A Practical Randomized CP Tensor Decomposition

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

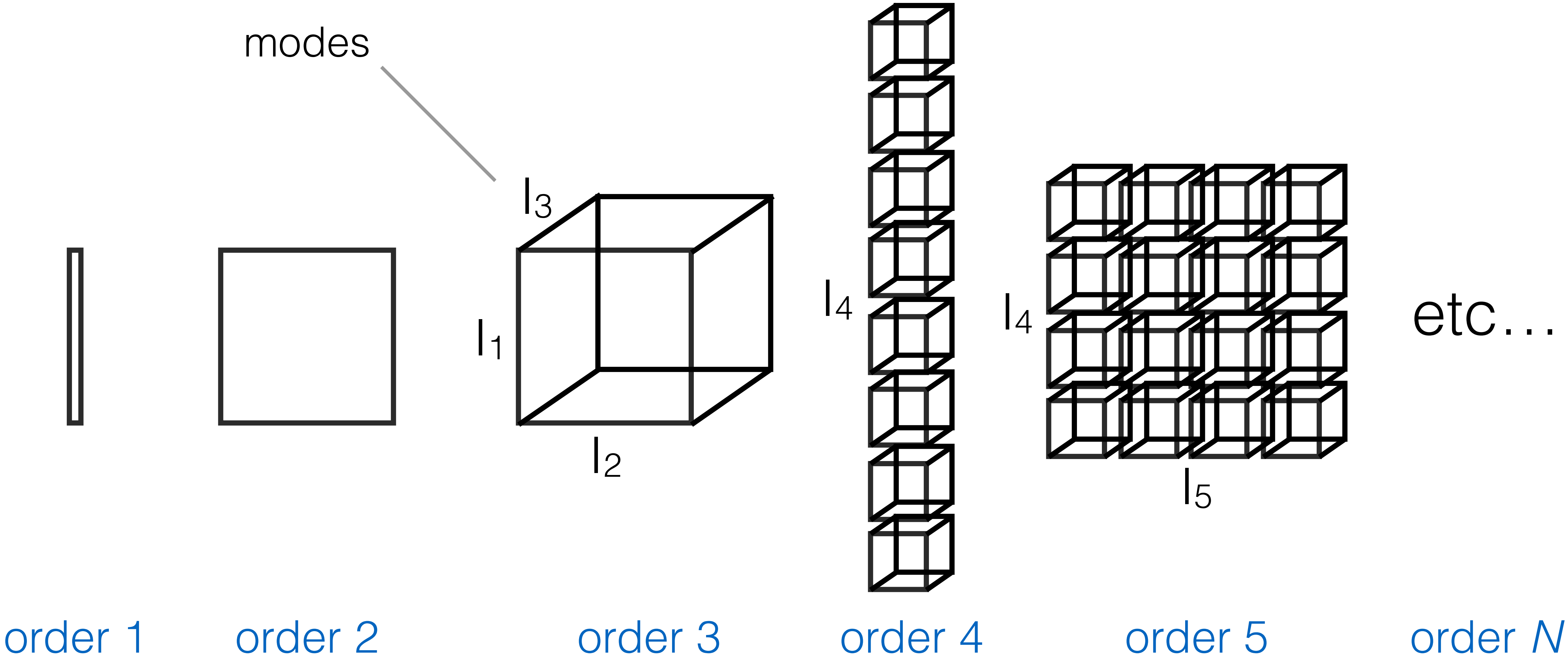
SIAM CSE 2017 Atlanta, GA

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

Wake Forest
University
[2]ballard@wfu.edu

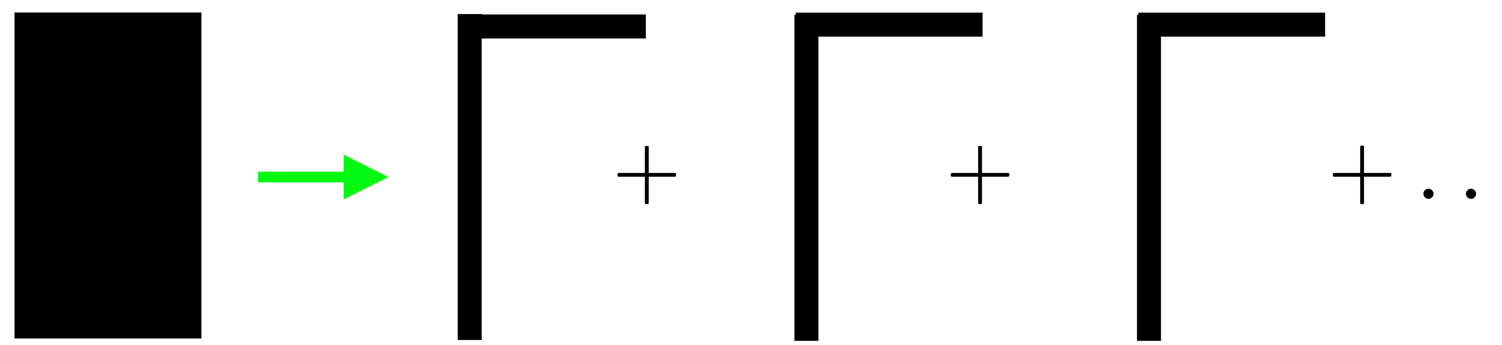
Sandia National Labs
[3]tgkolda@sandia.gov

Tensors


modes
I_3
I_1
I_2
I_4
I_4
I_5
etc…
order 1
order 2
order 3
order 4
order 5
order N

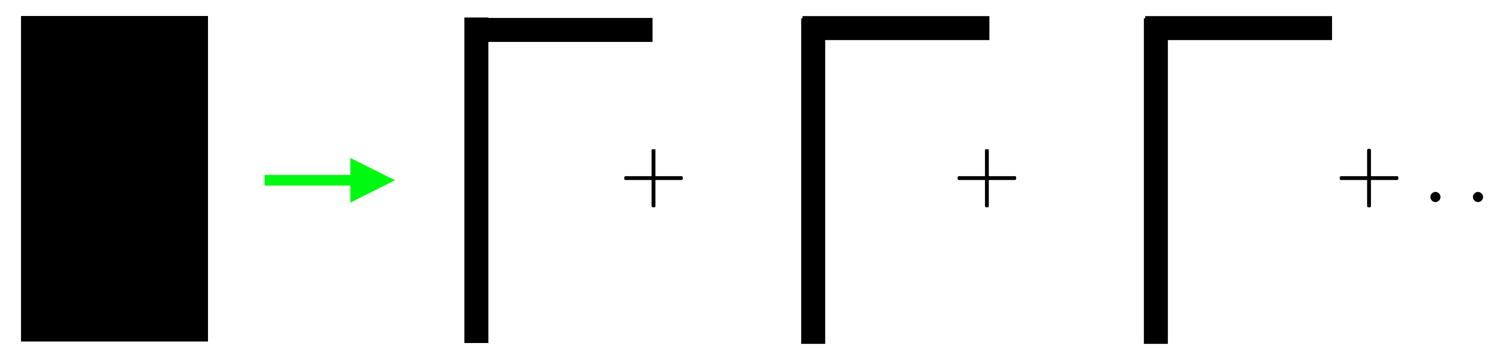
Problem

Matrix
Decomposition
(SVD)

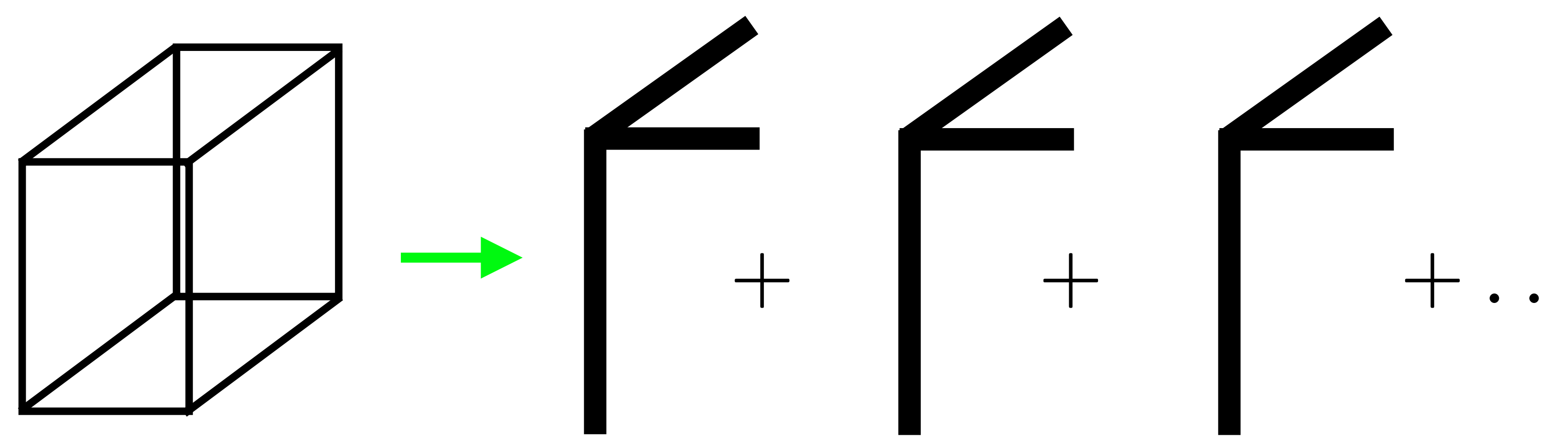


Problem

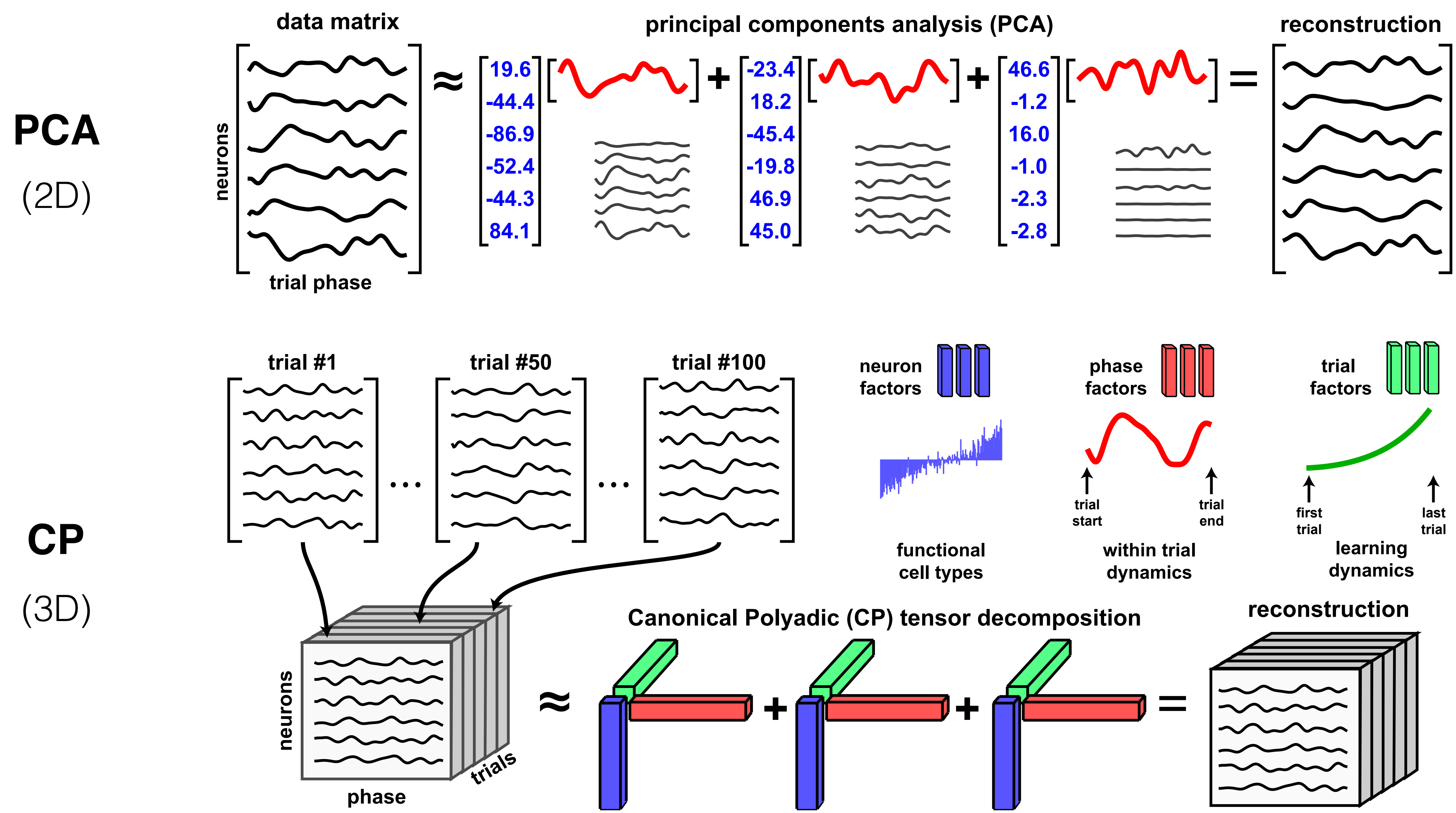
Matrix
Decomposition
(SVD)



Tensor
Decomposition
(CP)



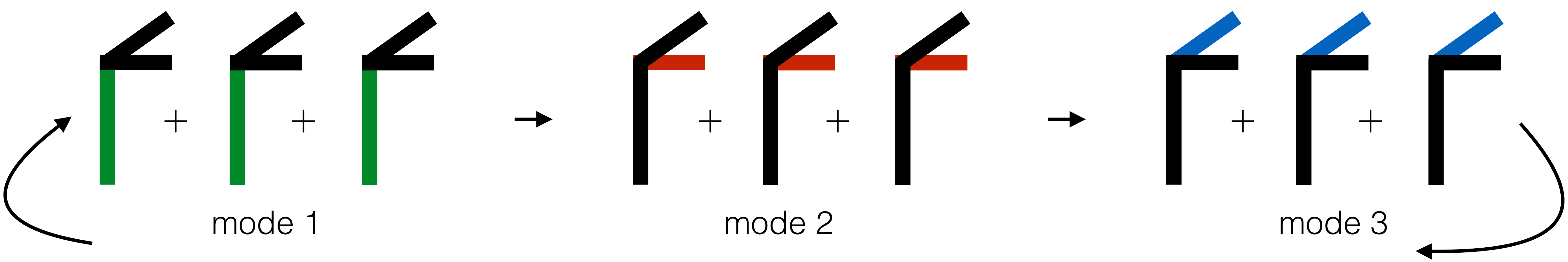
CP: Multi-Way Data Analysis



(source: Alex Williams, Stanford/Sandia)

Standard Method

Alternating Least Squares (CP-ALS)



Solve a sequence of least squares problems, one for each mode, repeat

$\mathbf{A}^{(1)}$ $\mathbf{A}^{(2)}$ $\mathbf{A}^{(3)}$

Our Method

Randomized Least Squares

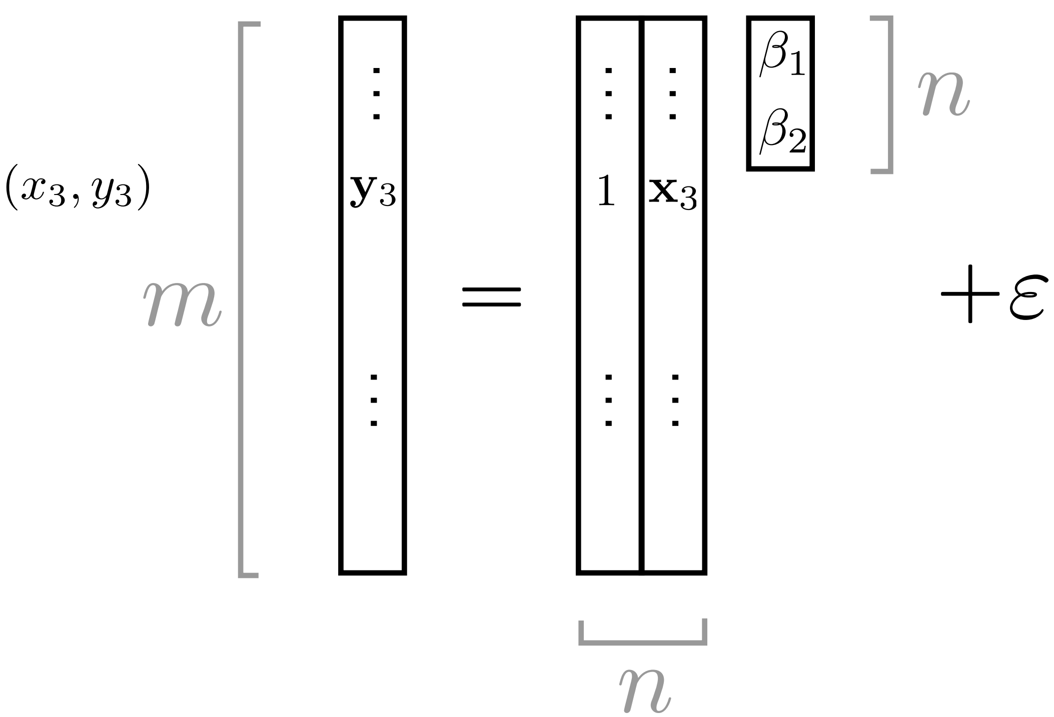
or “sketching”

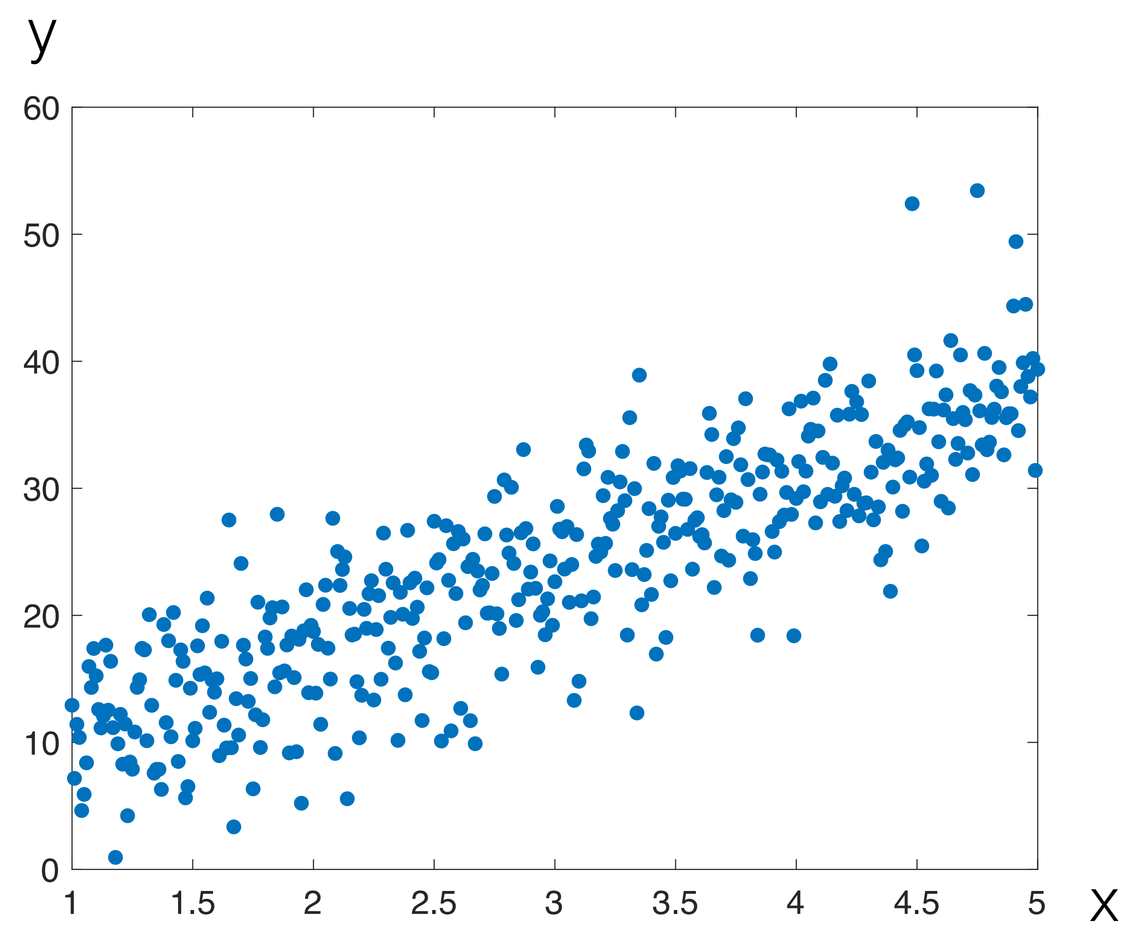
Contributions

- We show how randomized sketching techniques can extend from matrices to general dense tensors
- We demonstrate a novel randomized least squares algorithm for the CP decomposition, with a novel stopping condition
- This enables us to scale to much larger data sets
- In addition to speed, there is evidence that this randomization increases robustness!

