Unbalanced minibatch Optimal Transport; applications to Domain Adaptation

Kilian Fatras [1] **Thibault Séjourné** [2] **Nicolas Courty** [1] **Rémi Flamary** [3]

Abstract

Optimal transport distances have found many applications in machine learning for their capacity to compare non-parametric probability distributions. Yet their algorithmic complexity generally prevents their direct use on large scale datasets. Among the possible strategies to alleviate this issue, practitioners can rely on computing estimates of these distances over subsets of data, *i.e.* minibatches. While computationally appealing, we highlight in this paper some limits of this strategy, arguing it can lead to undesirable smoothing effects. As an alternative, we suggest that the same minibatch strategy coupled with unbalanced optimal transport can yield more robust behavior. We discuss the associated theoretical properties, such as unbiased estimators, existence of gradients and concentration bounds. Our experimental study shows that in challenging problems associated to domain adaptation, the use of unbalanced optimal transport leads to significantly better results, competing with or surpassing recent baselines.

1. Introduction

Computing distances between distributions is a fundamental problem in machine learning. As an example, considering the space of distributions $\mathcal{M}_+(\mathcal{X})$ over a space $\mathcal{X}$, and given an empirical distribution $\alpha \in \mathcal{M}_+(\mathcal{X})$, many machine learning problems amount to estimate a distribution β_λ parametrized by a vector λ which approximates the distribution α. In order to compute the dissimilarities between distributions, it is common to rely on a contrast function or divergence $L : \mathcal{M}_+(\mathcal{X}) \times \mathcal{M}_+(\mathcal{X}) \rightarrow \mathbb{R}_+$. In this setting, the goal is to find the optimal λ^* which minimizes the distance L between the distributions β_λ and α, *i.e.* $\lambda^* = \operatorname{argmin}_\lambda L(\alpha, \beta_\lambda)$. As the available distributions are mostly empirical and come from data, the function L

needs good statistical estimation properties and optimization guarantees when using modern optimization techniques. Optimal transport (OT) losses have emerged recently as a competitive loss candidate for generative models (Arjovsky et al., 2017; Genevay et al., 2018). It also proved to be competitive in the context of Domain Adaptation (Courty et al., 2017; Courty et al., 2017; Shen et al., 2018) or for missing data imputation (Muzellec et al., 2020). The corresponding estimator is usually found in the literature under the name of *Minimum Kantorovich Estimator* (Bassetti et al., 2006; Peyré & Cuturi, 2019). However the computation of OT losses is a challenging problem, its computational cost being of order $\mathcal{O}(n^3 log(n))$, where n is the number of samples. Variants and approximations of optimal transport have been proposed to reduce its complexity. One of the most popular consists in adding an entropic regularization (Cuturi, 2013), leading to the Sinkhorn algorithm with complexity $\tilde{\mathcal{O}}(n^2)$ in both space and time. However, when n is large, computing OT remains rather expensive and might not fit on GPUs. The KeOps package (Feydy et al., 2019) allows to overcome this difficulty and avoid overflows by storing operations as formulas and stream computation on the fly. It is still difficult to use it in deep learning applications which involves high dimensional data and repeated computations of gradients. Another approach is to focus on the Wasserstein-1 distance which has a nice reformulation but needs to approximate 1-Lipschitz functions, which meets some difficulties in practice (Arjovsky et al., 2017; Gulrajani et al., 2017).

Minibatch Optimal Transport. A straightforward and scalable approach consists in computing OT solutions over subsets (minibatches) of the original data (α and β) and averaging the results as a proxy for the original problem. Such idea stems from the need to scale OT in practice and was applied in several situations (Kolouri et al., 2016; Genevay et al., 2018; Damodaran et al., 2018; Liutkus et al., 2019). It was proven for generative models that minimizers of the minibatch loss converge to the true minimizer when the minibatch size increases (Bernton et al., 2019). There exists deviation bounds between the true OT loss and a single minibatch estimate (Sommerfeld et al., 2019). Finally, concentration bounds and optimization properties for averaged minibatch OT were exhibited in (Fatras et al., 2020; 2021). However, the gain in computation time is achieved at the expense of the quality of the final transport plan, which turns

---

[1]Univ. Bretagne-Sud, CNRS, INRIA, IRISA, France [2]ENS, PSL University [3]École Polytechnique, CMAP, France. Correspondence to: Kilian Fatras <kilian.fatras@irisa.fr>.
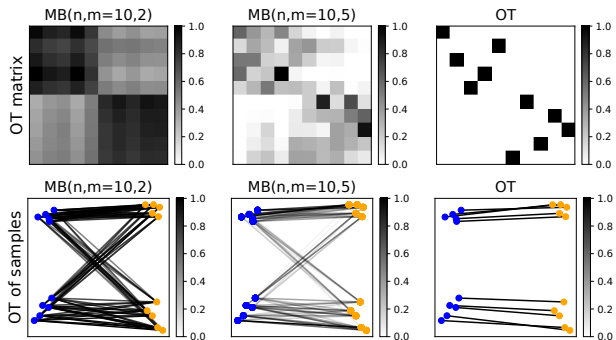
Figure 1. OT matrices, normalized by maximum value, between 2D distributions with $n = 10$ samples. The first row shows the MBOT plans for different minibatch size m. The second row is the corresponding 2D visualization of the transport plan support.

