A Practical Randomized CP Tensor Decomposition

Casey Battaglino[1], Grey Ballard[2], and Tamara G. Kolda[3]

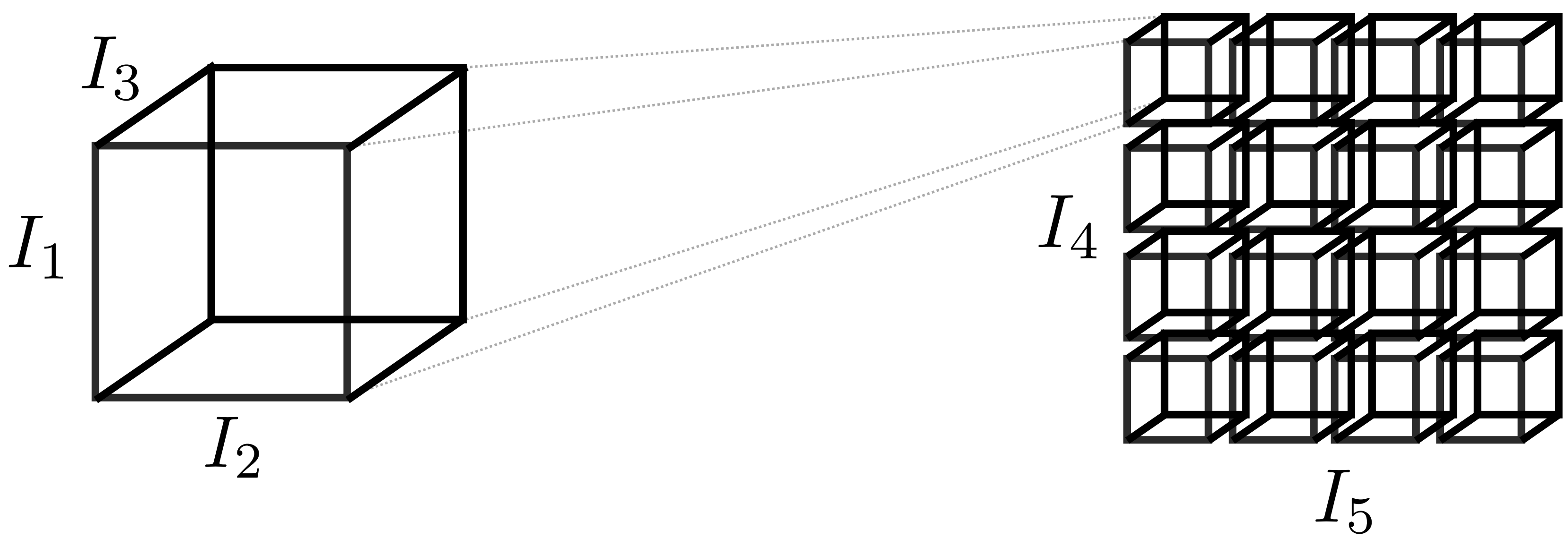
SIAM AN 2017, Pittsburgh, PA

[1]Georgia Tech
Computational Sci. and Engr.
cbattaglino3@gatech.edu

[2]Wake Forest
University
ballard@wfu.edu

[3]Sandia National Labs
tgkolda@sandia.gov

Terminology



$N = 3$
(3 *modes*)

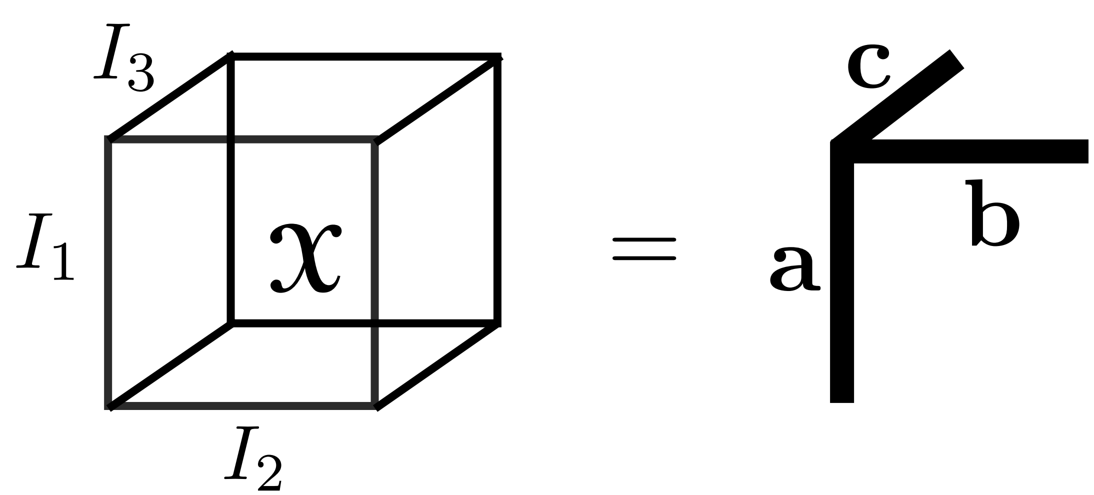
$N = 5$

$N \ldots \text{order}$

$I_n \ldots \text{size of mode } n$

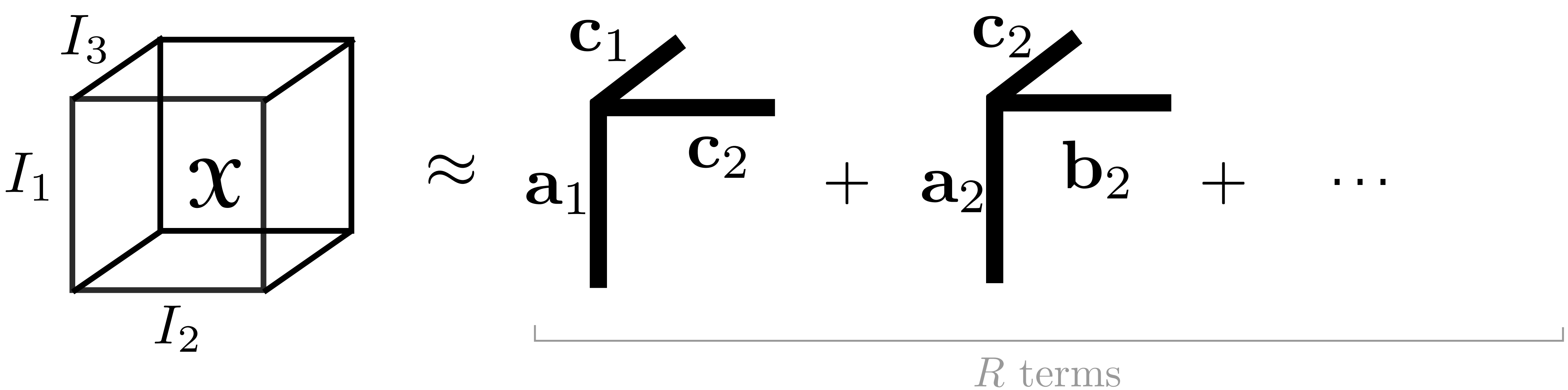
assume *dense* tensors in this work

Rank-*1* Tensor



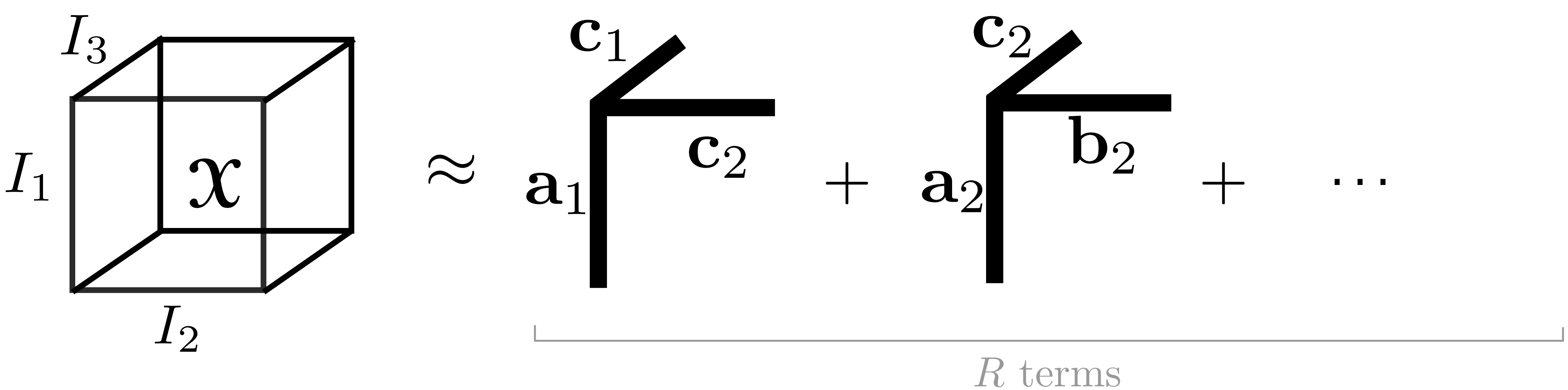
$$\mathcal{X} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$$

CP Decomposition



$$\mathcal{X} \approx \tilde{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

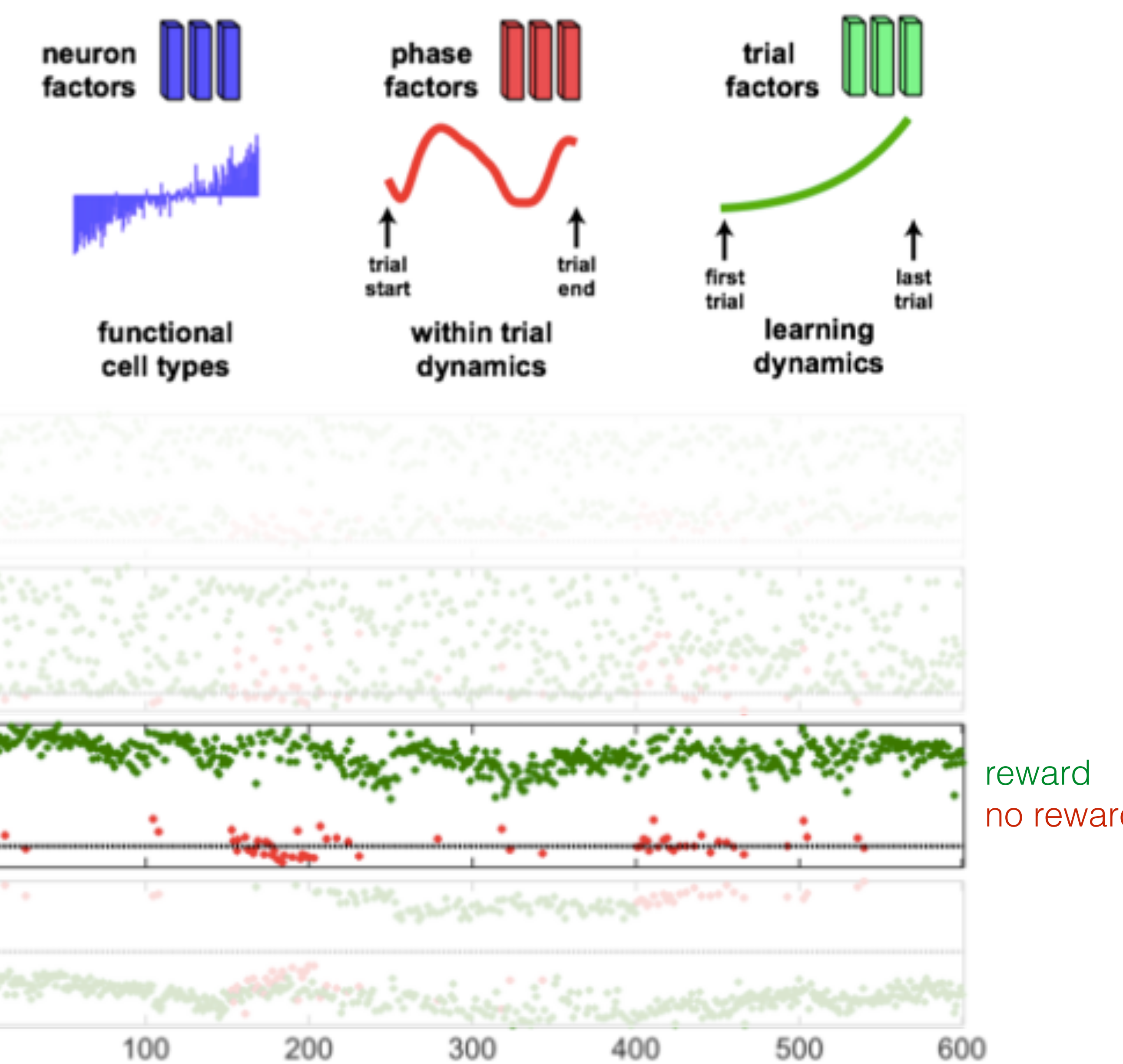
CP Decomposition



CP is valuable for its *intepretability*

CP Decomposition: Example

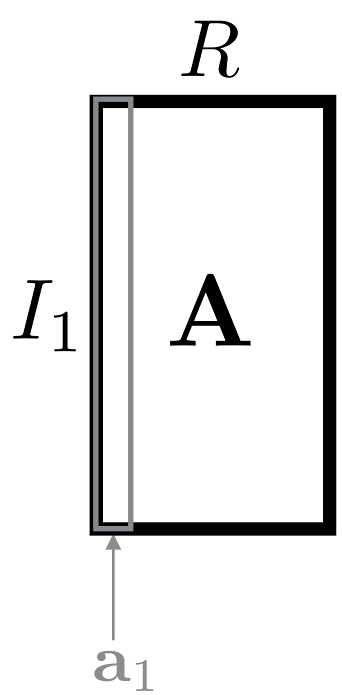
example:
neurons of mouse in maze,
over time,
over many trials.



