Report for CSE 5339 2018 — (OTMLSA)
Optimal Transport in Machine Learning and Shape Analysis

Entropic Regularization of Optimal Transport

Woojin Kim

Abstract

The problem of computing optimal transport distances (OT problems) arises in many different fields . However, the computational cost for an optimal transport distance becomes prohibitive once the size of the support of measures exceeds a few hundred. Referring to [1, 2], this note briefly summarizes the notion of regularization of OT problems and describes *sinkhorn algorithm*, which is for fast computation of optimal transportation distances. The strategy for fast computation is to regularize OT problems by adding an entropic regularization penalty to the original problems.

1 Probability vectors and couplings

Let $m, n \in \mathbf{N}$. For any $m \times n$ matrix M, let M^t be the transpose of M. Also, any vector in $\mathbf{R}^n$ will be understood as an $n \times 1$ matrix. We begin by introducing the notation we will use throughout this note.

Definition 1.1. *Let $m, n \in \mathbf{N}$.*

(i) Let $\sum_n := \{\vec{a} = (a_1, \ldots, a_n)^t \in \mathbf{R}^n : \sum_{i=1}^n a_i = 1\}$. We call any $\vec{a} \in \sum_n$ a probability vector.

(ii) For any $\vec{a} \in \sum_n, \vec{b} \in \sum_m$, let

$$U(\vec{a}, \vec{b}) := \left\{ P = (p_{ij}) \in \mathbf{R}^{n \times m} : \forall i \in \{1, \ldots, n\}, \sum_{j=1}^{m} p_{ij} = a_i \text{ and } \forall j \in \{1, \ldots, m\}, \sum_{i=1}^{n} p_{ij} = b_j \right\}.$$

In words, the set $U(\vec{a}, \vec{b})$ is the set of all couplings between $\vec{a}$ and $\vec{b}$.

We will regard $U(\vec{a}, \vec{b})$ as a subset of the Euclidean space $\mathbf{R}^{n \times m}$.

Remark 1.2 (Geometric properties of $U(\vec{a}, \vec{b})$)**.** *In Definition 1.1 (ii), the subset $U(\vec{a}, \vec{b})$ of the Euclidean space $\mathbf{R}^{n \times m}$ is bounded because for each $P = (p_{ij}) \in U(\vec{a}, \vec{b})$, $0 \leqslant p_{ij} \leqslant 1$. Also, $U(\vec{a}, \vec{b})$ is a convex polytope, which is drawn from the observation that the boundary of $U(\vec{a}, \vec{b})$ is the solution set of a system of linear equations and whenever $P, Q \in U(\vec{a}, \vec{b})$, $(1-t)P + tQ \in U(\vec{a}, \vec{b})$ for any $t \in [0, 1]$.*

2 Kantorvich's OT problem

We review the notion of Kantorvich's OT problem and then discuss the two drawbacks in making use of Kantorvich's OT problem in practice.

Kantorvich's OT problem. Let $n, m \in \mathbf{N}$. Fix certain probability vectors $\vec{a} \in \sum_n$ and $\vec{b} \in \sum_m$. Also, fix any matrix $C = (c_{ij}) \in \mathbf{R}_+^{n\times m}$, which we call a *cost* matrix. Define

$$L_C(\vec{a}, \vec{b}) := \min_{P \in U(\vec{a},\vec{b})} \langle C, P\rangle, \tag{1}$$

where $\langle C, P\rangle = \sum_{i,j} p_{ij}c_{ij}$. The problem of calculating $L_C(\vec{a}, \vec{b})$ is said to be a *Kantorvich's OT problem*. A optimal solution to (1) (i.e. a minimizer $P \in U(\vec{a}, \vec{b})$) always exists. Moreover, equation (1) has multiple optimal solutions in general.

There are two issues that hinder the applicability of Kantorvich's OT problem:

(i) **High computational cost.** The cost of computing $L_C(\vec{a}, \vec{b})$ can run to $O(n^3 \log n)$ when $\vec{a}, \vec{b} \in \sum_n$. This means that computing a single distance between a pair of measures supported by a few hundred points in a metric space can take more than a few seconds on a single CPU.