Example: Linear Regression

$$\mathbf{b} = \mathbf{A}\beta + \varepsilon$$



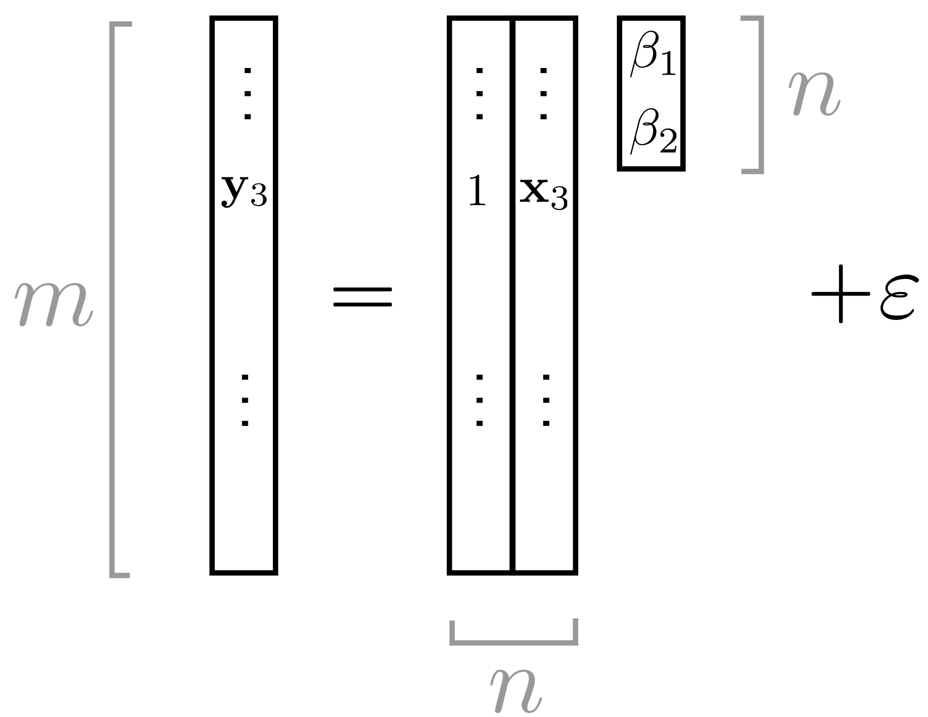


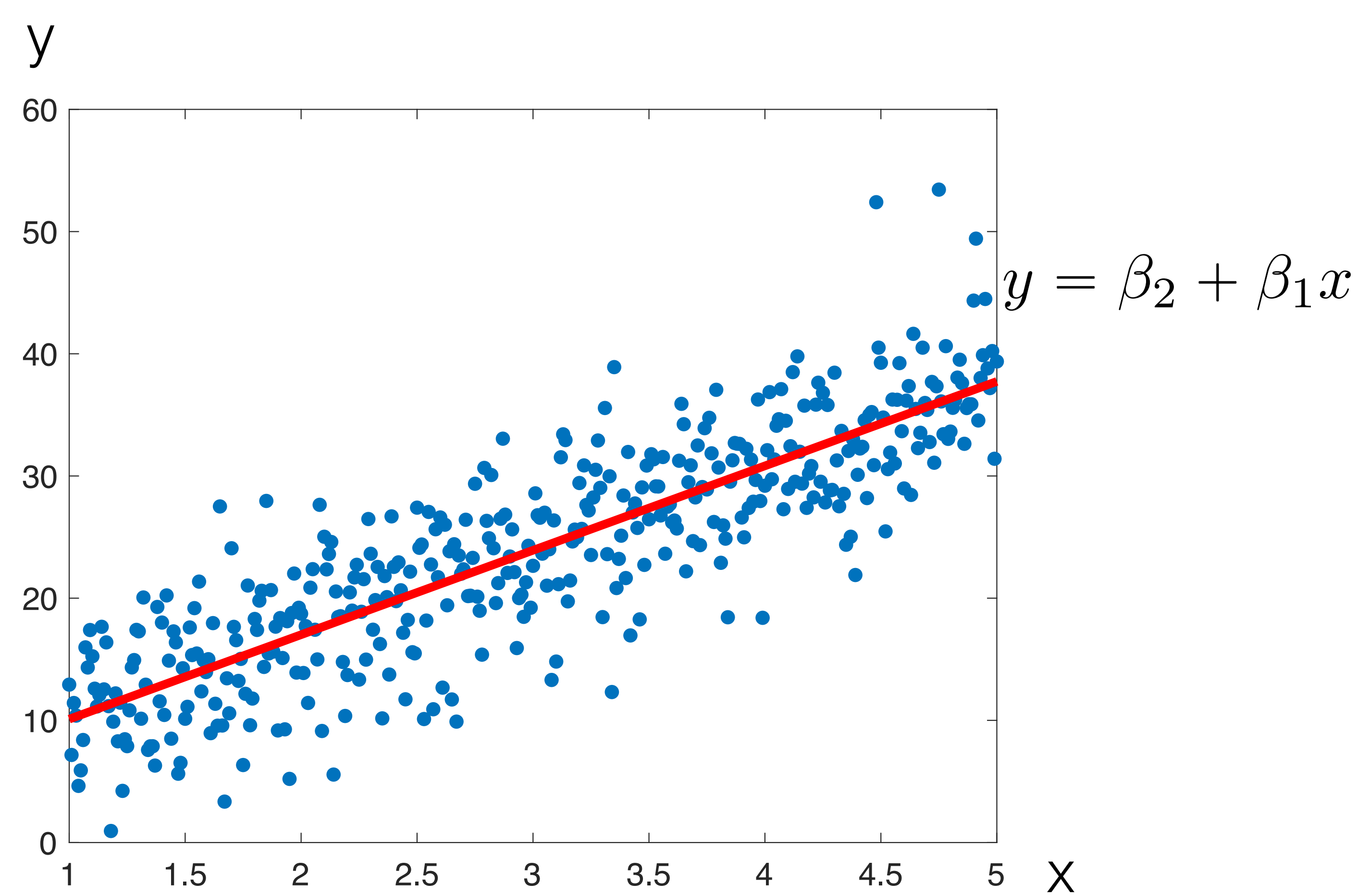
Find line $\beta_2 x + \beta_1$ satisfying:

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

Example: Linear Regression

$$\mathbf{b} = \mathbf{A}\beta + \varepsilon$$





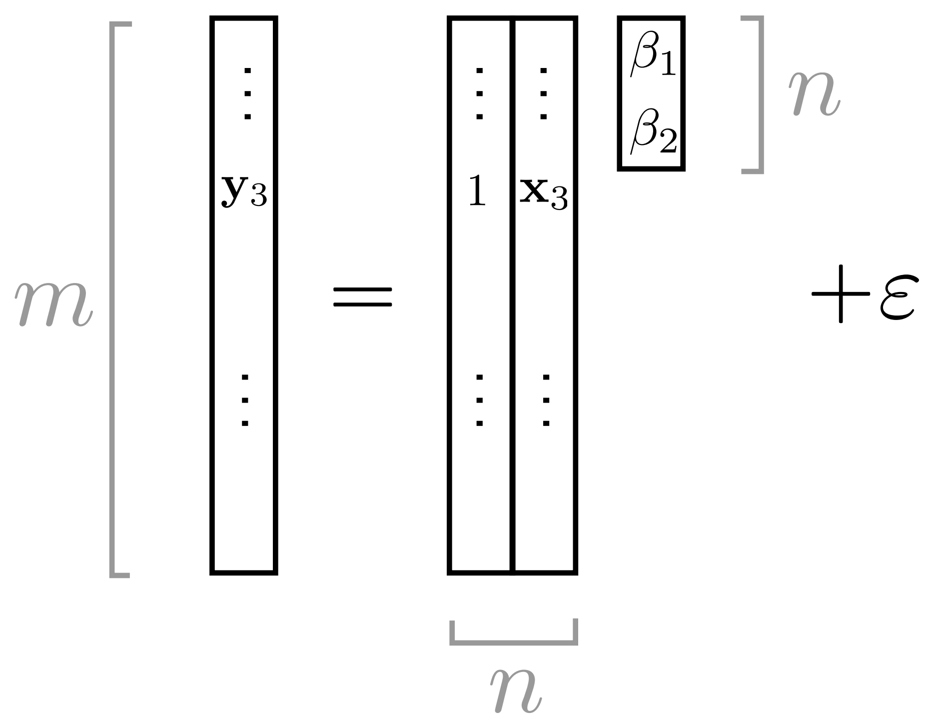
Method of Normal Equations
Solves for β:

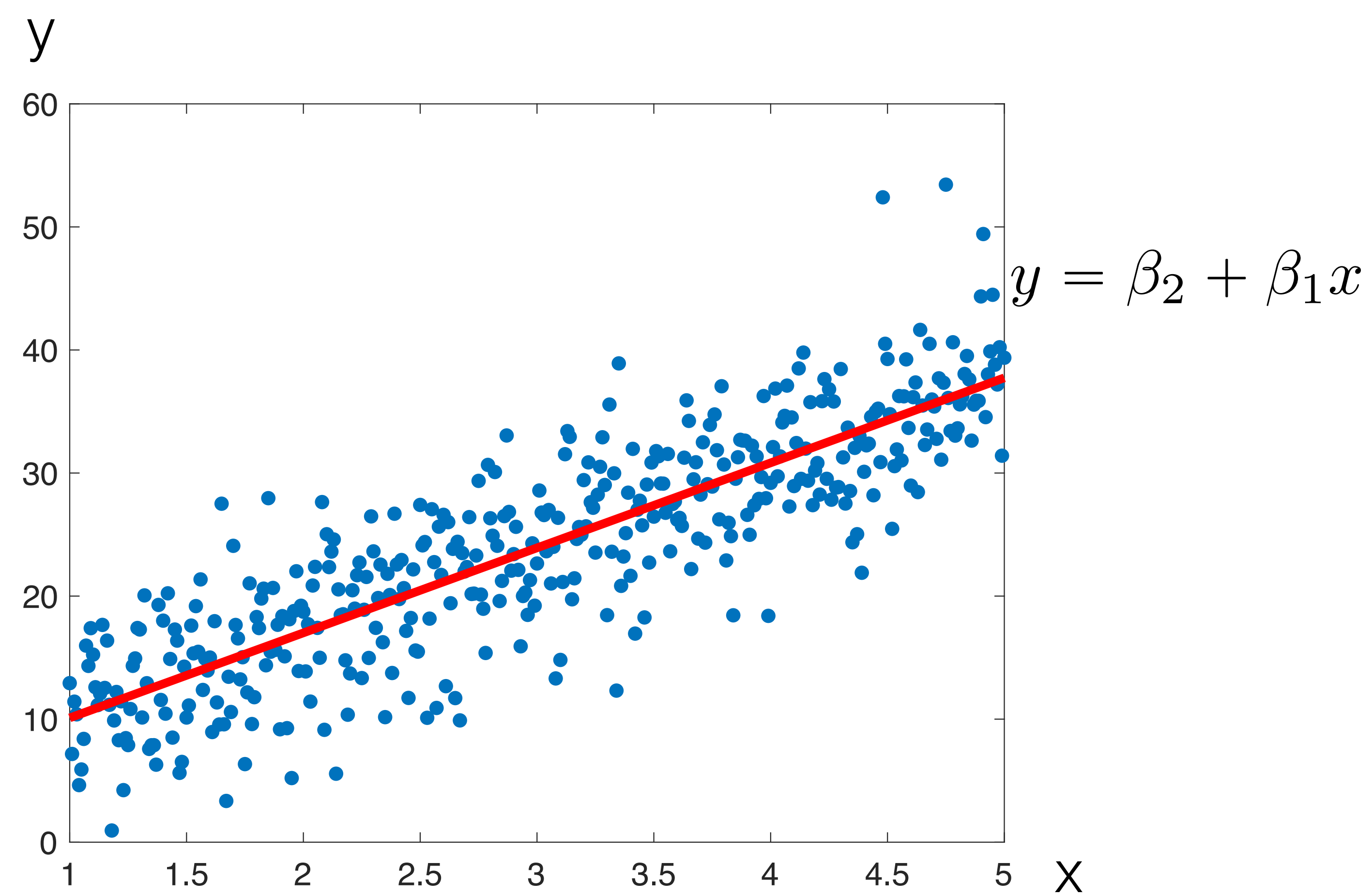
$$\min_{\beta} \|\mathbf{A}\beta - \mathbf{b}\| \rightarrow (\mathbf{A}^{\mathsf{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{A}\beta = (\mathbf{A}^{\mathsf{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{b}$$

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

Example: Linear Regression

$$\mathbf{b} = \mathbf{A}\beta + \varepsilon$$





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

In MATLAB:

$$\beta \leftarrow \mathbf{A}\backslash\mathbf{b}$$

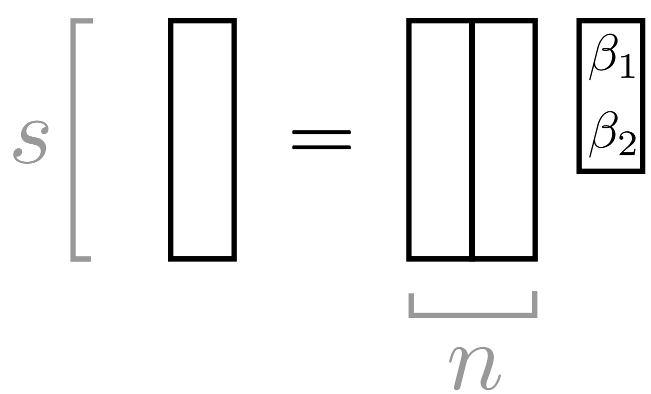
Sampling Linear Regression

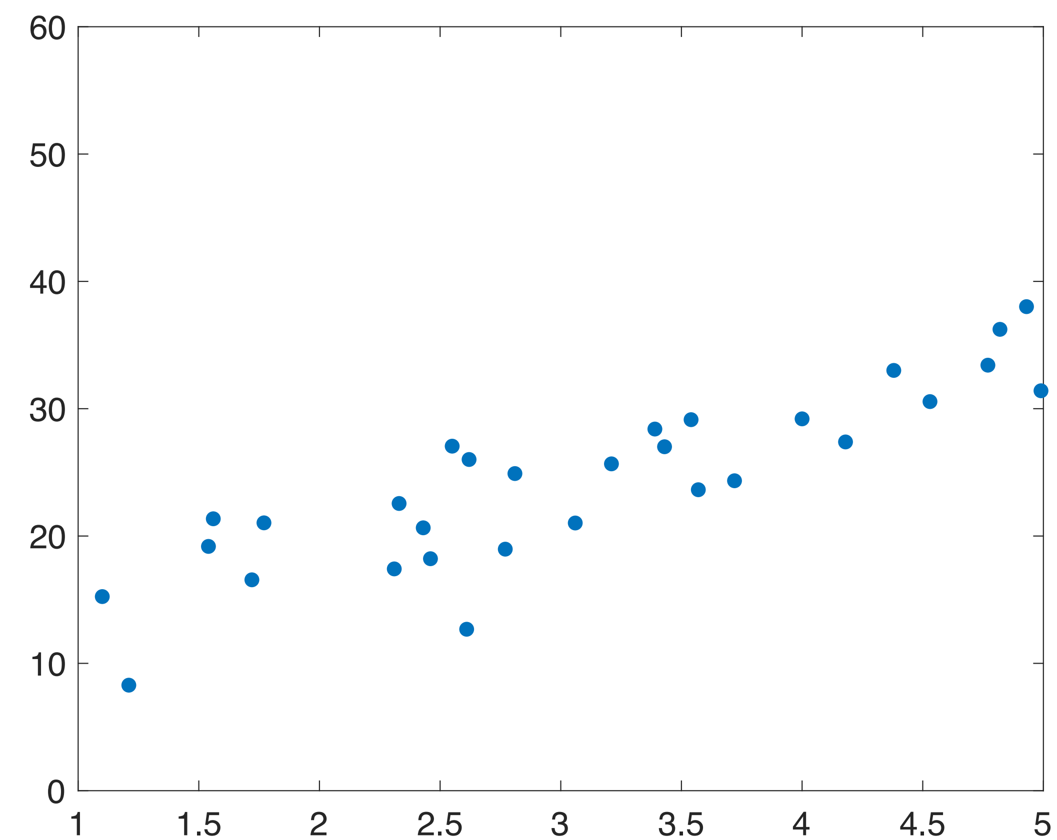
What if we sample points?

(Uniform, Random)

row-sampling
operator

$$\mathbf{Sb} = \mathbf{SA}\beta$$





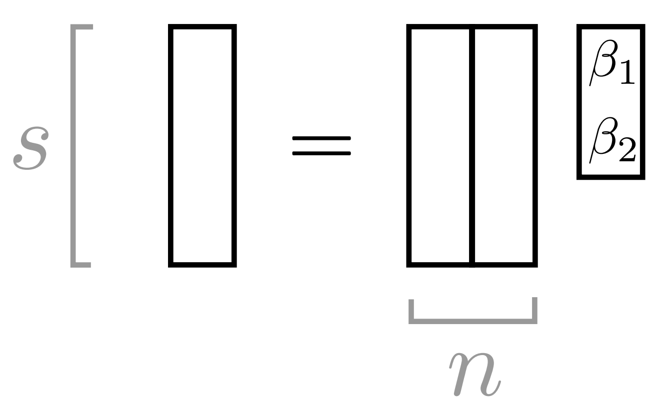
Sampling Linear Regression

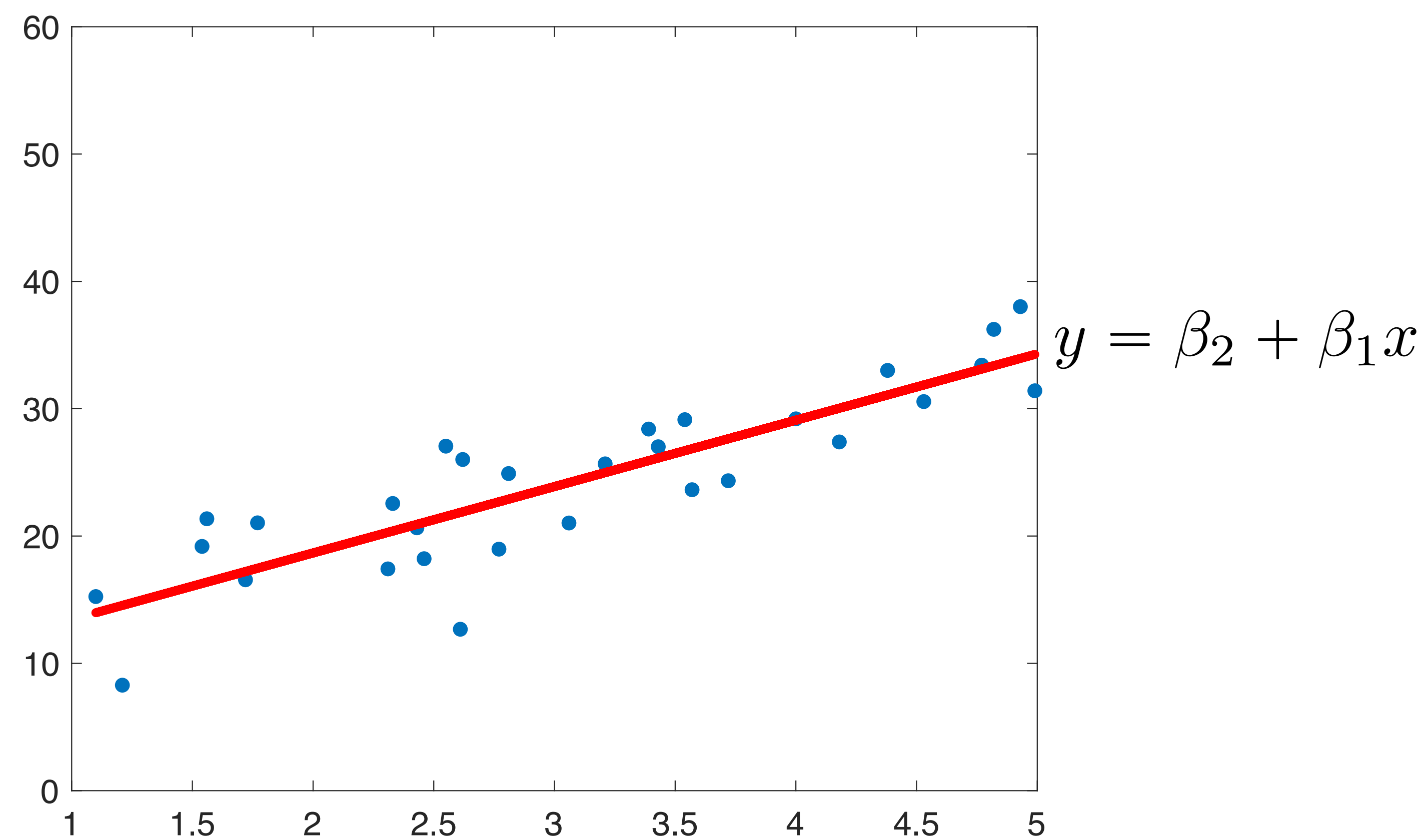
Sample A and b (MATLAB):

```
>> srows = randi(m,s,1); %sample s rows w. replacement
>> As = A(srows, :);
>> bs = b(srows);
>> x = A \ b;
>> xs = As \ bs;
```