Source: Williams, Wang, Kim, Schnitzer, Ganguli, Kolda, 2016 (Thx to Williams + Schnitzer grp @ Stanford)

CP-ALS uses Least Squares

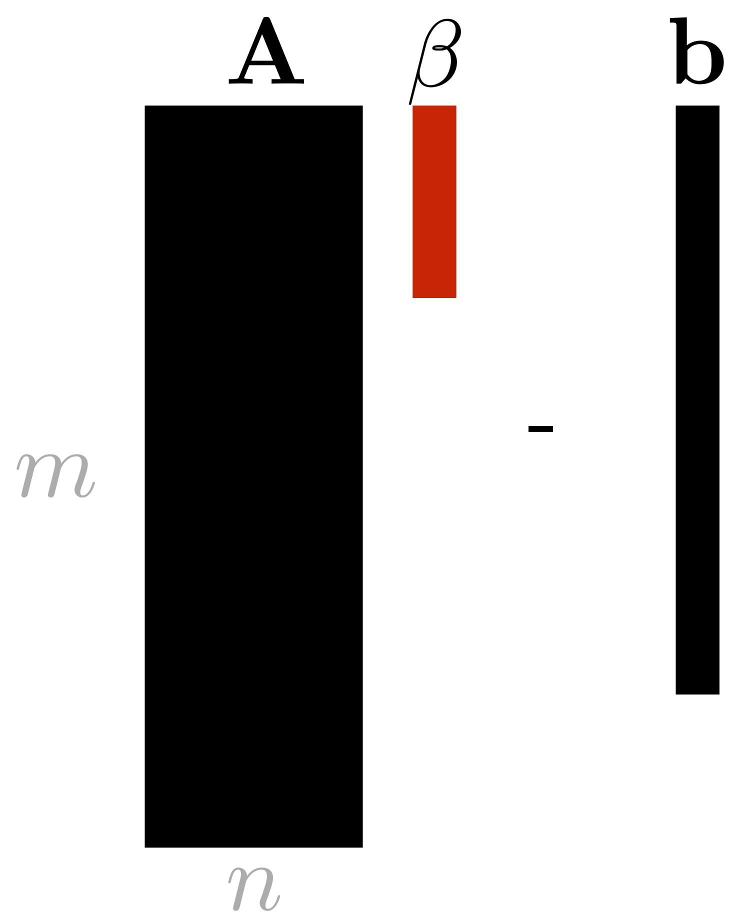
$$\mathcal{X} \approx \tilde{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$



$$\min_{\mathbf{A}} \sum_{ijk} \Big(x_{ijk} - \sum_{r} a_{ir} b_{jr} c_{kr} \Big)^2$$

We will explore the use of *randomized* least squares!

Least Squares



$$\min_{\beta} \|\mathbf{A}\beta - \mathbf{b}\|$$

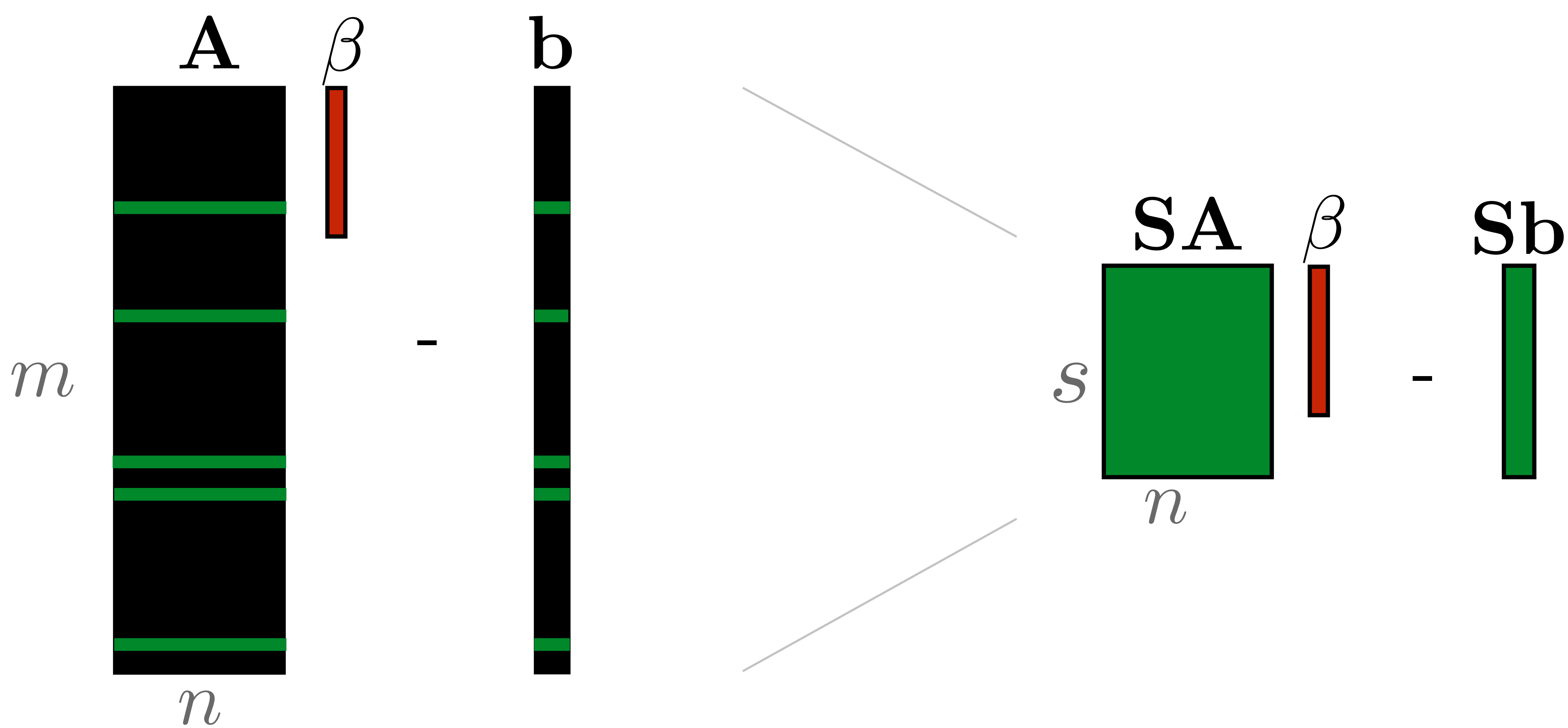
$$\beta = \mathbf{A}^{\dagger}\mathbf{b}$$

$\beta \leftarrow \mathbf{A}\backslash\mathbf{b}$ e.g. MATLAB solver (cholesky, qr, …)

Sampled Least Squares

Choose *s* rows, uniformly at random.

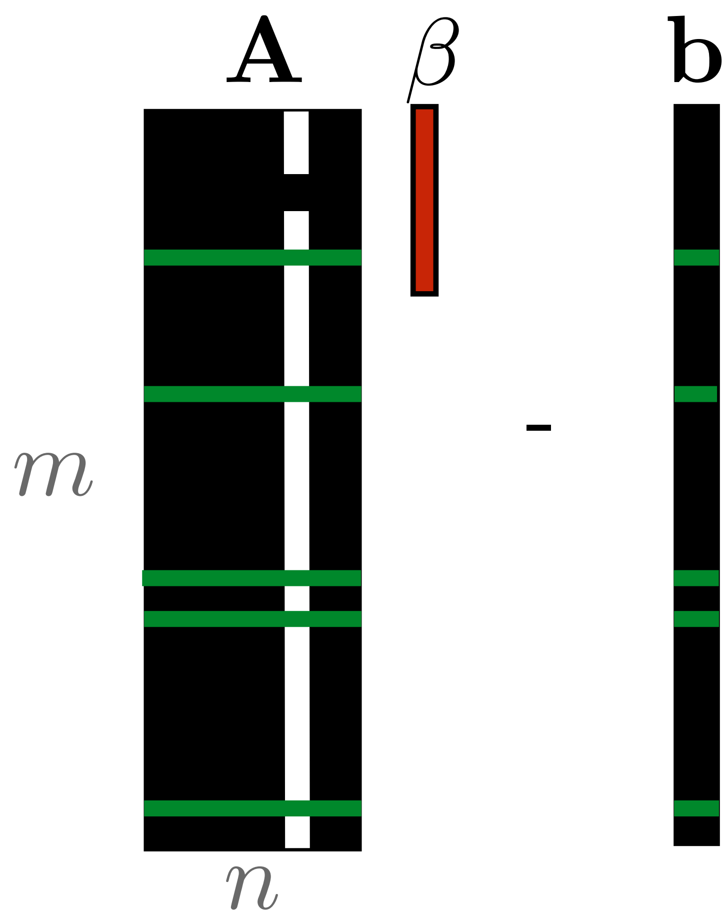
exact **approximate**

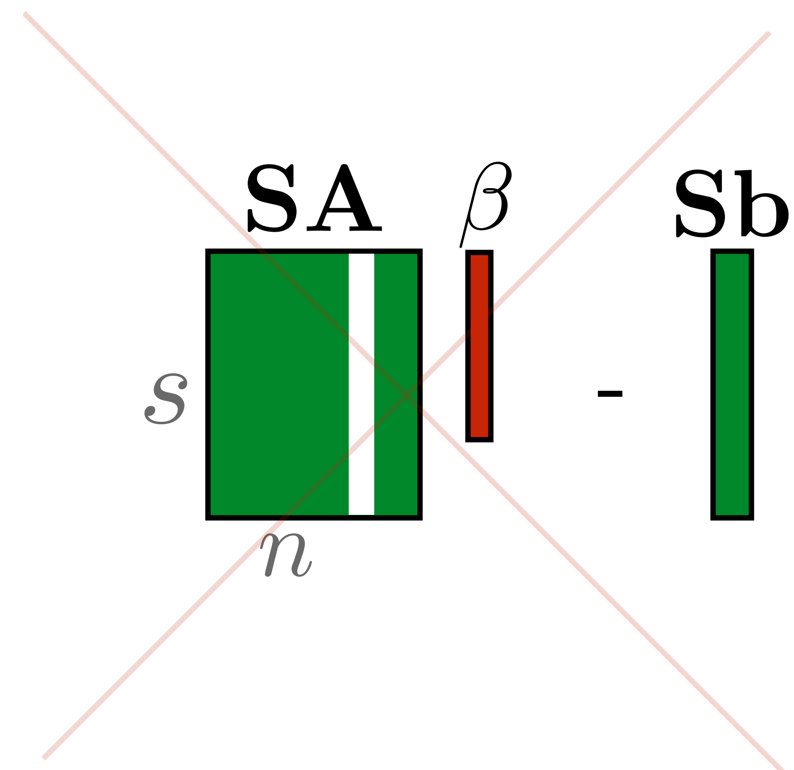


$\mathcal{O}(mn^2)$ $\mathcal{O}(sn^2)$

Sampled Least Squares

Risk[1]:

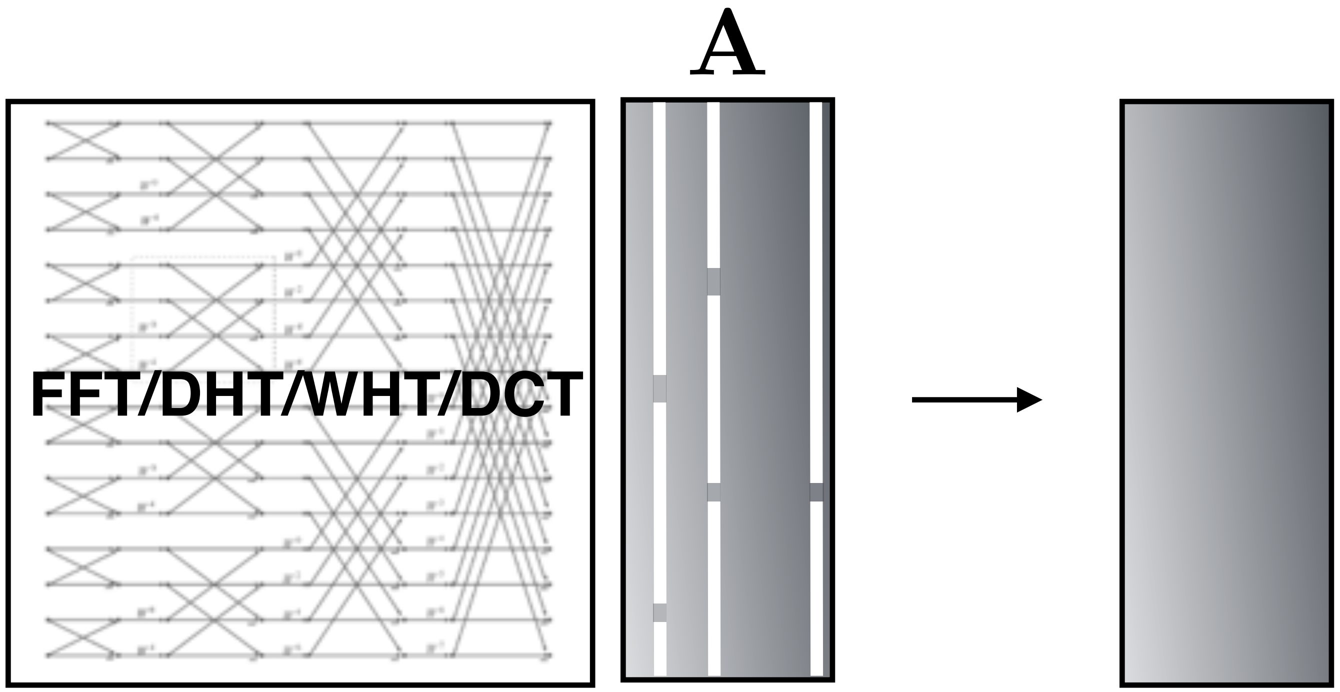




Key idea: change basis so that rows are “mixed” up

[1]: (more generally: any row with a high leverage score)

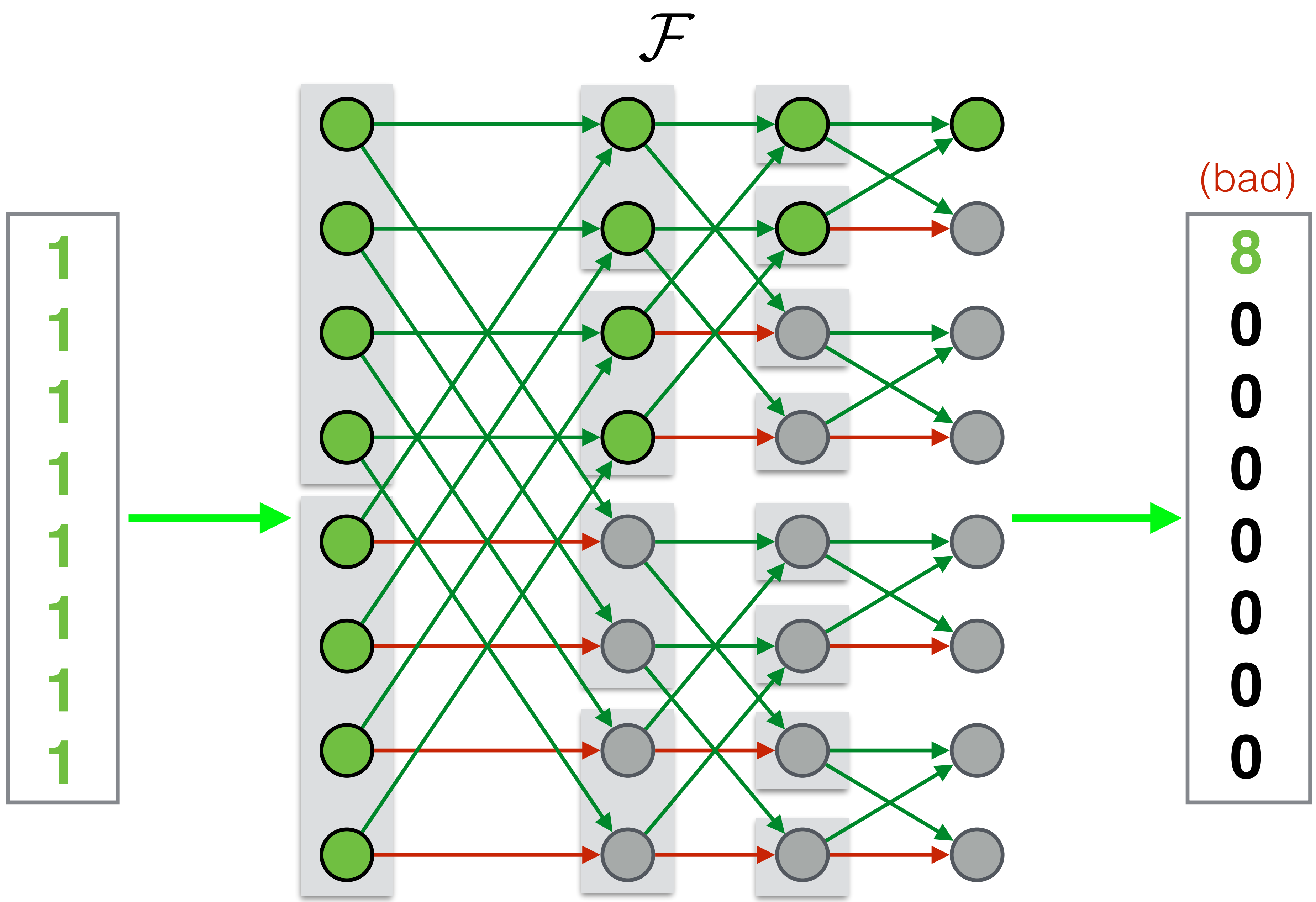
Mixing



Apply *orthogonal* operator
(change of basis)

