Robust computation of optimal transport by β-potential regularization

Shintaro Nakamura NAKAMURASHINTARO@G.ECC.U-TOKYO.AC.JP
The University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa City, Chiba 277-8561

Han Bao BAO@I.KYOTO-U.AC.JP
Kyoto University, Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501

Masashi Sugiyama SUGI@K.U-TOKYO.AC.JP
RIKEN AIP center, Nihonbashi 1-chome Mitsui Building, 15th floor, 1-4-1 Nihonbashi, Chuo-ku, Tokyo 103-0027
The University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa City, Chiba 277-8561

Editors: Emtiyaz Khan and Mehmet Gonen

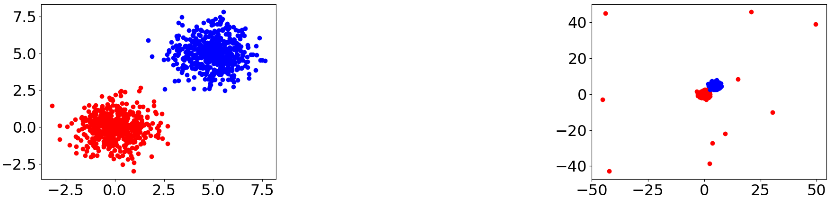
Abstract

Optimal transport (OT) has become a widely used tool in the machine learning field to measure the discrepancy between probability distributions. For instance, OT is a popular loss function that quantifies the discrepancy between an empirical distribution and a parametric model. Recently, an entropic penalty term and the celebrated Sinkhorn algorithm have been commonly used to approximate the original OT in a computationally efficient way. However, since the Sinkhorn algorithm runs a projection associated with the Kullback-Leibler divergence, it is often vulnerable to outliers. To overcome this problem, we propose regularizing OT with the β-potential term associated with the so-called β-divergence, which was developed in robust statistics. Our theoretical analysis reveals that the β-potential can prevent the mass from being transported to outliers. We experimentally demonstrate that the transport matrix computed with our algorithm helps estimate a probability distribution robustly even in the presence of outliers. In addition, our proposed method can successfully detect outliers from a contaminated dataset.

Keywords: Optimal transport; Robustness

1. Introduction

Many machine learning problems such as density estimation and generative modeling are often formulated by a discrepancy between probability distributions (Kanamori et al., 2009; Goodfellow et al., 2014). As a common choice, the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951) has been widely used since minimizing the KL-divergence of an empirical distribution from a parametric model corresponds to maximum likelihood estimation. However, the KL-divergence suffers from some problems. For instance, the KL-divergence of p from q is not well-defined when the support of p is not completely included in the support of q. Moreover, the KL-divergence does not satisfy the axioms of metrics in a probability space. On the other hand, *optimal transport* (OT) (Villani, 2008) does not suffer from these problems. OT does not require any conditions on the support of probability distributions and thus is expected to be more stable than the KL-divergence. Therefore, the divergence



(a) Sets of samples without outliers

(b) Sets of samples with outliers

Figure 1: (a) 500 samples (red) are drawn from $\mathcal{N}([0,0]^\top, I)$ and 500 samples (blue) are from $\mathcal{N}([5,5]^\top, I)$. I is the two-dimensional identity matrix. (b) 10 samples from two-dimensional uniform distribution $\mathrm{U}\{(x,y)| -50 \leq x, y \leq 50\}$ are added to the red samples.

Table 1: The ouput value of the Sinkhorn algorithm and our algorithm. The exact OT value is 50.13 for the sets of samples in Figure (1a).

	Figure 1a	Figure 1b
The Sinkhorn algorithm	50.74	92.19
Our algorithm	50.10	50.00

estimator is less prone to diverge to infinity. In addition, OT between two distributions is a metric in a probability space and therefore defines a proper distance between histograms and probability measures (Peyré and Cuturi, 2019). Owing to these nice properties, OT has been celebrated with many applications such as image processing (Rabin et al., 2012) and color modifications (Solomon et al., 2015).

However, the ordinary OT suffers from heavy computation. To cope with this problem, one of the common approaches is to regularize the ordinary OT problem with an entropic penalty term (Boltzmann–Shannon entropy (Dessein et al., 2018)) and use the Sinkhorn algorithm (Knopp and Sinkhorn, 1967) to approximate OT (Cuturi, 2013). The entropic penalty makes the objective strictly convex, ensuring the existence of the unique global optimal solution, and the Sinkhorn algorithm projects this global optimal solution onto a set of couplings in terms of the KL-divergence, a divergence associated with the Boltzmann–Shannon entropy (Dessein et al., 2018). Unfortunately, the KL projection in statistical estimation is often not robust in the presence of outliers (Basu et al., 1998). In our pilot study, we experimentally confirmed that the Sinkhorn algorithm is easily affected by outliers (Figure 1). As can be seen in Table 1, the output value of the Sinkhorn algorithm drastically increases even when only a small number of outliers are included in the dataset.

The high sensitivity of the Sinkhorn algorithm may lead to undesired solutions in probabilistic modeling when we deal with noisy and adversarial datasets (Kos et al., 2018). Several existing works have tackled this challenge. Staerman et al.(2021) proposed a median-of-means estimator of the 1-Wasserstein dual to suppress outlier sensitivity. However, the obtained solution is hard to be interpreted as an approximation to OT because the corresponding primal problem is unclear. On the other hand, the following works robustly approximate OT by sending only a small probability mass to outliers, allowing some violation of the coupling constraint: Balaji et al.(2020) used unbalanced OT (Chizat, 2017)

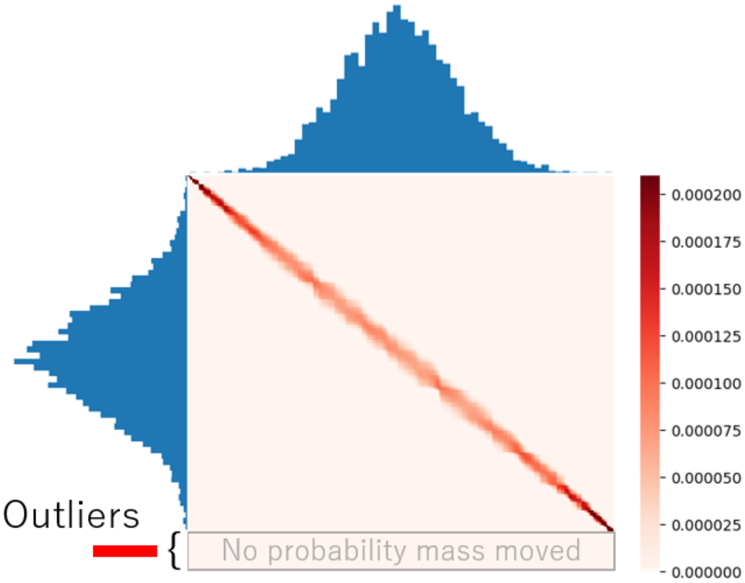


Figure 2: The heatmap of the transport matrix computed with our algorithm. The horizontal histogram is a set of 500 samples from a 1-dimensional standard normal distribution. The vertical histogram is a set of 495 samples from a 1-dimensional standard normal distribution added with 5 outliers with a value of 70. As can be seen from the heatmap, no mass was transported from outliers included in the source histogram.