row-sampling operator

$$\mathbf{Sb} = \mathbf{SA}\beta$$



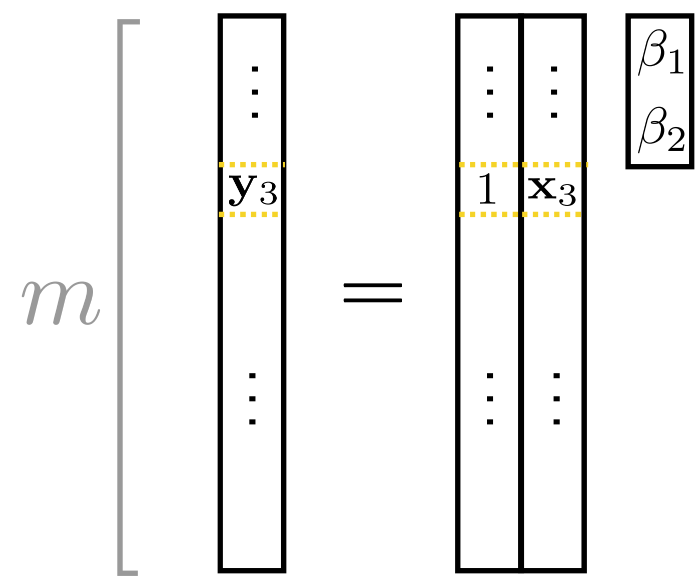


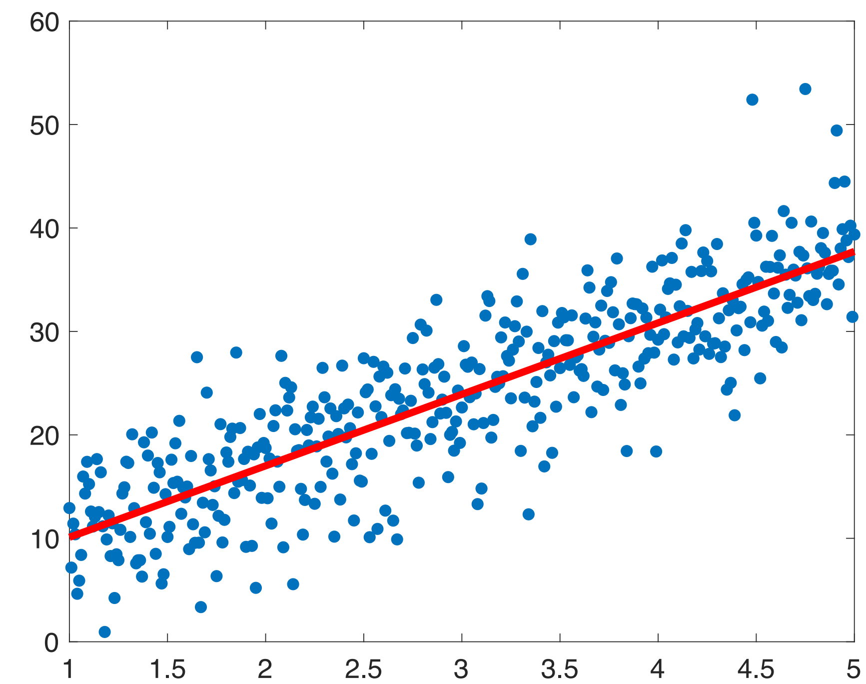
Sampling Linear Regression

Chol: $O(mn^2 + n^3)$ flops
QR: $O(mn^2)$ flops

$m \gg n$

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



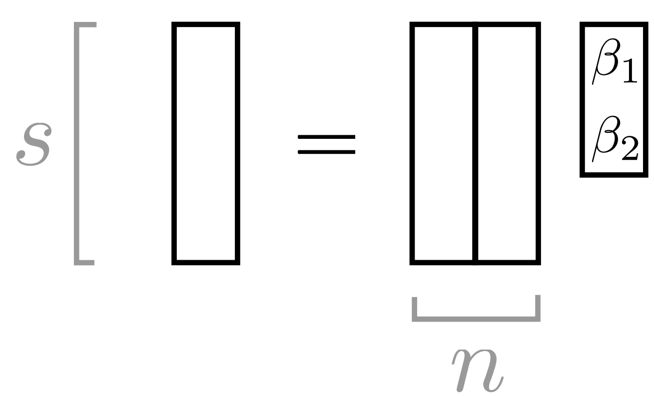


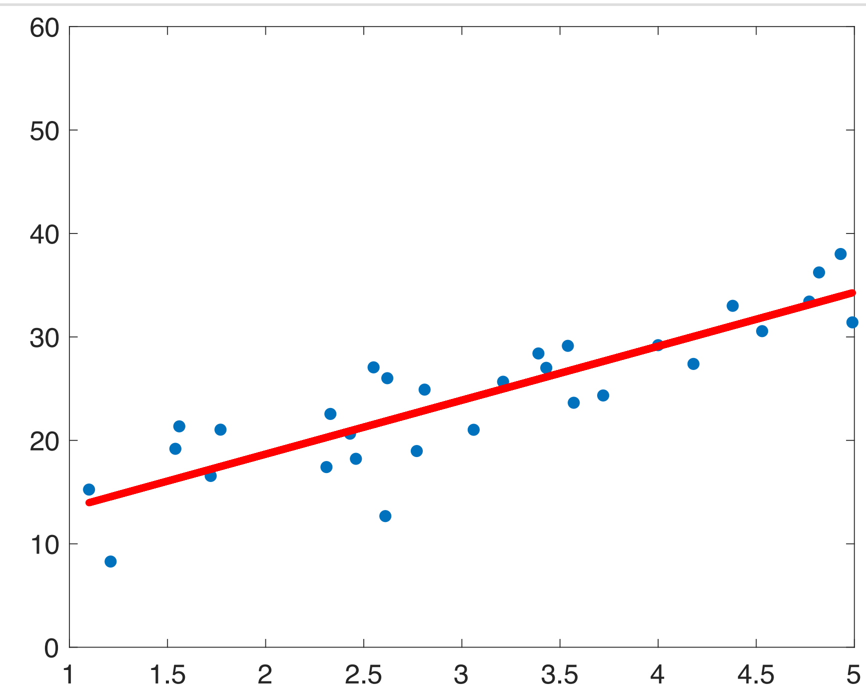
$y = \beta_2 + \beta_1 x$

Chol: $O(sn^2 + n^3)$ flops
QR: $O(sn^2)$ flops

$m \gg s > n$

$$\mathbf{Sb} = \mathbf{SA}\beta$$

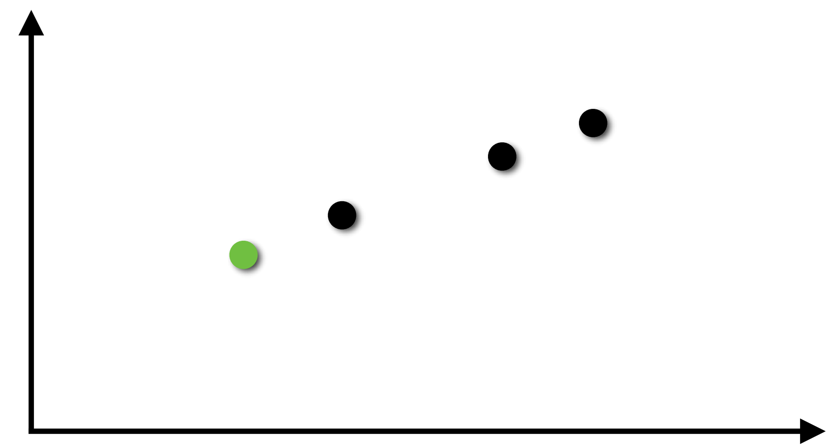




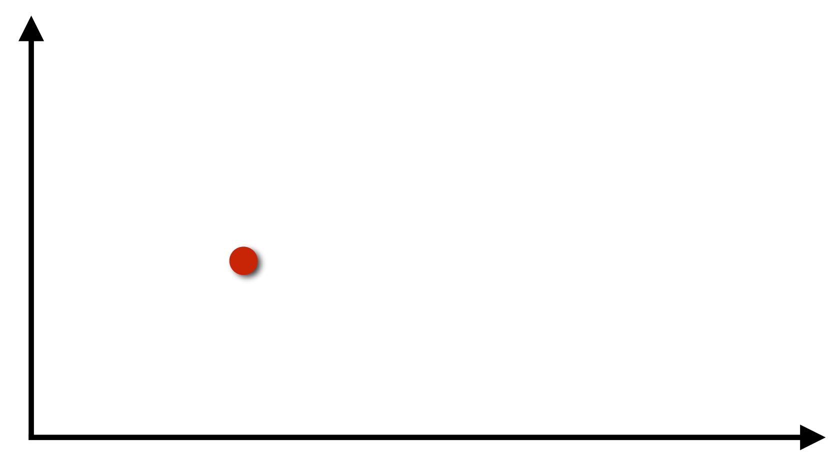
$y = \hat{\beta}_2 + \hat{\beta}_1 x$

Danger: Undetermined Sampling

$$\begin{bmatrix}5\\6\\8\\5\\9\end{bmatrix} = \overset{\mathbf{A}}{\begin{bmatrix}1 & 4\\1 & 5\\1 & 7\\1 & 4\\1 & 8\end{bmatrix}} \begin{bmatrix}\beta_1\\\beta_2\end{bmatrix}$$

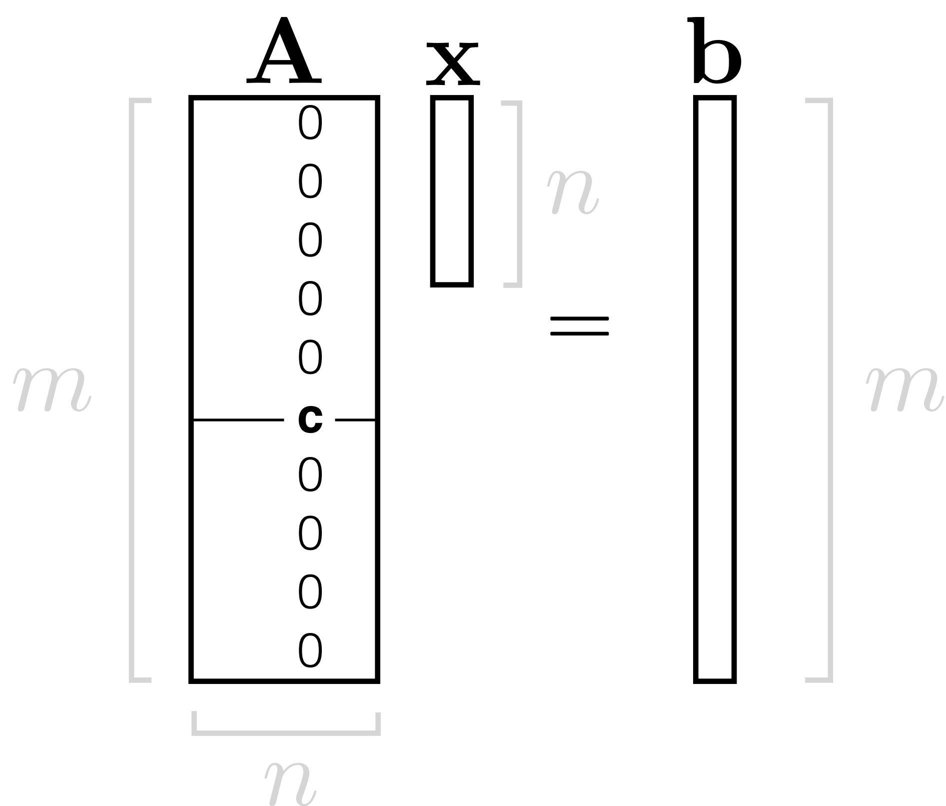


$$\begin{bmatrix}5\\5\end{bmatrix} = \overset{\mathbf{SA}}{\begin{bmatrix}1 & 4\\1 & 4\end{bmatrix}} \begin{bmatrix}\beta_1\\\beta_2\end{bmatrix}$$



$\mathbf{SA}$ may be rank-deficient —> underdetermined!

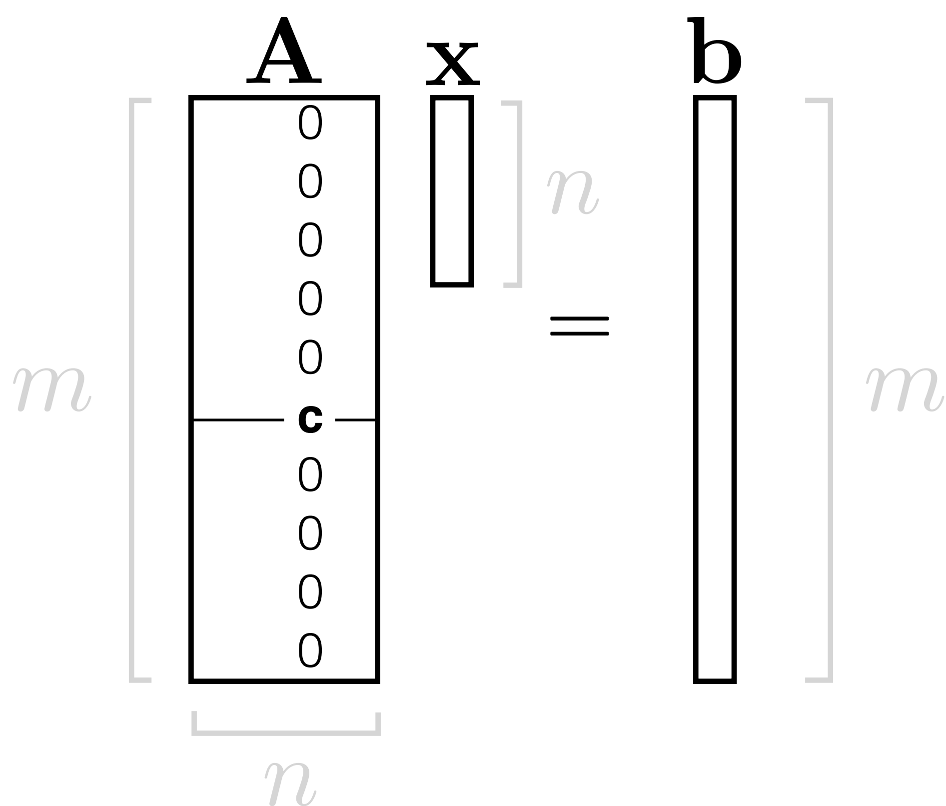
Underdetermined Sampling

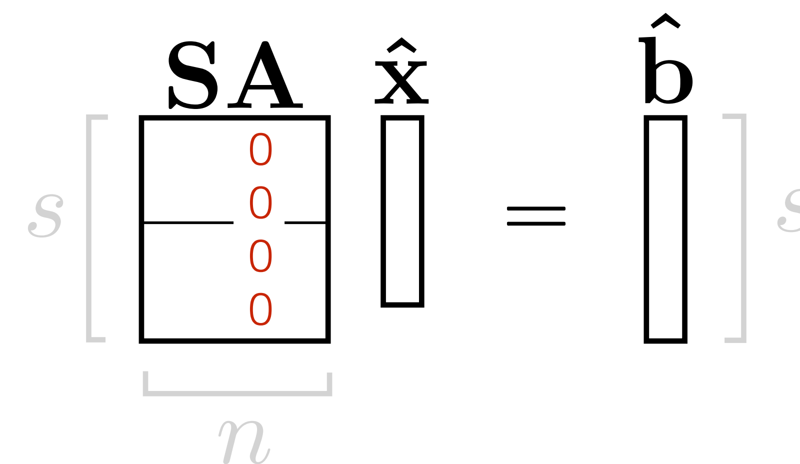


Given full-rank **A**:

Consider a row containing the only nonzero in its column

Underdetermined Sampling





Any sampling must contain that row <u>or</u> be rank-deficient.

Given full-rank **A**:

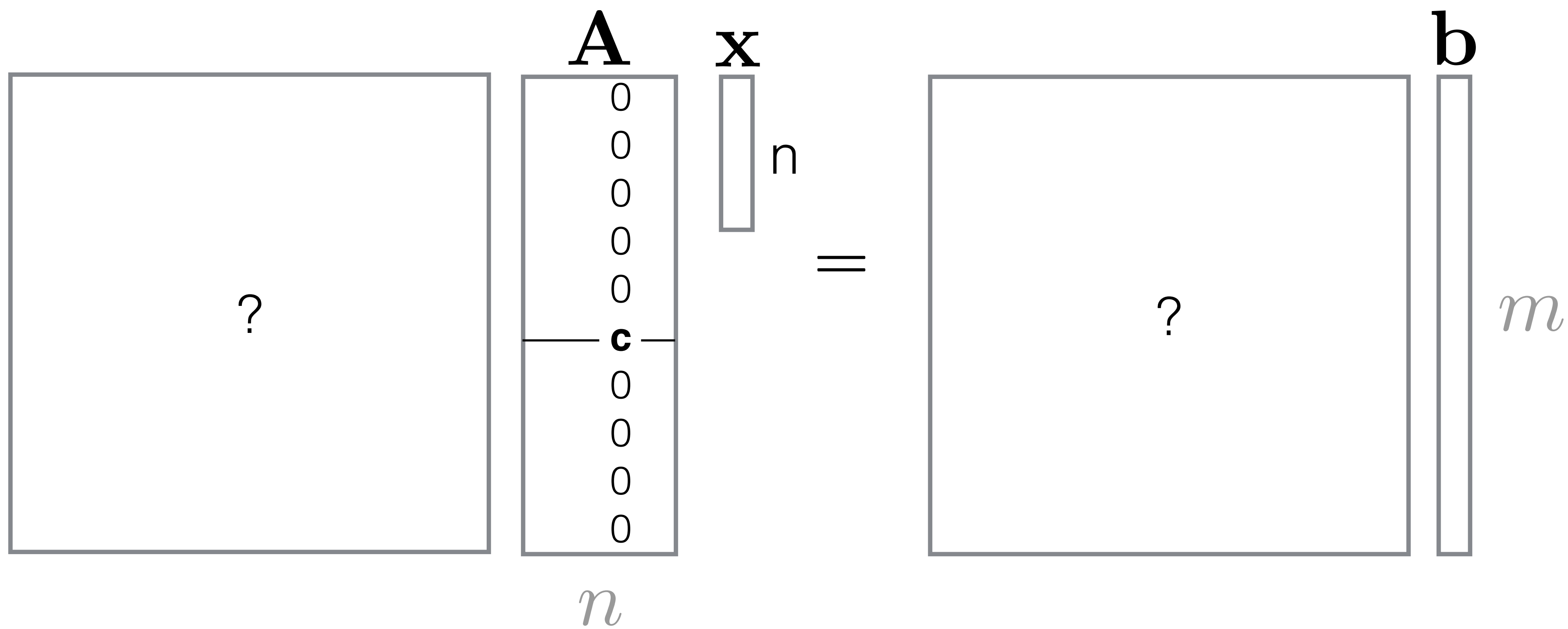
Consider a row containing the only nonzero in its column

Uniform sampling w/ replacement:
s = O($m\log m$) samples needed

(so $s > m$; sampling is useless here!)

(Avron, Maymounkov & Toledo 2010)