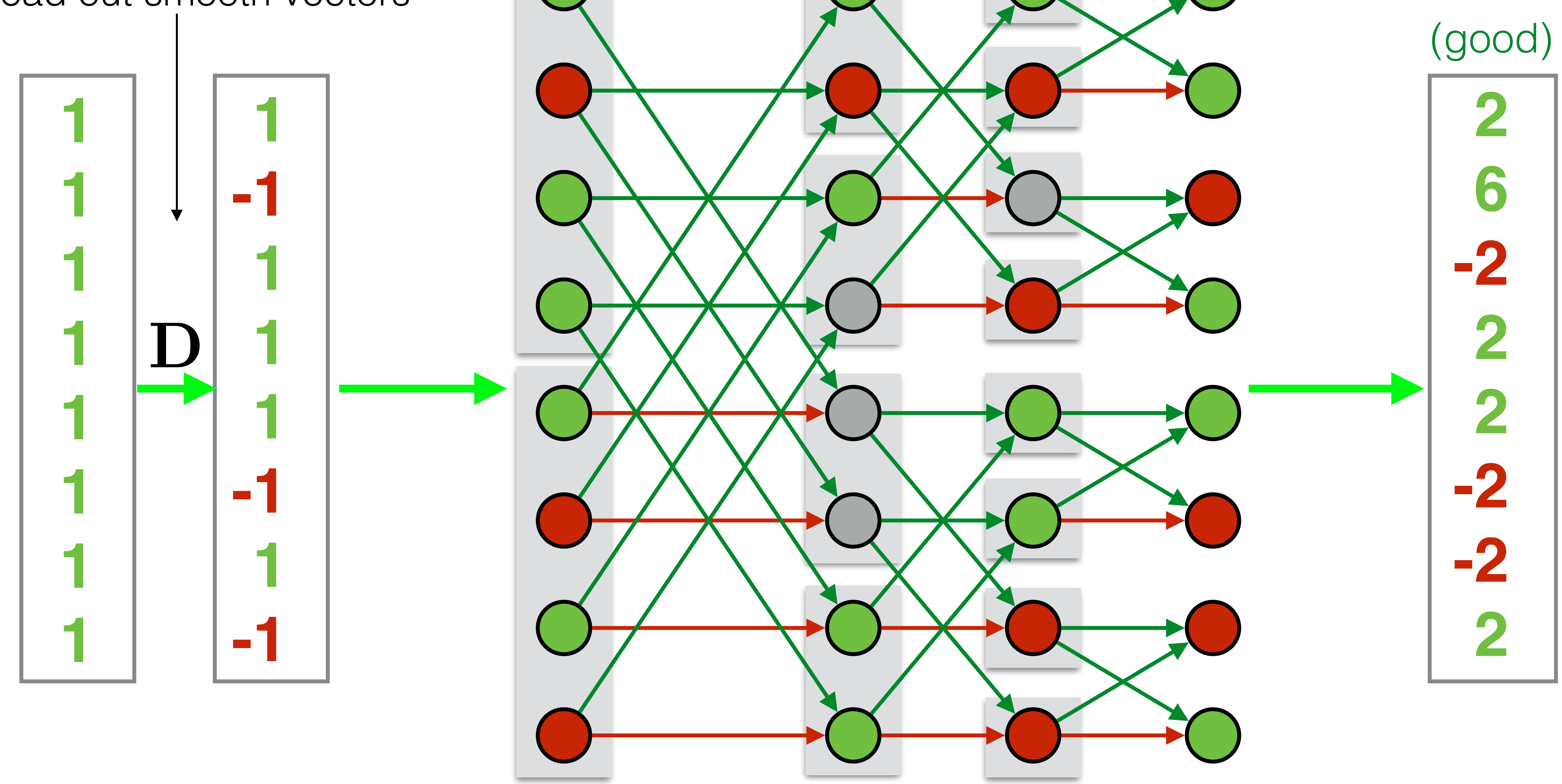
(Ailon & Chazelle, 2006 / Drineas, Mahoney, Muthukrishnan, Sarlós 2007 / Rokhlin & Tygert 2008)

One problem



Flip Signs!

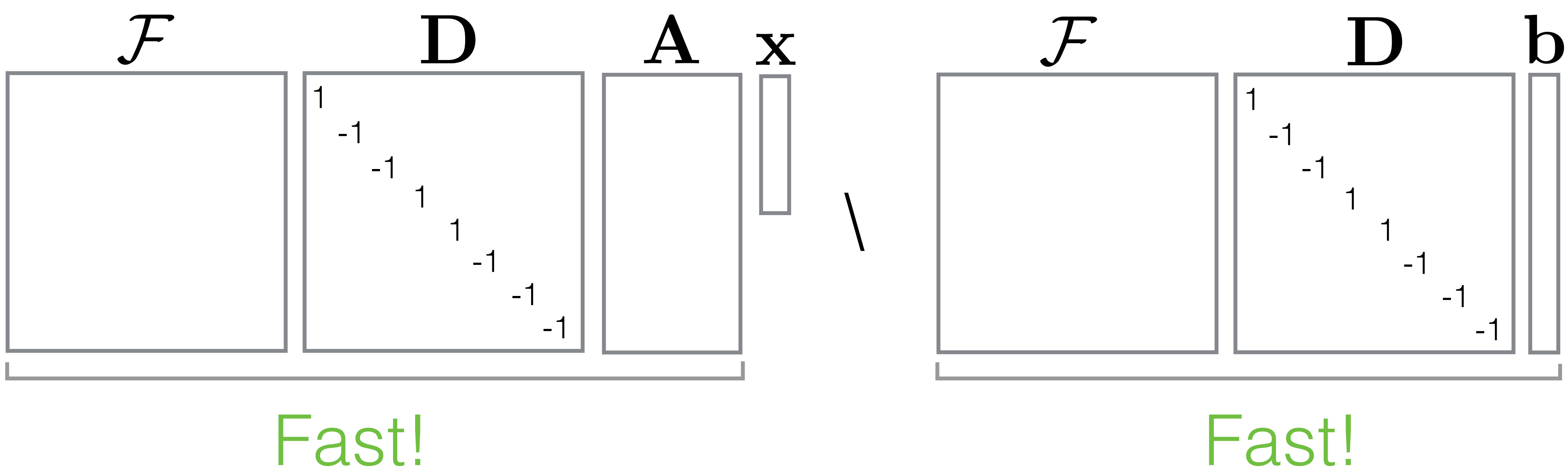
random sign flipping helps spread out smooth vectors

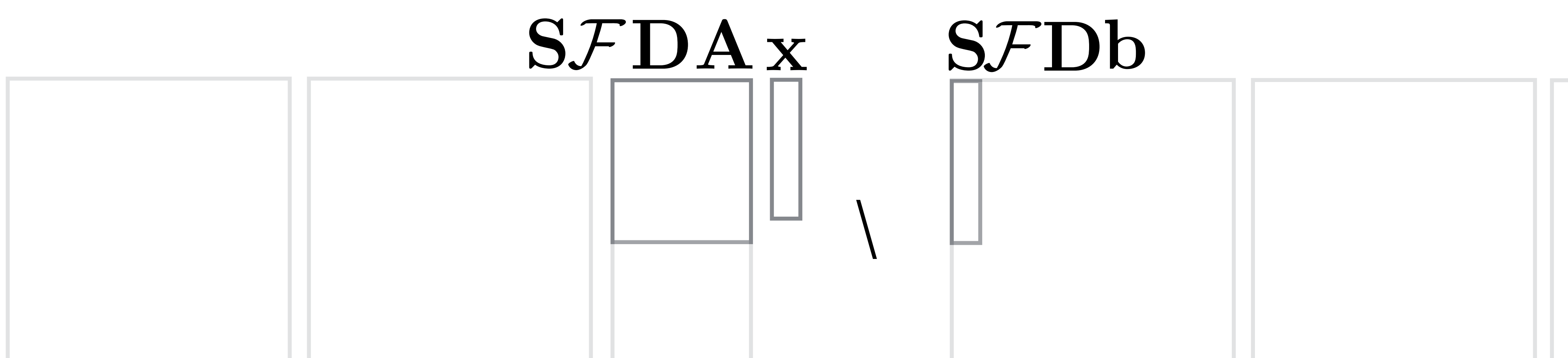


Mixing

"Fast Johnson-Lindenstrauss Transform"

(Ailon & Chazelle, 2006)




$\mathbf{S}\mathcal{F}\mathbf{D}\mathbf{A}\,\mathbf{x}$
$\backslash$
$\mathbf{S}\mathcal{F}\mathbf{D}\mathbf{b}$

Randomized Least Squares

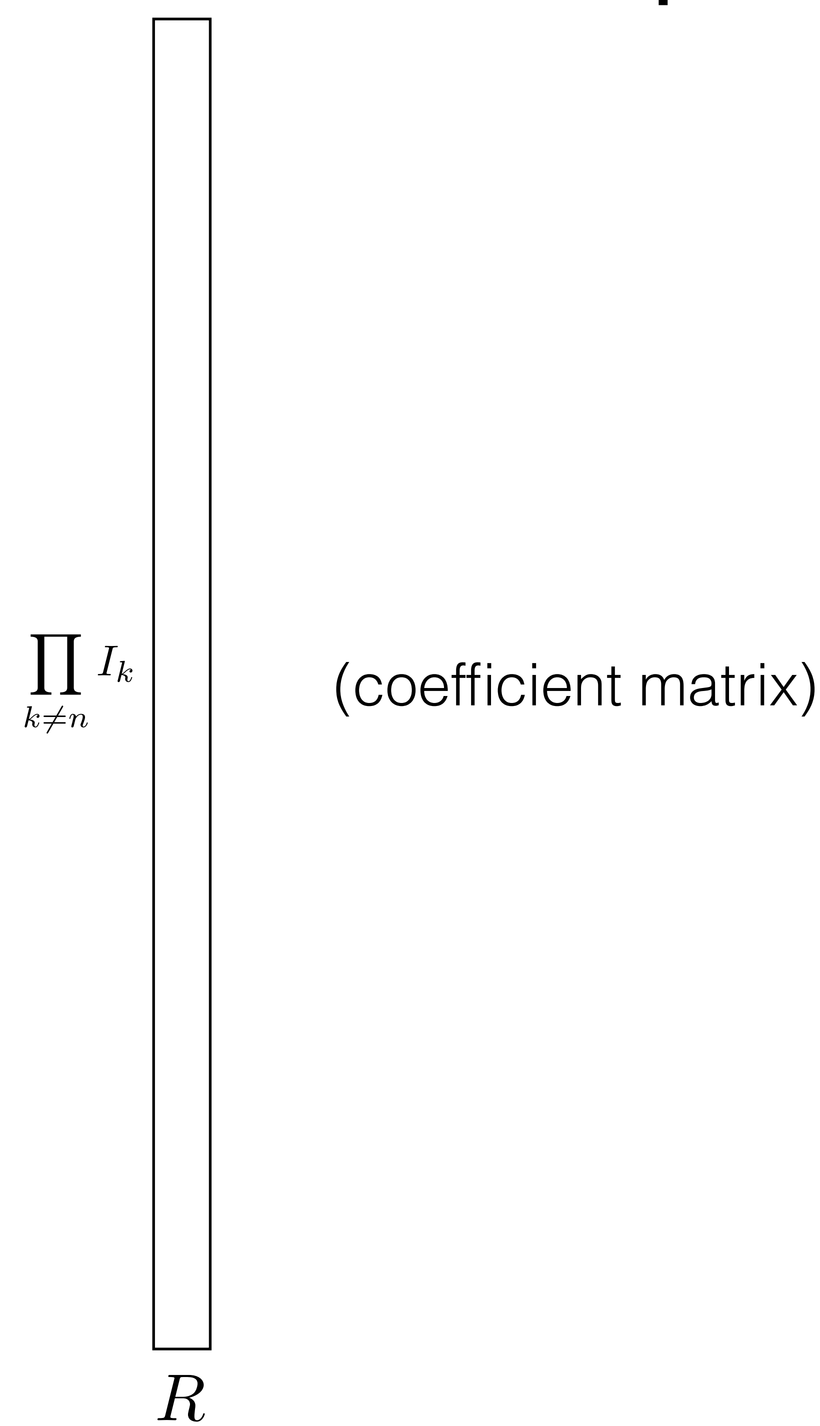
$$\min_{\mathbf{x}} \|\mathbf{S}\mathcal{F}\mathbf{D}\mathbf{A}\mathbf{x} - \mathbf{S}\mathcal{F}\mathbf{D}\mathbf{b}\|_2$$

Has found practical uses, e.g. "Blendenpik":