(ii) **Deficiency of Randomness.** Since the function $P \mapsto \langle C, P\rangle$ defined on $U(\vec{a}, \vec{b})$ is linear, the optimal solutions to (1) are obtained on the boundary of $U(\vec{a}, \vec{b}) \subset \mathbf{R}^{n\times m}$. This implies that any optimal $P \in U(\vec{a}, \vec{b})$ is sparse matrix.[1] From a probabilistic perspective, such coupling P is quasi-deterministic joint probability since if $p_{ij} > 0$ for some i, j, then very few probabilities $p_{ij'}$ for $j \neq j'$ will be non-zero in general. This deficiency of randomness in P results in the failure of traffic pattern prediction: Actual traffic patterns in a network do not agree with those predicted by the solution of the optimal transport problem. Indeed, actual traffic patterns are more diffuse than those obtained from solving the OT problem, which tend to rely on a few routes as a result of the sparsity of optimal couplings to the solution of (1).

3 Entropy of measures (or histograms)

Let $\vec{a} \in \sum_n$ and $\vec{b} \in \sum_m$. For any $P = (p_{ij}) \in U(\vec{a}, \vec{b}) \subset \mathbf{R}_+^{n\times m}$, the *entropy* of P is defined by

$$H(P) := -\sum_{ij} p_{ij} \log p_{ij}.$$

Also, the entropy a probability vector is defined in a similar way. Namely, for any $\vec{a} \in \sum_n$, $H(\vec{a}) := -\sum_i a_i \log a_i$. The entropy measures "unpredictability", so the more uniformly distributed a probability vector $\vec{a} \in \sum_n$ is, the larger its entropy $H(\vec{a})$ is.

Remark 3.1 (About entropy)**.** *Fix $\vec{a} \in \sum_n$ and $\vec{b} \in \sum_m$.*

(i) $\vec{a}\vec{b}^t = (a_ib_j) \in U(\vec{a}, \vec{b})$. Specifically, $\vec{a}\vec{b}^t$ is the joint probability between two independent random variables with the distributions $\vec{a}$ and $\vec{b}$.

(ii) The map $H : U(\vec{a}, \vec{b}) \to \mathbf{R}_+$ is strongly 1-concave*: It is not difficult to check that $\frac{\partial^2}{\partial p_{ij}^2} H(P) = -\frac{1}{p_{ij}}$. Hence the Hessian matrix $\partial^2 H(P)$, which is $((nm) \times (nm))$-matrix, is less than or equal to $-I = (-\delta_{ij})_{1\leqslant i,j\leqslant nm}$ component-wise.*

(iii) For any $P \in U(\vec{a}, \vec{b})$, it holds that $\frac{1}{2}\left(H(\vec{a}) + H(\vec{b})\right) \leqslant H(P) \leqslant H(\vec{a}) + H(\vec{b})$. In particular, when $P = \vec{a}\vec{b}^t$, the second inequality can be replaced by the equality.

[1] When $n = m$, any optimal P can contain only up to $2n - 1$ nonzero components ([1, 3]).

4 Regularized OT problem

We regularize Kantorvich's OT problem (1) by adding the entropic term as follows: Fix any cost matrix $C = (c_{ij}) \in \mathbf{R}_+^{n\times m}$. For $\varepsilon \geqslant 0$, define

$$L_C^{\varepsilon}(\vec{a},\vec{b}) := \min_{P\in U(\vec{a},\vec{b})} \left(\langle C, P\rangle - \varepsilon H(P)\right). \tag{2}$$

The problem defined in equation (2) is said to be *regularized OT problem.* From (2), observe the following: on the RHS, $\langle C, P\rangle$ is linear with respect to P, whereas $-\varepsilon H(P)$ is *strongly convex* with respect to P by Remark 3.1 (ii). Therefore, the map $F_\varepsilon : U(\vec{a},\vec{b}) \to \mathbf{R}$ defined by

$$P \mapsto \langle C, P\rangle - \varepsilon H(P)$$

is strongly convex. Invoking that $U(\vec{a},\vec{b})$ is a convex region in $\mathbf{R}^{n\times m}$, the strong convexity of the map F_ε implies that the minimizer of F_ε exists and is unique in $U(\vec{a},\vec{b})$. Therefore, equation (2) has a unique solution. Also, observe that qualified candidates $P \in U(\vec{a},\vec{b})$ as the minimizers of equation (2) should not only make $\langle C, P\rangle$ small but also $H(P)$ large. In particular, taking into account the entropy $H(P)$ in solving (2) indicates that one can circumvent issue (ii) **Deficiency of Randomness** in Section 2 by solving the regularized OT problem (2) instead of (1).

