Inter-University Research Institute Corporation /
Research Organization of Information and Systems
National Institute of Informatics

Non-negative low-rank approximations for multi-dimensional arrays on statistical manifold

Kazu Ghalamkari[1,2], Mahito Sugiyama[1,2]

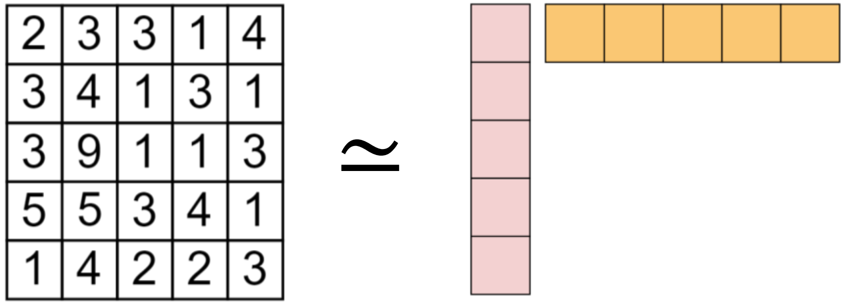
1 : The Graduate University for Advanced Studies, SOKENDAI
2 : National Institute of Informatics

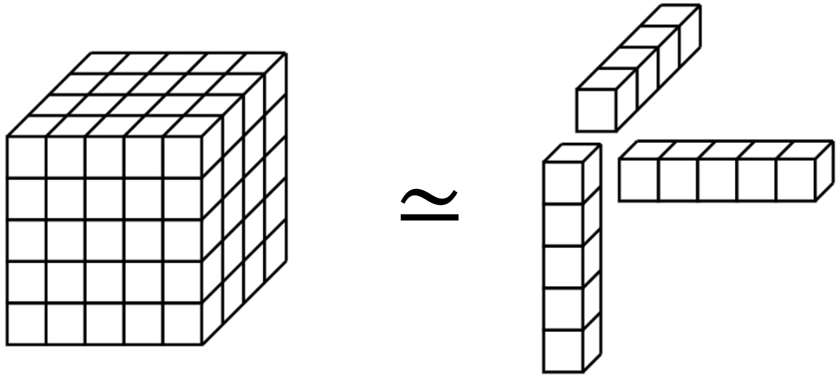
International Conference on Information Geometry for Data Science (IG4DS 2022)

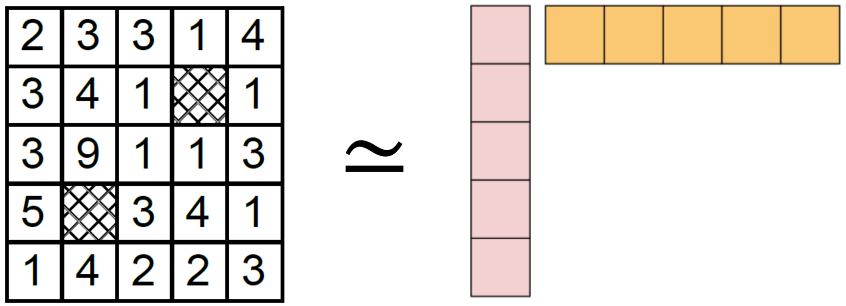
Motivation

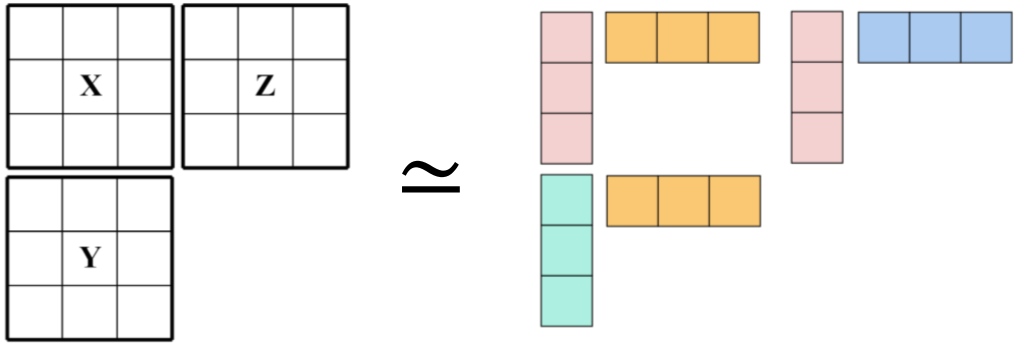
☐ Non-negative low-rank approximation of data with various structures

Approximates with a linear combination of fewer bases (principal components) for feature extraction, memory reduction, and pattern discovery.😀





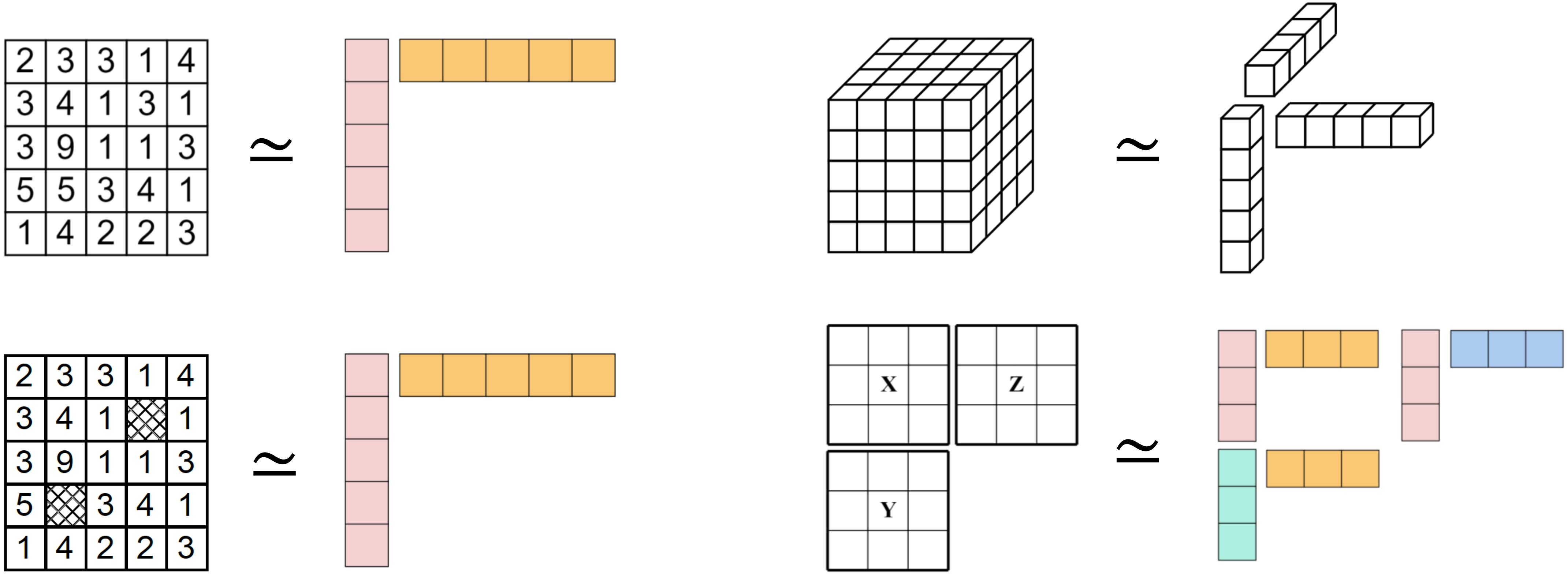




Motivation

☐ Non-negative low-rank approximation of data with various structures

Approximates with a linear combination of fewer bases (principal components) for feature extraction, memory reduction, and pattern discovery.😀

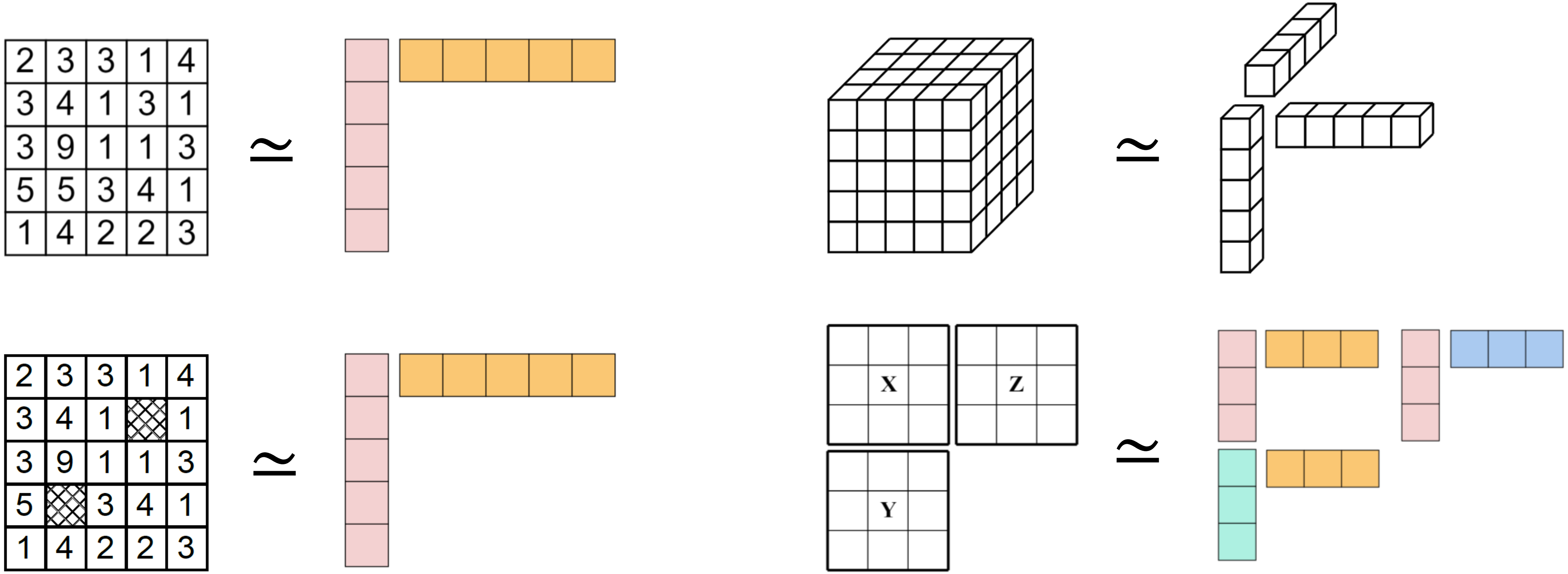


Non-negative constraint improves interpretability

Motivation

☐ Non-negative low-rank approximation of data with various structures

Approximates with a linear combination of fewer bases (principal components) for feature extraction, memory reduction, and pattern discovery.😀

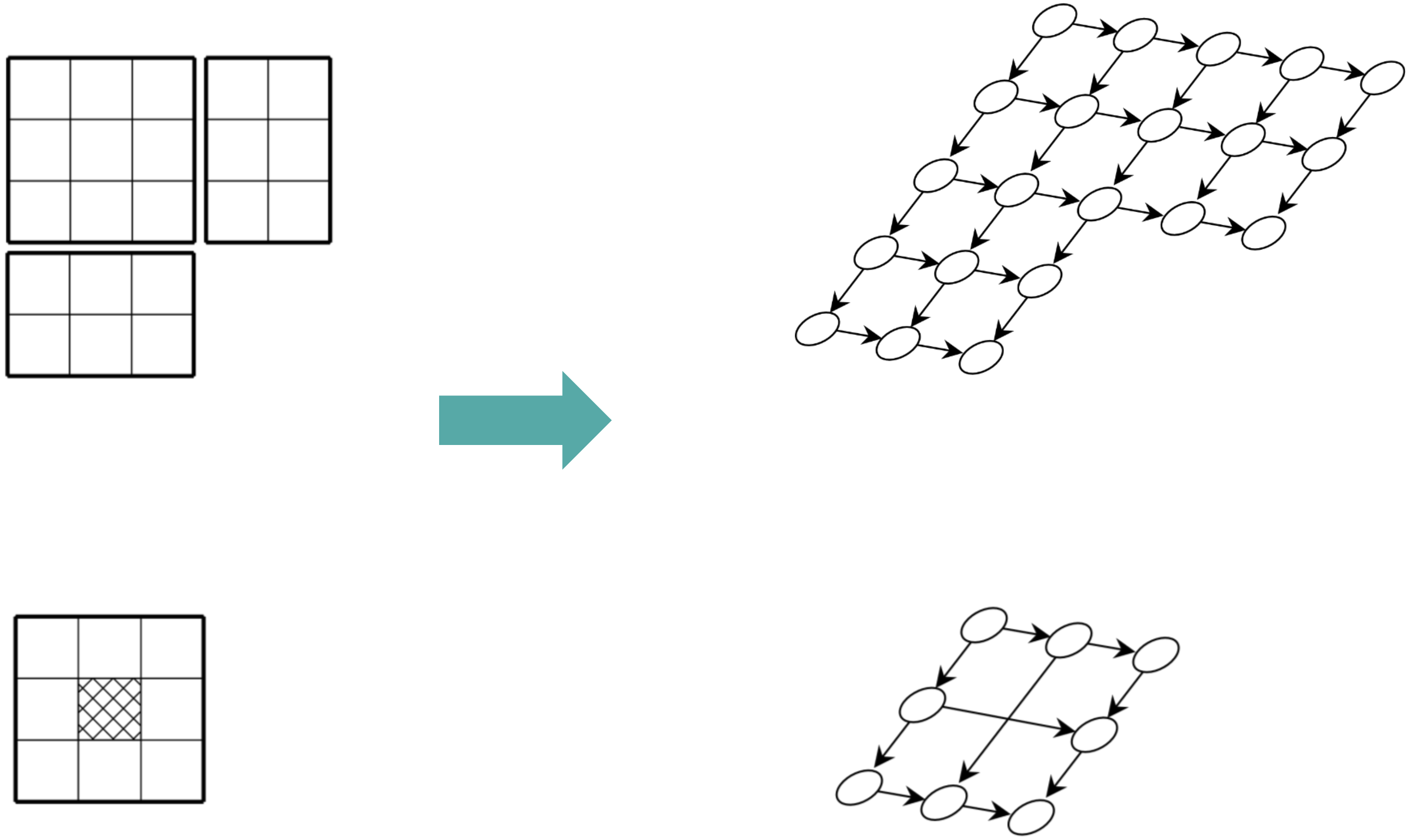


Non-negative constraint improves interpretability

Low-rank approximation with non-negative constraints are based on gradient methods.
→ Appropriate settings for stopping criteria, learning rate, and initial values are necessary 😥

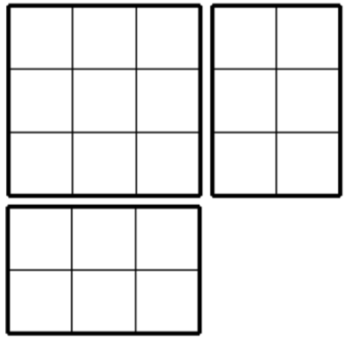
Strategy

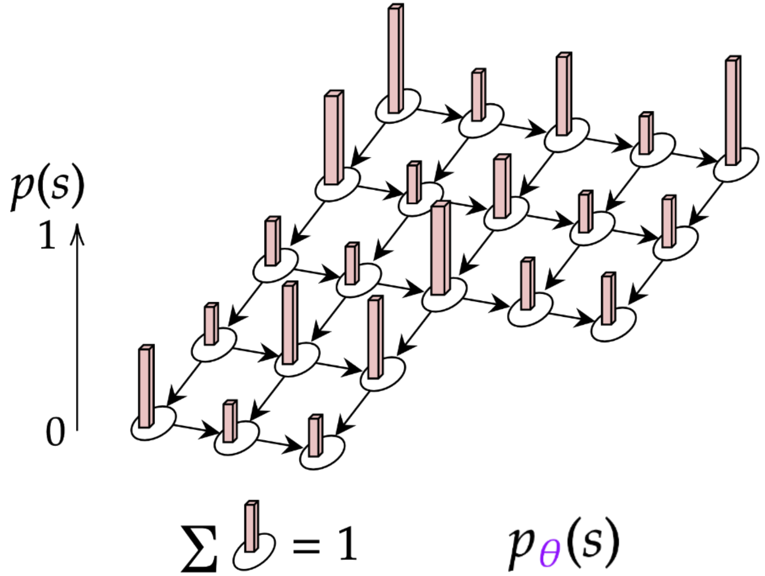
- ☐ Modeling with probability mass function on Directed Acyclic Graph(DAG).
- ☐ Utilize projection theory of information geometry.

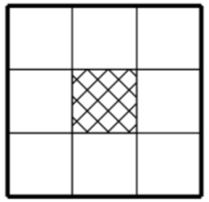


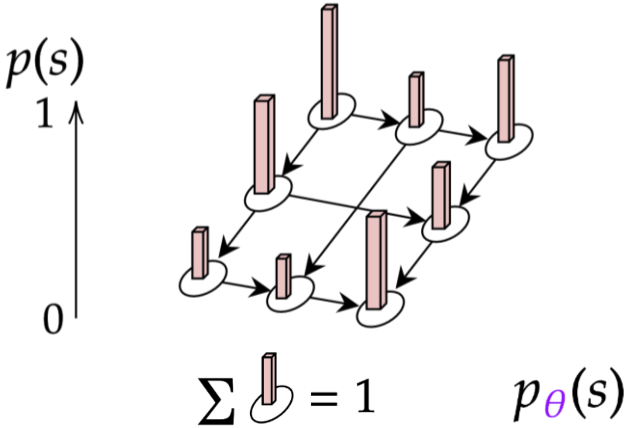
Strategy

- ☐ Modeling with probability mass function on Directed Acyclic Graph(DAG).
- ☐ Utilize projection theory of information geometry.



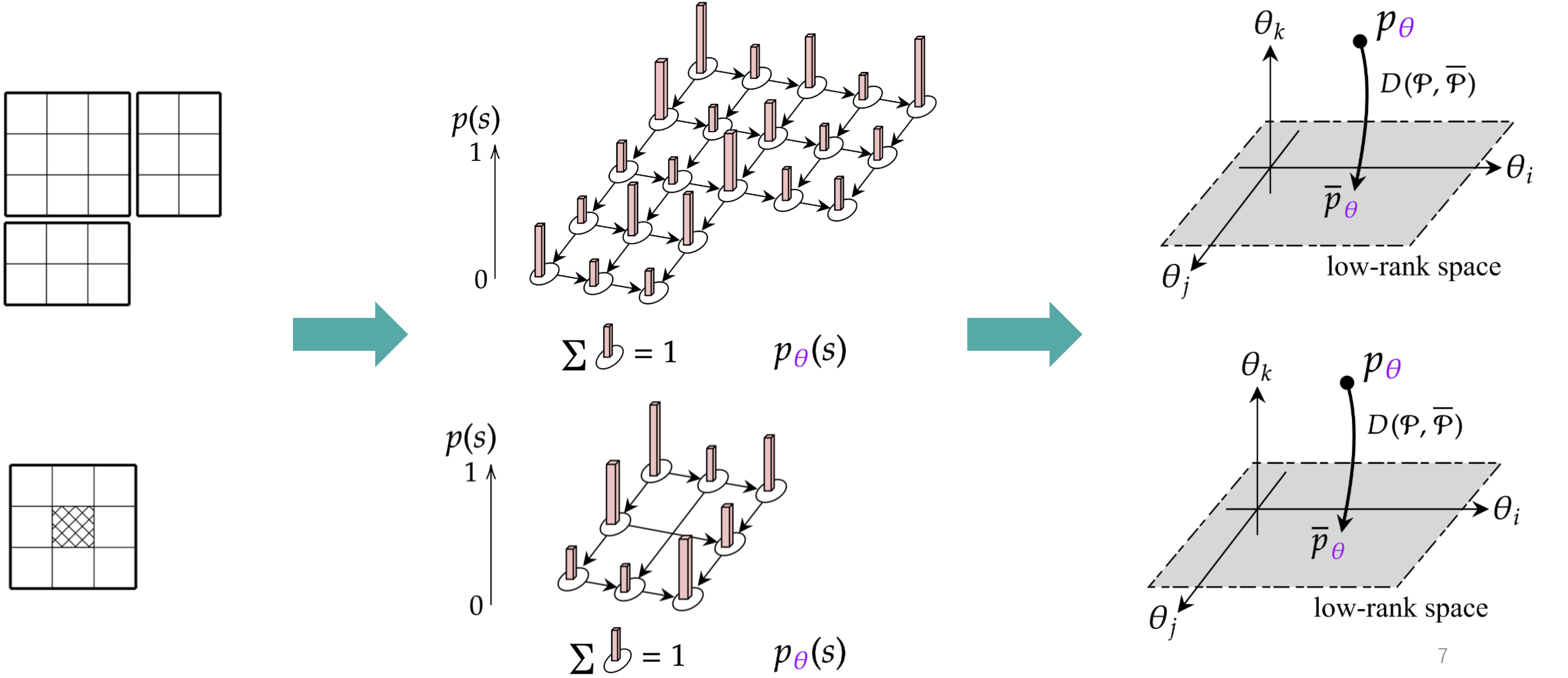






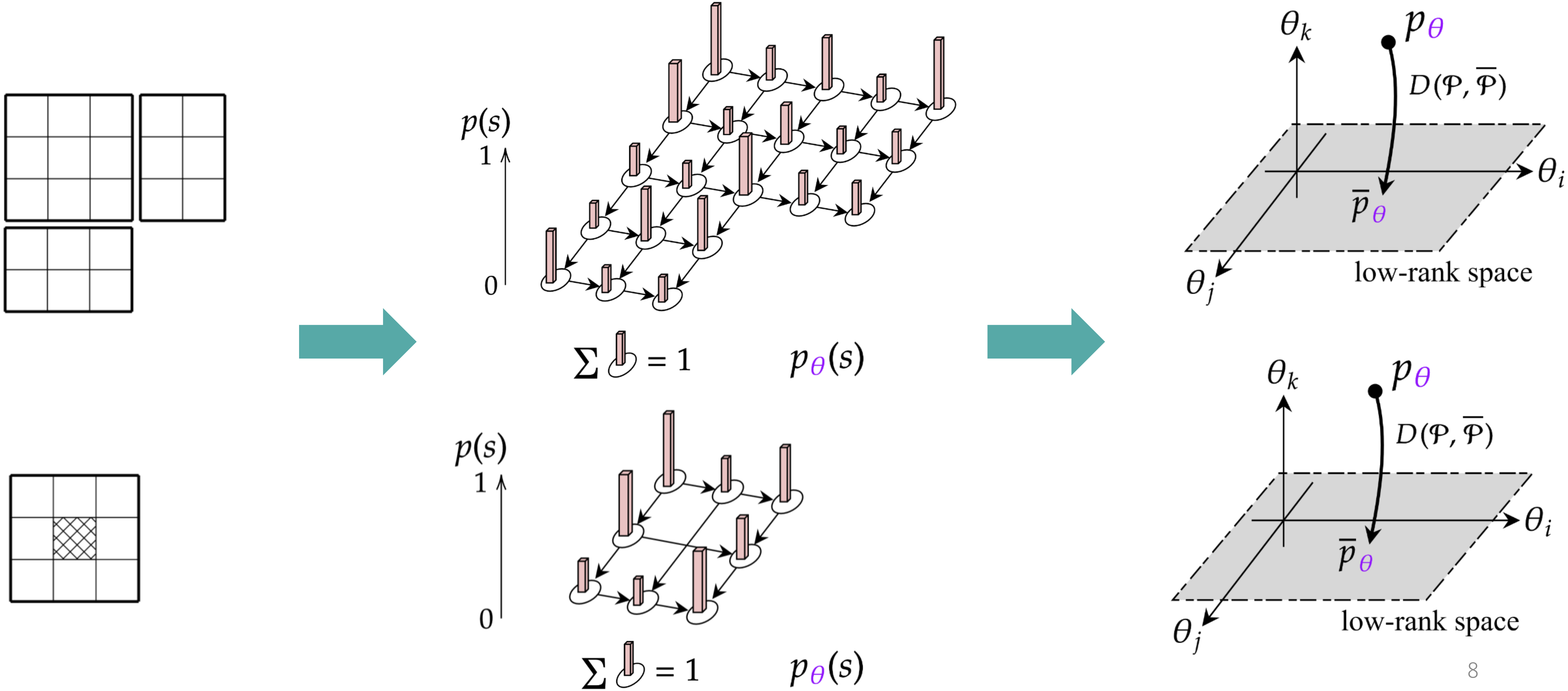
Strategy

- ☐ Modeling with probability mass function on Directed Acyclic Graph(DAG).
- ☐ Utilize projection theory of information geometry.



Strategy

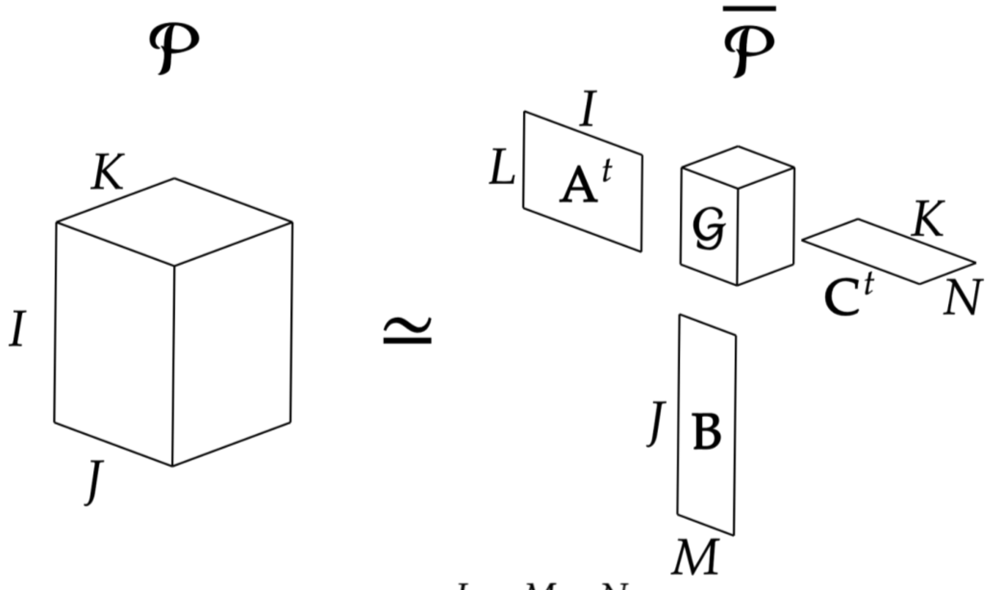
- ☐ Modeling with probability mass function on Directed Acyclic Graph(DAG).
- ☐ Utilize projection theory of information geometry.



Contribution

Information Geometric Analysis using Distributions on DAGs that Correspond to Data Structures

☐ LTR: Faster Tucker-rank Reduction



$$\mathcal{P}_{ijk} \simeq \overline{\mathcal{P}}_{ijk} = \sum_{l=1}^{L} \sum_{m=1}^{M} \sum_{n=1}^{N} \mathcal{G}_{ijk} \mathbf{A}_{il} \mathbf{B}_{jm} \mathbf{C}_{kn}$$

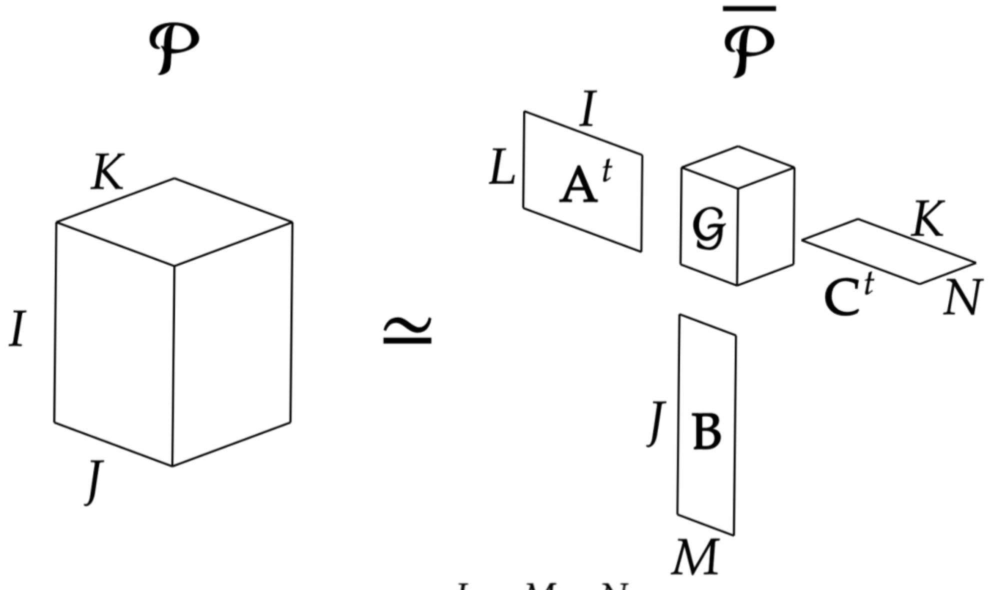
$$rank(\overline{\mathcal{P}}) = (L, M, N)$$

No worries about initial values, stopping criterion and learning rate 😄

Contribution

Information Geometric Analysis using Distributions on DAGs that Correspond to Data Structures

☐ LTR: Faster Tucker-rank Reduction



$$\mathcal{P}_{ijk} \simeq \overline{\mathcal{P}}_{ijk} = \sum_{l=1}^{L}\sum_{m=1}^{M}\sum_{n=1}^{N}\mathcal{G}_{ijk}\mathbf{A}_{il}\mathbf{B}_{jm}\mathbf{C}_{kn}$$

$$rank(\overline{\mathcal{P}}) = (L, M, N)$$

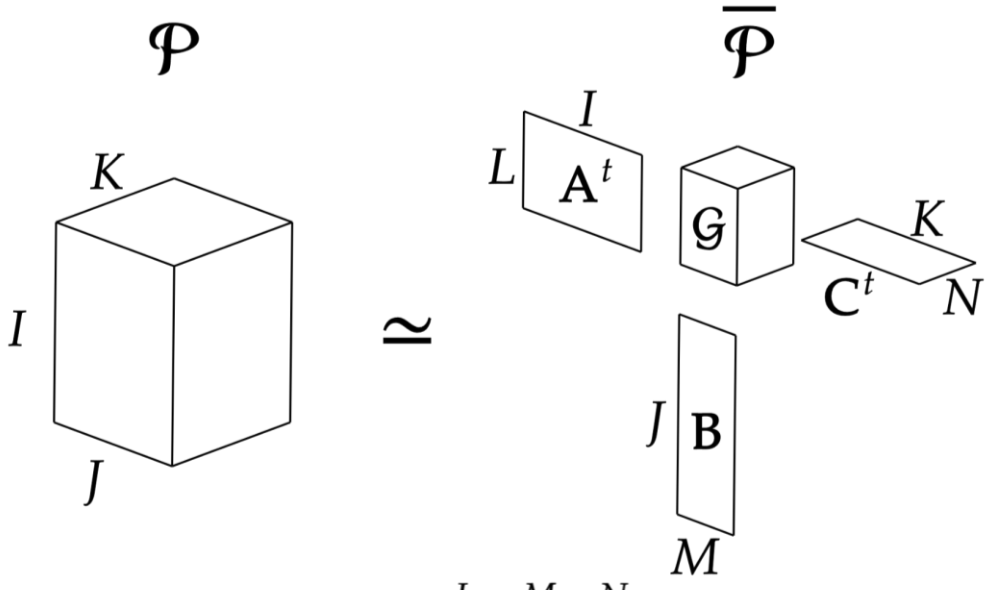
Rank-1 = rank(1,1,1)

No worries about initial values, stopping criterion and learning rate 😄

Contribution

Information Geometric Analysis using Distributions on DAGs that Correspond to Data Structures

☐ LTR: Faster Tucker-rank Reduction

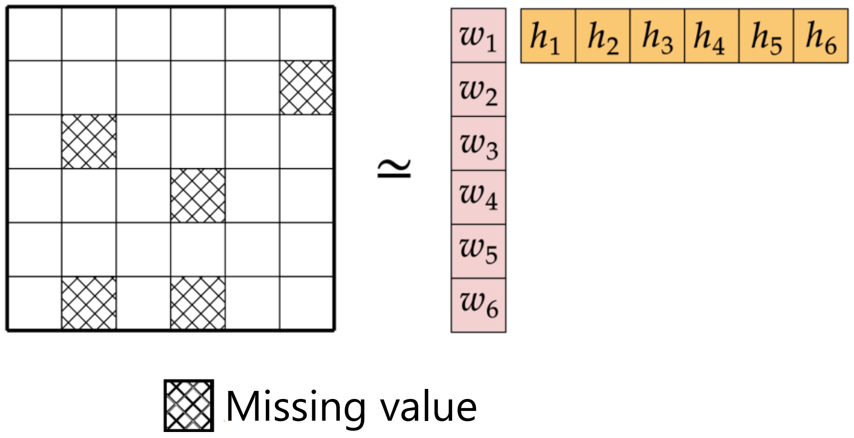


$$\mathcal{P}_{ijk} \simeq \overline{\mathcal{P}}_{ijk} = \sum_{l=1}^{L}\sum_{m=1}^{M}\sum_{n=1}^{N} \mathcal{G}_{ijk}\mathbf{A}_{il}\mathbf{B}_{jm}\mathbf{C}_{kn}$$

$$rank(\overline{\mathcal{P}}) = (L, M, N)$$

Rank-1 = rank(1,1,1)

☐ A1GM: Faster rank-1 missing NMF



Find the most dominant factor rapidly.

