Tensor Balancing on Statistical Manifold

Mahito Sugiyama$^{1,2}$ Hiroyuki Nakahara$^3$ Koji Tsuda$^{4,5,6}$

$^1$NII, $^2$JST PRESTO, $^3$RIKEN BSI, $^4$UTokyo, $^5$RIKEN AIP, $^6$NIMS
Results

- **Balancing** of higher order (more than two) tensors is firstly (theoretically) achieved
  - We present a balancing algorithm and prove its global convergence

- A fast balancing algorithm with **quadratic convergence** using Newton’s method
  - An existing algorithm is linear convergence

- **[Theory]** We provide dually flat Riemannian manifold of probability distributions with the structured outcome space
  - Information Geometry
  - Tensor balancing is an instance
Matrix Balancing

\[
\begin{bmatrix}
\rho_{11} & \rho_{12} & \rho_{13} & \rho_{14} \\
\rho_{21} & \rho_{22} & \rho_{23} & \rho_{24} \\
\rho_{31} & \rho_{32} & \rho_{33} & \rho_{34} \\
\rho_{41} & \rho_{42} & \rho_{43} & \rho_{44}
\end{bmatrix}
\]
Matrix Balancing

Find \( r \) and \( a \)

\[
\begin{bmatrix}
  p_{11} & p_{12} & p_{13} & p_{14} \\
  p_{21} & p_{22} & p_{23} & p_{24} \\
  p_{31} & p_{32} & p_{33} & p_{34} \\
  p_{41} & p_{42} & p_{43} & p_{44}
\end{bmatrix}
\begin{bmatrix}
  r_1 \\
  r_2 \\
  r_3 \\
  r_4
\end{bmatrix}
\begin{bmatrix}
  a_1 \\
  a_2 \\
  a_3 \\
  a_4
\end{bmatrix}
\]

\[
r_2a_1p_{21} + r_2a_2p_{22} + r_2a_3p_{23} + r_2a_4p_{24} = 1
\]

\[
r_1a_3p_{13} + r_2a_3p_{23} + r_3a_3p_{33} + r_4a_3p_{43} = 1
\]
Matrix Balancing

- Problem setting:
  Given a nonnegative matrix \( P = (p_{ij}) \in \mathbb{R}^{n \times n}_+ \), find \( r, s \in \mathbb{R}^n \) s.t.

\[
(RPS) \mathbf{1} = \mathbf{1} \quad \text{and} \quad (RPS)^T \mathbf{1} = \mathbf{1}
\]

- \( R = \text{diag}(r) \), \( S = \text{diag}(s) \)
- Each entry is given as \( p'_{ij} = p_{ij} r_i s_j \)

- A fundamental process to analyze and compare matrices in a wide range of applications
  - Input-output analysis in economics, seat assignments in elections, Hi-C data analysis, Sudoku puzzle
  - Approximate Wasserstein distance
Results on Hessenberg Matrix

<table>
<thead>
<tr>
<th>n</th>
<th>Number of iterations</th>
<th>Running time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>$10^{10}$</td>
<td>$10^{8}$</td>
</tr>
<tr>
<td>50</td>
<td>$10^{10}$</td>
<td>$10^{6}$</td>
</tr>
<tr>
<td>500</td>
<td>$10^{10}$</td>
<td>$10^{4}$</td>
</tr>
<tr>
<td>5000</td>
<td>$10^{10}$</td>
<td>$10^{4}$</td>
</tr>
</tbody>
</table>

- **Newton (proposed)**
- **Sinkhorn**
- **BNEWT**
Overview of Our Approach

Given tensor $P$

Multistochastic tensor $P'$

Tensor balancing

Every fiber sums to 1

Statistical manifold $S$ (dually flat Riemannian manifold)

Probability distribution $P$

Projection

Submanifold $S(\beta)$

Projected distribution $P_\beta$
Partially Ordered Set

- Partially ordered set (poset) \((S, \leq)\)
  
  (i) \(x \leq x\) (reflexivity)
  
  (ii) \(x \leq y, y \leq x \Rightarrow x = y\) (antisymmetry)
  
