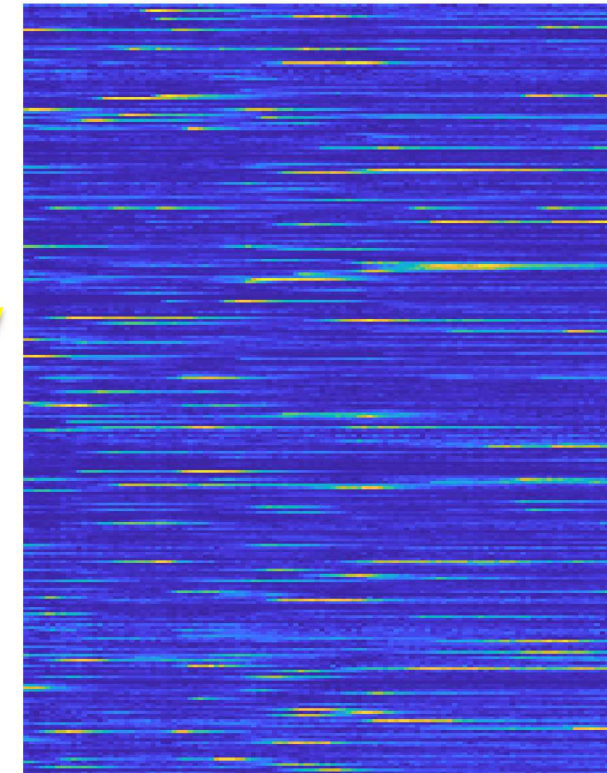
mouse
in "maze"



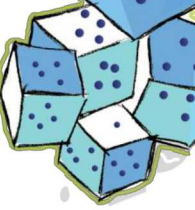
neural activity



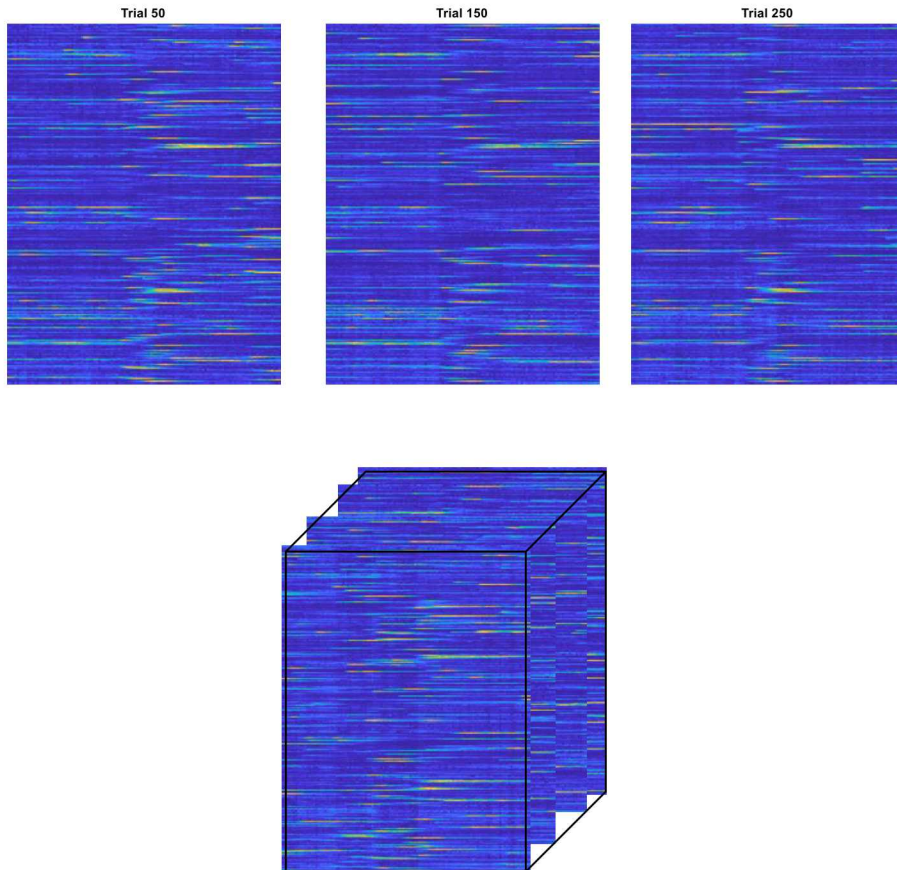
282 neurons $\times$ 111 time bins



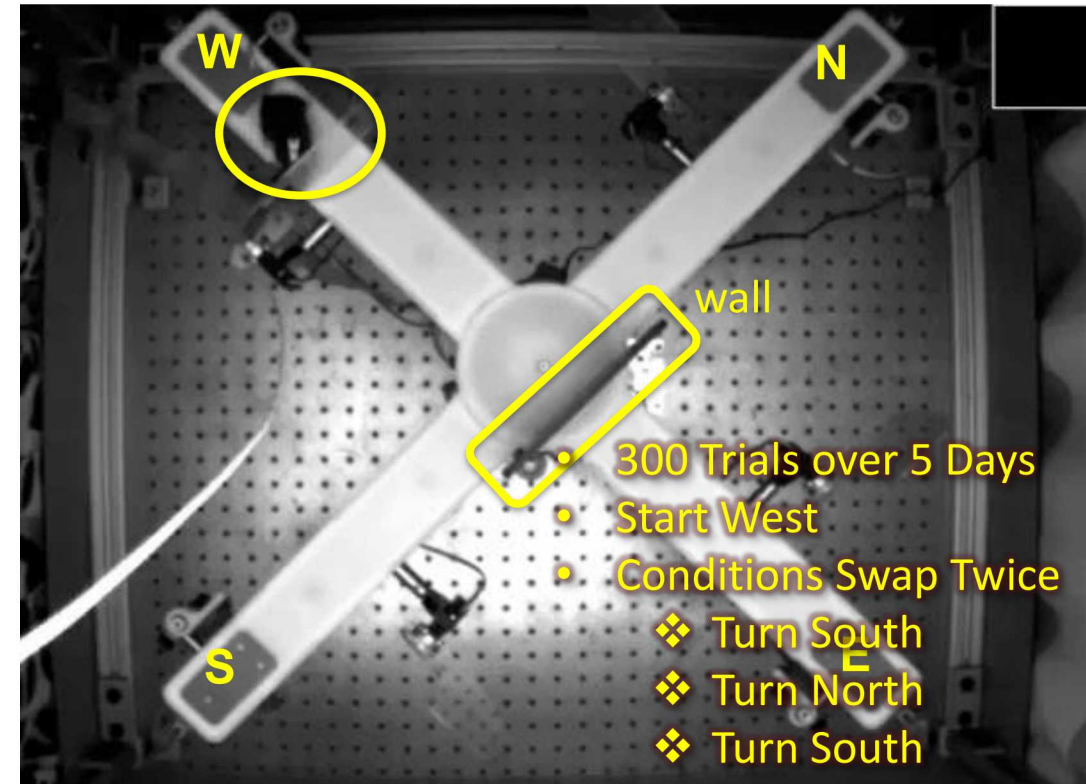
Williams et al., Neuron, 2018



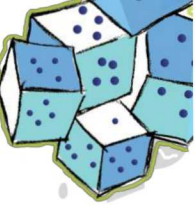
Multiple Trials Produces 3-way Tensor



282 neurons $\times$ 111 time bins $\times$ 300 trials



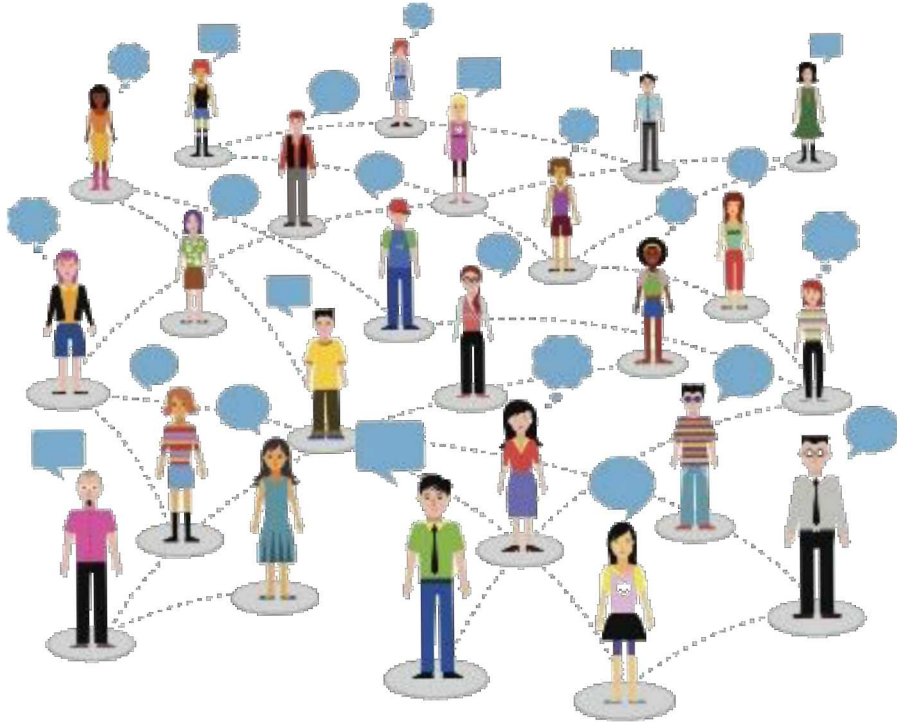
Williams et al., Neuron, 2018



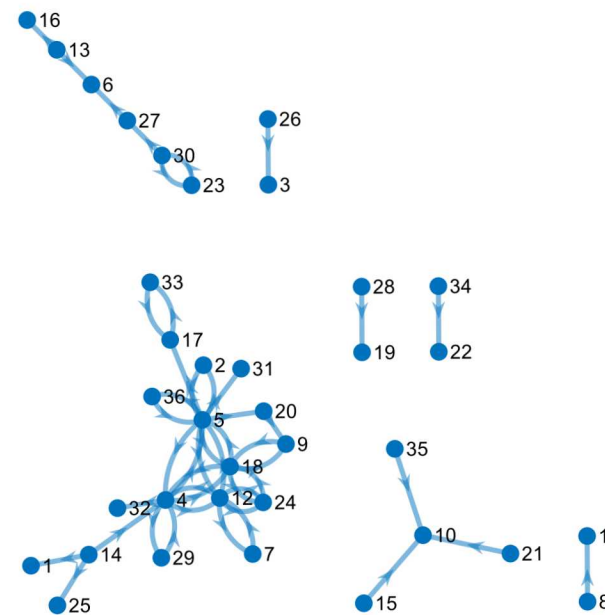
Example Tensor from Social Science

Source: Opsahl and Panzarasa, Clustering in weighted networks. *Social Networks*, 2009,
<http://doi.org/10.1016/j.socnet.2009.02.002> and
https://toreopsahl.com/datasets/#online_social_network

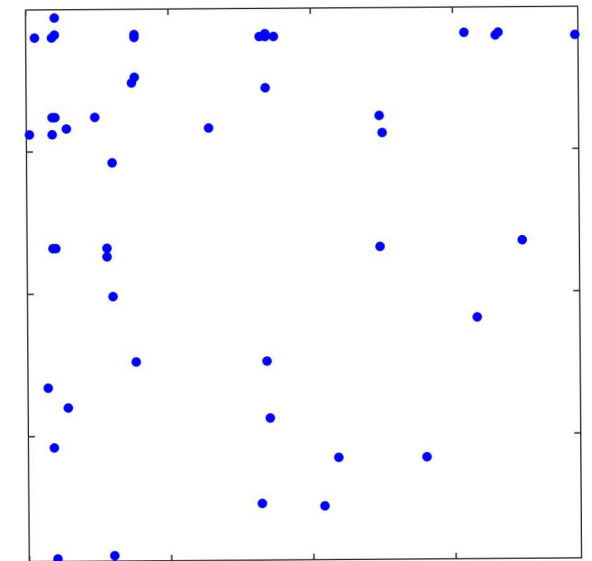
Chat Network is a Matrix



One Day of Interactions

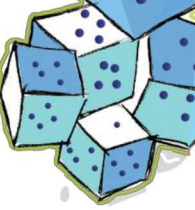


Sparse Matrix (200×200)

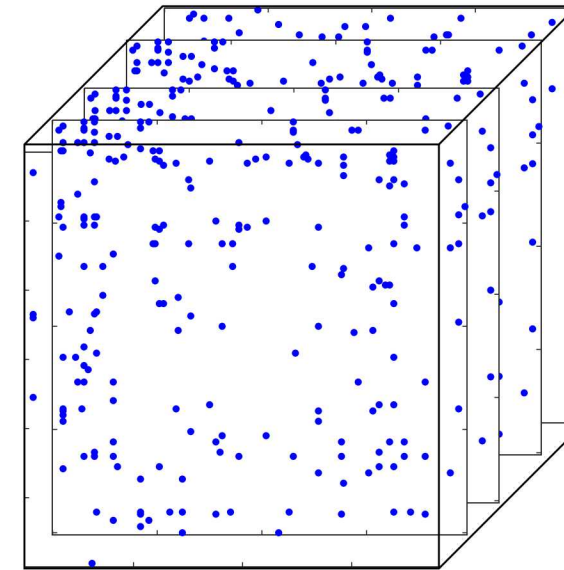
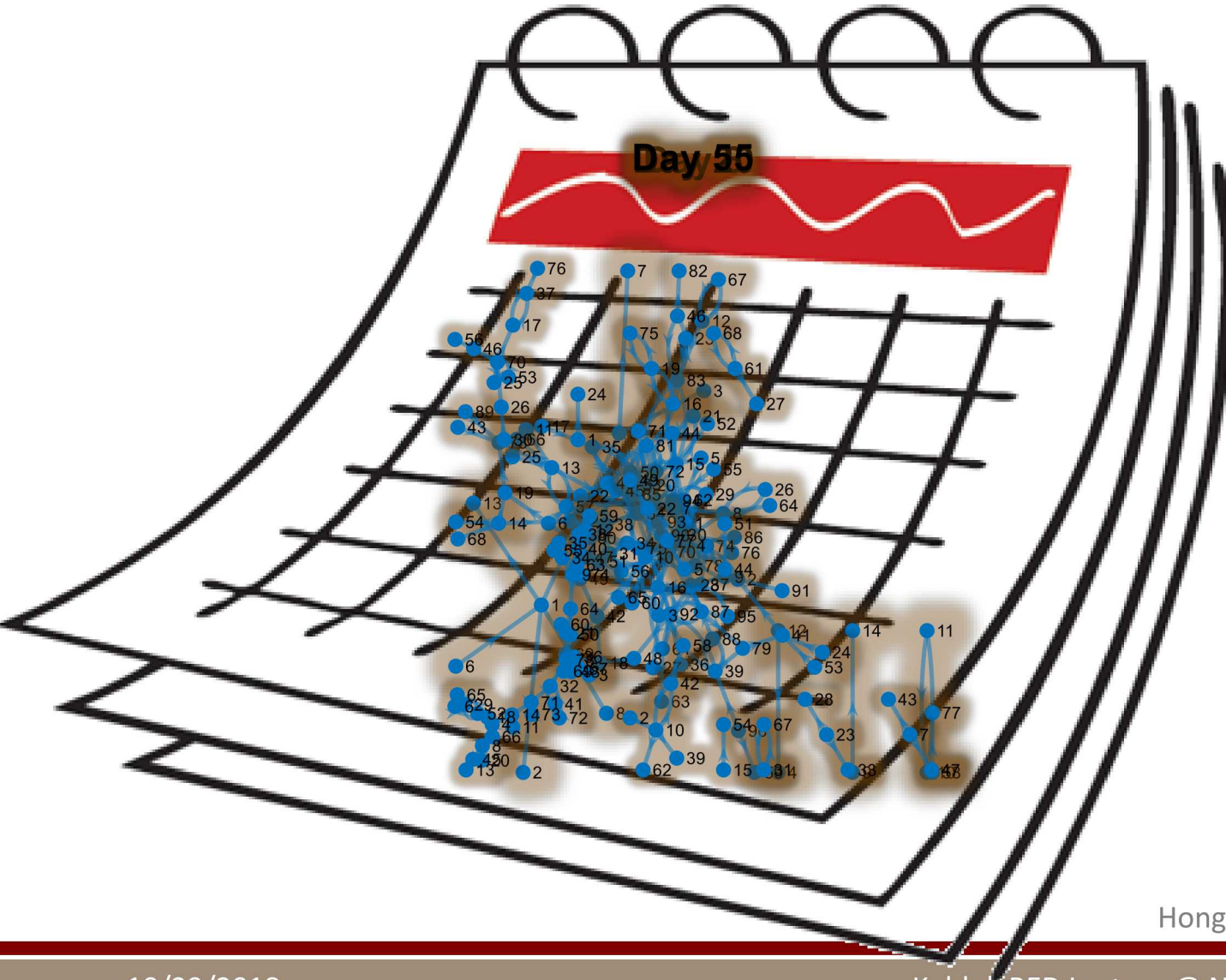


$$x(i, j) = \begin{cases} 1 & \text{if } i \text{ sent message to } j \\ 0 & \text{otherwise} \end{cases}$$

Opsahl and Panzarasa, Social Networks, 2009



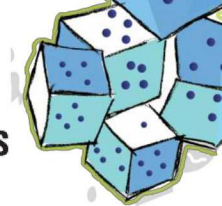
Time-Evolving Chat Network is a Tensor



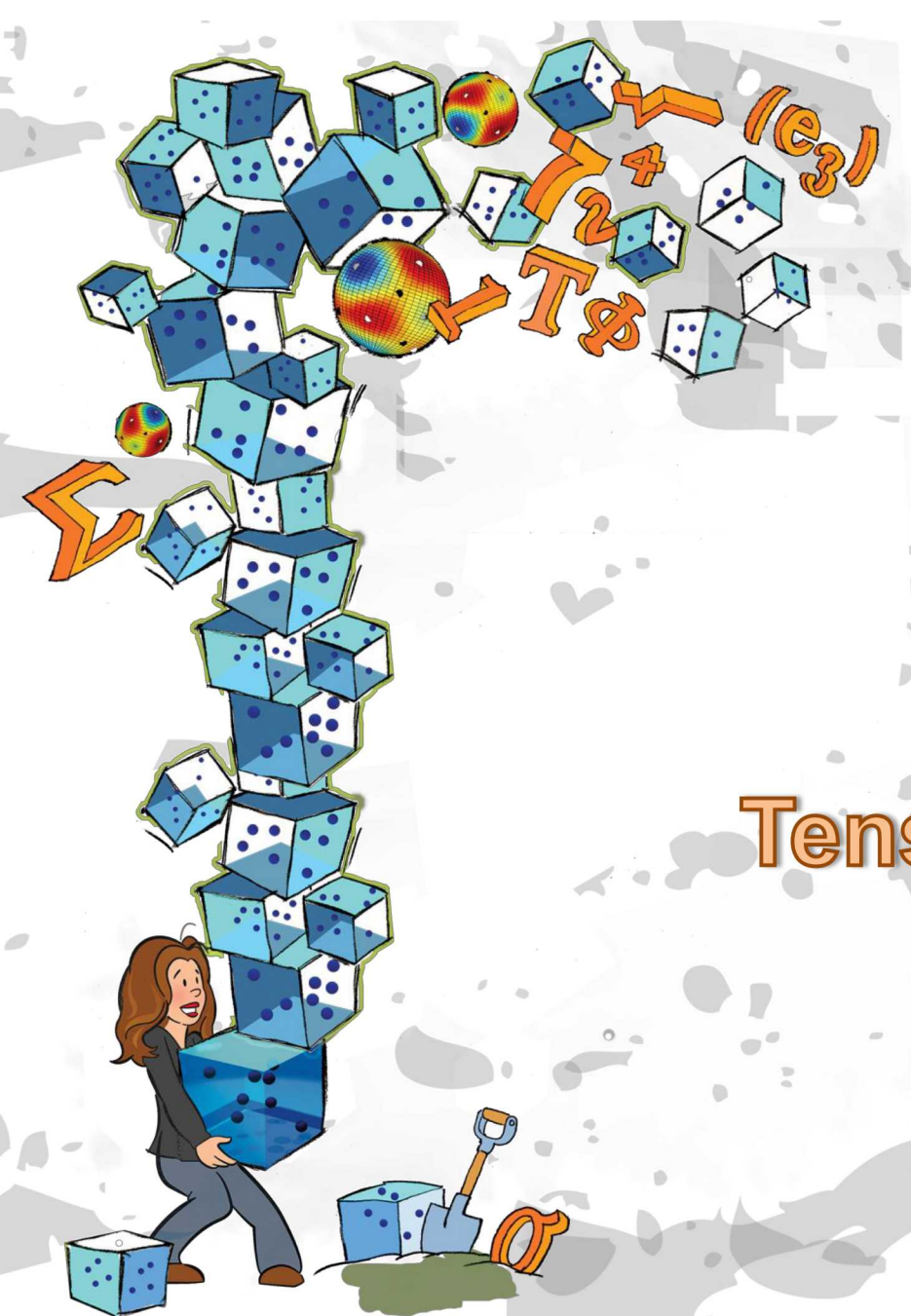
- 3-way binary tensor
 - Sender (200)
 - Receiver (200)
 - Day (195)
- 9,764 nonzeros (very sparse, 0.1%)

$$x_{ijk} = \begin{cases} 1 & \text{if } i \text{ messages } j \text{ on day } k \\ 0 & \text{otherwise} \end{cases}$$

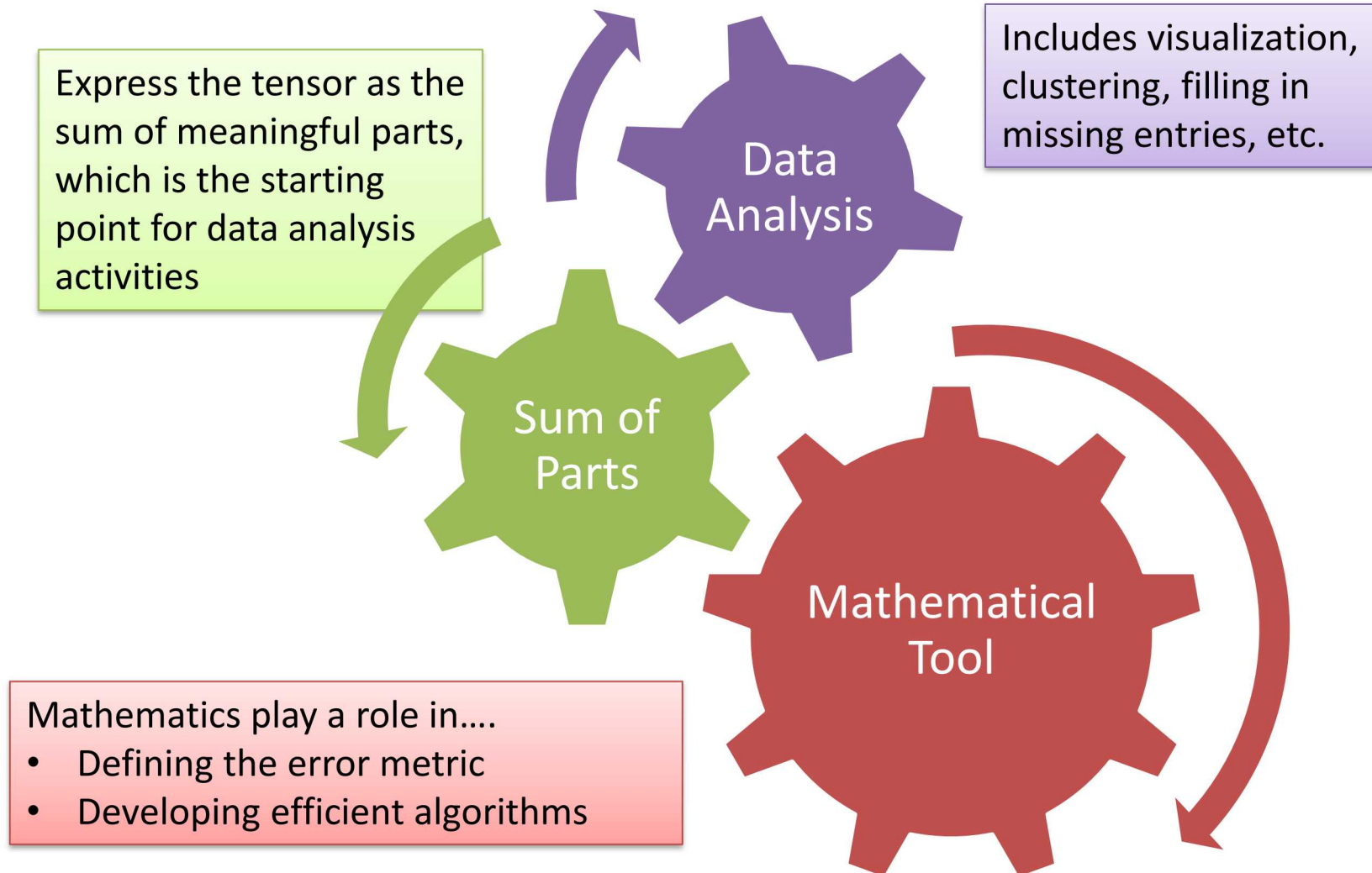
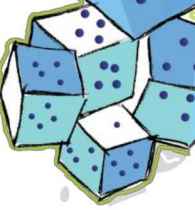
Hong, Kolda, Duersch, arXiv, 2018



Tensor Decomposition



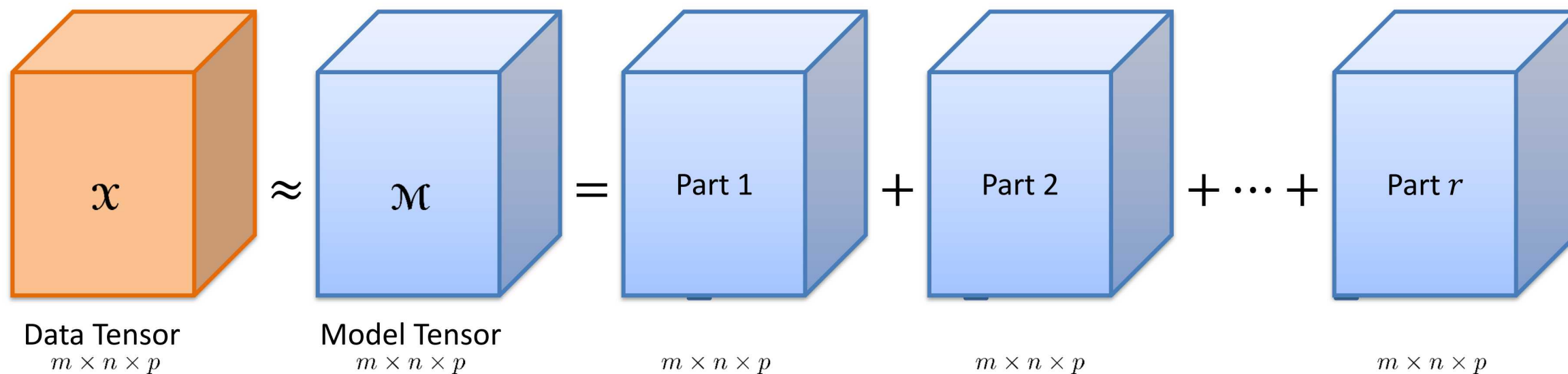
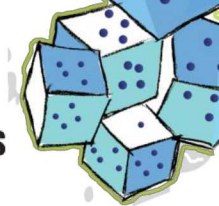
Tensor Decomposition: A Mathematical Tool for Analysis of Tensor Data



Related Concepts for Matrices

- Singular value decomposition (SVD)
- Principal component analysis (PCA) – *Tipping & Bishop, 1999*
- Independent component analysis (ICA)
- Nonnegative matrix factorization (NMF)
- Sparse matrix factorization
- Matrix completion

Break Tensor into Understandable Parts...