Remark 4.1. *For the map $F_\varepsilon : U(\vec{a},\vec{b}) \to \mathbf{R}$ above, observe the following.*

(i) As ε increases, couplings $P \in U(\vec{a},\vec{b})$ of large entropy are preferred in minimizing F_ε.

(ii) As ε decreases, couplings $P \in U(\vec{a},\vec{b})$ optimizing the transportation cost $\langle C, P\rangle$ are preferred.

The following proposition precisely describes the behavior of the unique minimizer P_ε of F_ε with respect to ε.

Proposition 4.2 (Convergence with ε [2, Proposition 4.1])**.** *For $\varepsilon > 0$, let $P_\varepsilon \in U(\vec{a},\vec{b})$ be the unique minimizer of the map F_ε.*

(i) As $\varepsilon \to \infty$, P_ε converges to $\vec{a}\vec{b}^t$, which has the maximal entropy amongst the elements in $U(\vec{a},\vec{b})$.

(ii) As $\varepsilon \searrow 0$, P_ε converges to the solution of equation (1) with maximal entropy within the set of all solutions of equation (1).

5 Sinkhorn's matrix scaling algorithm

Given any matrix $K = (k_{ij})$, let $e^K := (e^{k_{ij}})$. Also, given any vector $\vec{u} = (u_1, \ldots, u_n)^t \in \mathbf{R}^n$, let $\mathrm{diag}(\vec{u}) := (u_i\delta_{ij})_{1\leqslant i,j\leqslant n}$. The following proposition is crucial for establishing the sinkhorn algorithm for computing the solution to (2).

Proposition 5.1 ([2, Proposition 4.3])**.** *The unique solution $P_\varepsilon \in U(\vec{a},\vec{b})$ to (2) has the form*

$$P_\varepsilon = \mathrm{diag}(\vec{u}) \cdot K_\varepsilon \cdot \mathrm{diag}(\vec{v}) \tag{3}$$

for two (unknown) vectors $\vec{u} \in \mathbf{R}_+^n, \vec{v} \in \mathbf{R}^m$, where $K_\varepsilon := e^{-\frac{1}{\varepsilon}C}$.

By virtue of Proposition 5.1, in order to solve equation (2), it suffices to find $\vec{u}$ and $\vec{v}$ that solve equation (3). Before introducing the sinkhorn algorithm, we begin by introducing the notation.

For any $n \in \mathbf{N}$, let $\mathbf{1}_n$ be the column vector with n rows consisting solely of 1. Also, given any two vectors $\vec{\alpha}, \vec{\beta} \in \mathbf{R}^n$, let $\vec{\alpha} \odot \vec{\beta}$ be the vector of the same size obtained by component-wise

multiplication. Assuming $\vec{\beta}$ does not contain zero, the vector $\frac{\vec{\alpha}}{\vec{\beta}}$ is defined by component-wise division of $\vec{\alpha}$ by $\vec{\beta}$.

Let $P_\varepsilon \in U(\vec{a}, \vec{b})$ be the unique solution to (2). Since $P_\varepsilon \in U(\vec{a}, \vec{b})$, we have $P_\varepsilon \mathbf{1}_m = \vec{a}$ and $P_\varepsilon^t \mathbf{1}_n = \vec{b}$. Then by equation (3), we have

$$[\mathrm{diag}(\vec{u}) \cdot K \cdot \mathrm{diag}(\vec{v})]\mathbf{1}_m = \vec{a}, \qquad [\mathrm{diag}(\vec{u}) \cdot K \cdot \mathrm{diag}(\vec{v})]^t \mathbf{1}_n = \vec{b}.$$

Observing that $\mathrm{diag}(\vec{v}) \cdot \mathbf{1}_m = \vec{v}$, one can simplify the first equation above into $\vec{u} \odot K\vec{v} = \vec{a}$. Similarly, the second equation above is turned into $\vec{v} \odot K^t\vec{u} = \vec{b}$. Therefore, any vectors $\vec{u}$ and $\vec{v}$ satisfying equation (3) must satisfy the following equation system:

$$\begin{cases} \vec{u} \odot K\vec{v} = \vec{a}, \\ \vec{v} \odot K^t\vec{u} = \vec{b}. \end{cases} \tag{4}$$