  (iii) \(x \leq y, y \leq z \Rightarrow x \leq z\) (transitivity)
    
    - We assume that \(S\) is finite and includes the least element (bottom) \(\bot \in S\)

- Equivalent to a DAG
  
  - Each \(x \in S\) is a node
  
  - \(x \leq y \iff y\) is reachable from \(x\)
Log-Linear Model on Poset

Each \( x \in S \) has a triple:
\((p(x), \theta(x), \eta(x))\)

- A probability vector \( p: S \to (0, 1) \)
  s.t. \( \sum_{x \in S} p(x) = 1 \)
  - (Normalized) weight for each node

- We introduce \( \theta: S \to \mathbb{R} \) and \( \eta: S \to \mathbb{R} \) as
  \[
  \log p(x) = \sum_{s \leq x} \theta(s),
  \]
  \[
  \eta(x) = \sum_{s \geq x} p(s)
  \]
Our Model Includes Binary Case

- Our model:
  \[
  \log p(x) = \sum_{s \leq x} \theta(s), \quad \eta(x) = \sum_{s \geq x} p(s)
  \]
  is generalization of the log-linear model on binary vectors with \( x \in \{0, 1\}^n = S \):

  \[
  \log p(x) = \sum_i \theta^i x^i + \sum_{i < j} \theta^{ij} x^i x^j + \ldots
  \]
  \[
  + \theta^{1\ldots n} x^1 x^2 \ldots x^n - \psi,
  \]

  \[
  \eta^i = \mathbb{E}[x^i] = \Pr(x^i = 1),
  \]

  \[
  \eta^{ij} = \mathbb{E}[x^i x^j] = \Pr(x^i = x^j = 1),
  \]

  \frac{8}{19}
Dually Flat Structure

• \( \theta \) and \( \eta \) form a dual coordinate system:

\[
\nabla \psi(\theta) = \eta, \quad \nabla \varphi(\eta) = \theta
\]

- \( \psi(\theta) = -\theta(\perp) = -\log p(\perp), \quad \varphi(\eta) = \sum_{x \in S} p(x) \log p(x) \)

- \( \psi(\theta) \) and \( \varphi(\eta) \) are connected via the Legendre transformation:

\[
\varphi(\eta) = \max_{\theta'} \left( \theta' \eta - \psi(\theta') \right), \quad \theta' \eta = \sum_{x \in S \setminus \{\perp\}} \theta'(x) \eta(x)
\]

  ○ \( \psi(\theta) \) and \( \varphi(\eta) \) should be convex
Gradient and Riemannian Manifold

- The gradients: \( g(\theta) = \nabla \nabla \psi(\theta) = \nabla \eta, \ g(\eta) = \nabla \nabla \varphi(\eta) = \nabla \theta \)

\[
\begin{align*}
g_{xy}(\theta) &= \frac{\partial \eta(x)}{\partial \theta(y)} = \sum_{s \in S} \zeta(x, s) \zeta(y, s) p(s) - \eta(x) \eta(y) \\
g_{xy}(\eta) &= \frac{\partial \theta(x)}{\partial \eta(y)} = \sum_{s \in S} \mu(s, x) \mu(s, y) p(s)^{-1}
\end{align*}
\]

- \( \zeta \) and \( \mu \) are the zeta function and the Möbius function determined by the partial order (DAG) structure
- The manifold \((\mathcal{S}, g(\xi))\) is a Riemannian manifold with the set \( \mathcal{S} \) of probability vectors and the Riemannian metric \( g(\xi) \)
Möbius Function on Poset

- **Zeta function** $\zeta: S \times S \to \{0, 1\}$
  
  \[
  \zeta(s, x) = \begin{cases} 
  1 & \text{if } s \leq x, \\
  0 & \text{otherwise.}
  \end{cases}
  \]

- **Möbius function** $\mu: S \times S \to \mathbb{Z}$
  
  \[
  \mu(x, y) = \begin{cases} 
  1 & \text{if } x = y, \\
  - \sum_{x \leq s \leq y} \mu(x, s) & \text{if } x < y, \\
  0 & \text{otherwise}
  \end{cases}
  \]