Key: The parts have structure!

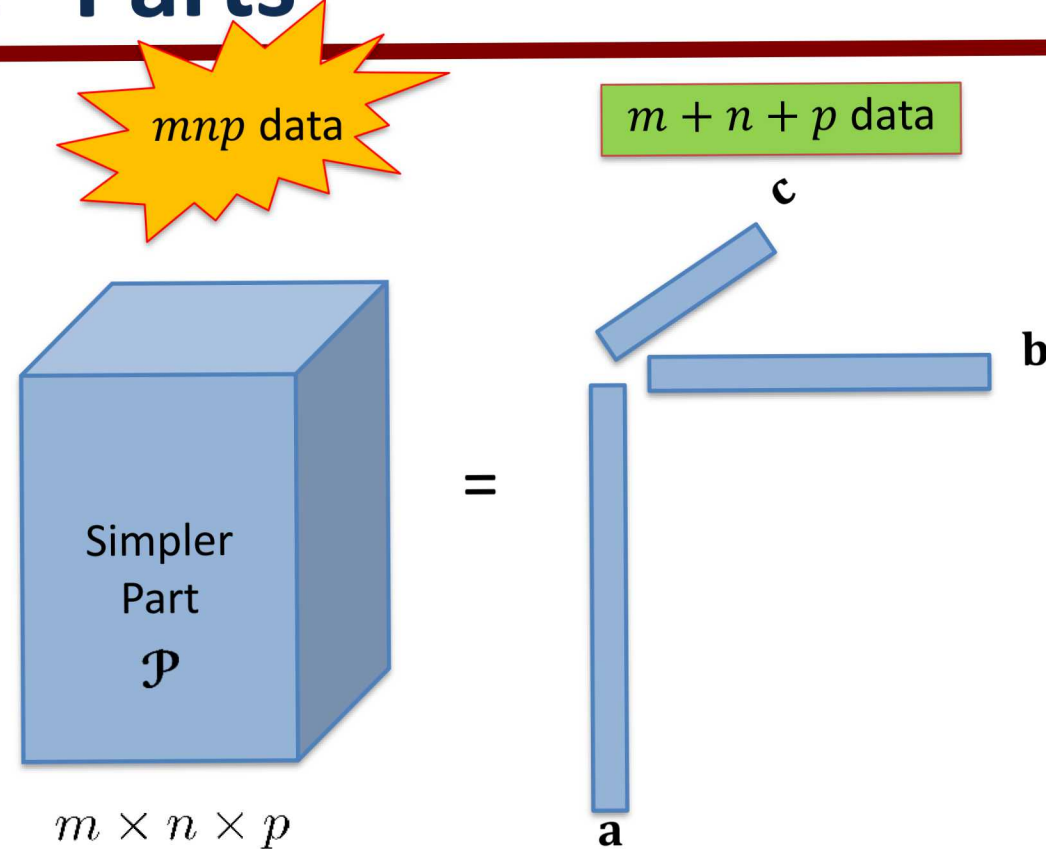
Rank-1 Tensors are the “Parts”

Given **three vectors**:

$$\mathbf{a} \in \mathbb{R}^m, \mathbf{b} \in \mathbb{R}^n, \mathbf{c} \in \mathbb{R}^p$$

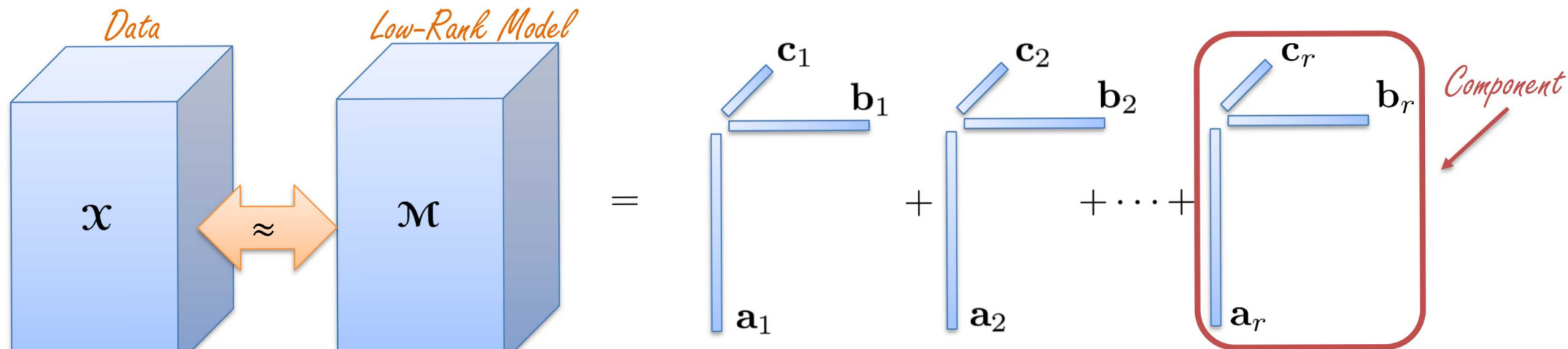
The **outer product** is

$$\mathcal{P} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c} \in \mathbb{R}^{m \times n \times p}$$



$$p_{ijk} = a_i b_j c_k$$

CANDECOMP/PARAFAC (CP) Tensor Factorization Uncovers the Rank-1 Parts

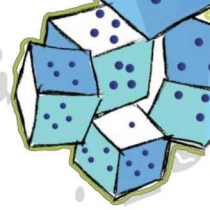


$$\mathcal{X} \approx \mathcal{M} = \sum_{\alpha=1}^r \mathbf{a}_{\alpha} \circ \mathbf{b}_{\alpha} \circ \mathbf{c}_{\alpha} = [\mathbf{A}, \mathbf{B}, \mathbf{C}]$$

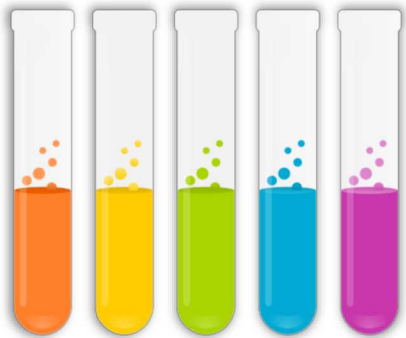
*Factor ↑
Matrices*

Hitchcock, 1927; Carroll and Chang, 1970; Harshman, 1970

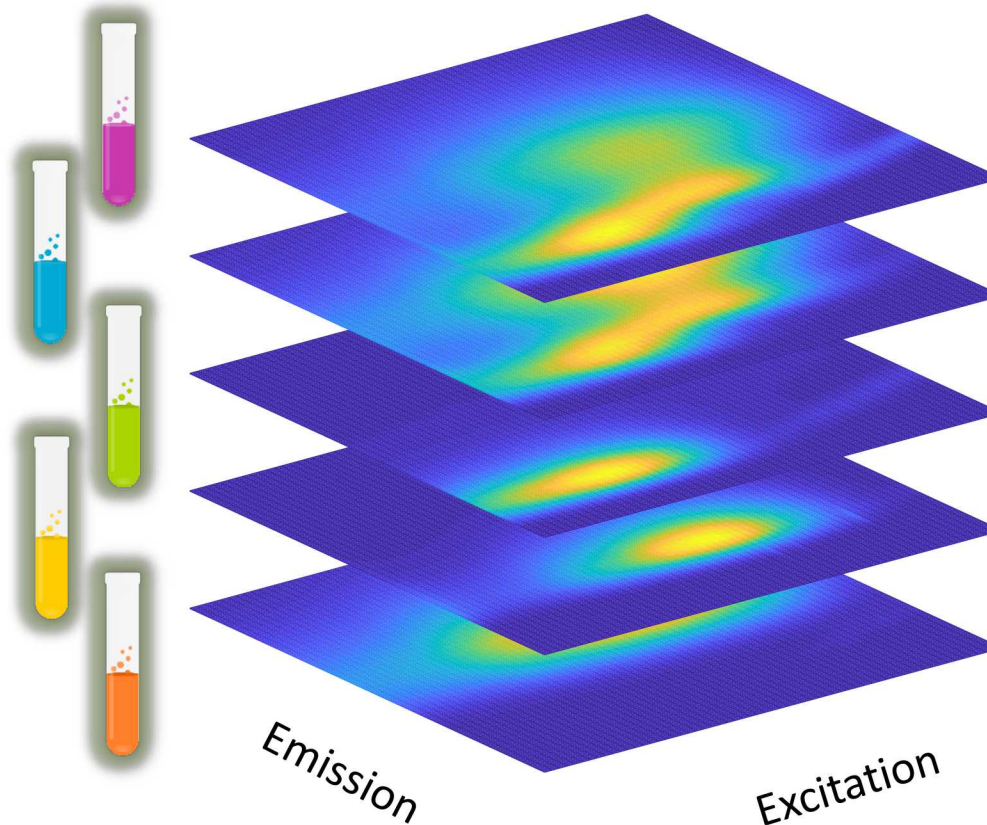
In Chemistry, Tensor Decomposition Discovers Make-up of Samples



Goal: Figure out what's in a given set of chemical samples

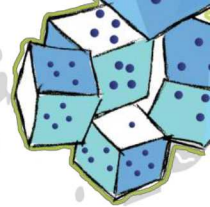


- Fluorescence measurements of samples containing 3 amino acids
 - Tryptophan
 - Tyrosine
 - Phenylalanine

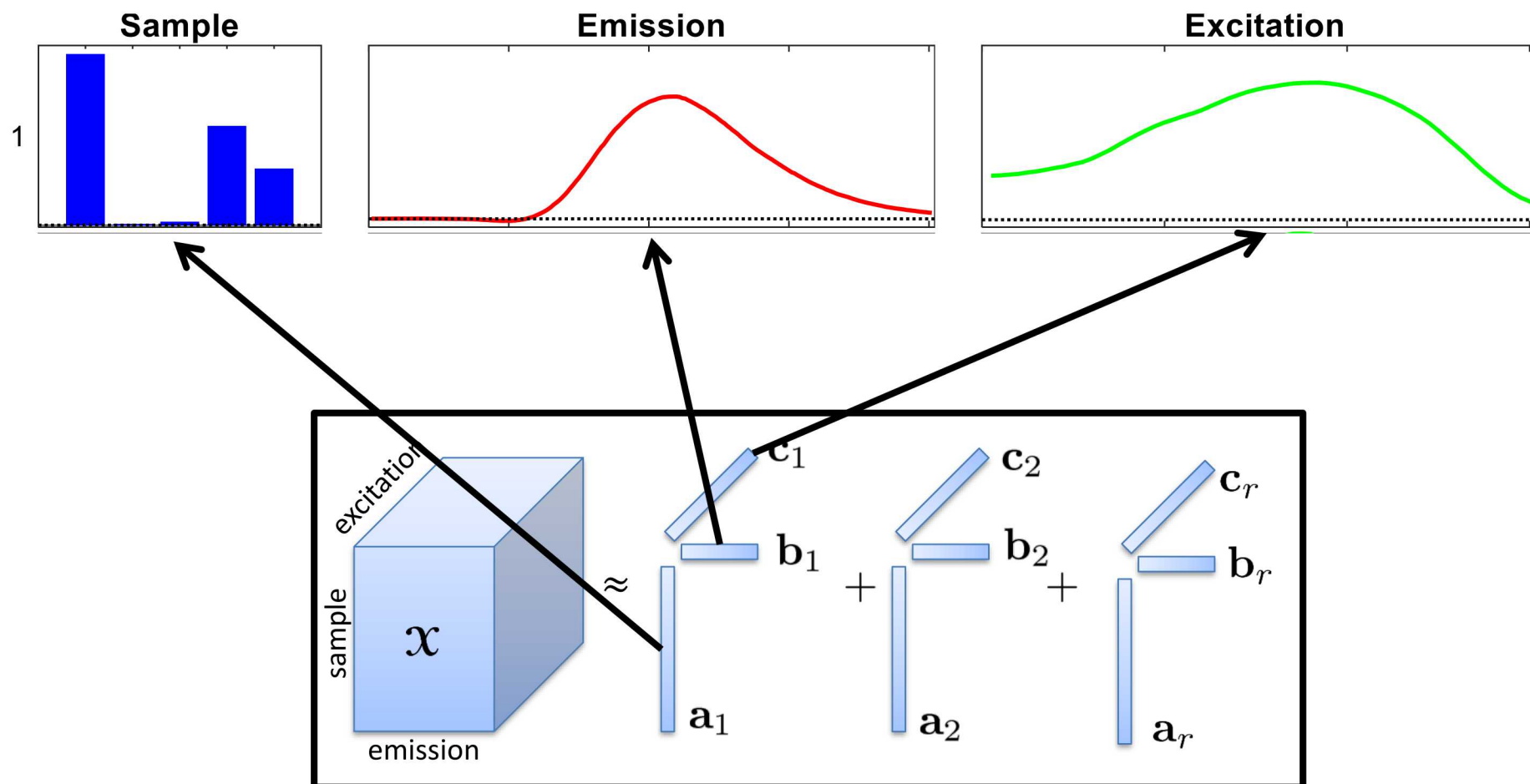


- 3-way Tensor
- Size: 5 x 201 x 61
 - 5 samples
 - 201 emissions
 - 61 excitations

Bro, PhD Thesis, 1998.

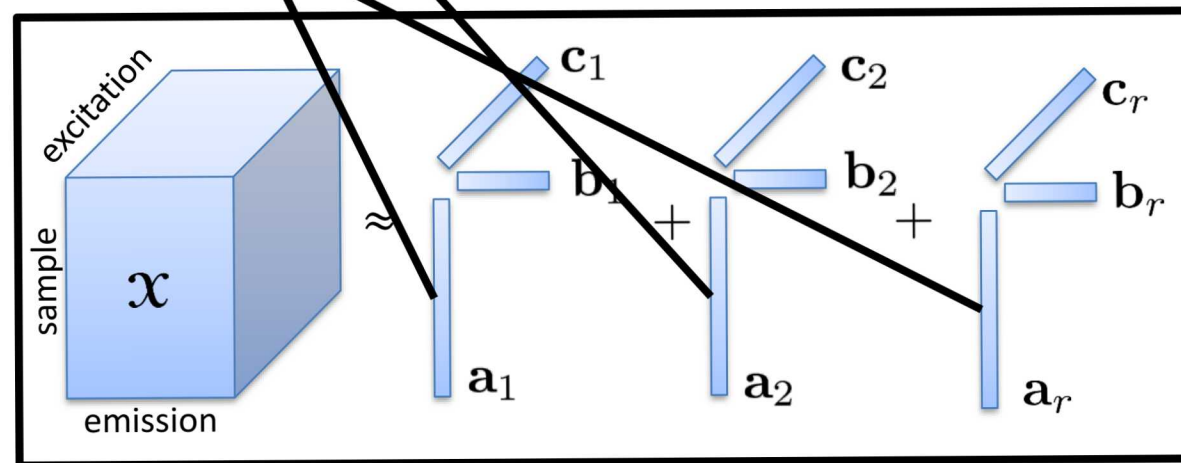
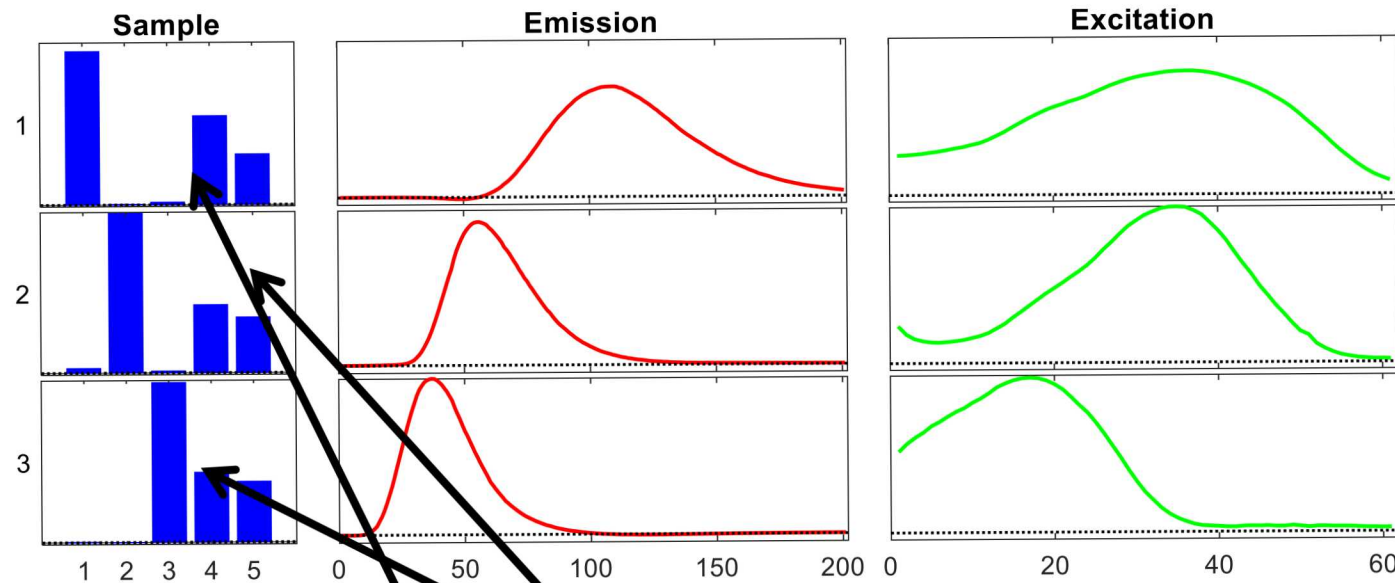


Each Component is One Row of Visualization

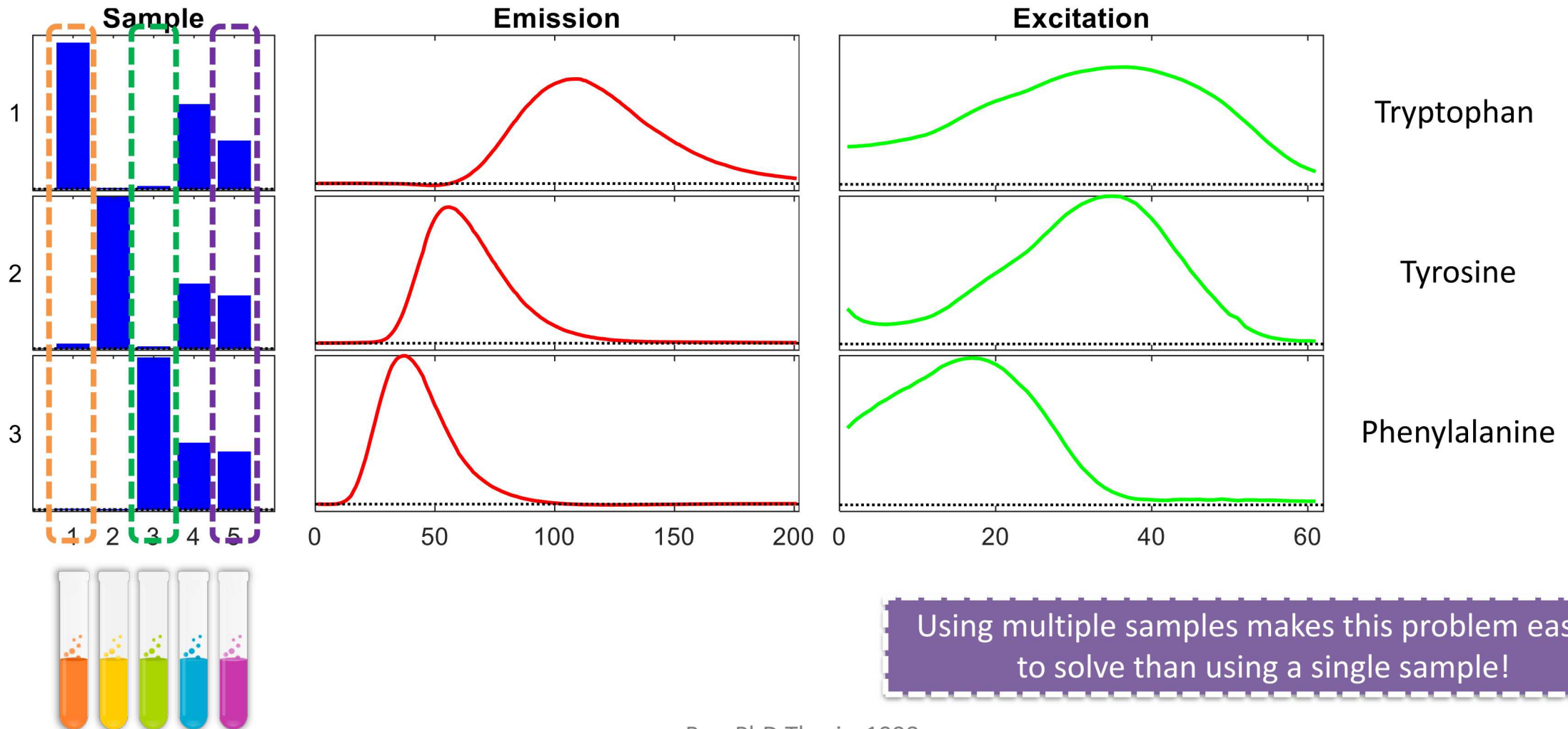
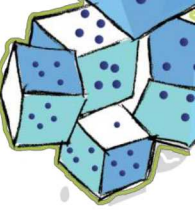


Bro, PhD Thesis, 1998.

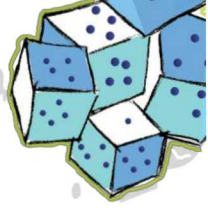
First Column is all the “a” Vectors



Rank-3 Tensor Decomposition Shows Sample of Each Amino Acid in Each Sample



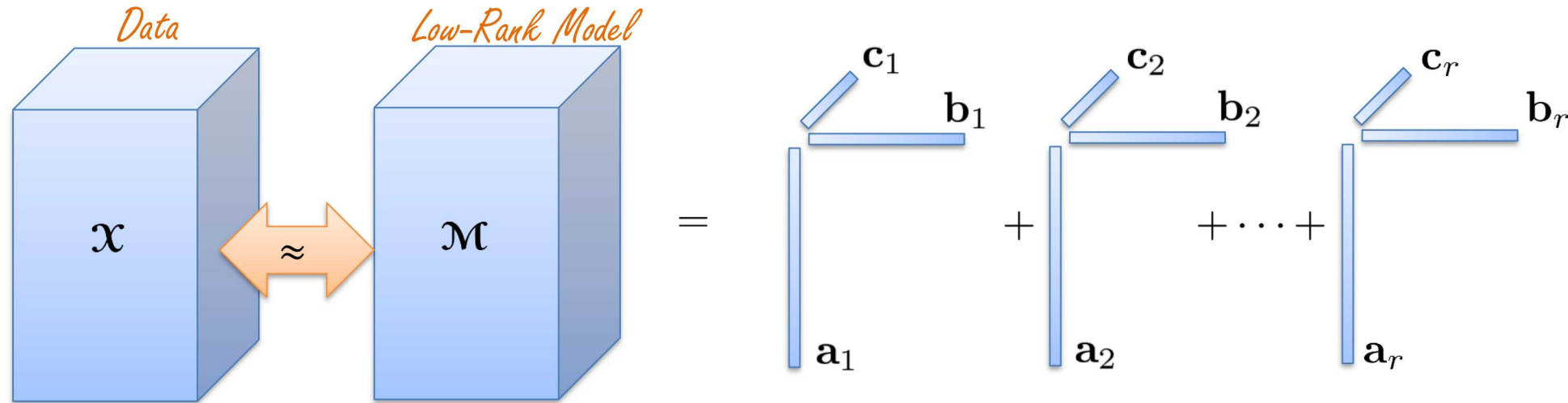
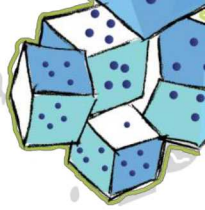
Bro, PhD Thesis, 1998.



How to Find a Low-Rank Model

(This involves some math!)