Now we introduce the sinkhorn algorithm. Initialize an arbitrary vector $\vec{v}^{(0)} \in \mathbf{R}_+^m$ consisting solely of strictly positive entries. The following recurrence formula defines the Sinkhorn's algorithm: for $l \in \mathbf{N}$,

$$(\vec{u})^{(l+1)} := \frac{\vec{a}}{K\vec{v}^{(l)}}, \qquad (\vec{v})^{l+1} := \frac{\vec{b}}{K^t(\vec{u})^{l+1}}.$$

The sequence of vectors obtained by the iteration process converges to the vectors $\vec{u}$ and $\vec{v}$ that solve equation (3). In other words, as l increases, $P_\varepsilon^{(l)} := \mathrm{diag}(\vec{u}^{(l)}) \cdot K_\varepsilon \cdot \mathrm{diag}(\vec{v}^{(l)})$ converges to the unique solution P_ε of equation (2).

Remark 5.2 (Computational complexity [2, Remark 4.5])**.** *Assume that $n = m \in \mathbf{N}$. Then, in order to find a coupling $\hat{P} \in U(\vec{a}, \vec{b})$ such that $\langle \hat{P}, C\rangle \leqslant L_C(\vec{a}, \vec{b}) + \tau$, one needs $O(n^2 \log(n)\tau^{-3})$ operations.*

We finish this note by summarizing the methods and results of computational experiments carried out in [1].

- They test the performance of the sinkhorn algorithm on the MNIST digits dataset. Each image in the MNIST digits dataset is converted into a vector of intensities on the 20×20 pixel grid, which are then normalized to sum to 1. The cost matrix $C = (c_{ij})$ is the 400×400 matrix, where c_{ij} is the Euclidean distance between the i-th bean and the j-th bean in the pixel grid. It turns out that for small ε and for images $\vec{a}$ and $\vec{b}$ in the dataset, $L_c^\varepsilon(\vec{a}, \vec{b})$ approximates $L_c(\vec{a}, \vec{b})$ with a high accuracy: Let P_ε denote the unique solution to equation (2) and let P_* denote *any* solution to (1). According the their experiments,

$$\frac{\langle C, P_\varepsilon\rangle - \langle C, P_*\rangle}{\langle C, P_*\rangle} = \begin{cases} 3.4\%, & \varepsilon = 0.02 \\ 1.2\%, & \varepsilon = 0.01, \end{cases}$$

  in terms of the median over 40^2 pairs of images from the MNIST database. P_ε above is actually $P_\varepsilon^{(l)} = \mathrm{diag}(\vec{u}^{(l)}) \cdot K_\varepsilon \cdot \mathrm{diag}(\vec{v}^{(l)})$ where l is the minimum value such that

$$\left| \frac{\langle C, P_\varepsilon^{(l)}\rangle}{\langle C, P_\varepsilon^{(l-1)}\rangle} - 1 \right| \leqslant 10^{-4}.$$

- They also compare the computational speed of calculating (1) and (2) with random cost matrices. Computing (2) is much more faster in general. One example is the following: Let $\varepsilon = 0.02$. For some probability vectors $\vec{a}$ and $\vec{b}$ with 2048 bins, computing $L_C(\vec{a}, \vec{b})$ takes about 100 seconds, whereas 1 second is enough for obtaining $L_C^\varepsilon(\vec{a}, \vec{b})$.

- As ε decreases down to 0, more iterations are necessary. However, the necessary number of iterations is not much affected by the dimension of the probability vectors $\vec{a}, \vec{b}$ that are compared. This makes the difference of the cost computing $L_C(\vec{a}, \vec{b})$ and $L_C^{\varepsilon}(\vec{a}, \vec{b})$ stark as the dimensions of $\vec{a}, \vec{b}$ increase.

References

[1] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transportation distances. `https://arxiv.org/pdf/1306.0895.pdf`.

[2] Gabriel Peyré, Marco Cuturi. Computational optimal transport. `https://arxiv.org/pdf/1803.00567.pdf`.

[3] Richard A. Brualdi. Combinatorial matrix classes, volume 108. Cambridge University Press, 2006.