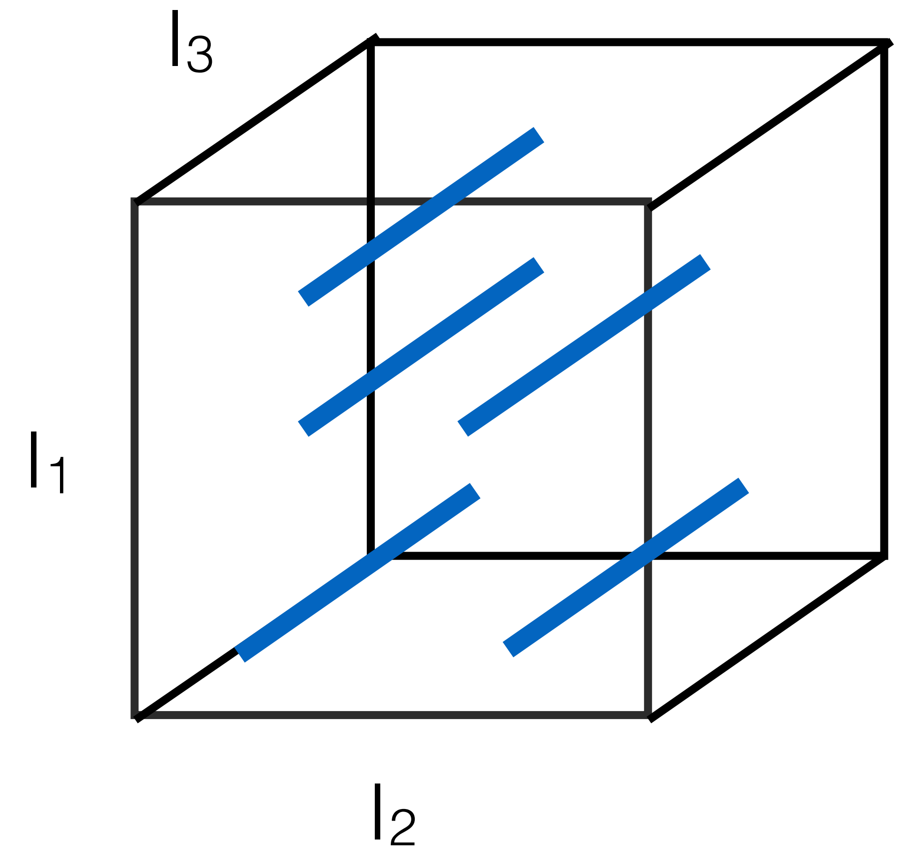
~4X overall speedup over LAPACK

(Avron, Maymounkov & Toledo 2010)

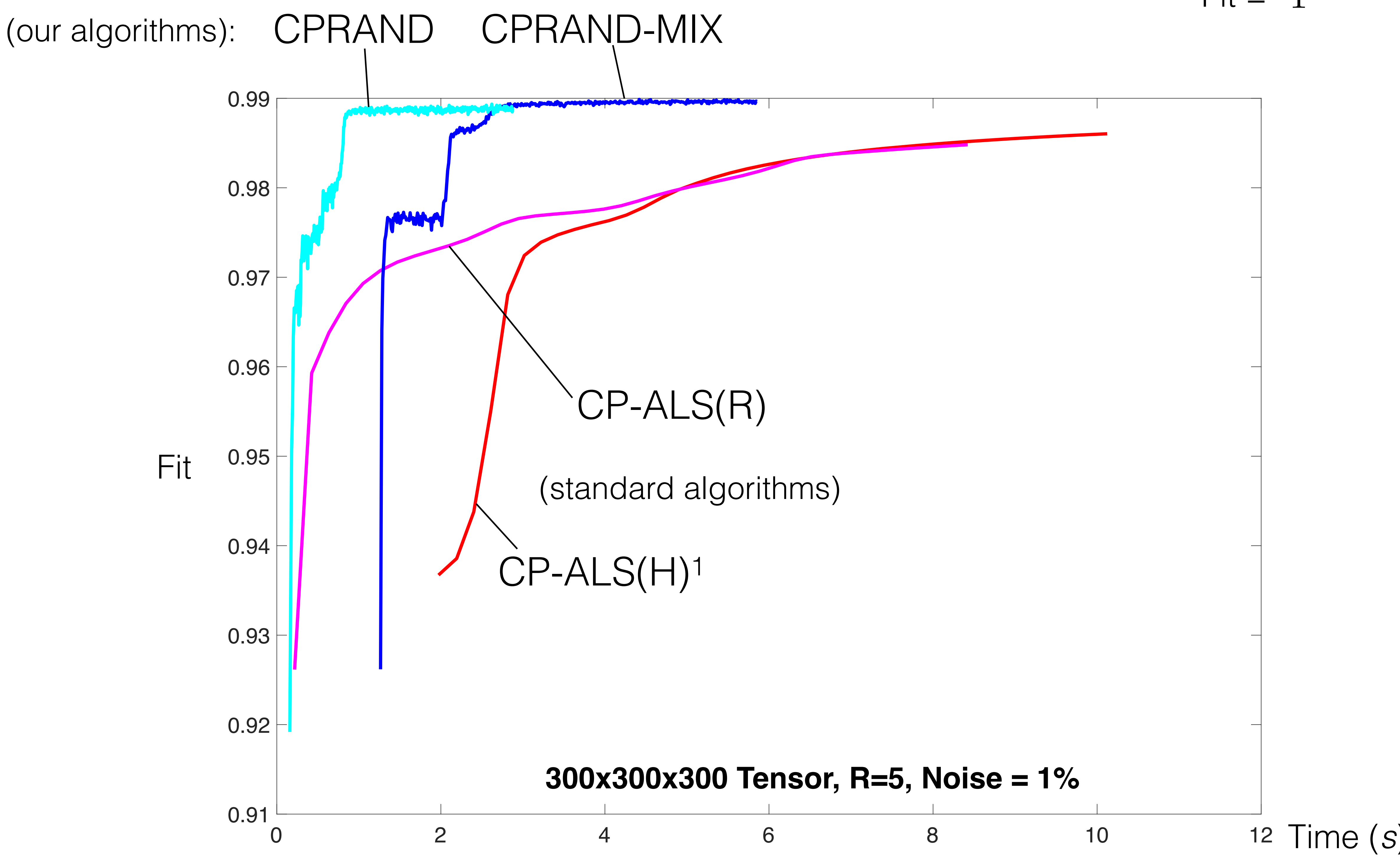
CP-ALS has Highly Overdetermined LS Subproblems!



Does this work on tensor problems?

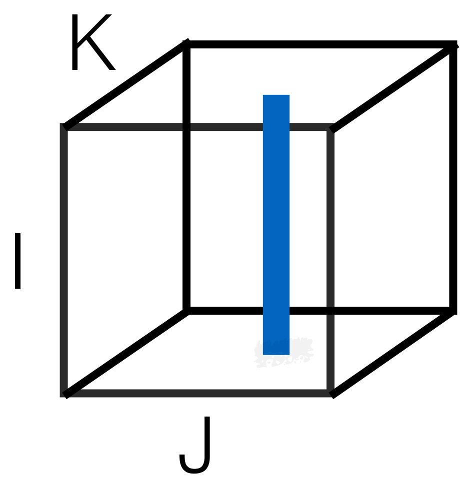


$$\text{Fit} = 1 - \frac{\|\mathcal{X} - \tilde{\mathcal{X}}\|}{\|\mathcal{X}\|}$$

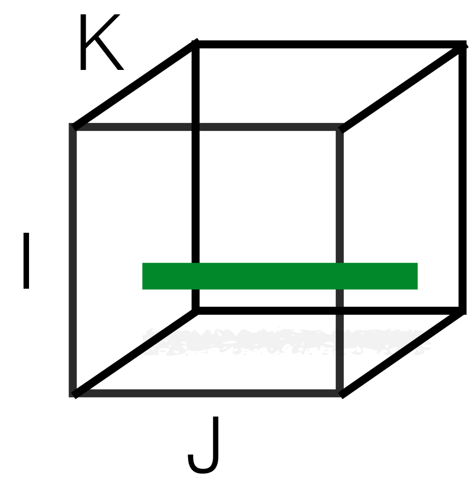


[1]: HOSVD init

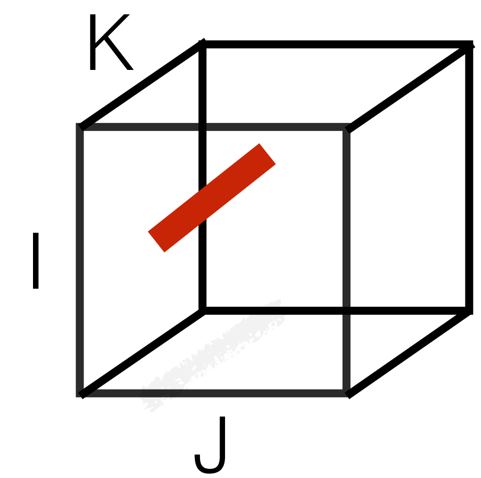
Tensor Fibers



mode-1 fibers



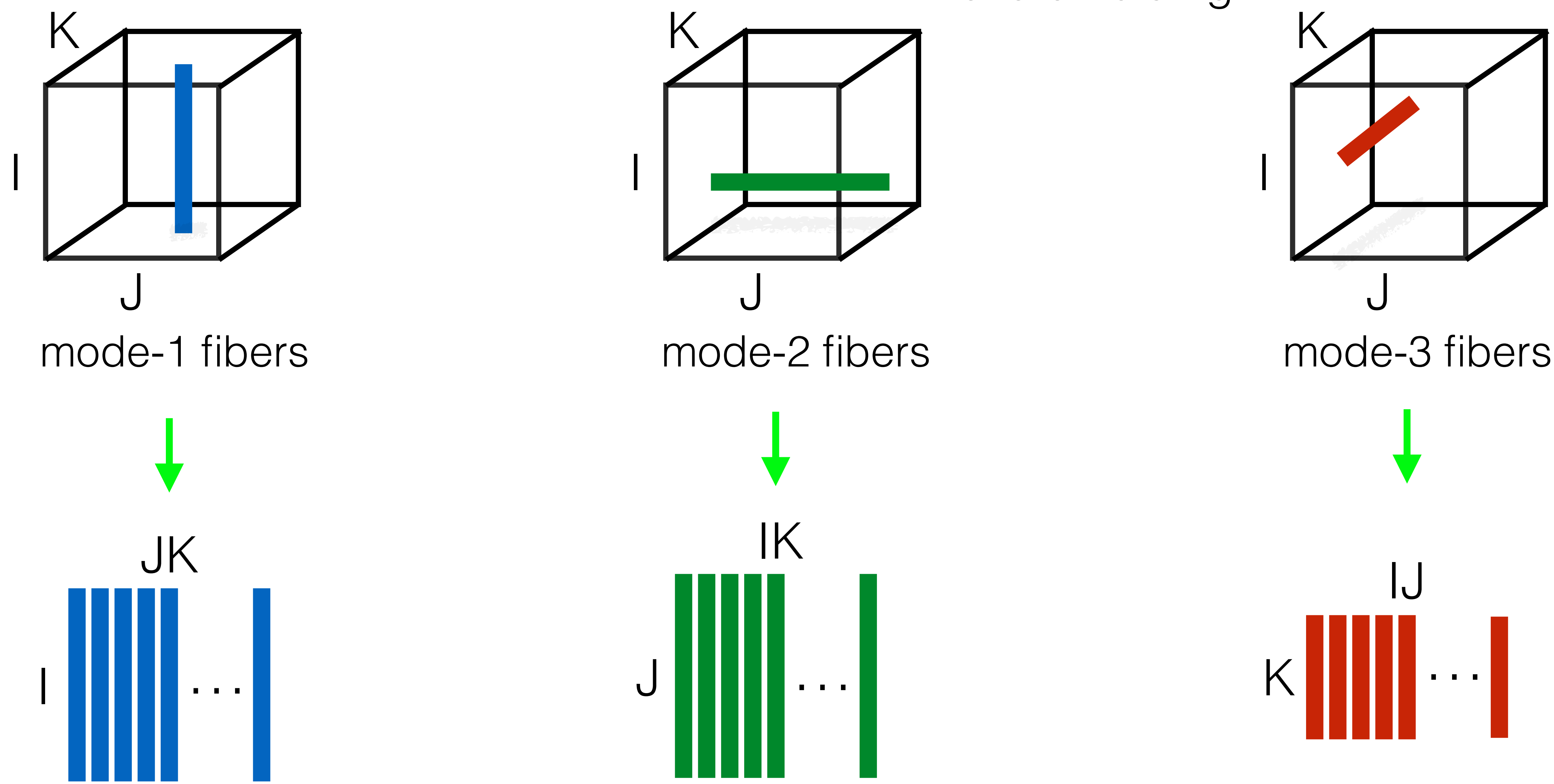
mode-2 fibers



mode-3 fibers

Matricization

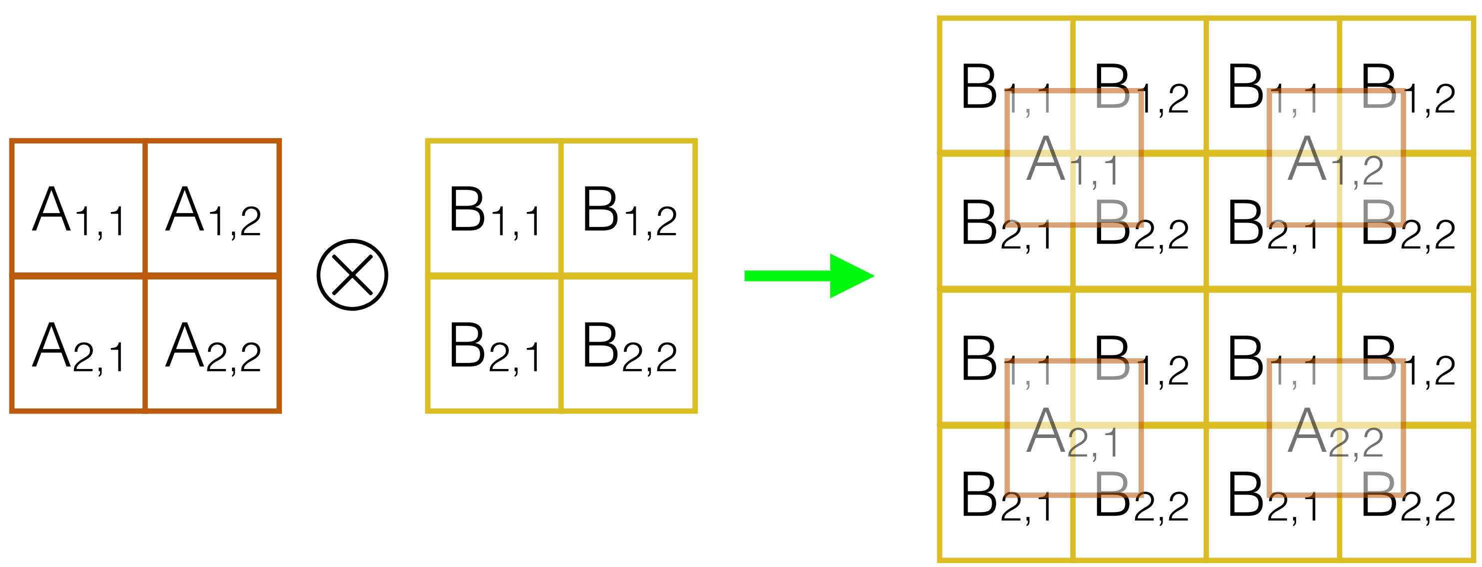
aka unfolding



$\mathbf{X}_{(1)}$ $\mathbf{X}_{(2)}$ $\mathbf{X}_{(3)}$

Kronecker Product

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix},$$



CP Decomposition & Khatri-Rao Product

The CP Representation: $$\mathcal{X} \approx \tilde{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$