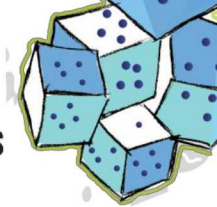
Formulate CP Tensor Decomposition as an Optimization Problem



$$\underset{m \times n \times p}{\mathcal{X}} \approx \underset{m \times n \times p}{\mathcal{M}} = \sum_{\alpha=1}^r \mathbf{a}_{\alpha} \circ \mathbf{b}_{\alpha} \circ \mathbf{c}_{\alpha} = \underset{m \times r}{[\mathbf{A}, \mathbf{B}, \mathbf{C}]} \underset{p \times r}{\overset{n \times r}{\downarrow}}$$

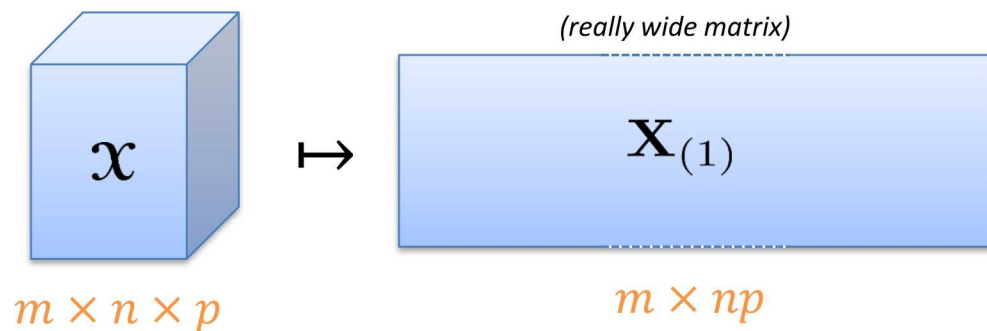
Optimization Problem: $\min \|\mathcal{X} - \mathcal{M}\|^2 \equiv \sum_i \sum_j \sum_k (x_{ijk} - m_{ijk})^2$

Kolda & Bader, SIREV, 2009



Rewriting Things in Terms of Matrices!

Unfolding tensor into a matrix reorganizes the entries

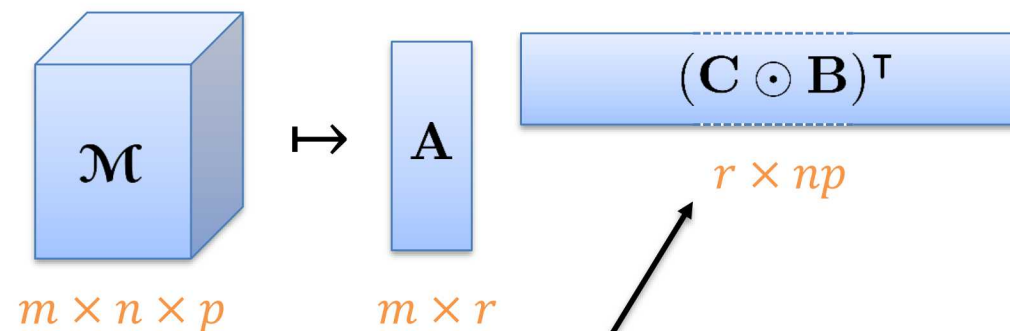


$\mathbf{X}_{(1)} = m \times np$ matrix, $(i, j, k) \mapsto (i, j + (n - 1)k)$

$\mathbf{X}_{(2)} = n \times mp$ matrix, $(i, j, k) \mapsto (j, i + (m - 1)k)$

$\mathbf{X}_{(3)} = p \times mn$ matrix, $(i, j, k) \mapsto (k, i + (m - 1)j)$

Unfolded model can be written as matrix multiply



$\mathbf{C} \odot \mathbf{B}$ is the "**Khatri-Rao**" product, a matrix of size $np \times r$ where entry $(j + (n - 1)k, r)$ is $b_{jr}c_{kr}$

$$\mathbf{M}_{(1)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^T$$

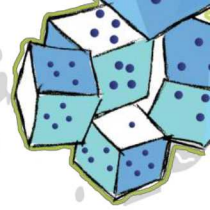
$$\mathbf{M}_{(2)} = \mathbf{B}(\mathbf{C} \odot \mathbf{A})^T$$

$$\mathbf{M}_{(3)} = \mathbf{C}(\mathbf{B} \odot \mathbf{A})^T$$



Kolda & Bader, SIREV, 2009

Computing the Gradient of the Optimization Problem using Unfoldings



$$\min F(\mathbf{A}, \mathbf{B}, \mathbf{C}) \equiv \frac{1}{2} \|\mathbf{X} - \mathbf{M}\|^2 \text{ subject to } \mathbf{M} = \llbracket \mathbf{A}, \mathbf{B}, \mathbf{C} \rrbracket$$

Rewrite in terms of the mode-1 unfolding...



$$F(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \frac{1}{2} \|\mathbf{X}_{(1)} - \mathbf{M}_{(1)}\|_F^2 \quad \text{with} \quad \mathbf{M}_{(1)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^\top$$

Calculate the derivative using the chain rule...

$$\begin{aligned} \frac{\partial F}{\partial \mathbf{A}} &= (\mathbf{X}_{(1)} - \mathbf{M}_{(1)}) \frac{\partial (\mathbf{X}_{(1)} - \mathbf{M}_{(1)})}{\partial \mathbf{A}} \\ &= (\mathbf{X}_{(1)} - \mathbf{M}_{(1)}) (\mathbf{C} \odot \mathbf{B}) \quad \text{Elementwise Product} \\ &= \underbrace{\mathbf{X}_{(1)}}_{m \times np} \underbrace{(\mathbf{C} \odot \mathbf{B})}_{np \times r} - \underbrace{\mathbf{A}}_{m \times r} \underbrace{(\mathbf{C}^\top \mathbf{C} * \mathbf{B}^\top \mathbf{B})}_{r \times r} \end{aligned}$$

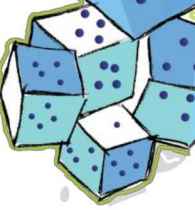
$$\frac{\partial F}{\partial \mathbf{A}} = \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B}) - \mathbf{A}(\mathbf{C}^\top \mathbf{C} * \mathbf{B}^\top \mathbf{B})$$

$$\frac{\partial F}{\partial \mathbf{B}} = \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A}) - \mathbf{B}(\mathbf{C}^\top \mathbf{C} * \mathbf{A}^\top \mathbf{A})$$

$$\frac{\partial F}{\partial \mathbf{C}} = \mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A}) - \mathbf{C}(\mathbf{B}^\top \mathbf{B} * \mathbf{A}^\top \mathbf{A})$$

Acar, Dunlavy, Kolda, J. Chem., 2011

Two Options for Optimization: Alternating or Direct



$$\min F(\mathbf{A}, \mathbf{B}, \mathbf{C}) \equiv \frac{1}{2} \|\mathbf{X} - \mathcal{M}\|^2 \text{ subject to } \mathcal{M} = [\mathbf{A}, \mathbf{B}, \mathbf{C}]$$

$$\frac{\partial F}{\partial \mathbf{A}} = \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B}) - \mathbf{A}(\mathbf{C}^\top \mathbf{C} * \mathbf{B}^\top \mathbf{B})$$

$$\frac{\partial F}{\partial \mathbf{B}} = \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A}) - \mathbf{B}(\mathbf{C}^\top \mathbf{C} * \mathbf{A}^\top \mathbf{A})$$

$$\frac{\partial F}{\partial \mathbf{C}} = \mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A}) - \mathbf{C}(\mathbf{B}^\top \mathbf{B} * \mathbf{A}^\top \mathbf{A})$$

Option 1: Alternating Least Squares

Repeat until convergence...

$$\mathbf{A} \leftarrow \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B})(\mathbf{C}^\top \mathbf{C} * \mathbf{B}^\top \mathbf{B})^{-1}$$

$$\mathbf{B} \leftarrow \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A})(\mathbf{C}^\top \mathbf{C} * \mathbf{A}^\top \mathbf{A})^{-1}$$

$$\mathbf{C} \leftarrow \mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A})(\mathbf{B}^\top \mathbf{B} * \mathbf{A}^\top \mathbf{A})^{-1}$$

Advantages:

- Convex subproblems
- Guaranteed decrease at each step

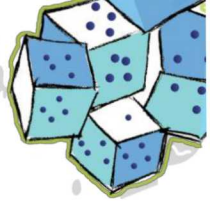
Option 2: Direct Optimization

Repeat until convergence...

$$[\mathbf{A}, \mathbf{B}, \mathbf{C}] \leftarrow [\mathbf{A}, \mathbf{B}, \mathbf{C}] - \alpha \left[\frac{\partial F}{\partial \mathbf{A}}, \frac{\partial F}{\partial \mathbf{B}}, \frac{\partial F}{\partial \mathbf{C}} \right]$$

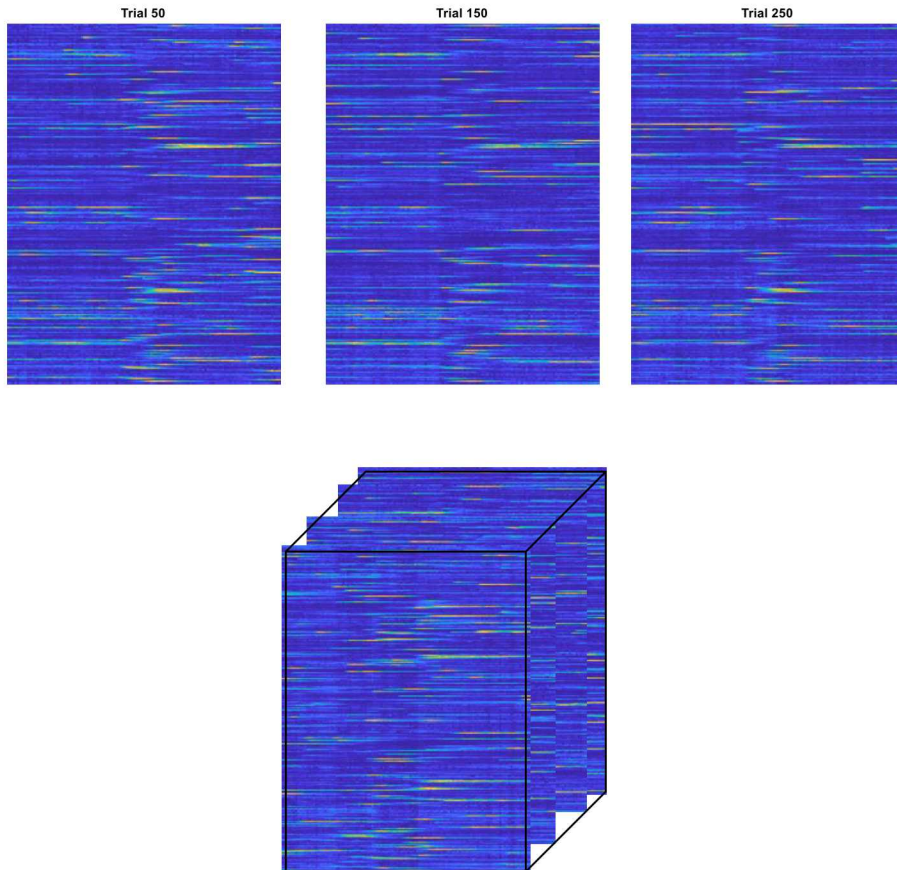
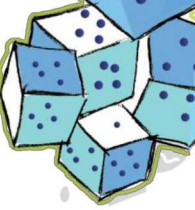
Advantages:

- Can use any optimization method!
- Easy to handle missing data

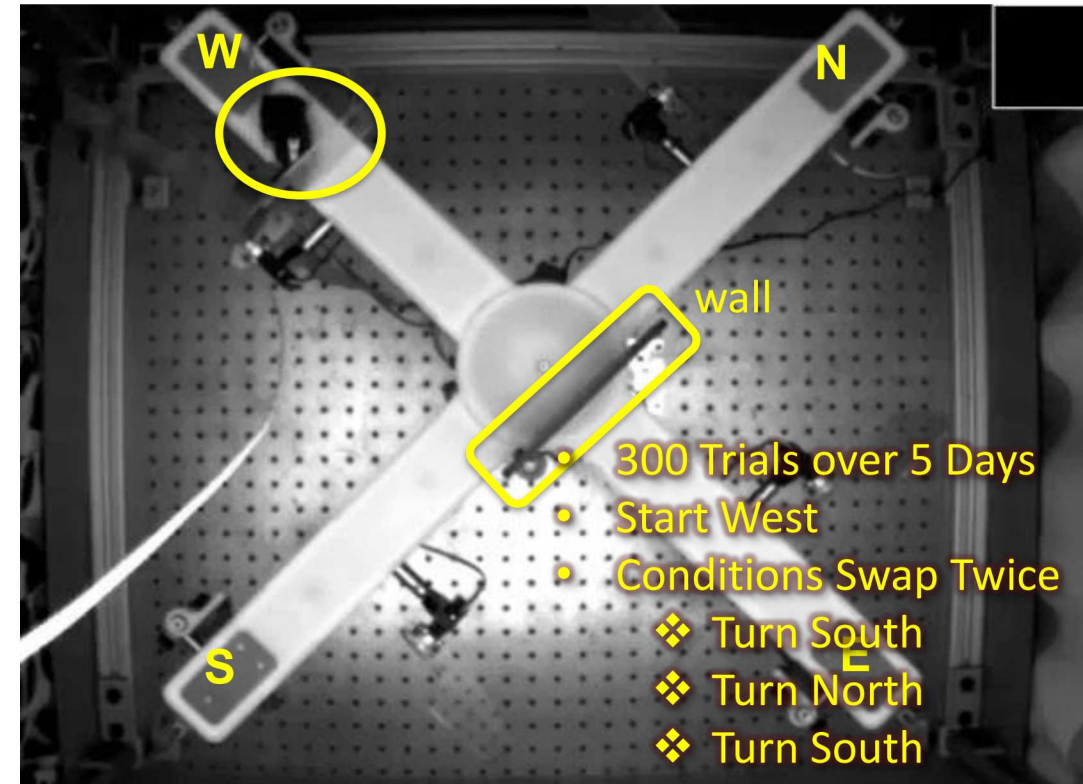


Tensor Decomposition for
Unsupervised Learning

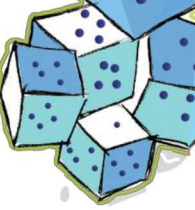
In Neuroscience, Tensor Decomposition Uncovers Patterns in Neuron Activity



282 neurons $\times$ 111 time bins $\times$ 300 trials



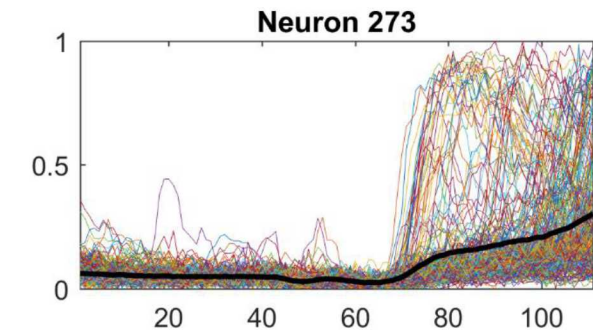
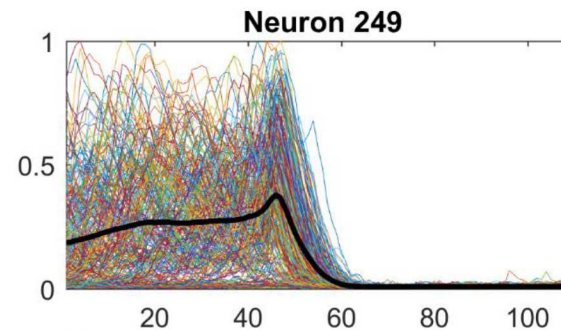
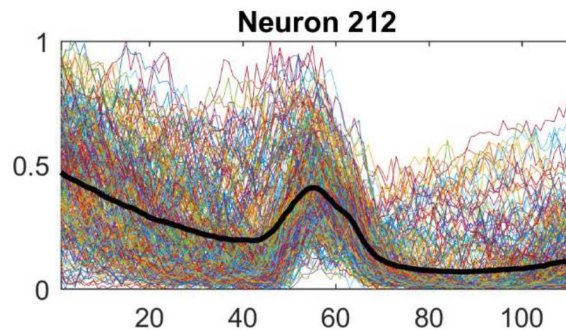
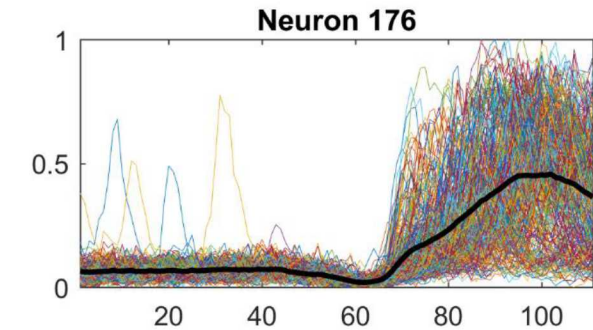
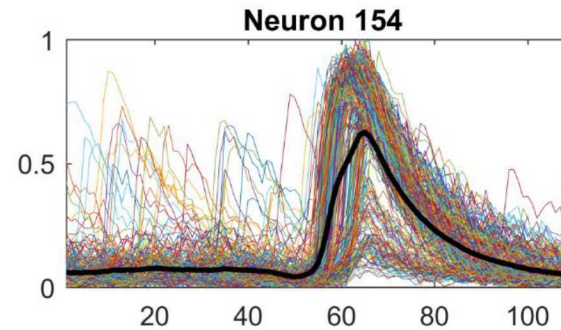
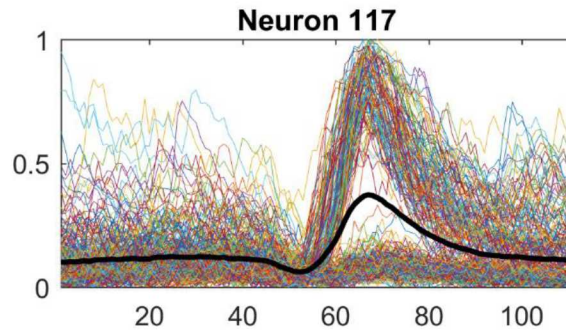
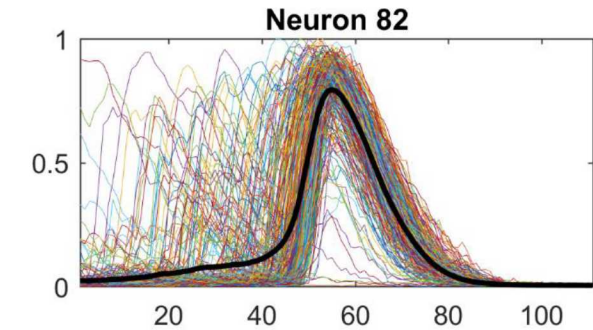
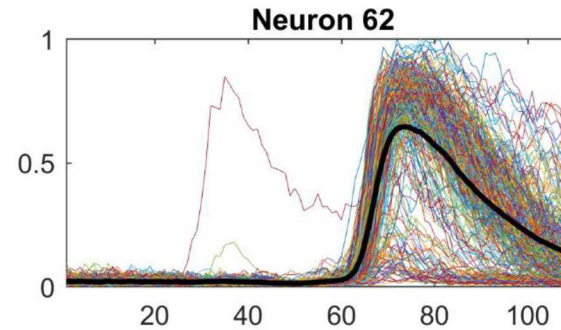
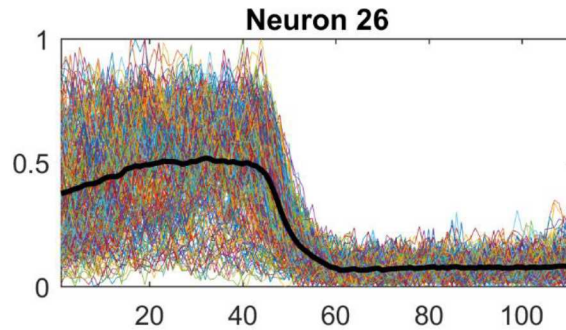
Williams et al., Neuron, 2018



Example Neuron Activity

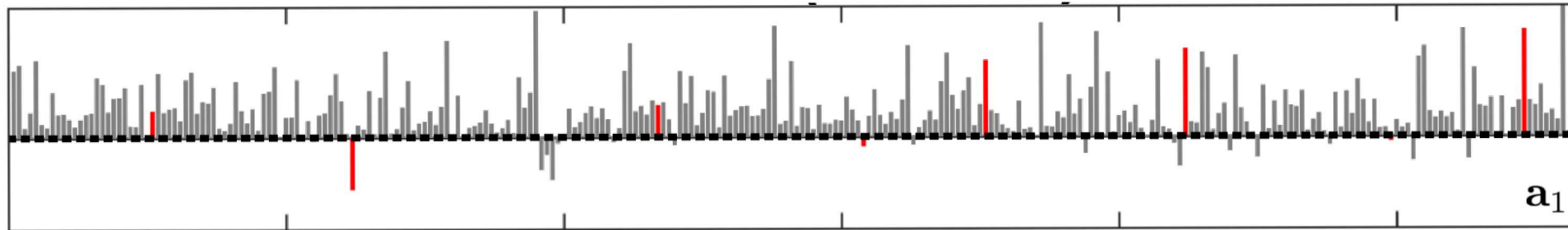
Thin lines
show 300
individual
trials

Thick line is
average

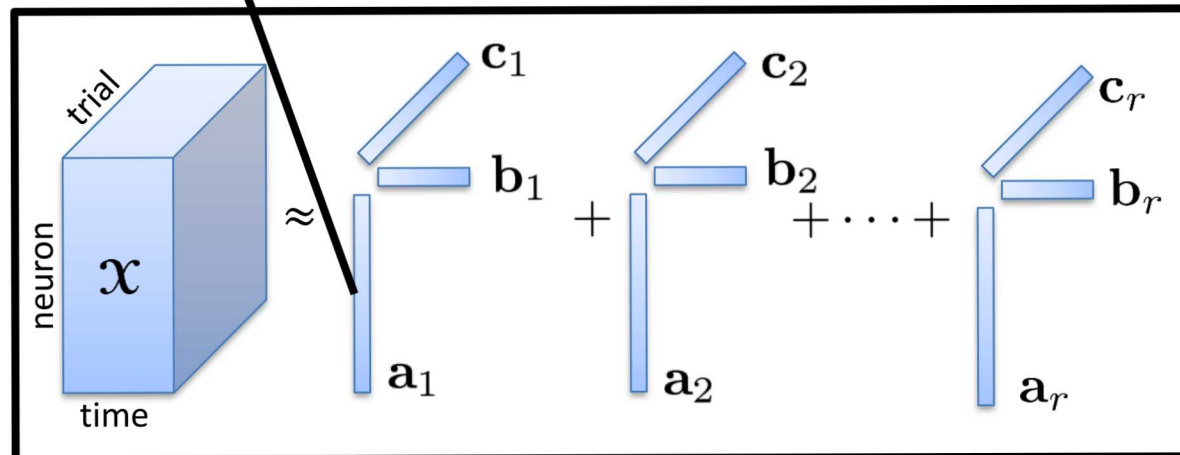


Hong, Kolda, Duersch, arXiv, 2018

Neuron Factor Vector Visualized as Bar Chart

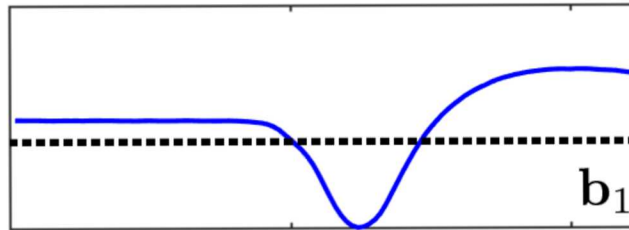


Neuron Modes Plotted as a Bar Chart
(Red Lines Correspond to Examples in Previous Slide)

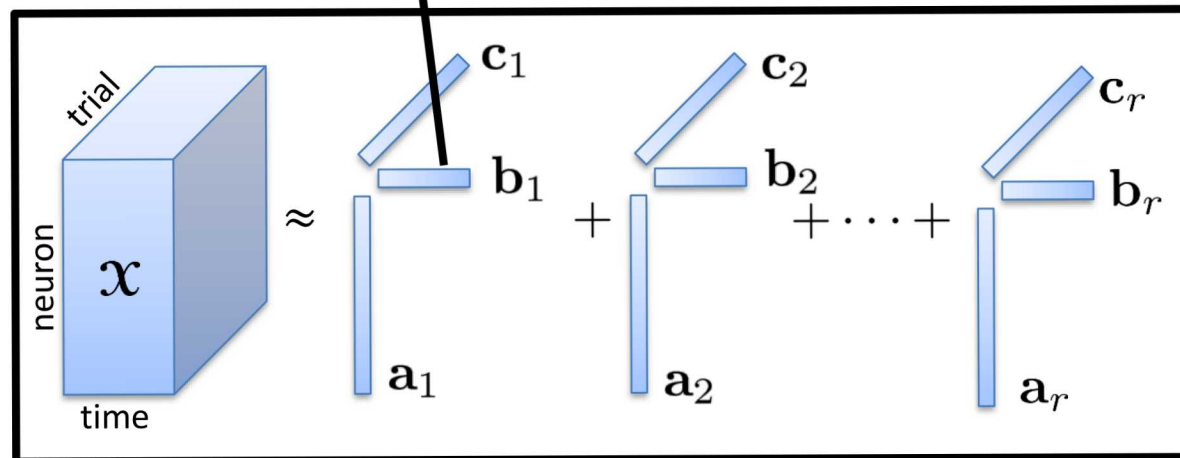


Hong, Kolda, Duersch, arXiv, 2018

Time Factor Vector Visualized as Line

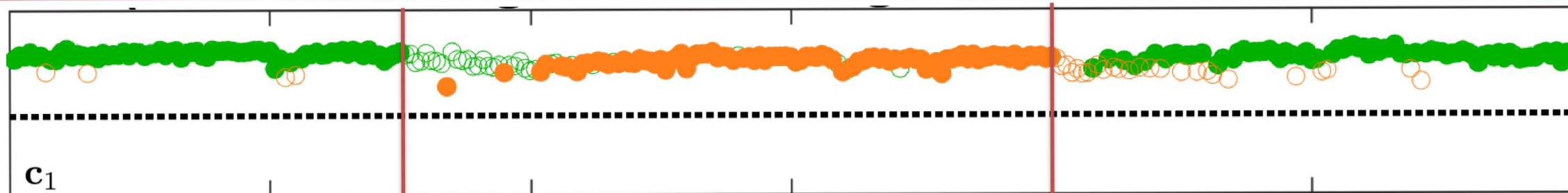


Time (within trial) Plotted as a Line
(Dashed Line is Zero)