- We have $\zeta \mu = I$ (convolutional inverse):
  
  \[
  \sum_{s \in S} \zeta(s, y) \mu(x, s) = \sum_{x \leq s \leq y} \mu(x, s) = \delta_{xy}
  \]
$e$-Projection and $m$-Projection

$e$-projection:

$\theta(x) = \alpha(x)$

$m$-projection:

$\eta(x) = \beta(x)$

$\forall x \in B = S^+ / A$
e-Projection and \( m \)-Projection

\[ \theta(x) = 0 \]

\[ \eta(x) = \eta(x) \]

\[
\text{MLE}
\]

Boltzmann machine

\[ e \text{-projection} \]

\[ m \text{-projection} \]

\[ \hat{P} \]

\[ R \]

\[ \text{Boltzmann machine} \]

\[ 000 \]

\[ 001010100 \]

\[ 011101110 \]

\[ 111 \]

\[ ^\hat{\text{ }} \]

\[ ^\hat{\text{ }} \]
Compute $e$-Projection by Newton’s Method

- Each step of Newton’s method:
  \[
  \begin{bmatrix}
  \eta^{(t)}_{P_{\beta}}(x) - \beta(x) \\
  \vdots \\
  \eta^{(t)}_{P_{\beta}}(x) - \beta(x)
  \end{bmatrix}
  + J
  \begin{bmatrix}
  \theta^{(t+1)}_{P_{\beta}}(y) - \theta^{(t)}_{P_{\beta}}(y) \\
  \vdots \\
  \theta^{(t+1)}_{P_{\beta}}(y) - \theta^{(t)}_{P_{\beta}}(y)
  \end{bmatrix}
  = 0,
  \]

- $J$ is the $|\text{dom}(\beta)| \times |\text{dom}(\beta)|$ Jacobian matrix given as
  \[
  J_{xy} = \frac{\partial \eta_{P_{\beta}}^{(t)}(x)}{\partial \theta_{P_{\beta}}^{(t)}(y)} = \sum_{s \in S} \zeta(x, s) \zeta(y, s) p_{\beta}^{(t)}(s) - \eta_{P_{\beta}}^{(t)}(x) \eta_{P_{\beta}}^{(t)}(y)
  \]
  for each $x, y \in \text{dom}(\beta)$
View Matrix as Poset

\[
\begin{bmatrix}
  p_{11} & p_{12} & p_{13} & p_{14} \\
  p_{21} & p_{22} & p_{23} & p_{24} \\
  p_{31} & p_{32} & p_{33} & p_{34} \\
  p_{41} & p_{42} & p_{43} & p_{44}
\end{bmatrix}
\]

\[
p_{11} \rightarrow p_{12} \rightarrow p_{13} \rightarrow p_{14}
\]

\[
p_{21} \rightarrow p_{22} \rightarrow p_{23} \rightarrow p_{24}
\]

\[
p_{31} \rightarrow p_{32} \rightarrow p_{33} \rightarrow p_{34}
\]

\[
p_{41} \rightarrow p_{42} \rightarrow p_{43} \rightarrow p_{44}
\]
Introduce \( \theta \) and \( \eta \)

\[
\begin{bmatrix}
\eta_{11} & \eta_{12} & \eta_{13} & \eta_{14} \\
\eta_{21} & \eta_{22} & \eta_{23} & \eta_{24} \\
\eta_{31} & \eta_{32} & \eta_{33} & \eta_{34} \\
\eta_{41} & \eta_{42} & \eta_{43} & \eta_{44}
\end{bmatrix}
\]

Matrix balancing is achieved if:
\( \eta_{11} = 4, \eta_{21} = 3, \eta_{31} = 2, \eta_{41} = 1 \)
\( \eta_{11} = 4, \eta_{12} = 3, \eta_{13} = 2, \eta_{14} = 1 \)
Introduce $\theta$ and $\eta$

Matrix balancing is achieved if:

$$
\begin{bmatrix}
\eta_{11} & \eta_{12} & \eta_{13} & \eta_{14} \\
\eta_{21} & \eta_{22} & \eta_{23} & \eta_{24} \\
\eta_{31} & \eta_{32} & \eta_{33} & \eta_{34} \\
\eta_{41} & \eta_{42} & \eta_{43} & \eta_{44}
\end{bmatrix}
$$