Solve the task as a coupled NMF.

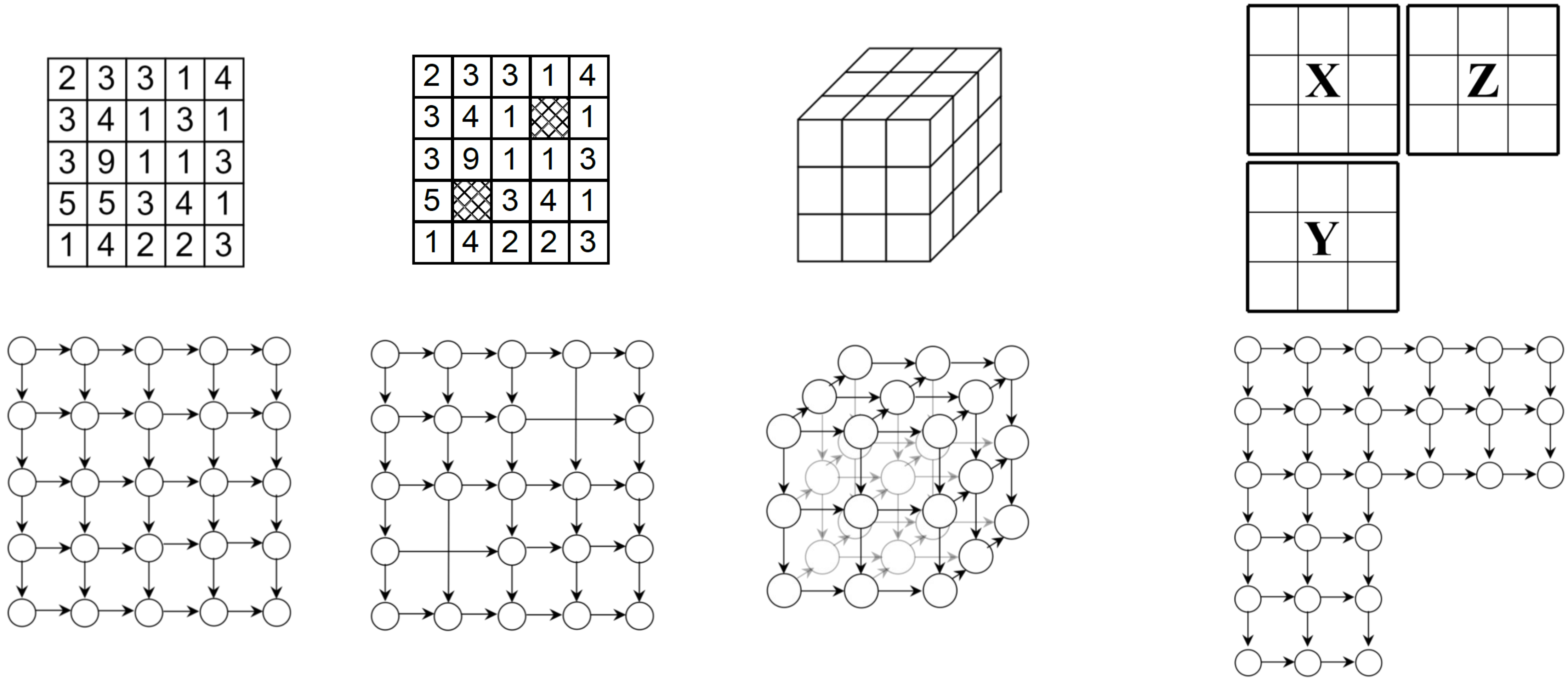
No worries about initial values, stopping criterion and learning rate 😄

Contents

3:00

Modeling tensor and matrix

- Flexible modeling is required to capture the structure of various data



Formulate low-rank approximations with probabilistic models on DAGs

Log-linear model on Directed Acyclic Graph (DAG)

□DAG(poset)

$\mathcal{S}$ is a DAG ⇔ for all $s_1, s_2, s_3 \in \mathcal{S}$ the following three properties are satisfied.

(1) **Reflexivity** : $s_1 \le s_1$ (2) **Antisymmetry:** $s_1 \le s_2, s_2 \le s_1 \Rightarrow s_1 = s_2$ (3)**Transitivity:**$s_1 \le s_2, s_2 \le s_3 \Rightarrow s_1 \le s_3$

Mahito Sugiyama, Hiroyuki Nakahara and Koji Tsuda
"Tensor balancing on statistical manifold"(2017) ICML.

Log-linear model on Directed Acyclic Graph (DAG)

□DAG(poset)

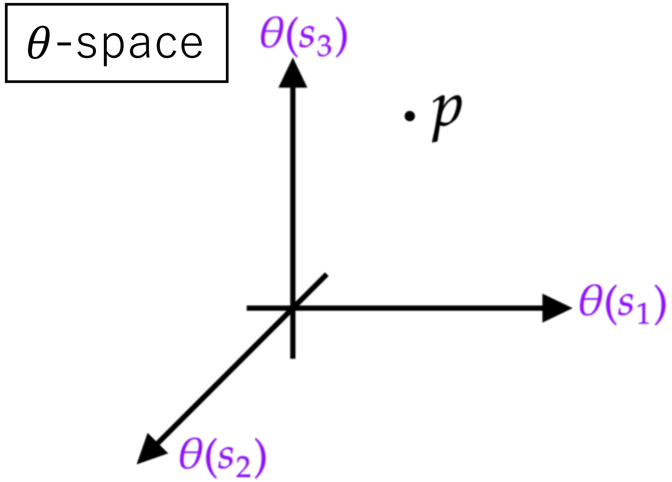
$\mathcal{S}$ is a DAG ⇔ for all $s_1, s_2, s_3 \in \mathcal{S}$ the following three properties are satisfied.

(1) **Reflexivity** : $s_1 \leq s_1$ (2) **Antisymmetry:** $s_1 \leq s_2, s_2 \leq s_1 \Rightarrow s_1 = s_2$ (3)**Transitivity:**$s_1 \leq s_2, s_2 \leq s_3 \Rightarrow s_1 \leq s_3$

□ log-linear model on DAG

We define the log-linear model on a DAG $\mathcal{S}$ as a mapping $p: \mathcal{S} \rightarrow (0,1)$. **Natural parameters $\boldsymbol{\theta}$** describe the model.

$$p_\theta(x) = \exp\left(\sum_{s \leq x} \theta(s)\right), \quad x \in \mathcal{S}$$



Mahito Sugiyama, Hiroyuki Nakahara and Koji Tsuda
"Tensor balancing on statistical manifold"(2017) ICML.

Log-linear model on Directed Acyclic Graph (DAG)

□DAG(poset)

$\mathcal{S}$ is a DAG ⇔ for all $s_1, s_2, s_3 \in \mathcal{S}$ the following three properties are satisfied.

(1) **Reflexivity** : $s_1 \leq s_1$ (2) **Antisymmetry:** $s_1 \leq s_2, s_2 \leq s_1 \Rightarrow s_1 = s_2$ (3)**Transitivity:**$s_1 \leq s_2, s_2 \leq s_3 \Rightarrow s_1 \leq s_3$

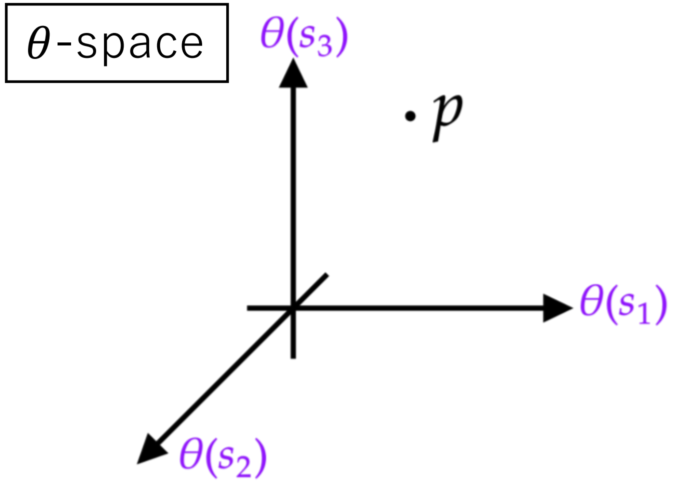
□ log-linear model on DAG

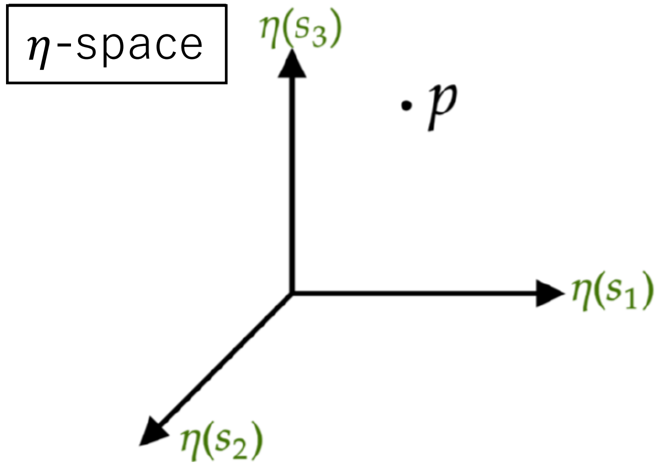
We define the log-linear model on a DAG $\mathcal{S}$ as a mapping $p\colon \mathcal{S} \to (0,1)$. **Natural parameters $\boldsymbol{\theta}$** describe the model.

$$p_\theta(x) = \exp\left(\sum_{s \leq x} \theta(s)\right), \quad x \in \mathcal{S}$$

We can also describe the model by **expectation parameters $\boldsymbol{\eta}$** with Möbius function.

$$\eta(x) = \sum_{s \geq x} p(s), \qquad p_\eta(x) = \sum_{s \in \mathcal{S}} \mu(x, s)\eta(s).$$





Möbius function

$$\mu(x, y) = \begin{cases} -\sum_{x \leq s < y} \mu(x, s) & \text{if } x < y \\ 1 & \text{if } x = y \\ 0 & \textit{otherwiese} \end{cases}$$

$s_1, s_2, s_3 \in \mathcal{S}$

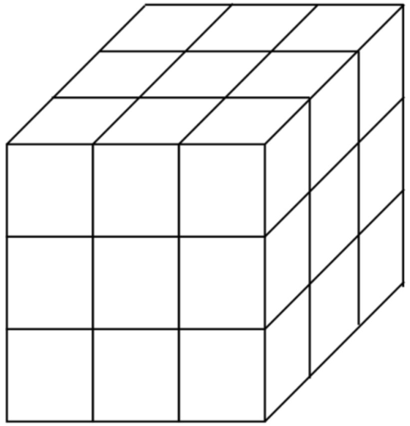
Mahito Sugiyama, Hiroyuki Nakahara and Koji Tsuda
"Tensor balancing on statistical manifold"(2017) ICML.

Contents

4:30

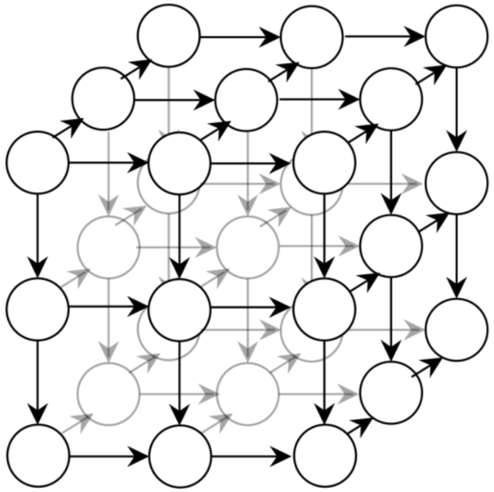
Introducing DAGs for Tensor

$\mathcal{P}$



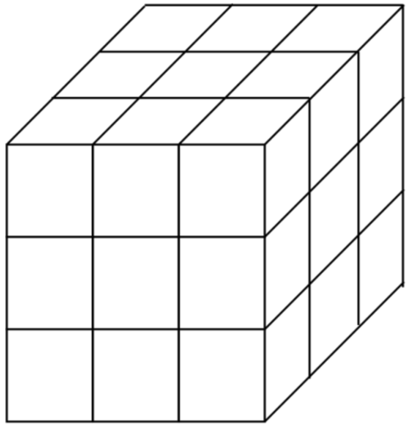
$$\sum_{i,j,k} \mathcal{P}_{ijk} = 1$$

$\mathcal{S}$



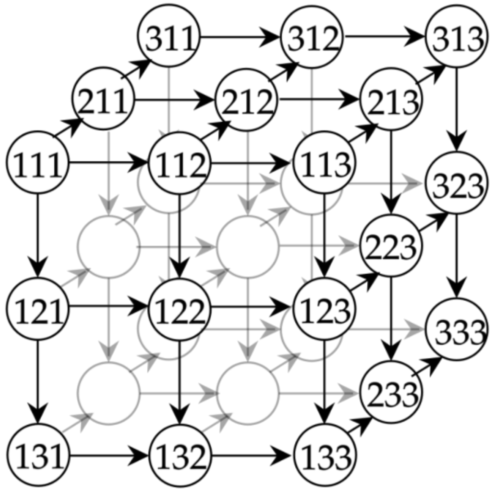
Introducing DAGs for Tensor

$\mathcal{P}$

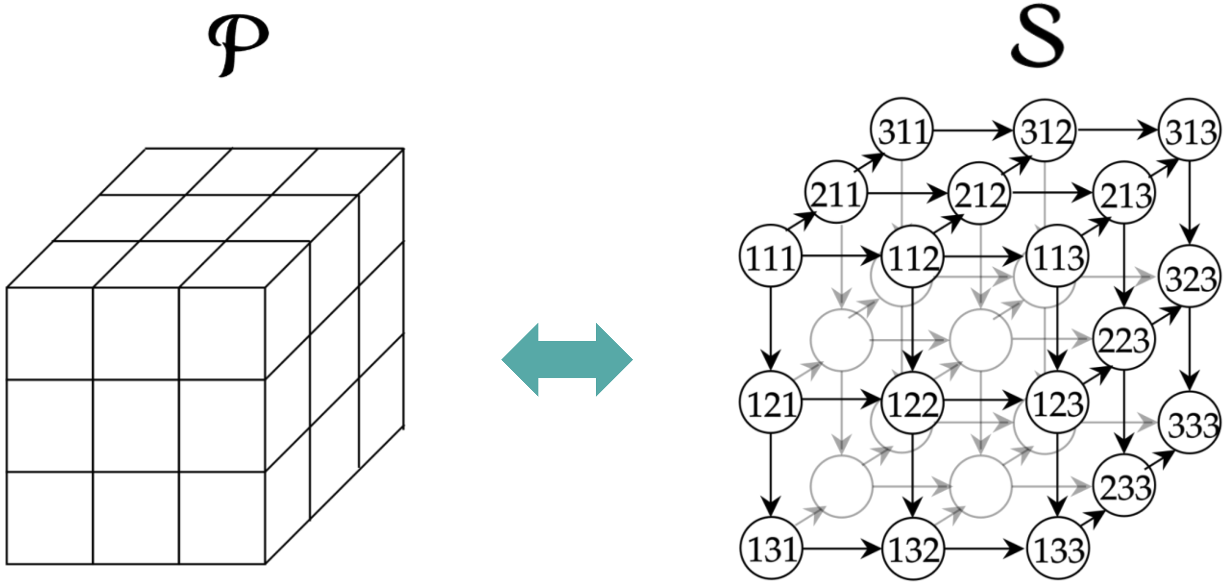


$$\sum_{i,j,k} \mathcal{P}_{ijk} = 1$$

$\mathcal{S}$



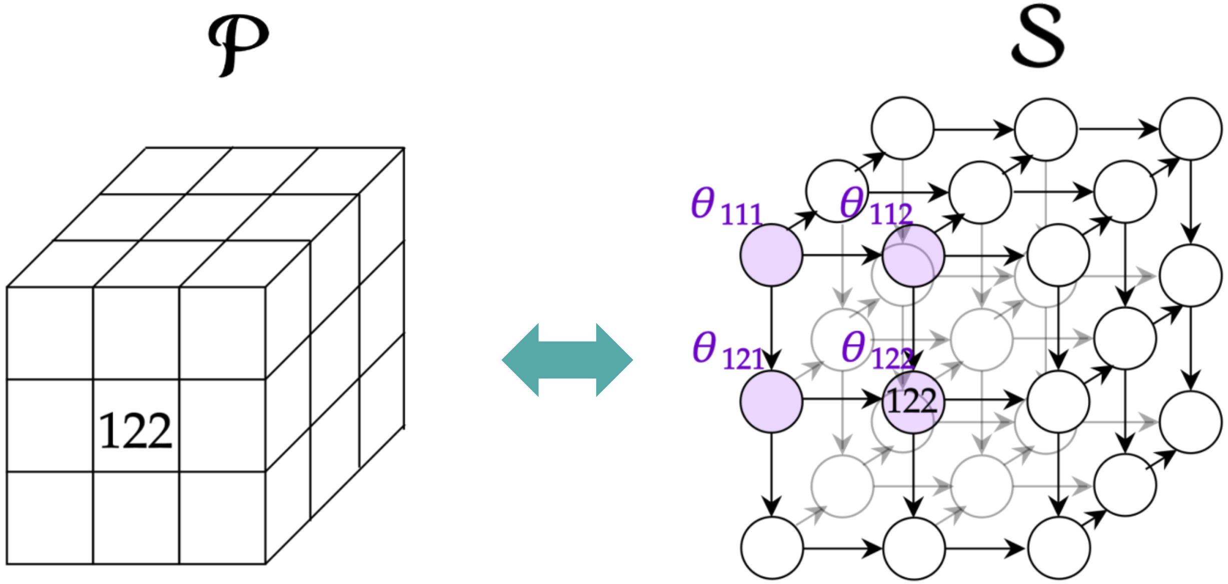
Introducing DAGs for Tensor



$$p(k,l,m) = \exp\left(\sum_{(s,t,u)\leq(k,l,m)} \theta_{stu}\right),$$

$$\eta_{klm} = \sum_{(k,l,m)\leq(s,t,u)} p(s,t,u).$$

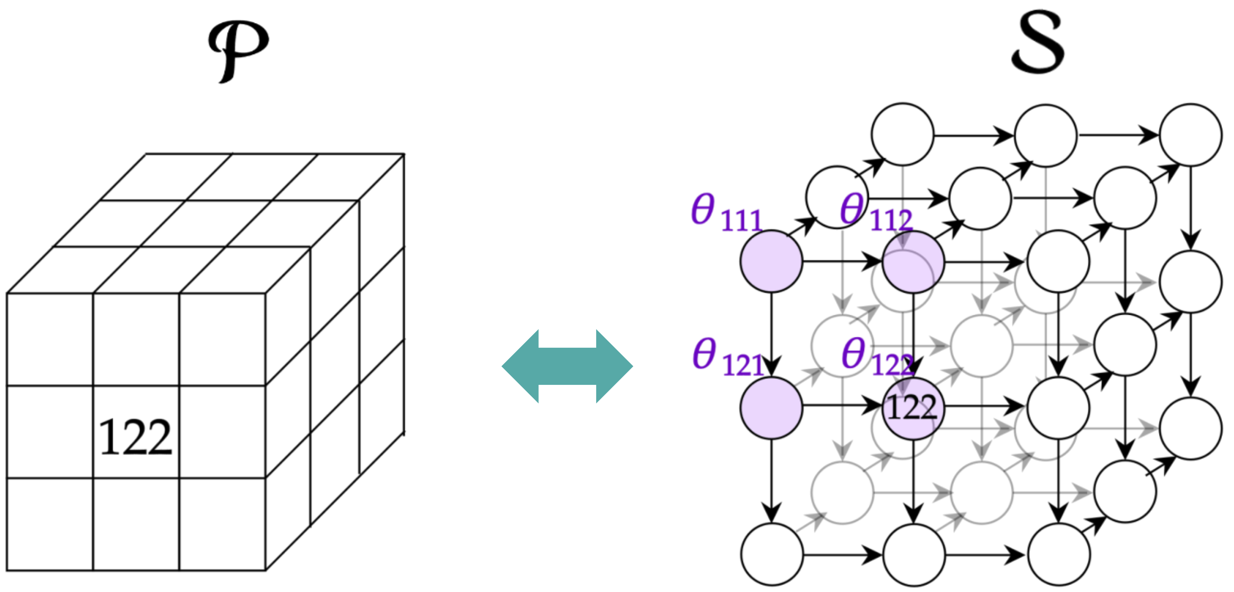
Describe a tensor with (θ,η)



$$p(k,l,m) = \exp\left(\sum_{(s,t,u)\le(k,l,m)} \theta_{stu} \right),$$

$$\eta_{klm} = \sum_{(k,l,m)\le(s,t,u)} p(s,t,u).$$

Describe a tensor with (θ,η)

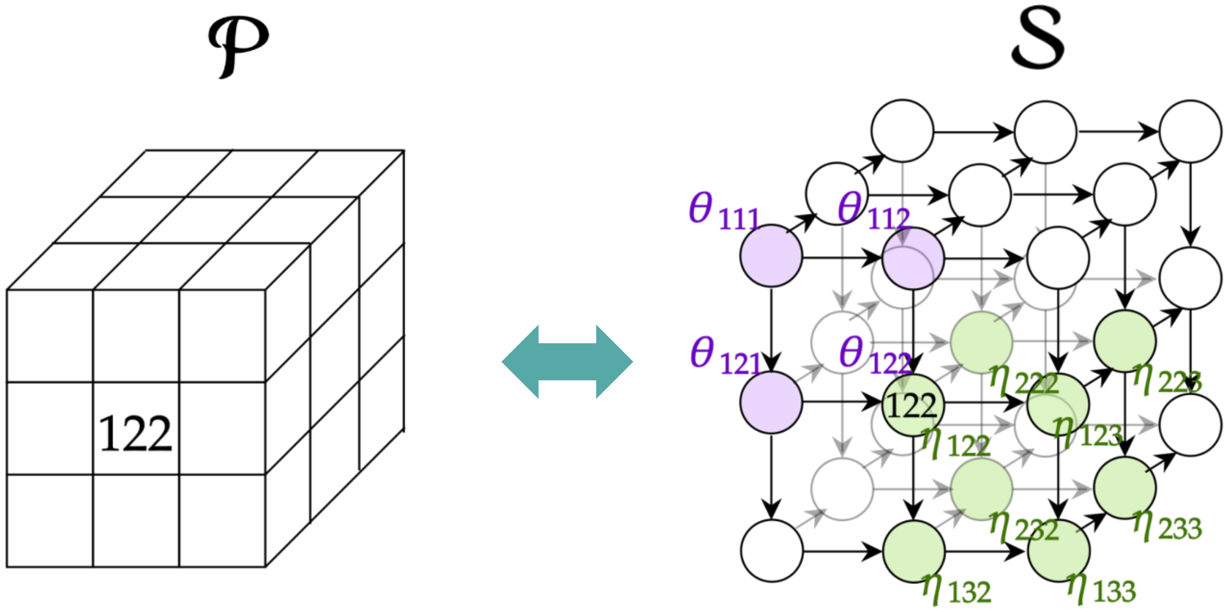


$$p(k,l,m) = \exp\left(\sum_{(s,t,u) \le (k,l,m)} \theta_{stu} \right),$$

$$\eta_{klm} = \sum_{(k,l,m) \le (s,t,u)} p(s,t,u).$$

$$p_\theta(1,2,2) = \exp(\theta_{111} + \theta_{112} + \theta_{121} + \theta_{122}),$$

Describe a tensor with (θ,η)



$$p(k,l,m) = \exp\left(\sum_{(s,t,u)\le(k,l,m)} \theta_{stu} \right),$$

$$\eta_{klm} = \sum_{(k,l,m)\le(s,t,u)} p(s,t,u).$$

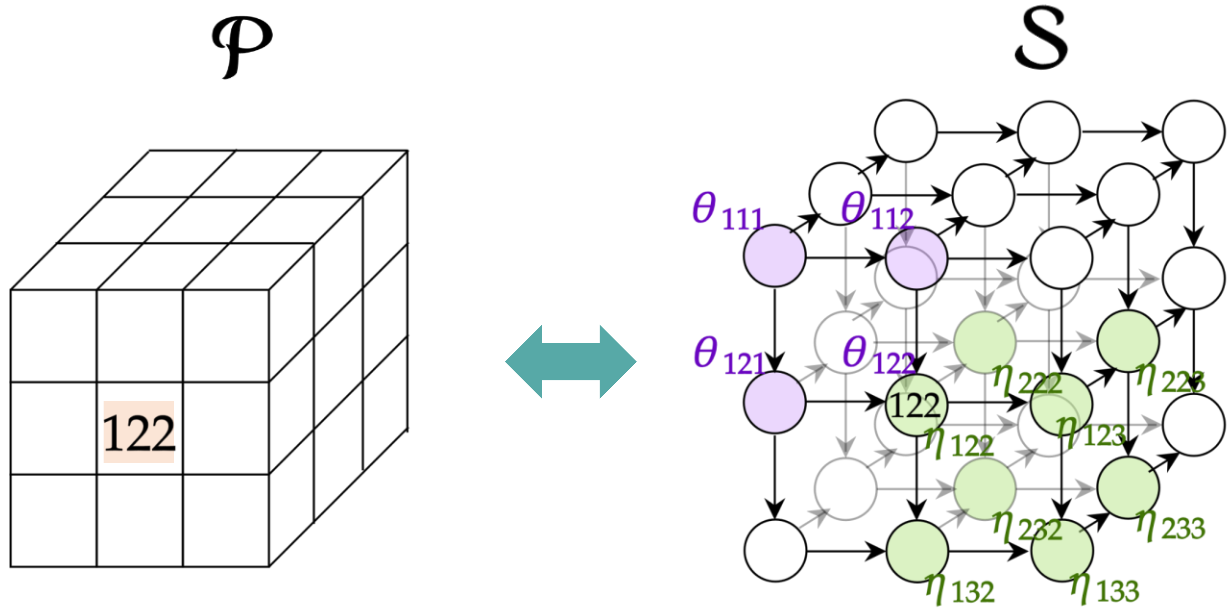
$$p_\theta(1,2,2) = \exp(\theta_{111} + \theta_{112} + \theta_{121} + \theta_{122}),$$

$$p_\eta(1,2,2) = \eta_{122} - \eta_{222} - \eta_{123} - \eta_{132} + \eta_{232} + \eta_{133} + \eta_{223} - \eta_{233}$$

Möbius inversion formula

$$p_\eta(i,j,k) = \sum_{(s,t,u)} \mu_{ijk}^{stu} \eta_{stu}$$

Describe a tensor with (θ,η)



$$p(k,l,m) = \exp\left(\sum_{(s,t,u)\le(k,l,m)} \theta_{stu}\right),$$

$$\eta_{klm} = \sum_{(k,l,m)\le(s,t,u)} p(s,t,u).$$

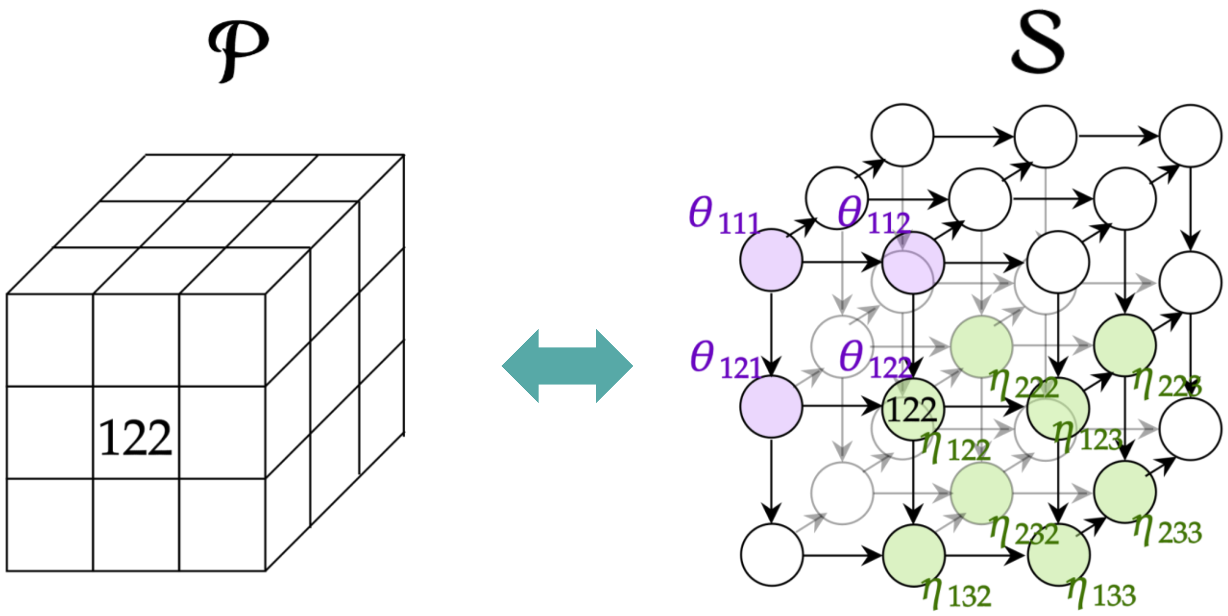
$$\mathcal{P}_{122} = p_\theta(1,2,2) = \exp(\theta_{111} + \theta_{112} + \theta_{121} + \theta_{122}),$$

$$\mathcal{P}_{122} = p_\eta(1,2,2) = \eta_{122} - \eta_{222} - \eta_{123} - \eta_{132} + \eta_{232} + \eta_{133} + \eta_{223} - \eta_{233}$$