Mixing

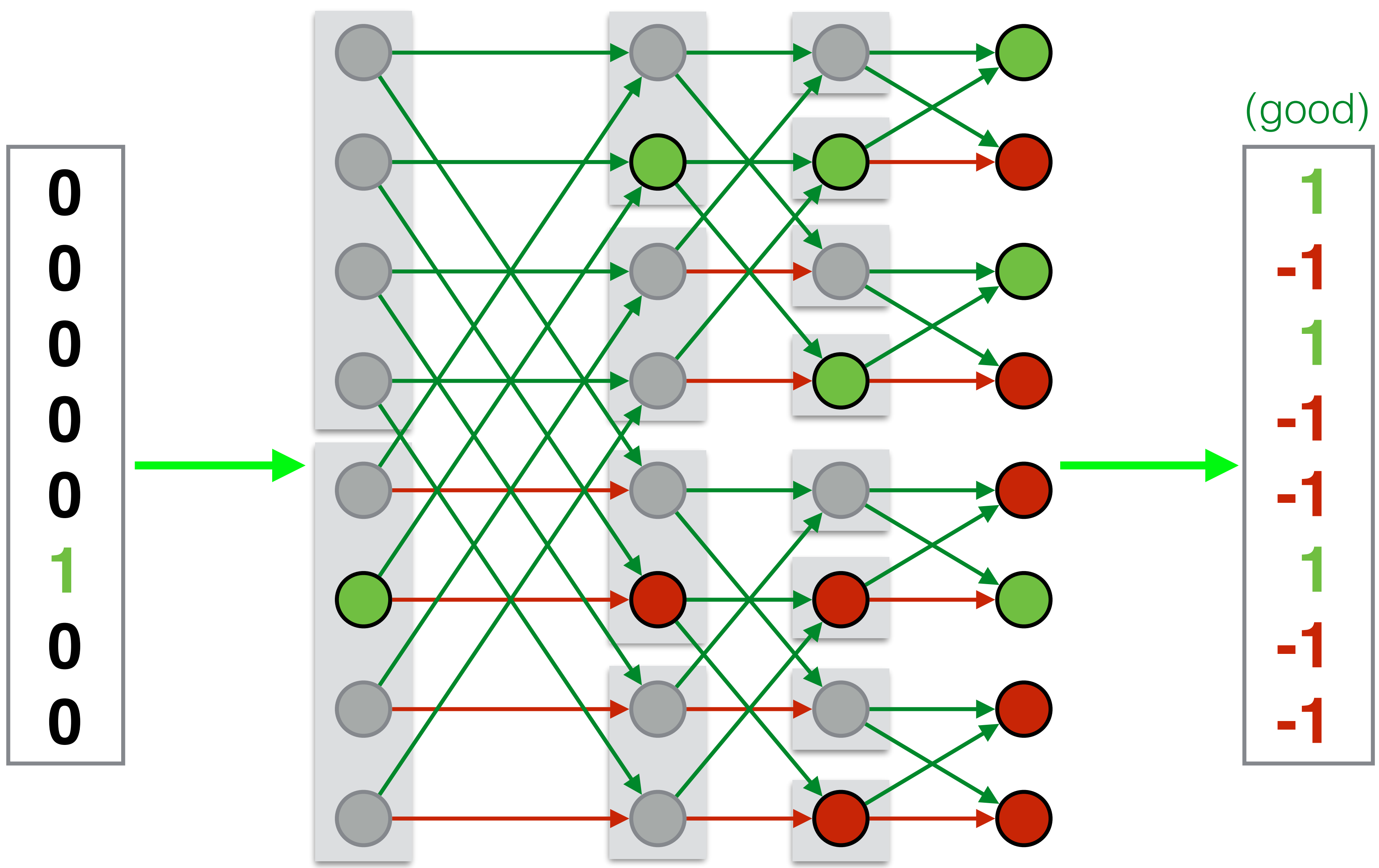


Idea: Transform **A**,**b** before sampling

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

Mixing

Trick: mix up the rows

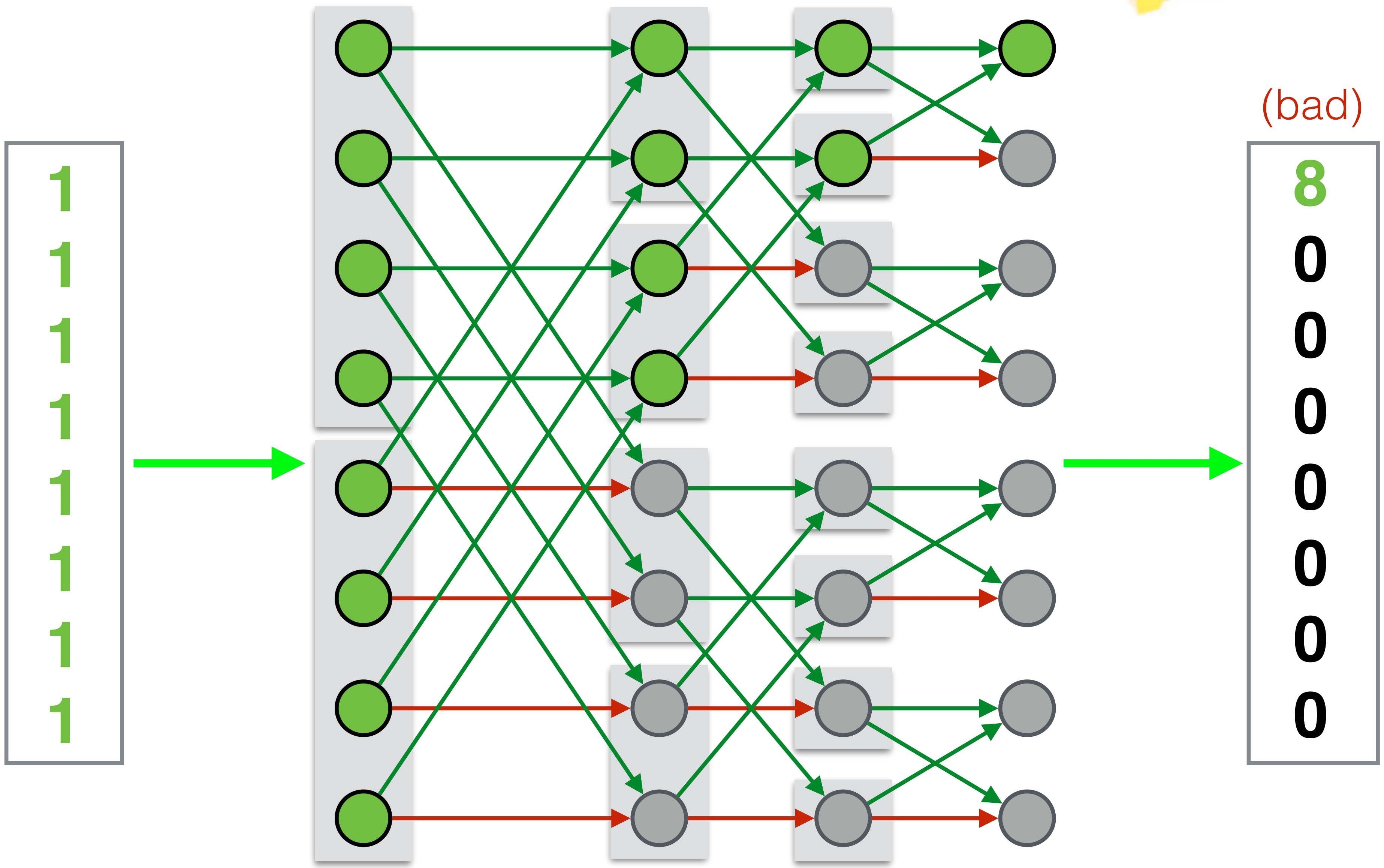


Mixing

Trick: mix up the rows

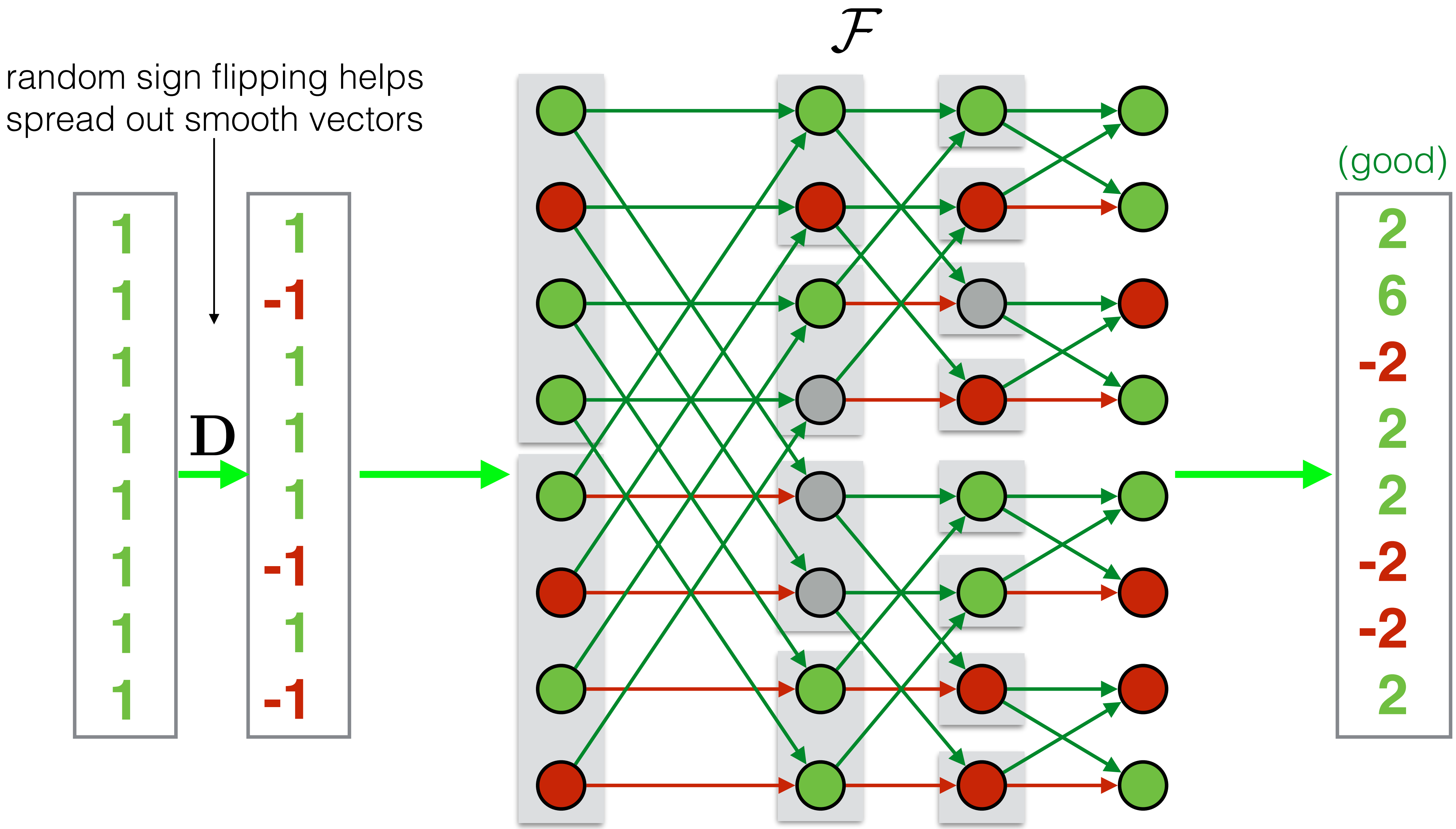
problem with vectors that are sparse in the frequency domain…

$\mathcal{F}$



Mixing

Trick: mix up the rows



Mixing

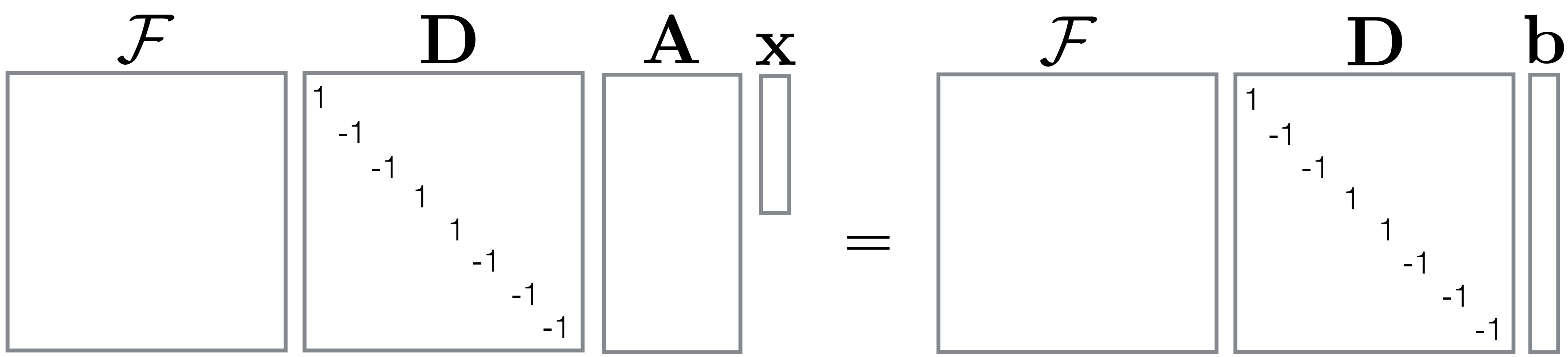
Trick: mix up the rows

"Fast Johnson-Lindenstrauss Transform"
(Ailon & Chazelle, 2006)

Randomly sign-flip rows, then perform one of:

- **FFT**
- **WHT**
- **DHT**

$\Big] \; \mathcal{F}$



Orthogonal transformations; Change of basis

Mixing

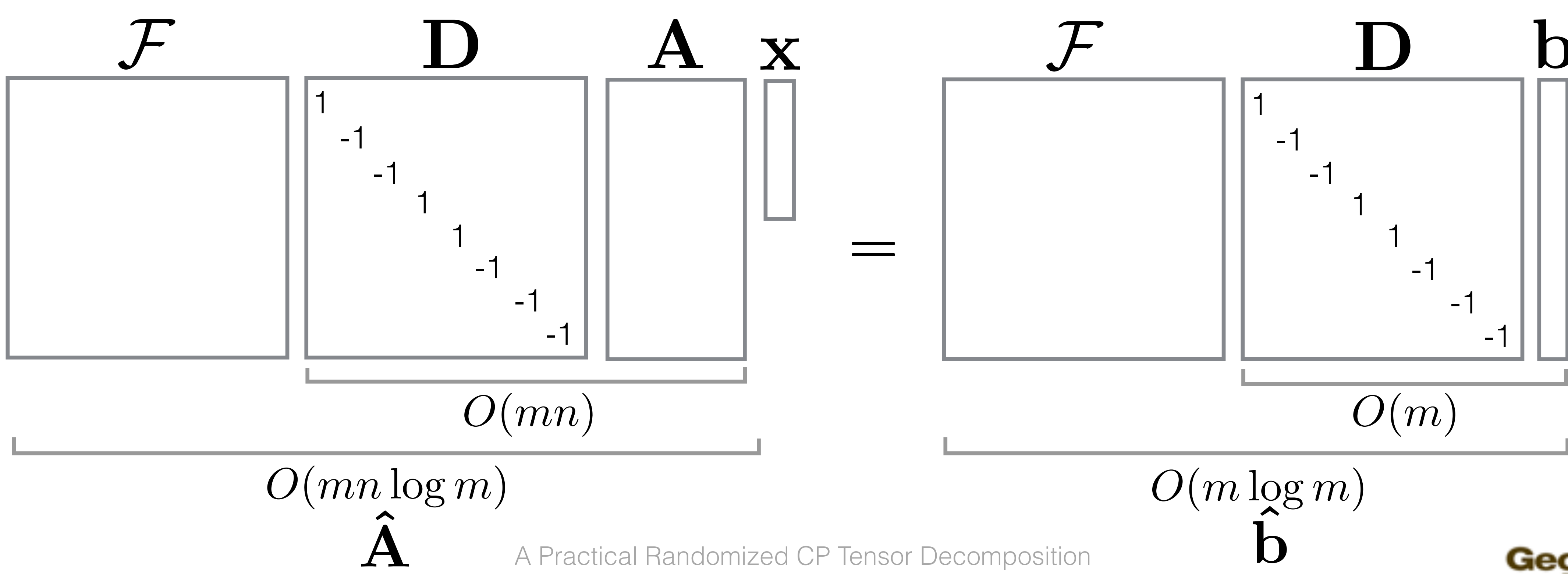
Trick: mix up the rows

"Fast Johnson-Lindenstrauss Transform"
(Ailon & Chazelle, 2006)

Randomly sign-flip rows, then perform one of:

- **FFT**
- **WHT**
- **DHT**

$\mathcal{F}$



Mixing

Trick: mix up the rows

"Fast Johnson-Lindenstrauss Transform"
(Ailon & Chazelle, 2006)

Randomly sign-flip rows, then perform one of:

- **FFT**
- **WHT**
- **DHT**

$\mathcal{F}$

(other options include DCT, random Givens rotations, random orthogonal matrix)

$$\mathbf{D} = \begin{bmatrix} 1 &&&&&&& \\ & -1 &&&&&& \\ && -1 &&&&& \\ &&& 1 &&&& \\ &&&& 1 &&& \\ &&&&& -1 && \\ &&&&&& -1 & \\ &&&&&&& -1 \end{bmatrix}$$

$$\underbrace{\mathcal{F}\,\underbrace{\mathbf{D}\,\mathbf{A}}_{O(mn)}}_{\substack{O(mn\log m) \\ \hat{\mathbf{A}}}}\,\mathbf{x} = \underbrace{\mathcal{F}\,\underbrace{\mathbf{D}\,\mathbf{b}}_{O(m)}}_{\substack{O(m\log m) \\ \hat{\mathbf{b}}}}$$

Mixing

Trick: mix up the rows

"Fast Johnson-Lindenstrauss Transform"
(Ailon & Chazelle, 2006)

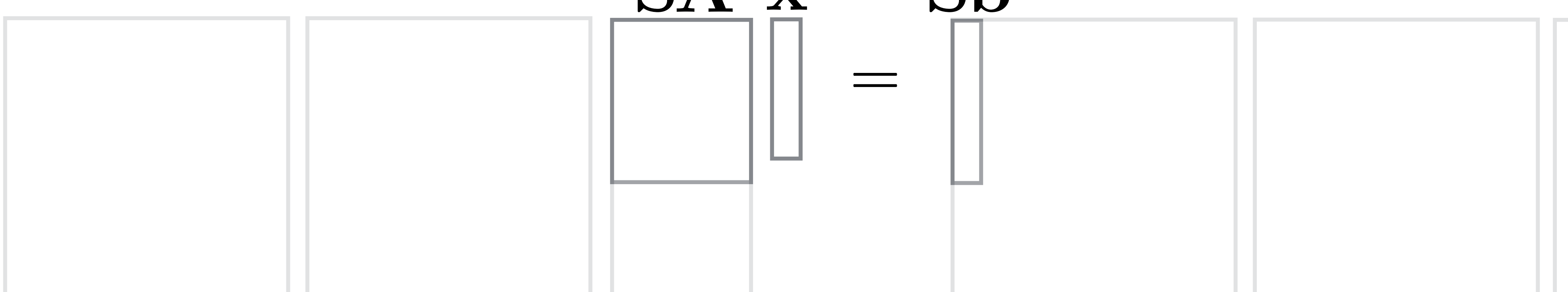
Randomly sign-flip rows, then perform one of:

- **FFT**
- **WHT**
- **DHT**

$\mathcal{F}$

(other options include DCT, random Givens rotations, random orthogonal matrix)

$$\mathbf{S}\hat{\mathbf{A}}\,\mathbf{x} = \mathbf{S}\hat{\mathbf{b}}$$



We should then be able to sample a small number of rows …

Coherence

Given a matrix **Q** whose columns are an orthonormal basis for **A:**

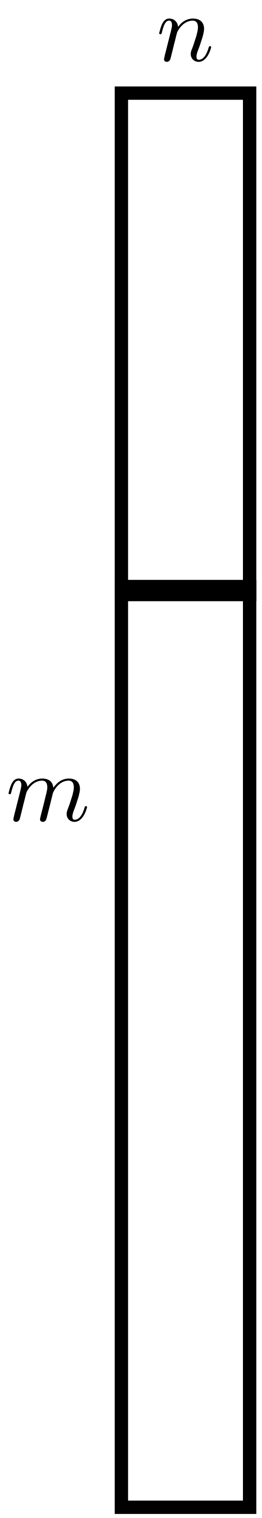
e.g. $[\mathbf{Q}, \mathbf{R}] = \mathrm{QR}(\mathbf{A})$

Coherence is the maximum leverage score:

$$\mu(\mathbf{A}) = \max_i \|\mathbf{Q}_{i,*}\|_2^2$$

(measuring correlation with the standard basis)

e.g. Well-mixed matrix $\quad \underset{\text{incoherent}}{\frac{n}{m}} \leqslant \mu(\mathbf{A}) \leqslant \underset{\text{coherent}}{1} \quad$ e.g. Identity matrix



#Samples

(to recover a full-rank preconditioner)

$$\frac{n}{m} \leqslant \mu(\mathbf{A}) \leqslant 1$$

incoherent — coherent

$$\mu(\mathbf{A}) = \frac{n}{m} \rightarrow s = O(n \log n)$$ **(tiny)**

$$\mu(\mathbf{A}) = 1 \rightarrow s = O(m \log m)$$ **(larger than input)**

(Avron, Maymounkov & Toledo 2010)

“Faster Approximate Least Squares”

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$$

mix

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

sample

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

solve exactly

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

use as approximate solution for original problem

(Drineas, Mahoney, Muthukrishnan, Sarlós 2007/11)

Sketching

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

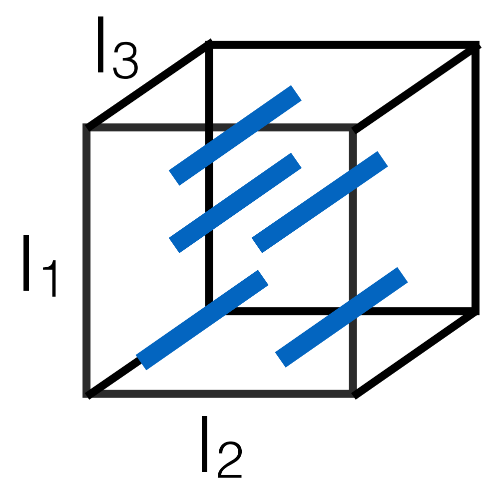
Has found practical uses, e.g. “Blendenpik”:

— Precondition using Randomized Least Squares
— Then apply a standard iterative method (LSQR)
— ~4X overall speedup over LAPACK

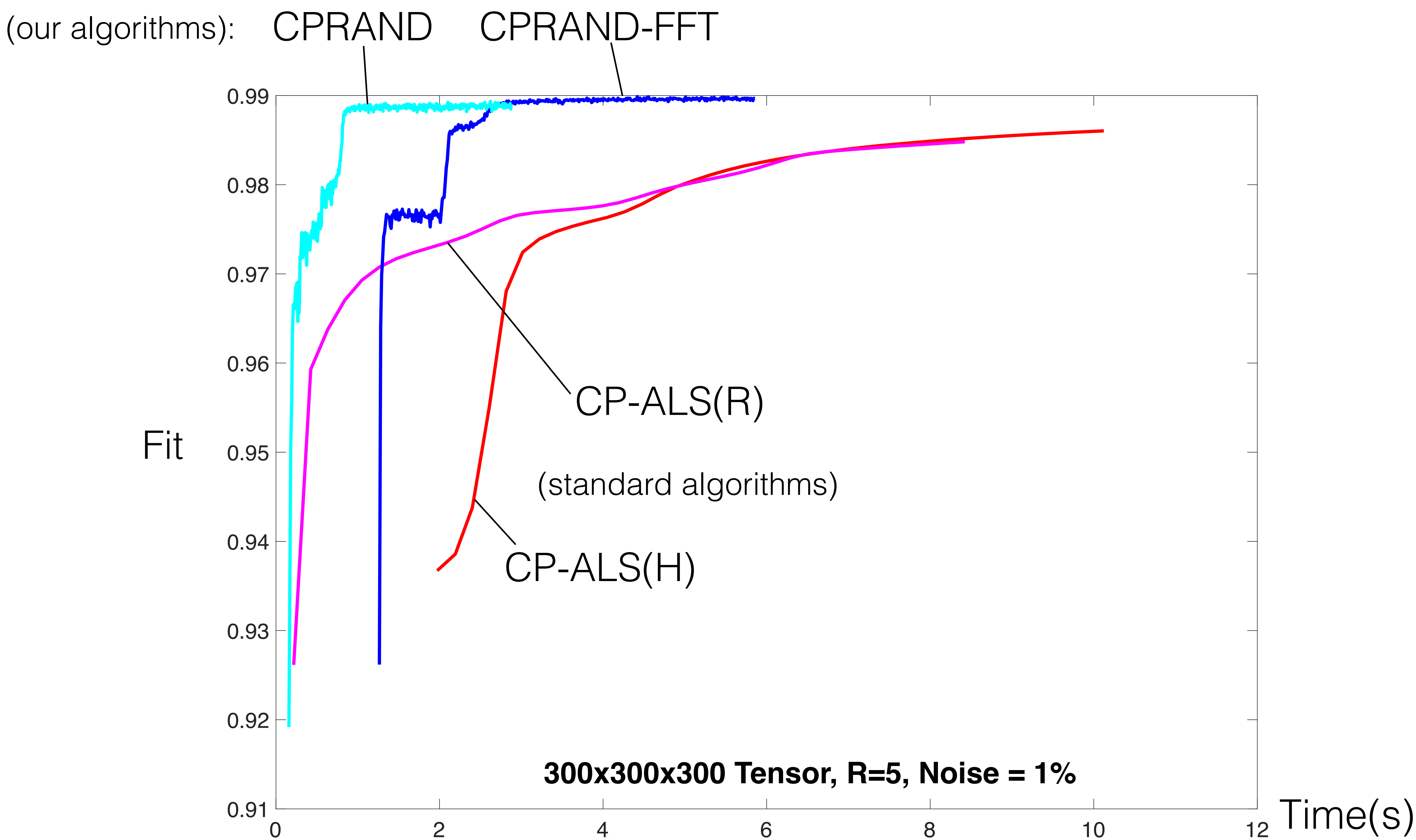
(Avron, Maymounkov & Toledo 2010)

Sketching

Can we apply this technique
to tensor applications?