Has matricized form: $$\mathbf{X}_{(1)} \approx \mathbf{A}(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$$

$$\underbrace{\begin{bmatrix} | & & | \\ \mathbf{a}_1 & \cdots & \mathbf{a}_R \\ | & & | \end{bmatrix}}_{\mathbf{A}}$$

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1, \mathbf{a}_2 \otimes \mathbf{b}_2, \ldots, \mathbf{a}_K \otimes \mathbf{b}_K]$$

CP Decomposition & Khatri-Rao Product

The CP Representation: $\mathcal{X} \approx \tilde{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$

Has matricized form: $\mathbf{X}_{(1)} \approx \mathbf{A}(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$

Solve for $\mathbf{A}$

Transposed least squares problem:

$$\Big[\quad\Big] - \begin{bmatrix} \mathbf{a}_1 \cdots \mathbf{a}_R \end{bmatrix} \begin{bmatrix} (\mathbf{c}_1 \otimes \mathbf{b}_1)^{\mathsf{T}} \\ \vdots \\ (\mathbf{c}_R \otimes \mathbf{b}_R)^{\mathsf{T}} \end{bmatrix}$$

$$\|\mathbf{X}_{(1)} - \mathbf{A} (\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}\|_F^2$$

Alternating Least Squares

CP-ALS uses a trick to solve a much smaller LS problem:

order 3:

$$\mathbf{A} \leftarrow \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B}) \quad / \quad \overbrace{(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{B}^{\mathsf{T}}\mathbf{B})}^{R \times R}$$

$$\mathbf{B} \leftarrow \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A}) \quad / \quad (\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{A}^{\mathsf{T}}\mathbf{A})$$

$$\mathbf{C} \leftarrow \mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A}) \quad / \quad (\mathbf{B}^{\mathsf{T}}\mathbf{B} * \mathbf{A}^{\mathsf{T}}\mathbf{A})$$

gram structure

until fit stops changing:

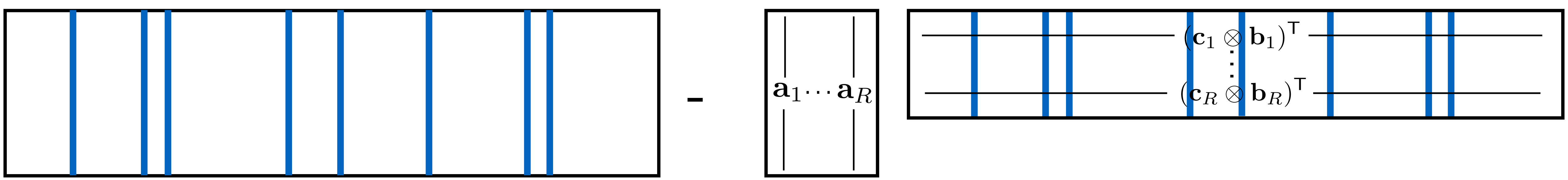
$$1 - \frac{\|\mathcal{X} - \tilde{\mathcal{X}}\|}{\|\mathcal{X}\|}$$

CPRAND

$$\mathbf{A} \leftarrow \mathbf{X}_{(1)} \quad / \quad (\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$$

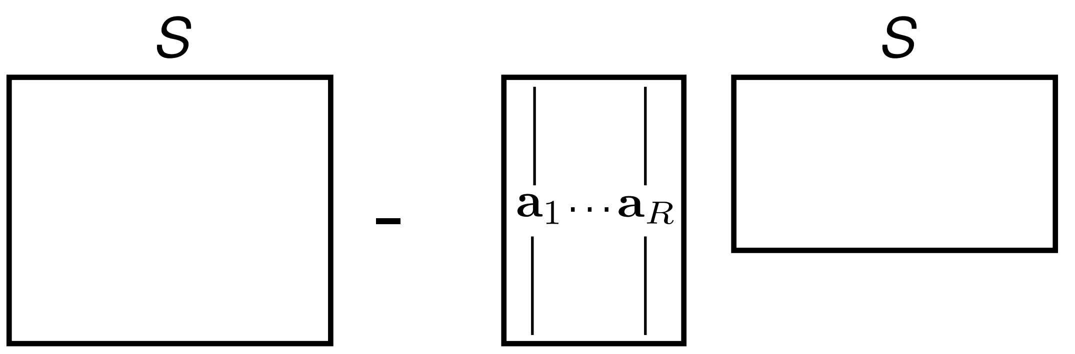
$$\Downarrow$$

$$\mathbf{A} \leftarrow \mathbf{X}_{(1)}\mathbf{S}^{\mathsf{T}} \quad / \quad (\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}\mathbf{S}^{\mathsf{T}}$$



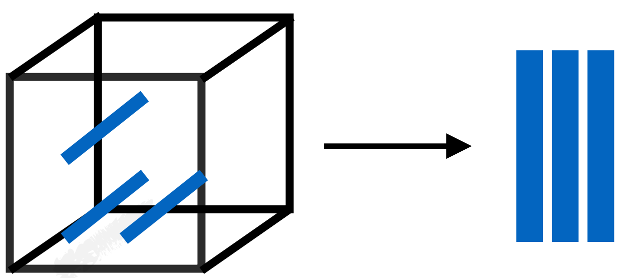
$\mathbf{X}_{(1)}\mathrm{S}^{\mathsf{T}}$ $\quad$ $\mathbf{A}$ $\quad$ $(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}\mathrm{S}^{\mathsf{T}}$

CPRAND



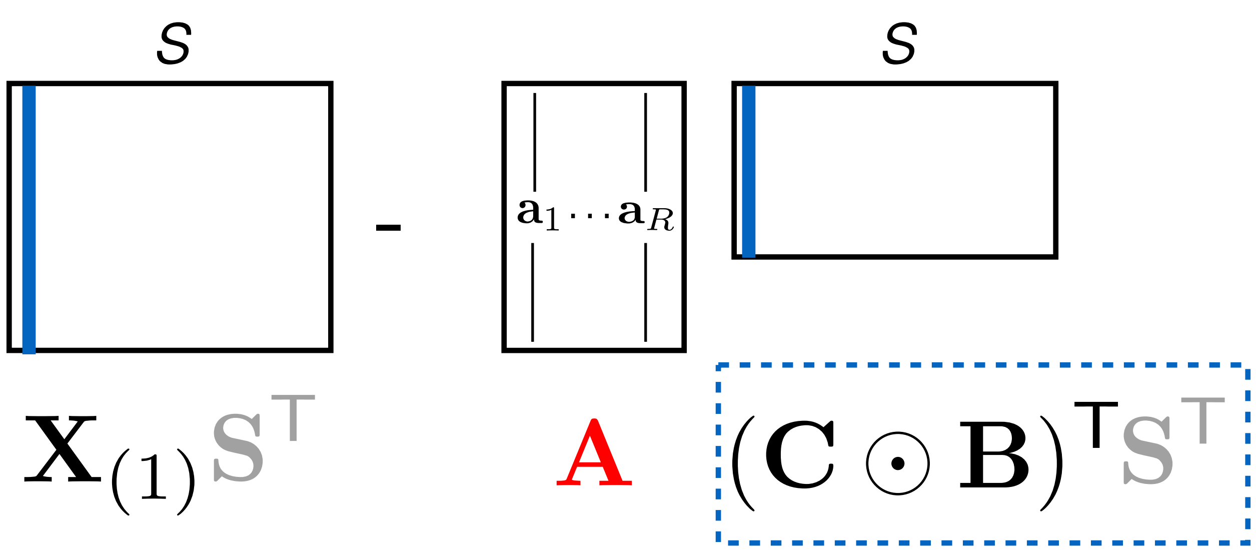
$$\boxed{\mathbf{X}_{(1)} \mathbf{S}^{\mathsf{T}}} \quad {\color{red}\mathbf{A}} \quad (\mathbf{C} \odot \mathbf{B})^{\mathsf{T}} \mathbf{S}^{\mathsf{T}}$$

Trick 1:

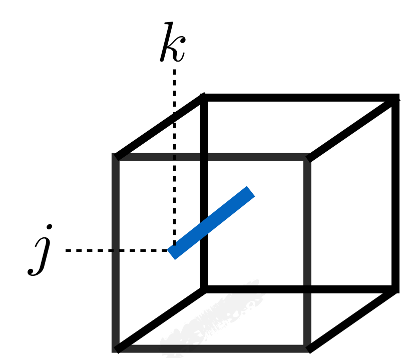


[don’t need to matricize.]()

CPRAND



Trick 2:



Each column is $(\mathbf{C}(:,k) * \mathbf{B}(:,j))^\mathsf{T}$
for each corresponding fiber sampled

don't need to form full KR product.

CPRAND

(Order 3 Example, extends to arbitrary order)

---

procedure CPRAND($\mathcal{X}, R, S$) $\qquad \triangleright\ \mathcal{X} \in \mathbb{R}^{I \times J \times K}$

  Initialize factor matrices $\mathbf{B}, \mathbf{C}$

  repeat

    Define uniform random sampling operators $\mathbf{S}_A, \mathbf{S}_B, \mathbf{S}_C$

    $\mathbf{A} \leftarrow \mathbf{X}_{(1)} \mathbf{S}_A^{\mathsf{T}} (SKR(\mathbf{S}_A, \mathbf{C}, \mathbf{B})^{\mathsf{T}})^{\dagger}$

    $\mathbf{B} \leftarrow \mathbf{X}_{(2)} \mathbf{S}_B^{\mathsf{T}} (SKR(\mathbf{S}_B, \mathbf{C}, \mathbf{A})^{\mathsf{T}})^{\dagger}$

    $\mathbf{C} \leftarrow \mathbf{X}_{(3)} \mathbf{S}_C^{\mathsf{T}} (SKR(\mathbf{S}_C, \mathbf{B}, \mathbf{A})^{\mathsf{T}})^{\dagger}$

  until termination criteria met

  return $\mathbf{A}, \mathbf{B}, \mathbf{C}$

end procedure

No mixing performed here, yet converges in many cases!

Result: Khatri Rao product actually performs some “mixing”

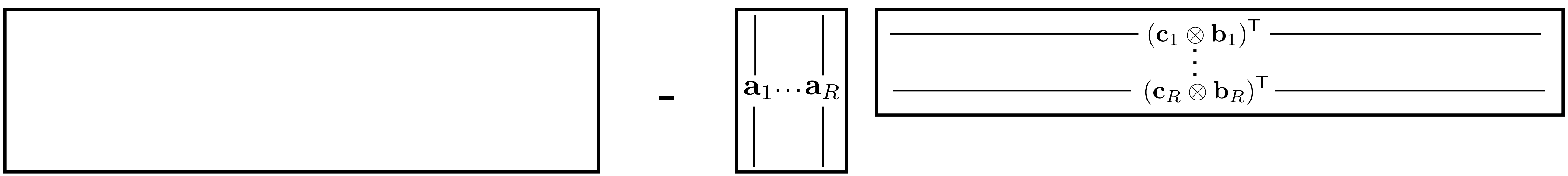
(see paper)

Mixing Tensors

Some data sets will still require mixing to converge well.

Mixing Tensors

(mix) (mix)



$$\| \mathbf{X}_{(1)} \quad - \quad \mathbf{A} \quad \boxed{(\mathbf{C} \odot \mathbf{B})^\mathsf{T}} \|_F^2$$

Trick 3: Instead of mixing $(\mathbf{C} \odot \mathbf{B})^\mathsf{T}$ mix $\mathbf{C}, \mathbf{B}$ separately:

$$(\mathcal{F}_C \mathbf{D}_C \mathbf{C} \odot \mathcal{F}_B \mathbf{D}_B \mathbf{B})$$

Mixing Factors

RHS:

$$(\mathcal{F}_C \mathbf{D}_C \mathbf{C} \odot \mathcal{F}_B \mathbf{D}_B \mathbf{B})^{\mathsf{T}}$$

$$= \underline{(\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B)}(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$$

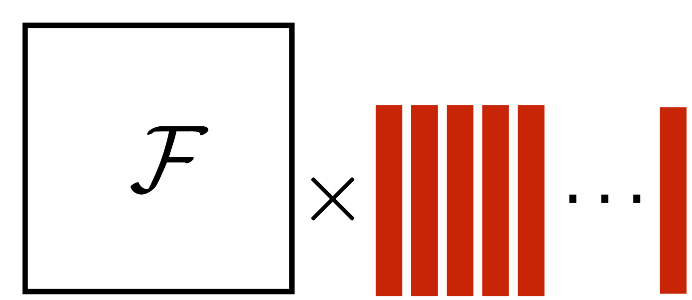
LHS:

$$\mathbf{X}_{(1)}\underline{(\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B)}$$

$$= \mathcal{X} \times_1 \mathcal{F}_C \mathbf{D}_C \times_2 \mathcal{F}_B \mathbf{D}_B$$

TTM:

$$\mathcal{X} \times_n \mathbf{U} = \mathbf{U}\mathbf{X}_{(n)}$$



mode-3 fibers

Mixing Factors

Trick 4:
Mix original tensor only once:

$$\hat{\mathcal{X}} = \mathcal{X} \times_3 \mathcal{F}_C \mathbf{D}_C \times_2 \mathcal{F}_B \mathbf{D}_B \times_1 \mathcal{F}_A \mathbf{D}_A$$

matricize in one mode

$$\hat{\mathbf{X}}_{(1)} = \mathbf{D}_A \mathcal{F}_A \mathbf{X}_{(1)} (\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B)$$

sample rows

$$\mathbf{D}_A \mathcal{F}_A \mathbf{X}_{(1)} (\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B) \mathbf{S}^T$$

solve least squares

$$\hat{\mathbf{A}} \leftarrow \mathbf{D}_A \mathcal{F}_A \mathbf{X}_{(1)} (\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B) \mathbf{S}^T (\mathbf{S}\mathbf{Z}^{(1)\mathrm{T}})^\dagger$$

inverse mix the final, small, solution

$$\mathbf{A} \leftarrow \mathbf{D}_A^{-1} \mathcal{F}_A^{-1} \left[\cdots \right]$$

Mixing Factors

Trick 4:
Mix original tensor only once:

$$\hat{\mathcal{X}} = \mathcal{X} \times_3 \mathcal{F}_C \mathbf{D}_C \times_2 \mathcal{F}_B \mathbf{D}_B \times_1 \mathcal{F}_A \mathbf{D}_A$$

matricize in one mode

$$\hat{\mathbf{X}}_{(1)} = \mathbf{D}_A \mathcal{F}_A \mathbf{X}_{(1)} (\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B)$$

sample rows

$$\mathbf{D}_A \mathcal{F}_A \mathbf{X}_{(1)} (\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B) \mathbf{S}^T$$

solve least squares

$$\hat{\mathbf{A}} \leftarrow \mathbf{D}_A \mathcal{F}_A \mathbf{X}_{(1)} (\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B) \mathbf{S}^T (\mathbf{S}\mathbf{Z}^{(1)\mathrm{T}})^\dagger$$

inverse mix the final, small, solution

$$\mathbf{A} \leftarrow \mathbf{D}_A^{-1} \mathcal{F}_A^{-1} \left[\hat{\mathbf{A}} \right]$$

CPRAND-MIX

Summary:

Trick 4:
Mix original tensor only once:

inverse mix small result at end

solve small system of size *s* * *R*

$$\mathbf{A}^{(n)} \leftarrow \overbrace{\mathbf{D}_n^{-1}\mathcal{F}_n^{-1}}^{\text{inverse mix small result at end}}(\hat{\mathbf{X}}_{(n)}\mathbf{S}^{\intercal})\overbrace{[(\mathbf{S}\hat{\mathbf{Z}}^{(n)})^{*}]^{\dagger}}^{\text{solve small system of size } s * R}$$

efficiently sample tensor **(Trick 1)**

don't form KRP **(Trick 2)**

mix individual factors **(Trick 3)**

Algorithm 3: CPRAND-MIX

(Order 3 Example, extends to arbitrary order)

procedure CPRAND$(\mathcal{X}, R, S)$ $\quad \triangleright\ \mathcal{X} \in \mathbb{R}^{I \times J \times K}$

Mix: $\hat{\mathcal{X}} \leftarrow \mathcal{X} \times_3 \mathcal{F}_C \mathbf{D}_C \times_2 \mathcal{F}_B \mathbf{D}_B \times_1 \mathcal{F}_A \mathbf{D}_A$

Initialize factor matrices $\mathbf{B}, \mathbf{C}$

repeat

Define uniform random sampling operators $\mathbf{S}_A, \mathbf{S}_B, \mathbf{S}_C$

$$\mathbf{A} \leftarrow \mathbf{D}_A^{-1} \mathcal{F}_A^{-1} \left[\hat{\mathbf{X}}_{(1)} \mathbf{S}_A^\intercal (SKR(\mathbf{S}_A, \mathcal{F}_C \mathbf{D}_C \mathbf{C}, \mathcal{F}_B \mathbf{D}_B \mathbf{B})^\intercal)^\dagger \right]$$

$$\mathbf{B} \leftarrow \mathbf{D}_B^{-1} \mathcal{F}_B^{-1} \left[\hat{\mathbf{X}}_{(2)} \mathbf{S}_B^\intercal (SKR(\mathbf{S}_B, \mathcal{F}_C \mathbf{D}_C \mathbf{C}, \mathcal{F}_A \mathbf{D}_A \mathbf{A})^\intercal)^\dagger \right]$$

$$\mathbf{C} \leftarrow \mathbf{D}_C^{-1} \mathcal{F}_C^{-1} \left[\hat{\mathbf{X}}_{(3)} \mathbf{S}_C^\intercal (SKR(\mathbf{S}_C, \mathcal{F}_B \mathbf{D}_B \mathbf{B}, \mathcal{F}_C \mathbf{D}_C \mathbf{C})^\intercal)^\dagger \right]$$

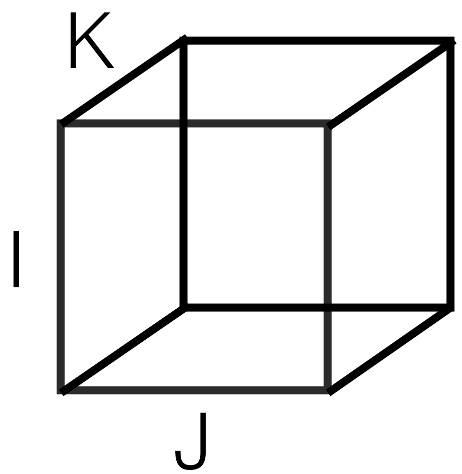
until termination criteria met

return $\mathbf{A}, \mathbf{B}, \mathbf{C}$

end procedure

(Trick 5)

Trick 5: # Stopping Criterion

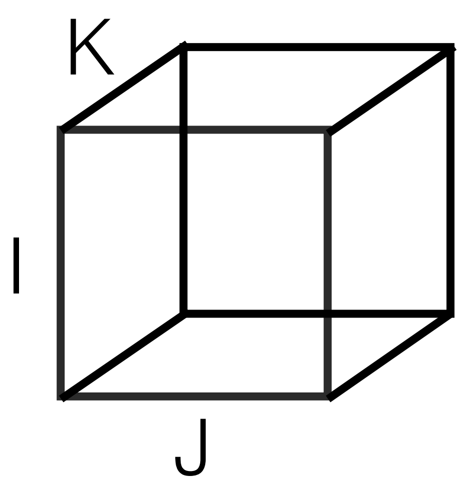


CP-ALS:

expensive inner product

$$1 - \frac{\|\mathcal{X} - \tilde{\mathcal{X}}\|}{\|\mathcal{X}\|} = 1 - \frac{\sqrt{\|\mathcal{X}\|^2 + \|\tilde{\mathcal{X}}\|^2 - 2 < \mathcal{X}, \tilde{\mathcal{X}} >}}{\|\mathcal{X}\|}$$

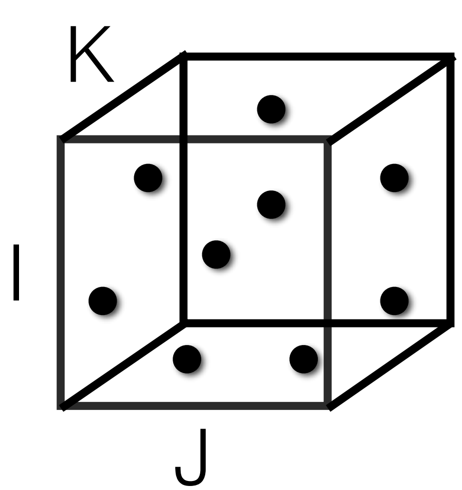
Trick 5: Stopping Criterion



CP-ALS:

expensive inner product

$$1 - \frac{\|\mathcal{X} - \tilde{\mathcal{X}}\|}{\|\mathcal{X}\|} = 1 - \frac{\sqrt{\|\mathcal{X}\|^2 + \|\tilde{\mathcal{X}}\|^2 - 2 < \mathcal{X}, \tilde{\mathcal{X}} >}}{\|\mathcal{X}\|}$$



CPRAND:

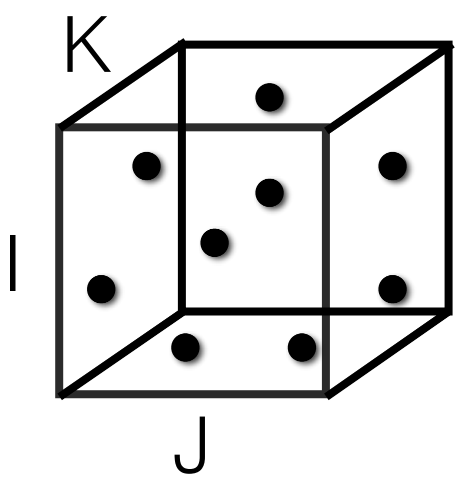
Uniformly sample $\mathcal{X}$:

$$\mathcal{I}' \subset \mathcal{I} \equiv [I] \otimes [J] \otimes [K]$$

approximate fit

$$\text{fit} = 1 - \frac{\sqrt{|\mathcal{I}| \cdot \text{mean}\{\, e_{\mathbf{i}}^2 \mid \mathbf{i} \in \mathcal{I} \,\}}}{\|\mathcal{X}\|} \approx \boxed{1 - \frac{\sqrt{|\mathcal{I}| \cdot \text{mean}\{\, e_{\mathbf{i}}^2 \mid \mathbf{i} \in \mathcal{I}' \,\}}}{\|\mathcal{X}\|}}$$

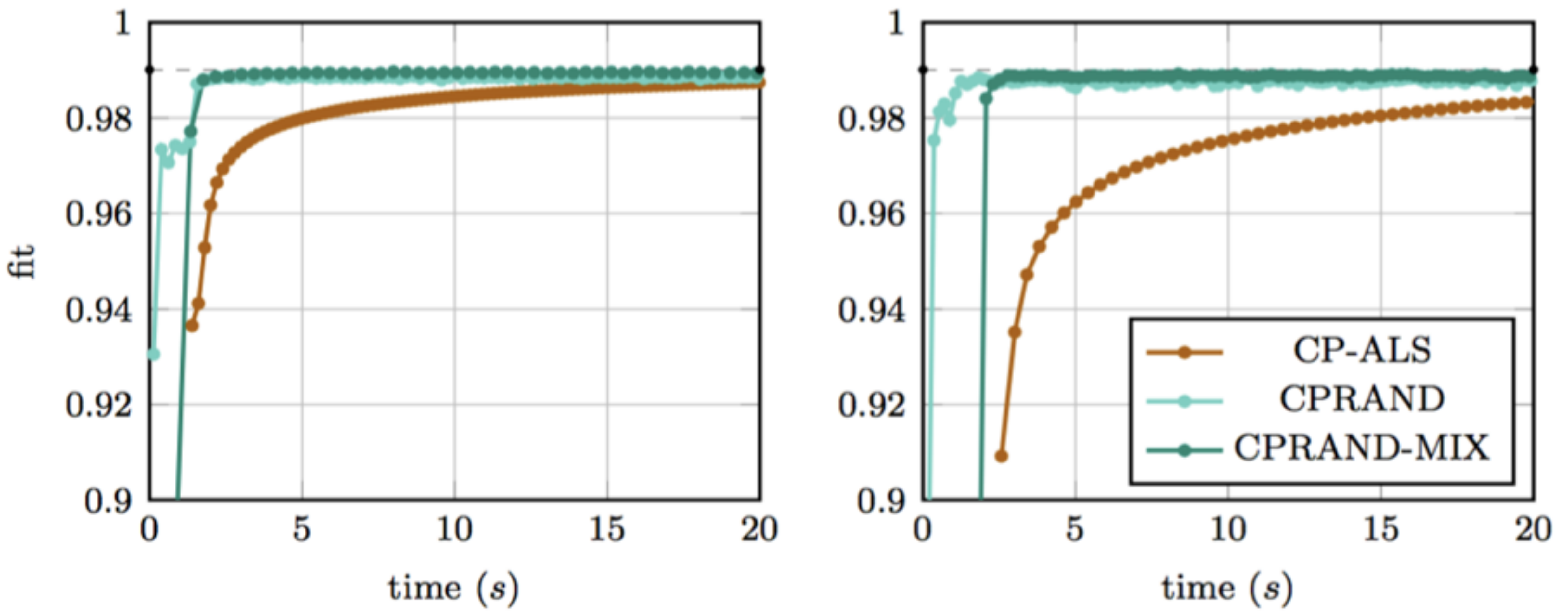
Trick 5: Stopping Criterion



Result: Chernoff-Hoeffding gives a prob. guarantee on error!