with the χ^2-divergence as their f-divergence penalty on marginal violation to compute OT robustly. This formulation requires access to the outlier proportion, which is usually not available. Moreover, in Section 4.2, we show their method relying on the optimization package CVXPY (Diamond and Boyd, 2016) does not scale to large samples. Mukherjee et al.(2021) mainly focused on outlier detection by truncating the distance matrix in OT. As a downside, one needs to set an appropriate threshold to use their method, which is hardly known in advance. Hence, we still lack a robust OT formulation independent of sensitive hyperparameters, with easily-accessible primal transport matrices.

In this work, we propose to mitigate the outlier sensitivity of the Sinkhorn algorithm by regularizing OT with the β-potential term instead of the Boltzmann–Shannon entropy. This formulation can be regarded as a projection based on the β-divergence (Basu et al., 1998; Futami et al., 2018). With some computational tricks, our algorithm is guaranteed not to move any probability mass to outliers (Figure 2). It also suggests that our algorithm computes an approximate OT between the inliers. The approximate OT computed by our method was 50.10 and 50.00 in the settings of Figures 1a and 1b, respectively (Table 1), meaning that our method is less prone to be affected by outliers. Through numerical experiments, we demonstrate that our proposed method can measure a distance between datasets more robustly than the Sinkhorn algorithm. As a practical application, we show our proposed method can be applied to an outlier detection task.

2. Background

In this section, we first show the formulation of ordinary discrete OT. Subsequently, we review the Bregman divergence. Finally, we introduce the convex regularized discrete OT (CROT) formulation and the alternate Bregman projection to obtain the solution to the CROT.

2.1. Optimal Transport (OT)

We introduce OT in a discrete setting. In this case, OT can be regarded as the cheapest plan to deliver items from m suppliers to n consumers, where each supplier and consumer has supply $\frac{1}{m}$ and demand $\frac{1}{n}$, respectively. In this work, we mainly focus on measuring the transportation cost between two probability distributions. Suppose we have two sets of independent samples $\{\boldsymbol{x}_i\}_{i=1}^m$ and $\{\boldsymbol{y}_j\}_{j=1}^n$ drawn from two distributions P_x and P_y, respectively. We write the corresponding empirical measures by $\hat{P}_x \coloneqq \frac{1}{m}\sum_{i=1}^m \boldsymbol{x}_i \delta_{\boldsymbol{x}_i}$ and $\hat{P}_y \coloneqq \frac{1}{n}\sum_{i=1}^n \boldsymbol{y}_i \delta_{\boldsymbol{y}_i}$, where $\delta_{\boldsymbol{x}}$ is the delta function at position $\boldsymbol{x}$. Let $\boldsymbol{\gamma} \in \mathbb{R}_+^{m\times n}$ be the distance matrix, where γ_{ij} denotes the distance between $\boldsymbol{x}_i$ and $\boldsymbol{y}_j$. Transport matrices are confined to

$$\left\{ \boldsymbol{\Pi} \in \mathbb{R}_+^{m\times n} \,\middle|\, \boldsymbol{\Pi}\mathbf{1}_n = \frac{\mathbf{1}_m}{m}, \boldsymbol{\Pi}^\top \mathbf{1}_m = \frac{\mathbf{1}_n}{n} \right\} =: \mathcal{G}\left(\frac{\mathbf{1}_m}{m}, \frac{\mathbf{1}_n}{n}\right), \tag{1}$$

where $\mathbb{R}_+^{m\times n}$ is the set of non-negative reals. We call $\mathcal{G}(\frac{\mathbf{1}_m}{m}, \frac{\mathbf{1}_n}{n})$ the *coupling constraint*. In order to keep the notation concise, the Frobenius inner product between two matrices $\boldsymbol{\pi}, \boldsymbol{\gamma} \in \mathbb{R}_+^{m\times n}$ is denoted by

$$\langle \boldsymbol{\pi}, \boldsymbol{\gamma} \rangle \coloneqq \sum_{i,j} \pi_{ij}\gamma_{ij}. \tag{2}$$

Then, OT between the two empirical distributions $\hat{P}_x$ and $\hat{P}_y$ is defined as follows (Peyré and Cuturi, 2019):

$$\mathrm{OT}(\hat{P}_x \| \hat{P}_y) \quad \coloneqq \quad \min_{\boldsymbol{\pi}\in\mathcal{G}(\frac{\mathbf{1}_m}{m}, \frac{\mathbf{1}_n}{n})} \langle \boldsymbol{\pi}, \boldsymbol{\gamma} \rangle. \tag{3}$$

2.2. Bregman divergence

Let $\mathcal{E}$ be a Euclidean space with inner product $\langle \cdot, \cdot \rangle$ and induced norm $\|\cdot\|$. Let $\phi : \mathcal{E} \to \mathbb{R}$ be a strictly convex function on $\mathcal{E}$ that is differentiable on $\mathrm{int}(\mathrm{dom}\,\phi) \neq \emptyset$. The *Bregman divergence* generated by ϕ is defined as follows:

$$B_\phi(\boldsymbol{x}\|\boldsymbol{y}) \coloneqq \phi(\boldsymbol{x}) - \phi(\boldsymbol{y}) - \langle \boldsymbol{x} - \boldsymbol{y}, \nabla\phi(\boldsymbol{y}) \rangle, \tag{4}$$

for all $\boldsymbol{x} \in \mathrm{dom}\,\phi$ and $\boldsymbol{y} \in \mathrm{dom}\,\phi$. In this paper, for the sake of simplicity, we consider the so-called *separable* Bregman divergences (Dessein et al., 2018) over the set of transport matrices $\mathcal{G}\left(\frac{\mathbf{1}_m}{m}, \frac{\mathbf{1}_n}{n}\right)$, which can be decomposed as the element-wise summation:

$$B_\phi(\boldsymbol{\pi}\|\boldsymbol{\xi}) \quad = \quad \sum_{i=1}^m \sum_{j=1}^n B_\phi(\pi_{ij}\|\xi_{ij}), \tag{5}$$

$$\phi(\boldsymbol{\pi}) \quad = \quad \sum_{i=1}^m \sum_{j=1}^n \phi(\pi_{ij}), \tag{6}$$