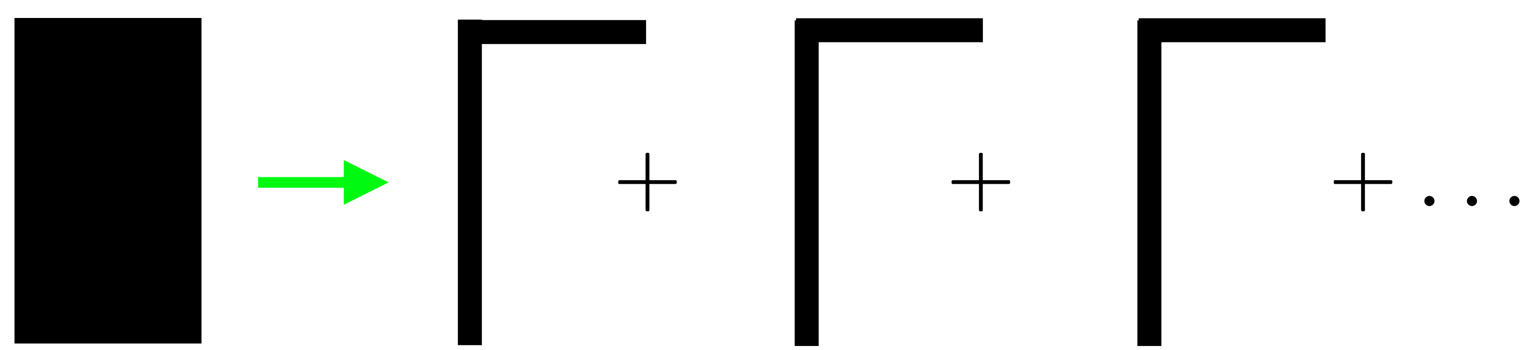
Yes


(our algorithms):
CPRAND
CPRAND-FFT
Fit
Time(s)
CP-ALS(R)
(standard algorithms)
CP-ALS(H)
300x300x300 Tensor, R=5, Noise = 1%
0.99
0.98
0.97
0.96
0.95
0.94
0.93
0.92
0.91
0
2
4
6
8
10
12

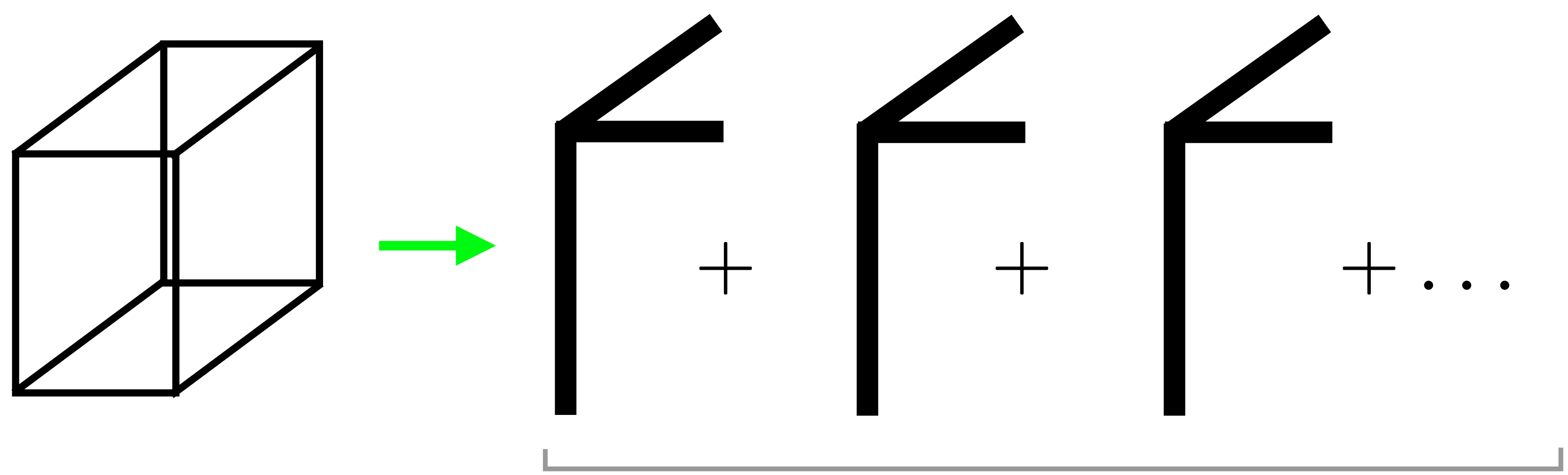
CP Decomposition

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

SVD:



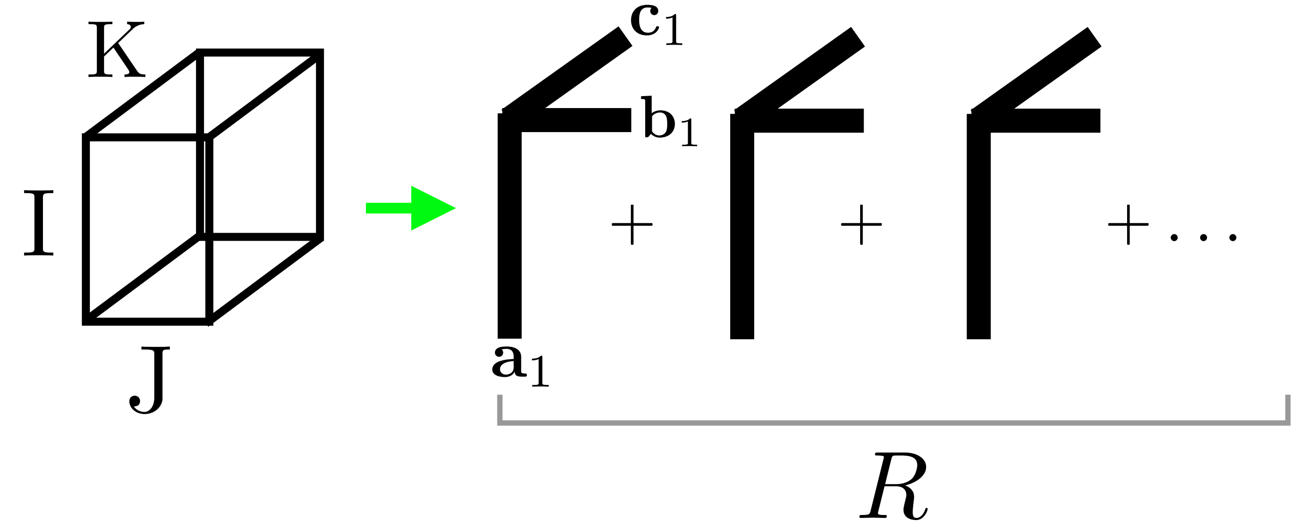
3D CP:



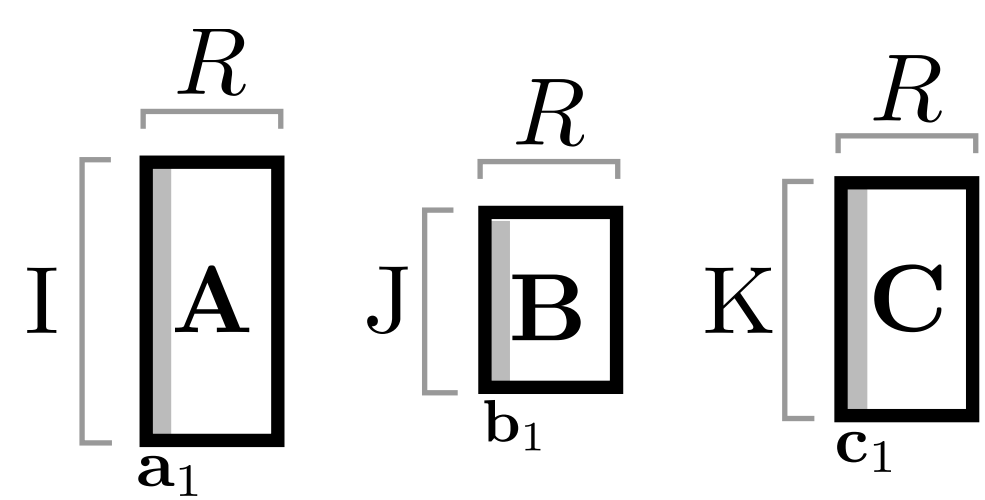
R … number of components

Deriving CP-ALS

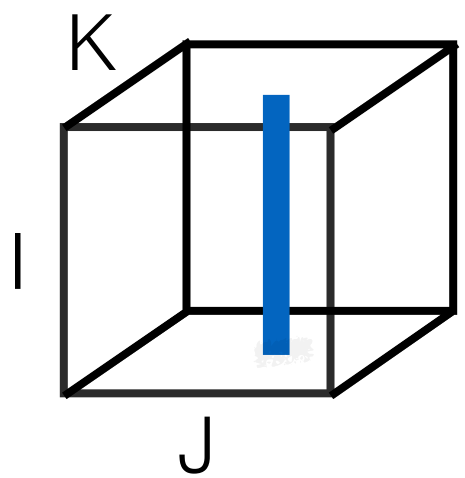
$$\mathcal{X} \approx \tilde{\mathcal{X}} = \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$$



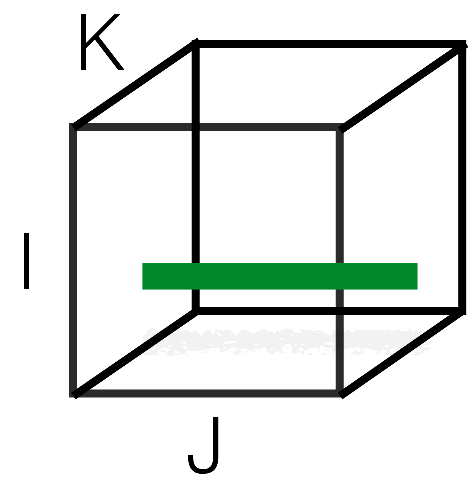
Store factors in matrix form:



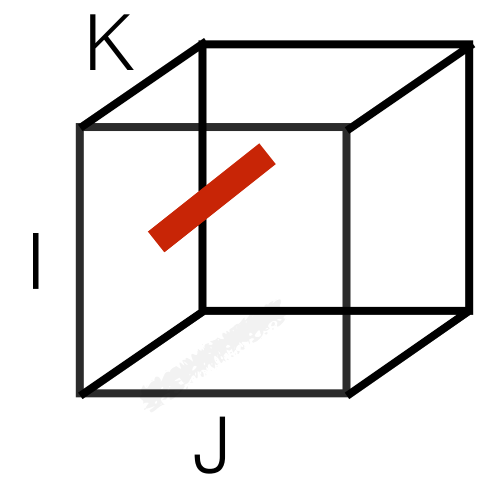
Tensor Fibers



mode-1 fibers



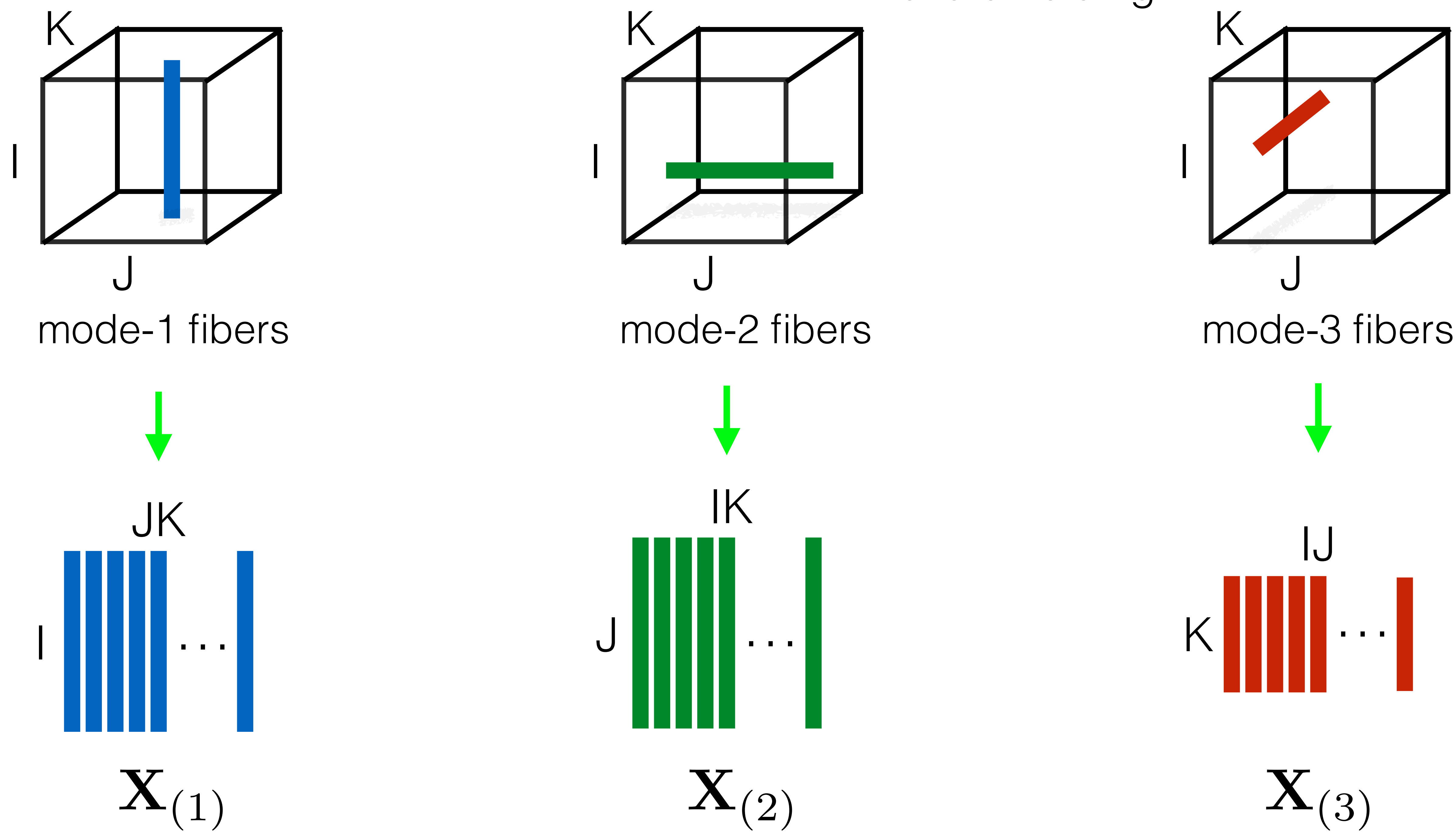
mode-2 fibers



mode-3 fibers

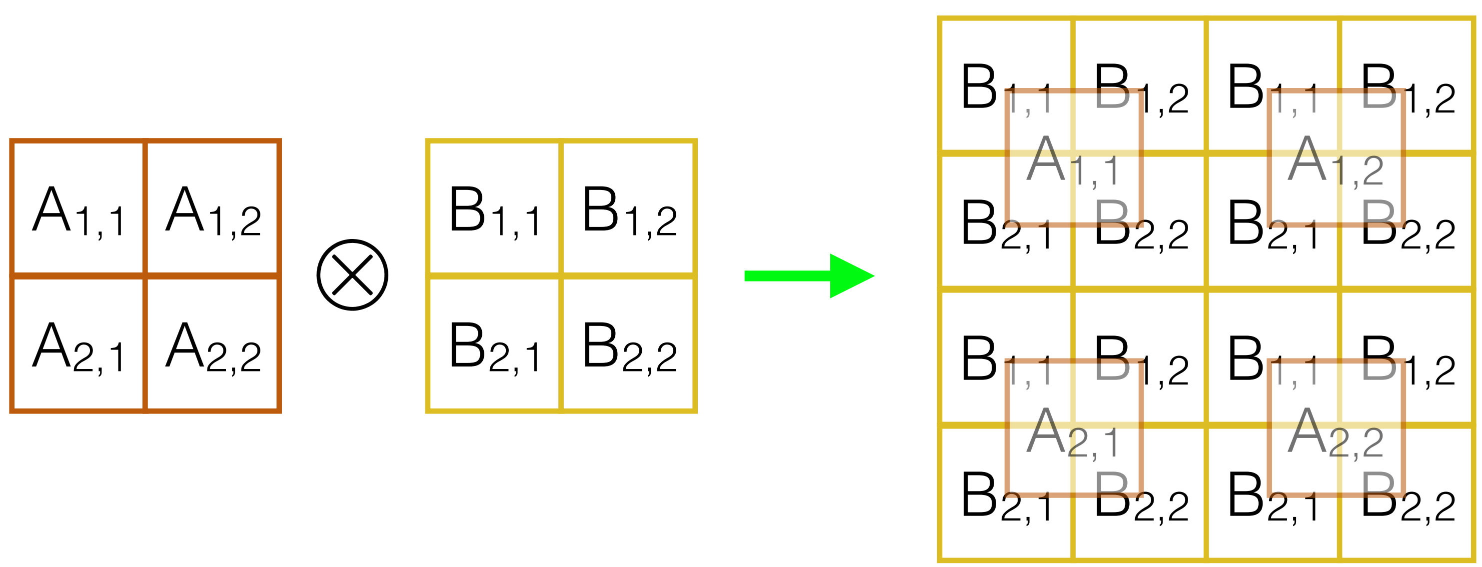
Matricization

aka unfolding



Kronecker Product

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



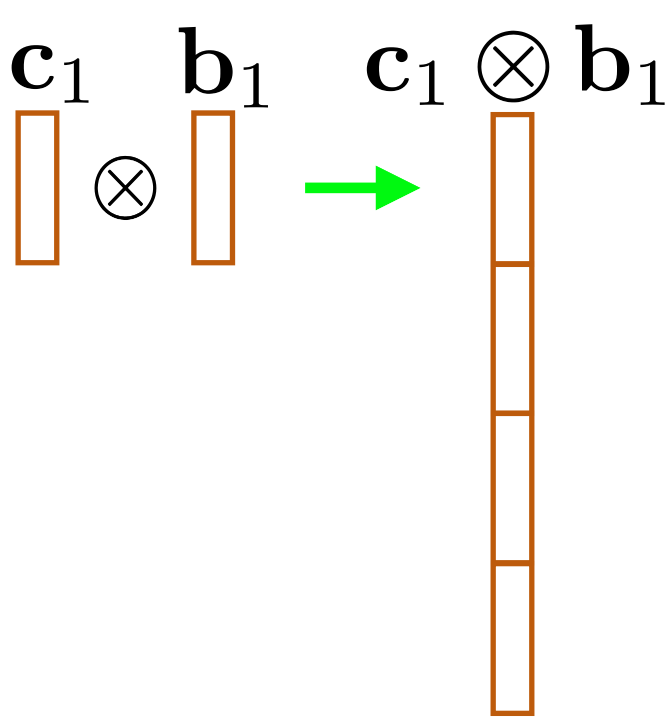
every pair of scalar entries multiplied, in a block structure

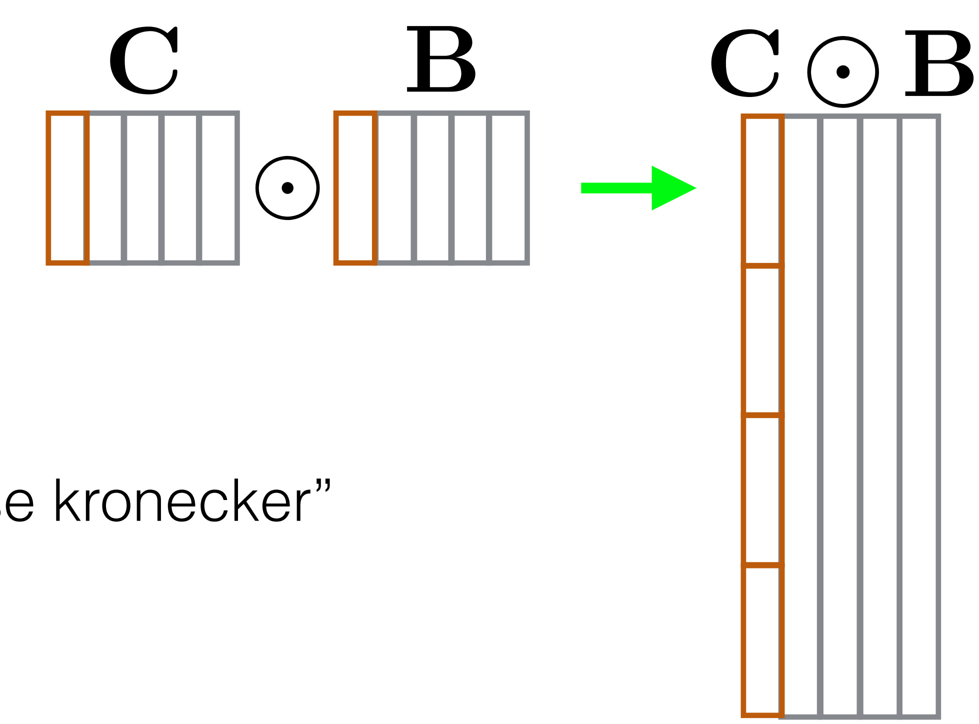
CP Decomposition & Khatri-Rao Product

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

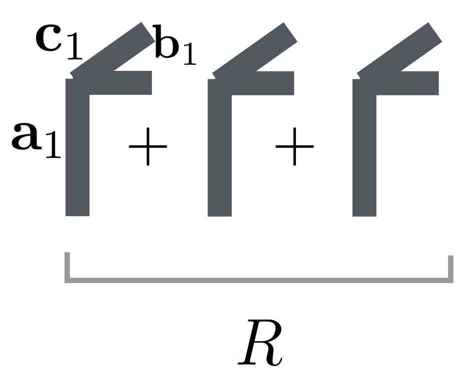
$$\mathbf{A} \odot \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1, \mathbf{a}_2 \otimes \mathbf{b}_2, \ldots, \mathbf{a}_K \otimes \mathbf{b}_K]$$



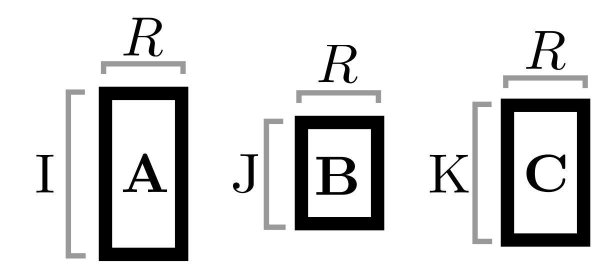


"columnwise kronecker"

CP Decomposition



(order 3, extends to order *N*)



Matricized form:

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

Allows us to express
the least squares problem:

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

unknown

coefficient
matrix

Alternating Least Squares

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

unknown

coefficient matrix

Solve for each factor **A**, **B**, **C** in alternating fashion:

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

(Transpose least squares, so forward-slash)