Möbius inversion formula

$$p_\eta(i,j,k) = \sum_{(s,t,u)} \mu_{ijk}^{stu} \eta_{stu}$$

Describe a tensor with (θ,η)



$$p(k,l,m) = \exp\left(\sum_{(s,t,u)\le(k,l,m)} \theta_{stu} \right),$$

$$\eta_{klm} = \sum_{(k,l,m)\le(s,t,u)} p(s,t,u).$$

$$\mathcal{P}_{122} = p_\theta(1,2,2) = \exp(\theta_{111} + \theta_{112} + \theta_{121} + \theta_{122}),$$

$$\mathcal{P}_{122} = p_\eta(1,2,2) = \eta_{122} - \eta_{222} - \eta_{123} - \eta_{132} + \eta_{232} + \eta_{133} + \eta_{223} - \eta_{233}$$

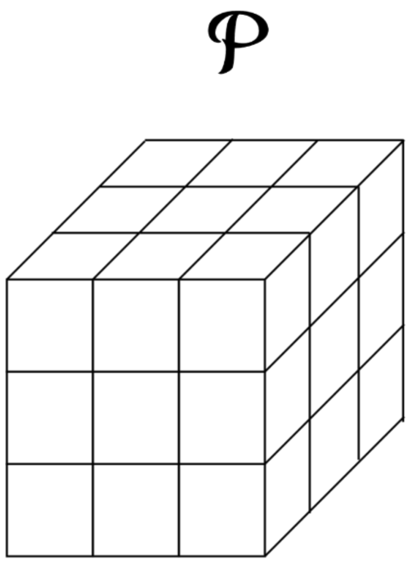
Relation between distribution and tensor

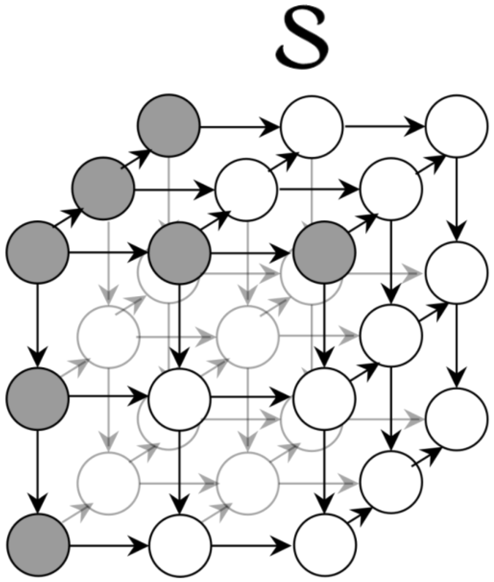
Random variables : i, j, k , **indices of the tensor**
Sample space : **index set**
Probability values : **tensor values** $\mathcal{P}_{ijk}$

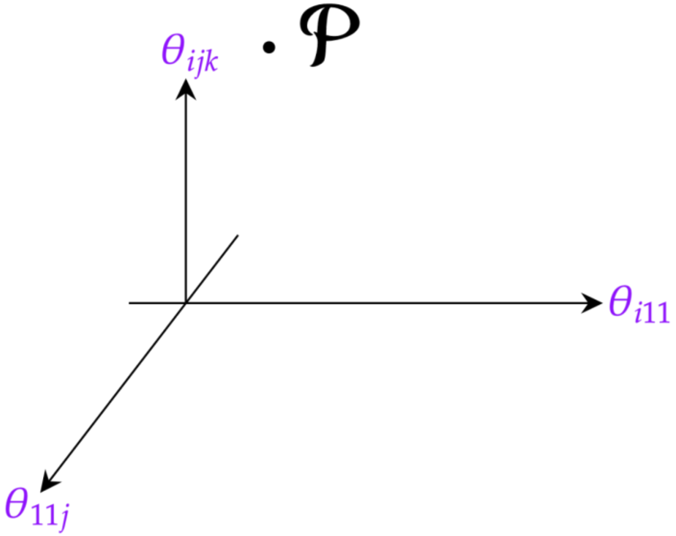
Möbius inversion formula

$$p_\eta(i,j,k) = \sum_{(s,t,u)} \mu_{ijk}^{stu} \eta_{stu}$$

One-body and many-body parameters





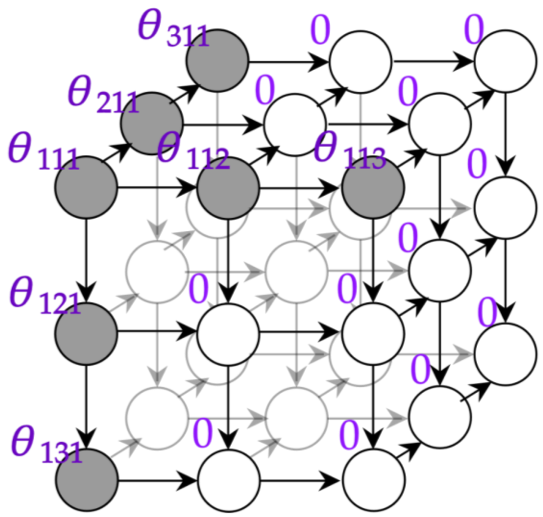


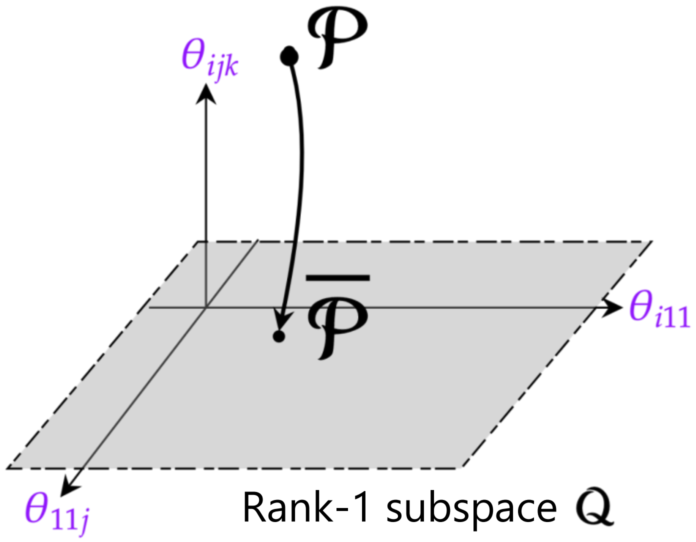
One-body parameter **Many-body parameter**

θ-representation of rank-1 tensor

Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.



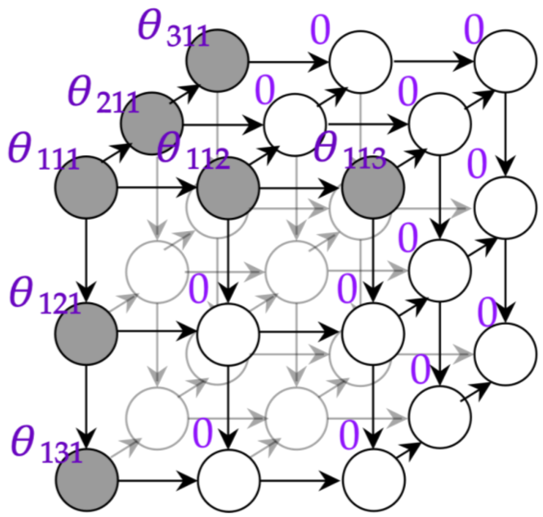


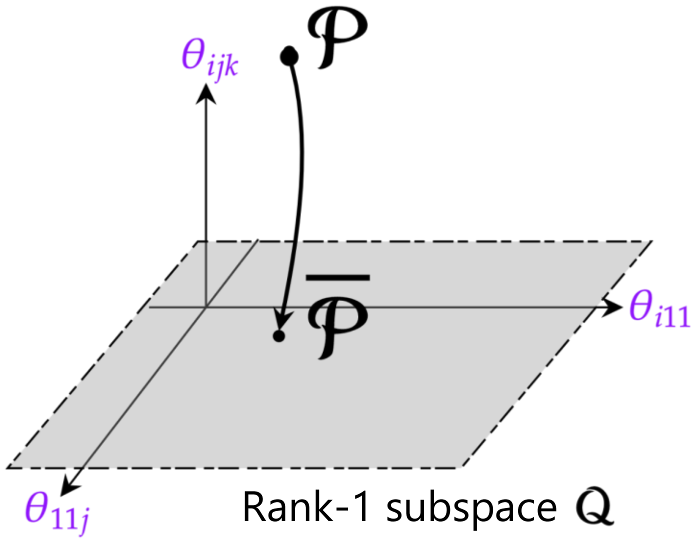
One-body parameter **Many-body parameter**

θ-representation of rank-1 tensor

Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.





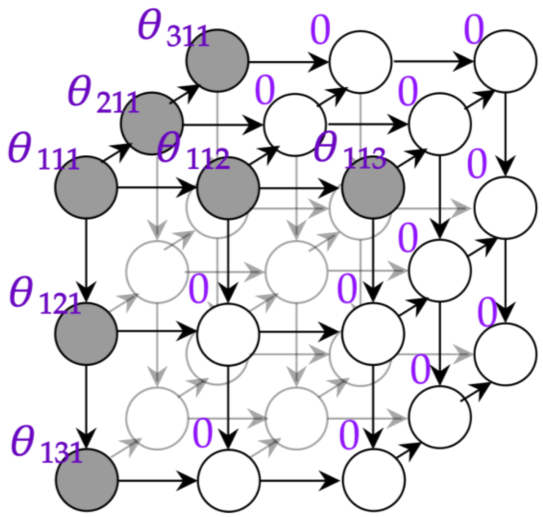
$\mathbb{Q}$ is *e*-flat. The projection is unique.

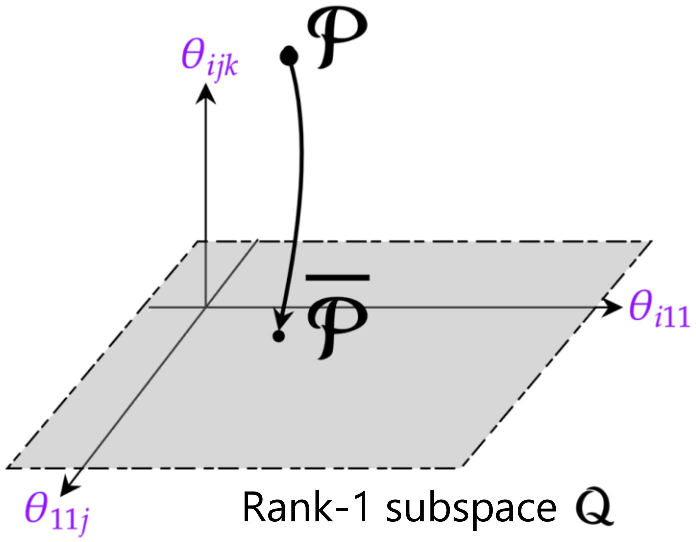
One-body parameter **Many-body parameter**

θ-representation of rank-1 tensor

Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.





Q is *e*-flat. The projection is unique.

One-body parameter **Many-body parameter**

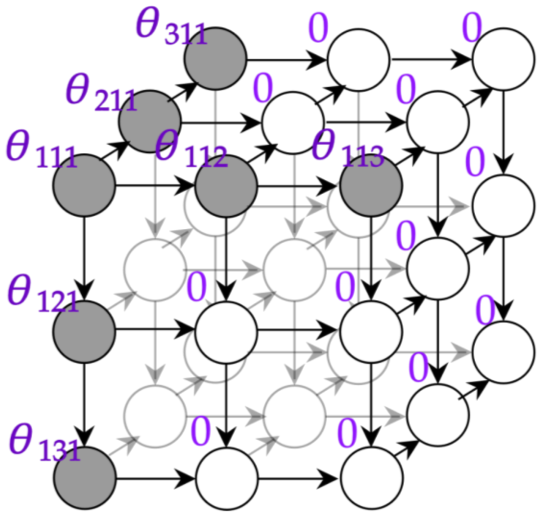
We can find the projection destination by a gradient-method.

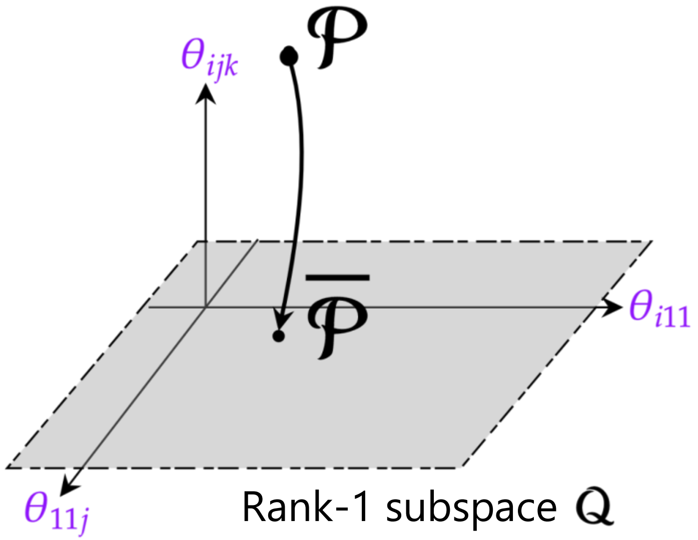
But gradient-methods require Appropriate settings for stopping criteria, learning rate, and initial values 😟

θ-representation of rank-1 tensor

Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.





$\mathbb{Q}$ is *e*-flat. The projection is unique.

● **One-body parameter** ○ **Many-body parameter**

We can find the projection destination by a gradient-method.

But gradient-methods require Appropriate settings for stopping criteria, learning rate, and initial values 😟

➡ **Let us describe the rank-1 condition with the η-parameter.**

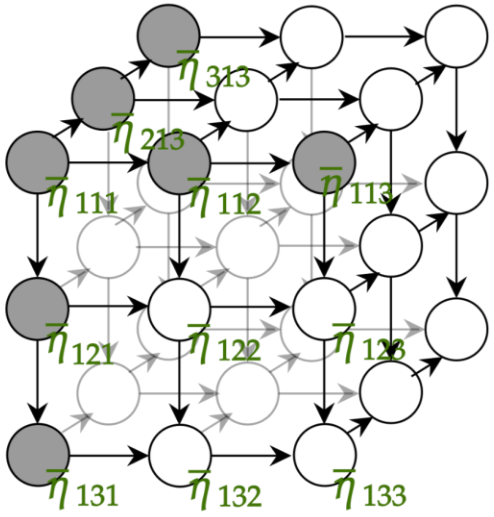
η-representation of rank-1 tensor

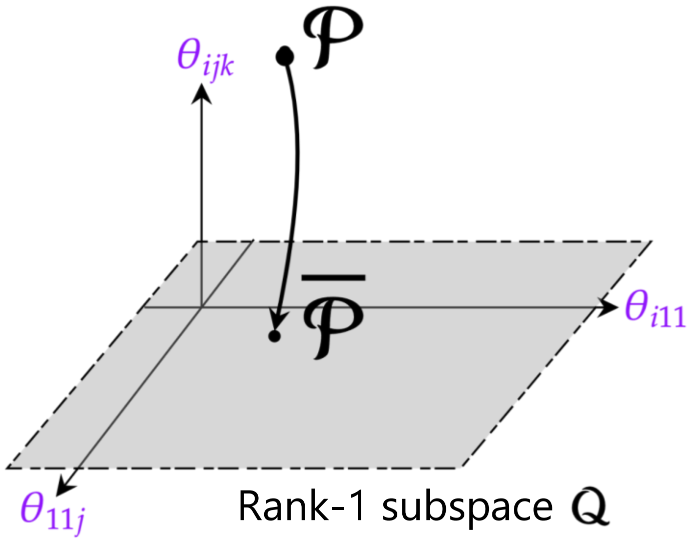
Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.

Rank-1 condition ($\boldsymbol{\eta}$- representation)

$$\eta_{ijk} = \eta_{i11}\eta_{1j1}\eta_{11k}$$





One-body parameter   Many-body parameter

η_{ijk} : η-parameter before the projection.

$\bar{\eta}_{ijk}$: η-parameter after the projection.

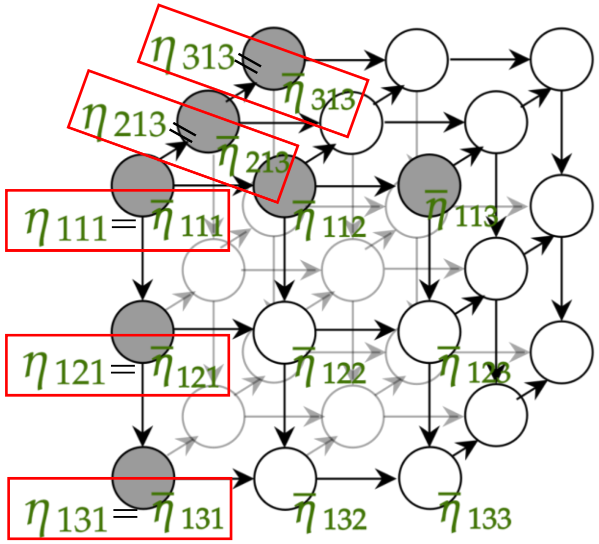
η-representation of rank-1 tensor

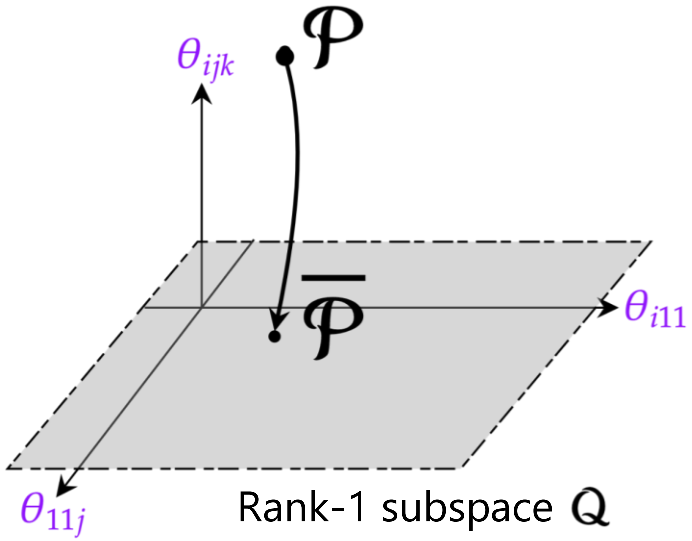
Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.

Rank-1 condition ($\boldsymbol{\eta}$- representation)

$$\eta_{ijk} = \eta_{i11}\eta_{1j1}\eta_{11k}$$





● **One-body parameter** ○ **Many-body parameter**

η_{ijk} : η-parameter before the projection.

$\bar{\eta}_{ijk}$: η-parameter after the projection.

The m-projection does not change one-body $\boldsymbol{\eta}$-parameter

Shun-ichi Amari, Information Geometry and Its Applications, 2008, Theorem 11.6

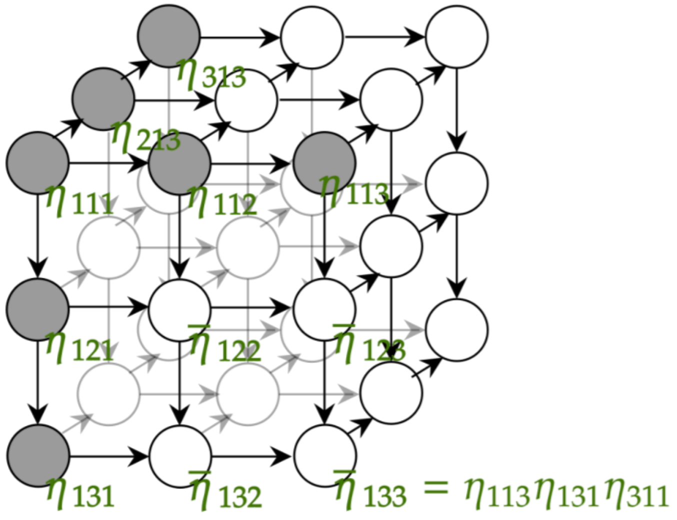
Find the best rank-1 approximation

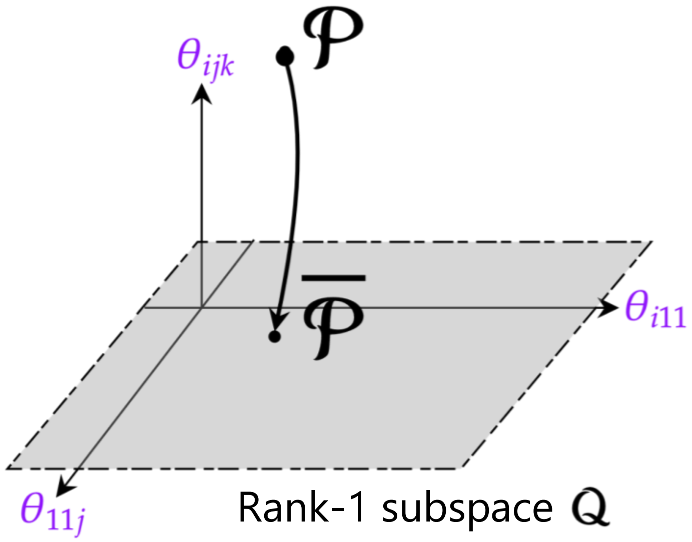
Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.

Rank-1 condition ($\boldsymbol{\eta}$- representation)

$$\bar{\eta}_{ijk} = \bar{\eta}_{i11}\bar{\eta}_{1j1}\bar{\eta}_{11k}$$





● **One-body parameter** ○ **Many-body parameter**

η_{ijk} : η-parameter before the projection.

$\bar{\eta}_{ijk}$: η-parameter after the projection.

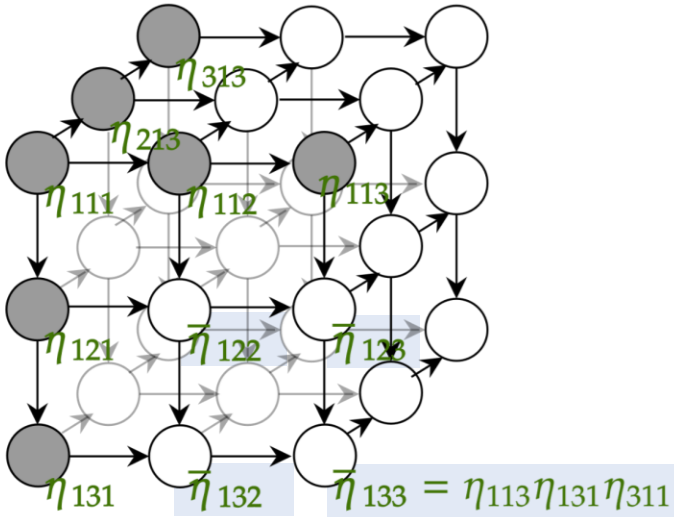
Find the best rank-1 approximation

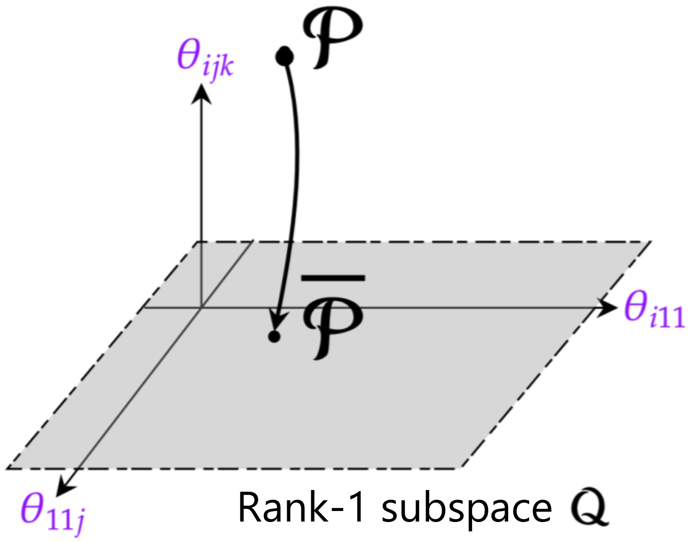
Rank-1 condition ($\boldsymbol{\theta}$-representation)

Its all many-body θ-parameters are 0.

Rank-1 condition ($\boldsymbol{\eta}$- representation)

$$\bar{\eta}_{ijk} = \bar{\eta}_{i11}\bar{\eta}_{1j1}\bar{\eta}_{11k}$$
$$= \eta_{i11}\eta_{1j1}\eta_{11k}$$





One-body parameter **Many-body parameter**

η_{ijk} : η-parameter before the projection.
$\bar{\eta}_{ijk}$: η-parameter after the projection.

Möbius inversion formula

$$p_{\eta}(i,j,k) = \sum_{(s,t,u)} \mu_{ijk}^{stu}\eta_{stu}$$

All $\boldsymbol{\eta}$-parameters after the projection are identified.
Using inversion formula, we found the projection destination.

Mean-field approximation and rank-1 approximation

Best rank-1 tensor formula for minimizing KL divergence ($d = 3$)

The best rank-1 approximation of $\mathcal{P} \in \mathbb{R}_{>0}^{I\times J\times K}$ is given as

$$\overline{\mathcal{P}}_{ijk} = \left(\sum_{j'=1}^{J}\sum_{k'=1}^{K}\mathcal{P}_{ij'k'}\right)\left(\sum_{k'=1}^{K}\sum_{i'=1}^{I}\mathcal{P}_{i'jk'}\right)\left(\sum_{i'=1}^{I}\sum_{j'=1}^{J}\mathcal{P}_{i'j'k}\right)$$

which minimizes KL divergence from $\mathcal{P}$.

We reproduce the result in K.Huang, et al. "Kullback-Leibler principal component for tensors is not NP-hard." *ACSSC* 2017

9:00

Mean-field approximation and rank-1 approximation

Best rank-1 tensor formula for minimizing KL divergence ($d = 3$)

The best rank-1 approximation of $\mathcal{P} \in \mathbb{R}_{>0}^{I \times J \times K}$ is given as

$$\overline{\mathcal{P}}_{ijk} = \left(\sum_{j'=1}^{J} \sum_{k'=1}^{K} \mathcal{P}_{ij'k'} \right) \left(\sum_{k'=1}^{K} \sum_{i'=1}^{I} \mathcal{P}_{i'jk'} \right) \left(\sum_{i'=1}^{I} \sum_{j'=1}^{J} \mathcal{P}_{i'j'k} \right)$$

which minimizes KL divergence from $\mathcal{P}$.

We reproduce the result in K.Huang, et al. "Kullback-Leibler principal component for tensors is not NP-hard." *ACSSC* 2017

By the way, Frobenius error minimization is **NP-hard**

Mean-field approximation and rank-1 approximation

Best rank-1 tensor formula for minimizing KL divergence ($d = 3$)

The best rank-1 approximation of $\mathcal{P} \in \mathbb{R}_{>0}^{I \times J \times K}$ is given as

$$\overline{\mathcal{P}}_{ijk} = \left(\sum_{j'=1}^{J}\sum_{k'=1}^{K}\mathcal{P}_{ij'k'}\right)\left(\sum_{k'=1}^{K}\sum_{i'=1}^{I}\mathcal{P}_{i'jk'}\right)\left(\sum_{i'=1}^{I}\sum_{j'=1}^{J}\mathcal{P}_{i'j'k}\right)$$

Normalized vector depending on only i — Normalized vector depending on only j — Normalized vector depending on only k

which minimizes KL divergence from $\mathcal{P}$.

We reproduce the result in K.Huang, et al. "Kullback-Leibler principal component for tensors is not NP-hard." *ACSSC* 2017

By the way, Frobenius error minimization is **NP-hard**

A tensor with d indices is a joint distribution with d random variables.
A vector with only 1 index is an independent distribution with only one random variable.

Mean-field approximation and rank-1 approximation

Best rank-1 tensor formula for minimizing KL divergence ($d = 3$)

By the way, Frobenius error minimization is **NP-hard**

The best rank-1 approximation of $\mathcal{P} \in \mathbb{R}_{>0}^{I \times J \times K}$ is given as

$$\overline{\mathcal{P}}_{ijk} = \left(\sum_{j'=1}^{J} \sum_{k'=1}^{K} \mathcal{P}_{ij'k'} \right) \left(\sum_{k'=1}^{K} \sum_{i'=1}^{I} \mathcal{P}_{i'jk'} \right) \left(\sum_{i'=1}^{I} \sum_{j'=1}^{J} \mathcal{P}_{i'j'k} \right)$$

Normalized vector depending on only i

Normalized vector depending on only j

Normalized vector depending on only k

which minimizes KL divergence from $\mathcal{P}$.

We reproduce the result in K.Huang, et al. "Kullback-Leibler principal component for tensors is not NP-hard." *ACSSC* 2017

A tensor with d indices is a joint distribution with d random variables.
A vector with only 1 index is an independent distribution with only one random variable.

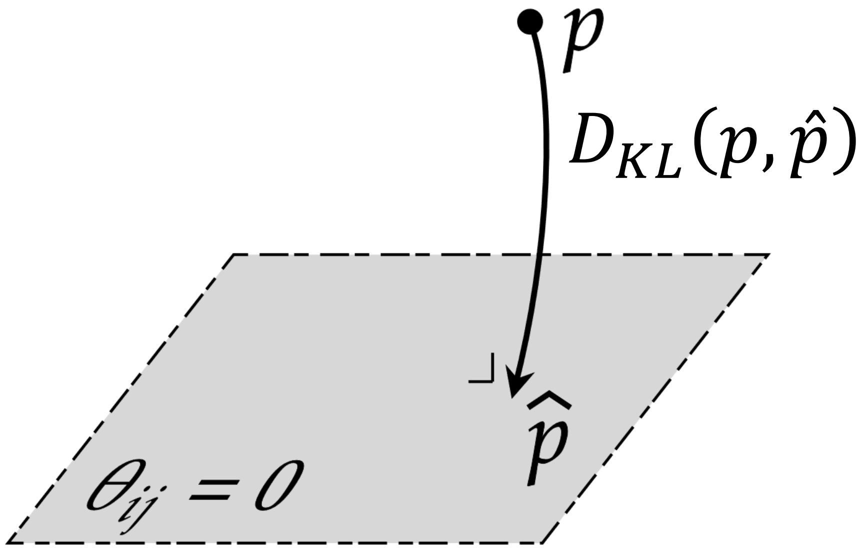
Rank-1 approximation approximates a joint distribution by a product of independent distributions.