Hong, Kolda, Duersch, arXiv, 2018

Trial Factor Vector Visualized as Color-Coded Scatter Plot



Rule
Change

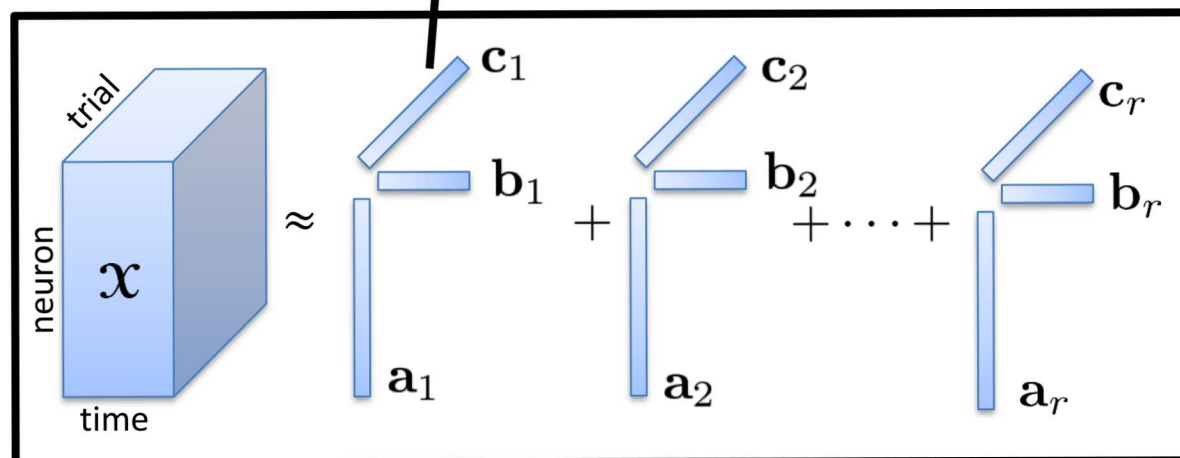
Trial Plotted as Scatter Graph

Right turn = Green

Left turn = Orange

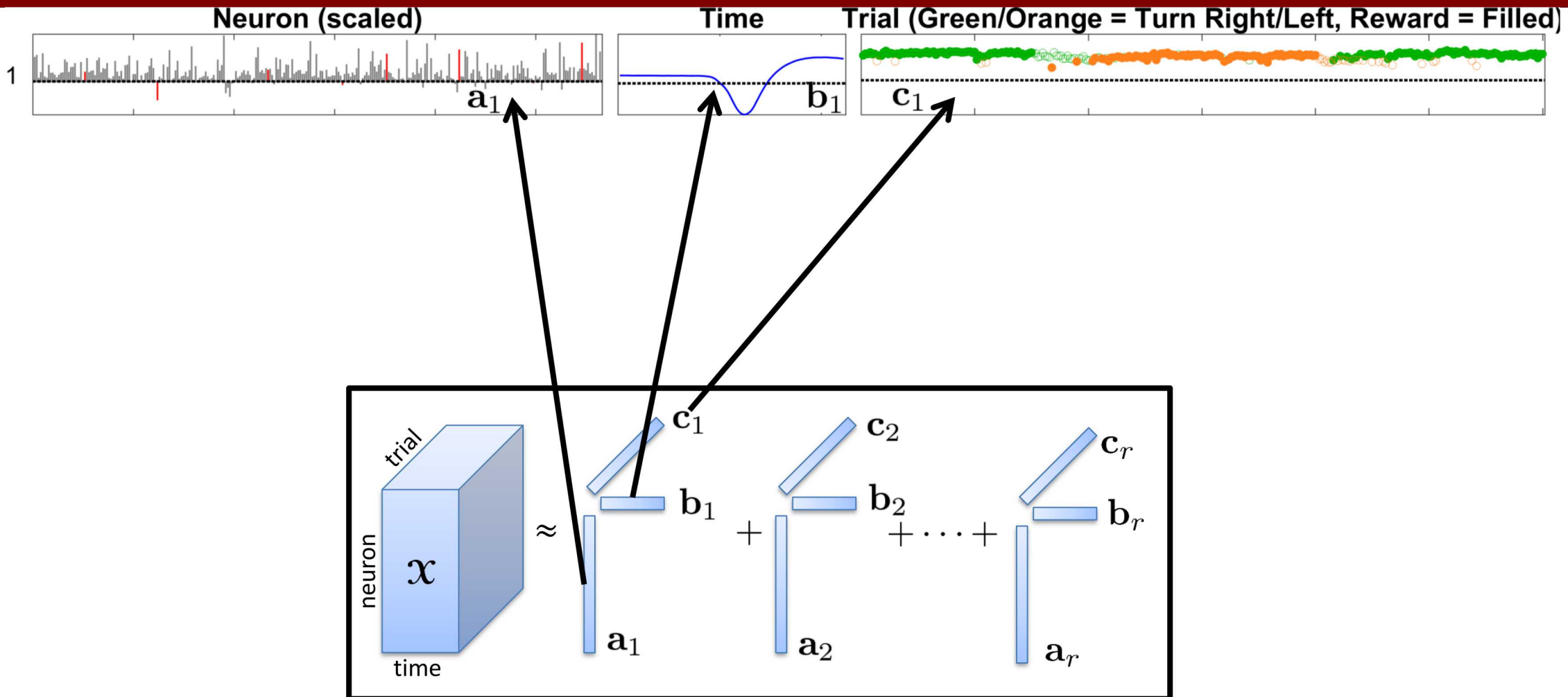
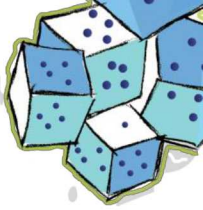
Filled = Reward

Rule
Change



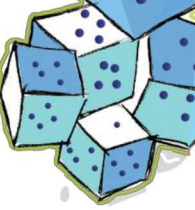
Hong, Kolda, Duersch, arXiv, 2018

Visualization of CP Tensor Decomposition Shows the Factors (Vectors)

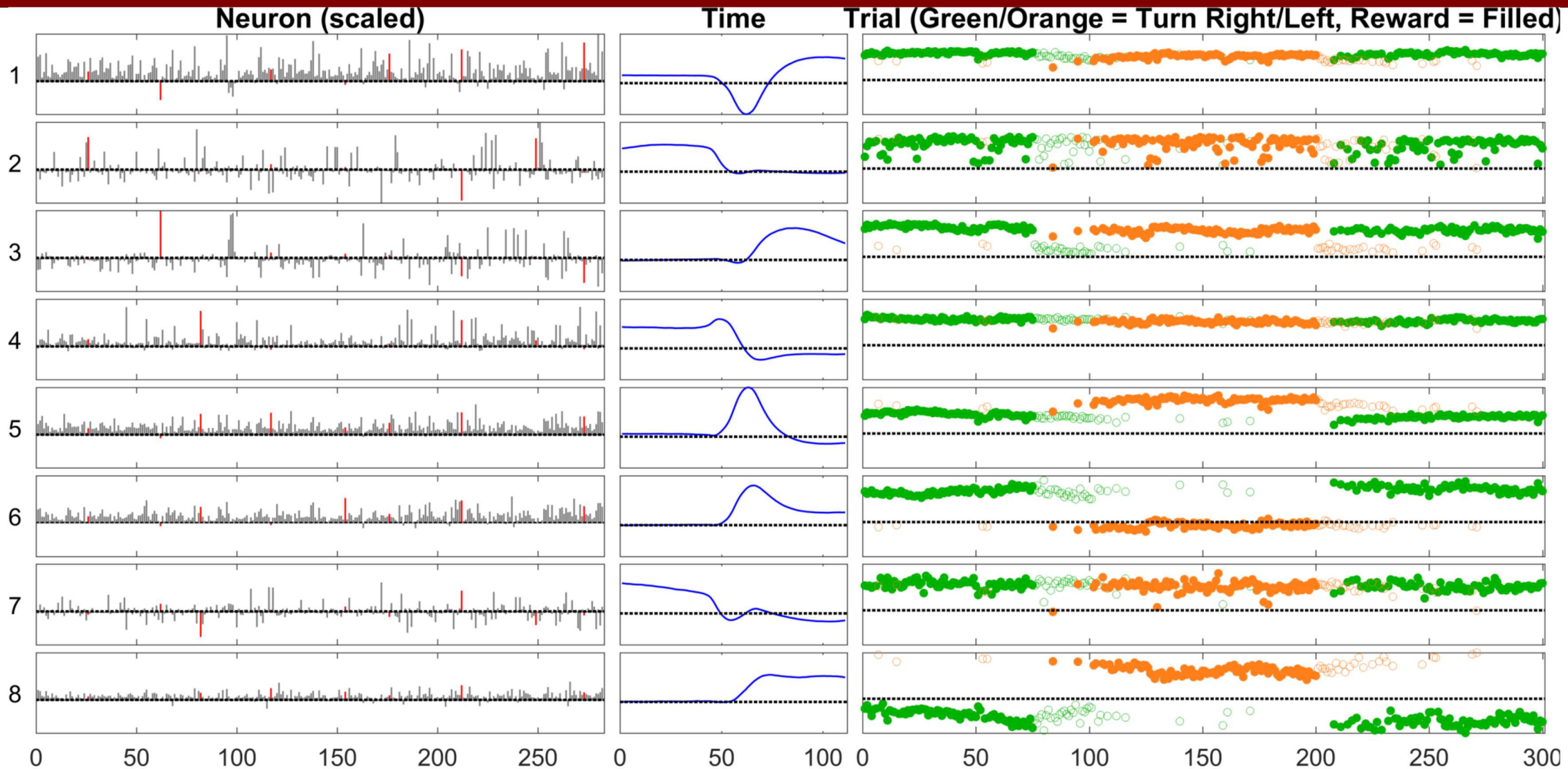


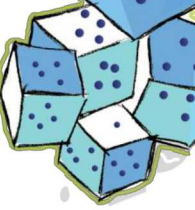
Hong, Kolda, Duersch, arXiv, 2018



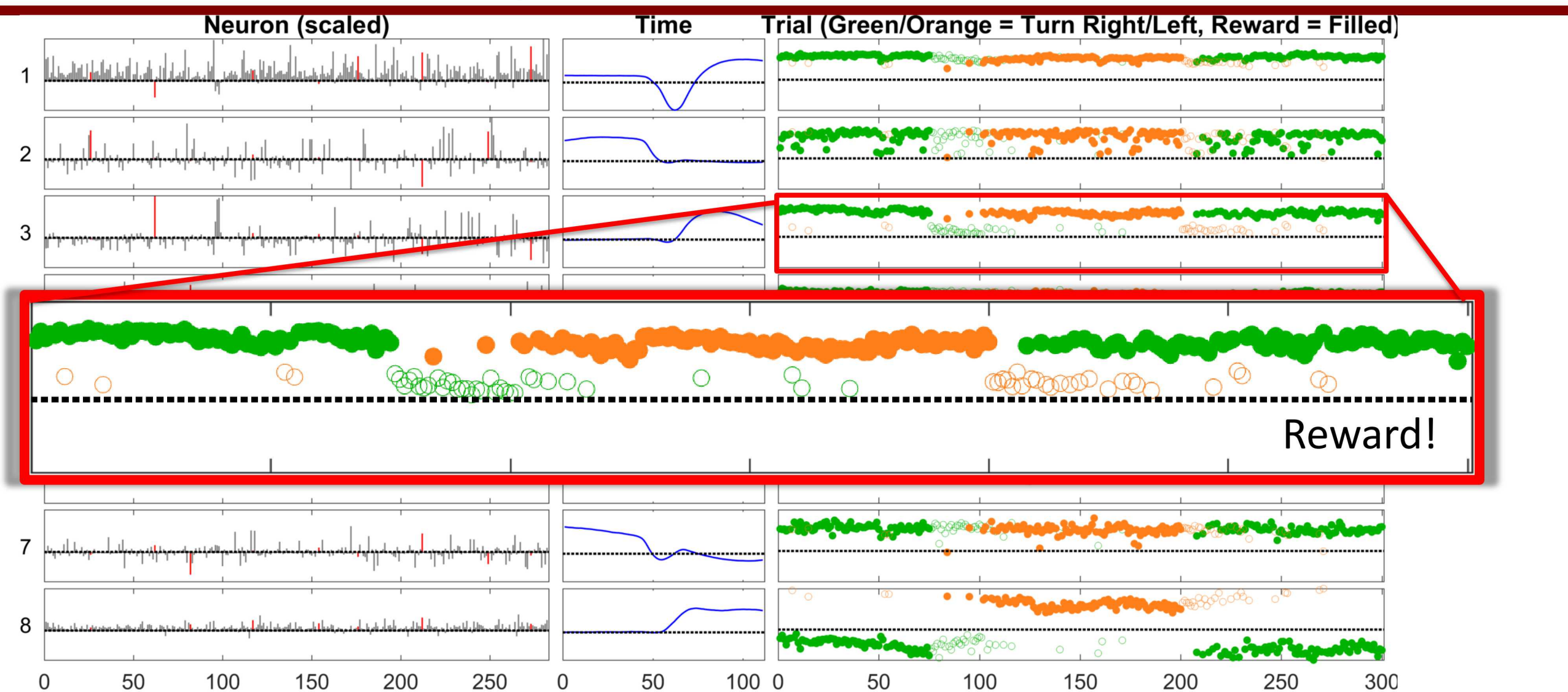


Rank-8 Decomposition of Mouse Data

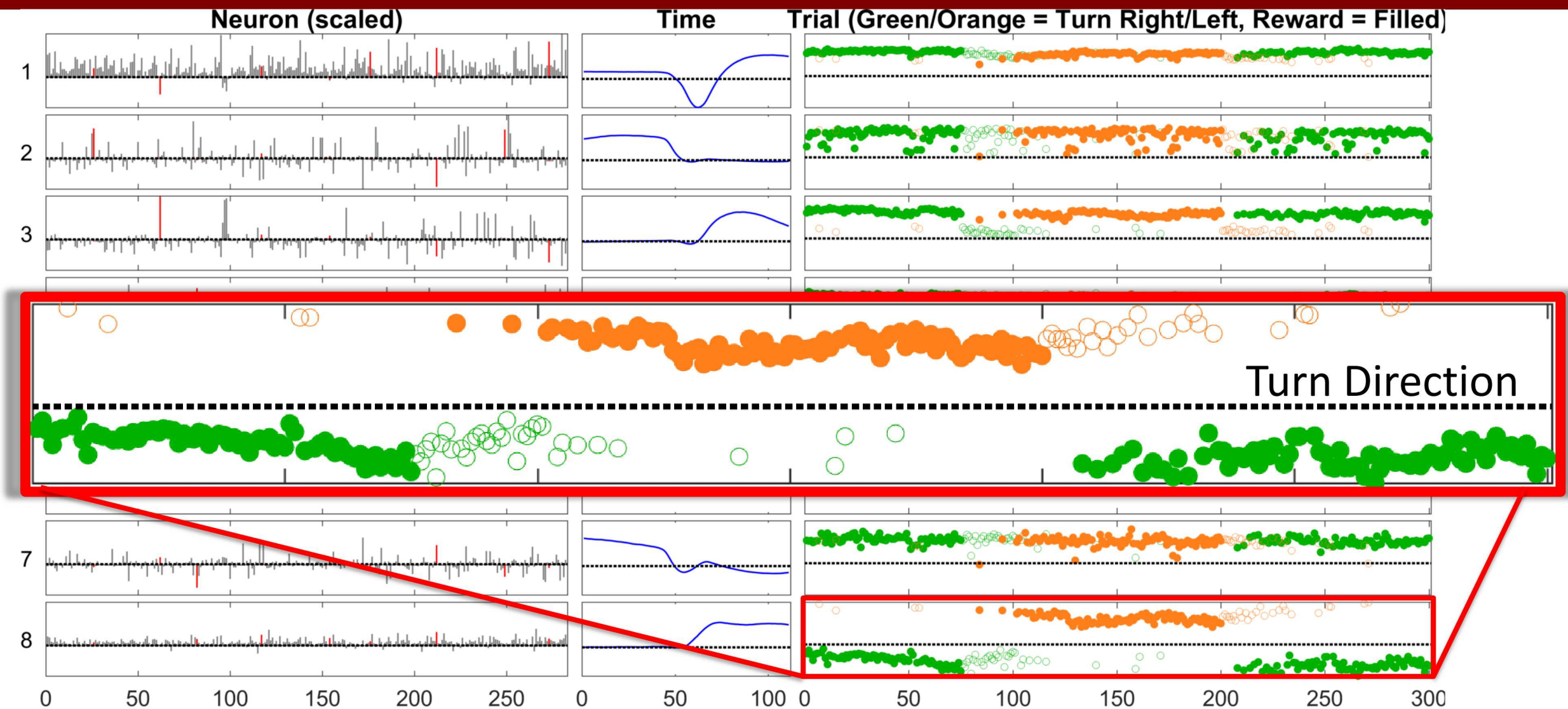
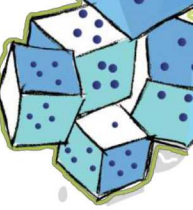




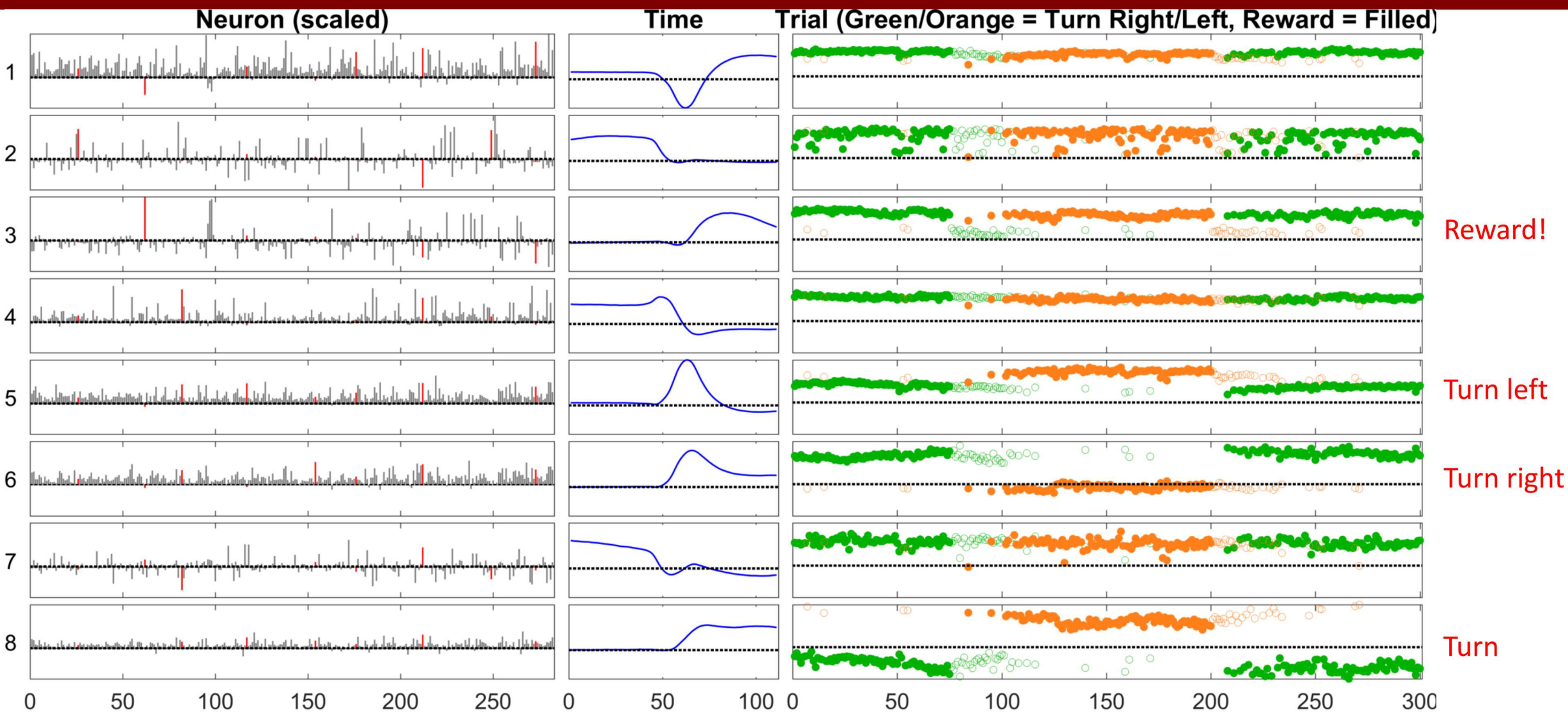
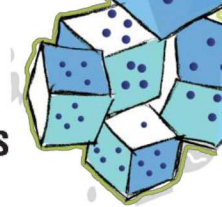
CP Tensor Decomposition “Sees” Reward

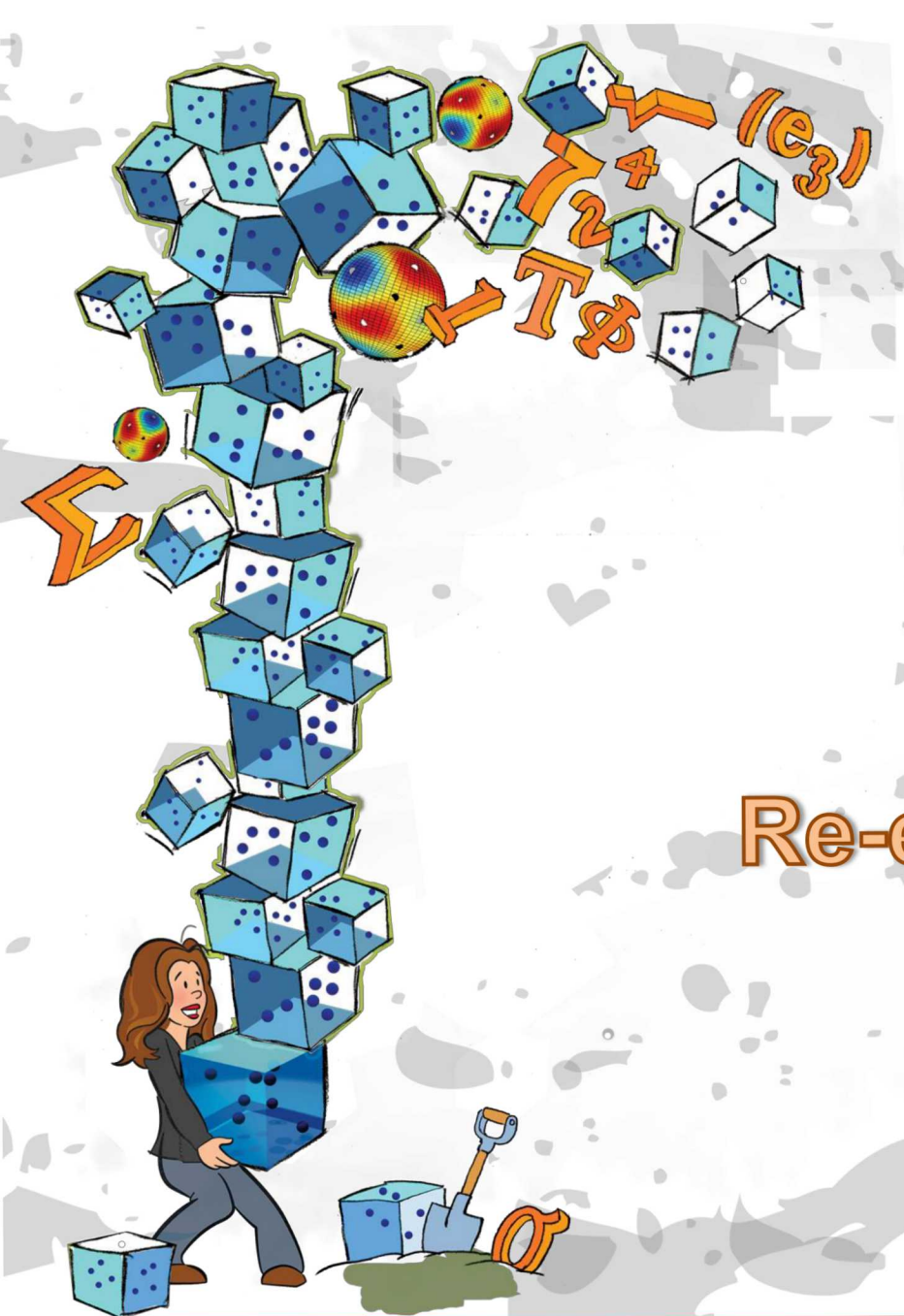
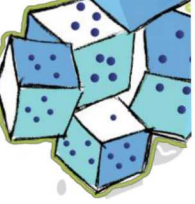


CP Tensor Decomposition “Sees” Turn Direction



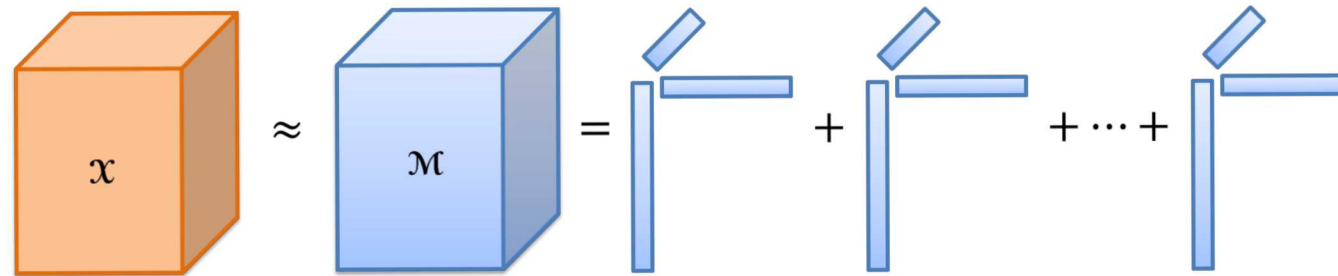
CP Tensor Decomposition “Sees” Turn Direction and if Reward Received





Re-examining the Fitting Function

Generalizing CP Loss Function (GCP)



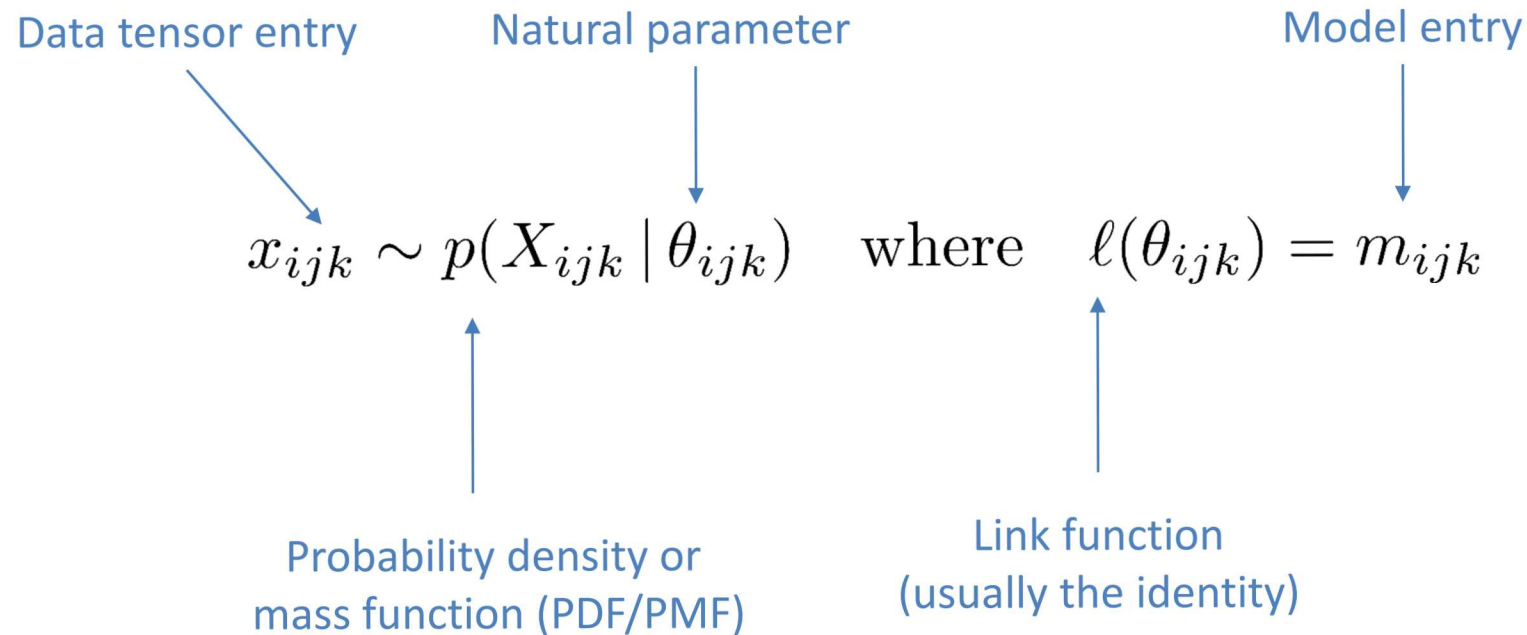
Usual CP loss function: $F(\mathbf{A}, \mathbf{B}, \mathbf{C}) \equiv \sum_{ijk} (x_{ijk} - m_{ijk})^2$ with $\mathbf{M}_{(1)} = \mathbf{A}(\mathbf{C} \odot \mathbf{B})^\top$

Generalized CP (GCP) loss function: $F(\mathbf{A}, \mathbf{B}, \mathbf{C}) \equiv \sum_{ijk} f(x_{ijk}, m_{ijk})$

Motivation: To handle non-Gaussian data, such as discrete or nonnegative.