In practice: $\hat{P} = 2^{14}$ yields error $< 10^{-3}$ on synthetic data

Example Run

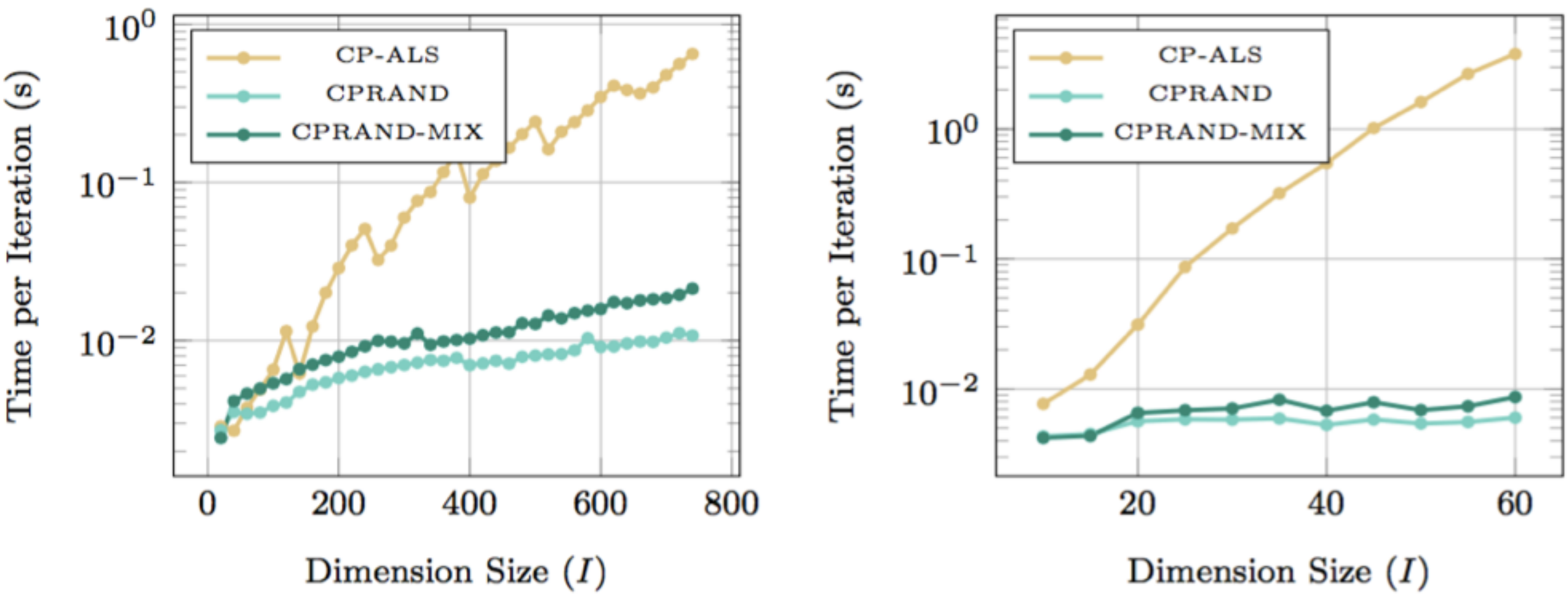


(a) Random 300 × 300 × 300 tensor

(b) Random 80 × 80 × 80 × 80 tensor

True Rank = 5, Target Rank = 5, Noise = 1%, Collinearity 0.9

Per-Iteration Times



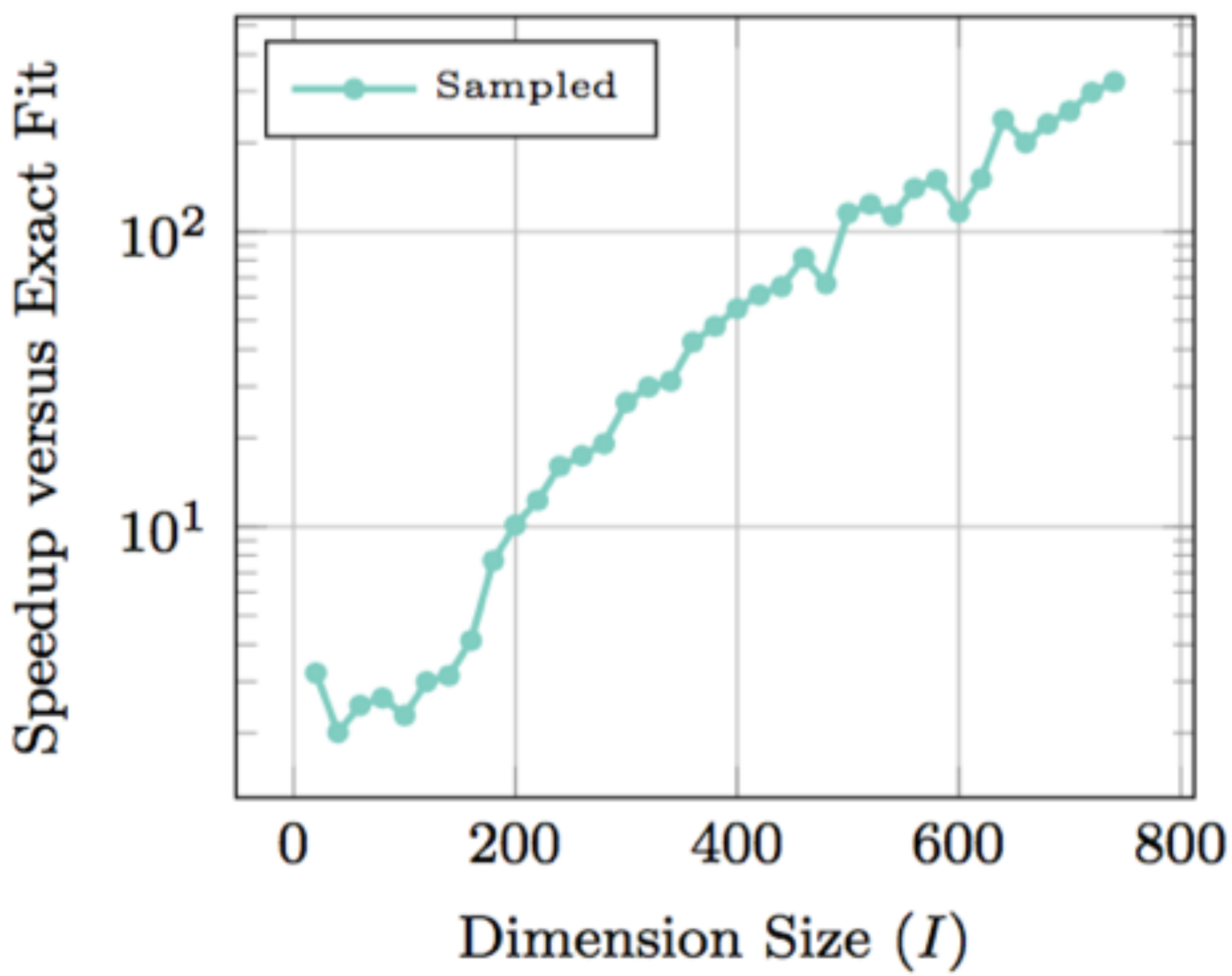
(a) Order 3: $I \times I \times I$

(b) Order 5: $I \times I \times I \times I \times I$

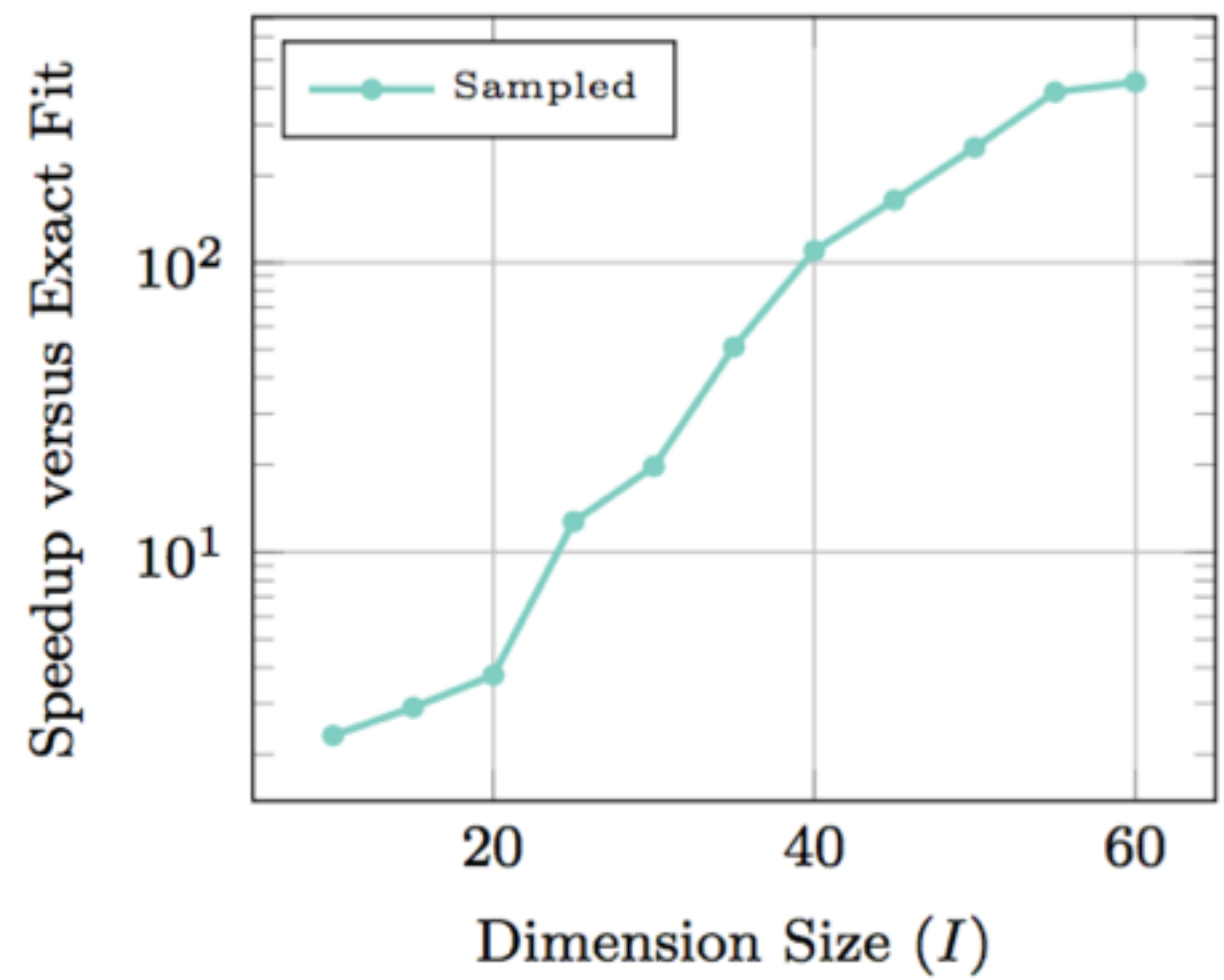
$$R = 5 \qquad S = 10R\log R$$

Stopping Criterion Speedup

(stopping condition with $\hat{P} = 2^{14}$)



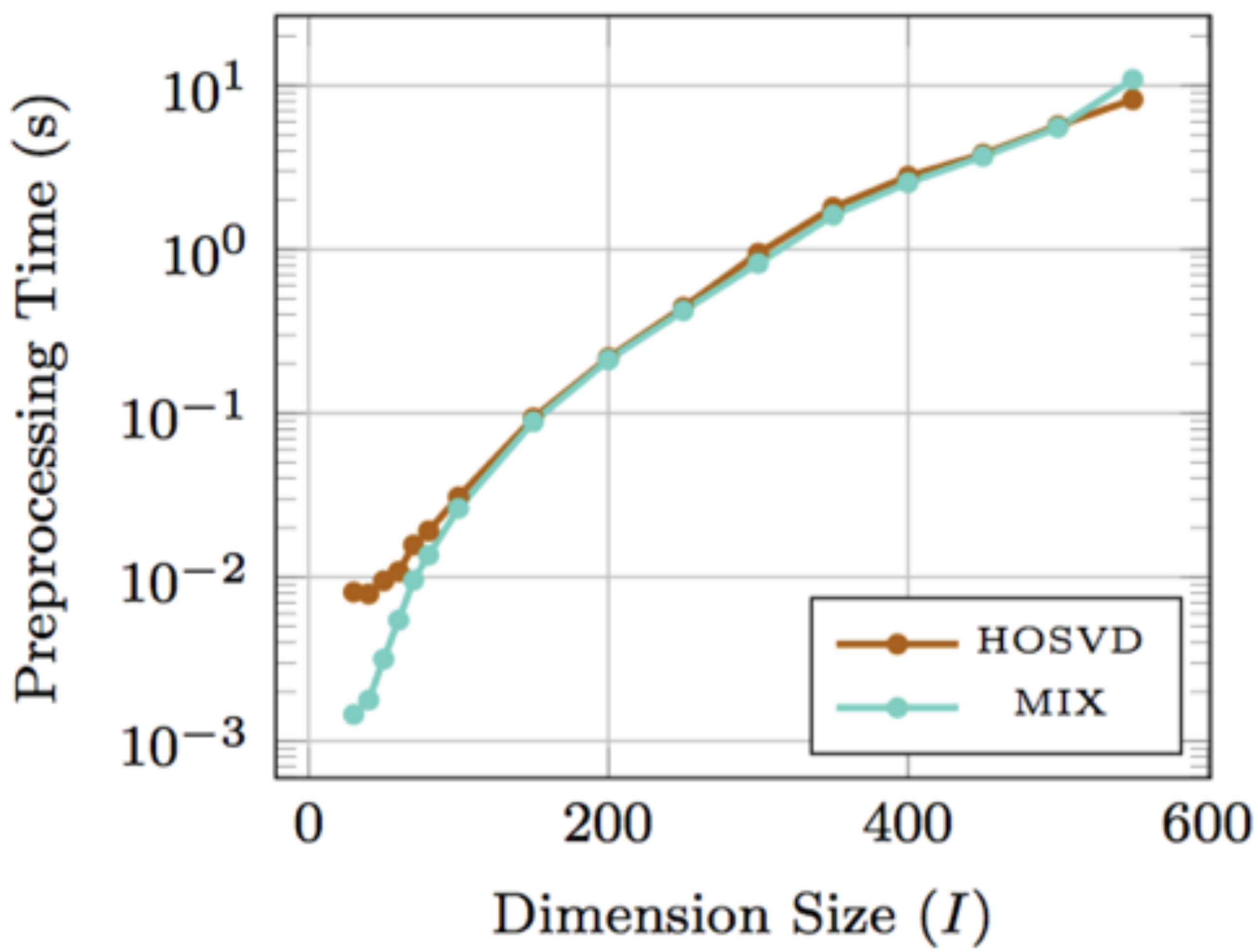
(a) Order 3: $I \times I \times I$



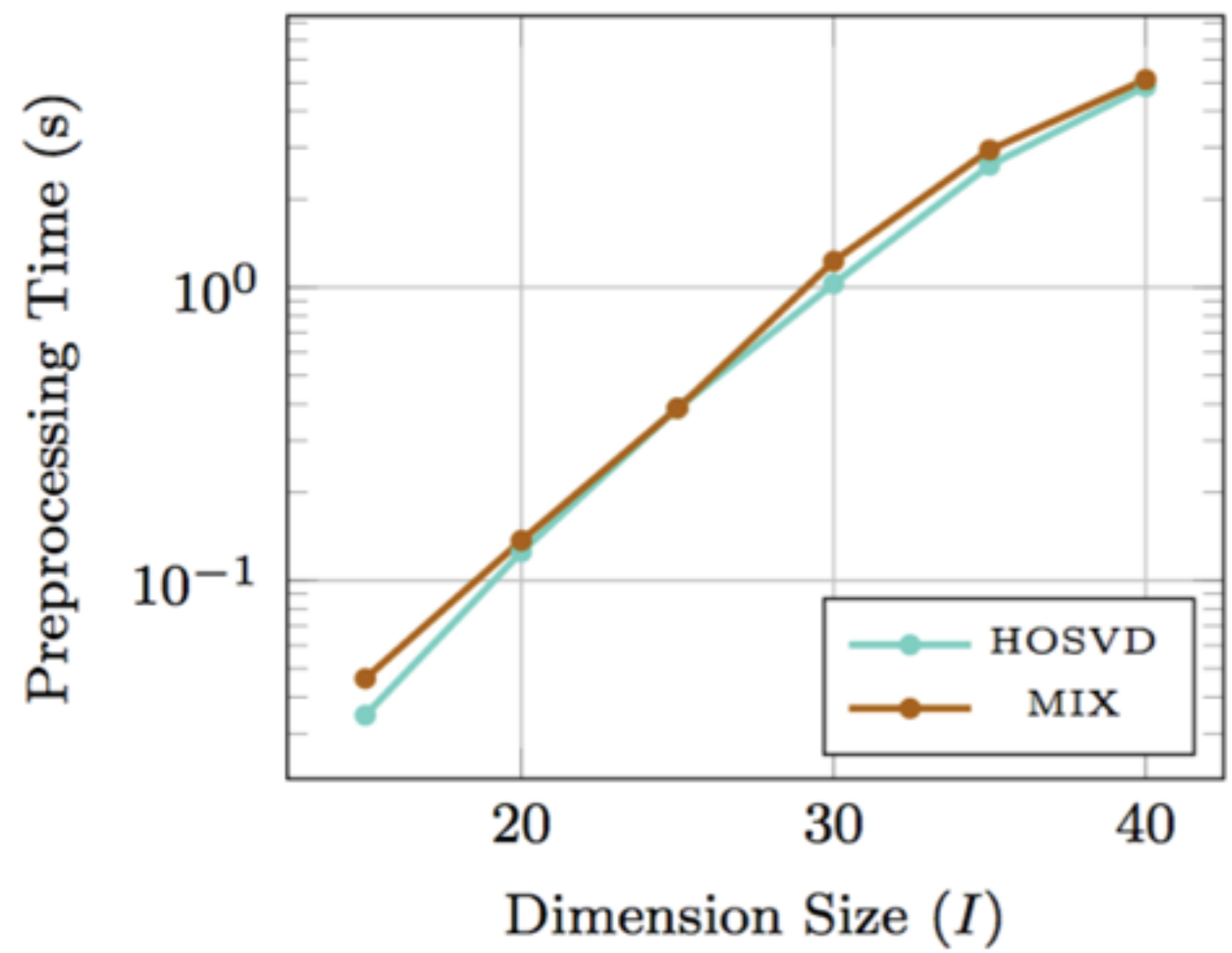
(b) Order 5: $I \times I \times I \times I \times I$

Preprocessing Times

nvec on each mode-*n* unfolding vs. **fft** on fibers of each mode



(a) Order 3: $I \times I \times I$



(b) Order 5: $I \times I \times I \times I \times I$

Synthetic Trials

Data

$$\mathcal{X} = \mathcal{X}_{\text{true}} + \eta \left(\frac{\|\mathcal{X}_{\text{true}}\|}{\|\mathcal{N}\|} \right) \mathcal{N}$$