Mean-field approximation **: a methodology in physics for reducing a many-body problem to a one-body problem.**

Mean-field approximation and rank-1 approximation

MFA of Boltzmann-machine

$$p(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left[\underbrace{\sum_i \theta_i x_i}_{\text{Bias}} + \underbrace{\sum_{i<j} \theta_{ij} x_i x_j}_{\text{Interaction}}\right] \qquad \eta_i = \sum_{x_1=0}^{1} \cdots \sum_{x_n=0}^{1} x_i p(\boldsymbol{x})$$

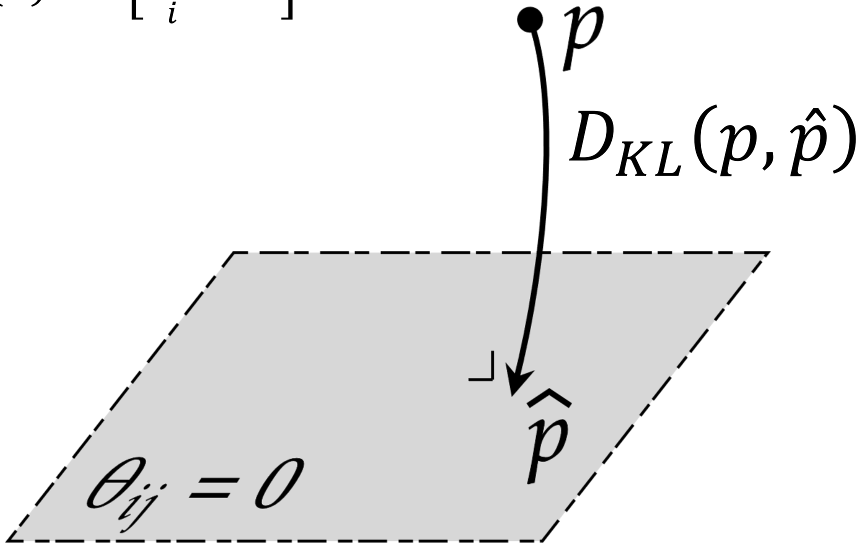


Mean-field approximation and rank-1 approximation

MFA of Boltzmann-machine

$$p(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left[\underbrace{\sum_i \theta_i x_i}_{\text{Bias}} + \underbrace{\sum_{i<j} \theta_{ij} x_i x_j}_{\text{Interaction}}\right] \qquad \eta_i = \sum_{x_1=0}^{1} \cdots \sum_{x_n=0}^{1} x_i p(\boldsymbol{x})$$

$$= \frac{1}{Z(\boldsymbol{\theta})} \exp\left[\sum_i \theta_i x_i\right] = p(x_1) \dots p(x_n)$$



Mean-field approximation and rank-1 approximation

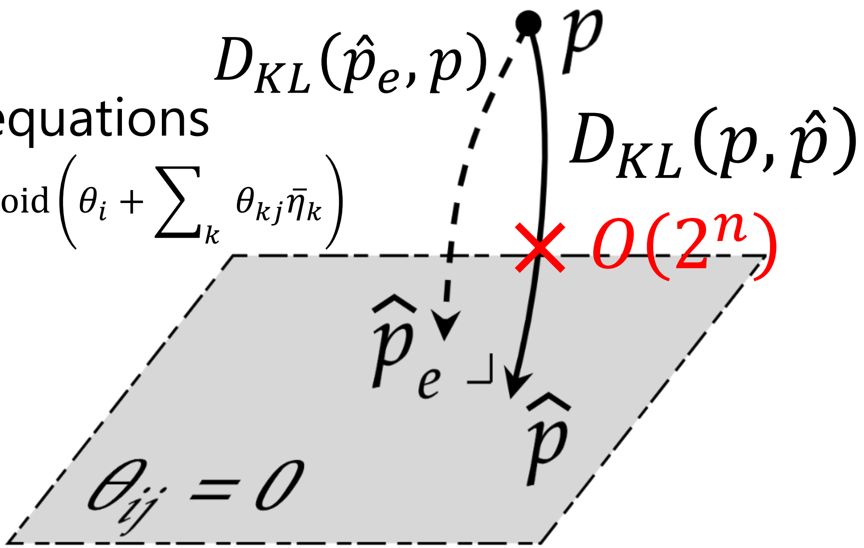
MFA of Boltzmann-machine

$$p(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\Big[\underbrace{\sum_i \theta_i x_i}_{\text{Bias}} + \underbrace{\sum_{i<j} \theta_{ij} x_i x_j}_{\text{Interaction}}\Big]$$

$$\eta_i = \overbrace{\sum_{x_1=0}^{1} \cdots \sum_{x_n=0}^{1}}^{O(2^n)} x_i p(\boldsymbol{x})$$

MF equations

$$\bar{\eta}_i = \mathrm{sigmoid}\left(\theta_i + \sum_k \theta_{kj} \bar{\eta}_k\right)$$



Mean-field approximation and rank-1 approximation

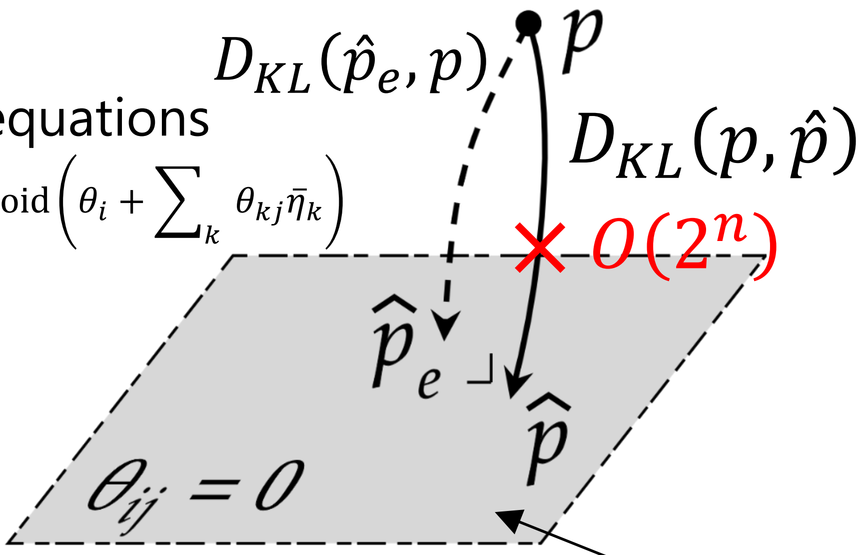
MFA of Boltzmann-machine

$$p(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})}\exp\Big[\underbrace{\sum_i \theta_i x_i}_{\text{Bias}} + \underbrace{\sum_{i<j} \theta_{ij} x_i x_j}_{\text{Interaction}}\Big]$$

$$\eta_i = \overbrace{\sum_{x_1=0}^{1} \cdots \sum_{x_n=0}^{1}}^{O(2^n)} x_i p(\boldsymbol{x})$$

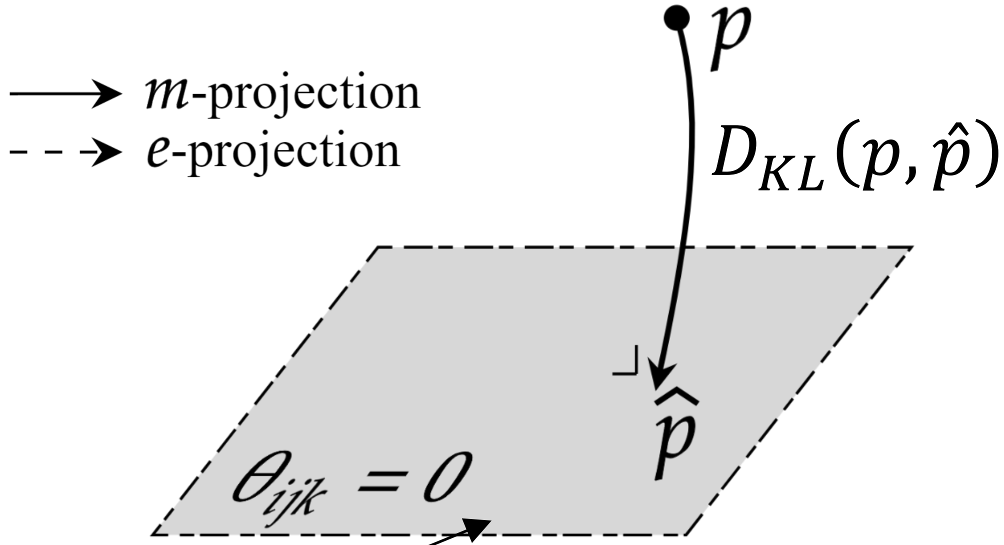
MF equations

$$\bar{\eta}_i = \mathrm{sigmoid}\Big(\theta_i + \sum_k \theta_{kj}\bar{\eta}_k\Big)$$



Rank-1 approximation

$$p_\theta(i,j,k) = \exp\Big[\sum_{i'=1}^{i}\sum_{j'=1}^{j}\sum_{k'=1}^{k} \theta_{i'j'k'}\Big] \qquad \eta_{i11} = \sum_{j'=1}^{J}\sum_{k'=1}^{K} \mathcal{P}_{ij'k'}$$



Set of products of independent distributions

Mean-field approximation and rank-1 approximation

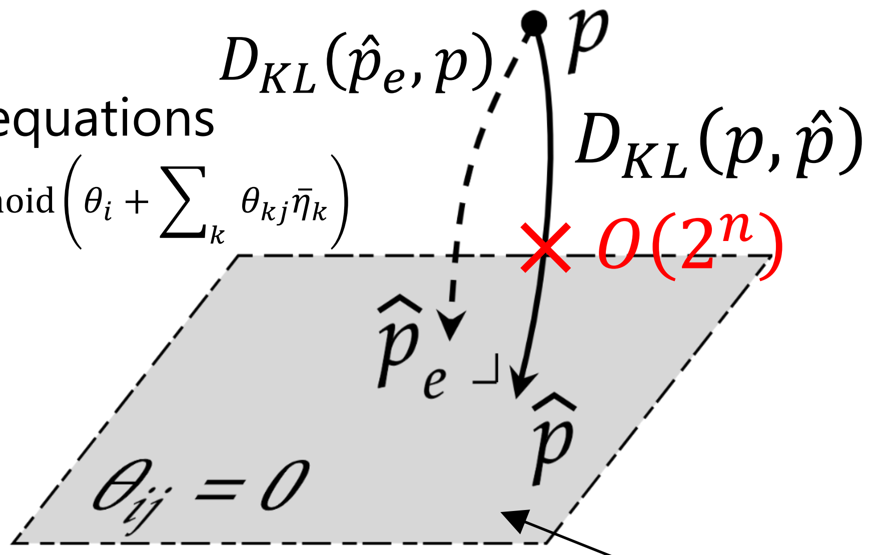
MFA of Boltzmann-machine

$$p(\boldsymbol{x}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left[\underbrace{\sum_i \theta_i x_i}_{\text{Bias}} + \underbrace{\sum_{i<j} \theta_{ij} x_i x_j}_{\text{Interaction}}\right]$$

$$\eta_i = \overbrace{\sum_{x_1=0}^{1} \cdots \sum_{x_n=0}^{1}}^{O(2^n)} x_i p(\boldsymbol{x})$$

MF equations

$$\bar{\eta}_i = \text{sigmoid}\left(\theta_i + \sum_k \theta_{kj} \bar{\eta}_k\right)$$

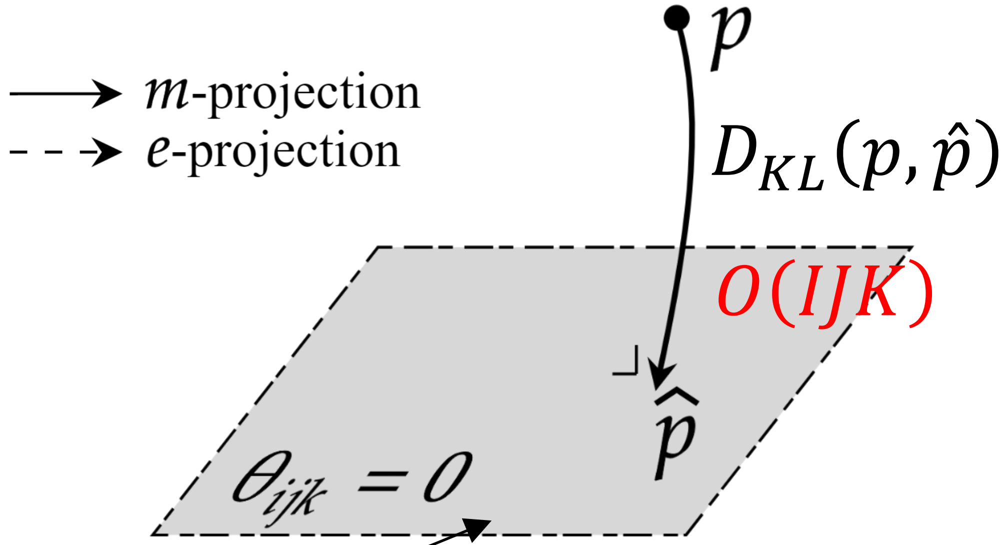


Rank-1 approximation

$$p_\theta(i,j,k) = \exp\left[\sum_{i'=1}^{i} \sum_{j'=1}^{j} \sum_{k'=1}^{k} \theta_{i'j'k'}\right]$$

Computable

$$\eta_{i11} = \sum_{j'=1}^{J} \sum_{k'=1}^{K} \mathcal{P}_{ij'k'}$$



Set of products of independent distributions

Mean-field approximation and rank-1 approximation

	Minimizing KL divergence *m*-projection	Minimizing inverse-KL divergence *e*-projection
Mean-field Approximation of BM Projection onto *e*-flat space	impossible $O(2^n)$ unique	$\eta_i = \sigma\left(\theta_i + \sum_k \theta_{kj}\eta_k\right)$ not unique
Rank-1 approximation Projection onto *e*-flat space	**Closed-formula** unique	

Contents

☐ Motivation, Strategy, and Contributions

☐ Introduction of log-linear model on DAG

☐ The best rank-1 approximation formula

☐ Legendre Tucker-Rank Reduction(LTR)

github.com/gkazunii/ Legendre-tucker-rank-reduction

☐ The best rank-1 NMMF

☐A1GM: faster rank-1 missing NMF

github.com/gkazunii/A1GM

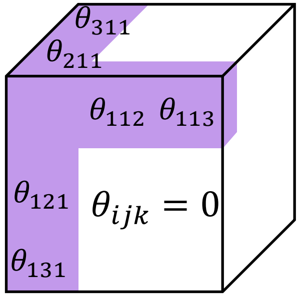
☐ Theoretical Remarks

☐ Conclusion

Formulate Tucker rank reduction by relaxing the rank-1 condition

Rank-1 condition (**$\boldsymbol{\theta}$**-representation)

$$\mathrm{rank}(\mathcal{P}) = 1 \Leftrightarrow \text{ its all many} - \text{body } \theta \text{ parameters are } 0$$

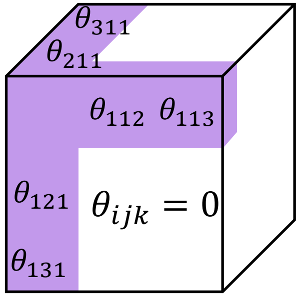


Expand the tensor by focusing on the m-th axis into a rectangular matrix $\theta^{(m)}$ (mode-m expansion)

Formulate Tucker rank reduction by relaxing the rank-1 condition

Rank-1 condition (*θ*-representation)

$$\mathrm{rank}(\mathcal{P}) = 1 \Leftrightarrow \text{ its all many} - \text{body } \theta \text{ parameters are } 0$$



Expand the tensor by focusing on the m-th axis into a rectangular matrix $\theta^{(m)}$ (mode-m expansion)

$$\theta^{(1)} = \begin{bmatrix} \theta_{111} & \theta_{121} & \theta_{131} & \theta_{112} & 0 & 0 & \theta_{113} & 0 & 0 \\ \theta_{211} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{311} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\theta^{(2)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{112} & 0 & 0 & \theta_{311} & 0 & 0 \\ \theta_{121} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{131} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

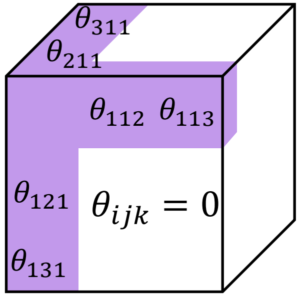
$$\theta^{(3)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{121} & 0 & 0 & \theta_{131} & 0 & 0 \\ \theta_{112} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{113} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Rank (1,1,1)

Formulate Tucker rank reduction by relaxing the rank-1 condition

Rank-1 condition ($\boldsymbol{\theta}$-representation)

$\mathrm{rank}(\mathcal{P}) = 1 \Leftrightarrow$ its all many − body θ parameters are 0



Expand the tensor by focusing on the m-th axis into a rectangular matrix $\theta^{(m)}$ (mode-m expansion)

$$\theta^{(1)} = \begin{bmatrix} \theta_{111} & \theta_{121} & \theta_{131} & \theta_{112} & 0 & 0 & \theta_{113} & 0 & 0 \\ \theta_{211} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{311} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Two bingos

$$\theta^{(2)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{112} & 0 & 0 & \theta_{311} & 0 & 0 \\ \theta_{121} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{131} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Two bingos

$$\theta^{(3)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{121} & 0 & 0 & \theta_{131} & 0 & 0 \\ \theta_{112} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{113} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

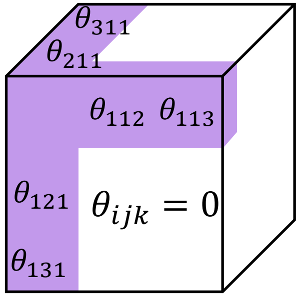
Two bingos

Rank (1,1,1)

Formulate Tucker rank reduction by relaxing the rank-1 condition

Rank-1 condition ($\boldsymbol{\theta}$-representation)

$$\mathrm{rank}(\mathcal{P}) = 1 \Leftrightarrow \text{ its all many}-\text{body } \theta \text{ parameters are } 0$$



Expand the tensor by focusing on the m-th axis into a rectangular matrix $\theta^{(m)}$ (mode-m expansion)

The first row and first column are the scaling factors

$$\theta^{(1)} = \begin{bmatrix} \theta_{111} & \theta_{121} & \theta_{131} & \theta_{112} & 0 & 0 & \theta_{113} & 0 & 0 \\ \theta_{211} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{311} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Two bingos

$$\theta^{(2)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{112} & 0 & 0 & \theta_{311} & 0 & 0 \\ \theta_{121} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{131} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Two bingos

$$\theta^{(3)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{121} & 0 & 0 & \theta_{131} & 0 & 0 \\ \theta_{112} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{113} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Two bingos

Rank (1,1,1)

The relationship between bingo and rank

$$\theta^{(1)} = \begin{bmatrix} \theta_{111} & \theta_{121} & \theta_{131} & \theta_{112} & 0 & 0 & \theta_{113} & 0 & 0 \\ \theta_{211} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{311} & \theta_{321} & \theta_{331} & \theta_{312} & \theta_{322} & \theta_{332} & \theta_{313} & \theta_{323} & \theta_{333} \end{bmatrix}$$

One bingo

$$\theta^{(2)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{112} & 0 & \theta_{312} & \theta_{311} & 0 & \theta_{313} \\ \theta_{121} & 0 & \theta_{321} & 0 & 0 & \theta_{322} & 0 & 0 & \theta_{323} \\ \theta_{131} & 0 & \theta_{331} & 0 & 0 & \theta_{332} & 0 & 0 & \theta_{333} \end{bmatrix}$$

No bingo

$$\theta^{(3)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{121} & 0 & \theta_{321} & \theta_{131} & 0 & \theta_{331} \\ \theta_{112} & 0 & \theta_{312} & 0 & 0 & \theta_{322} & 0 & 0 & \theta_{332} \\ \theta_{113} & 0 & \theta_{313} & 0 & 0 & \theta_{323} & 0 & 0 & \theta_{333} \end{bmatrix}$$

No bingo

Rank (2,3,3)

The relationship between bingo and rank

$$\theta^{(1)} = \begin{bmatrix} \theta_{111} & \theta_{121} & \theta_{131} & \theta_{112} & 0 & 0 & \theta_{113} & 0 & 0 \\ \theta_{211} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{311} & \theta_{321} & \theta_{331} & \theta_{312} & \theta_{322} & \theta_{332} & \theta_{313} & \theta_{323} & \theta_{333} \end{bmatrix}$$

One bingo

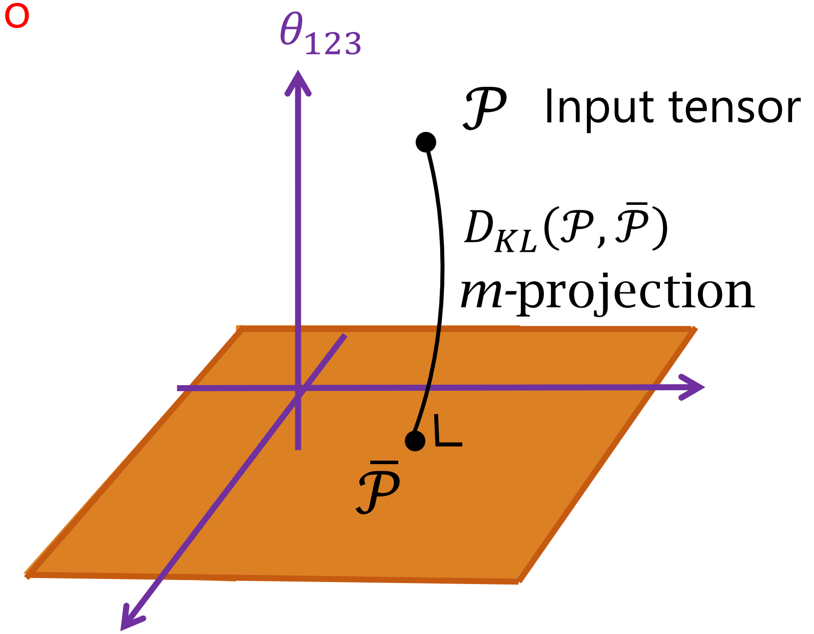
$$\theta^{(2)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{112} & 0 & \theta_{312} & \theta_{311} & 0 & \theta_{313} \\ \theta_{121} & 0 & \theta_{321} & 0 & 0 & \theta_{322} & 0 & 0 & \theta_{323} \\ \theta_{131} & 0 & \theta_{331} & 0 & 0 & \theta_{332} & 0 & 0 & \theta_{333} \end{bmatrix}$$

No bingo

$$\theta^{(3)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{121} & 0 & \theta_{321} & \theta_{131} & 0 & \theta_{331} \\ \theta_{112} & 0 & \theta_{312} & 0 & 0 & \theta_{322} & 0 & 0 & \theta_{332} \\ \theta_{113} & 0 & \theta_{313} & 0 & 0 & \theta_{323} & 0 & 0 & \theta_{333} \end{bmatrix}$$

No bingo

Rank (2,3,3)



Subspace with one bingo in the mode-1 direction $\mathcal{B}^{(1)}$

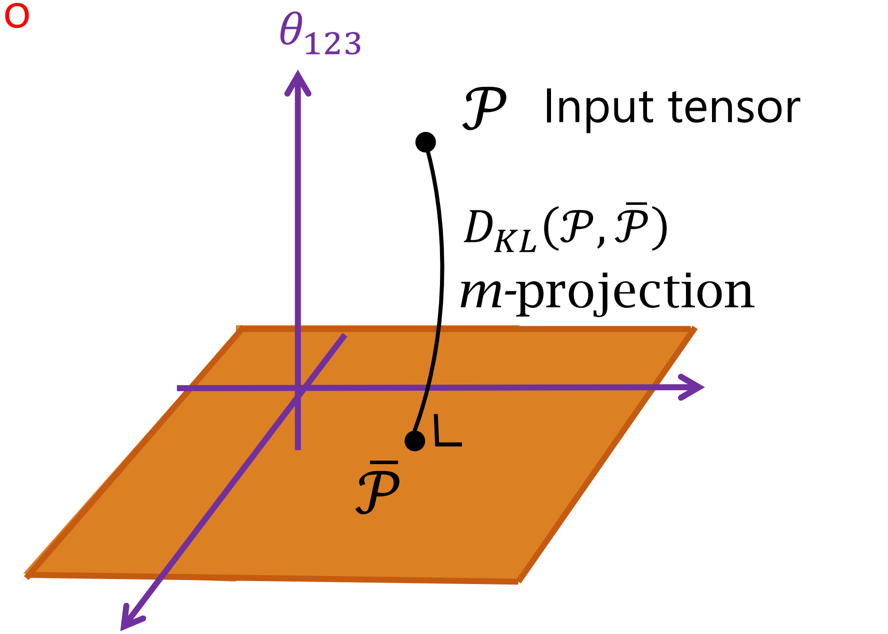
The relationship between bingo and rank

$$\theta^{(1)} = \begin{bmatrix} \theta_{111} & \theta_{121} & \theta_{131} & \theta_{112} & 0 & 0 & \theta_{113} & 0 & 0 \\ \theta_{211} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \theta_{311} & \theta_{321} & \theta_{331} & \theta_{312} & \theta_{322} & \theta_{332} & \theta_{313} & \theta_{323} & \theta_{333} \end{bmatrix}$$ → One bingo

$$\theta^{(2)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{112} & 0 & \theta_{312} & \theta_{311} & 0 & \theta_{313} \\ \theta_{121} & 0 & \theta_{321} & 0 & 0 & \theta_{322} & 0 & 0 & \theta_{323} \\ \theta_{131} & 0 & \theta_{331} & 0 & 0 & \theta_{332} & 0 & 0 & \theta_{333} \end{bmatrix}$$ No bingo

$$\theta^{(3)} = \begin{bmatrix} \theta_{111} & \theta_{211} & \theta_{311} & \theta_{121} & 0 & \theta_{321} & \theta_{131} & 0 & \theta_{331} \\ \theta_{112} & 0 & \theta_{312} & 0 & 0 & \theta_{322} & 0 & 0 & \theta_{332} \\ \theta_{113} & 0 & \theta_{313} & 0 & 0 & \theta_{323} & 0 & 0 & \theta_{333} \end{bmatrix}$$ No bingo

Rank (2,3,3)

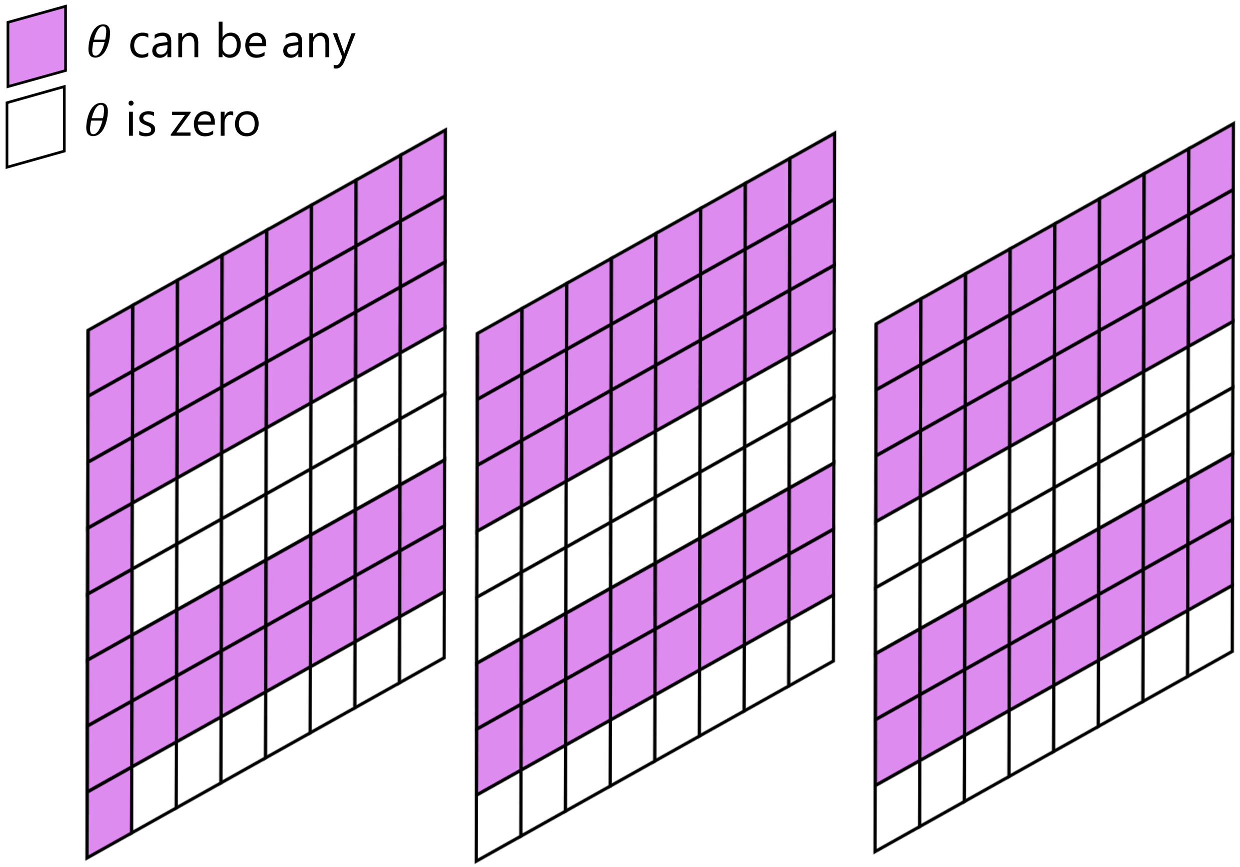


Subspace with one bingo in the mode-1 direction $\mathcal{B}^{(1)}$

Bingo rule ($d = 3$)

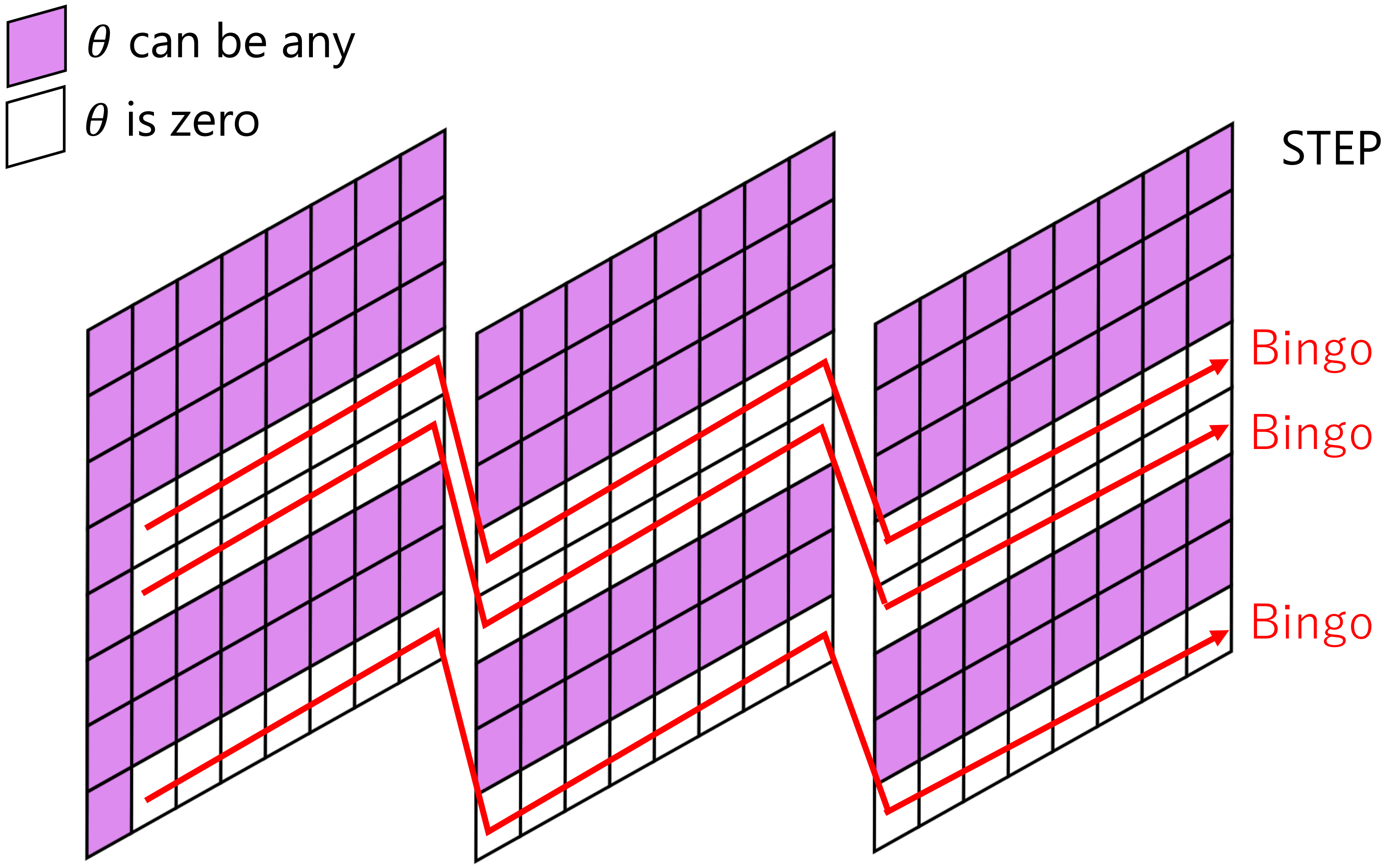
The mode-k expansion $\theta^{(k)}$ of the natural parameter of a tensor $\mathcal{P} \in \mathbb{R}_{>0}^{I_1 \times I_2 \times I_3}$ has b_k bingos
$\Rightarrow \mathrm{rank}(\mathcal{P}) \leq (I_1 - b_1, I_2 - b_2, I_3 - b_3)$