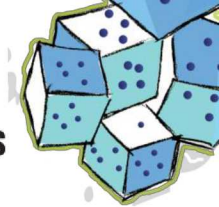
Statistical Framework for Loss Function

Goal: Maximize the *likelihood* of a model that parameterizes a distribution



Hong, Kolda, Duersch, arXiv, 2018

Statistical Framework for Loss Function



Goal: Maximize the *likelihood* of a model that parameterizes a distribution

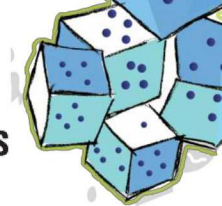
$$x_{ijk} \sim p(x_{ijk} \mid \theta_{ijk}) \quad \text{where} \quad \ell(\theta_{ijk}) = m_{ijk}$$

Because the entries of the data tensor ($\mathcal{X}$) are conditionally independent, the **maximum likelihood estimate** of model ($\mathcal{M}$) is the solution to

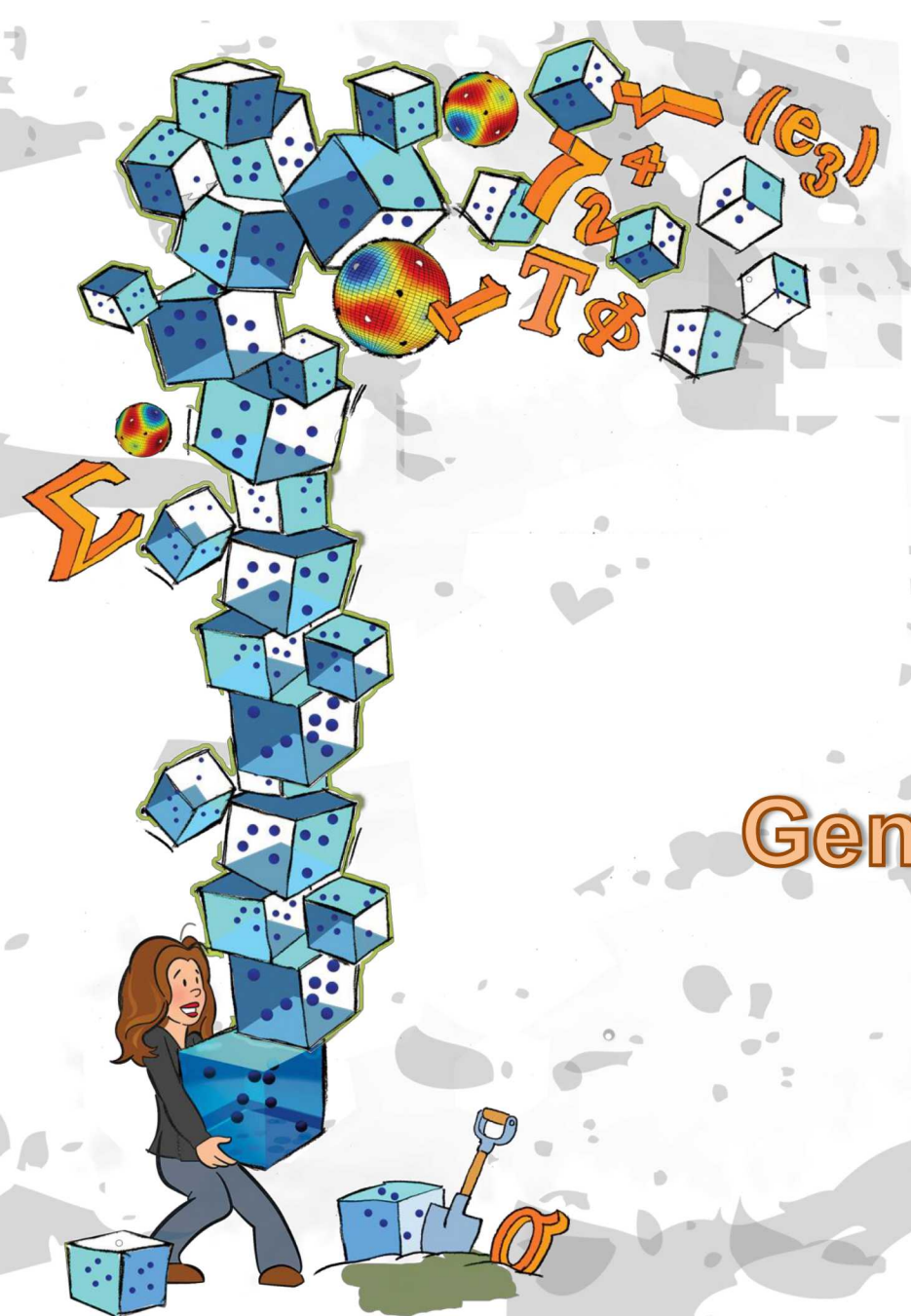
$$\max_{\mathcal{M}} L(\mathcal{M}; \mathcal{X}) \equiv \prod_{ijk} p(x_{ijk} \mid \theta_{ijk}) \quad \text{with} \quad \ell(\theta_{ijk}) = m_{ijk}$$

We instead work with the **negative log-likelihood** (and eliminate θ):

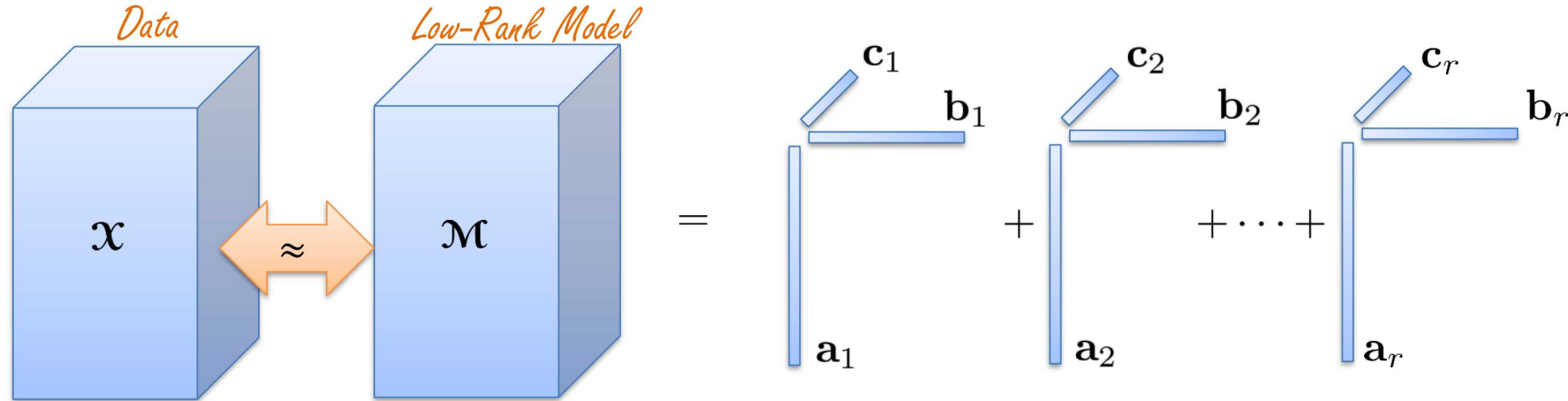
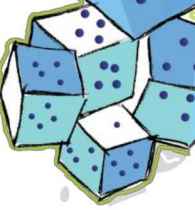
$$\min F(\mathcal{M}; \mathcal{X}) \equiv \sum_{ijk} f(x_{ijk}, m_{ijk}) \quad \text{where} \quad f(x_{ijk}, m_{ijk}) \equiv -\log p(x_{ijk} \mid \ell^{-1}(m_{ijk}))$$



Generalized CP



Generalized CP Optimization Problem Requires User-Defined Loss Function

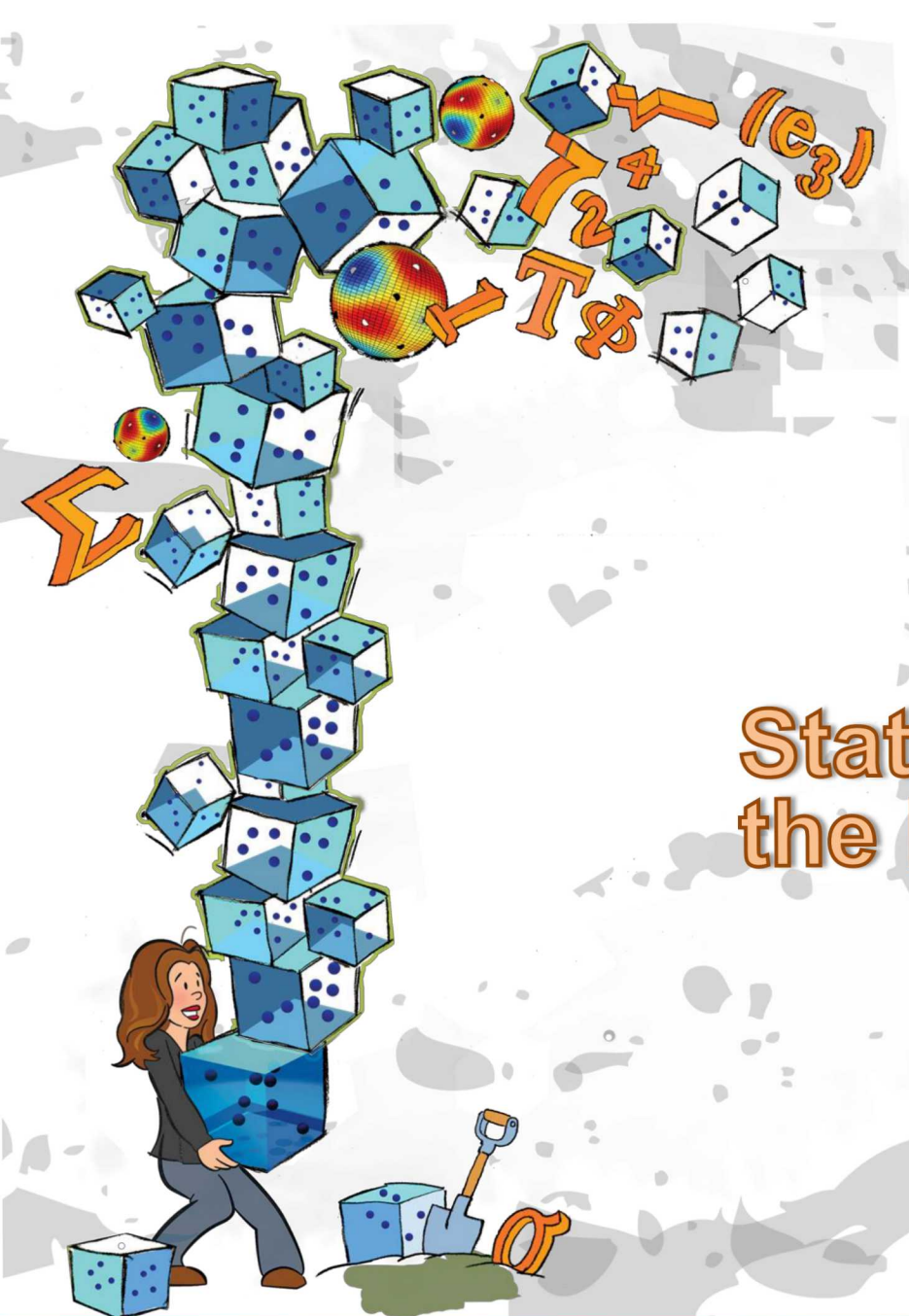
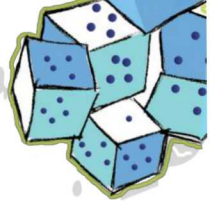


$$\mathcal{X} \approx \mathcal{M} = \sum_{\alpha=1}^r \mathbf{a}_{\alpha} \circ \mathbf{b}_{\alpha} \circ \mathbf{c}_{\alpha} = [\mathbf{A}, \mathbf{B}, \mathbf{C}]$$

$f(x, m)$ provided by user

Optimization Problem: $\min \|\mathcal{X} - \mathcal{M}\|^2 \equiv \sum_i \sum_j \sum_k f(x_{ijk}, m_{ijk})$

Hong, Kolda, Duersch, arXiv, 2018



Statistically-Motivated Choices for the Loss Function

Usually Assume Low-Rank plus White Noise

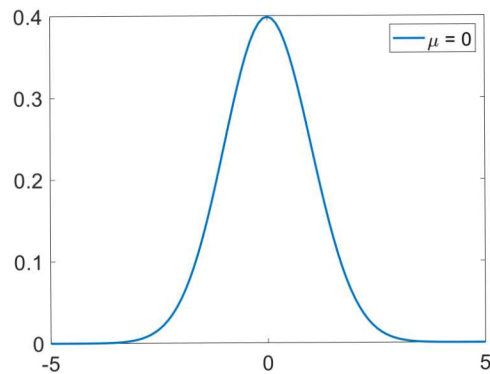
Standard assumption: Data has low-rank structure but contaminated by uncorrelated white noise with **constant standard deviation** σ :

$$x_{ijk} = m_{ijk} + \epsilon_{ijk} \quad \text{with} \quad \epsilon_{ijk} \sim \mathcal{N}(0, \sigma)$$



$$x_{ijk} \sim \mathcal{N}(\mu_{ijk}, \sigma) \quad \text{with} \quad \mu_{ijk} = m_{ijk}$$

Distribution of White Noise
with $\sigma = 1$



PDF for Normal Distribution

$$p(x \mid \mu, \sigma) = \frac{e^{-(x-\mu)^2 / 2\sigma^2}}{\sqrt{2\pi\sigma^2}}$$

and

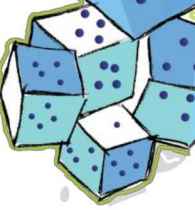
Link Function

$$m = \mu$$

σ constant

Hong, Kolda, Duersch, arXiv, 2018

Standard CP Corresponds to Gaussian Distribution Assumption



PDF for Normal Distribution

$$p(x | \mu, \sigma) = \frac{e^{-(x-\mu)^2 / 2\sigma^2}}{\sqrt{2\pi\sigma^2}}$$

and

Link Function

$$m = \mu$$

σ constant

Negative log-likelihood:

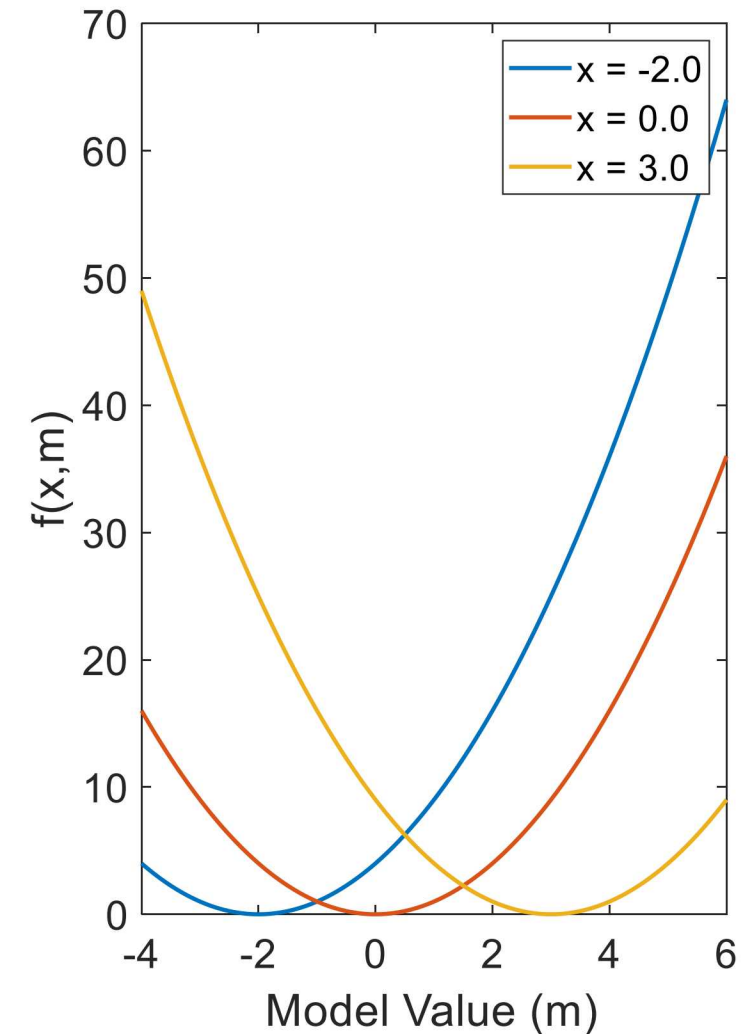
$$-\log p(x|\mu, \sigma) = \frac{(x-u)^2}{2\sigma^2} + \frac{1}{2} \log(2\pi\sigma^2)$$

Eliminate natural parameter
via link function:

$$f(x, m) = \frac{(x-m)^2}{2\sigma^2} + \frac{1}{2} \log(2\pi\sigma^2)$$

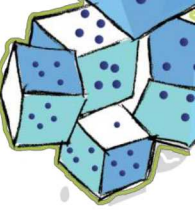
Eliminate constants:

$$f(x, m) = (x - m)^2$$



Hong, Kolda, Duersch, arXiv, 2018

Consider Bernoulli for Binary Data (Version 1: Odds Link)



Bernoulli random variable

$$x \in \{0, 1\}$$

ρ = probability of a 1

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}, \quad x \in \{0, 1\}$$

PDF for Bernoulli Distribution

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}$$

$$x \in \{0, 1\}$$

and

Link Function

$$m = \frac{\rho}{(1 - \rho)}$$

**Odds
Link**

$$\ell(\rho) = \rho / (1 - \rho)$$

$$\ell^{-1}(m) = m / (1 + m)$$

Odds (m)	Probability (ρ)
¼	20%
1	50%
4	80%
10	90%

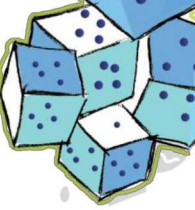
Negative log-likelihood:

$$-\log p(x | \rho) = \log \frac{1}{1 - \rho} - x \log \frac{\rho}{1 - \rho}$$

Eliminate natural parameter
via link function:

$$f(x, m) = \log(1 + m) - x \log m \quad \text{for } m > 0$$

Consider Bernoulli for Binary Data (Version 1: Odds Link)

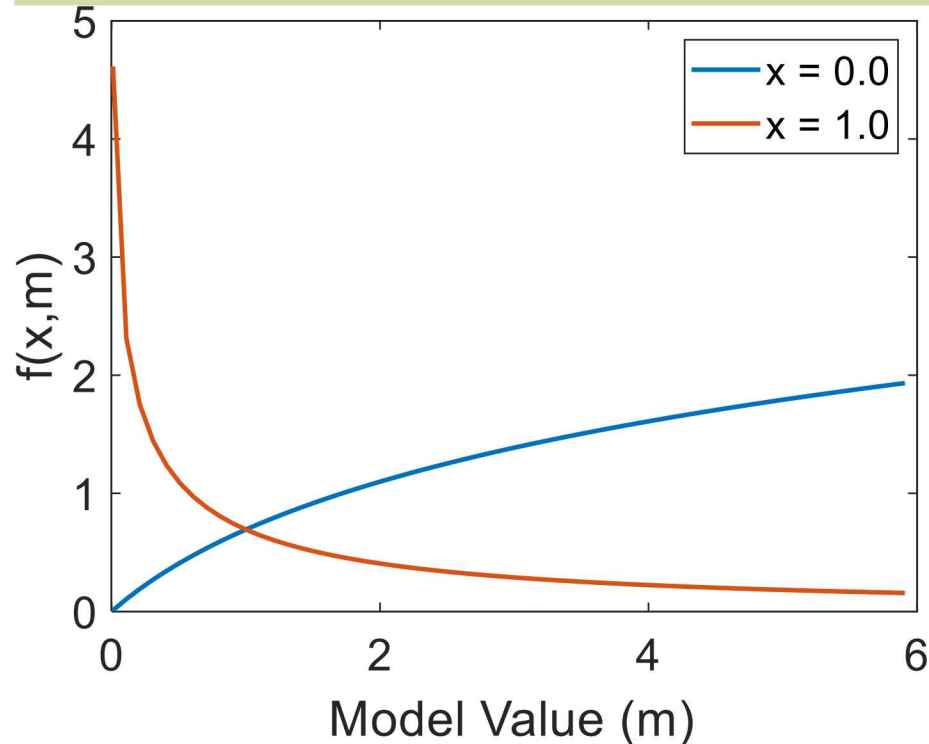


Bernoulli random variable

$$x \in \{0, 1\}$$

ρ = probability of a 1

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}, \quad x \in \{0, 1\}$$



PDF for Bernoulli Distribution

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}$$

$$x \in \{0, 1\}$$

and

Link Function

$$m = \frac{\rho}{(1 - \rho)}$$

Negative log-likelihood:

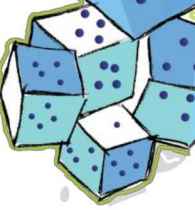
$$-\log p(x | \rho) = \log \frac{1}{1 - \rho} - x \log \frac{\rho}{1 - \rho}$$

Eliminate natural parameter
via link function:

$$f(x, m) = \log(1 + m) - x \log m \quad \text{for } m > 0$$

Hong, Kolda, Duersch, arXiv, 2018

Consider Bernoulli for Binary Data (Version 2: LogitLink)



Bernoulli random variable

$$x \in \{0, 1\}$$

ρ = probability of a 1

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}, \quad x \in \{0, 1\}$$

PDF for Bernoulli Distribution

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}$$
$$x \in \{0, 1\}$$

and

Link Function

$$m = \log \frac{\rho}{(1-\rho)}$$

Logit
(Log-Odds)
Link

$$\ell(\rho) = \log(\rho / (1 - \rho))$$

$$\ell^{-1}(m) = e^m / (1 + e^m)$$

Log-Odds(m)	Probability (ρ)
-1.39	20%
0	50%
1.39	80%
2.30	90%

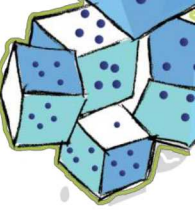
Negative log-likelihood:

$$-\log p(x | \rho) = \log \frac{1}{1 - \rho} - x \log \frac{\rho}{1 - \rho}$$

Eliminate natural parameter
via link function:

$$f(x, m) = \log(1 + e^m) - xm \quad \text{for } m \in \mathbb{R}$$

Consider Bernoulli for Binary Data (Version 2: LogitLink)