Alternating Least Squares

repeat

order 3:

$$\mathbf{A} \leftarrow \mathbf{X}_{(1)} \quad / \quad (\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$$
$$\mathbf{B} \leftarrow \mathbf{X}_{(2)} \quad / \quad (\mathbf{C} \odot \mathbf{A})^{\mathsf{T}}$$
$$\mathbf{C} \leftarrow \mathbf{X}_{(3)} \quad / \quad (\mathbf{B} \odot \mathbf{A})^{\mathsf{T}}$$

$$\mathbf{A} \leftarrow \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B}) \quad / \quad \overbrace{(\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{B}^{\mathsf{T}}\mathbf{B})}^{R \times R}$$
$$\mathbf{B} \leftarrow \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A}) \quad / \quad (\mathbf{C}^{\mathsf{T}}\mathbf{C} * \mathbf{A}^{\mathsf{T}}\mathbf{A})$$
$$\mathbf{C} \leftarrow \mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A}) \quad / \quad (\mathbf{B}^{\mathsf{T}}\mathbf{B} * \mathbf{A}^{\mathsf{T}}\mathbf{A})$$

special structure

CP-ALS

Algorithm 1 CP-ALS

procedure CP-ALS($\mathcal{X}, R$) $\quad \triangleright\ \mathcal{X} \in \mathbb{R}^{I \times J \times K}$
    Initialize factor matrices $\mathbf{B}, \mathbf{C}$
    repeat
        $\mathbf{A} \leftarrow \mathbf{X}_{(1)}[(\mathbf{C} \odot \mathbf{B})^\mathsf{T}]^\dagger = \mathbf{X}_{(1)}(\mathbf{C} \odot \mathbf{B})(\mathbf{C}^\mathsf{T}\mathbf{C} * \mathbf{B}^\mathsf{T}\mathbf{B})^\dagger$
        $\mathbf{B} \leftarrow \mathbf{X}_{(2)}[(\mathbf{C} \odot \mathbf{A})^\mathsf{T}]^\dagger = \mathbf{X}_{(2)}(\mathbf{C} \odot \mathbf{A})(\mathbf{C}^\mathsf{T}\mathbf{C} * \mathbf{A}^\mathsf{T}\mathbf{A})^\dagger$
        $\mathbf{C} \leftarrow \mathbf{X}_{(3)}[(\mathbf{B} \odot \mathbf{A})^\mathsf{T}]^\dagger = \mathbf{X}_{(3)}(\mathbf{B} \odot \mathbf{A})(\mathbf{B}^\mathsf{T}\mathbf{B} * \mathbf{A}^\mathsf{T}\mathbf{A})^\dagger$
    until termination criteria met
    return factor matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$
end procedure

init: *random* or *nvec*

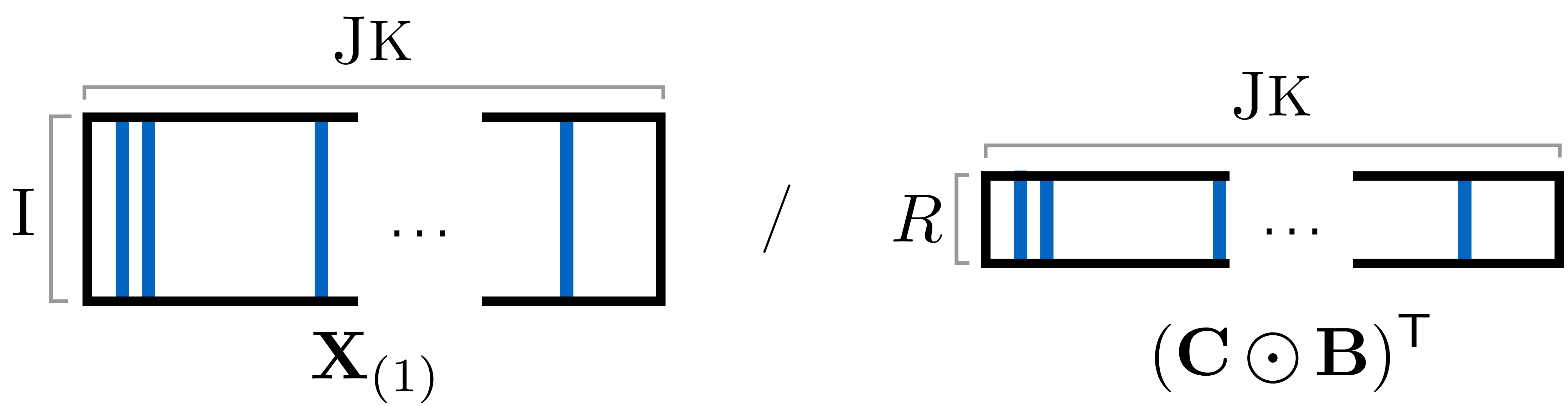
terminate when fit stops changing:

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

CPRAND

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

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

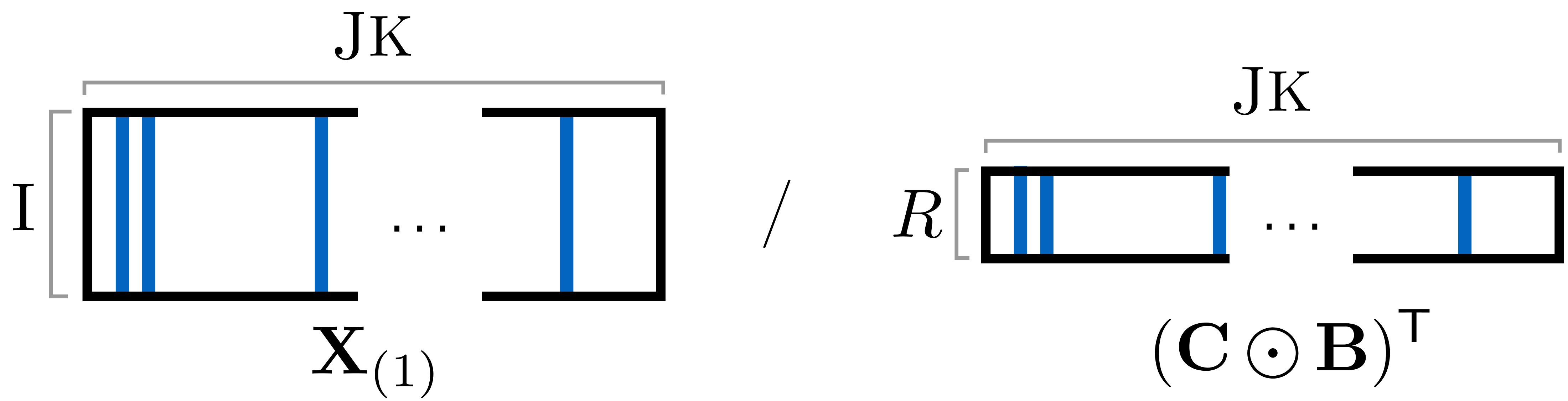


CP-RAND: Just sample $\mathbf{X}_{(1)}$ and $(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$

CPRAND

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

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



CP-RAND: Just sample $\mathbf{X}_{(1)}$ and $(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$

. . . and hope that $(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$ is incoherent. . .

CPRAND

(Order 3 Example, extends to arbitrary order)

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

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

  repeat

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

    $\mathbf{A} \leftarrow \mathbf{X}_{(1)} \mathbf{S}_A^\intercal (\mathbf{S}_A (\mathbf{C} \odot \mathbf{B})^\intercal)^\dagger$

    $\mathbf{B} \leftarrow \mathbf{X}_{(2)} \mathbf{S}_B^\intercal (\mathbf{S}_B (\mathbf{C} \odot \mathbf{A})^\intercal)^\dagger$

    $\mathbf{C} \leftarrow \mathbf{X}_{(3)} \mathbf{S}_C^\intercal (\mathbf{S}_C (\mathbf{B} \odot \mathbf{C})^\intercal)^\dagger$

  until termination criteria met

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

end procedure

CPRAND

(Order 3 Example, extends to arbitrary order)

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

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

  repeat

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

    $\mathbf{A} \leftarrow \mathbf{X}_{(1)}\mathbf{S}_A^\intercal(\mathbf{S}_A(\mathbf{C} \odot \mathbf{B})^\intercal)^\dagger$

    $\mathbf{B} \leftarrow \mathbf{X}_{(2)}\mathbf{S}_B^\intercal(\mathbf{S}_B(\mathbf{C} \odot \mathbf{A})^\intercal)^\dagger$

    $\mathbf{C} \leftarrow \mathbf{X}_{(3)}\mathbf{S}_C^\intercal(\mathbf{S}_C(\mathbf{B} \odot \mathbf{C})^\intercal)^\dagger$

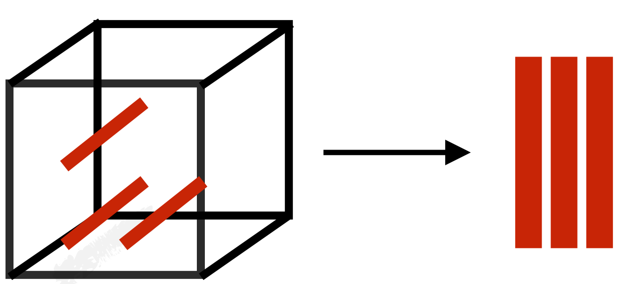
  until termination criteria met

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

end procedure

Trick 1:

Instead of matricizing $\mathfrak{X}$ and then sampling, sample fibers from $\mathfrak{X}$ and then matricize the result.



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 $\mathbf{S}_A, \mathbf{S}_B, \mathbf{S}_C$

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

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

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

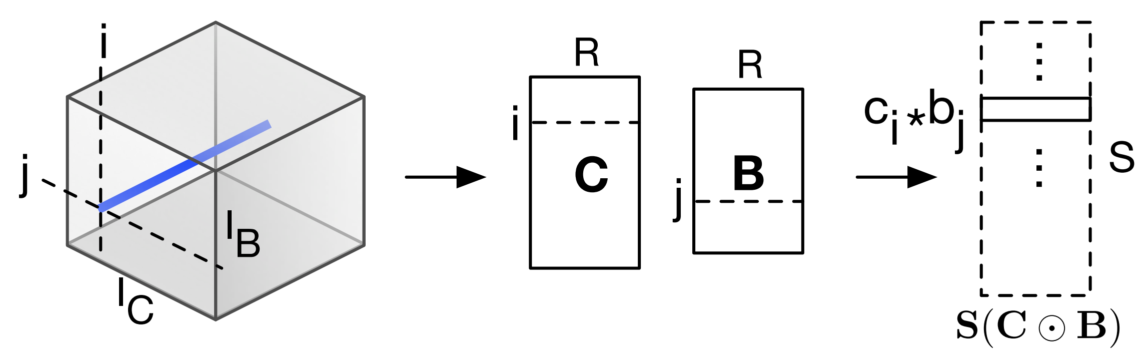
  until termination criteria met

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

end procedure

SKR: Sample KRP without forming

Trick 2:
We don't need to form full KR-Product



CPRAND

(Order 3 Example, extends to arbitrary order)

---

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

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

  repeat

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

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

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

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

  until termination criteria met

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

end procedure

No mixing performed here, yet converges in many cases.

Our result: $\mu(\mathbf{C} \odot \mathbf{B}) \leqslant \mu(\mathbf{C})\mu(\mathbf{B})$

Coherence of Kronecker Product

identity:
$\mathbf{AB} \otimes \mathbf{CD} = (\mathbf{A} \otimes \mathbf{C})(\mathbf{B} \otimes \mathbf{D})$

Lemma 1:

$$\mu(\mathbf{A} \otimes \mathbf{B}) = \mu(\mathbf{A})\mu(\mathbf{B})$$

Proof:

$$\mathbf{A} \otimes \mathbf{B} = \mathbf{Q}_A\mathbf{R}_A \otimes \mathbf{Q}_B\mathbf{R}_B = \underbrace{(\mathbf{Q}_A \otimes \mathbf{Q}_B)}_{\mathbf{Q}_{\mathbf{A}\otimes\mathbf{B}}}\underbrace{(\mathbf{R}_A \otimes \mathbf{R}_B)}_{\mathbf{R}_{\mathbf{A}\otimes\mathbf{B}}}$$

Each row of $\mathbf{Q}_{\mathbf{A}\otimes\mathbf{B}}$ has form $\mathbf{Q}_A(i,:) \otimes \mathbf{Q}_B(j,:)$

Proof follows from $\|\mathbf{a} \otimes \mathbf{b}\| = \|\mathbf{a}\|\|\mathbf{b}\|$

Coherence of KR Product

identity:
$\mathbf{AB} \odot \mathbf{CD} = (\mathbf{A} \otimes \mathbf{C})(\mathbf{B} \odot \mathbf{D})$

Lemma 2:

$$\mu(\mathbf{A} \odot \mathbf{B}) \leqslant \mu(\mathbf{A})\mu(\mathbf{B})$$

Proof:

$$(\mathbf{Q}_A \otimes \mathbf{Q}_B)(\mathbf{R}_A \odot \mathbf{R}_B) = \underbrace{(\mathbf{Q}_A \otimes \mathbf{Q}_B)\mathbf{Q}_R}_{\mathbf{Q}_{\mathbf{A}\odot\mathbf{B}}}\underbrace{\mathbf{R}_R}_{\mathbf{R}_{\mathbf{A}\odot\mathbf{B}}} = \mathbf{Q}_{\mathbf{A}\odot\mathbf{B}}\mathbf{R}_{\mathbf{A}\odot\mathbf{B}}.$$

$$\hat{\mathbf{q}}_i^{\mathsf{T}} \ \ldots \text{row } i \text{ of } \mathbf{Q}_A \otimes \mathbf{Q}_B$$

$$\hat{l}_i = ||\hat{\mathbf{q}}_i^{\mathsf{T}}\mathbf{Q}_R|| = ||\mathbf{Q}_R^{\mathsf{T}}\hat{\mathbf{q}}_i|| \leqslant ||\mathbf{Q}_R^{\mathsf{T}}||\,||\hat{\mathbf{q}}_i||$$

Mixing Tensors

For better convergence & guarantees, we need to mix

Goals:

— Space Efficient

(A single mixed tensor, avoid forming full KRP)

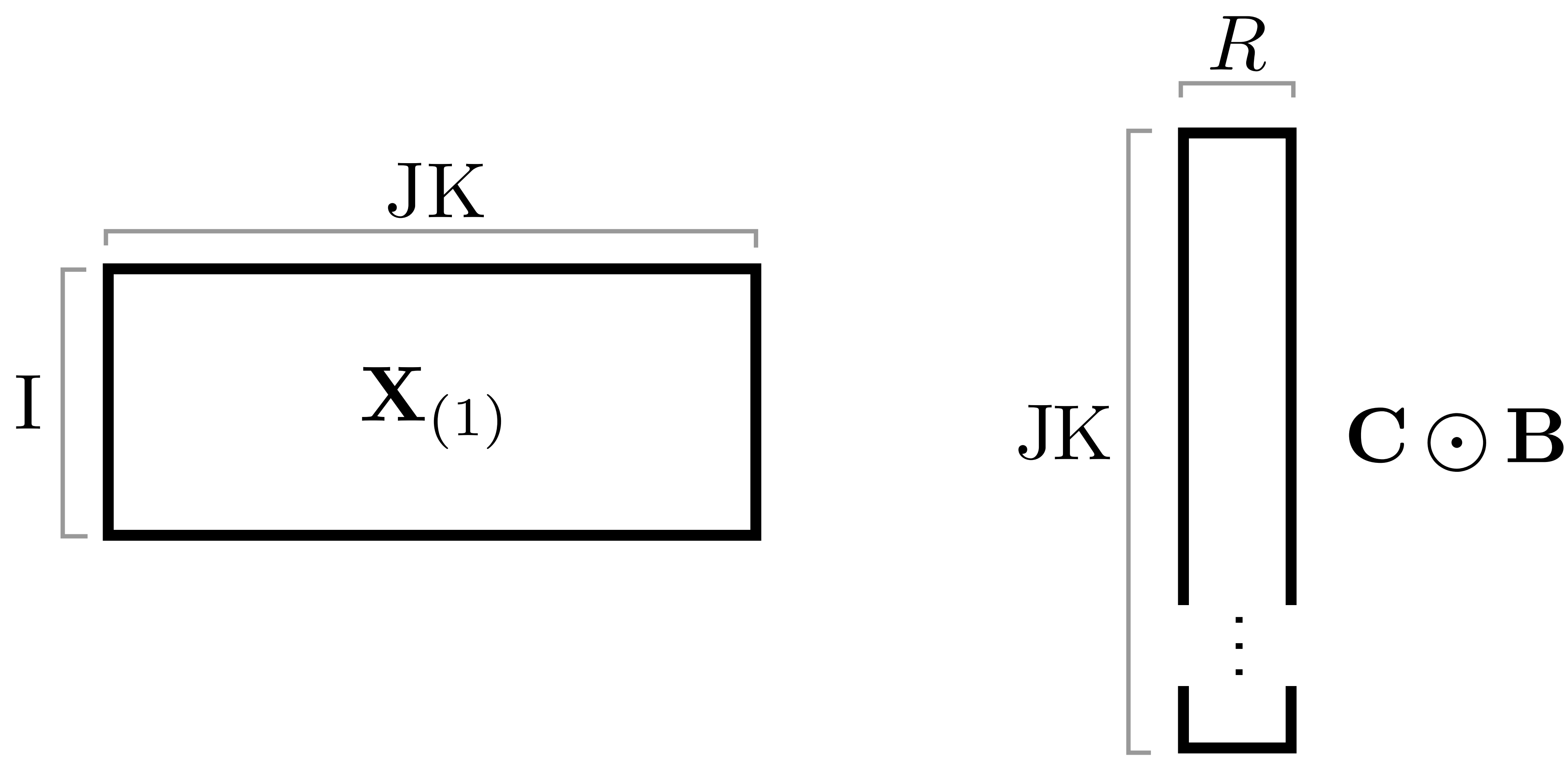
— Time Efficient

(Minimize mixing costs, sample sizes)

Mixing Tensors

CPRAND *without* mixing was: $\mathbf{A} \leftarrow \mathbf{X}_{(1)}\mathbf{S}^{\mathsf{T}}$ / $\mathbf{S}(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$

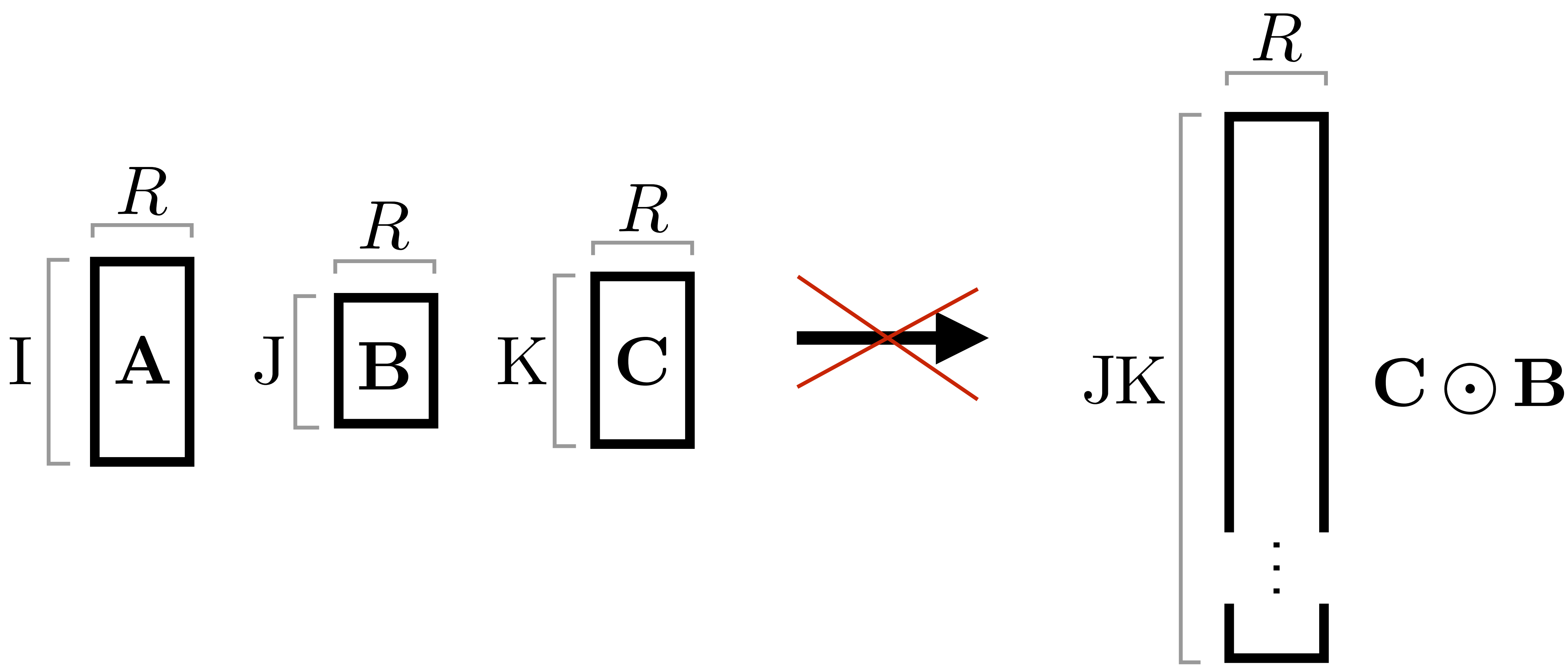
Naive approach: mix $\mathbf{X}_{(1)}, \mathbf{C} \odot \mathbf{B}$ directly, then sample.



$(Expensive)$

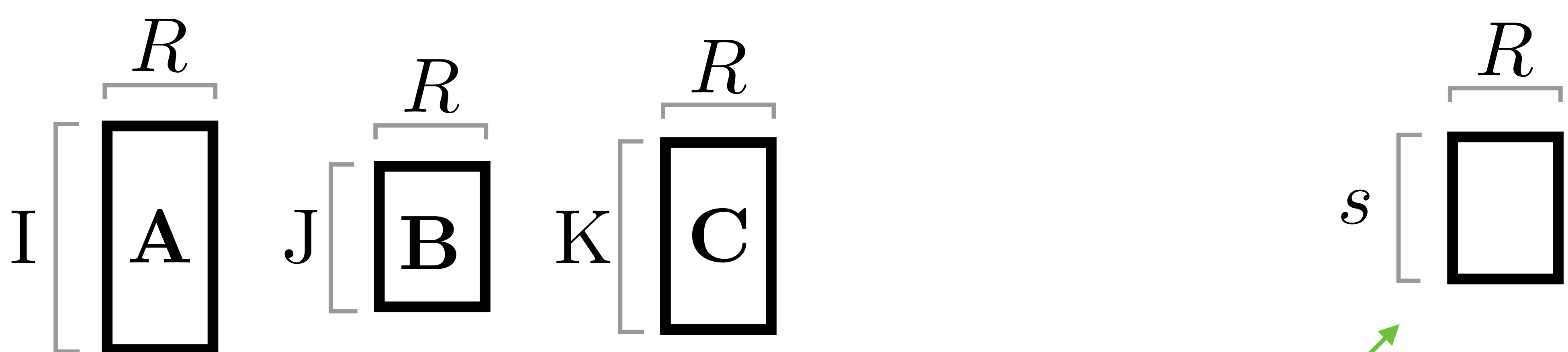
Mixing Factors

Can we avoid forming the full KRP?