Example: Reduce the rank of (8,8,3) tensor to (5,8,3) or less



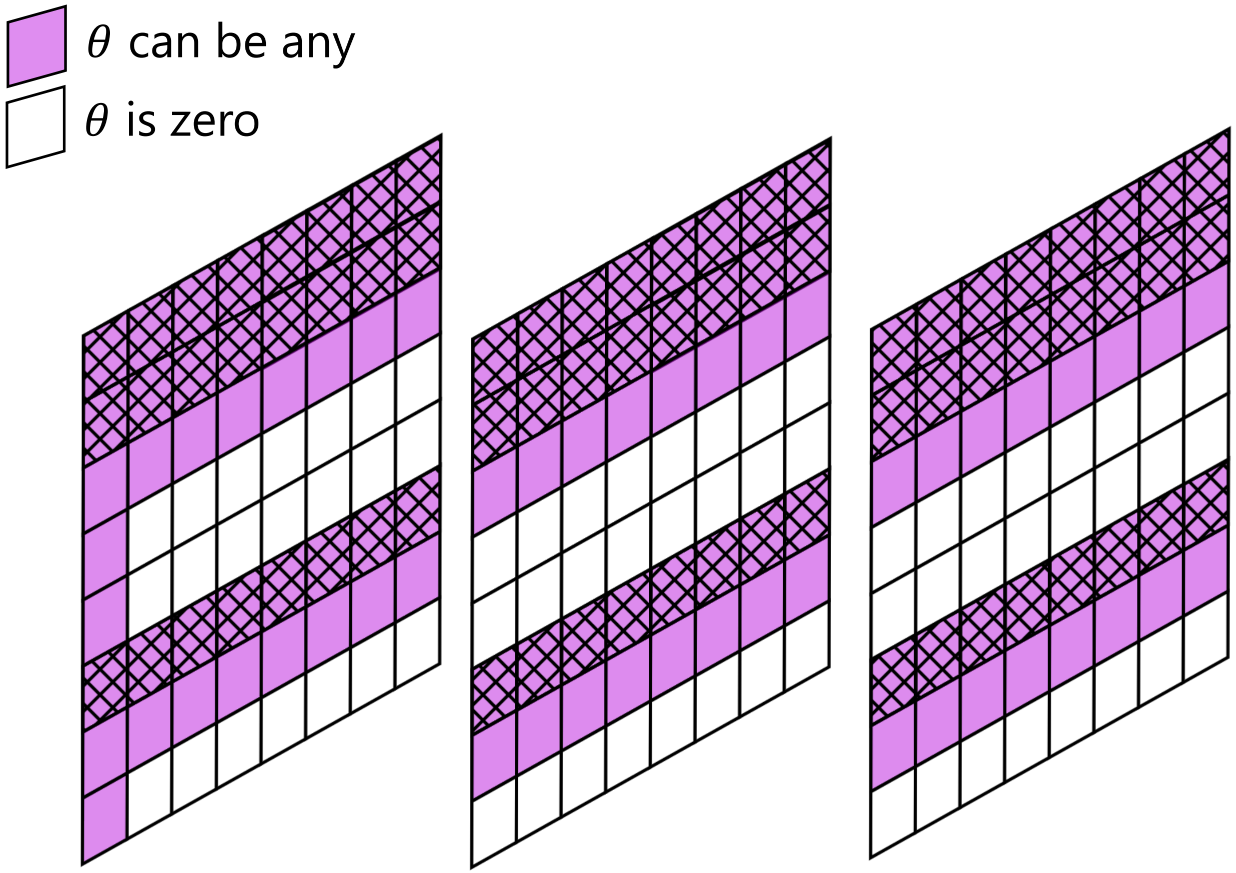
STEP1 : Choose a bingo location.

Example: Reduce the rank of (8,8,3) tensor to (5,8,3) or less



STEP1 : Choose a bingo location.

Example: Reduce the rank of (8,8,3) tensor to (5,8,3) or less

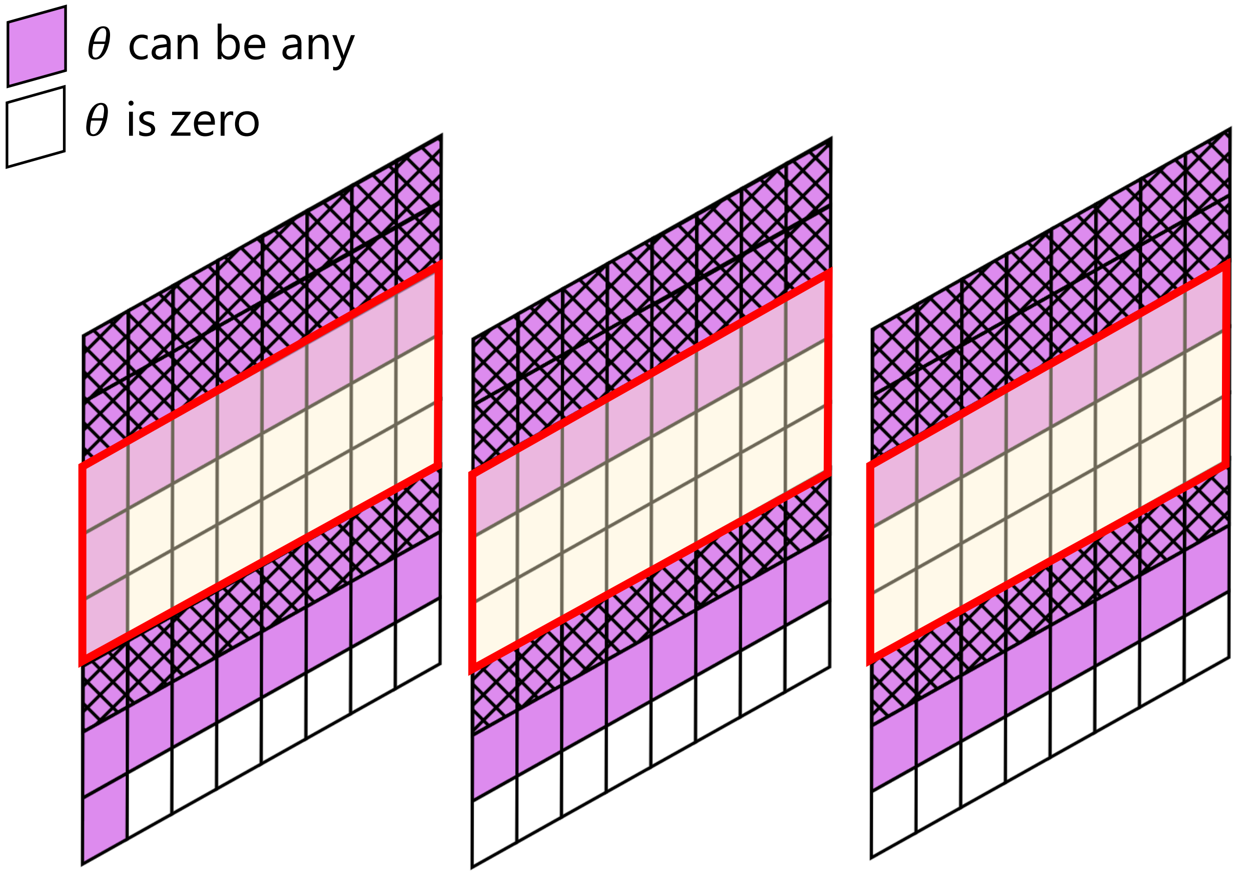


STEP1 : Choose a bingo location.

STEP2 : Replace the bingo part with the best rank-1 tensor.

The shaded areas do not change their values in the projection.

Example: Reduce the rank of (8,8,3) tensor to (5,8,3) or less



STEP1 : Choose a bingo location.

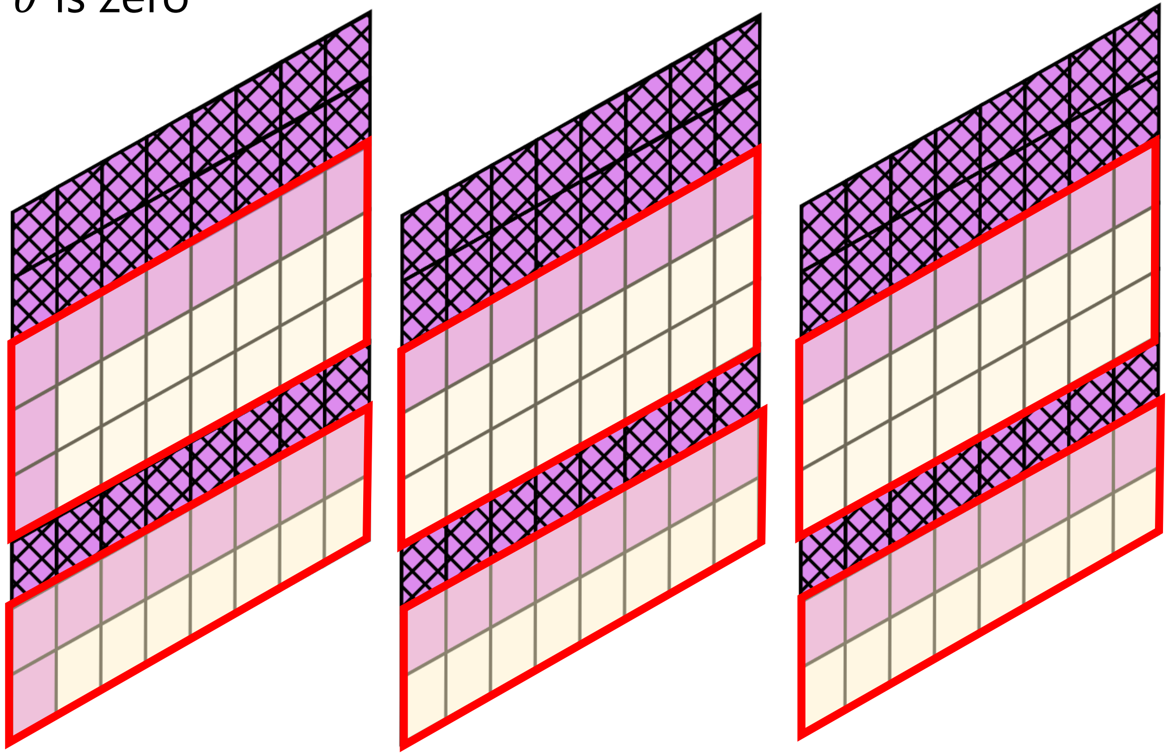
STEP2 : Replace the bingo part with the best rank-1 tensor.

Replace the partial tensor in the red box using the best rank-1 approximation formula

Example: Reduce the rank of (8,8,3) tensor to (5,8,3) or less

θ can be any

θ is zero

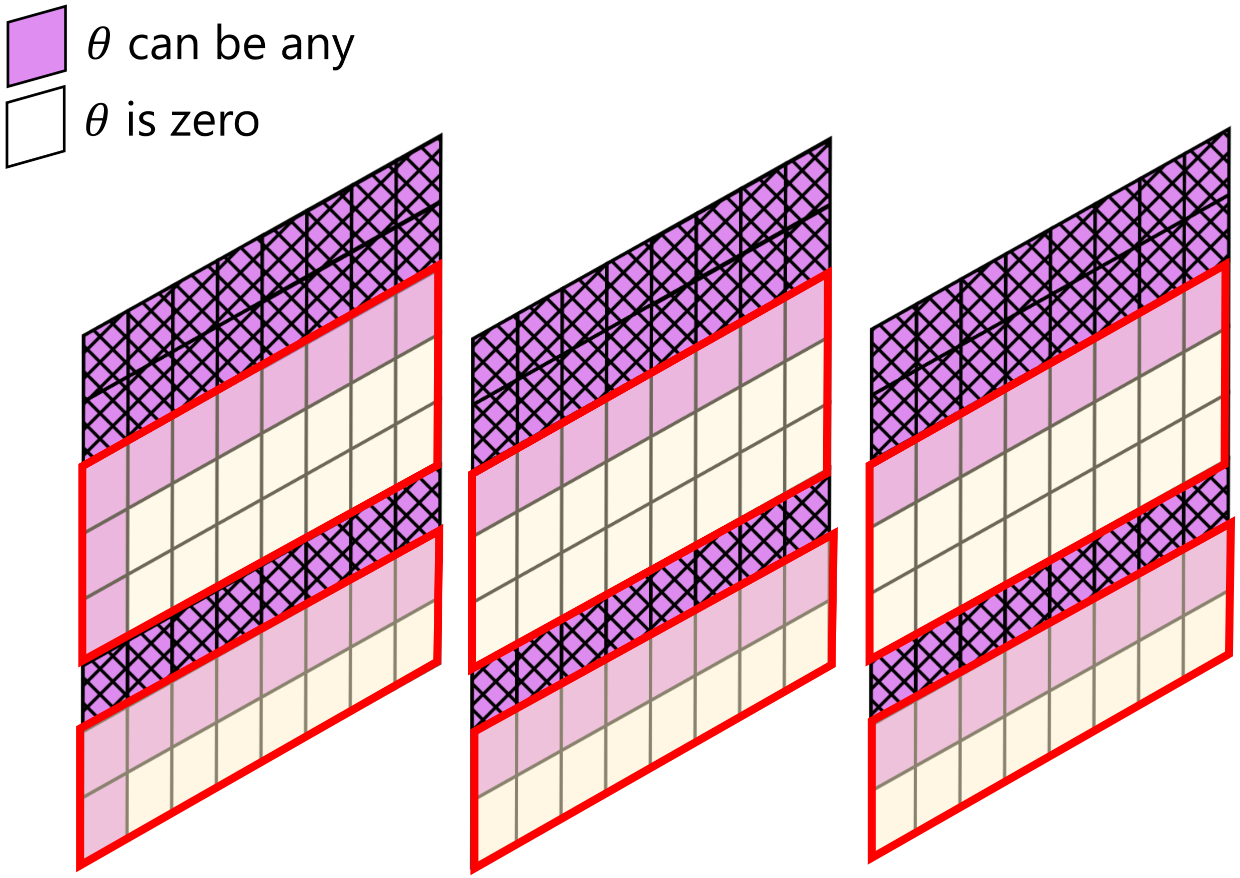


STEP1 : Choose a bingo location.

STEP2 : Replace the bingo part with the best rank-1 tensor.

Replace the partial tensor in the red box using the best rank-1 approximation formula

Example: Reduce the rank of (8,8,3) tensor to (5,8,3) or less



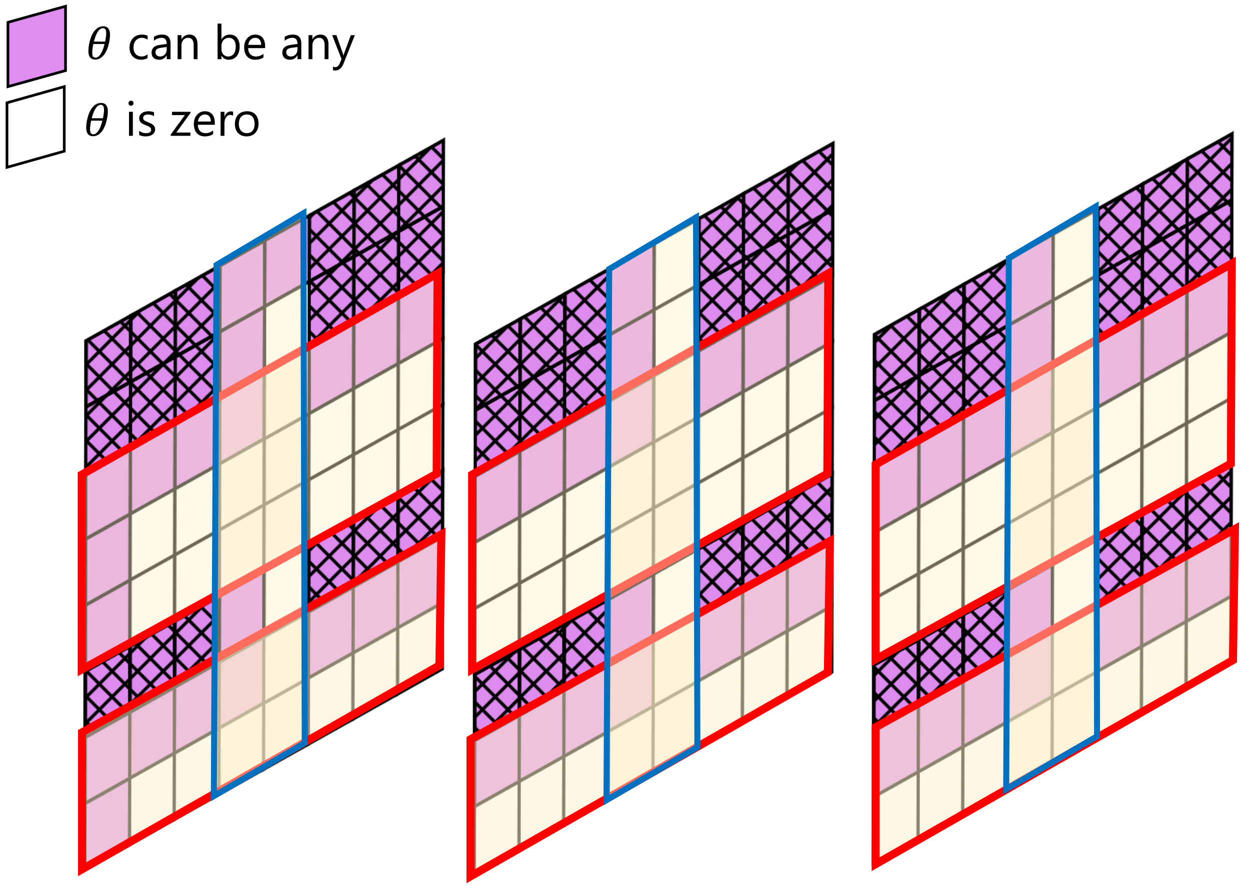
STEP1 : Choose a bingo location.

STEP2 : Replace the bingo part with the best rank-1 tensor.

Replace the partial tensor in the red box using the best rank-1 approximation formula

The best tensor is obtained in the specified bingo space. 😄
There is no guarantee that it is the best rank (5,8,3) approximation. 😢

Example: Reduce the rank of (8,8,3) tensor to (5,7,3) or less

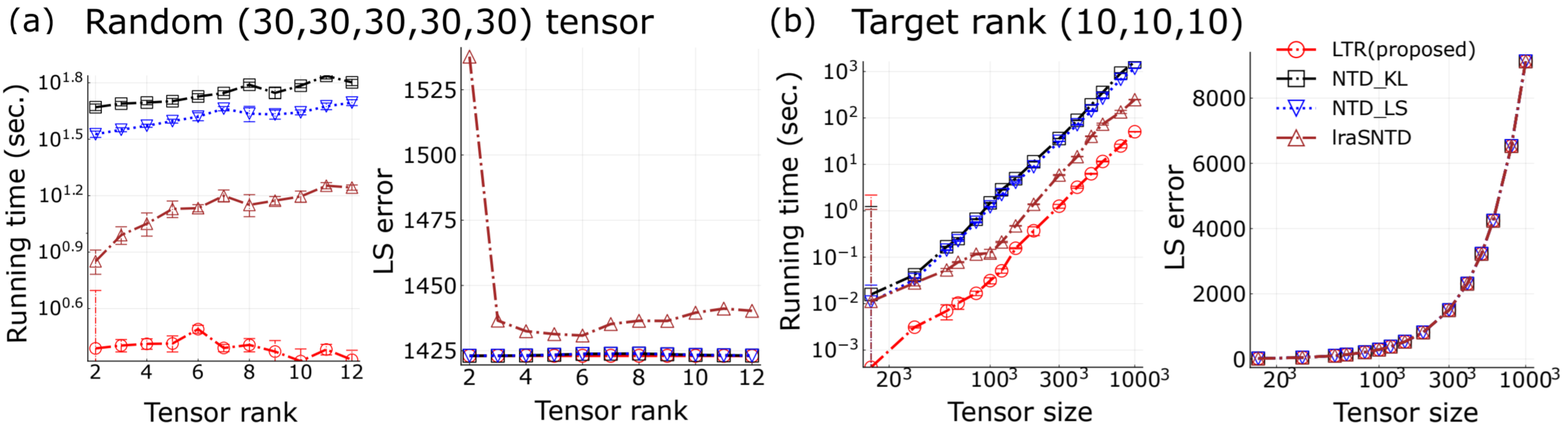


STEP1 : Choose a bingo location.

STEP2 : Replace the bingo part with the best rank-1 tensor.

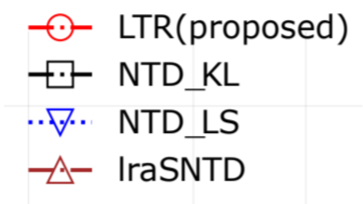
The shaded areas do not change their values in the projection.

Experimental results (synthetic data)

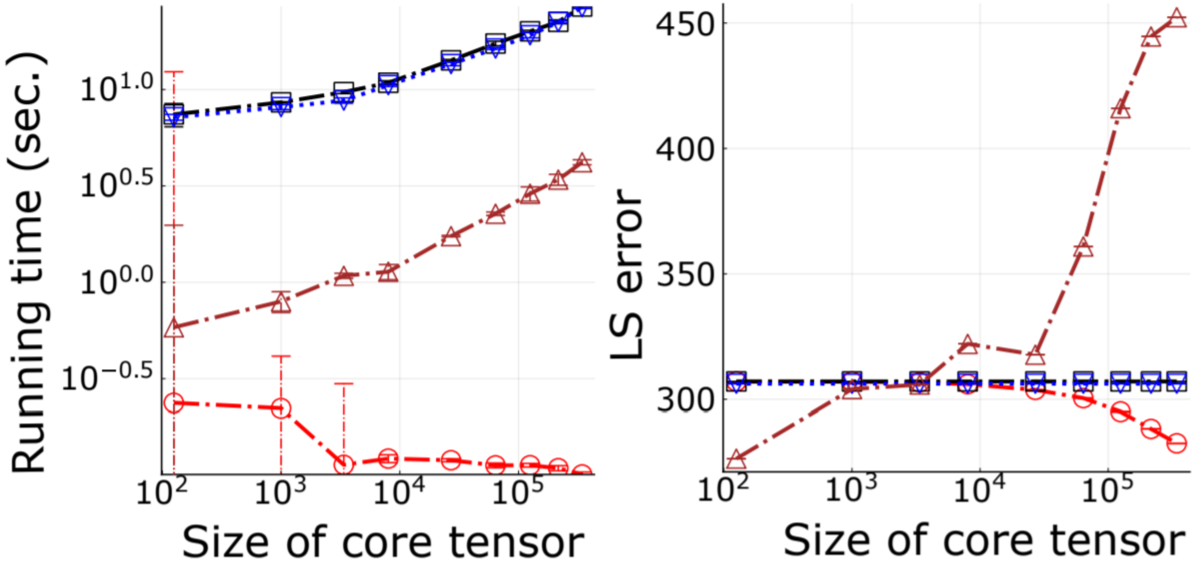


LTR is faster with the competitive approximation performance.

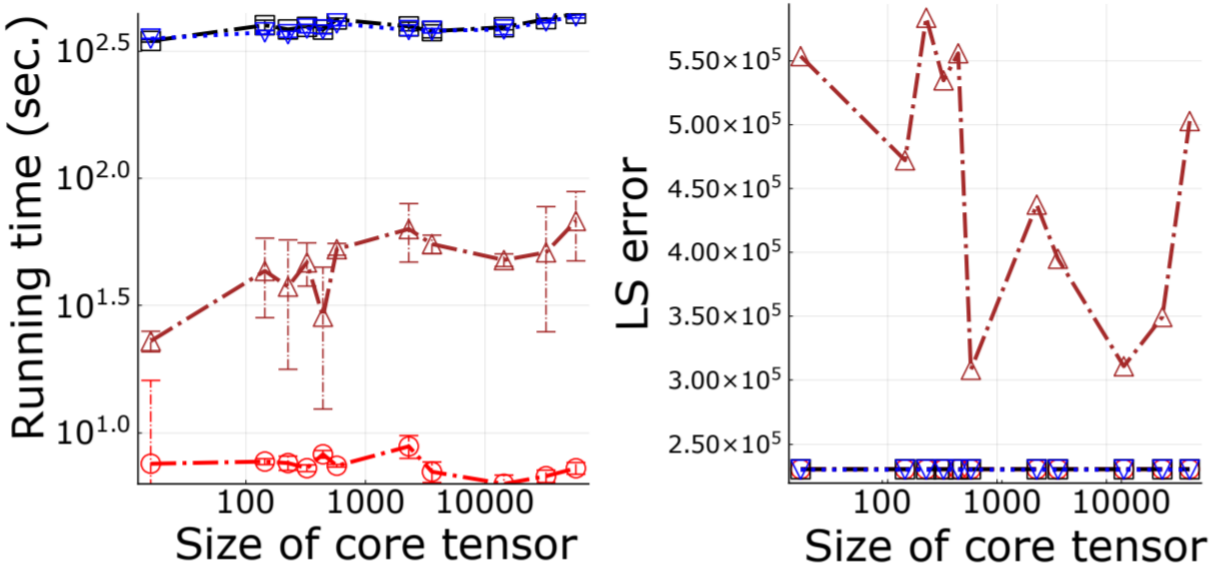
Experimental results (real data)



(c) AttFace (92, 112, 400)



(d) 4DLFD (9, 9, 512, 512, 3)

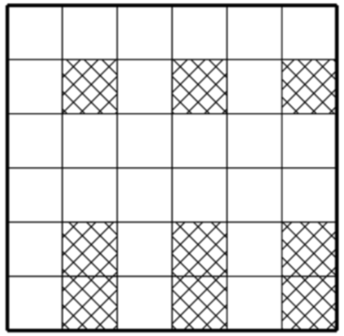


LTR is faster with the competitive approximation performance.