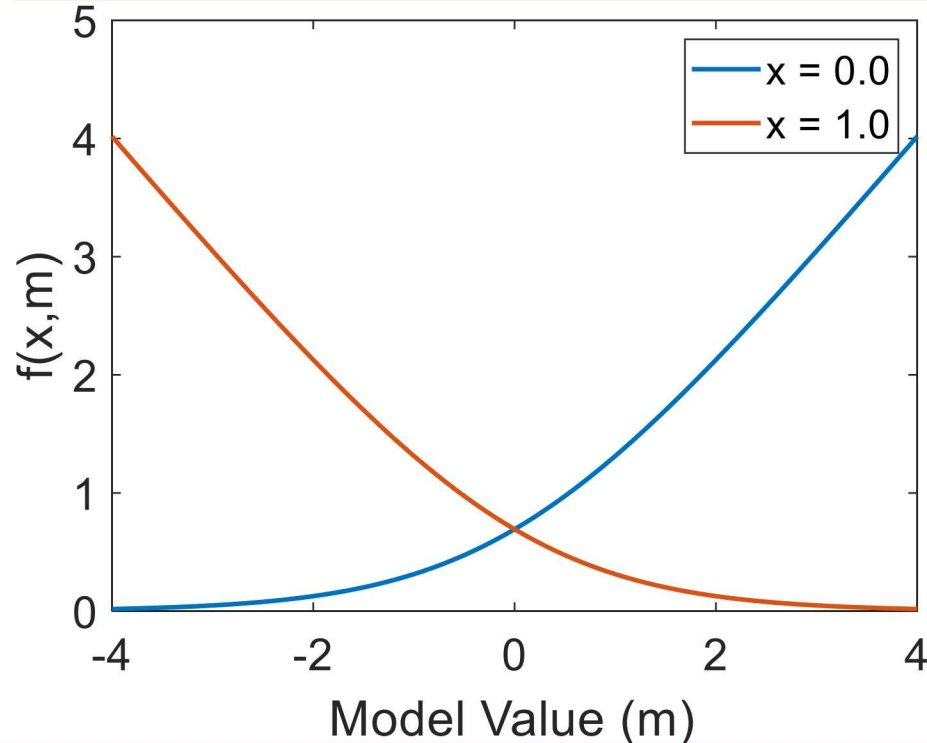


Bernoulli random variable

$$x \in \{0, 1\}$$

ρ = probability of a 1

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}, \quad x \in \{0, 1\}$$



PDF for Bernoulli Distribution

$$p(x | \rho) = \rho^x (1 - \rho)^{(1-x)}$$

$$x \in \{0, 1\}$$

and

Link Function

$$m = \log \frac{\rho}{(1-\rho)}$$

Negative log-likelihood:

$$-\log p(x | \rho) = \log \frac{1}{1 - \rho} - x \log \frac{\rho}{1 - \rho}$$

Eliminate natural parameter
via link function:

$$f(x, m) = \log(1 + e^m) - xm \quad \text{for } m \in \mathbb{R}$$

Hong, Kolda, Duersch, arXiv, 2018

Poisson Distribution for Count Data (Version 1: Identity Link)

PDF for Poisson Distribution

$$p\lambda = e^{-\lambda} \lambda^x / x!$$

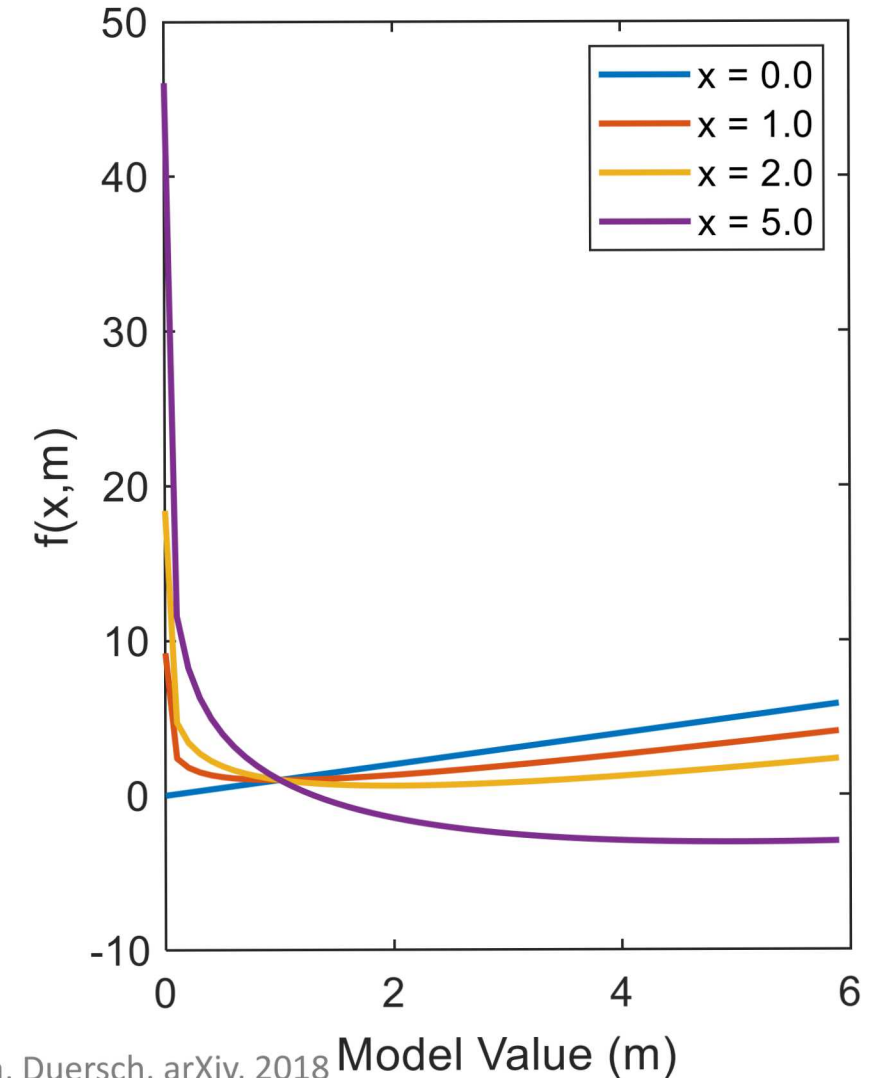
$$x \in \mathbb{N}$$

and

Link Function

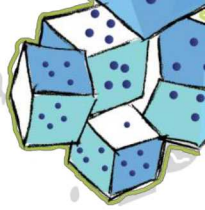
$$m = \lambda$$

$$f(x, m) = m - x \log m \quad \text{for } x \in \mathbb{N}, m > 0$$



Chi & Kolda (2012); Hansen, Plantenga, Kolda (2015); Hong, Kolda, Duersch, arXiv, 2018

Poisson Distribution for Count Data (Version 2: Log Link)



PDF for Poisson Distribution

$$p\lambda = e^{-\lambda} \lambda^x / x!$$

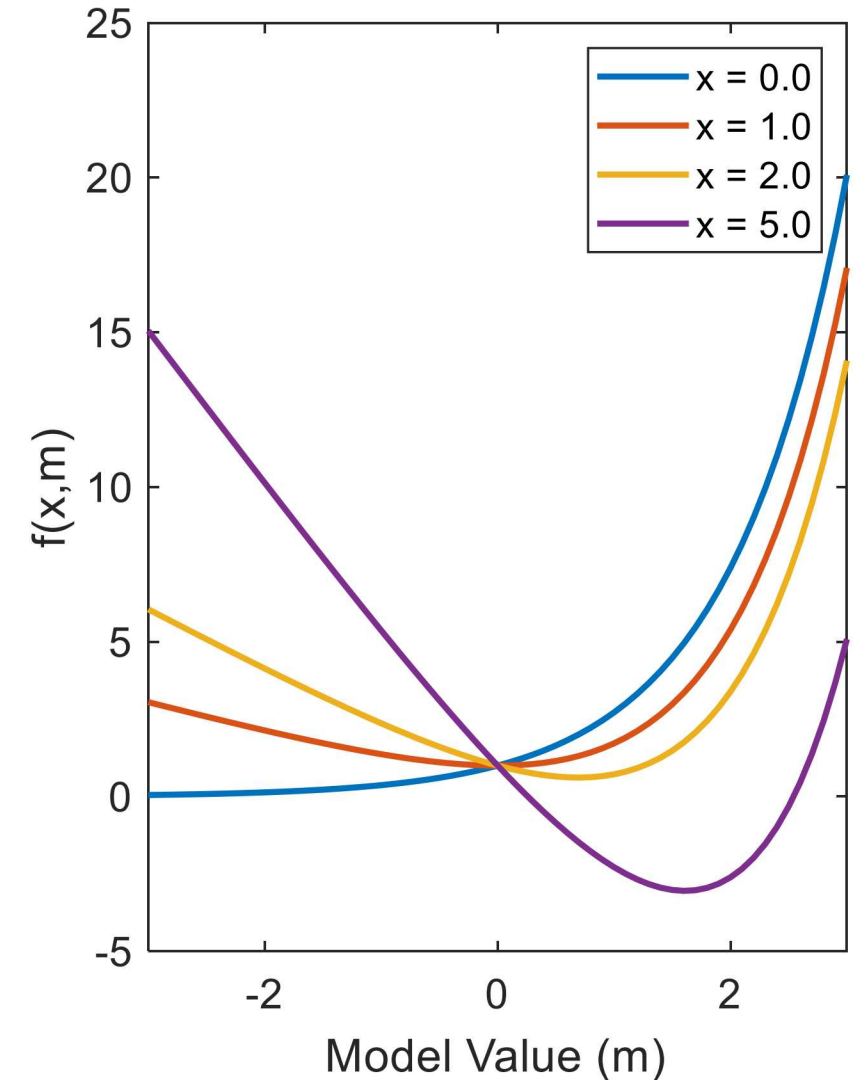
$$x \in \mathbb{N}$$

and

Link Function

$$m = \log(\lambda)$$

$$f(x, m) = e^m - xm \quad \text{for } x \in \mathbb{N}, m \in \mathbb{R}$$



Hong, Kolda, Duersch, arXiv, 2018

Gamma Distribution for Positive Data

PDF for Gamma Distribution

$$p(x | k, \theta) = \left(\frac{x^{k-1}}{\Gamma(k) \theta^k} \right) e^{-x/\theta} \quad x > 0$$

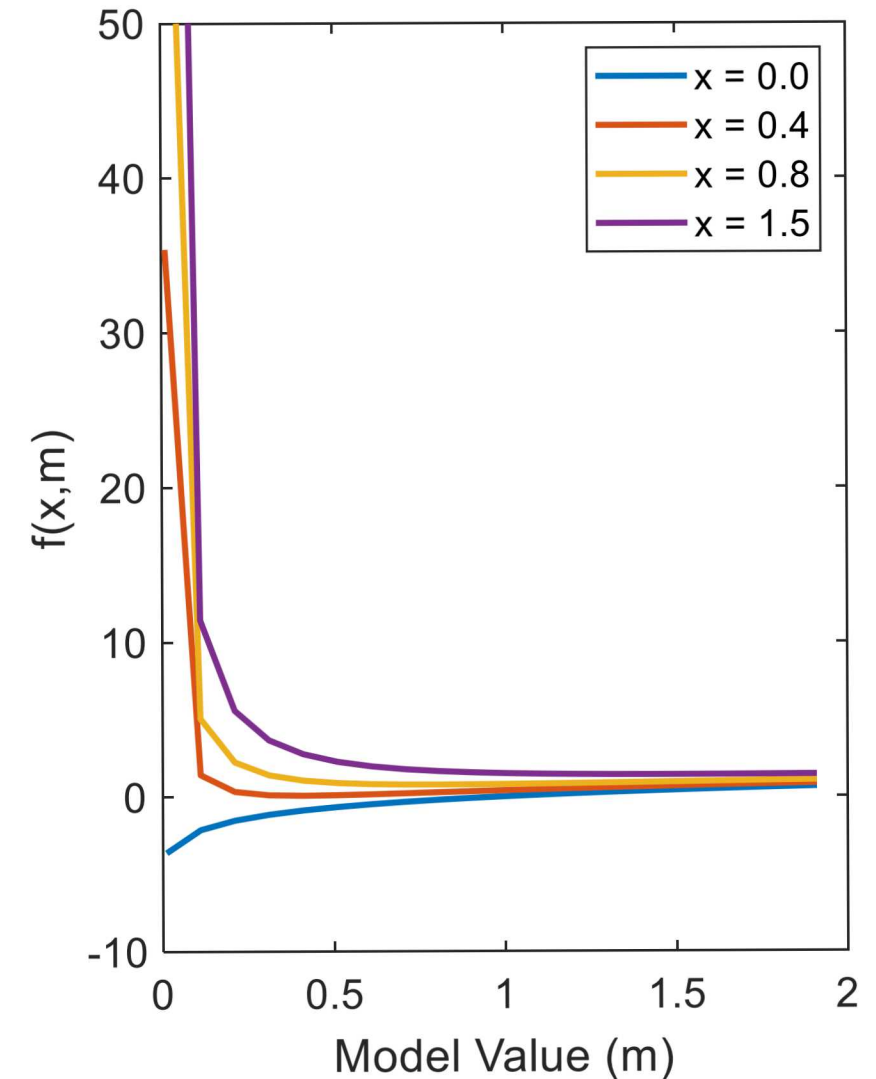
and

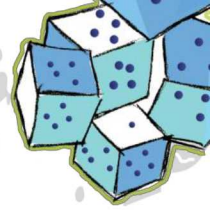
Link Function

$$m = \theta / k$$

k constant

$$f(x, m) = \log(m) + x/m \quad \text{for } x > 0, m > 0$$





Summary of Loss Functions

Distribution	Link function	Loss function	Constraints
$\mathcal{N}(\mu, \sigma)$	$m = \mu$	$(x - m)^2$	$x, m \in \mathbb{R}$
Gamma(k, σ)	$m = k\sigma$	$x / (m + \epsilon) + \log(m + \epsilon)$	$x > 0, m \geq 0$
Poisson(λ)	$m = \lambda$	$m - x \log(m + \epsilon)$	$x \in \mathbb{N}, m \geq 0$
	$m = \log \lambda$	$e^m - xm$	$x \in \mathbb{N}, m \in \mathbb{R}$
Bernoulli(ρ)	$m = \rho / (1 - \rho)$	$\log(m + 1) - x \log(m + \epsilon)$	$x \in \{0, 1\}, m \geq 0$
	$m = \log(\rho / (1 - \rho))$	$\log(1 + e^m) - xm$	$x \in \{0, 1\}, m \in \mathbb{R}$

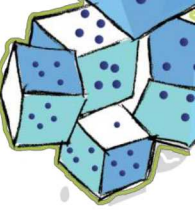
Exponential family: Normal, Gamma with k constant, Poisson with log link, Bernoulli with logit link

Matrix case w/ exponential distributions: Collins, Dasgupta, Schapire, NIPS'02

Hong, Kolda, Duersch, arXiv, 2018

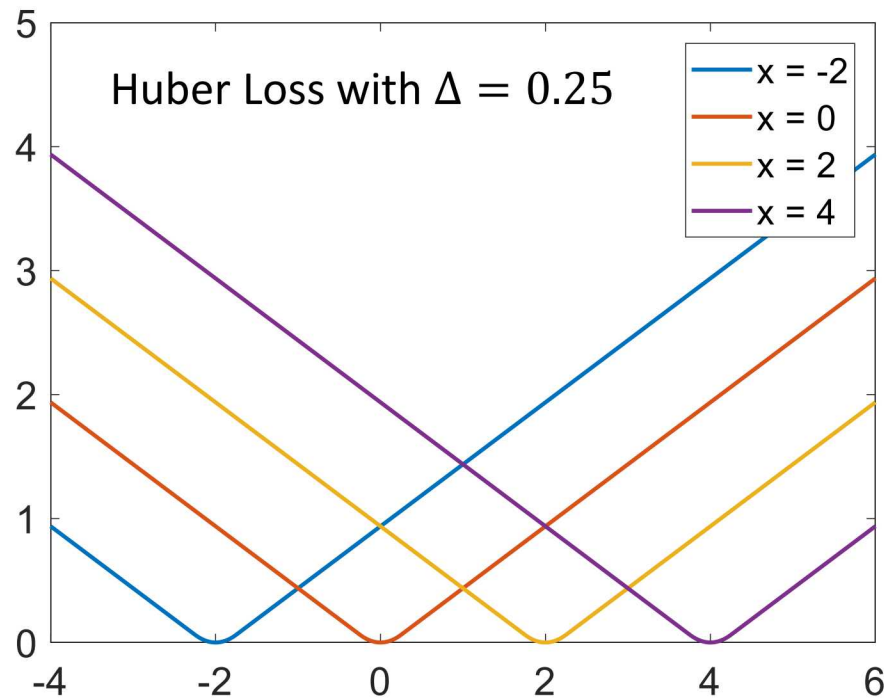


Other Loss Functions



Huber Loss

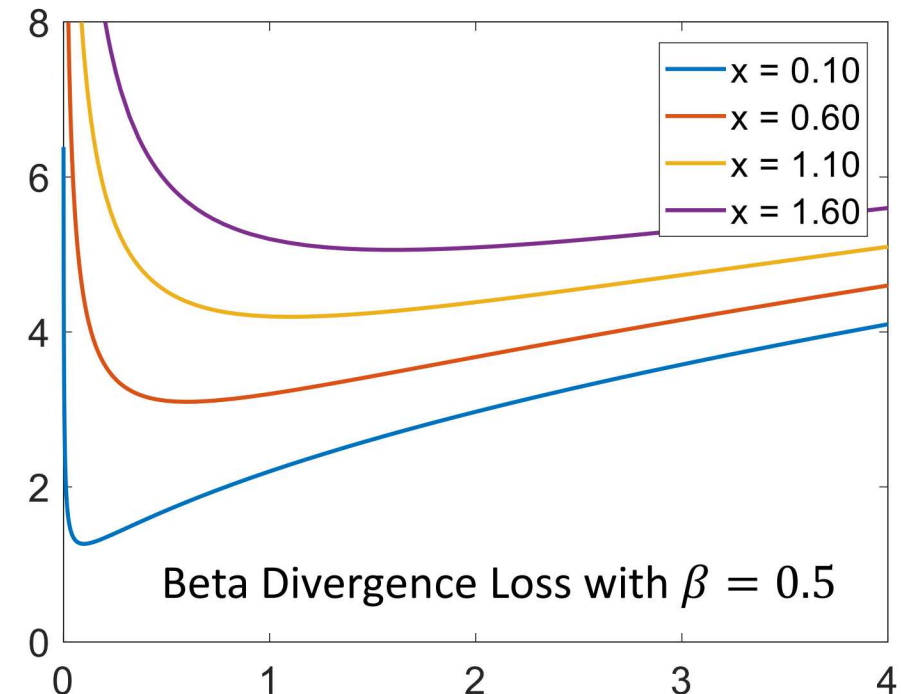
$$f(x, m; \Delta) = \begin{cases} (x - m)^2 & \text{if } |x - m| \leq \Delta \\ 2\Delta|x - m| - \Delta^2 & \text{otherwise} \end{cases}$$

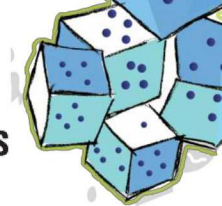


Hong, Kolda, Duersch, arXiv, 2018

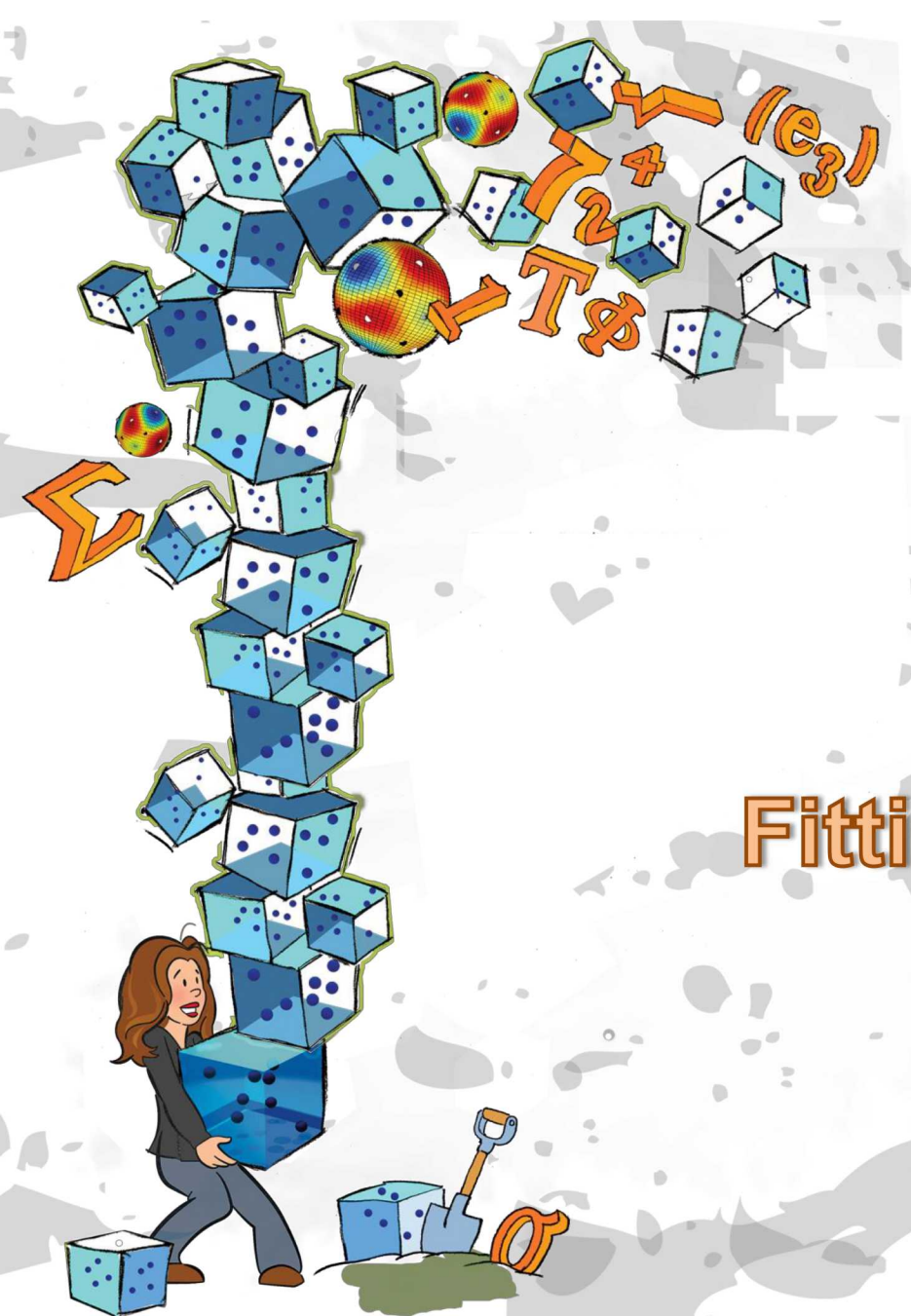
Beta Divergence

$$f(x, m; \beta) = \begin{cases} \frac{1}{\beta}m^\beta - \frac{1}{\beta-1}xm^{\beta-1} & \text{if } \beta \in \mathbb{R} \setminus \{0, 1\} \\ m - x \log m & \text{if } \beta = 1 \quad (\text{Poisson}) \\ \frac{x}{m} + \log m & \text{if } \beta = 0 \quad (\text{Gamma}) \end{cases}$$

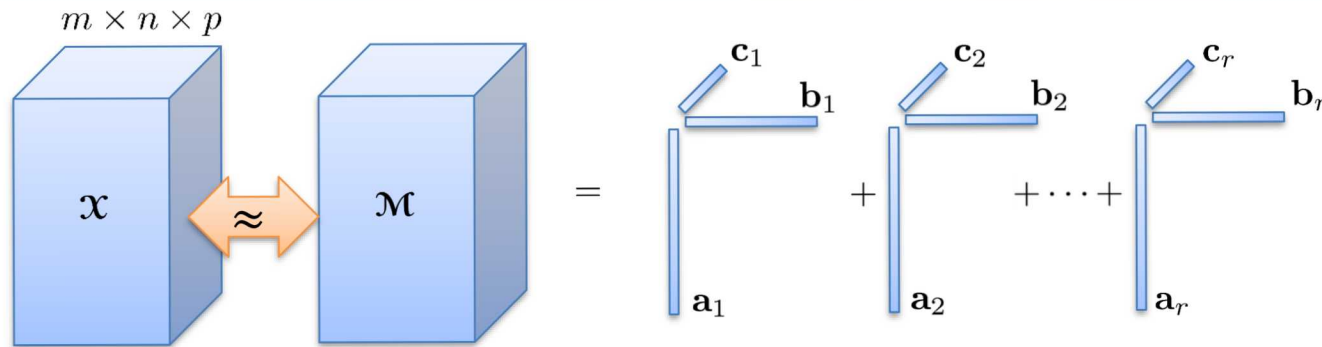




Fitting GCP



Fitting the CP Tensor Decomposition: Direct Optimization Derivation



Direct Optimization

Repeat until convergence...

$$[\mathbf{A}, \mathbf{B}, \mathbf{C}] \leftarrow [\mathbf{A}, \mathbf{B}, \mathbf{C}] - \alpha \left[\frac{\partial F}{\partial \mathbf{A}}, \frac{\partial F}{\partial \mathbf{B}}, \frac{\partial F}{\partial \mathbf{C}} \right]$$

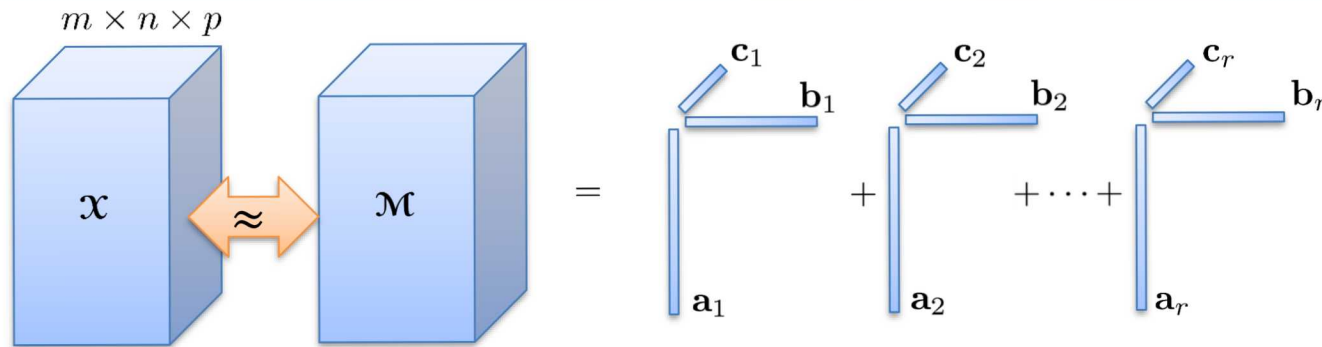
Goal is to minimize: $F(\mathbf{A}, \mathbf{B}, \mathbf{C}) \equiv \|\mathcal{X} - \mathcal{M}\|^2$