<table>
<thead>
<tr>
<th>$\eta_{11}$</th>
<th>$\eta_{12}$</th>
<th>$\eta_{13}$</th>
<th>$\eta_{14}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$$
\begin{bmatrix}
\theta_{11} & \theta_{12} & \theta_{13} & \theta_{14} \\
\theta_{21} & \theta_{22} & \theta_{23} & \theta_{24} \\
\theta_{31} & \theta_{32} & \theta_{33} & \theta_{34} \\
\theta_{41} & \theta_{42} & \theta_{43} & \theta_{44}
\end{bmatrix}
$$

Matrix balancing is achieved if:

- $\eta_{11} = 4, \eta_{21} = 3, \eta_{31} = 2, \eta_{41} = 1$
- $\eta_{11} = 4, \eta_{12} = 3, \eta_{13} = 2, \eta_{14} = 1$
Introduce $\theta$ and $\eta$

Matrix balancing is achieved if:

$\eta_{11} = 4, \eta_{21} = 3, \eta_{31} = 2, \eta_{41} = 1$

$\eta_{11} = 4, \eta_{12} = 3, \eta_{13} = 2, \eta_{14} = 1$
Introduce $\theta$ and $\eta$

\[
\begin{bmatrix}
\eta_{11} & \eta_{12} & \eta_{13} & \eta_{14} \\
\eta_{21} & \eta_{22} & \eta_{23} & \eta_{24} \\
\eta_{31} & \eta_{32} & \eta_{33} & \eta_{34} \\
\eta_{41} & \eta_{42} & \eta_{43} & \eta_{44}
\end{bmatrix}
\]

Matrix balancing is achieved if:
- $\eta_{11} = 4, \eta_{21} = 3, \eta_{31} = 2, \eta_{41} = 1$
- $\eta_{11} = 4, \eta_{12} = 3, \eta_{13} = 2, \eta_{14} = 1$
Introduce $\theta$ and $\eta$

Matrix balancing is achieved if:

$\eta_{11} = 4, \eta_{21} = 3, \eta_{31} = 2, \eta_{41} = 1$

$\eta_{11} = 4, \eta_{12} = 3, \eta_{13} = 2, \eta_{14} = 1$

$$
\begin{bmatrix}
\eta_{11} & \eta_{12} & \eta_{13} & \eta_{14} \\
\eta_{21} & \eta_{22} & \eta_{23} & \eta_{24} \\
\eta_{31} & \eta_{32} & \eta_{33} & \eta_{34} \\
\eta_{41} & \eta_{42} & \eta_{43} & \eta_{44}
\end{bmatrix}
$$
Matrix balancing is achieved if:

$\eta_{11} = 4, \eta_{21} = 3, \eta_{31} = 2, \eta_{41} = 1$

$\eta_{11} = 4, \eta_{12} = 3, \eta_{13} = 2, \eta_{14} = 1$

$\eta_{11} \eta_{12} \eta_{13} \ldots \theta_{42} \theta_{43} \theta_{44}$
Given tensor \( \eta_{ii} = \eta_{ii} = i \)

Balanced tensors

\[ \eta = \begin{pmatrix} \eta_{11} & \eta_{12} & \eta_{13} & \eta_{14} \\ \eta_{21} & \theta_{22} & \theta_{23} & \theta_{24} \\ \eta_{31} & \theta_{32} & \theta_{33} & \theta_{34} \\ \eta_{41} & \theta_{42} & \theta_{43} & \theta_{44} \end{pmatrix} \]
Conclusion

- We have achieved efficient tensor balancing with Newton’s method
- We have introduced the dually flat structure into distribution of partially ordered outcome space
  - e-projection =
    - Tensor balancing
    - Maximum Likelihood Estimation
- Partial order structure (discrete) + Information geometry (continuous) = efficient and effective data analysis methods!
Möbius Inversion

• The Möbius inversion formula [Rota (1964)]:

\[
g(x) = \sum_{s \in S} \zeta(s, x)f(s) = \sum_{s \leq x} f(s)
\]

\[
\iff f(x) = \sum_{s \in S} \mu(s, x)g(s),
\]