Regularization term	dom ϕ	dom ψ
β-potential ($\beta > 1$)	$\mathbb{R}_+$	$(\frac{1}{1-\beta}, \infty)$
Boltzman-Shannon entropy	$\mathbb{R}_+$	$\mathbb{R}$

Table 2: Domains of each regularizer and its Fenchel conjugate.

where we used $\phi : \mathbb{R} \to \mathbb{R}$ to denote the generator function same across all elements, with a slight abuse of notation. Suppose now that ϕ is of the Legendre type (Bauschke and Borwein, 1997), and let $\mathcal{C} \subseteq \mathcal{E}$ be a closed convex set such that $\mathcal{C} \cap \operatorname{int}(\operatorname{dom} \phi) \neq \emptyset$. Then, for any point $\boldsymbol{y} \in \operatorname{int}(\operatorname{dom} \phi)$, the following problem,

$$T_{\mathcal{C}}(\boldsymbol{y}) = \operatorname*{argmin}_{\boldsymbol{x} \in \mathcal{C}} B_\phi(\boldsymbol{x} \| \boldsymbol{y}), \tag{7}$$

has a unique solution. $T_{\mathcal{C}}(\boldsymbol{y})$ is called the *Bregman* projection of $\boldsymbol{y}$ onto $\mathcal{C}$ (Dessein et al., 2018).

2.3. Formulation of CROT

Here, we give the formulation of the CROT and show that obtaining the optimal solution of the CROT corresponds to minimizing the Bregman divergence between two matrices.

The CROT is formulated as a regularized version of (3) by ϕ as follows:

$$L_\phi(\boldsymbol{\pi}) := \min_{\boldsymbol{\pi} \in \mathcal{G}(\frac{\mathbf{1}m}{m}, \frac{\mathbf{1}n}{n})} \langle \boldsymbol{\pi}, \boldsymbol{\gamma} \rangle + \lambda \phi(\boldsymbol{\pi}), \tag{8}$$

where $\lambda > 0$ is a regularization parameter. Subsequently, we often work on the dual variable of $\boldsymbol{\pi}$. The dual variable $\boldsymbol{\theta}$ satisfies the following conditions:[1]

$$\boldsymbol{\pi} = \nabla \psi(\boldsymbol{\theta}), \tag{9}$$
$$\boldsymbol{\theta} = \nabla \phi(\boldsymbol{\pi}), \tag{10}$$

where ψ is the Fenchel conjugate of ϕ (Dessein et al., 2018). The optimal solution of (8) can be understood via the Bregman projection. Let us consider the unconstrained version of (8):

$$\min_{\boldsymbol{\pi} \in \mathbb{R}^{m \times n}} \langle \boldsymbol{\pi}, \boldsymbol{\gamma} \rangle + \lambda \phi(\boldsymbol{\pi}). \tag{11}$$

Since $\langle \boldsymbol{\pi}, \boldsymbol{\gamma} \rangle$ is linear and ϕ is strictly convex with respect to $\boldsymbol{\pi}$, there is a unique optimal solution $\boldsymbol{\xi}$ for (11):

$$\boldsymbol{\xi} = \nabla \psi(-\boldsymbol{\gamma}/\lambda), \tag{12}$$

which can be obtained by solving the first-order optimality condition of (11) with the dual relationship $(\nabla \phi)^{-1} = \nabla \psi$:

$$\boldsymbol{\gamma} + \lambda \nabla \phi(\boldsymbol{\xi}) = 0. \tag{13}$$

[1] This mapping based on gradients (not subgradients) is legitimate only when ϕ is of the Legendre type.

Then,

$$\boldsymbol{\pi}^*_\lambda \coloneqq \operatorname*{argmin}_{\boldsymbol{\pi}\in\mathcal{G}(\frac{\mathbf{1}_m}{m},\frac{\mathbf{1}_n}{n})} L_\phi(\boldsymbol{\pi}) \tag{14}$$

$$= \operatorname*{argmin}_{\boldsymbol{\pi}\in\mathcal{G}(\frac{\mathbf{1}_m}{m},\frac{\mathbf{1}_n}{n})} B_\phi(\boldsymbol{\pi}\|\boldsymbol{\xi}), \tag{15}$$

where the last equality is due to the following equation:

$$\langle \boldsymbol{\pi}, \boldsymbol{\gamma}\rangle + \lambda\phi(\boldsymbol{\pi}) - \lambda\phi(\boldsymbol{\xi}) - \langle \boldsymbol{\xi}, \boldsymbol{\gamma}\rangle = \lambda B_\phi(\boldsymbol{\pi}\|\boldsymbol{\xi}). \tag{16}$$

Therefore, the solution of (8) can be interpreted as the Bregman projection of the unconstrained solution $\boldsymbol{\xi}$ onto $\mathcal{G}(\frac{\mathbf{1}_m}{m},\frac{\mathbf{1}_n}{n})$. The Sinkhorn algorithm can be used to obtain a solution to OT regularized with the negative of Boltzmann–Shannon entropy $\phi(\pi) = \pi\log\pi - \pi + 1$ (Table 2), and runs a projection associated with the KL-divergence where $B_\phi(\pi\|\xi) = \pi\log\frac{\pi}{\xi} - \pi + \xi$.

2.4. Alternate Bregman projection

Here, we demonstrate how the Bregman projection onto $\mathcal{G}(\frac{\mathbf{1}_m}{m},\frac{\mathbf{1}_n}{n})$ is executed based on Dessein et al.[2018].

Let $\mathcal{C}_0, \mathcal{C}_1, \mathcal{C}_2$ be the following convex sets:

$$\mathcal{C}_0 = \mathbb{R}^{m\times n}_+, \tag{17}$$

$$\mathcal{C}_1 = \left\{\boldsymbol{\pi}\in\mathbb{R}^{m\times n} \,\middle|\, \boldsymbol{\pi}\mathbf{1}_n = \tfrac{\mathbf{1}_m}{m}\right\}, \tag{18}$$

$$\mathcal{C}_2 = \left\{\boldsymbol{\pi}\in\mathbb{R}^{m\times n} \,\middle|\, \boldsymbol{\pi}^\top\mathbf{1}_m = \tfrac{\mathbf{1}_n}{n}\right\}. \tag{19}$$

Then, $\mathcal{G}(\frac{\mathbf{1}_m}{m},\frac{\mathbf{1}_n}{n})$ can be written as follows:

$$\mathcal{G}(\tfrac{\mathbf{1}_m}{m},\tfrac{\mathbf{1}_n}{n}) = \mathcal{C}_0\cap\mathcal{C}_1\cap\mathcal{C}_2. \tag{20}$$

We can get the Bregman projection onto $\mathcal{G}(\frac{\mathbf{1}_m}{m},\frac{\mathbf{1}_n}{n})$ by alternately performing projections onto $\mathcal{C}_0$, $\mathcal{C}_1$, and $\mathcal{C}_2$.

