Tensor Balancing on Statistical Manifold
Mahito Sugiyama (NII, JST PRESTO), Hiroyuki Nakahara (RIKEN BSI), Koji Tsuda (UTokyo, RIKEN AIP, NIMS)
The 34th International Conference on Machine Learning (ICML 2017), August 6–11, 2017

Results
- Balancing of higher order (more than two) tensors is firstly (theoretically) achieved
- A fast balancing algorithm with quadratic convergence using Newton’s method (an existing algorithm is linear convergence)

[Theory] We provide a convolutional and flat Riemannian manifold of probability distributions with the structured outcome space

Matrix (Tensor) Balancing

Each $x \in \mathcal{S}$ has a triple:

$$(p(x), \theta(x), \eta(x))$$

- Let $(S, s)$ be a poset (= DAG)
- A probability vector $p: S \to (0, 1)$ s.t. $\sum_{s \in S} p(x) = 1$
- (Normalized) weight for each node
- We introduce $\theta: S \to \mathbb{R}$ and $\eta: S \to \mathbb{R}$ as

$$\log p(x) = \sum_{s \in S} \theta(s), \eta(x) = \sum_{s \in S} p(s)$$

- Our model is generalization of the log-linear model on binary vectors with $x \in \{0, 1\}^n$:

$$\log p(x) = \sum_{s \in S} \theta(s) x^i + \sum_{i<j} \theta_{i,j} x^i x^j + \ldots + \theta_{1,1} x^1 \ldots x^n - \psi, \quad \eta^i = E[x^i] = Pr(x^i = 1),$$


- Dually Flat Structure

• $\theta$ and $\eta$ form a dual coordinate system: $\nabla \phi(\theta) = \eta, \nabla \phi(\eta) = \theta$
- $\phi(\theta) = -\theta(1) = -\log p(1), \phi(\eta) = \sum_{s \in S} p(x) \log p(x)$
- $\phi(\theta)$ and $\phi(\eta)$ are connected via the Legendre transformation:

$$\phi(\eta) = \max \{ \theta(\eta - \phi(\theta)) \}, \quad \theta(\eta) = \sum_{s \in S} \phi(s) \theta(x) \eta(x)$$

- The gradients: $g(\theta) = \nabla \phi(\theta) = \nabla \eta, g(\eta) = \nabla \phi(\eta) = \nabla \theta$

$$\begin{cases}
g_s(\theta) = \frac{\partial \eta(x)}{\partial x^i} = \sum_{s \in S} (x^i \cdot s \cdot y \cdot p(s) - \eta(x) \eta(y)) \\
g_s(\eta) = \frac{\partial \theta(x)}{\partial x^i} = \sum_{s \in S} (s \cdot y \cdot p(s) - \eta(x) \eta(y))
\end{cases}$$

- Zeta function: $\zeta(x, s) = 1$ if $s \leq x, \zeta(x, s) = 0$ otherwise
- Möbius function: $\mu: S \times S \to \mathbb{Z}$ satisfying $\zeta(x, s) = 1$
- The manifold $(\mathcal{S}, g(\xi))$ is a Riemannian manifold with the set $\mathcal{S}$ of probability vectors and the Riemannian metric $g(\xi)$

Log-Linear on Poset

Realize balancing as “projection” ~10,000x faster!!

Matrix balancing: Given a nonnegative matrix $P = (p_{ij}) \in \mathbb{R}^{n \times n}$, find $\eta, \theta \in \mathbb{R}^n$ s.t.

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

- $R = \text{diag}(\eta), S = \text{diag}(\theta)$, each entry is given as $p_{ij} = p_{ij}/\theta_i \eta_j$
- Applications: input-output analysis, Hi-C data analysis, the Sudoku puzzle, and Wasserstein metric approximation
- Standard balancing algorithm: Sinkhorn–Knopp algorithm

Matrix balancing is achieved if:

$\eta_1 = 4, \eta_2 = 3, \eta_3 = 2, \eta_4 = 1$

Balancing = projection

Log-Linear on Poset

Realize balancing as “projection” ~10,000x faster!!

Matrix balancing: Given a nonnegative matrix $P = (p_{ij}) \in \mathbb{R}^{n \times n}$, find $\eta, \theta \in \mathbb{R}^n$ s.t.

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

- $R = \text{diag}(\eta), S = \text{diag}(\theta)$, each entry is given as $p_{ij} = p_{ij}/\theta_i \eta_j$
- Applications: input-output analysis, Hi-C data analysis, the Sudoku puzzle, and Wasserstein metric approximation
- Standard balancing algorithm: Sinkhorn–Knopp algorithm

Matrix balancing is achieved if:

$\eta_1 = 4, \eta_2 = 3, \eta_3 = 2, \eta_4 = 1$