A-1/A-10
Möbius Function Is Generalization of Inclusion-Exclusion Principle

- For sets $A, B, C$,
  \[ |A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C| \]

- In general, for $A_1, A_2, \ldots, A_n$,
  \[
  \left| \bigcup_{i} A_i \right| = \sum_{J \subseteq \{1, \ldots, n\}, J \neq \emptyset} (-1)^{|J|-1} \left| \bigcap_{j \in J} A_j \right|
  \]

- The Möbius function $\mu$ is the generalization of “$(-1)^{|J|-1}$”
Fisher Information Matrix and Orthogonality

- Since \( g(\xi) \) coincides with the Fisher information matrix,
  \[
  E \left[ \frac{\partial}{\partial \theta(x)} \log p(s) \frac{\partial}{\partial \theta(y)} \log p(s) \right] = \sum_{s \in S} \zeta(x, s) \zeta(y, s) p(s) - \eta(x) \eta(y),
  \]
  \[
  E \left[ \frac{\partial}{\partial \eta(x)} \log p(s) \frac{\partial}{\partial \eta(y)} \log p(s) \right] = \sum_{s \in S} \mu(s, x) \mu(s, y) p(s)^{-1}
  \]
- \( \theta \) and \( \eta \) are orthogonal, i.e.,
  \[
  E \left[ \frac{\partial}{\partial \theta(x)} \log p(s) \frac{\partial}{\partial \eta(y)} \log p(s) \right] = \sum_{s \in S} \zeta(x, s) \mu(s, y) = \delta_{xy}
  \]
$m$-Projection

- Submanifold by $\beta$: $\mathcal{S}(\beta) = \{ P \in \mathcal{S} \mid \theta_P(x) = \beta(x), \ \forall x \in \text{dom}(\beta) \}$

- $m$-projection of $P \in \mathcal{S}$ onto $\mathcal{S}(\beta)$ is $P_\beta \in \mathcal{S}(\beta)$ s.t.
  \[
  \begin{cases}
  \theta_{P_\beta}(x) = \beta(x) & \text{if } x \in \text{dom}(\beta), \\
  \eta_{P_\beta}(x) = \eta_P(x) & \text{if } x \in (S \setminus \{\perp\}) \setminus \text{dom}(\beta)
  \end{cases}
  \]
  - This is the minimizer of the KL divergence from $P$ to $\mathcal{S}(\beta)$:
    \[P_\beta = \arg\min_{Q \in \mathcal{S}(\beta)} D_{\text{KL}}[P, Q]\]
  - The projected distribution $P_\beta$ always uniquely exists

- Pythagorean theorem: $D_{\text{KL}}[P, Q] = D_{\text{KL}}[P, P_\beta] + D_{\text{KL}}[P_\beta, Q]$ for all $Q \in \mathcal{S}(\beta)$
e-Projection

• Submanifold by $\beta$: $\mathcal{S}(\beta) = \{P \in \mathcal{S} \mid \eta_P(x) = \beta(x), \forall x \in \text{dom}(\beta)\}$

• e-projection of $P \in \mathcal{S}$ onto $\mathcal{S}(\beta)$ is $P_\beta \in \mathcal{S}(\beta)$ s.t.
  \[
  \begin{cases}
  \theta_{P_\beta}(x) = \theta_P(x) & \text{if } x \in (\mathcal{S} \setminus \{\perp\}) \setminus \text{dom}(\beta), \\
  \eta_{P_\beta}(x) = \beta(x) & \text{if } x \in \text{dom}(\beta)
  \end{cases}
  \]

  – This is the minimizer of the KL divergence from $P$ to $\mathcal{S}(\beta)$:
  \[P_\beta = \arg\min_{Q \in \mathcal{S}(\beta)} D_{KL}[P, Q]\]

  – The projected distribution $P_\beta$ always uniquely exists

• Pythagorean theorem: $D_{KL}[P, Q] = D_{KL}[P, P_\beta] + D_{KL}[P_\beta, Q]$ for all $Q \in \mathcal{S}(\beta)$
Computation of e-Projection