Next, let us consider the projection of a given matrix $\overline{\boldsymbol{\pi}} \in \operatorname{int}(\operatorname{dom}\phi)$ onto $\mathcal{C}_0$, $\mathcal{C}_1$, and $\mathcal{C}_2$. The corresponding projection onto each set is denoted by $\boldsymbol{\pi}^*_0$, $\boldsymbol{\pi}^*_1$, and $\boldsymbol{\pi}^*_2$, respectively. Subsequently, we show how to obtain them (Dessein et al., 2018) (see Section A in the supplementary file for details).

2.4.1. Projection onto $\mathcal{C}_0$

When considering the separable Bregman divergence, the projection onto $\mathcal{C}_0$ can be performed with a closed-form expression in terms of primal parameters:

$$\pi^*_{0,ij} = \max\{0, \overline{\pi}_{ij}\}, \tag{21}$$

where, $\pi^*_{0,ij}$ is the (i,j)-element of matrix $\boldsymbol{\pi}^*_0$. Since ϕ' is increasing, this is equivalently expressed in terms of the dual parameters of $\boldsymbol{\pi}^*_0$, $\boldsymbol{\theta}^*_0$, as

$$\theta^*_{0,ij} = \max\{\phi'(0), \overline{\theta}_{ij}\}. \tag{22}$$

Here, the dual coordinate of the input matrix $\overline{\boldsymbol{\pi}}$ is denoted by $\overline{\boldsymbol{\theta}} = \nabla\psi(\overline{\boldsymbol{\pi}})$.

Algorithm 1 Non-negative alternate scaling algorithm for β-divergence when $\beta > 1$

```
$\tilde{\boldsymbol{\theta}} \leftarrow -\boldsymbol{\gamma}/\lambda$
$\boldsymbol{\theta^*} \leftarrow \max\{\nabla\phi(\mathbf{0}_{m\times n}), \tilde{\boldsymbol{\theta}}\}$
for $t = 1, 2, \ldots, T$ do
    $\boldsymbol{\tau} = \frac{\nabla\psi(\boldsymbol{\theta^*})\mathbf{1}_n - \frac{\mathbf{1}_m}{m}}{\nabla^2\psi(\boldsymbol{\theta}^*)\mathbf{1}_n}$
    $\boldsymbol{\tau} \leftarrow \max(\boldsymbol{\tau},\ \hat{\boldsymbol{\theta}}^* - \nabla\phi(\frac{\mathbf{1}_m}{m}))$
    $\tilde{\boldsymbol{\theta}} \leftarrow \tilde{\boldsymbol{\theta}} - \boldsymbol{\tau}\mathbf{1}_n{}^\top$
    $\boldsymbol{\theta^*} \leftarrow \max\{\nabla\phi(\mathbf{0}), \tilde{\boldsymbol{\theta}}\}$
    $\boldsymbol{\sigma} = \frac{\mathbf{1}_m{}^\top\nabla\psi(\boldsymbol{\theta^*}) - (\frac{\mathbf{1}_n}{n})^\top}{\mathbf{1}_m{}^\top\nabla^2\psi(\boldsymbol{\theta}^*)}$
    $\boldsymbol{\sigma} \leftarrow \max(\boldsymbol{\sigma}, \hat{\boldsymbol{\theta}}^* - \nabla\phi(\frac{\mathbf{1}_n}{n}))$
    $\tilde{\boldsymbol{\theta}} \leftarrow \tilde{\boldsymbol{\theta}} - \mathbf{1}_m\boldsymbol{\sigma}$
    $\boldsymbol{\theta^*} \leftarrow \max\{\nabla\phi(\mathbf{0}_{m\times n}), \tilde{\boldsymbol{\theta}}\}$
end for
$\boldsymbol{\pi^*} \leftarrow \boldsymbol{\nabla\psi(\theta^*)}$
```

2.4.2. Projections onto $\mathcal{C}_1$ and $\mathcal{C}_2$

Next, we consider the Bregman projection onto $\mathcal{C}_1$. The projection onto $\mathcal{C}_2$ can be executed in the same way and thus omitted here. The Lagrangian associated to the Bregman projection $\boldsymbol{\pi}_1^*$ of a given matrix $\overline{\boldsymbol{\pi}} \in \mathrm{int}(\mathrm{dom}\,\phi)$ onto $\mathcal{C}_1$ is given as follows:

$$\mathcal{L}_1(\boldsymbol{\pi}, \boldsymbol{\mu}) = \phi(\boldsymbol{\pi}) - \langle\boldsymbol{\pi}, \nabla\phi(\overline{\boldsymbol{\pi}})\rangle + \boldsymbol{\mu}^\top(\boldsymbol{\pi}\mathbf{1}_n - \tfrac{\mathbf{1}_m}{m}),$$

where $\boldsymbol{\mu} \in \mathbb{R}^m$ are Lagrange multipliers. Their gradients are given on $\mathrm{int}(\mathrm{dom}\,\phi)$ by

$$\nabla_{\boldsymbol{\pi}}\mathcal{L}_1(\boldsymbol{\pi}, \boldsymbol{\mu}) = \nabla_{\boldsymbol{\pi}}\phi(\boldsymbol{\pi}) - \nabla_{\boldsymbol{\pi}}\phi(\overline{\boldsymbol{\pi}}) + \boldsymbol{\mu}\mathbf{1}_n{}^\top, \tag{23}$$

and by noting $(\nabla\phi)^{-1} = \nabla\psi$, $\nabla_{\boldsymbol{\pi}}\mathcal{L}_1(\boldsymbol{\pi}_1, \boldsymbol{\mu}) = \mathbf{0}_{m\times n}$ if and only if (Dessein et al., 2018),

$$\boldsymbol{\pi}_1^* = \nabla\psi(\nabla\phi(\overline{\boldsymbol{\pi}}) - \boldsymbol{\mu}\mathbf{1}_n{}^\top). \tag{24}$$

By multiplying $\mathbf{1}_n$ on the both sides of (24), the following equation system is obtained:

$$\nabla\psi(\nabla\phi(\overline{\boldsymbol{\pi}}) - \boldsymbol{\mu}\mathbf{1}_n{}^\top)\mathbf{1}_n = \tfrac{\mathbf{1}_m}{m}. \tag{25}$$

Due to the separability, the projection onto $\mathcal{C}_1$ can be divided into m subproblems in each coordinate of the dual variable as follows:

$$\sum_{j=1}^{n} \psi'(\overline{\theta}_{ij} - \mu_i) = \frac{1}{m}. \tag{26}$$

To solve equation (26) with respect to μ_i, we use the Newton–Raphson method (Akram and ul Ann, 2015).