Mixing Factors

CPRAND *without* mixing was: $\mathbf{A} \leftarrow \mathbf{X}_{(1)}\mathbf{S}^{\mathsf{T}} \quad / \quad \mathbf{S}(\mathbf{C} \odot \mathbf{B})^{\mathsf{T}}$



Trick 3:

Transform individual factors

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

Sample Khatri Rao product without fully forming

Mixing Factors

Fact:

$$\mathbf{AB} \odot \mathbf{CD} = (\mathbf{A} \otimes \mathbf{C})(\mathbf{B} \odot \mathbf{D})$$

So the mixed factors expand:

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

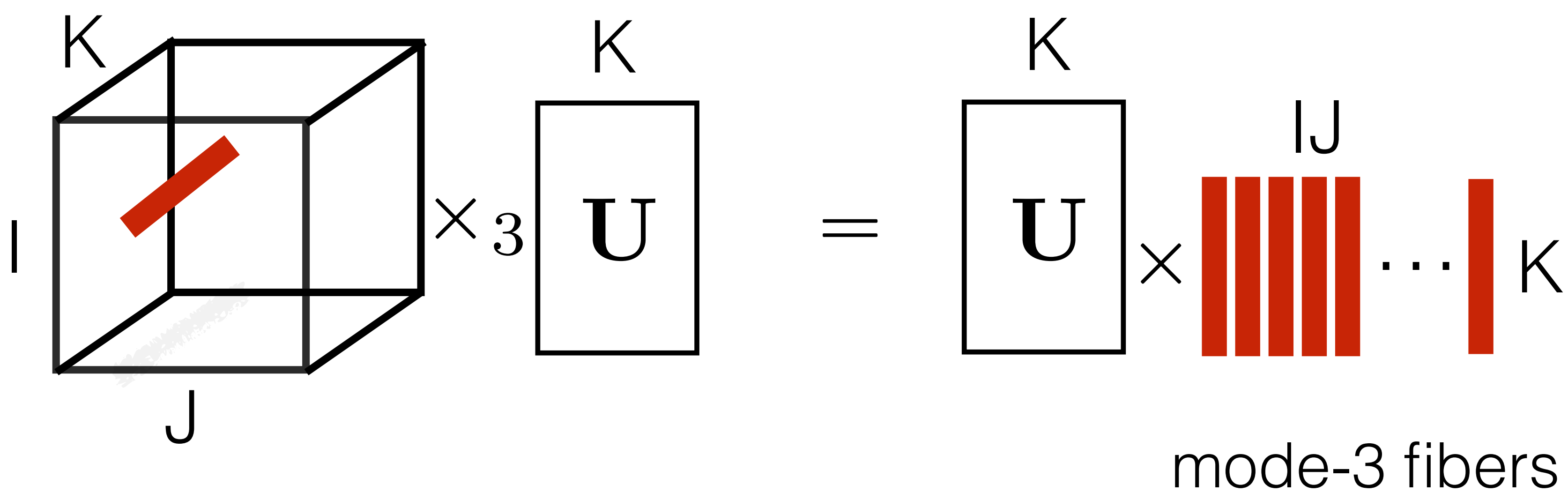
$$= \quad \underline{(\mathcal{F}_C \mathbf{D}_C \otimes \mathcal{F}_B \mathbf{D}_B)}(\mathbf{C} \odot \mathbf{B})$$

apply to $\mathbf{X}_{(1)}$ (the LHS)

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

Mode-*n* Multiplication

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



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

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

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

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

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

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

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

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

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

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

inverse mix the final, small, solution

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

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

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

inverse mix the final, small, solution

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

Mixing Factors

Trick 4:
Mix original tensor only once:

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

Summary:

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

efficiently sample S fibers from mode n of $\hat{\mathcal{X}}$ *before* reshaping **(Trick 1)**

efficiently sample S rows of KRP **(Trick 2)**

mix individual factors **(Trick 3)**

Algorithm 3: CPRAND-MIX

(Order 3 Example, extends to arbitrary order)

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

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

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

$$\arg\min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2 = \arg\min_{\mathbf{x} \in \mathbb{R}^n} \left\| \begin{bmatrix} \Re(\mathbf{A}) \\ \Im(\mathbf{A}) \end{bmatrix} \mathbf{x} - \begin{bmatrix} \Re(\mathbf{b}) \\ \Im(\mathbf{b}) \end{bmatrix} \right\|_2$$

(Trick 5)

Algorithm 4: CPRAND-PREMIX

If we use a real-valued orthogonal transform such as the DCT,
We can simply mix, call CPRAND, then unmix the output.

---

function CPRAND-PREMIX($\mathcal{X}, R, S$) $\quad \triangleright\ \mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_N}$

  Define random sign-flip operators $\mathbf{D}_m$ and orthogonal matrices $\mathcal{F}_m,\ m \in \{1, \ldots, N\}$

  Mix: $\hat{\mathcal{X}} \leftarrow \mathcal{X} \times_1 \mathcal{F}_1 \mathbf{D}_1 \times \cdots \times_N \mathcal{F}_N \mathbf{D}_N$

  $[\boldsymbol{\lambda}, \{\hat{\mathbf{A}}^{(n)}\}] =$ CPRAND($\hat{\mathcal{X}}, R, S$)

  for $n = 1, \ldots, N$ **do**

    Unmix: $\mathbf{A}^{(n)} = \mathbf{D}_n \mathcal{F}_n^{\mathsf{T}} \hat{\mathbf{A}}^{(n)}$

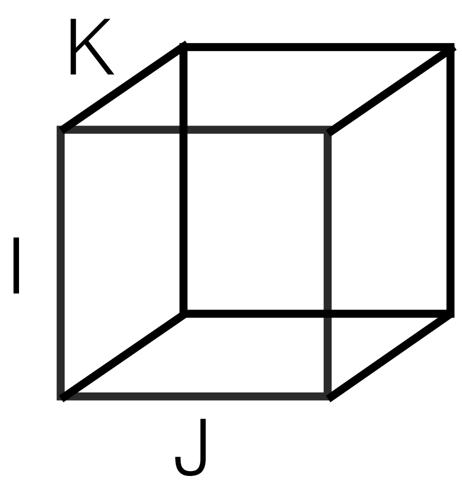
  end for

  return factor matrices $\{\mathbf{A}^{(n)}\}$

end function

Trick 5:

Stopping Criterion

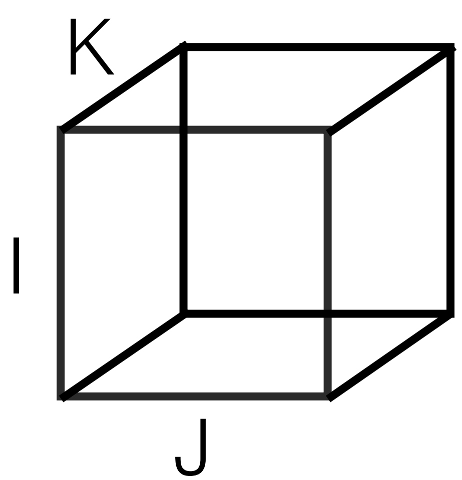


CP-ALS:

expensive inner product

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

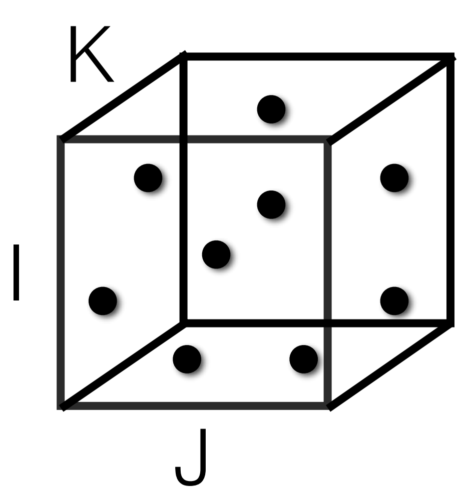
Trick 5: Stopping Criterion



CP-ALS:

expensive inner product

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



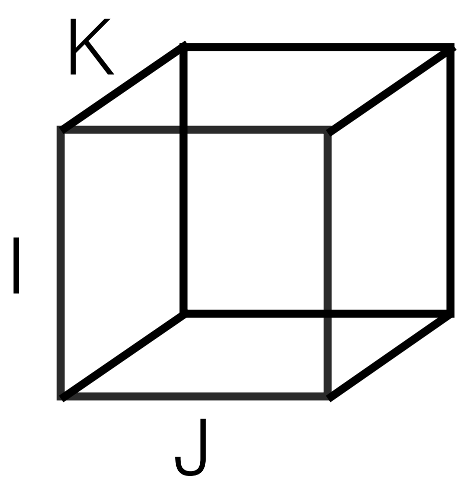
CPRAND:

Uniformly sample $\mathcal{X}$:

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

Choose some subset of tensor entries

Trick 5:
Stopping Criterion

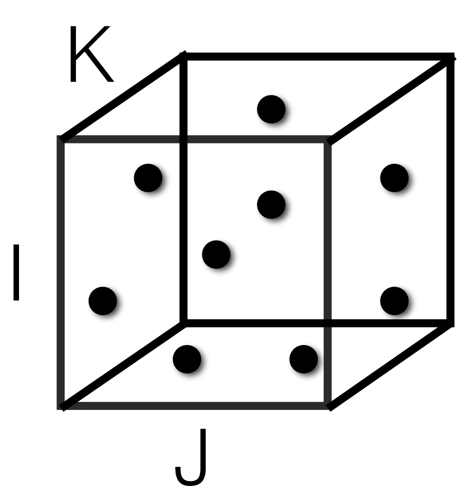


CP-ALS:

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:



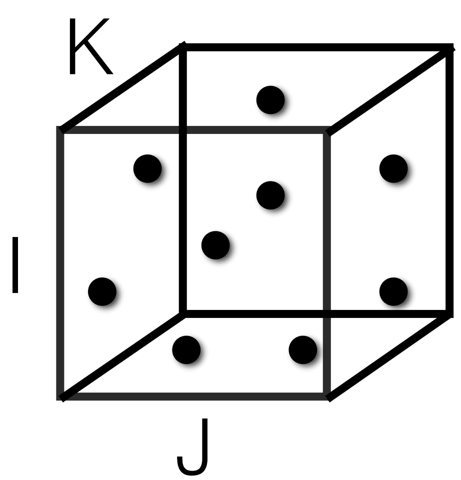
Uniformly sample $\mathcal{X}$:

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

approximate fit

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

Trick 5: Stopping Criterion



CPRAND:

Uniformly sample $\mathcal{X}$:

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

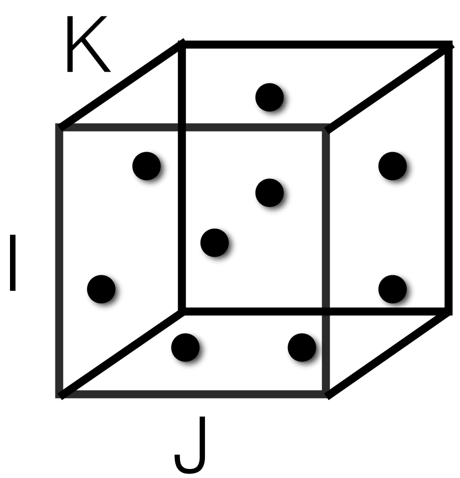
approximate fit

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

How many samples are needed?
Since we are approximating a mean,
we can apply the Chernoff-Hoeffding bound

Trick 5:

Stopping Criterion



approximate fit

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

Very conservative bound:
(assuming e_i is i.i.d)

$(\mu_{\max} = \max(e_i^2))$

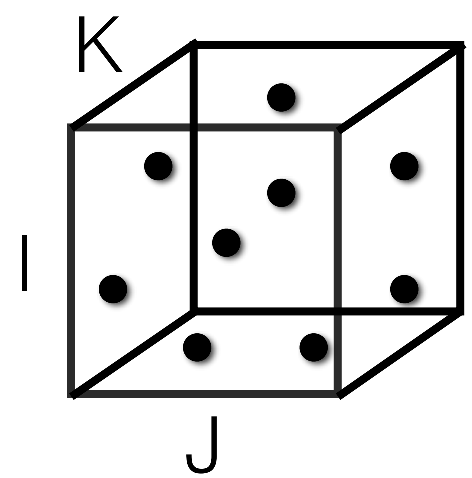
#samples $\hat{P} \geq 372\mu_{\max}^2$

Yields 95% confidence that the sampled fit is within 5% of true fit

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

Trick 5:

Stopping Criterion



approximate fit

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

Terminate after fit ceases to decrease after a certain number of iterations

Experimental Setup

Environment

— MATLAB R2016a & Tensor Toolbox v2.6.