Contents

Strategy for rank-1 NMF with missing values

☐ Collect missing values in a corner of matrix to solve as coupled NMF



 Missing value

$$D(\mathbf{\Phi} \circ \mathbf{X}, \mathbf{\Phi} \circ (\boldsymbol{w} \otimes \boldsymbol{h}))$$

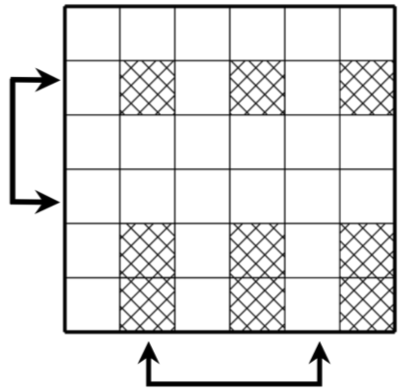
Element-wise product (↑ $\circ$)

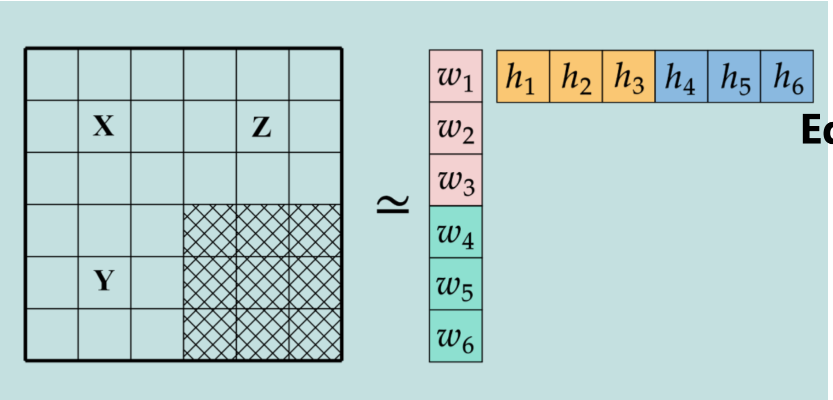
$$\mathbf{\Phi}_{ij} = \begin{cases} 0 & \text{If } \mathbf{X}_{ij} \text{ is missing} \\ 1 & \text{otherwise} \end{cases}$$

Strategy for rank-1 NMF with missing values

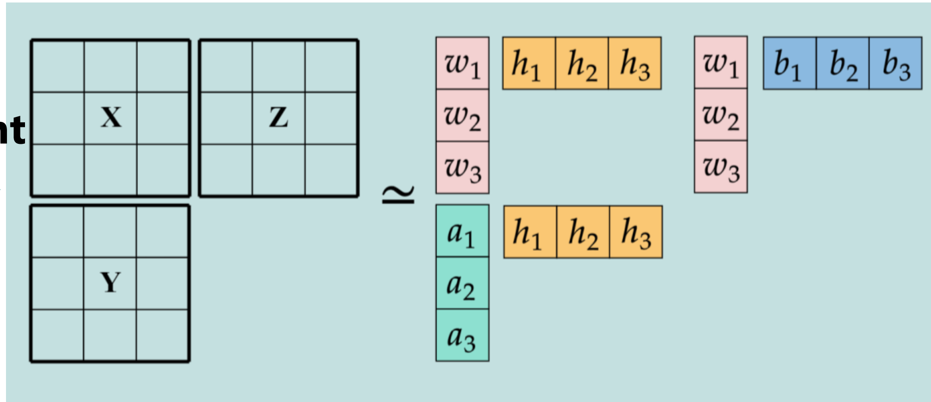
☐ Collect missing values in a corner of matrix to solve as coupled NMF

NMMF (Takeuchi et al., 2013)





Equivalent



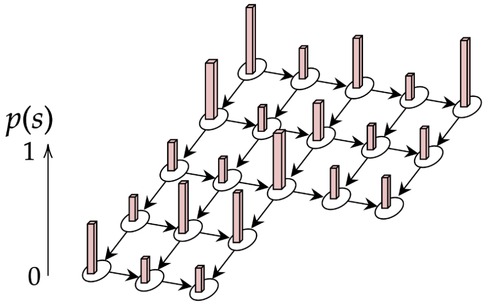
Missing value

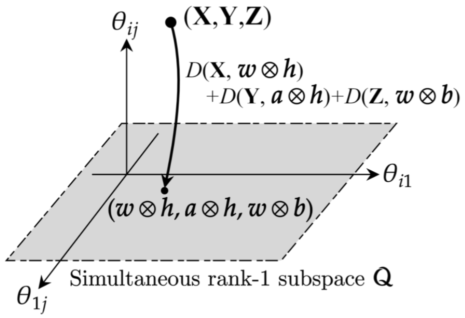
$$D(\mathbf{X}, w \otimes h) + D(\mathbf{Y}, a \otimes h) + D(\mathbf{Z}, w \otimes b)$$

$$D(\boldsymbol{\Phi} \circ \mathbf{X}, \boldsymbol{\Phi} \circ (\boldsymbol{w} \otimes \boldsymbol{h}))$$

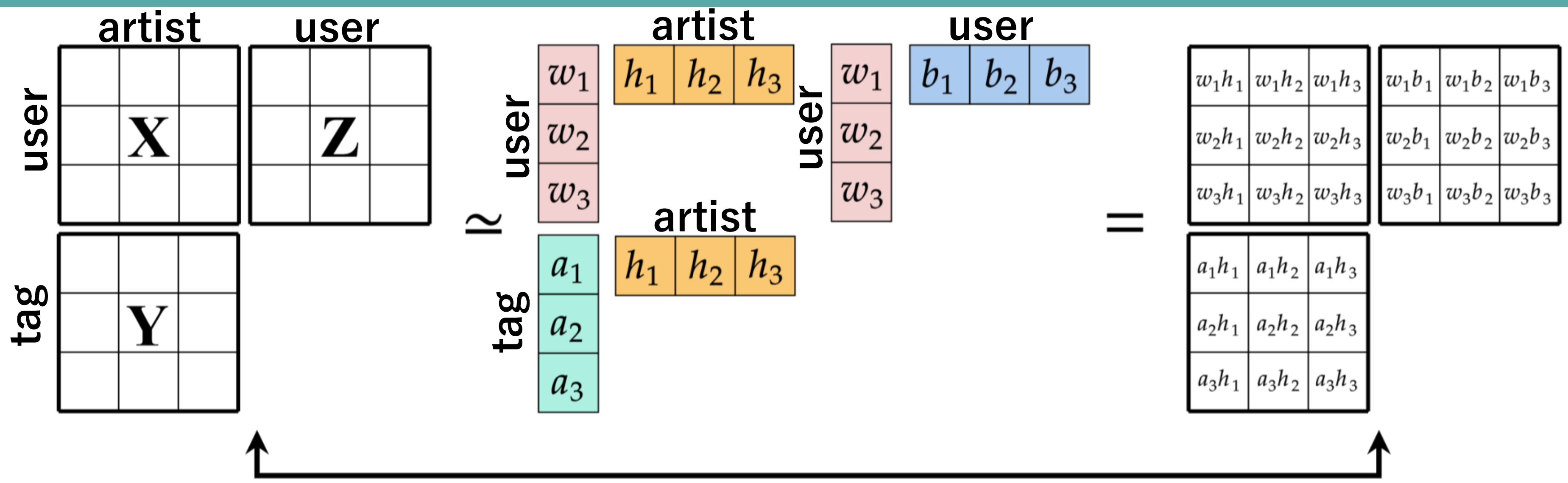
Element-wise product

$$\boldsymbol{\Phi}_{ij} = \begin{cases} 0 & \text{If } \mathbf{X}_{ij} \text{ is missing} \\ 1 & \text{otherwise} \end{cases}$$



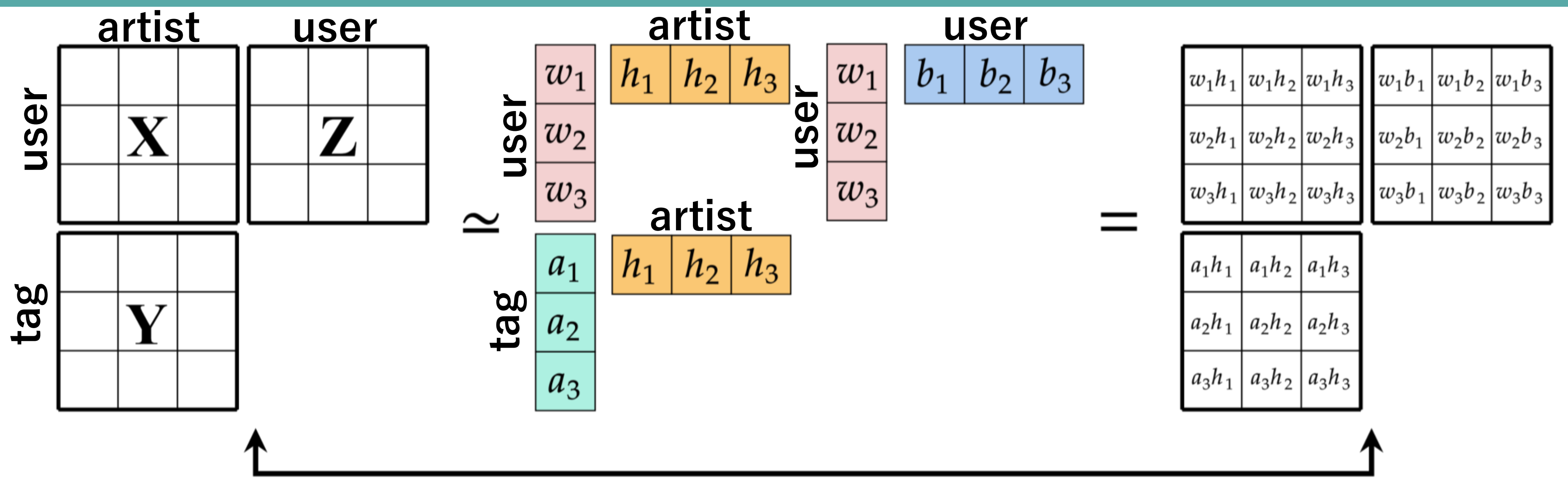


NMMF, Nonnegative multiple matrix factorization (Takeuchi et al., 2013)



$$D(\mathrm{X}, w \otimes h) + \alpha D(\mathrm{Y}, a \otimes h) + \beta D(\mathrm{Z}, w \otimes b)$$

The best rank-1 approximation of NMMF



$$D(\mathrm{X}, w \otimes h) + \alpha D(\mathrm{Y}, a \otimes h) + \beta D(\mathrm{Z},\ w \otimes b)$$

The best rank-1 approximation of NMMF

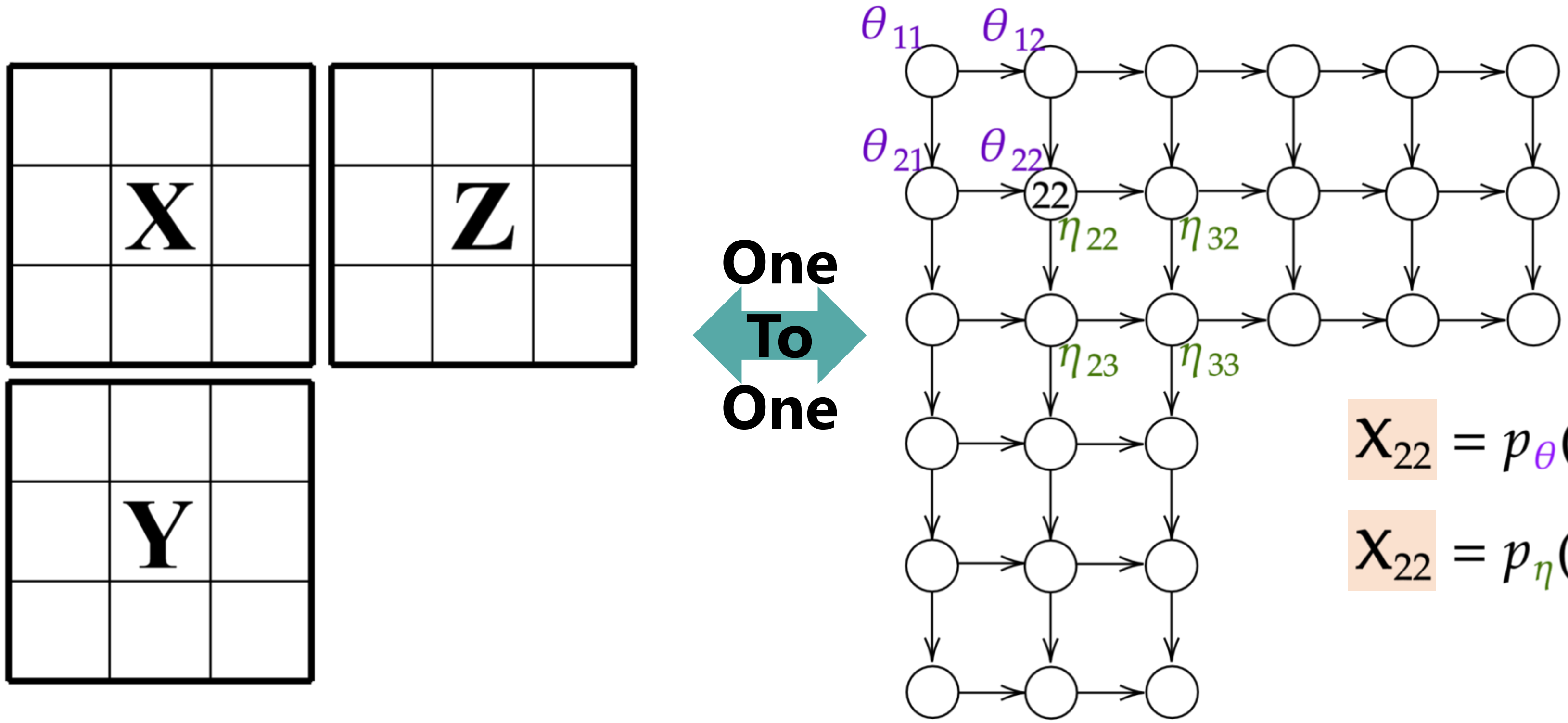
For given $\mathrm{X} \in \mathbb{R}_{>0}^{I \times J}$, $\mathrm{Y} \in \mathbb{R}_{>0}^{N \times J}$, and $\mathrm{Z} \in \mathbb{R}_{>0}^{I \times M}$ the best rank-1 NMMF is given as

$$w_i = \frac{\sqrt{S(\mathrm{X})}}{S(\mathrm{X}) + \beta S(\mathrm{Z})} \left\{ \sum_{j=1}^{J} \mathrm{X}_{ij} + \beta \sum_{m=1}^{M} \mathrm{Z}_{im} \right\} \qquad a_n = \frac{\sum_{j=1}^{J} \mathrm{Y}_{nj}}{\sqrt{S(\mathrm{X})}}$$

$$h_j = \frac{\sqrt{S(\mathrm{X})}}{S(\mathrm{X}) + \alpha S(\mathrm{Y})} \left\{ \sum_{i=1}^{I} \mathrm{X}_{ij} + \alpha \sum_{n=1}^{N} \mathrm{Y}_{nj} \right\} \qquad b_m = \frac{\sum_{i=1}^{I} \mathrm{Z}_{im}}{\sqrt{S(\mathrm{X})}}$$

$S(\mathrm{X})$ is sum of all elements of X.

Modeling of NMMF



$$X_{22} = p_\theta(2,2) = \exp(\theta_{11} + \theta_{21} + \theta_{12} + \theta_{22})$$

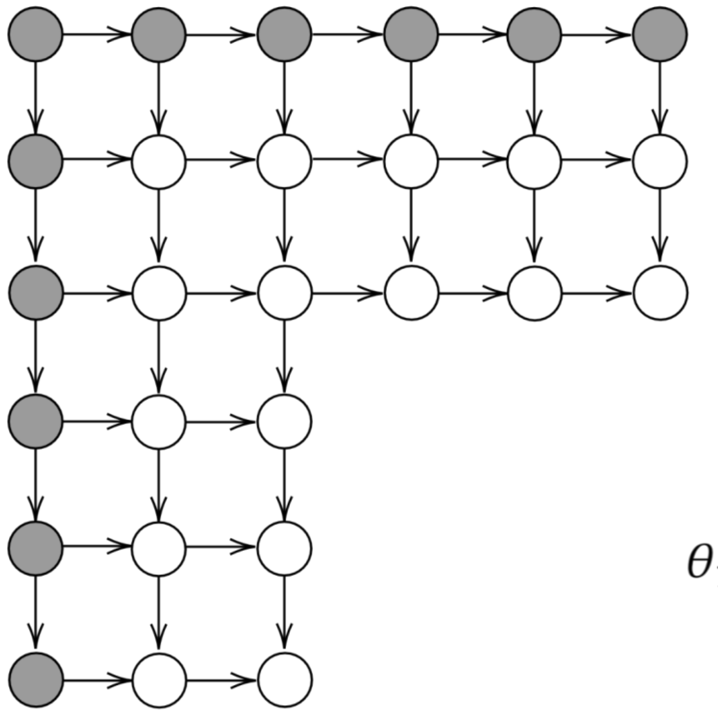
$$X_{22} = p_\eta(2,2) = \eta_{22} - \eta_{23} - \eta_{32} + \eta_{33}$$

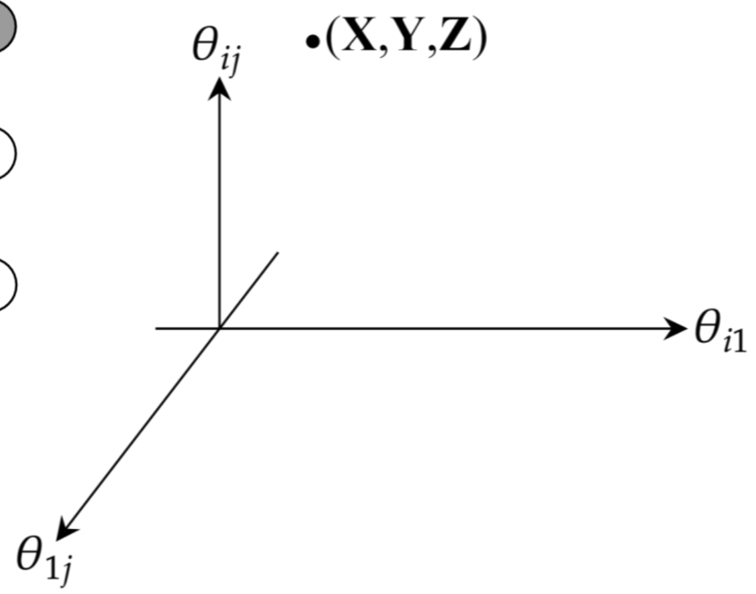
$$p_\theta(k,l) = \exp\left(\sum_{(s,t)\le(k,l)} \theta_{st}\right), \quad \eta_{kl} = \sum_{(k,l)\le(s,t)} p(s,t).$$

One-body and many-body parameters

$(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z})$ is *simultaneously rank-1 decomposable*. ⇔ It can be written as $(\boldsymbol{w} \otimes \boldsymbol{h}, \boldsymbol{a} \otimes \boldsymbol{h}, \boldsymbol{w} \otimes \boldsymbol{b})$.

● **One-body** parameter ○ **Two-body** parameter





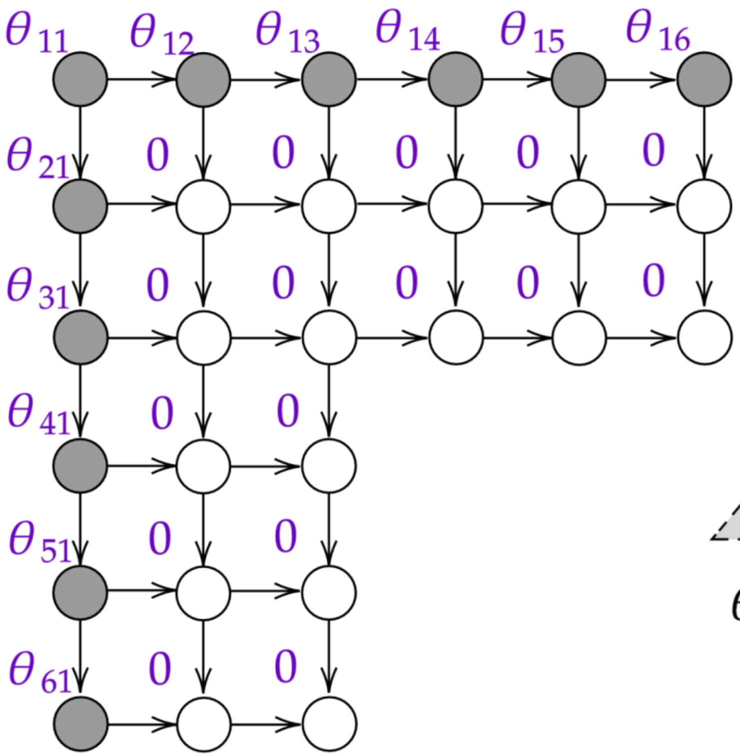
Information geometry of rank-1 NMMF

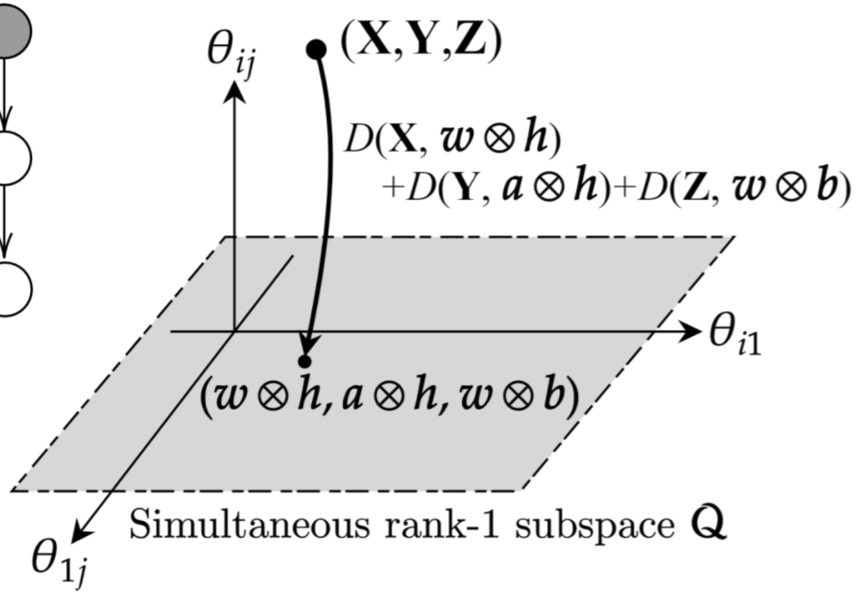
$(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z})$ is *simultaneously rank-1 decomposable*. ⇔ It can be written as $(\boldsymbol{w} \otimes \boldsymbol{h}, \boldsymbol{a} \otimes \boldsymbol{h}, \boldsymbol{w} \otimes \boldsymbol{b})$.

● **One-body** parameter ○ **Two-body** parameter

Simultaneous Rank-1 $\boldsymbol{\theta}$-condition

Its all two-body θ-parameters are 0.





Information geometry of rank-1 NMMF

$(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z})$ is *simultaneously rank-1 decomposable*. ⇔ It can be written as $(\boldsymbol{w} \otimes \boldsymbol{h}, \boldsymbol{a} \otimes \boldsymbol{h}, \boldsymbol{w} \otimes \boldsymbol{b})$.

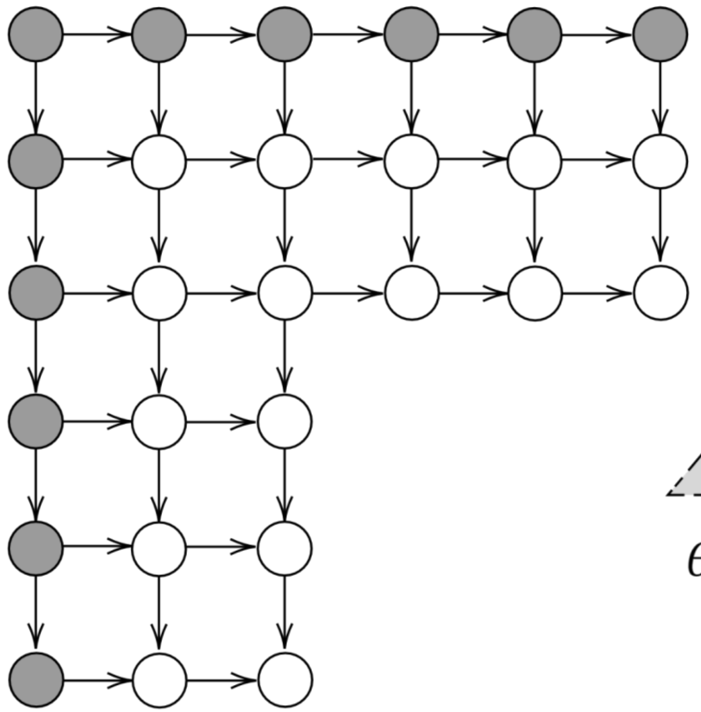
One-body parameter **Two-body** parameter

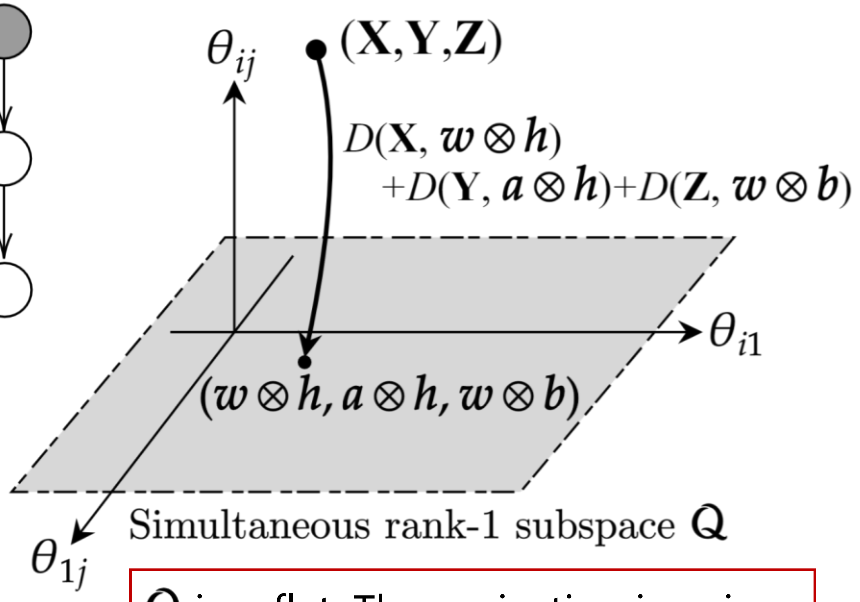
Simultaneous Rank-1 $\boldsymbol{\theta}$-condition

Its all two-body θ-parameters are 0.

Simultaneous Rank-1 $\boldsymbol{\eta}$-condition

$$\eta_{ij} = \eta_{i1}\eta_{1j}$$





$\mathbf{Q}$ is *e*-flat. The projection is unique.

Find the global optimal solution of rank-1 NMMF

$(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z})$ is *simultaneously rank-1 decomposable*. ⇔ It can be written as $(\boldsymbol{w} \otimes \boldsymbol{h}, \boldsymbol{a} \otimes \boldsymbol{h}, \boldsymbol{w} \otimes \boldsymbol{b})$.

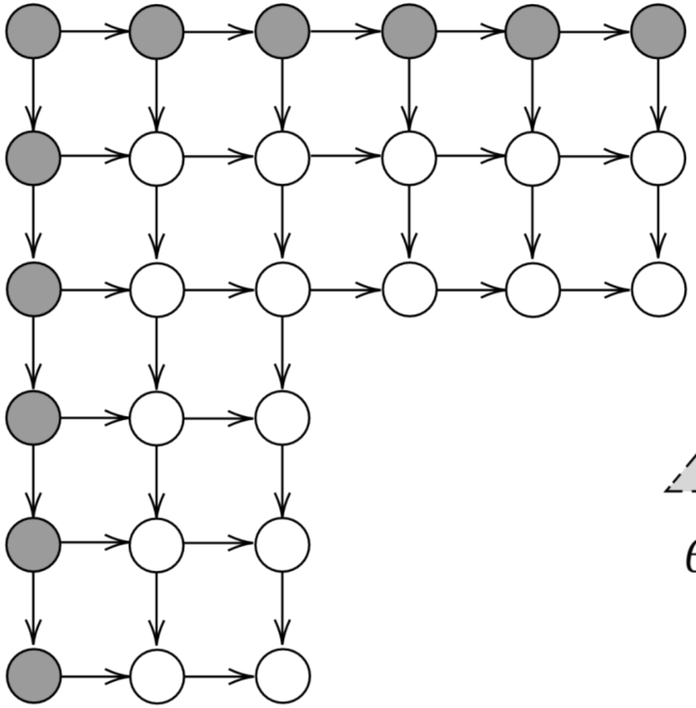
● **One-body** parameter ○ **Two-body** parameter

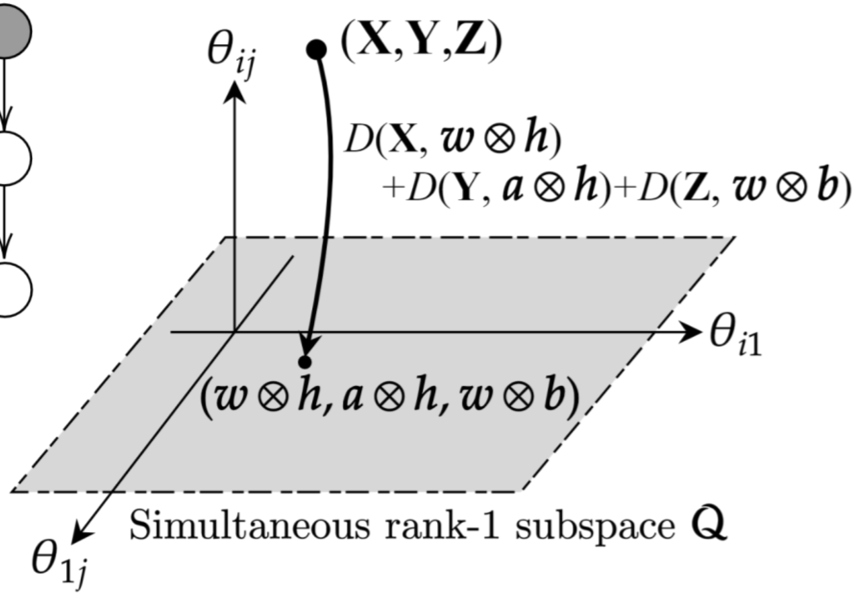
Simultaneous Rank-1 $\boldsymbol{\theta}$-condition

Its all two-body θ-parameters are 0.

Simultaneous Rank-1 $\boldsymbol{\eta}$-condition

$$\eta_{ij} = \eta_{i1}\eta_{1j}$$





The *m*-projection does not change one-body $\boldsymbol{\eta}$-parameter

Shun-ichi Amari, Information Geometry and Its Applications, 2008, Theorem 11.6

Find the global optimal solution of rank-1 NMMF

$(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z})$ is *simultaneously rank-1 decomposable*. ⇔ It can be written as $(\boldsymbol{w} \otimes \boldsymbol{h}, \boldsymbol{a} \otimes \boldsymbol{h}, \boldsymbol{w} \otimes \boldsymbol{b})$.

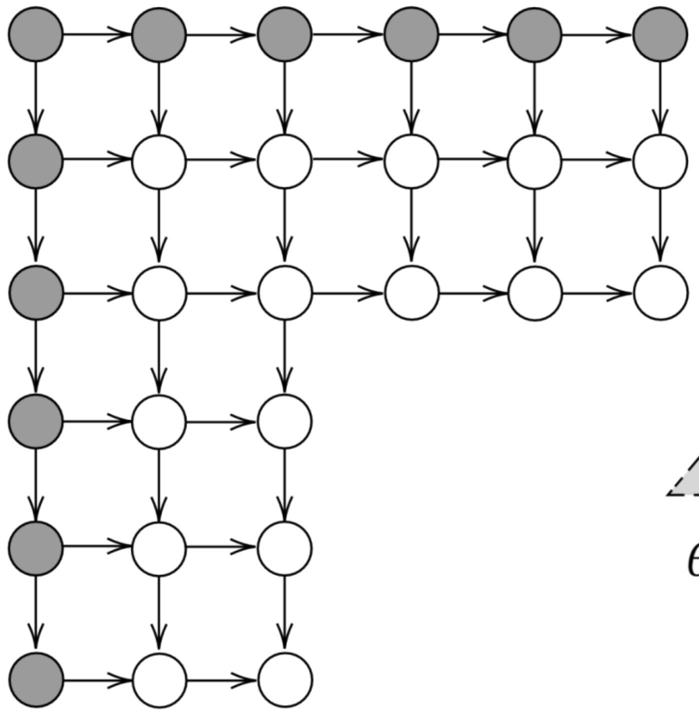
● **One-body** parameter ○ **Two-body** parameter

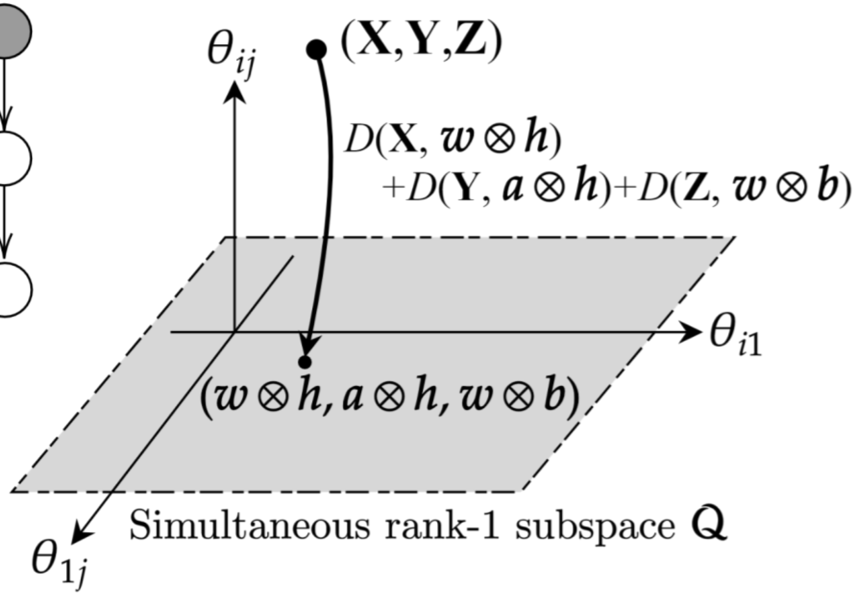
Simultaneous Rank-1 $\boldsymbol{\theta}$-condition

Its all two-body θ-parameters are 0.

Simultaneous Rank-1 $\boldsymbol{\eta}$-condition

$$\eta_{ij} = \eta_{i1}\eta_{1j}$$





The *m*-projection does not change one-body $\boldsymbol{\eta}$-parameter

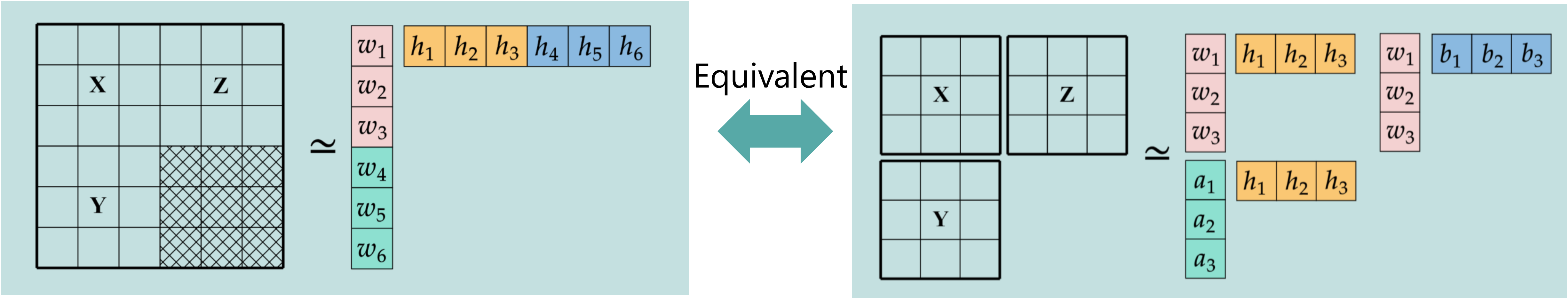
Shun-ichi Amari, Information Geometry and Its Applications, 2008, Theorem 11.6

➡ All $\boldsymbol{\eta}$-parameters after the projection are identified.