Gradients

$$\left\{ \begin{aligned} \frac{\partial F}{\partial \mathbf{A}} &= (\mathbf{X}_{(1)} - \mathbf{M}_{(1)}) (\mathbf{C} \odot \mathbf{B}) \\ \frac{\partial F}{\partial \mathbf{B}} &= (\mathbf{X}_{(2)} - \mathbf{M}_{(2)}) (\mathbf{C} \odot \mathbf{A}) \\ \frac{\partial F}{\partial \mathbf{C}} &= (\mathbf{X}_{(3)} - \mathbf{M}_{(3)}) (\mathbf{B} \odot \mathbf{A}) \end{aligned} \right.$$

Hong, Kolda, Duersch, arXiv, 2018

Fitting the GCP Tensor Decomposition: Direct Optimization Derivation



Direct Optimization

Repeat until convergence...

$$[\mathbf{A}, \mathbf{B}, \mathbf{C}] \leftarrow [\mathbf{A}, \mathbf{B}, \mathbf{C}] - \alpha \left[\frac{\partial F}{\partial \mathbf{A}}, \frac{\partial F}{\partial \mathbf{B}}, \frac{\partial F}{\partial \mathbf{C}} \right]$$

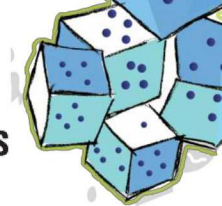
Goal is to minimize: $F(\mathbf{A}, \mathbf{B}, \mathbf{C}) \equiv \sum_{ijk} f(x, m)$

Gradients

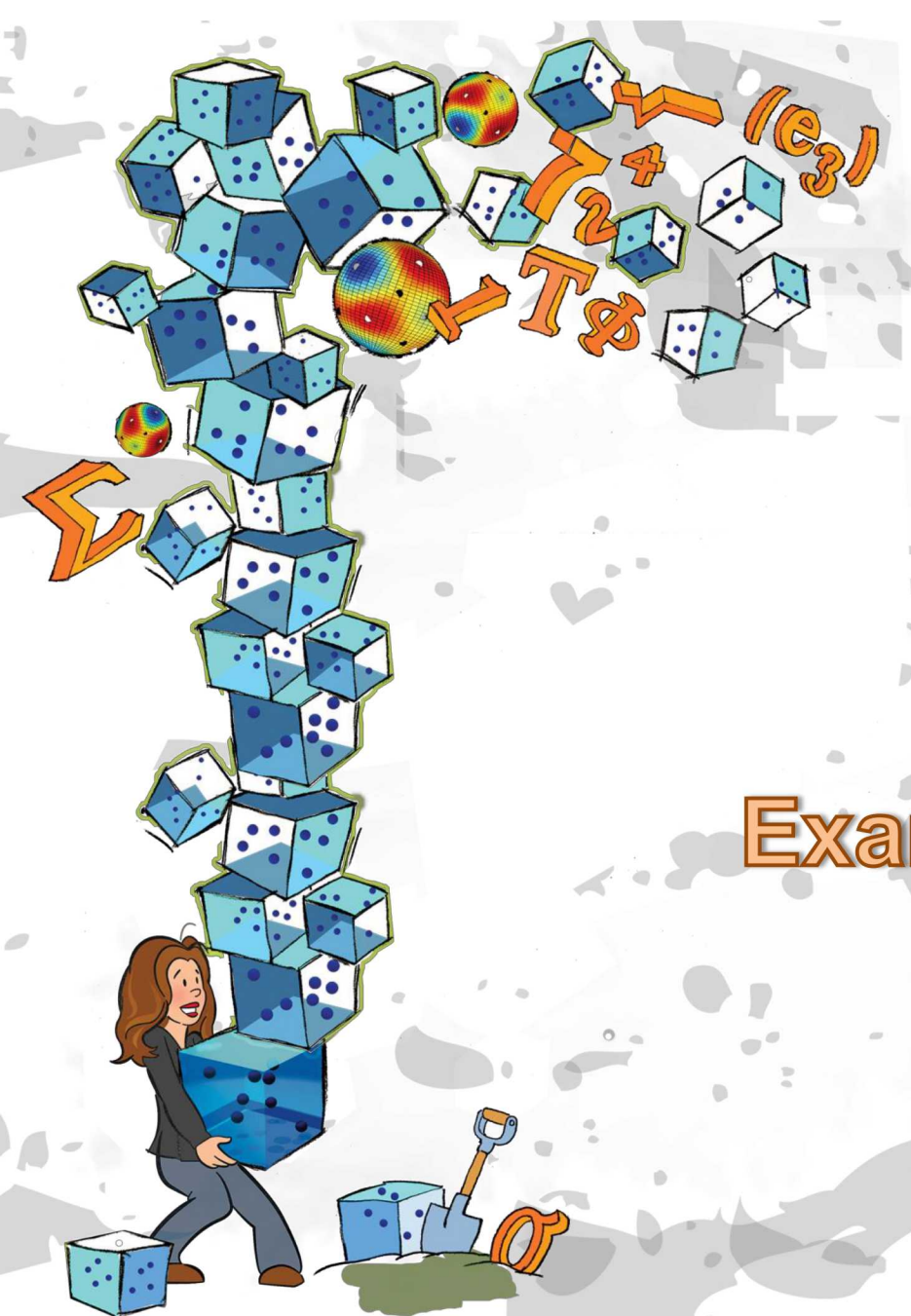
$$\left\{ \begin{aligned} \frac{\partial F}{\partial \mathbf{A}} &= f'(\mathbf{X}_{(1)}, \mathbf{M}_{(1)}) (\mathbf{C} \odot \mathbf{B}) \\ \frac{\partial F}{\partial \mathbf{B}} &= f'(\mathbf{X}_{(2)}, \mathbf{M}_{(2)}) (\mathbf{C} \odot \mathbf{A}) \\ \frac{\partial F}{\partial \mathbf{C}} &= f'(\mathbf{X}_{(3)}, \mathbf{M}_{(3)}) (\mathbf{B} \odot \mathbf{A}) \end{aligned} \right.$$

f' is the derivative of f wrt m , evaluated elementwise

Hong, Kolda, Duersch, arXiv, 2018



Examples of GCP



Social Network – Binary Dataset

■ UC Irvine Chat Network

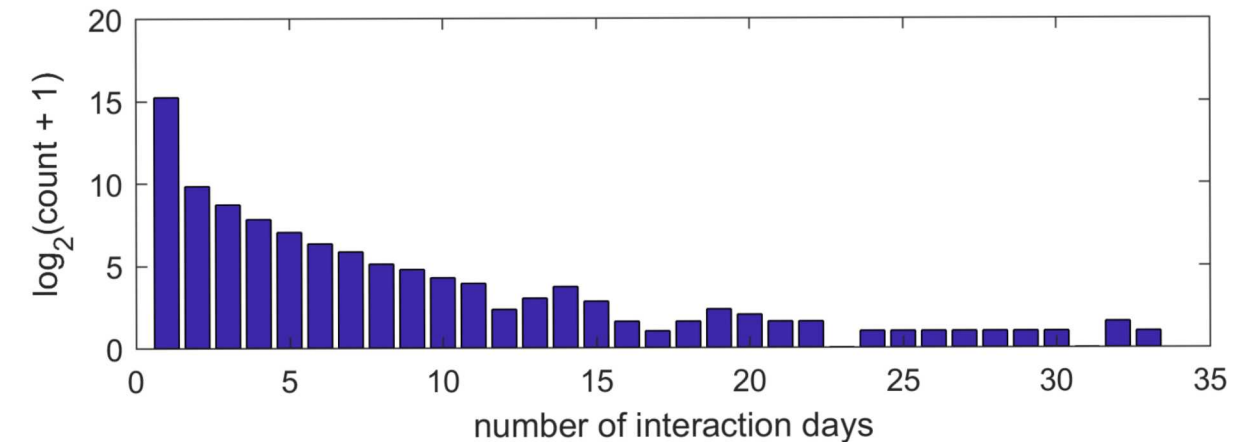
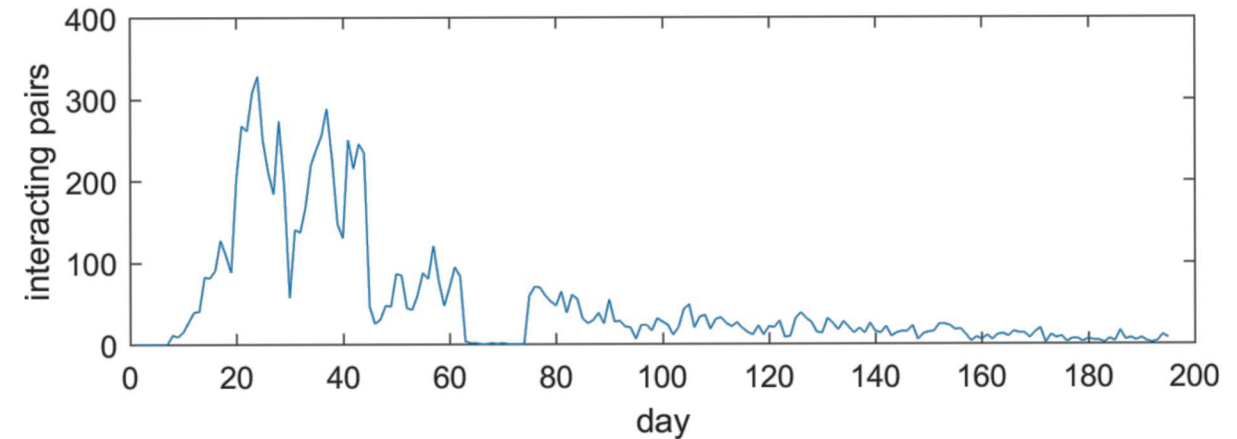
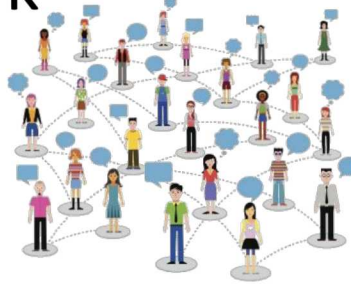
■ 3-way binary tensor

■ Sender (200)

■ Receiver (200)

■ Day (195)

■ 9,764 nonzeros (very sparse)

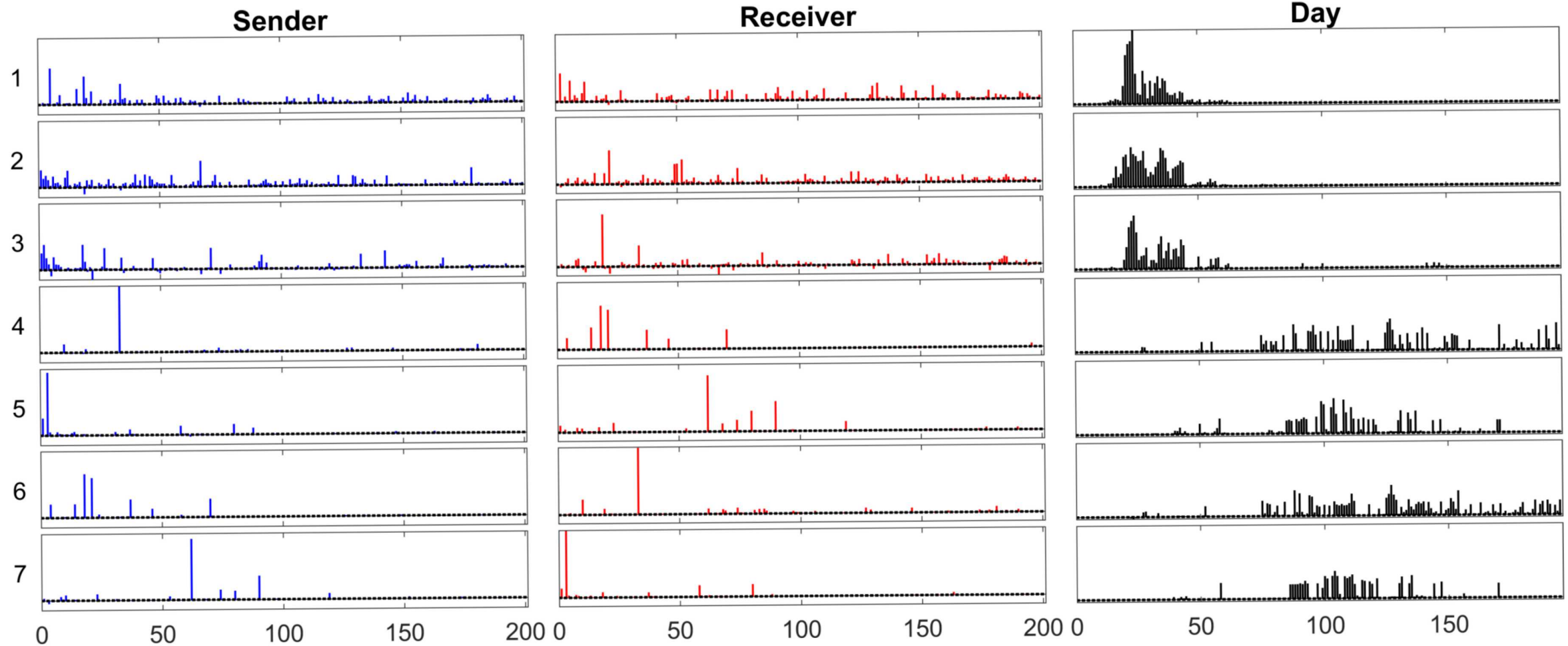


■ Use GCP to compute rank-7 decomposition

Hong, Kolda, Duersch, arXiv, 2018

Standard CP on Social Network

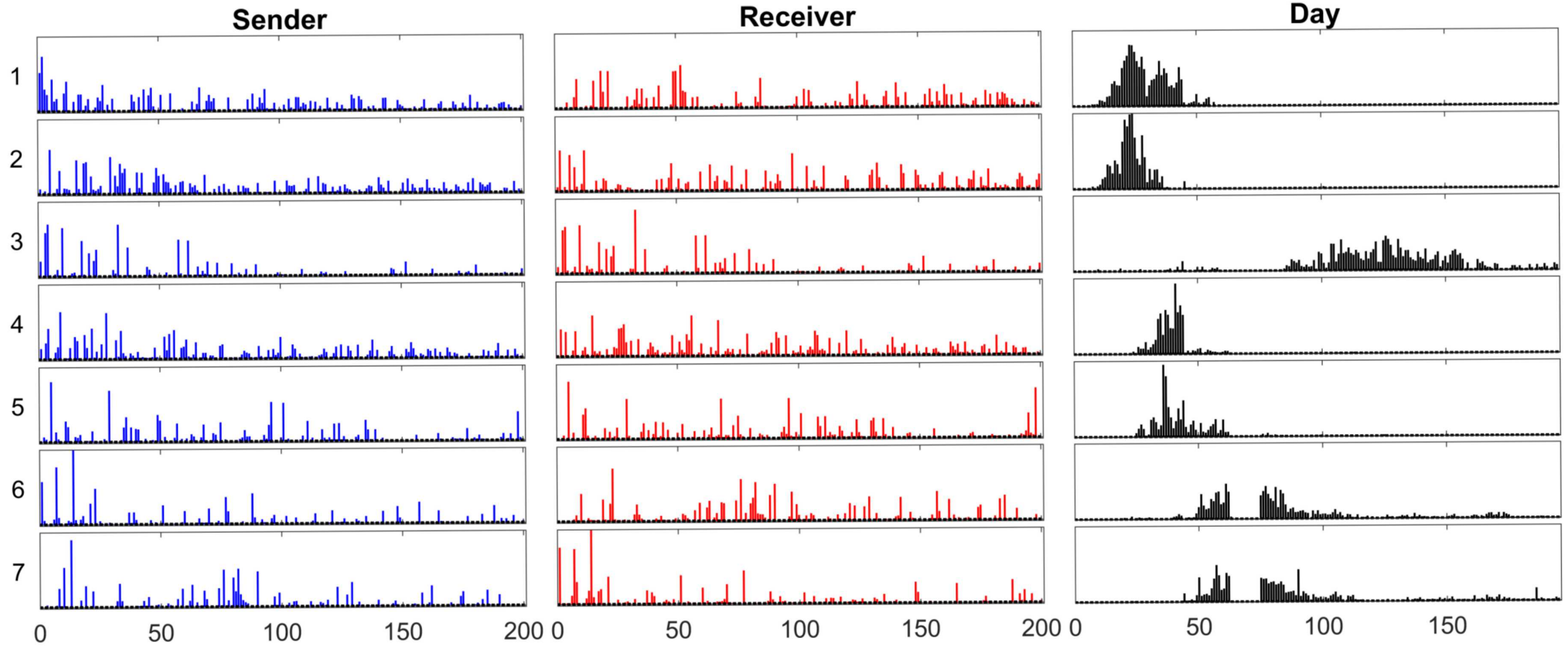
$$f(x, m) = (x - m)^2$$



Hong, Kolda, Duersch, arXiv, 2018

Bernoulli Odds on Social Network

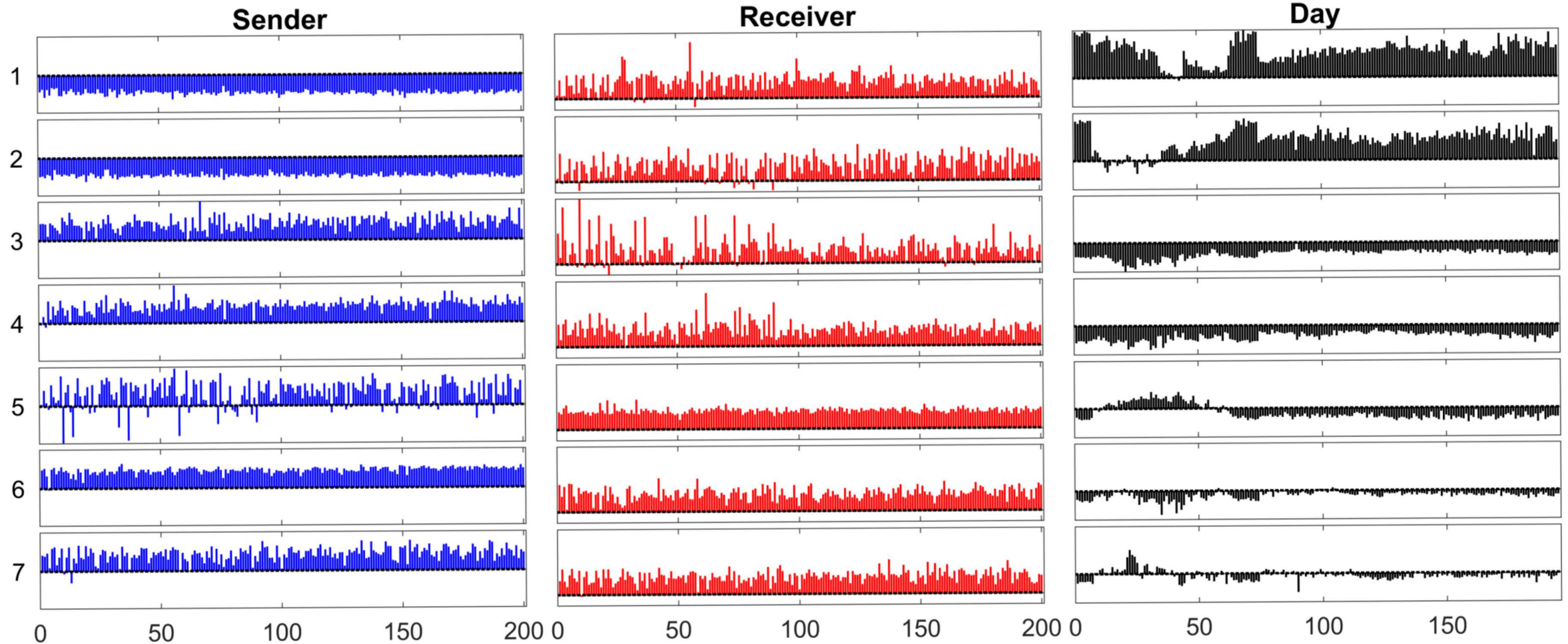
$$f(x, m) = \log(1 + m) - x \log m$$



Hong, Kolda, Duersch, arXiv, 2018

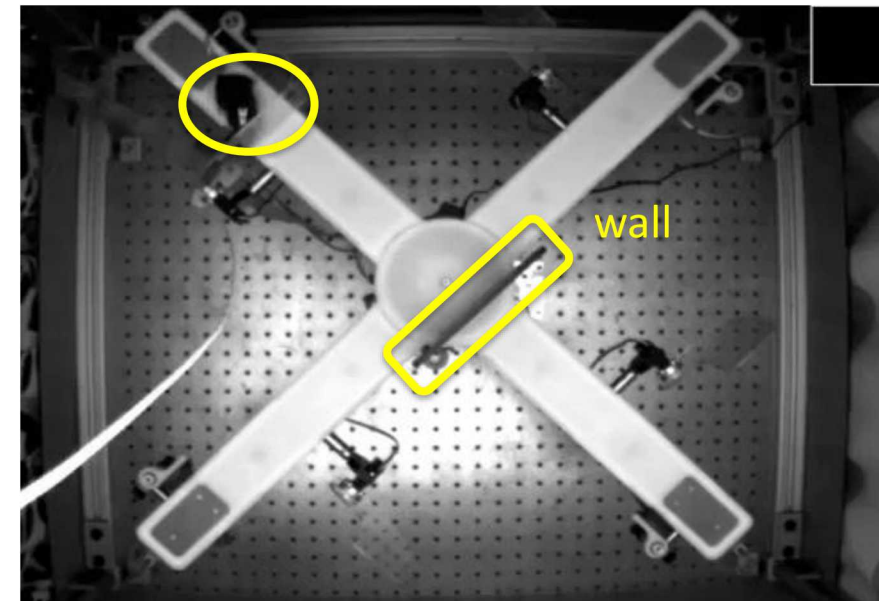
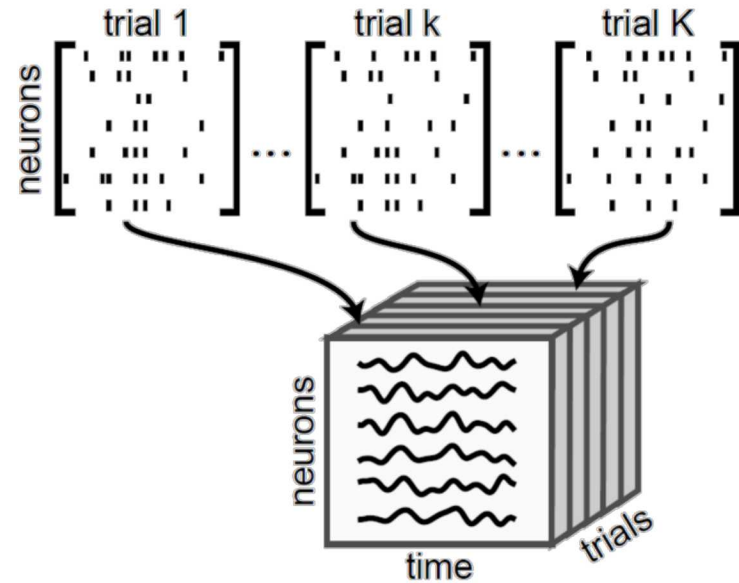
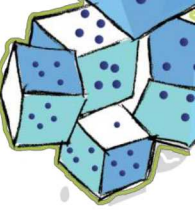
Bernoulli Log-odds on Social Network

$$f(x, m) = \log(1 + e^m) - xm$$



Hong, Kolda, Duersch, arXiv, 2018

Trial Conditions Change Twice, so Mouse Must Adapt Strategy to Receive Reward

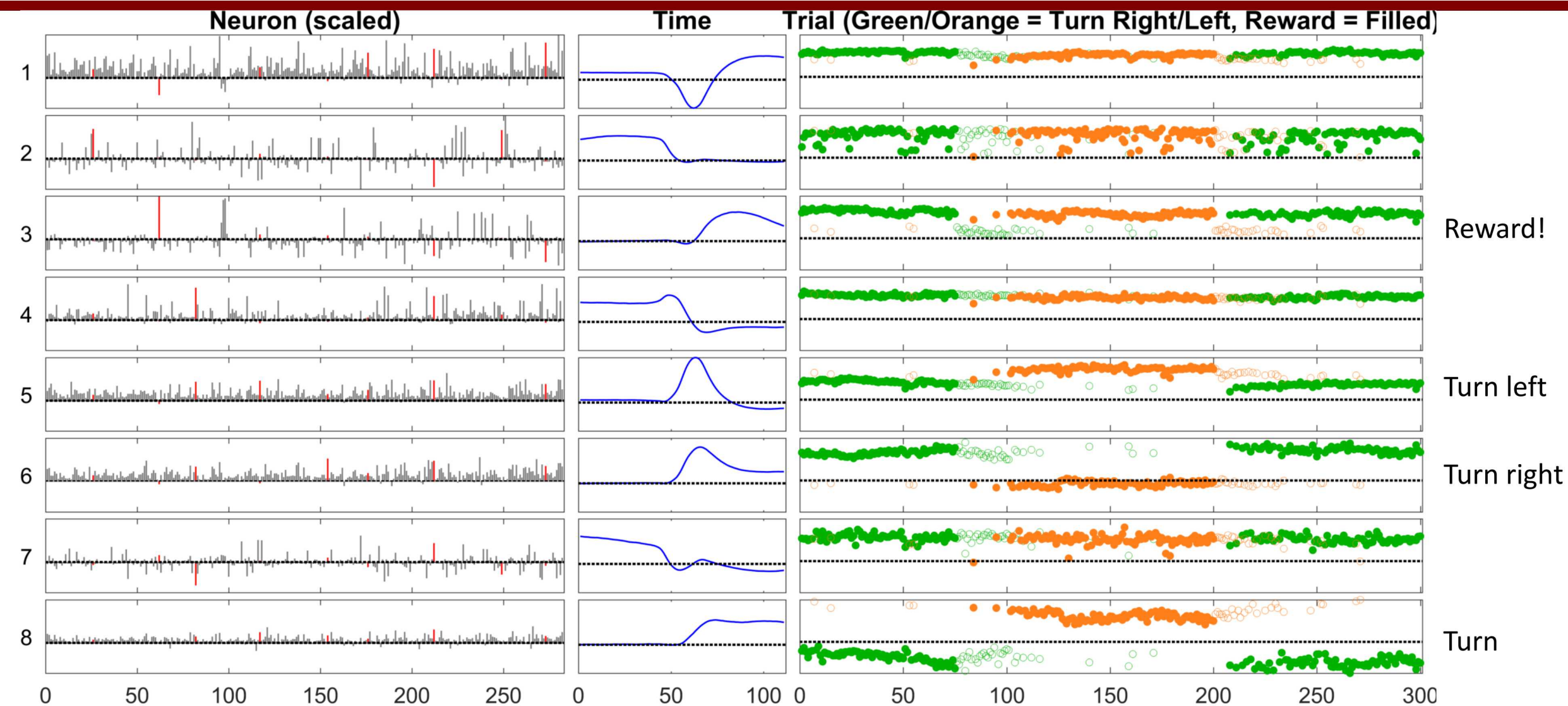
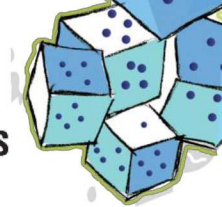