— Intel Xeon E5-2650 Ivy Bridge 2.0 GHz machine with 32 GB of memory

Data

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

— Synthetic random tensor

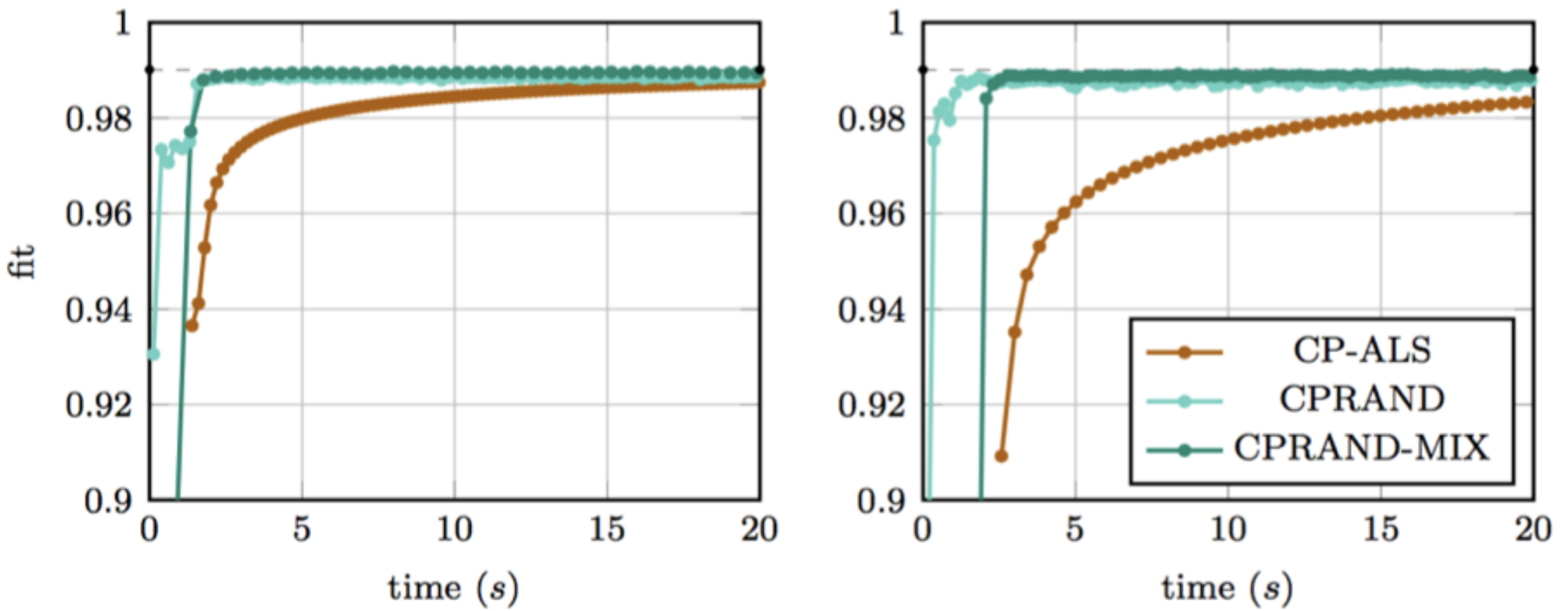
— Factors of rank R_{true}

— 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

Example Run

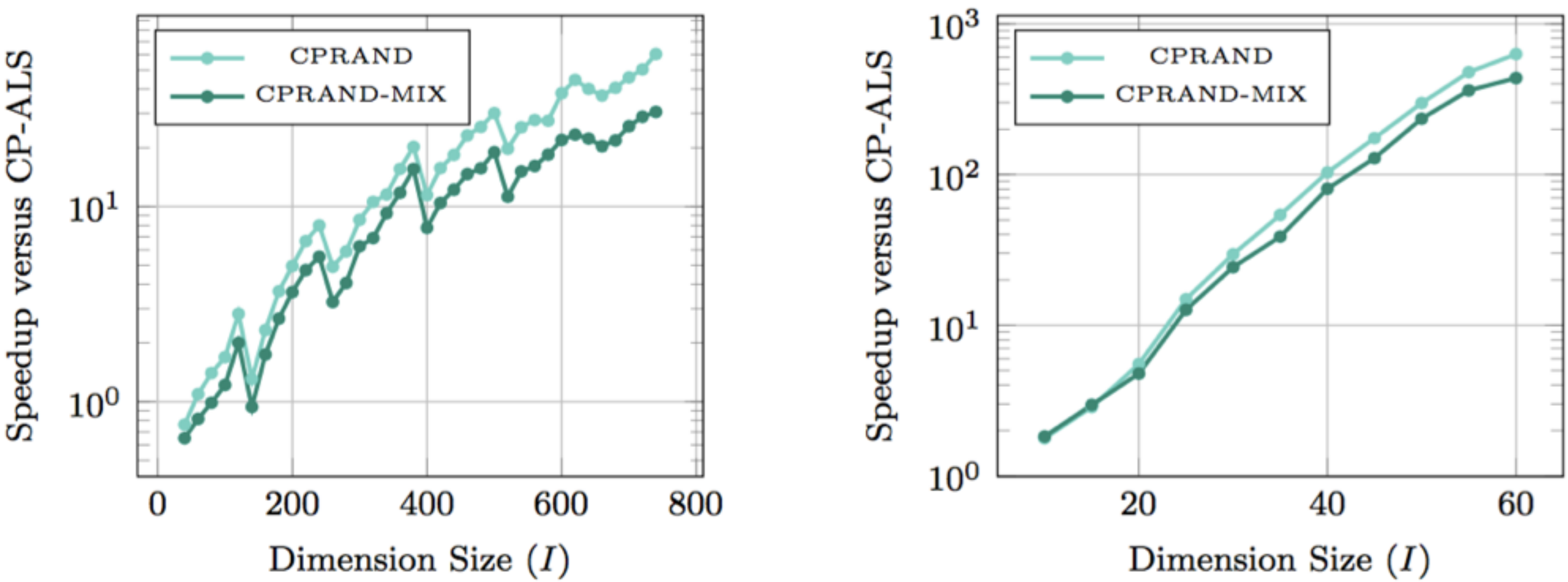


(a) Random 300 × 300 × 300 tensor

(b) Random 80 × 80 × 80 × 80 tensor

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

Per-Iteration Times



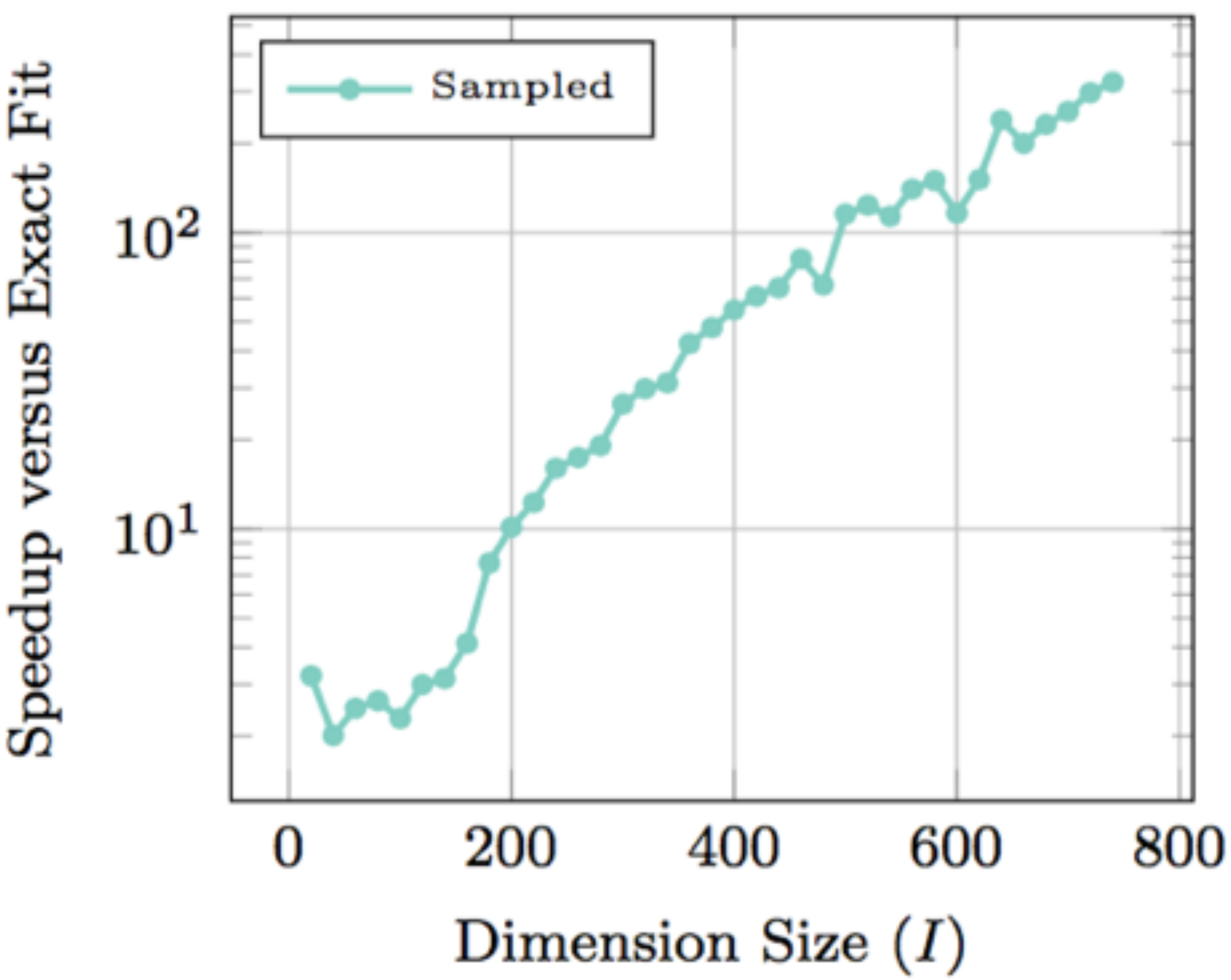
(a) Order 3: $I \times I \times I$

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

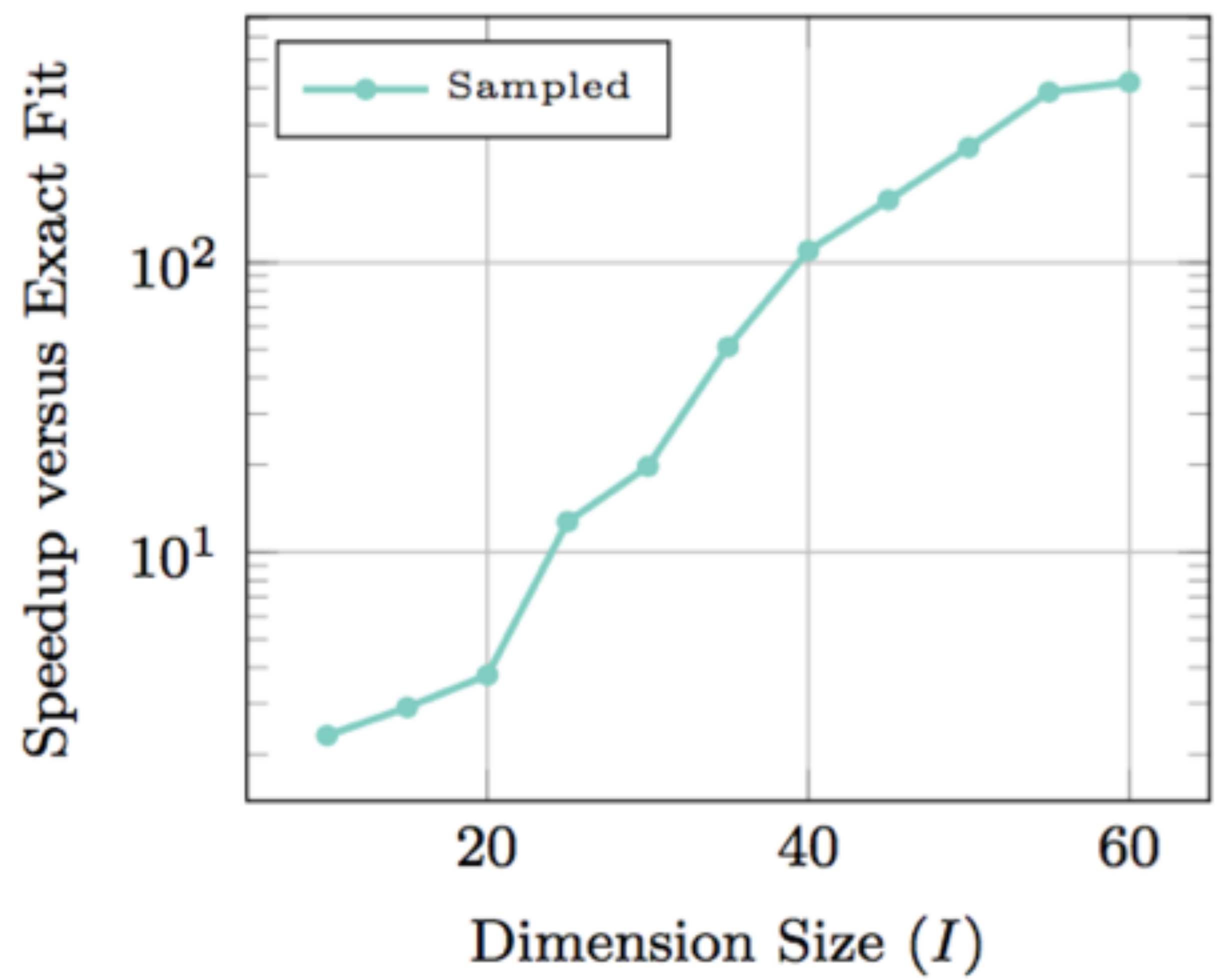
$$R = 5 \qquad S = 10 R \log R$$

Stopping Criterion Speedup

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



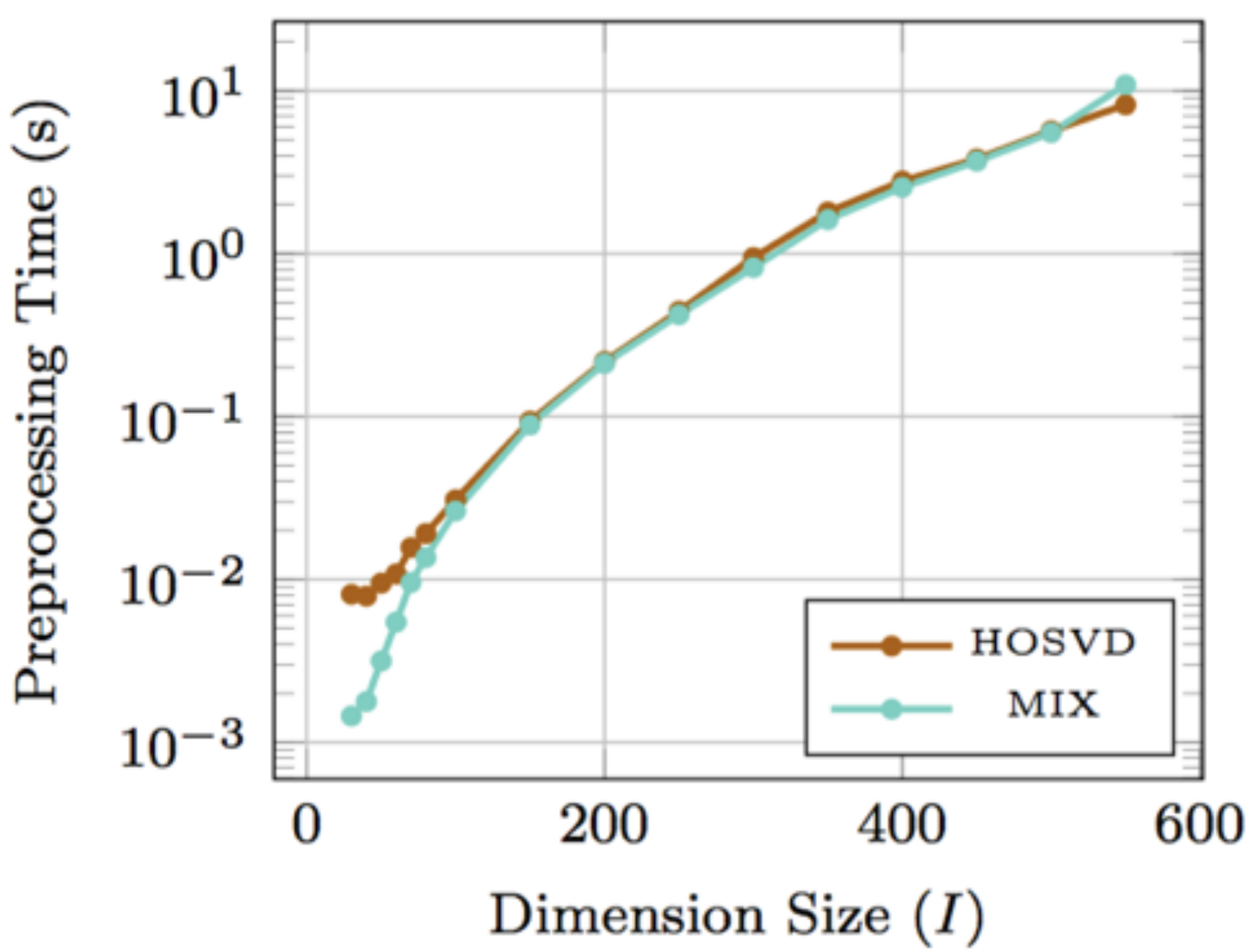
(a) Order 3: $I \times I \times I$



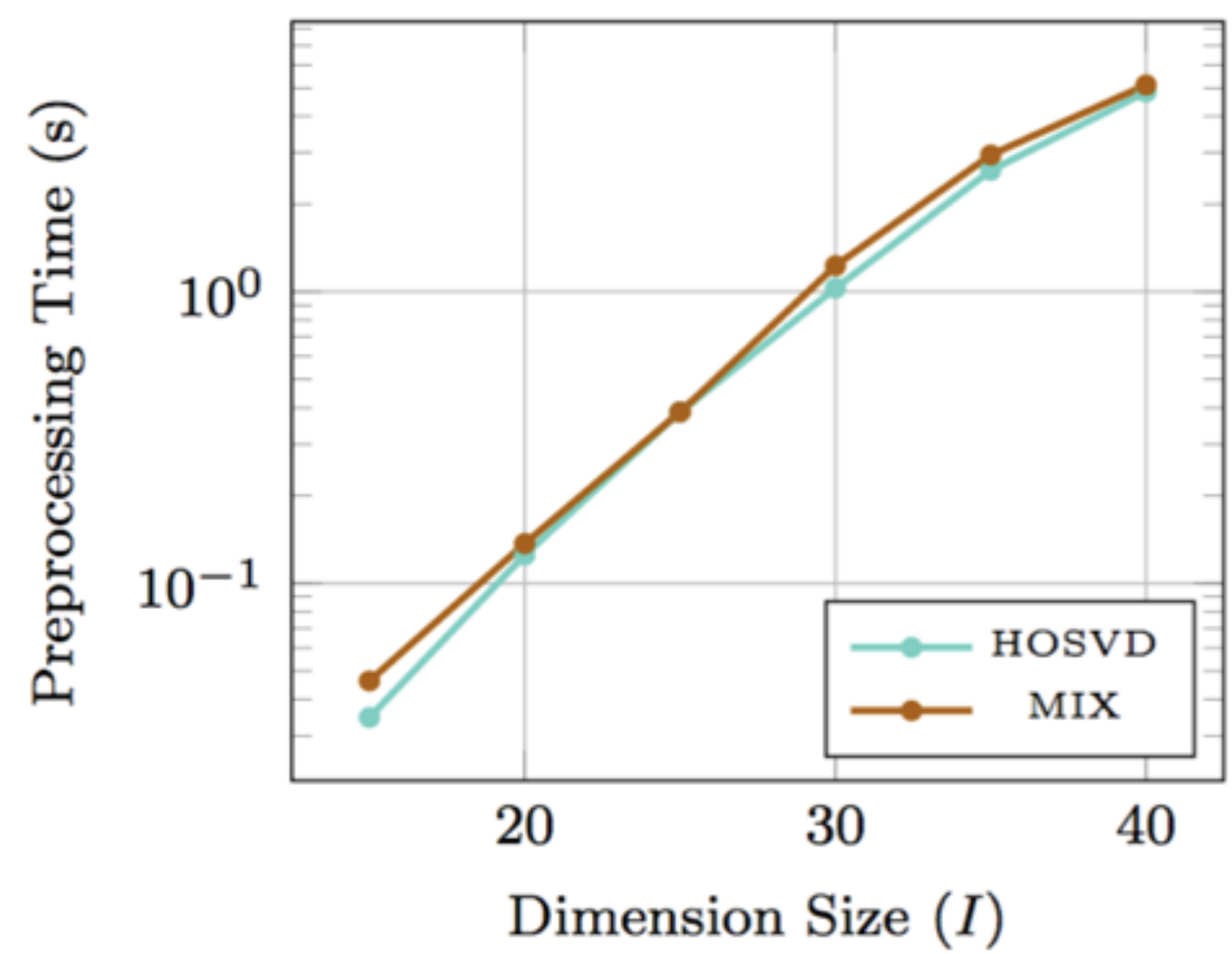
(b) Order 5: $I \times I \times I \times I \times I$

Preprocessing Times

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

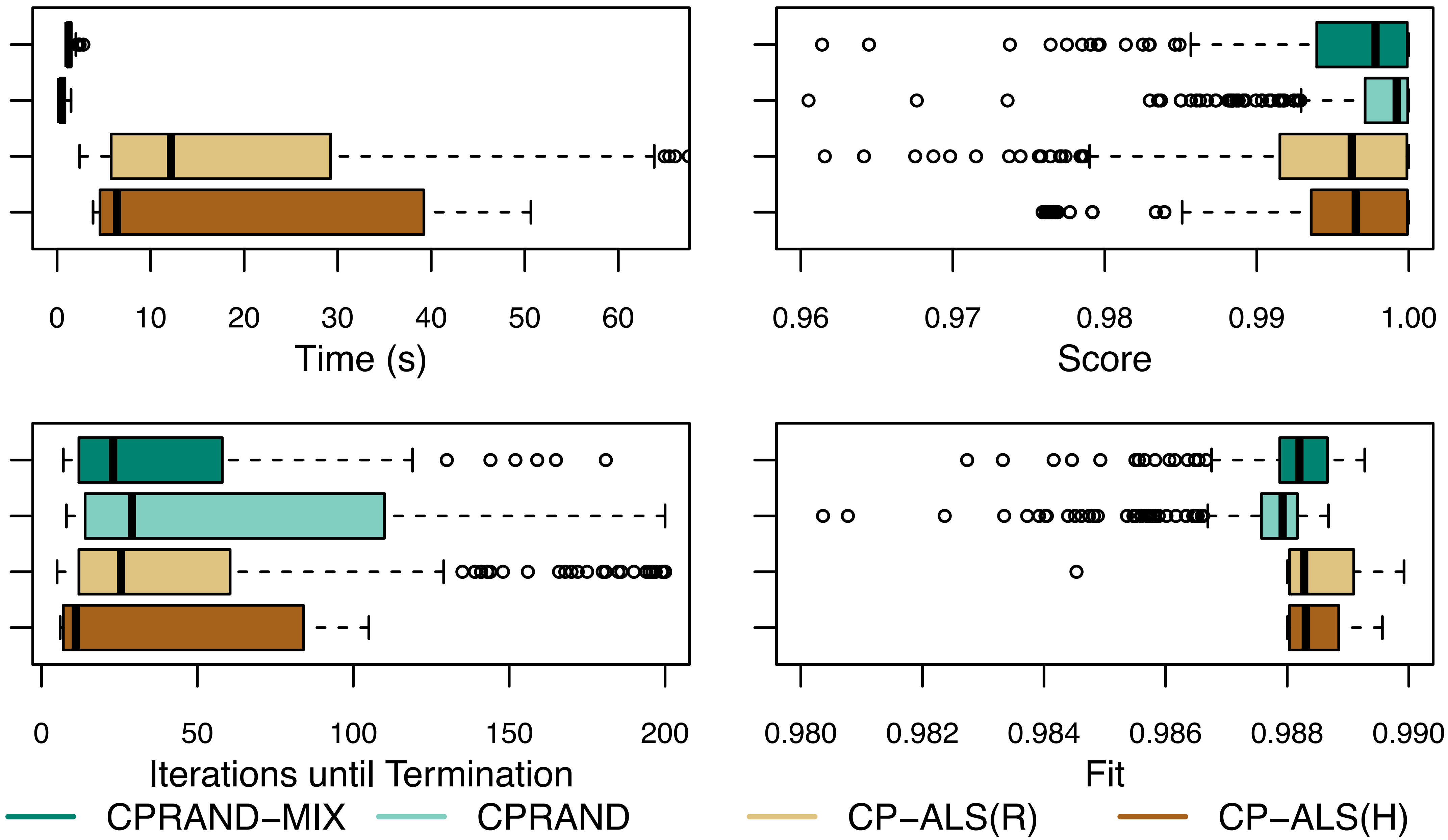


(a) Order 3: $I \times I \times I$



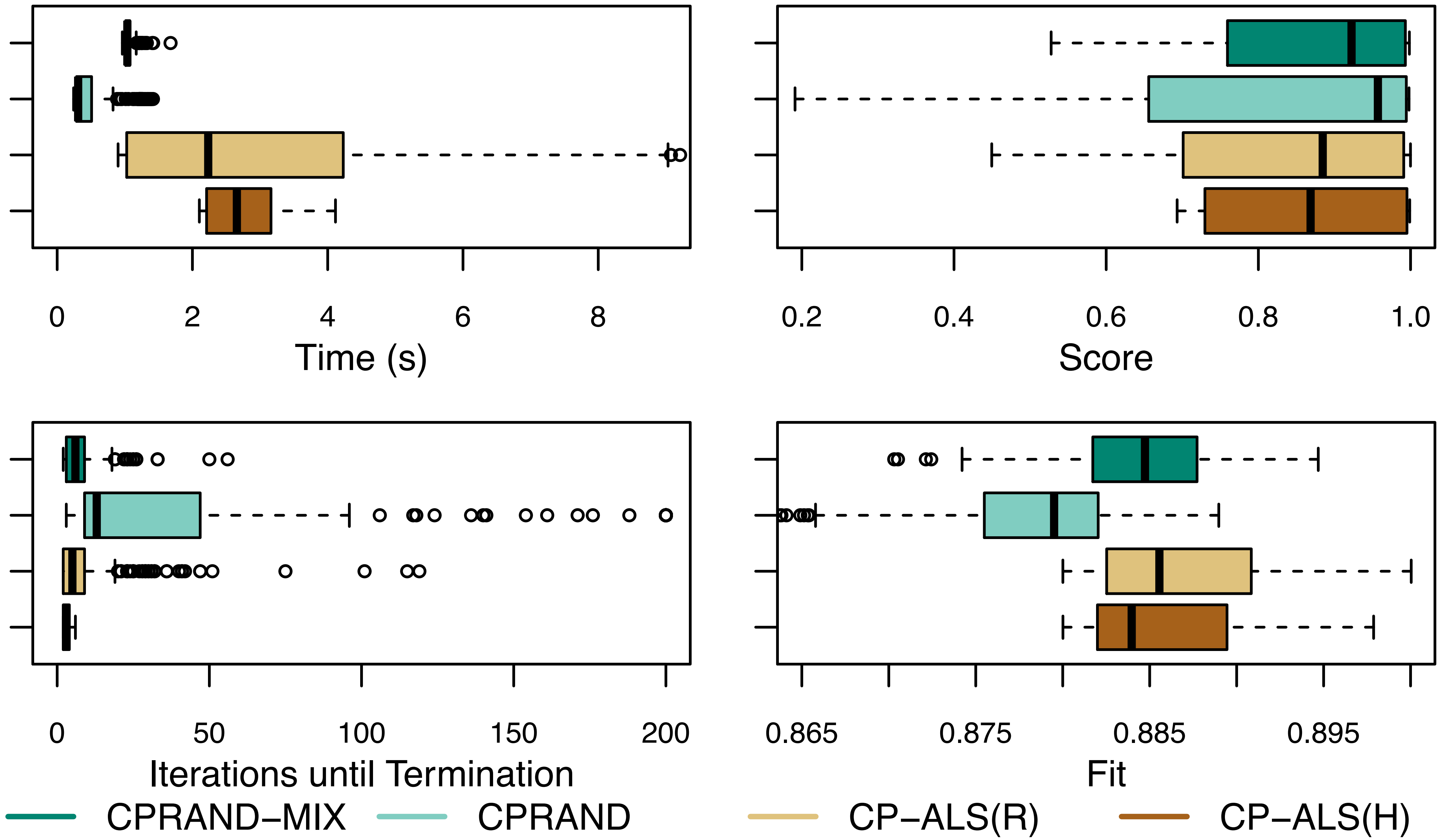
(b) Order 5: $I \times I \times I \times I \times I$

Experiment 1: order 4



Size: 90x90x90x90	$\mathbf{R}_{true}$: 5	**R**: {5,6}
Noise: 1%	600 Runs	**Collinearity**: {0.5,0.9}

Experiment 1: order 4



Size: 90x90x90x90	**R**$_{true}$: 5	**R**: {5,6}
Noise: 10%	600 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.

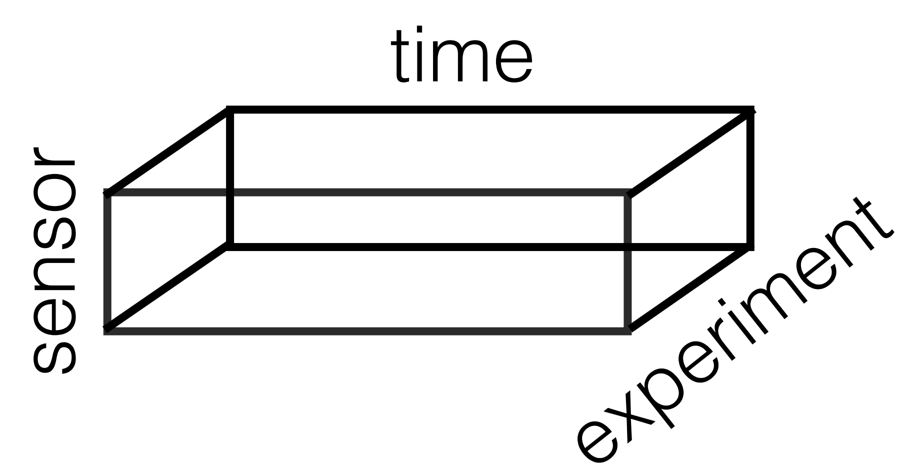
Experiment 2: Hazardous Gas Classification

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!

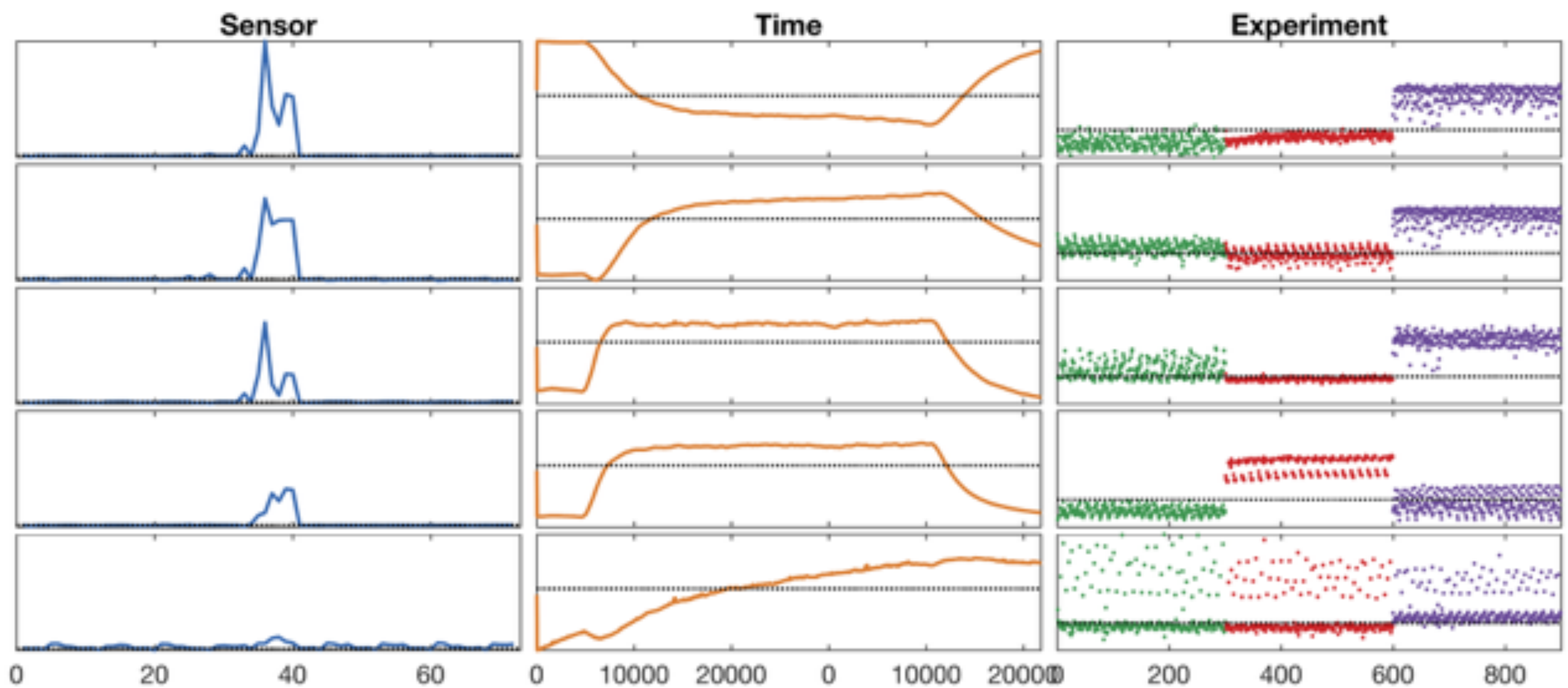
Experiment 2: Hazardous Gas Classification

(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 Work

- Randomization in the Tucker model
- Randomization in sparse CP

(for scalability)

Contact

- Casey Battaglino (cbattaglino3@gatech.edu)