— Synthetic random

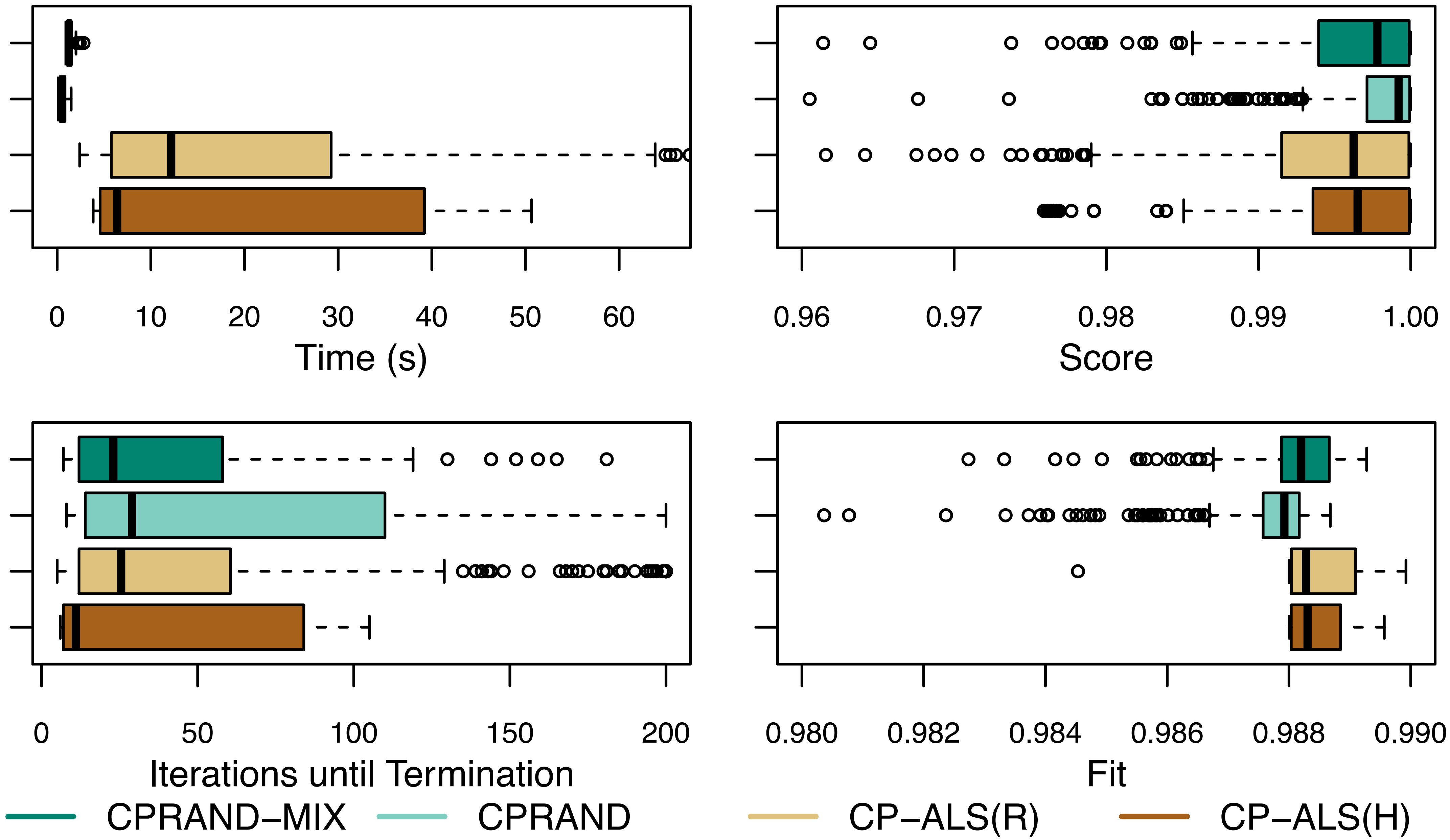
— Factors of rank R_{true}

— Factors are generated with *collinearity*

— Observation is combined with Gaussian noise

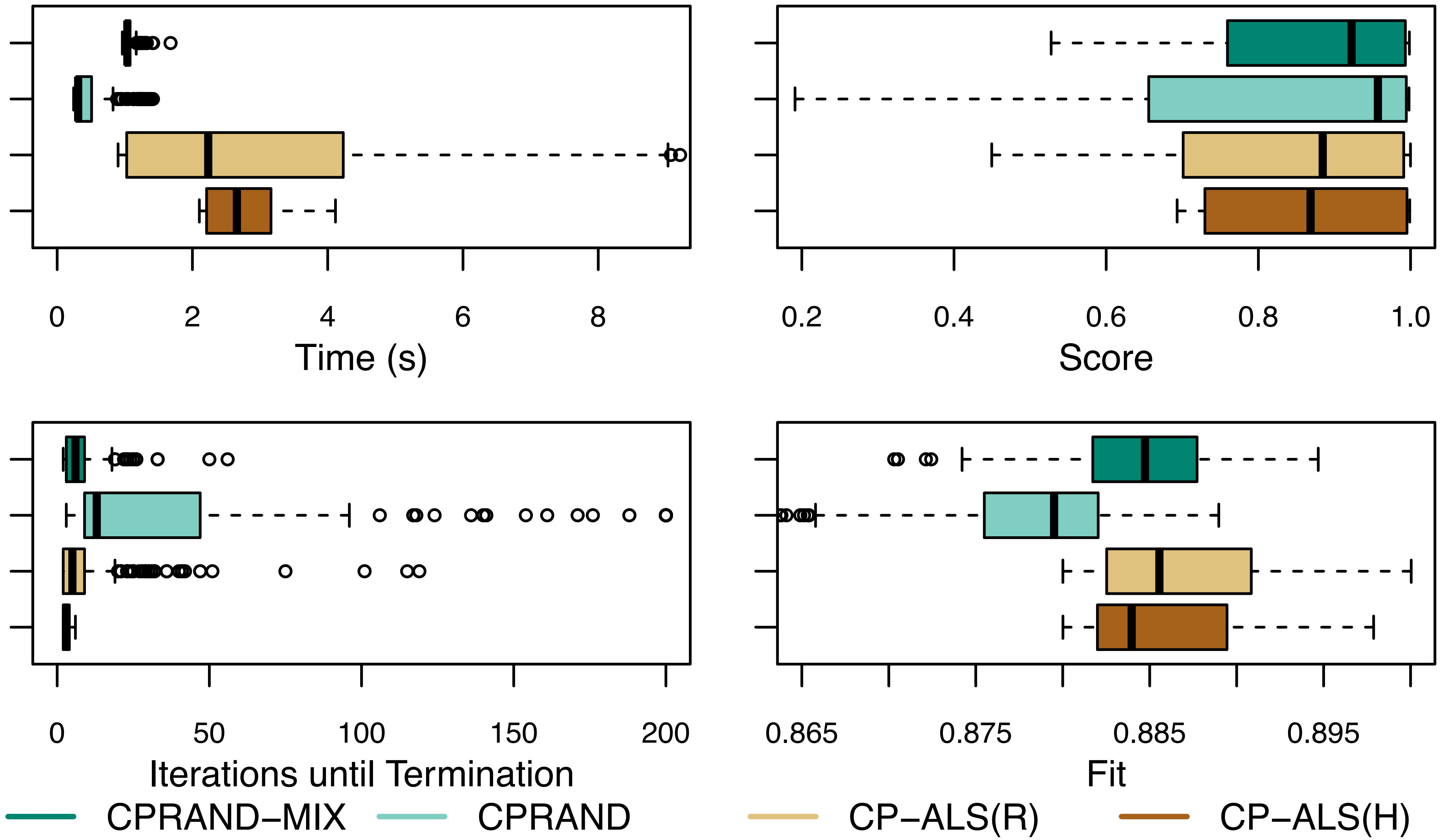
— **“Score” is a measure of how well the original factors are recovered, from 0 to 1**

Order 4 Trials



Size: 90x90x90x90	**R**true: 5	**R**: {5,6}
Noise: 1%	200 Runs	**Collinearity**: {0.5,0.9}

Order 4 Trials



Size: 90x90x90x90	R_{true}: 5	**R**: {5,6}
Noise: 10%	200 Runs	**Collinearity**: {0.5,0.9}

Conclusion
(Experiment 1)

- CPRAND can surprisingly outperform CP-ALS in both time and factor recovery (given the same termination criteria).

- CPRAND has very low per-iteration time, but may take more iterations until termination.

- CPRAND-FFT takes fewer iterations and samples at the cost of pre-processing time.

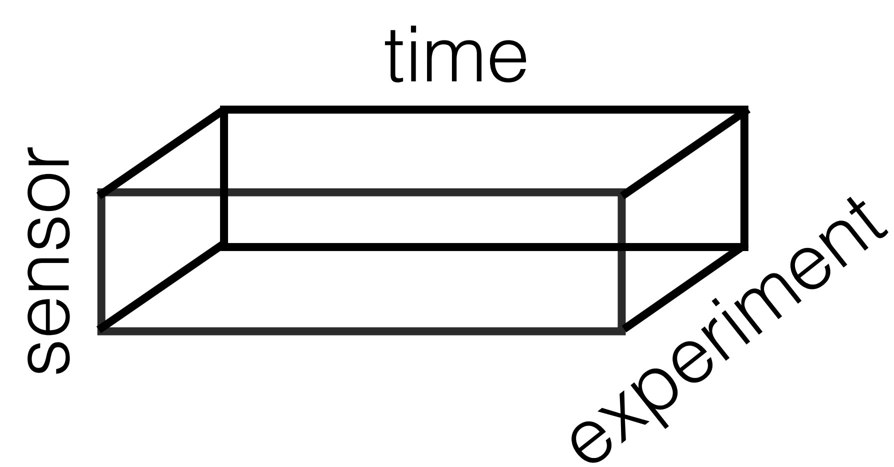
Experiment 2: Hazardous Gas Classification

We replicate an experiment from Vervliet & Lauthauwer, 2016:

72 sensors
25,900 time steps
300 experiments per gas
(3 gases: CO, Acetaldehyde, Ammonia)

↓

25900 x 72 x 900 Tensor
13.4 GB, dense



Goal: Classify which gas is released in each experiment

Here we use CPRAND without mixing!

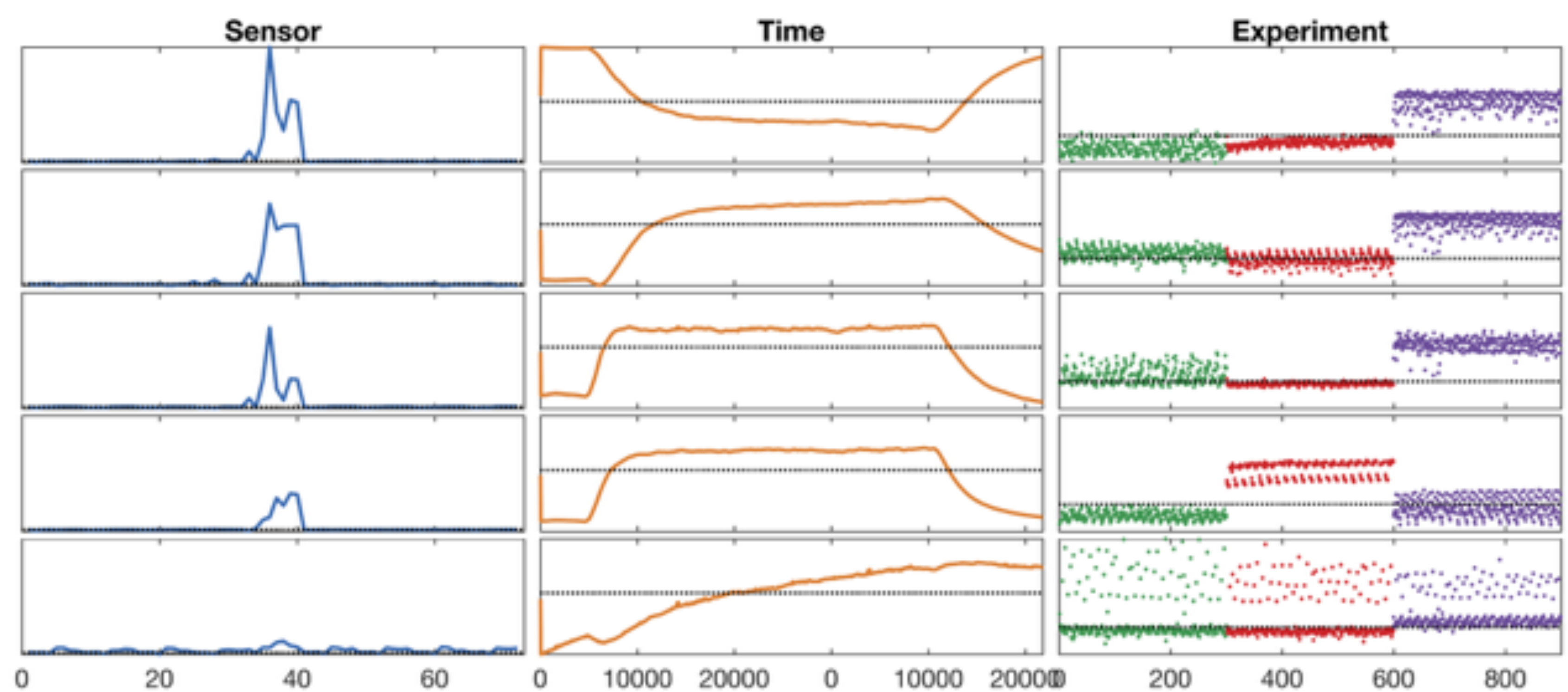
Experiment 2: Hazardous Gas Classification

(over 10 trials)

Method	Median Time (s)	Median Fit	Median Classification Error
CPRAND	53.6	0.715	0.61%
CP-ALS(H)	578.4	0.724	0.67%
CP-ALS(R)	204.7	0.724	0.67%

$S = 1000, \quad \hat{P} = 2^{14}$

Factors Recovered by CPRAND:



Experiment 3: COIL-100 Image Set

We replicate an experiment from Zhou, Cichocki, and Xie (2013):

COIL-100: 100 objects, images of each from 72 different poses.
Each image is RGB with size 128x128 pixels.
$\underbrace{128 \times 128}_{\text{image size}} \times \underbrace{3}_{\text{color channel}} \times \underbrace{7200}_{\text{image id}}$ Tensor (2.8 GB)

http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php

Experiment 3: COIL-100 Image Set

Here we use CPRAND-FFT (mixing):

# Samples (S)	Median Speedup	Median Fit
400	8.38	0.674
450	7.98	0.676
500	6.63	0.677
550	7.29	0.678
600	4.75	0.680
650	4.73	0.680
700	4.77	0.680
750	4.52	0.681
800	3.70	0.682
850	4.90	0.678
900	4.95	0.679
950	4.22	0.682
1000	2.84	0.684

$$5 \text{ trials}, \ R = 20, \ \hat{P} = 2^{14}$$

Future

- Sketching algorithms for distributed-memory HOSVD (in progress)
- Sparse projections for CP, Tucker

Contact

- Casey Battaglino (cbattaglino3@gatech.edu)
- Grey Ballard (ballard@wfu.edu)
- Tamara Kolda (tgkolda@sandia.gov)

Casey Battaglino, Grey Ballard, and Tamara G Kolda. A practical randomized cp tensor decomposition. arXiv preprint arXiv:1701.06600, 2017. URL: https://arxiv.org/abs/1701.06600