3. Outlier-robust CROT

In this section, we first formalize a model of outliers. To make the CROT robust against outliers under the model, we propose the CROT with the β-potential ($\beta > 1$) and introduce how to compute the CROT with the β-potential. Finally, we show its theoretical properties.

3.1. Definition of outliers

In this paper, outliers are formally defined as follows. Suppose we have two datasets $\{\boldsymbol{x}_i\}_{i=1}^m$ and $\{\boldsymbol{y}_j\}_{j=1}^n$. We assume $\{\boldsymbol{x}_i\}_{i=1}^m$ are samples from a clean distribution, while $\{\boldsymbol{y}_j\}_{j=1}^n$ are samples that are contaminated by outliers. Let $\boldsymbol{\gamma}$ be the distance matrix.

Definition 1. *For $z > 0$, the indices of outliers J are defined as follows:*

$$\forall j \in J,\ \forall i \in \{1, \dots, m\}, \gamma_{ij} \geq z. \tag{27}$$

This means that any point in $\{\boldsymbol{y}_j\}_{j=1}^n$ that is more than or equal to z away from any point in $\{\boldsymbol{x}_i\}_{i=1}^m$ is considered as outliers.

3.2. β-potential regularization

We use the β-potential

$$\phi(\pi) = \frac{1}{\beta(\beta-1)}(\pi^\beta - \beta\pi + \beta - 1), \tag{28}$$

associated with the β-divergence,

$$B_\phi(\pi\|\xi) = \frac{1}{\beta(\beta-1)}(\pi^\beta + (\beta-1)\xi^\beta - \beta\pi\xi^{\beta-1}), \tag{29}$$

to robustify the CROT, where $\beta > 1$. The domains of primal ϕ and its Fenchel conjugate ψ are shown in Table 2.

Our proposed algorithm is shown in Algorithm 1. The dual coordinate of the unconstrained CROT solution is denoted by $\tilde{\boldsymbol{\theta}} = \nabla\phi(\boldsymbol{\xi})$. We execute the projections in the cyclic order of $\mathcal{C}_0 \to \mathcal{C}_1 \to \mathcal{C}_0 \to \mathcal{C}_2 \to \mathcal{C}_0 \to \mathcal{C}_1 \to \mathcal{C}_0 \to \mathcal{C}_2 \to \cdots$.

Lines 2, 7, and 11 in Algorithm 1 enforce the dual constraint $\theta^*_{ij} \geq \frac{1}{1-\beta}$ corresponding to $\operatorname{dom}\psi = (\frac{1}{1-\beta}, \infty)$ (Table 2). Lines 4–6 correspond to the projection onto $\mathcal{C}_1$ implemented on the dual coordinate. Since the dual variable must satisfy $\theta^*_{ij} \geq \frac{1}{1-\beta}$ due to $\operatorname{dom}\psi = (\frac{1}{1-\beta}, \infty)$, we update the dual variable only once in the Newton–Raphson method (line 4) since $\theta^*_{ij} \geq \frac{1}{1-\beta}$ is no longer guaranteed after the first update. Similarily, the projection onto $\mathcal{C}_2$ is shown in lines 8–10.

The procedure in line 5 is based on Section 4.6 in Dessein et al.(2018) accelerating the convergence of Algorithm 1 by truncating the optimization variable $\boldsymbol{\tau}$, which we describe subsequently. Recall that, for any i, we have the following condition,

$$\forall j,\ 0 \leq \pi^*_{1,ij} \leq \tfrac{1}{m}, \tag{30}$$

implicitly from the coupling constraint (1). Since naively updating Newton–Raphson method can "overshoot", we truncate $\boldsymbol{\tau}$ so that (30) is satisfied after each update. Below, we show this condition is satisfied mathematically. Let $\hat{\boldsymbol{\theta}}^*$ be the m-dimensional vector whose ith element is the largest value in the ith row of $\boldsymbol{\theta}^*$ defined as follows:

$$\hat{\theta}_i^* \quad := \quad \max\{\theta_{ij}^*\}_{1\leq j\leq n}. \tag{31}$$

Since ϕ is convex,

$$\begin{aligned} & 0 \leq \pi_{1,ij}^* \leq \tfrac{1}{m} \\ \iff \quad & \phi'(0) \leq \phi'(\pi_{1,ij}^*) = \theta_{1,ij}^* \leq \phi'(\tfrac{1}{m}) \end{aligned} \tag{32}$$

holds. Hence, for every i, if we lower-bound τ_i, the Newton–Raphson decrement for the ith row of $\boldsymbol{\theta}^*$ as

$$\tau_i \quad \leftarrow \quad \max\{\tau_i, \hat{\theta}_i^* - \phi'\left(\tfrac{1}{m}\right)\}, \tag{33}$$

then, for any j,

$$\begin{aligned} \tilde{\theta}_{ij} - \tau_i \quad & \leq \quad \theta_{ij}^* - \tau_i & (34) \\ & \leq \quad \hat{\theta}_i^* - \tau_i & (35) \\ & \leq \quad \phi'\left(\tfrac{1}{m}\right). & (36) \end{aligned}$$

This means that every element in the ith row of $\tilde{\boldsymbol{\theta}}$ computed in line 6 in Algorithm 1 is no larger than $\phi'(\frac{1}{m})$. After line 7, $\boldsymbol{\theta}^*$ satisfies the condition (30). Similarly, we force $\pi_{2,ij}$ to satisfy the following conditions:

$$\forall i,\ 0 \leq \pi_{2,ij} \leq \tfrac{1}{n}. \tag{37}$$

After line 11, this condition is satisfied.

3.3. Theoretical analysis

In the presence of outliers, we expect to approximate the OT by preventing mass transport to outliers. This property is formalized below.

Definition 2. *Suppose $\boldsymbol{\pi} \in \mathbb{R}_+^{m\times n}$ and a set of indices $O \subseteq \{1, \ldots, n\}$ satisfies the following condition:*

$$\forall i, \pi_{ij} = 0 \quad \text{if} \quad j \in O. \tag{38}$$

Then, we say $\boldsymbol{\pi}$ transports no mass to O.

Although we do not expect to transport any mass to outliers, the optimal solution of the CROT must satisfy the coupling constraint and then the condition (38) is never satisfied. To ensure (38), we consider solving the CROT with only a finite number of updates subsequently. Then, an intermediate solution can satisfy (38), although the coupling constraint is not satisfied. This is in stark contrast to the previous works (Chizat, 2017; Balaji et al., 2020), which cannot avoid transporting some mass to outliers.