19:20

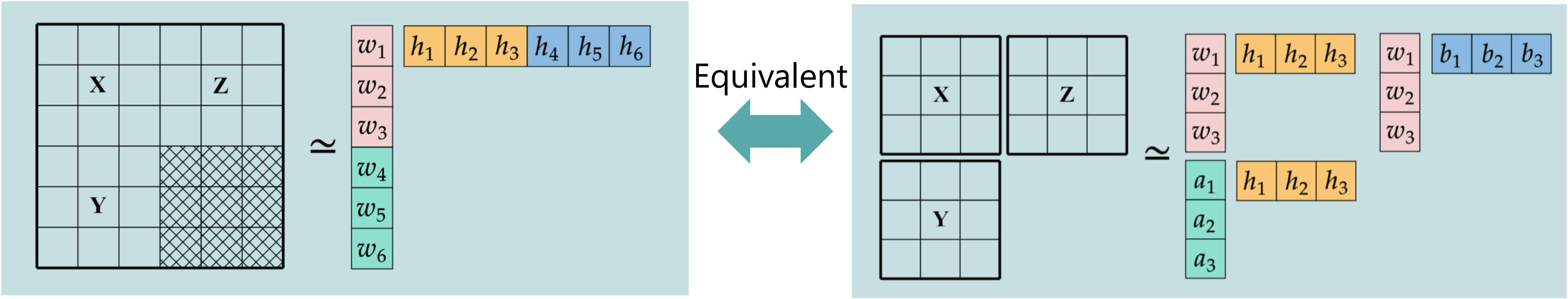
Rank-1 NMF with missing values

- NMMF can be viewed as a special case of NMF with missing values.

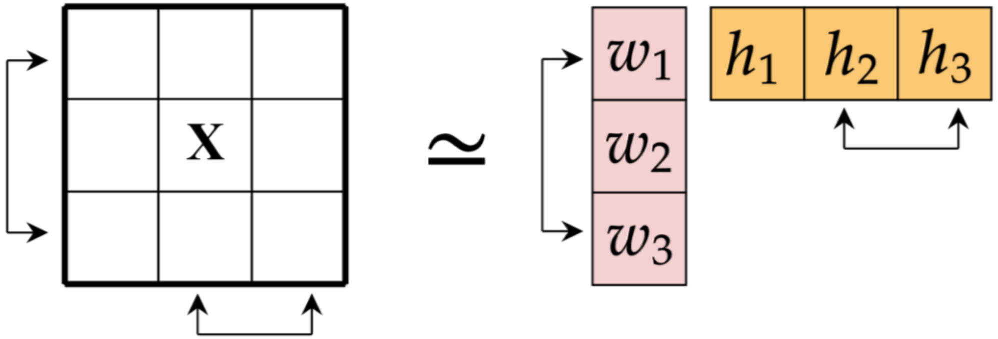


Rank-1 NMF with missing values

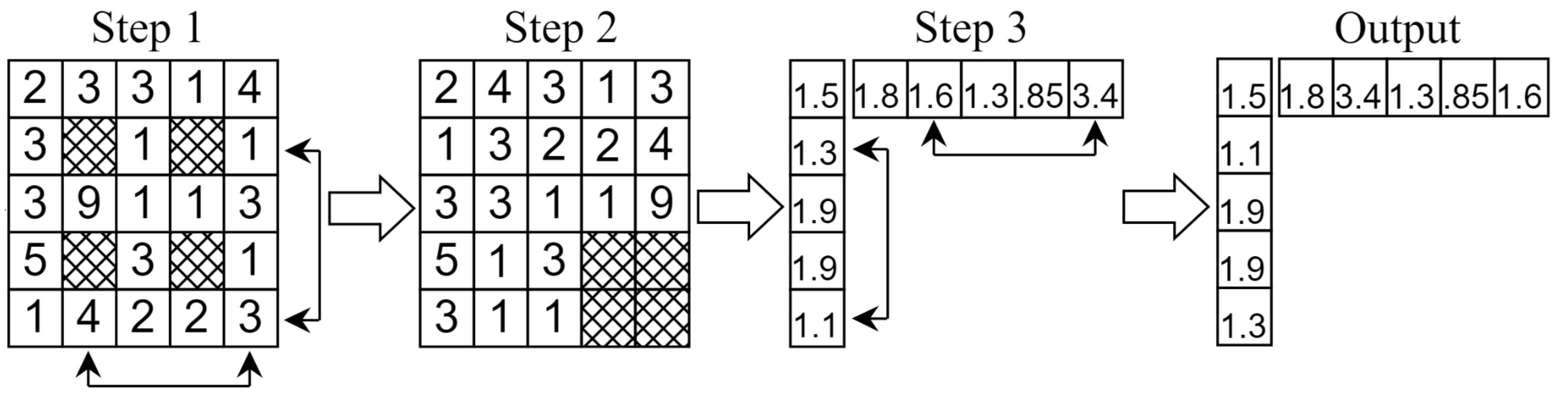
- ☐ NMMF can be viewed as a special case of NMF with missing values.



- ☐ NMF is homogeneous for row and column permutations



A1GM: Algorithm



Step 1 : Gather missing values in the bottom right.

Step 2 : Use the formula of the best rank-1 NMMF.

Step 3 : Repermutate

Find exact solution 🤔❓

Examples that permutations cannot collect missing values into corners

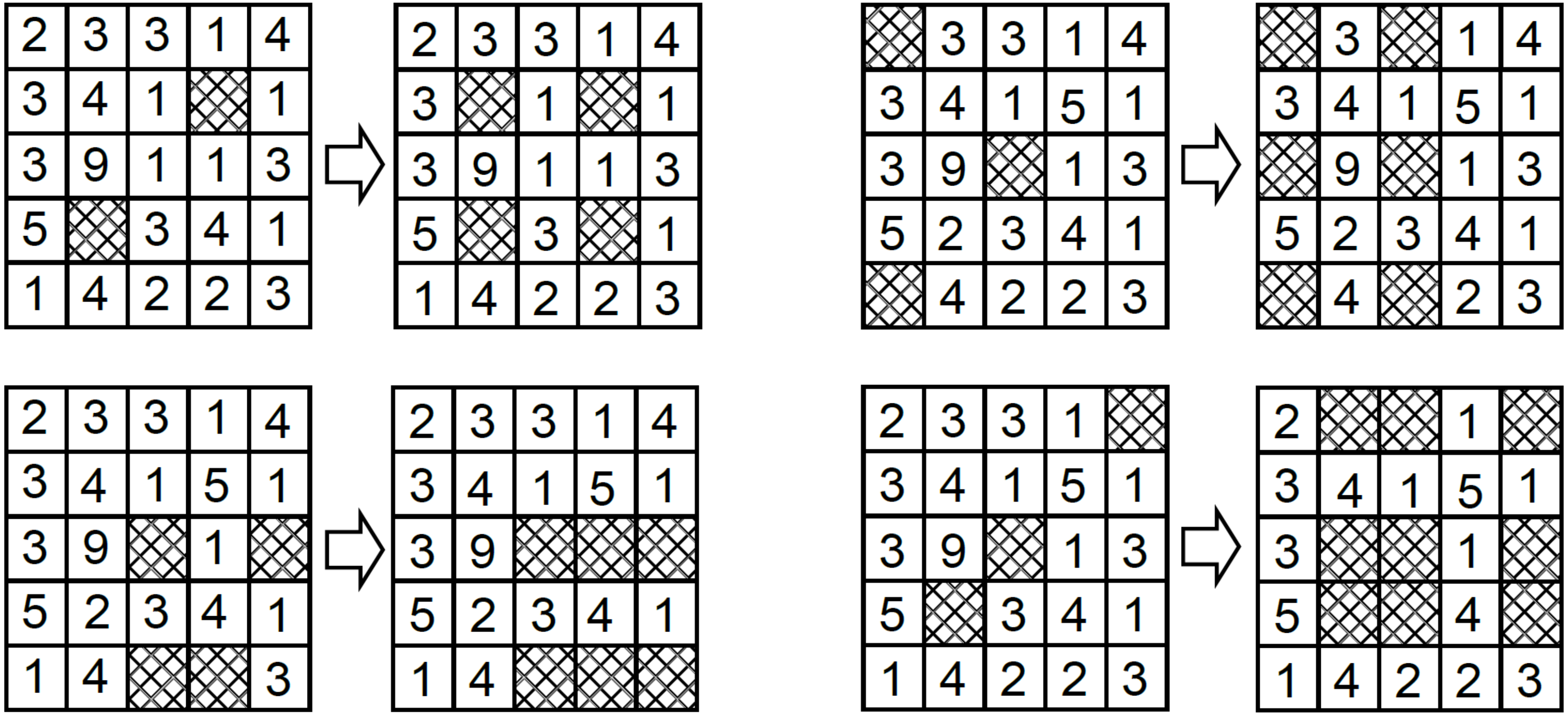
2	3	3	1	4
3	4	1		1
3	9	1	1	3
5		3	4	1
1	4	2	2	3

	3	3	1	4
3	4	1	5	1
3	9		1	3
5	2	3	4	1
	4	2	2	3

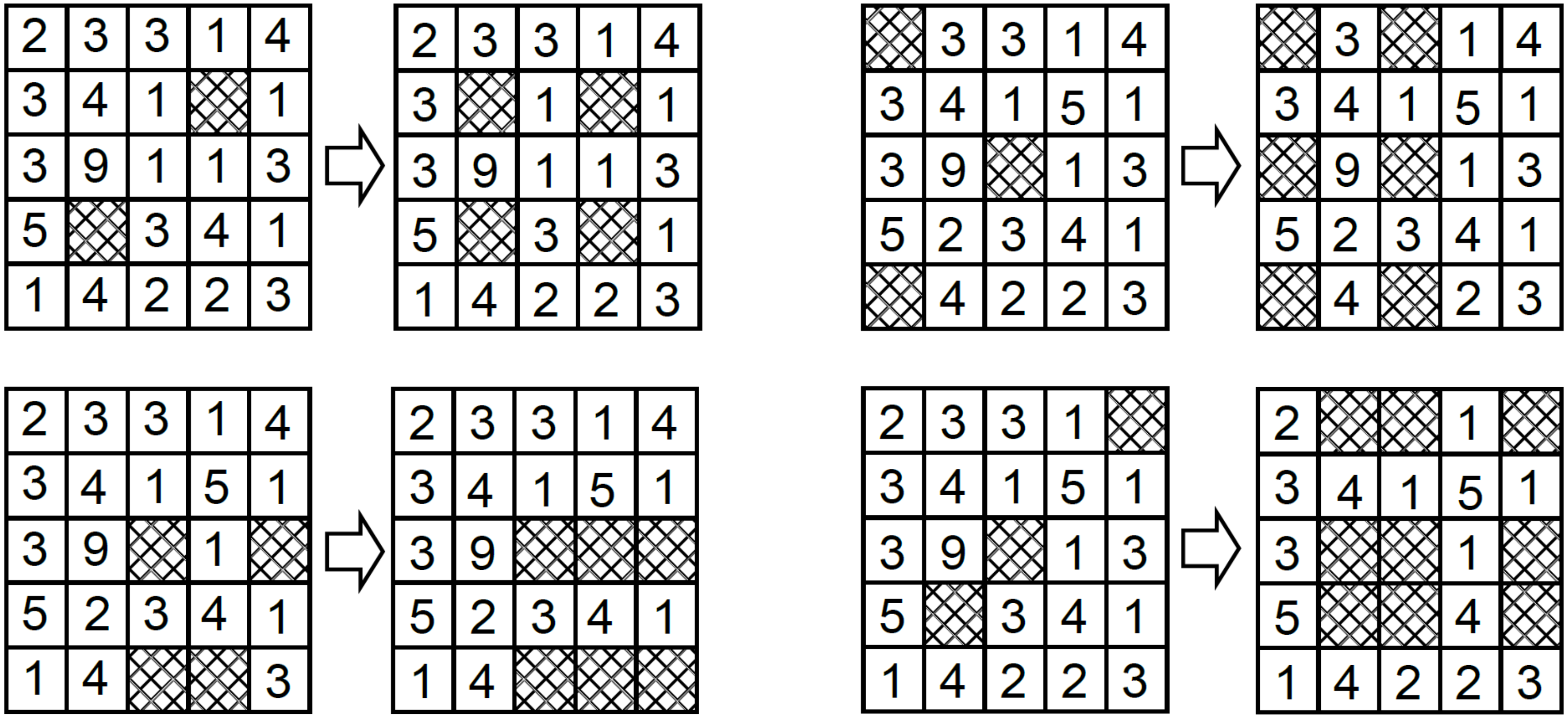
2	3	3	1	4
3	4	1	5	1
3	9		1	
5	2	3	4	1
1	4			3

2	3	3	1	
3	4	1	5	1
3	9		1	3
5		3	4	1
1	4	2	2	3

Add missing values to solve the problem as NMMF

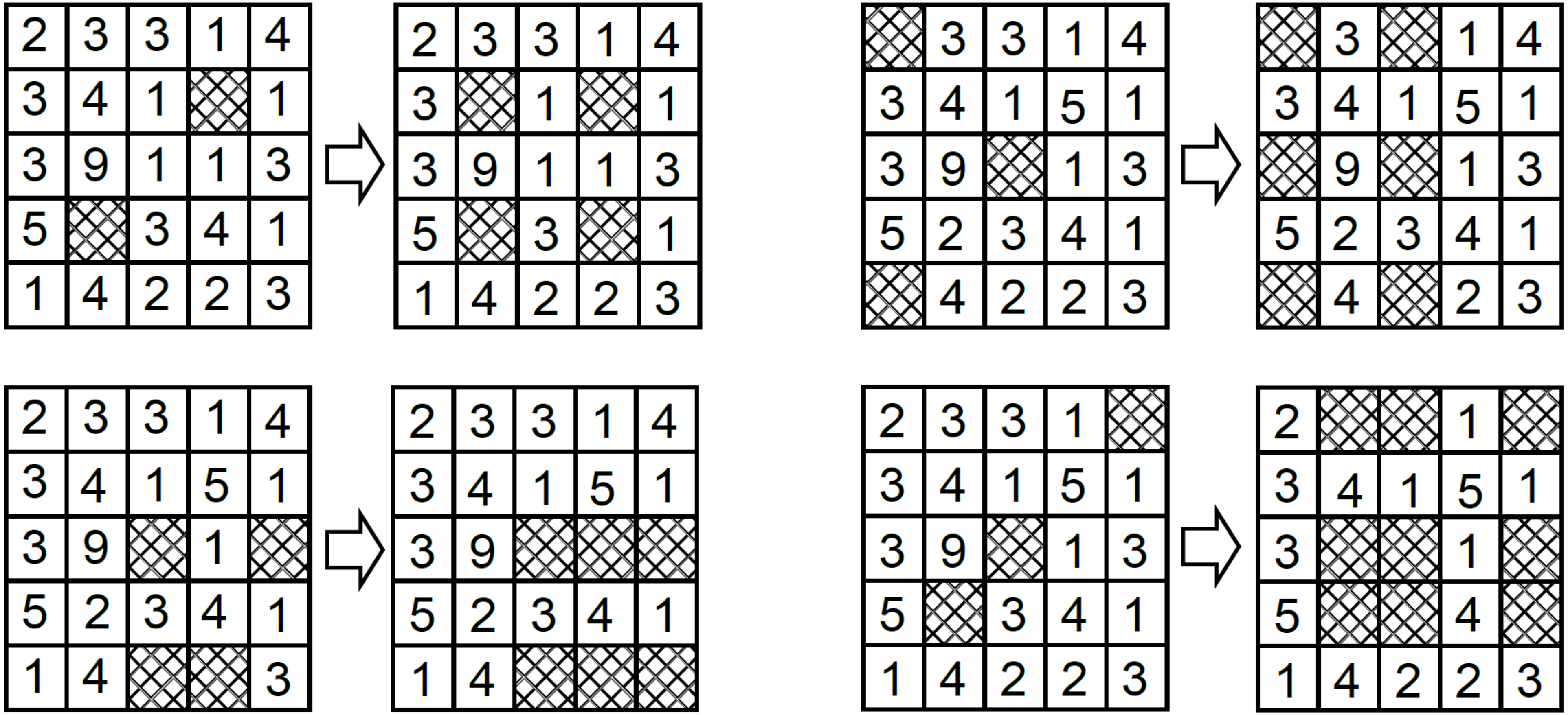


Add missing values to solve the problem as NMMF



Reconstruction error worsens 😟

Add missing values to solve the problem as NMMF

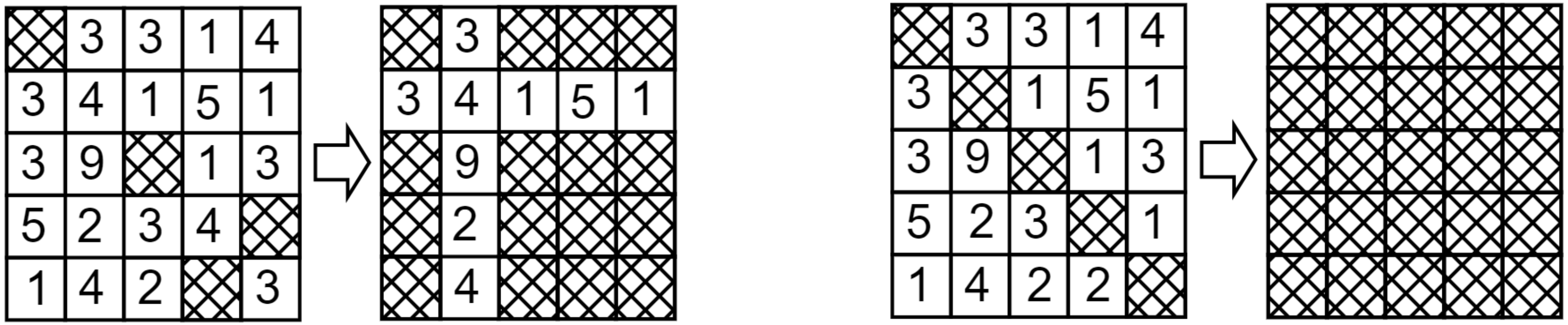


Reconstruction error worsens 😥

Gain in efficiency 😀

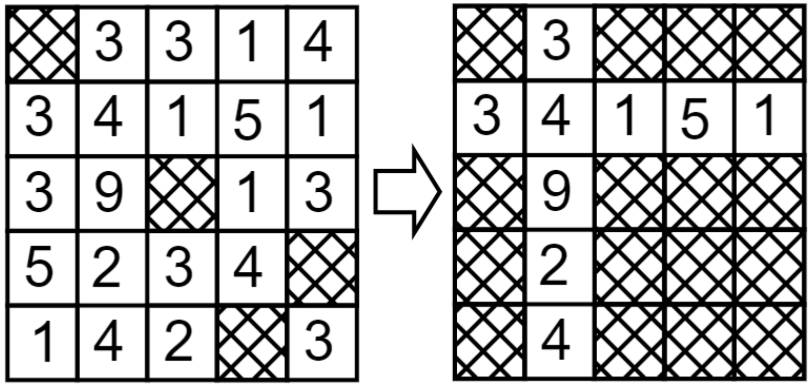
Data that A1GM is good at and not good at

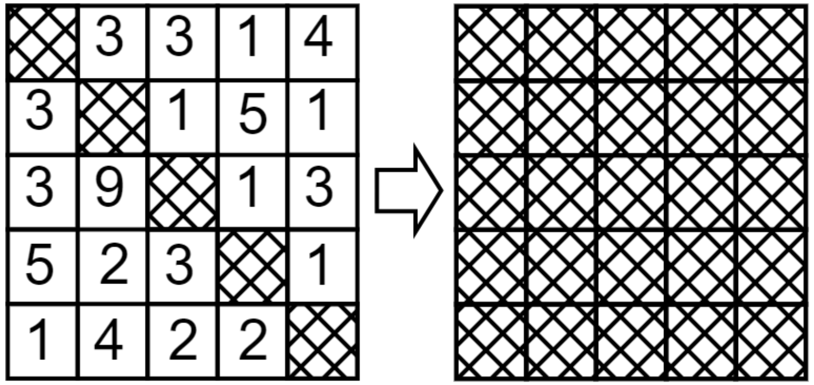
Missing values are evenly distributed in each row and column.



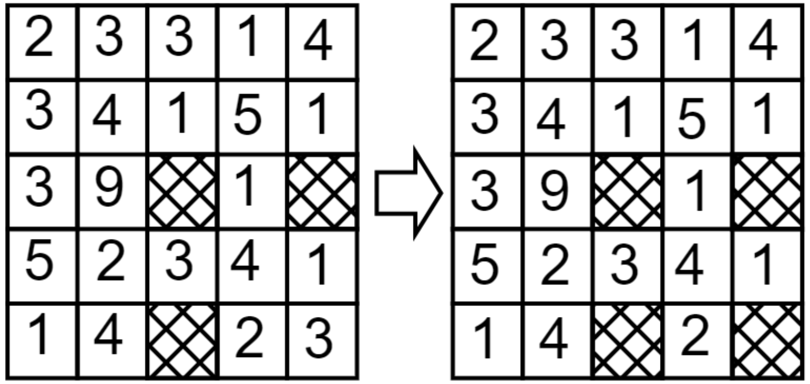
Data that A1GM is good at and not good at

Missing values are evenly distributed in each row and column.





Missing are heavily distributed in certain rows and columns.

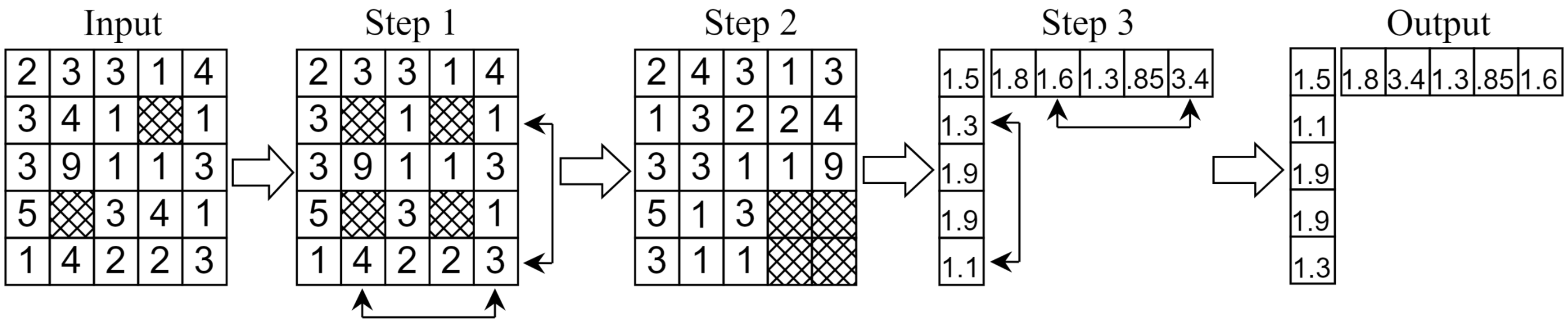


Missing values tend to be in certain columns in some real datasets.

ex) disconnected sensing device,

optional answer field in questionnaire form

A1GM: Algorithm



Step 1 : Increase the number of missing values.

Step 2 : Gather missing values in the bottom right.

Step 3 : Use the formula of rank-1 NMMF and repermutate.

Experiments on real data

☐ A1GM is compared with gradient-based KL-WNMF

- Relative runtime < 1 means A1GM is faster than KL-WNMF.
- Relative error > 1 means worse reconstruction error of A1GM than KL-WNMF.
- Increase rate is the ratio of # missing values after addition of missing values at step1.

×5 – 10 times faster!

Find the best solution

Add missing values.
Accuracy decreases.

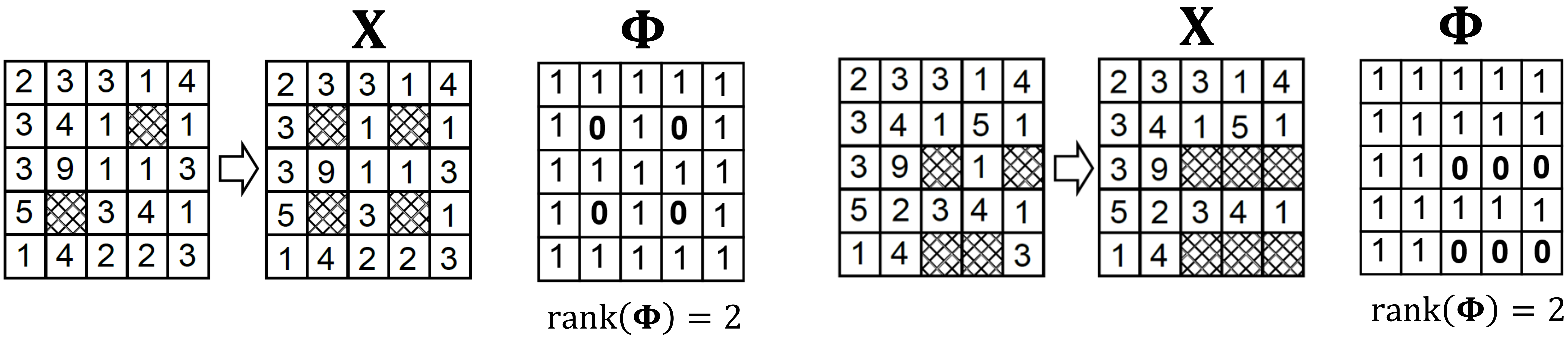
DataSet	size	# missing values	increase rate	relative error	relative runtime
IndianPop	(24,13)	1	1	1	0.19784
Autompg	(398, 8)	6	1	1	0.12957
DailySunSpot	(73718, 9)	3247	1	1	0.12845
CaliforniaHousing	(20640, 9)	207	1	1	0.11821
MTSLibrary	(1533078, 4)	1247722	1	1	0.18327
BigMartSaleForecas	(8522, 5)	1463	1	1	0.12699
BoardGameGeekData	(101375, 17)	21	1	1	0.14625
CreditCardApproval	(590, 7)	25	1.92	1.0018	0.12212
HumanResourceAnaly	(14999, 7)	519	1.96146	1.0168	0.11858
concretemiss	(1030,9)	99	2	1.0010	0.11108
heartdisease	(303, 14)	6	2	1	0.12259
lungcancer	(32, 57)	5	2	1.0001	0.13803
PerthHousePrice	(33656, 14)	16585	2.61345	1.0004	0.15382
SleepData	(62, 8)	12	2.75	1.0211	0.18208
HCVData	(615,11)	31	4.1935	1.0068	0.11246
arrhythmia	(452, 280)	408	4.70588	1.0148	0.11387
Bostonhousing	(506, 14)	120	5.6	1.003	0.1097
LifeExpectancyData	(2938, 19)	2563	7.04097	5.7983	0.095773
HCCSurvivalDataSet	(165, 50)	826	8.3632	3.2898	0.07113
wiki4HE	(913, 53)	1995	18.10175	1.2363	0.066256

Contents

Theoretical Remarks 1 : Extended NMMF.

- ☐ The rank of weight matrix is 2 after adding missing values.

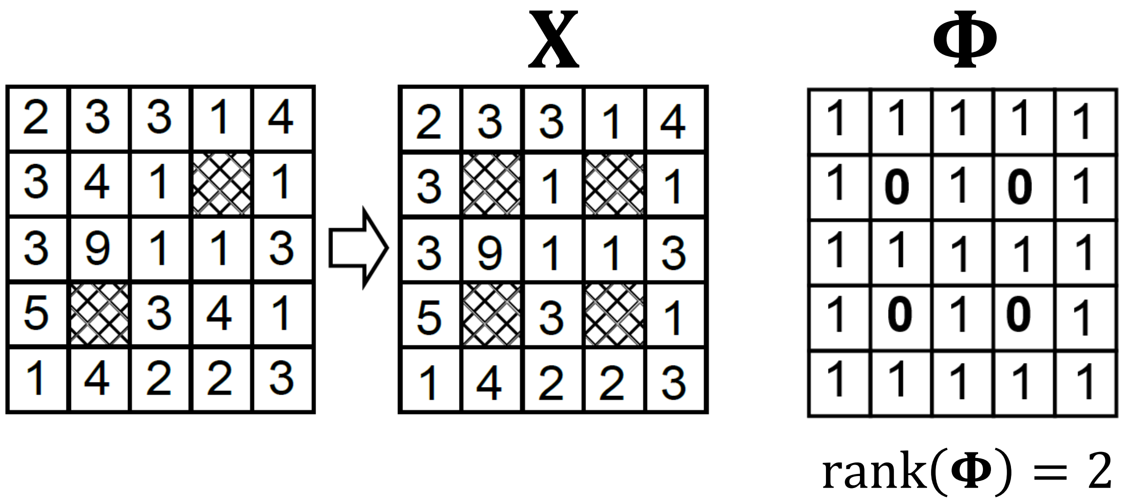
$$\boldsymbol{\Phi}_{ij} = \begin{cases} 0 & \text{If } \mathbf{X}_{ij} \text{ is missing} \\ 1 & \text{otherwise} \end{cases}$$

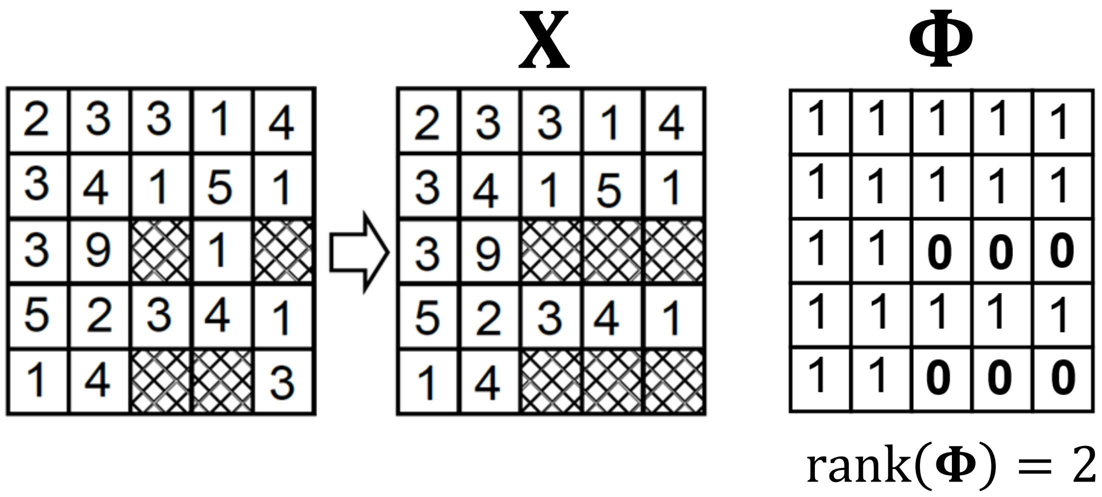


Theoretical Remarks 1 : Extended NMMF.