- *Simplified version of dataset from Neuron paper*
- 300 Trials over 5 Days
- Conditions Swap Twice
 - ❖ Turn right
 - ❖ Turn left
 - ❖ Turn right

Williams et al., Neuron 2018; Hong, Kolda, Duersch, 2018

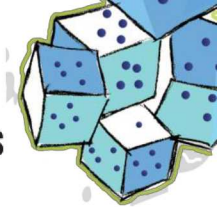
Standard CP Solution

$$f(x, m) = (x - m)^2$$



GCP with Huber Loss

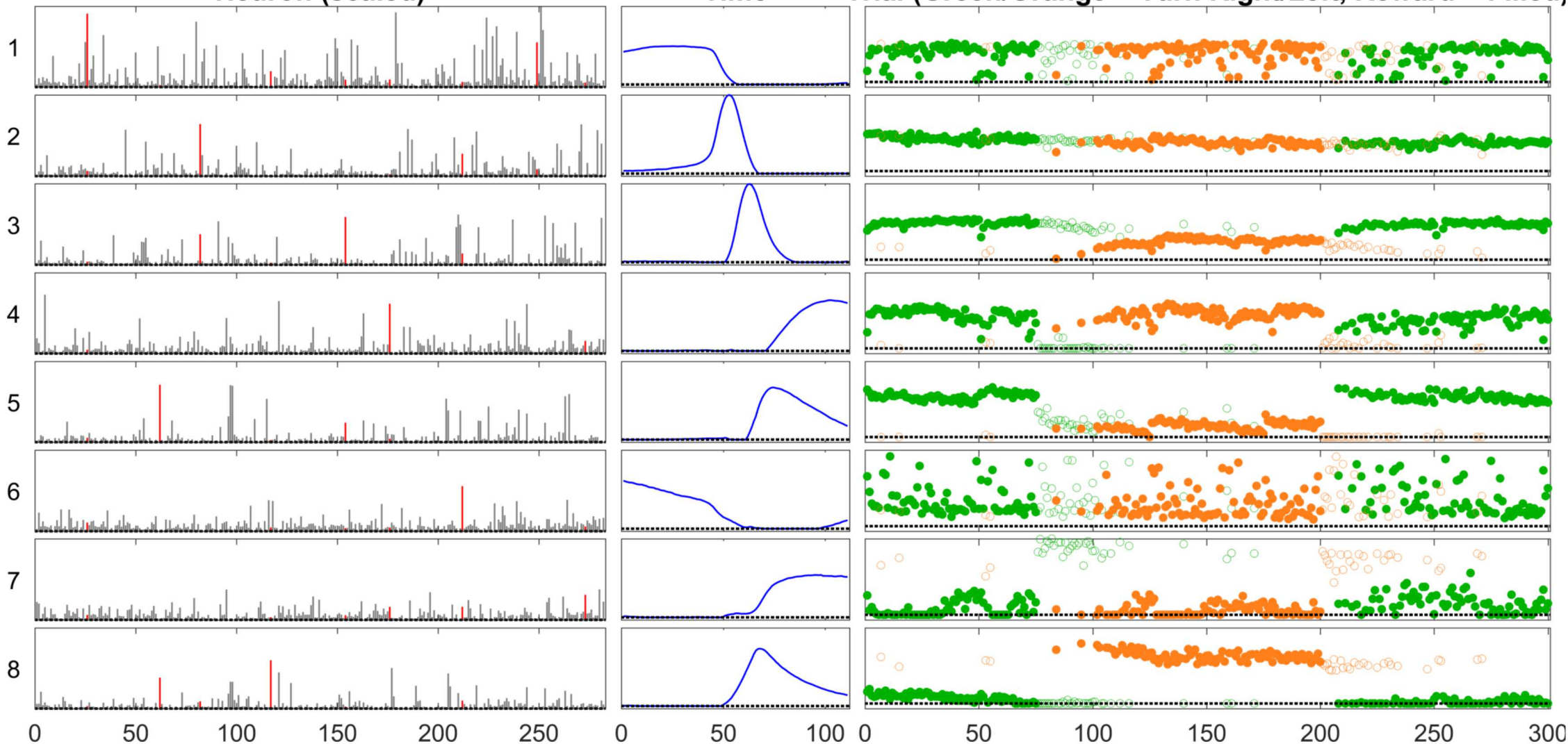
$$f(x, m; \delta = \frac{1}{4}) = \begin{cases} \frac{1}{2}(x - m)^2 & \text{if } |x - m| < \delta \\ \delta(|x - m| - \frac{1}{2}\delta) & \text{otherwise} \end{cases}$$

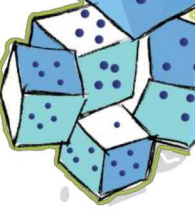


Neuron (scaled)

Time

Trial (Green/Orange = Turn Right/Left, Reward = Filled)





India Rainfall Data

- Kaggle/India Open Data Sets
- 115 years of data (1901-2015)
- 36 regions
- 12 months
- <1% missing data

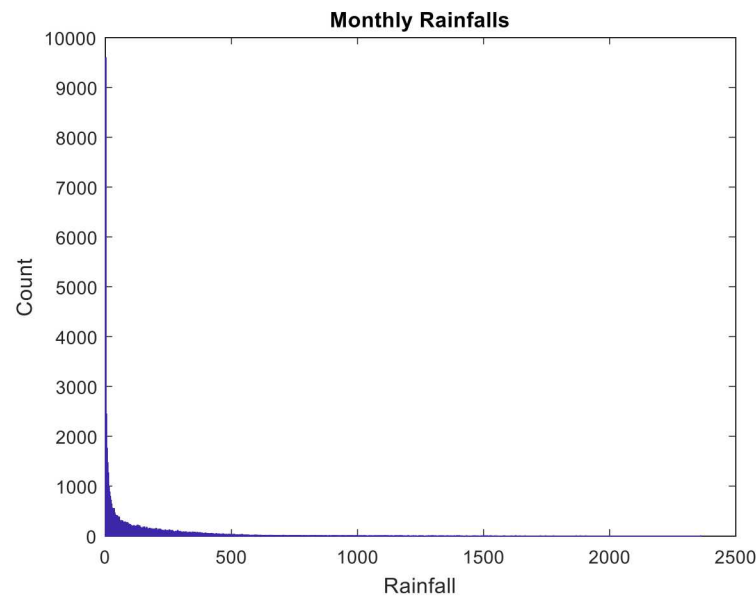
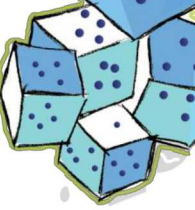
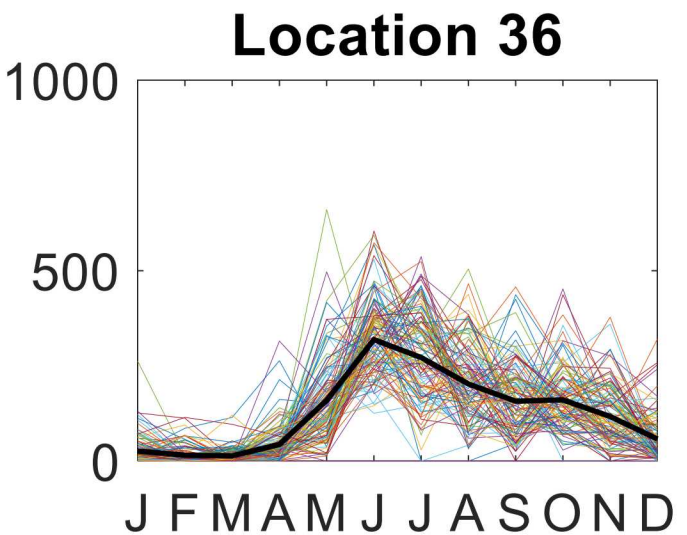
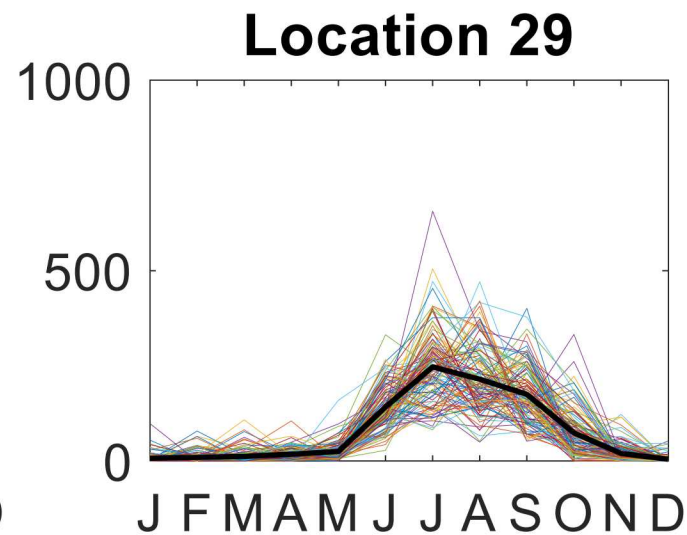
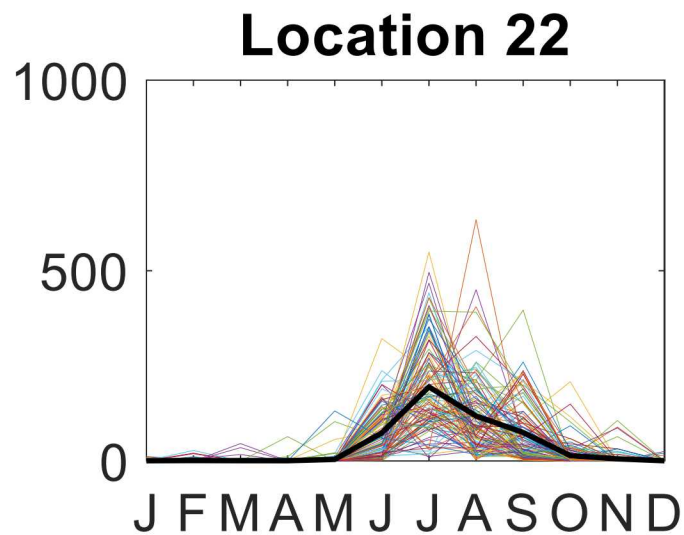
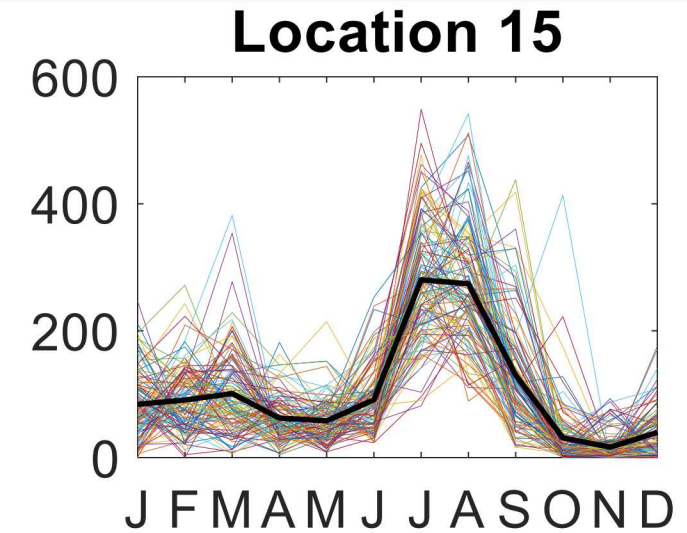
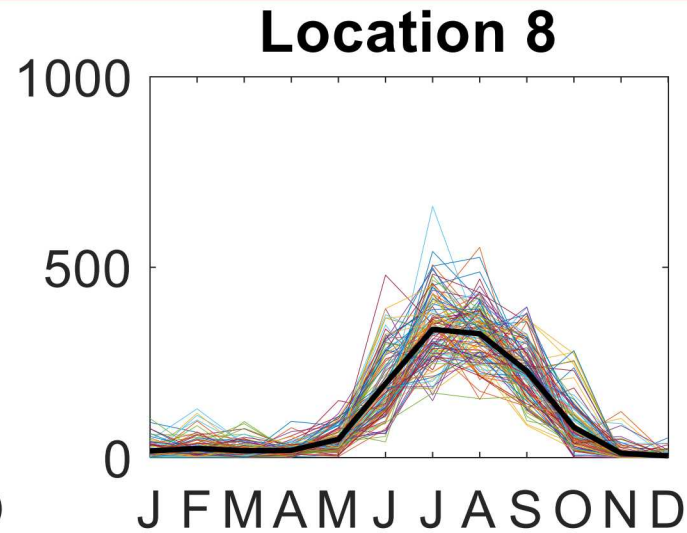
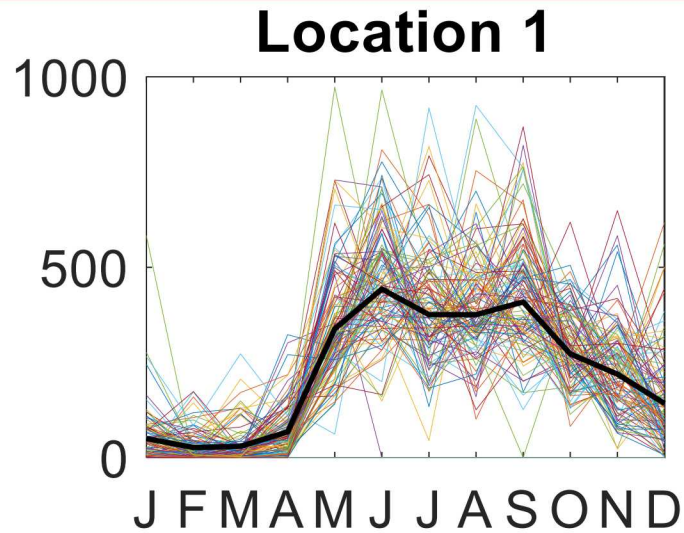


Image source: Kumar, Jain & Singh. (2010). Analysis of long-term rainfall trends in India. Hydrological Sciences Journal-journal Des Sciences Hydrologiques 55. 484-496. 10.1080/02626667.2010.481373.

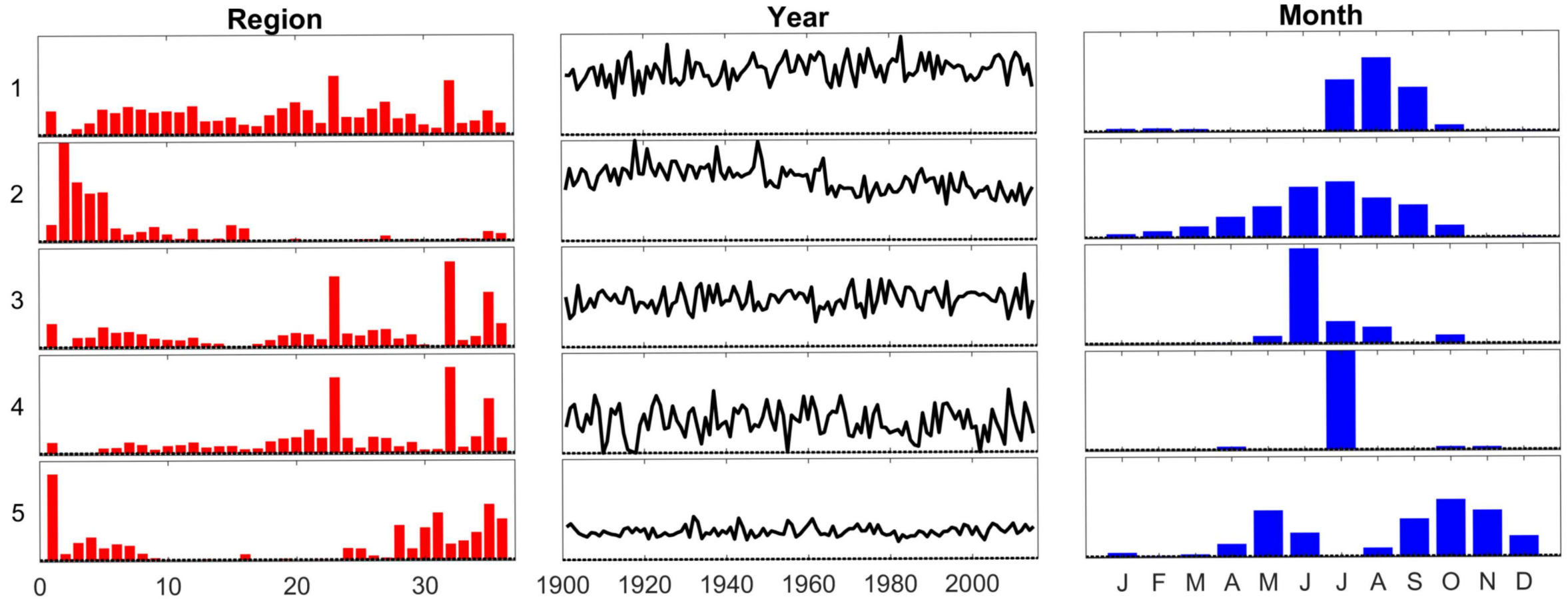


Rain Patterns at Sample Locations



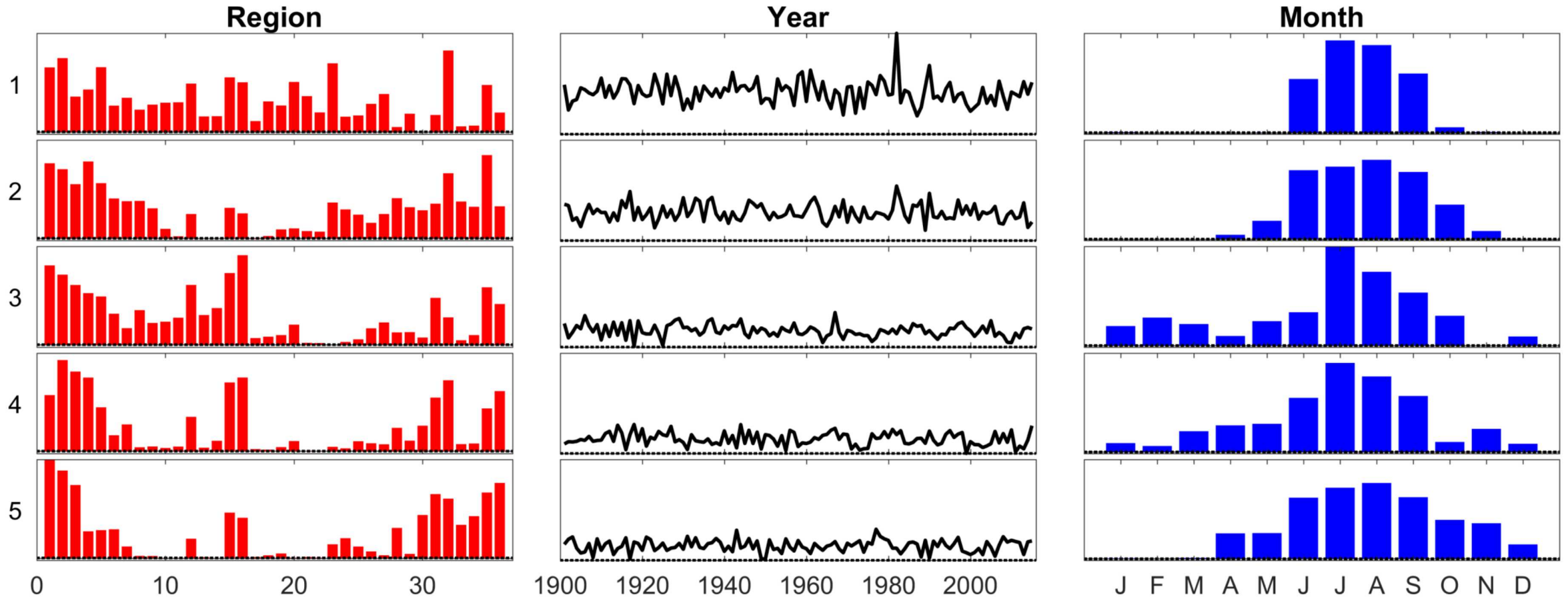
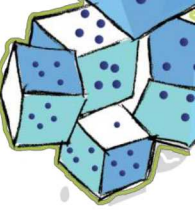
Rainfall with Standard CP

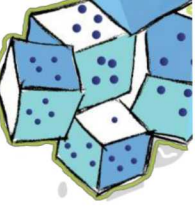
$$f(x, m) = (x - m)^2$$



GCP with Gamma Loss

$$f(x, m) = \log m + x/m$$





Brief Message from the Society for Industrial and Applied Mathematics (SIAM) (and me)

Submit your work at *simods.siam.org*

SIAM JOURNAL ON Mathematics of Data Science

SIAM Journal on Mathematics of Data Science (SIMODS) publishes work that advances mathematical, statistical, and computational methods in the realm of data and information sciences.

We invite papers that present significant advances in this context, including applications to science, engineering, business, and medicine.

EDITOR-IN-CHIEF

Tamara G. Kolda, *Sandia National Laboratories*

EDITORIAL BOARD 2018

Section Editors

Alfred Hero
Michael Jordan
Robert D. Nowak
Joel A. Tropp

Monique Laurent
Elizavita Levina
Yi Ma
Michael Mahoney

Associate Editors

Maria Florina Balcan
Rina Foygel Barber
Robert Calderbank
Venkat Chandrasekaran
Jennifer Chayes
Alexandre d'Aspremont
Ioana Dumitriu
Maryam Fazel
Emily Fox
Mark Girolami
David F. Gleich
Ashish Goel
Ilse Ipsen
Eric D. Kolaczyk
Gitta Kutyniok

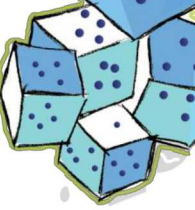
Long Nguyen
Ivan Oseledets
Natesh Pillai
Ali Pinar
Mason Porter
Bala Rajaratnam
Phillipe Rigollet
Justin Romberg
Ronitt Rubinfeld
Richard Samworth
Katya Scheinberg
Amit Singer
Marc Teboulle
Ramon van Handel
Weichung Wang
Rachel Ward
Rebecca Willet

journals.siam.org/simods

siam

Society for Industrial and Applied Mathematics

SIMODS@SIAM.ORG / 800-447-7426 (USA & Canada) / 1-215-382-9800 (Worldwide)



Conclusions

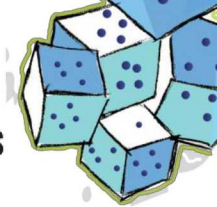
- CP Tensor Decomposition
 - Powerful tool for data analysis
 - Higher-order analogue of PCA, SVD, NMF, etc.
 - Fit the model using numerical optimization
- GCP Tensor Decomposition
 - Extends CP by allowing a different fit function
 - Motivated by need to fit binary, count, nonnegative data
 - Framework allows any continuously differentiable choice of function and invertible link function
 - Ongoing work with stochastic gradient descent (SGD) to extend to large-scale and sparse tensors
- Applications of CP/GCP Tensor Decomposition
 - Neuroscience
 - Social Science
 - Chemistry
 - Text Analysis
 - Signal Processing
 - And many more!



NC State
RED Talks
*Where Data
Meets Science*

October 29, 2018

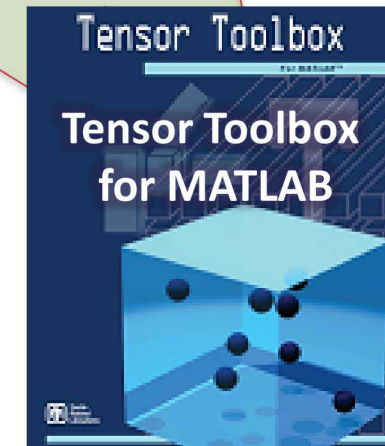




References and Collaborators

- **D. Hong**, T. G. Kolda, J. A. Duersch. **Generalized Canonical Polyadic Tensor Decomposition**. arXiv, 2018. <http://arxiv.org/abs/1808.07452>
- **A. H. Williams**, T. H. Kim, F. Wang, S. Vyas, S. I. Ryu, K. V. Shenoy, M. Schnitzer, T. G. Kolda, S. Ganguli. **Unsupervised Discovery of Demixed, Low-dimensional Neural Dynamics across Multiple Timescales through Tensor Components Analysis**. *Neuron*, 98(6):1099-1115, 2018. <https://doi.org/10.1016/j.neuron.2018.05.015>
- Older relevant work...
 - E. C. Chi, T. G. Kolda. **On Tensors, Sparsity, and Nonnegative Factorizations**. *SIAM Journal on Matrix Analysis and Applications*, 33(4):1272-1299, 2013. <https://doi.org/10.1137/110859063>
 - E. Acar, D. M. Dunlavy, T. G. Kolda. **A Scalable Optimization Approach for Fitting Canonical Tensor Decompositions**. *Journal of Chemometrics* 25(2):67-86, 2011. <https://doi.org/10.1002/cem.1335>
 - T. G. Kolda, B. W. Bader. **Tensor Decompositions and Applications**. *SIAM Review* 51(3):455-500, 2009. <https://doi.org/10.1137/07070111X>
- Plus thanks to other collaborators including Cliff Anderson-Bergman, Casey Battaglino, Grey Ballard, Brett Larsen, Sam Sherman, Kina Winoto, and many more

Thanks for your attention! Email me at tgkolda@sandia.gov with questions!



www.tensortoolbox.org