The following proposition provides sufficient conditions on the number of iterations T to ensure the condition (38). Refer to Section B in the supplementary file for the proof.

Proposition 1. *For a given z $(> \frac{\lambda}{\beta-1})$, let $J \subseteq \{1, \ldots, n\}$ be a subset of indices which satisfies the condition shown in Definition 1. Suppose we obtained a transport matrix $\boldsymbol{\pi}^{\text{output}}$ by running the alogrithm T times satisfying the following condition:*

$$T < \frac{\frac{z}{\lambda}(\beta - 1) - 1}{(\frac{1}{m})^{\beta-1} + (\frac{1}{n})^{\beta-1}}. \tag{39}$$

Then, $\boldsymbol{\pi}^{\text{output}}$ transports no mass to J.

Here, $z > \frac{\lambda}{\beta-1}$ is necessary so that T is upper-bounded by a positive number. Intuitively, this means that the transport matrix obtained by Algorithm 1 disregards points distant from inliers more than or equal to z. Note that the condition (39) tells us that a sufficiently small number of iterations T leads to an approximate CROT solution that does not transport any mass to outliers.

We discuss the selection of hyperparameters β and λ in Sections 4.3.

4. Experiments

Here, we show two applications of our method to demonstrate the practical effectiveness. In both of the experiments in Sections 4.1 and 4.2, we set the hyperparameters in the proposed method as $\beta = 1.2$ and $\lambda = 2$. We discuss the selection of these hyperparameters in Section 4.3.

4.1. Measuring distance between datasets

In the first experiments, we numerically confirm that our method can compute the distance more robustly than the Sinkhorn algorithm. We used the following benchmark datasets: MNIST (Deng, 2012), FashionMNIST (Xiao et al., 2017), KMNIST (Clanuwat et al., 2018), and EMNIST(Letters)(Cohen et al., 2017). From each benchmark dataset, we randomly sampled 10000 data points, and split them into two subsets, each containing 5000 data. We regarded these data points as inliers. Then a portion of one subset was replaced by data from another benchmark dataset which were regarded as outliers. We computed CROT and outlier-robust CROT between these two subsets, and investigated how they changed when the outlier ratio are 5%, 10 %, 15%, 20%, 25% and 30%. We simply used the raw data to compute the distance matrix $\gamma_{ij} = \|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2$, i.e., the Euclidean distance between raw data, and used their median value as the threshold z. In this way, we expect that outliers are distinguished from inliers. The results are shown in Figure 3. Although the distance computed by the Sinkhorn algorithm drastically changes when the outlier ratio gets larger, the degree of change in the output values of our algorithm is milder in every dataset. Therefore, we can see that our algorithm computes the distance between datasets more stably than the Sinkhorn algorithm.

4.2. Applications to outlier detection

Our algorithm enables us to detect outliers. Let μ_m be a clean dataset and ν_n be a dataset which is polluted with outliers. We regard the jth data point in ν_n is an outlier if Algorithm 1 outputs a transport matrix whose jth column is all zeros.

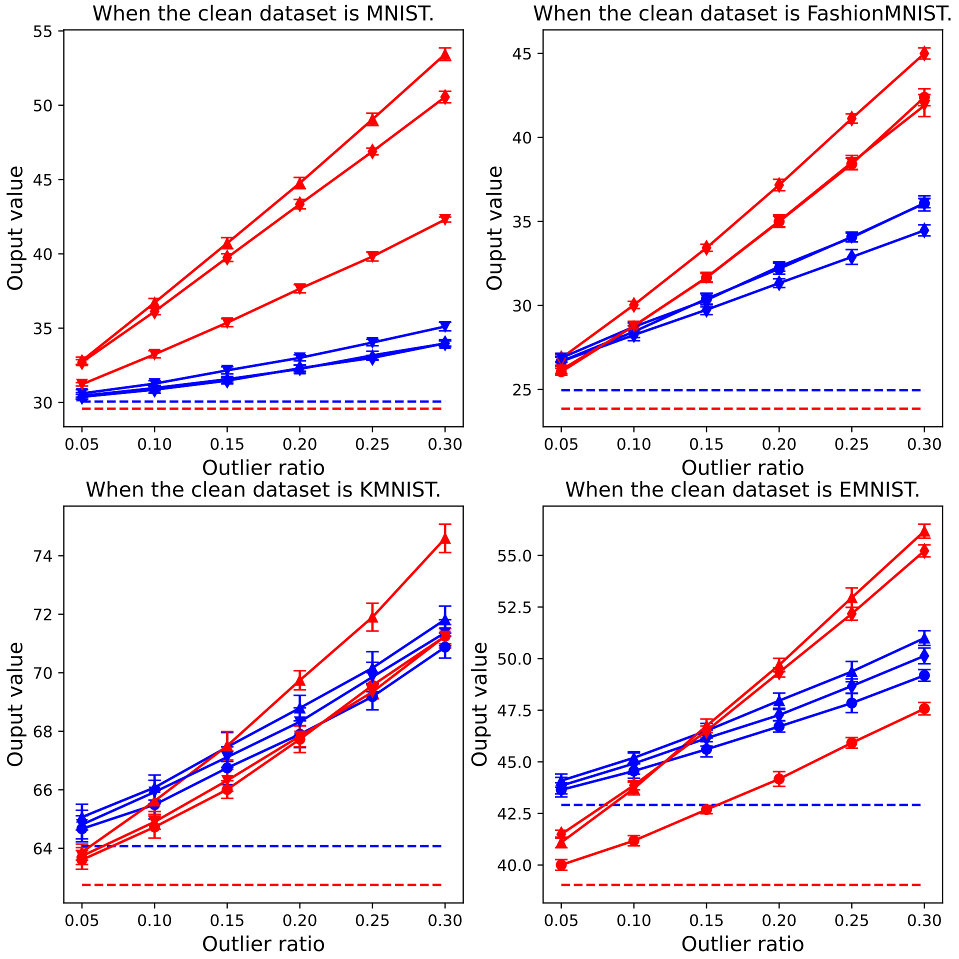


Figure 3: The mean and standard deviation of the output value of the Sinkhorn algorithm (red) and our algorithm (blue) over 20 runs. ○, △, ▽, and ♢ represents when the outlier dataset are MNIST, FashionMNIST, KMNIST, and EMNIST, respectively. The dotted line is the output value when the dataset is clean.