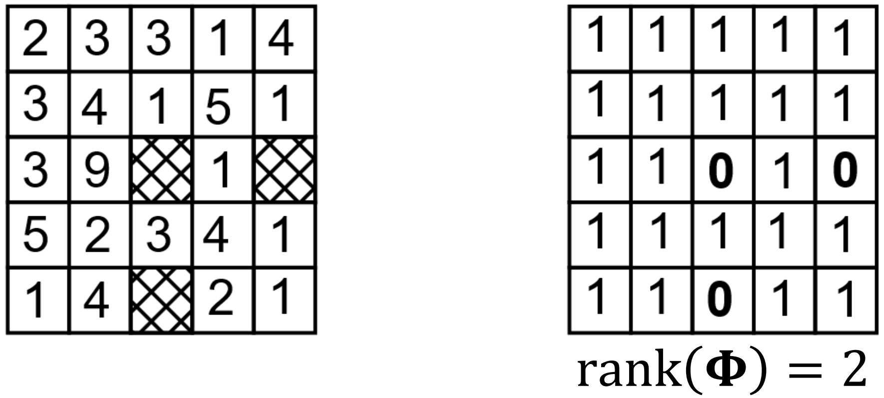
- ☐ The rank of weight matrix is 2 after adding missing values.

$$\mathbf{\Phi}_{ij} = \begin{cases} 0 & \text{If } \mathbf{X}_{ij} \text{ is missing} \\ 1 & \text{otherwise} \end{cases}$$

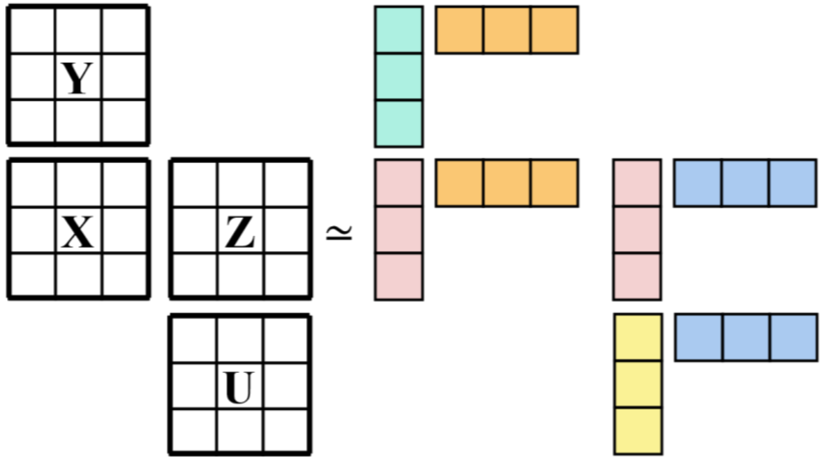




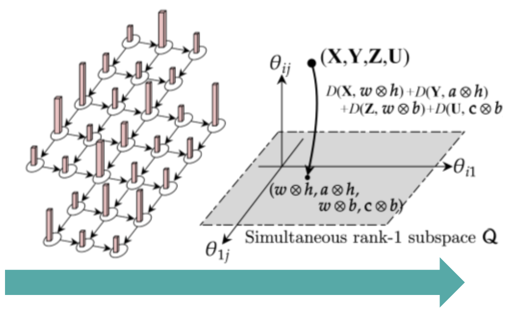
- ☐ Can we exactly solve rank-1 NMF if the rank($\mathbf{\Phi}$) = 2?



Theoretical Remarks 1 : Extended NMMF.



$$D(\mathbf{X}, w \otimes h) + \alpha D(\mathbf{Y}, a \otimes h) + \beta D(\mathbf{Z}, w \otimes b) + \gamma D(\mathbf{U}, c \otimes b)$$



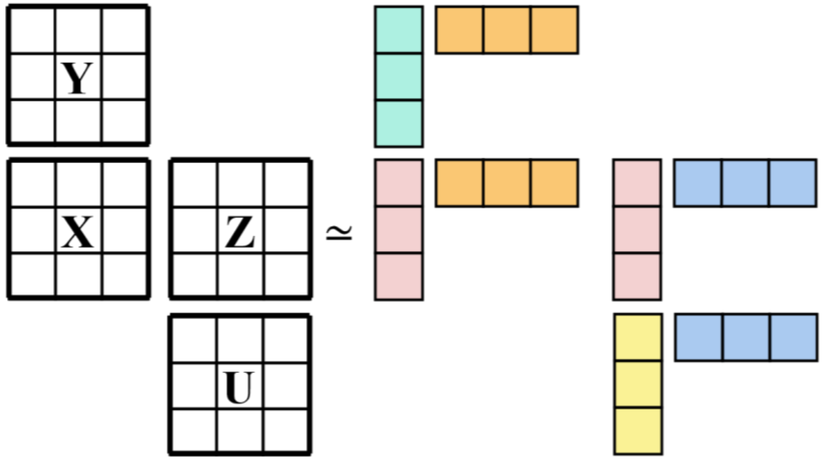
The best rank-1 approximation of extended NMMF

$$w_i = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{X}) + \beta S(\mathbf{Z})} \left\{ \sum_{j=1}^{J} \mathbf{X}_{ij} + \beta \sum_{m=1}^{M} \mathbf{Z}_{im} \right\} \qquad a_n = \frac{\sum_{j=1}^{J} \mathbf{Y}_{nj}}{\sqrt{S(\mathbf{X})}}$$

$$h_j = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{X}) + \alpha S(\mathbf{Y})} \left\{ \sum_{i=1}^{I} \mathbf{X}_{ij} + \alpha \sum_{n=1}^{N} \mathbf{Y}_{nj} \right\} \qquad c_l = \frac{\sqrt{S(X)}}{S(\mathbf{Z})} \sum_{m=1}^{M} \mathbf{U}_{lm}$$

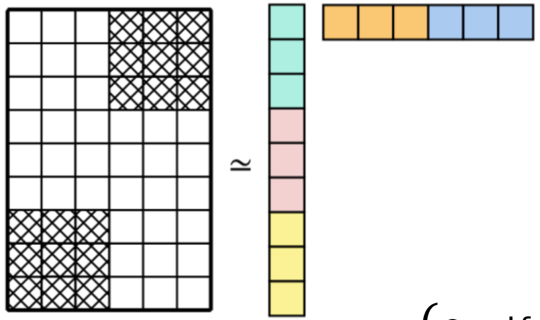
$$b_m = \frac{S(\mathbf{Z})}{\beta S(\mathbf{Z}) + \gamma S(\mathbf{U})} \frac{1}{\sqrt{S(\mathbf{X})}} \left\{ \beta \sum_{i=1}^{I} \mathbf{Z}_{im} + \gamma \sum_{l=1}^{L} \mathbf{U}_{lm} \right\}$$

Theoretical Remarks 1 : Extended NMMF.

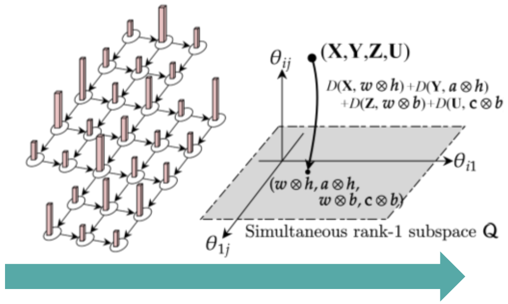


$$D(\mathbf{X}, w \otimes h) + \alpha D(\mathbf{Y}, a \otimes h) + \beta D(\mathbf{Z}, w \otimes b) + \gamma D(\mathbf{U}, c \otimes b)$$

Equivalent



$$D(\boldsymbol{\Phi} \circ \mathbf{X}, \boldsymbol{\Phi} \circ (\boldsymbol{w} \otimes \boldsymbol{h})) \qquad \boldsymbol{\Phi}_{ij} = \begin{cases} 0 & \text{If } \mathbf{X}_{ij} \text{ is missing} \\ 1 & \text{otherwise} \end{cases}$$



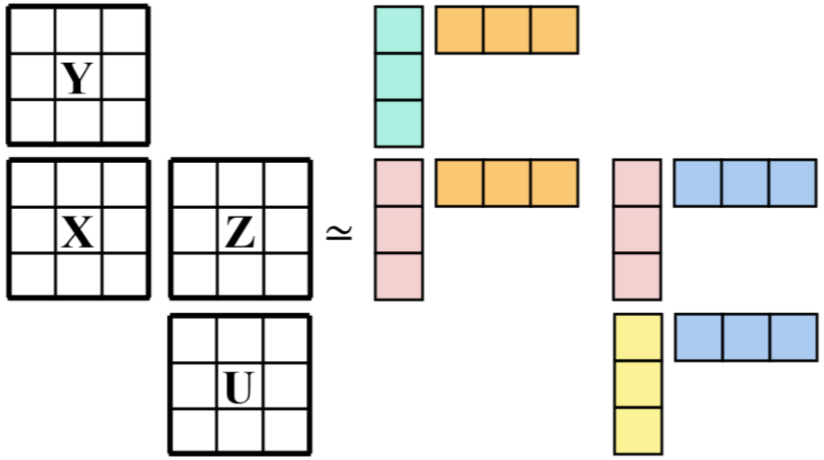
The best rank-1 approximation of extended NMMF

$$w_i = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{X}) + \beta S(\mathbf{Z})} \left\{ \sum_{j=1}^{J} \mathbf{X}_{ij} + \beta \sum_{m=1}^{M} \mathbf{Z}_{im} \right\} \qquad a_n = \frac{\sum_{j=1}^{J} \mathbf{Y}_{nj}}{\sqrt{S(\mathbf{X})}}$$

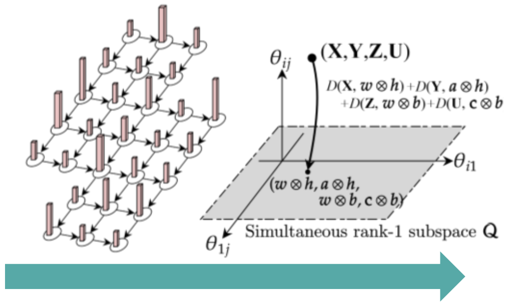
$$h_j = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{X}) + \alpha S(\mathbf{Y})} \left\{ \sum_{i=1}^{I} \mathbf{X}_{ij} + \alpha \sum_{n=1}^{N} \mathbf{Y}_{nj} \right\} \qquad c_l = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{Z})} \sum_{m=1}^{M} \mathbf{U}_{lm}$$

$$b_m = \frac{S(\mathbf{Z})}{\beta S(\mathbf{Z}) + \gamma S(\mathbf{U})} \frac{1}{\sqrt{S(\mathbf{X})}} \left\{ \beta \sum_{i=1}^{I} \mathbf{Z}_{im} + \gamma \sum_{l=1}^{L} \mathbf{U}_{lm} \right\}$$

Theoretical Remarks 1 : Extended NMMF.



$$D(\mathbf{X}, w \otimes h) + \alpha D(\mathbf{Y}, a \otimes h) + \beta D(\mathbf{Z}, w \otimes b) + \gamma D(\mathbf{U}, c \otimes b)$$



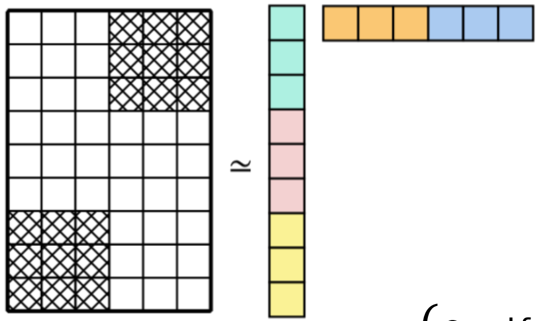
The best rank-1 approximation of extended NMMF

$$w_i = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{X}) + \beta S(\mathbf{Z})}\left\{\sum_{j=1}^{J} \mathbf{X}_{ij} + \beta \sum_{m=1}^{M} \mathbf{Z}_{im}\right\} \qquad a_n = \frac{\sum_{j=1}^{J} \mathbf{Y}_{nj}}{\sqrt{S(\mathbf{X})}}$$

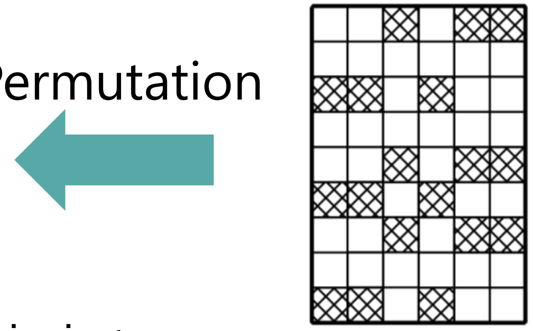
$$h_j = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{X}) + \alpha S(\mathbf{Y})}\left\{\sum_{i=1}^{I} \mathbf{X}_{ij} + \alpha \sum_{n=1}^{N} \mathbf{Y}_{nj}\right\} \qquad c_l = \frac{\sqrt{S(\mathbf{X})}}{S(\mathbf{Z})} \sum_{m=1}^{M} \mathbf{U}_{lm}$$

$$b_m = \frac{S(\mathbf{Z})}{\beta S(\mathbf{Z}) + \gamma S(\mathbf{U})} \frac{1}{\sqrt{S(\mathbf{X})}}\left\{\beta \sum_{i=1}^{I} \mathbf{Z}_{im} + \gamma \sum_{l=1}^{L} \mathbf{U}_{lm}\right\}$$

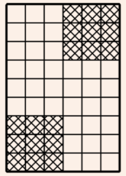
Equivalent



Permutation



If rank($\mathbf{\Phi}$) ≦2,
the matrix can be transformed into the form



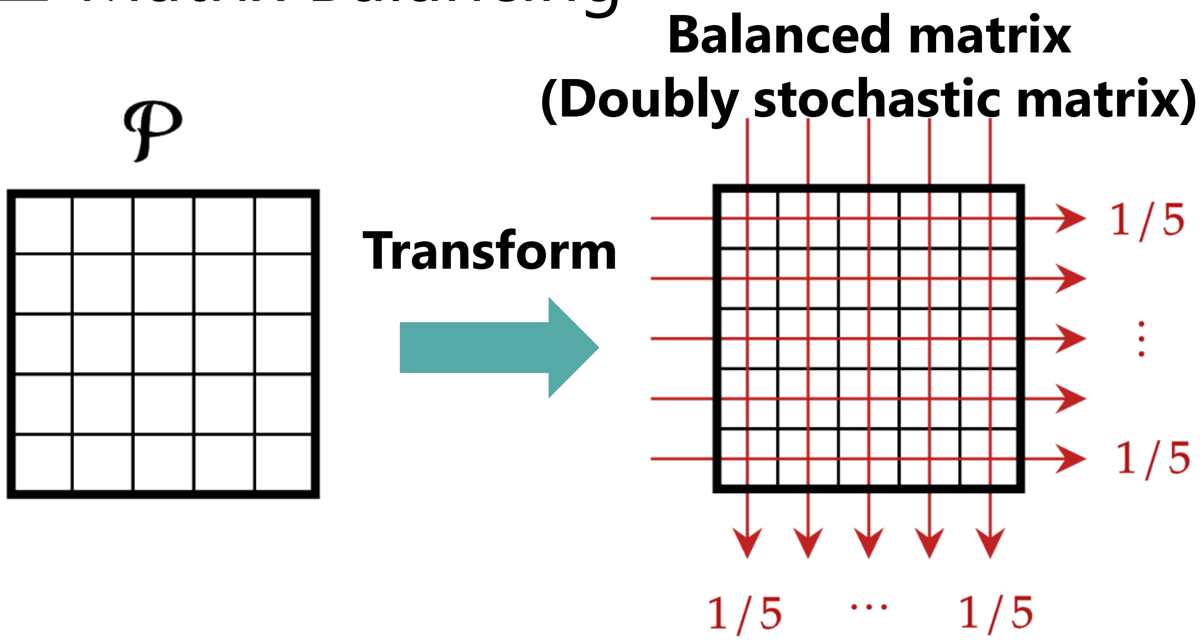
$$D(\mathbf{\Phi} \circ \mathbf{X}, \mathbf{\Phi} \circ (\boldsymbol{w} \otimes \boldsymbol{h})) \qquad \mathbf{\Phi}_{ij} = \begin{cases} 0 & \text{If } \mathbf{X}_{ij} \text{ is missing} \\ 1 & \text{otherwise} \end{cases}$$

We can exactly solve rank-1 NMF with missing values by permutation if rank(Φ) ≦2.

Theoretical Remarks 2 : Connection to balancing.

- Matrix Balancing

Mahito Sugiyama, Hiroyuki Nakahara and Koji Tsuda
"Tensor balancing on statistical manifold"(2017) ICML.

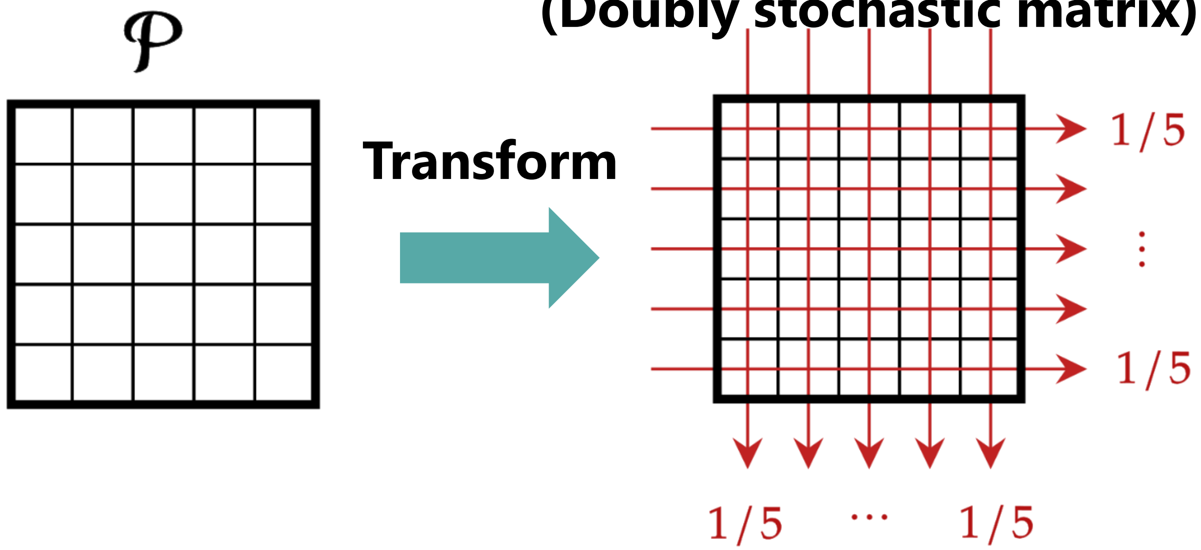


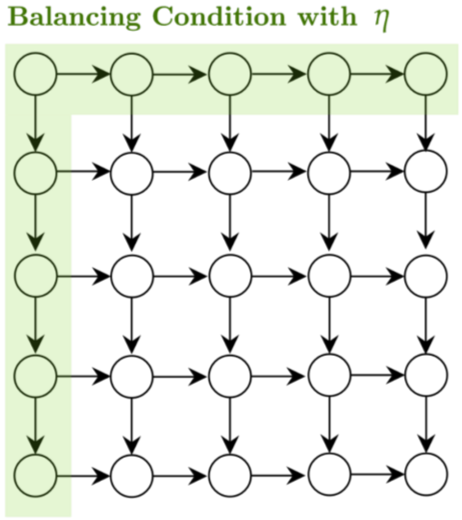
Theoretical Remarks 2 : Connection to balancing.

□ Matrix Balancing

Mahito Sugiyama, Hiroyuki Nakahara and Koji Tsuda
"Tensor balancing on statistical manifold"(2017) ICML.

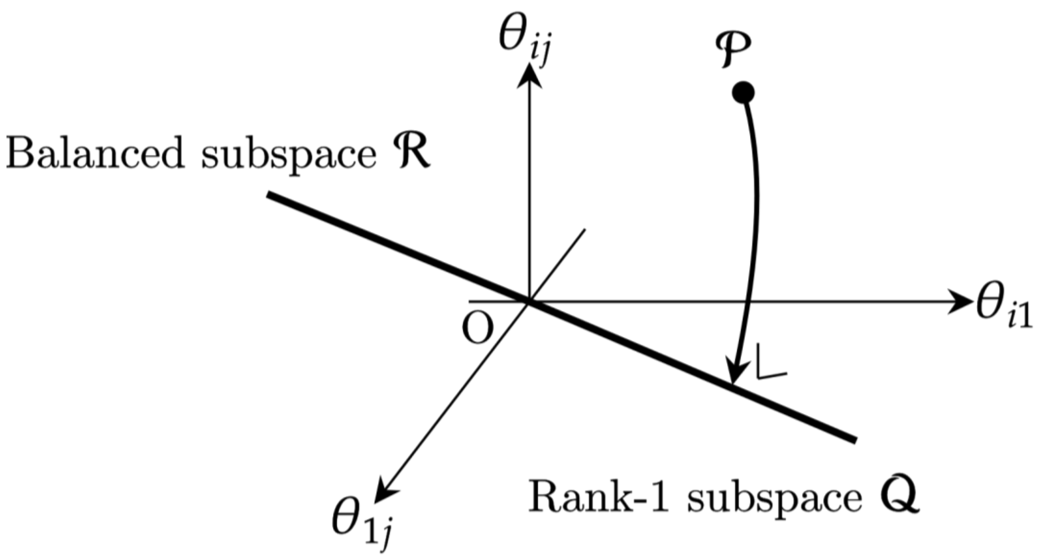
Balanced matrix
(Doubly stochastic matrix)





Balancing condition

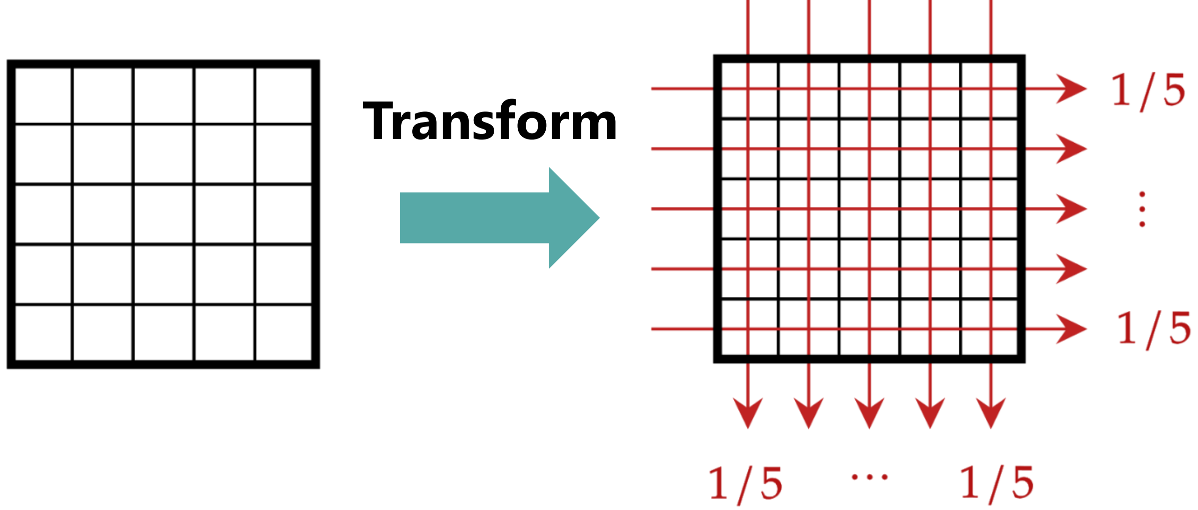
$$\eta_{i1} = \eta_{1i} = \frac{n-i+1}{n}$$

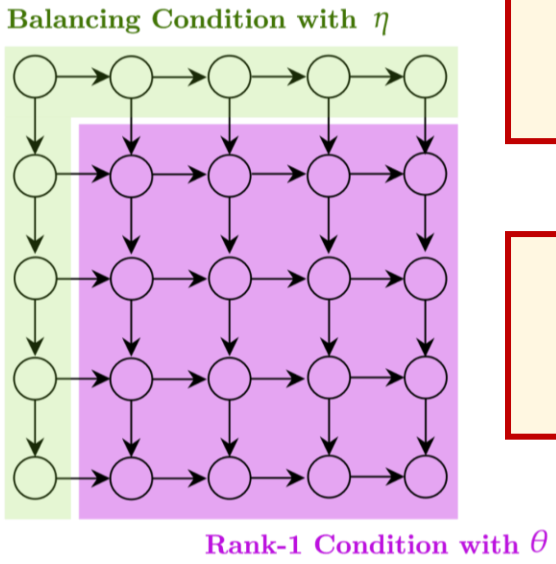


Theoretical Remarks 2 : Connection to balancing.

☐ Matrix Balancing

Balanced matrix (Doubly stochastic matrix)



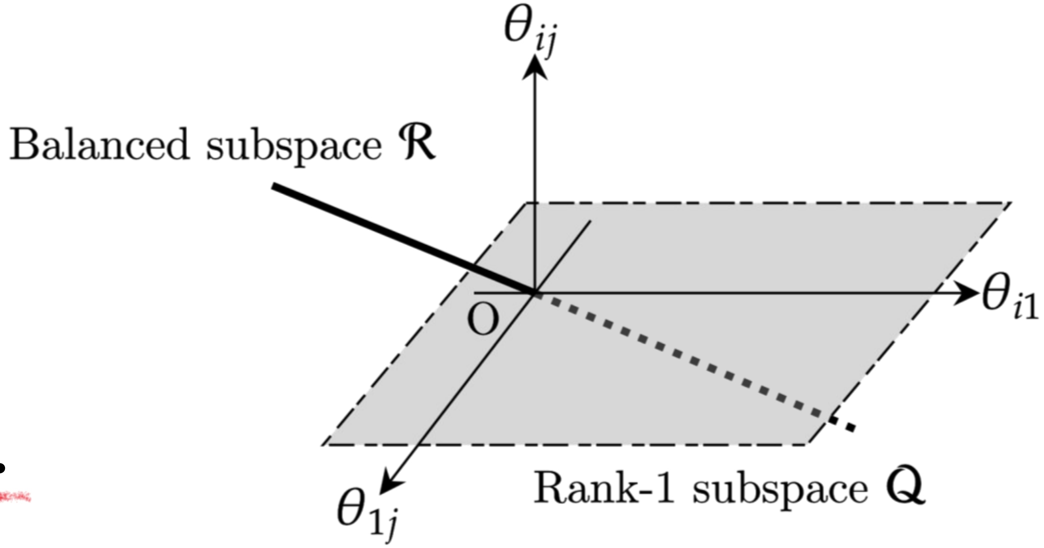


Balancing condition

$$\eta_{i1} = \eta_{1i} = \frac{n-i+1}{n}$$

Rank-1 condition

Its all many-body θ-parameters are 0.



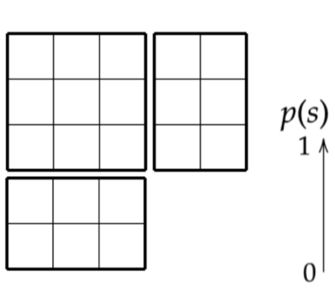
Balanced rank-1 matrix is unique.

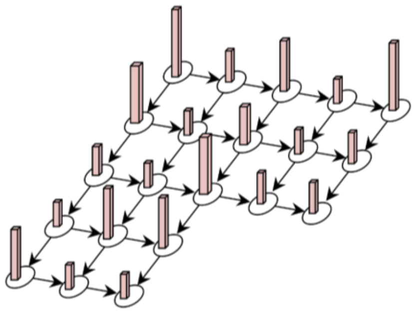
Conclusion

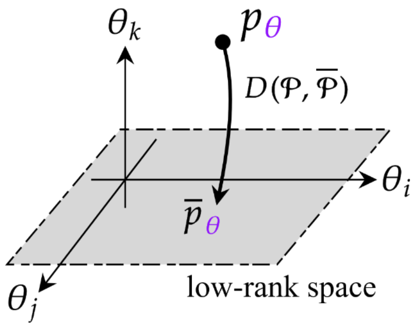
Data structure

DAG

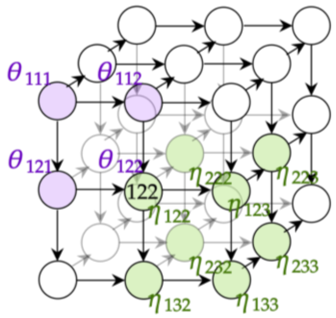
Infor-Geo







☐ Describe low-rank condition using (θ,η)



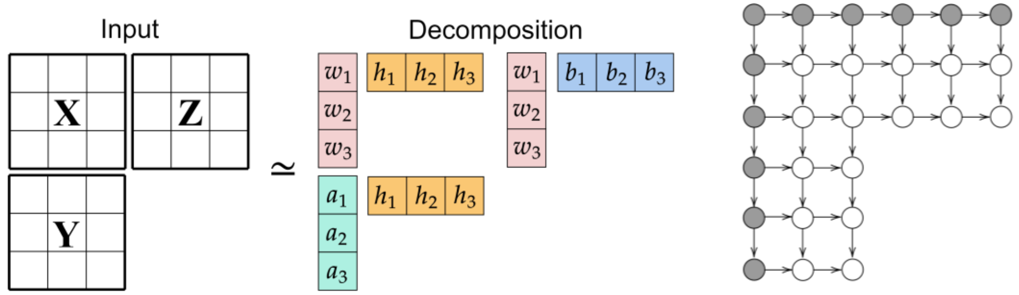
Rank-1 condition (**η**-representation)

$$\bar{\eta}_{ijk} = \bar{\eta}_{i11}\bar{\eta}_{1j1}\bar{\eta}_{11k}$$

Rank-1 condition (**θ**-representation)

All many body $\bar{\theta}_{ijk}$ are 0

☐ Closed Formula of the Best Rank-1 NMMF



The best rank-1 approximation for NMMF

For given $\mathbf{X} \in \mathbb{R}^{I\times J}_{>0}$, $\mathbf{Y} \in \mathbb{R}^{N\times J}_{>0}$, and $\mathbf{Z} \in \mathbb{R}^{I\times M}_{>0}$ the best rank-1 NMMF is given as

$$w_i = \frac{\sqrt{S(\mathrm{X})}}{S(\mathrm{X}) + \beta S(\mathrm{Z})}\left\{\sum_{j=1}^{J} \mathrm{X}_{ij} + \beta \sum_{m=1}^{M} \mathrm{Z}_{im}\right\} \qquad a_n = \frac{\sum_{j=1}^{J} \mathrm{Y}_{nj}}{\sqrt{S(\mathrm{X})}}$$

$$h_j = \frac{\sqrt{S(\mathrm{X})}}{S(\mathrm{X}) + \alpha S(\mathrm{Y})}\left\{\sum_{i=1}^{I} \mathrm{X}_{ij} + \alpha \sum_{n=1}^{N} \mathrm{Y}_{nj}\right\} \qquad b_m = \frac{\sum_{i=1}^{I} \mathrm{Z}_{im}}{\sqrt{S(\mathrm{X})}}$$

$S(\mathbf{X})$ is sum of all elements of $\mathbf{X}$.

☐ A1GM: Faster Rank-1 NMF with missing values