- Given $P$ and $\beta$, we compute $P_\beta$ such that
  \[
  \begin{cases}
  \theta_{P_\beta}(x) = \theta_{P}(x) & \text{if } x \in (S \setminus \{\perp\}) \setminus \text{dom}(\beta), \\
  \eta_{P_\beta}(x) = \beta(x) & \text{if } x \in \text{dom}(\beta)
  \end{cases}
  \]

- Initialize with $P^{(o)}_\beta = P$ and, at each step $t$,
  update $\eta^{(t)}_{P_\beta}(x)$ for $x \in \text{dom}(\beta)$
  - Since $\theta$ and $\eta$ are orthogonal, we can change $\eta^{(t)}_{P_\beta}(x)$
    while fixing $\theta^{(t)}_{P_\beta}(y)$ for $y \notin \text{dom}(\beta)$
Matrix And Tensor Balancing

- Given a nonnegative matrix $P = (p_{ij}) \in \mathbb{R}_{+}^{n \times n}$, find $r, s \in \mathbb{R}^n$ s.t.
  $$(RPS)\mathbf{1} = \mathbf{1} \quad \text{and} \quad (RPS)^T \mathbf{1} = \mathbf{1},$$
  where $R = \text{diag}(r), S = \text{diag}(s)$

- Given a tensor $P \in \mathbb{R}^{n_1 \times n_2 \times \cdots \times n_N}$ with $n_1 = \cdots = n_N = n$, find $(N-1)$ order tensors $R^1, R^2, \ldots, R^N$ s.t. $\forall m \in [N]$ $P' \times_m \mathbf{1} = \mathbf{1} \in \mathbb{R}^{n_1 \times \cdots \times n_{m-1} \times n_{m+1} \times \cdots \times n_N}$
  - Each entry $p'_{i_1i_2\ldots i_N}$ of the balanced tensor $P'$ is given as
    $$p'_{i_1i_2\ldots i_N} = p_{i_1i_2\ldots i_N} \prod_{m \in [N]} R^m_{i_1\ldots i_{m-1}i_{m+1}\ldots i_N}$$
  - The balanced tensor $P'$ is called multistochastic
Matrix balancing is achieved if:
\[
\begin{align*}
\eta_{11} &= 4, \quad \eta_{21} = 3, \quad \eta_{31} = 2, \quad \eta_{41} = 1 \\
\eta_{11} &= 4, \quad \eta_{12} = 3, \quad \eta_{13} = 2, \quad \eta_{14} = 1
\end{align*}
\]
Results on Hessenberg Matrix ($n = 20$)

Number of iterations

Residual

$10^{-7}$

$10^{-5}$

$10^{-3}$

$10^{-1}$

$10^{0}$

$10^{1}$

$10^{2}$

$10^{3}$

$10^{4}$

$10^{5}$

$10^{6}$

$10^{7}$

$10^{8}$

$10^{9}$

$10^{10}$

Sinkhorn

Newton

BNEWT

Number of iterations

Residual

$10^{-7}$

$10^{-5}$

$10^{-3}$

$10^{-1}$

$10^{0}$

$10^{1}$

$10^{2}$

$10^{3}$

$10^{4}$

$10^{5}$

$10^{6}$

$10^{7}$

$10^{8}$

$10^{9}$

$10^{10}$

Sinkhorn

Newton

BNEWT

A-9/A-10
Results on Trefethen Matrix

<table>
<thead>
<tr>
<th>$n$</th>
<th>Number of iterations</th>
<th>Running time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>10</td>
<td>$10^{-3}$</td>
</tr>
<tr>
<td>100</td>
<td>$10^{3}$</td>
<td>$10^{-1}$</td>
</tr>
<tr>
<td>200</td>
<td>$10^{6}$</td>
<td>$10^{1}$</td>
</tr>
<tr>
<td>300</td>
<td>$10^{9}$</td>
<td>$10^{3}$</td>
</tr>
</tbody>
</table>

Graphs showing the number of iterations and running time for Newton (proposed), Sinkhorn, and BNEWT methods as a function of $n$. The number of iterations generally increases with $n$, while the running time increases significantly with larger $n$ for all methods.