	Outliers	Inliers
One-class SVM	49.78 ± 1.83 %	49.99 ± 0.10 %
Local outlier factor	49.43 ± 3.68 %	99.13 ± 0.11 %
Isolation forest	42.78 ± 6.95 %	72.81 ± 3.34 %
Elliptical envelope	95.57 ± 2.77 %	69.37 ± 5.61 %
Baseline technique (95th)	92.78 ± 1.67 %	92.66 ± 0.44 %
Baseline technique (97.5th)	84.19 ± 2.10 %	96.41 ± 0.32 %
Baseline technique (99th)	65.04 ± 2.75 %	98.60 ± 0.16 %
ROBOT (95th)	99.96 ± 0.08 %	68.76 ± 0.49 %
ROBOT (97.5th)	99.89 ± 0.14 %	77.22 ± 0.63 %
ROBOT (99th)	99.48 ± 0.31 %	84.79 ± 0.47 %
Our Method (95th)	98.98 ± 0.66 %	86.72 ± 0.72 %
Our Method (97.5th)	96.96 ± 1.71 %	91.58 ± 0.38 %
Our Method (99th)	92.25 ± 1.53 %	95.73 ± 0.34 %

Table 3: The percentage of true outliers/inliers detected as outliers/inliers over 50 runs. The numbers show the mean and standard deviation. "(Xth)" means Xth percentile was used in its subsampling phase.

In this experiment, we used Fashion-MNIST (Xiao et al., 2017) as a clean dataset and MNIST (Deng, 2012) as outliers. ν_n consists of 9500 images from Fashion-MNIST and 500 images from MNIST. μ_m consists of 10000 images from Fashion-MNIST. We computed the transport matrix with the two datasets and identified the outlying MNIST images. We simply used the raw data to compute the distance matrix $\gamma_{ij} = \|\boldsymbol{x}_i - \boldsymbol{y}_j\|_2^2$, i.e., the Euclidean distance between raw data.

We compared the proposed method with the "ROBust Optimal Transport" (ROBOT) method (Mukherjee et al., 2021) and the method proposed by Balaji et al.(2020) , which are existing methods to compute OT robustly. We also compared our method with a variety of popular outlier detection algorithms available in scikit-learn (Pedregosa et al., 2011): the one-class support vector machine (SVM) (Schölkopf et al., 1999), local outlier factor (Breunig et al., 2000), isolation forest (Liu et al., 2008), and elliptical envelope (Rousseeuw and van Driessen, 1999). In the ROBOT method, we set the cost truncation hyperparameter to the (1) 95th (2) 97.5th (3) 99th percentile of the distance matrix in the subsampling phase (Mukherjee et al., 2021).

For our method, the distance tolerance parameter z in Definition 1 is necessary to detect outliers by leveraging Proposition 1. Once z is chosen, after running the algorithm $\left\lfloor \frac{\frac{z}{\lambda}(\beta-1)-1}{(\frac{1}{m})^{\beta-1}+(\frac{1}{n})^{\beta-1}} \right\rfloor$ times satisfying the condition (39), points in ν_n that are far from any points in μ_m with more than or equal to distance z are regarded as outliers. To determine z, we need a subsampling phase using the clean dataset similar to Mukherjee et al.(2021). We propose the following heuristics: since we know that μ_m is clean, we subsample two datasets from it and compute the distance matrix. Then, we choose the minimum value for each row and use the largest value among them as z. This procedure is essentially estimating the maximum distance between two samples in the clean dataset. In order to

	Outliers	Inliers
Balaji et al.[2020]	89.0 ± 16.9 %	67.0 ± 8.9 %
Our Method	96.6 ± 2.0 %	88.0 ± 0.7 %

Table 4: Comparison with Balaji et al.[2020] with 1000 data points. The numbers show the mean and standard deviation of the percetage of the true outliers/inliers detected as outliers/inliers over 10 runs.

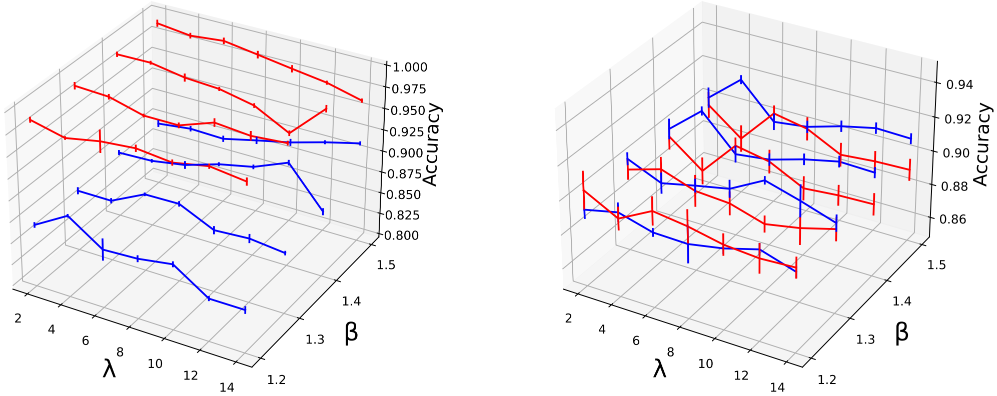
avoid subsampling noise, we used the (1) 95th (2) 97.5th (3) 99th percentile instead of the maximum. Additionaly, we compared our method with a natural baseline to identify a data point as an outlier if the minimum distance to the clean dataset is larger than the distance computed in the subsampling phase. We call this method "the baseline technique". The results are shown in Table 3. One can see that our method has a high performance in detecting not only outliers but also inliers.

We also tried the code of Balaji et al.(2020) based on CVXPY (Diamond and Boyd, 2016), which is not scalable so that the computational time is not negligible even with 1000 data points. Similar to the previous experiments, the clean dataset μ_m consists of 1000 Fashion-MNIST data points and the polluted dataset ν_n consists of 950 Fashion-MNIST data as inliers and 50 MNIST data points as outliers. Table 4 shows the mean accuracy and standard deviations over 10 runs. The run-time of their method was 820 ± 17 seconds, while that of our method was 6 ± 0.2 seconds. Our method outperforms the method by Balaji et al.[2020] in terms of not only outlier detection performance but also computation time.

4.3. The selection of hyperparamters β and λ

Here, we dicuss the selection of hyperparameters β and λ. Figure 4a shows the sensitivity to the hyperparameters for the same outlier task above. We see $2 \leq \lambda \leq 14$ and $1.2 \leq \beta \leq 1.5$ are good choices of possible hyperparameters.

Then, how about other β or λ? Since, we are running the algorithm $\left\lfloor \frac{\frac{z}{\lambda}(\beta-1)-1}{(\frac{1}{m})^{\beta-1}+(\frac{1}{n})^{\beta-1}} \right\rfloor$ times, if we choose β excessively large or λ excessively small, we will harmfully increase the computation time. On the other hand, if we choose β excessively small or excessively λ large, $\left\lfloor \frac{\frac{z}{\lambda}(\beta-1)-1}{(\frac{1}{m})^{\beta-1}+(\frac{1}{n})^{\beta-1}} \right\rfloor$ will become less than or equal to 0, which means that we can not start running the algorithm. However, these discussions are when z is fixed. If we scale the raw value of data by constant multiplication, z will also change. By scaling z, we can adjust the number of times running the algorithm so that it will be larger than 0, and at the same time, not too large. In the MNIST detection task, we scaled the raw data so that the number of times running the algorithm will fit in $(0, 20)$ when $(\beta, \lambda) =$ (1.4, 14), (1.3, 10), (1.3, 12), (1.3, 14), (1.2, 6), (1.2, 8), (1.2, 10), (1.2, 12), (1.2, 14). We can see that scaling the raw data by constant multiplication has no problem in detecting outliers (Figure 4a). Therefore, since we can scale z, we can adjust the number of times running the algorithm with limited $\lambda \in [2, 14]$ and $\beta \in [1.2, 1.5]$.



(a) The hyperparameter sensitivity in the Fashion-MNIST detection task. (Blue) The inlier detection accuracy. (Red) The outlier detection arruracy. Error bars represent the mean and standard deviation.

(b) The hyperparameter sensitivity in the credit card fraud detection task. (Blue) The inlier detection accuracy. (Red) The outlier detection arruracy. Error bars represent the mean and standard deviation.

Figure 4: The hyperparameter (β and λ) sensitivity in the Fashion-MNIST dataset and in the credit card fraud detection dataset.

Below, we confirm that the proposed method is sufficiently stable in the above range of hyperparameters by using the credit card fraud detection dataset[2]. We experimentally observe the sensitivity of the proposed method to the choice of the hyperparameters β and λ. We used the credit card fraud detection dataset to verify that the proposed method is sufficiently stable in a certain range of the hyperparameters.

The credit card fraud detection dataset contains transactions made by credit cards in 2013 by European cardholders. Due to confidentiality issues, it does not provide the original features and more background information about the data. Instead, it contains 28-dimensional numerical feature vectors, which are the result of a principal component analysis transformation. We used these feature vectors to compute the cost matrix, which is the L2 distance among them. The task is to detect 450 frauds out of 9000 transactions. We conducted ten experiments for each pair of β and λ.

We show the results in Figure 4b. We can see that the detection accuracy is sufficiently stable when $1.2 \leq \beta \leq 1.5$ and $1 \leq \lambda \leq 14$.

5. Conclusion

In this work, we proposed to robustly approximate OT by regularizing the ordinary OT with the β-potential term. By leveraging the domain of the Fenchel conjugate of the β-potential, our algorithm does not move any probability mass to outliers. We demonstrated that our proposed method can be used in estimating a probability distribution robustly even in the presence of outliers and successfully detecting outliers from a contaminated dataset.

[2] https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

Acknowledgments

SN was supported by JST SPRING, Grant Number JPMJSP2108. MS was supported by JST CREST Grant Number JPMJCR18A2.

References

Sabahat Akram and Qurrat ul Ann. Newton Raphson method. 2015.

Yogesh Balaji, Rama Chellappa, and Soheil Feizi. Robust optimal transport with applications in generative modeling and domain adaptation. In *NeurIPS*, pages 12934–12944, 2020.

Ayanendranath Basu, Ian R. Harris, Nils L. Hjort, and M. C. Jones. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559, 1998.

H.H. Bauschke and J.M. Borwein. Legendre functions and the method of random Bregman projections. *Journal of Convex Analysis*, 1997.

Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. LOF: Identifying density-based local outliers. *SIGMOD Record*, 29(2):93–104, 2000.

Lenaic Chizat. *Unbalanced Optimal Transport : Models, Numerical Methods, Applications*. Theses, Université Paris sciences et Lettres, 2017.

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical Japanese literature. *CoRR*, abs/1812.01718, 2018. URL http://arxiv.org/abs/1812.01718.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of mnist to handwritten letters, 2017. URL https://arxiv.org/abs/1702.05373.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, NIPS'13, page 2292–2300, 2013.

Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Arnaud Dessein, Nicolas Papadakis, and Jean-Luc Rouas. Regularized optimal transport and the ROT mover's distance. *JMLR*, 19(1):590–642, 2018.

Steven Diamond and Stephen Boyd. CVXPY: A python-embedded modeling language for convex optimization. *J. Mach. Learn. Res.*, 17(1):2909–2913, jan 2016. ISSN 1532-4435.

Futoshi Futami, Issei Sato, and Masashi Sugiyama. Variational inference based on robust divergences. In *AISTATS*, volume 84, pages 813–822. PMLR, 2018.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, page 2672–2680, Cambridge, MA, USA, 2014.

Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. A least-squares approach to direct importance estimation. *JMLR*, 10:1391–1445, 2009.

Paul Knopp and Richard Sinkhorn. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343 – 348, 1967.

Jernej Kos, Ian Fischer, and Dawn Xiaodong Song. Adversarial examples for generative models. *2018 IEEE Security and Privacy Workshops (SPW)*, pages 36–42, 2018.

S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86, 1951.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, 2008.

Debarghya Mukherjee, Aritra Guha, Justin M Solomon, Yuekai Sun, and Mikhail Yurochkin. Outlier-robust optimal transport. In *ICML*, pages 7850–7860, 2021.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *JMLR*, 12:2825–2830, 2011.

Gabriel Peyré and Marco Cuturi. Computational optimal transport. *Foundations and Trends in Machine Learning*, 11 (5-6):355–602, 2019.

Julien Rabin, Gabriel Peyré, Julie Delon, and Marc Bernot. Wasserstein barycenter and its application to texture mixing. In *Scale Space and Variational Methods in Computer Vision*, pages 435–446, 2012.

Peter J. Rousseeuw and Katrien van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

Bernhard Schölkopf, Robert Williamson, Alex Smola, John Shawe-Taylor, and John Platt. Support vector method for novelty detection. In *NeurIPS*, 1999.

Justin Solomon, Fernando de Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas. Convolutional Wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Transactions on Graphics*, 34(4), 2015.

Guillaume Staerman, Pierre Laforgue, Pavlo Mozharovskyi, and Florence d'Alché Buc. When OT meets MoM: Robust estimation of wasserstein distance. In *AISTATS*, volume 130, pages 136–144, 2021.

Cédric Villani. *Optimal transport – Old and new*, volume 338. Springer Berlin, Heidelberg, 2008.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, 2017.