out to be notably less sparse, leading to undesired pairings between samples that would not be coupled with exact OT. Figure 1 illustrates this effect on a 2D toy example which shows that samples from the same cluster in the source probability can be coupled to two different clusters in the target. We propose to handle this problem by leveraging the theory of unbalanced OT and computing a more robust transport plan at the minibatch level.

Contributions and outline of the paper. We study in this work an alternative formulation of the minibatch OT where the unbalanced OT program, a variant with relaxed marginal constraints (Liero et al., 2017), is used at the minibatch level. Our rationale is that a geometrically robust version of OT computed between minibatches decreases the influence of undesired couplings between samples. The benefits of unbalanced MBOT are twofold: *i)* it yields a loss function more robust to minibatch sampling effects *ii)* our formulation approximates unbalanced OT but scales computationally w.r.t. the minibatch size, which allows its practical use for large datasets and deep learning applications. The contributions of the paper are the following. First we review the existing UOT formulations and introduce the one we consider in Section 2. We discuss the limits of minibatch OT in Section 3. We present the minibatch framework, study its statistical and optimization properties in Section 4. Finally, we design a new domain adaptation (DA) method whose performances are evaluated on several problems, where we show evidences that our strategy surpasses substantially other classical OT formulations, and is on par or better than recent state-of-the-art competitors. Our empirical results suggest that UOT might be more suitable than OT when dealing with real world data.

Notations. In this paper, we use the following notations. Let $\boldsymbol{X} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n)$ (resp. $\boldsymbol{Y} = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_n)$) be n *iid* random vectors in $\mathbb{R}^d$ drawn from a distribution α (resp. β) on the source (resp. target) domain. We associate to

$\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\}$ and $\{\boldsymbol{y}_1, \cdots, \boldsymbol{y}_n\}$ uniform vectors denoted $(\boldsymbol{u}_n)_i = (\frac{m_\alpha}{n})_i$, where $m_\alpha = \int d\alpha$ is the mass of α. The quantities $\boldsymbol{X}$ and $\boldsymbol{u}_n$ allows one to recover an empirical distributions as $\alpha_n = \frac{1}{n} \sum_i \delta_{\boldsymbol{x}_i}$. We denote $\alpha^{\otimes m}$ for a sample of m random variables following the distribution α. The ground cost can be formalised as the following map:

$$C^n : (\boldsymbol{X}, \boldsymbol{Y}) \mapsto \big(d(\boldsymbol{x}_i, \boldsymbol{y}_j)\big)_{1 \leq i,j \leq n} \in \mathcal{M}_n(\mathbb{R}). \quad (1)$$

We further suppose that α and β have compact support which means that the ground cost is bounded by a strictly positive constant M. This assumption holds for most machine learning applications where distributions are given by empirical samples.

2. Related work and background

We review in this section previous Unbalanced OT formulations, detail the one we consider in our approach and discuss the use of OT in robust machine learning.

Unbalanced Optimal Transport. Unbalanced OT is a generalization of 'classical' OT that relaxes the conservation of mass constraints by allowing the system to either transport or create and destroy mass. Our loss builds upon (Liero et al., 2017) which replaces the 'hard' marginal constraints of OT by 'soft' penalties using Csiszàr divergences. There exists other extensions of the static formulations of OT. A famous one is partial OT which consists in transporting a fixed budget of mass (Figalli, 2010) or to move mass in and out of the system at a fixed cost (Figalli & Gigli, 2010). Another line of work proposes to optimize over various sets of Lipschitz functions (Hanin, 1992; Piccoli & Rossi, 2014; Schmitzer & Wirth, 2017). One can also replace Csiszàr divergences by integral probability metrics (Nath, 2020).

Consider a convex, positive, lower-semicontinuous function ϕ such that $\phi(1) = 0$. Define $\phi'_\infty = \lim_{x \to +\infty} \phi(\boldsymbol{x})/\boldsymbol{x}$ that we suppose strictly positive. Csiszàr divergences D_ϕ are measures of discrepancy that compare pointwise ratios of mass using a penalty ϕ and are defined as $D_\phi(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\boldsymbol{y}_i \neq 0} \boldsymbol{y}_i \phi\big(\frac{\boldsymbol{x}_i}{\boldsymbol{y}_i}\big) + \phi'_\infty \sum_{\boldsymbol{y}_i = 0} \boldsymbol{x}_i$. Total Variation and Kullback-Leibler divergences ($\mathrm{KL}(\boldsymbol{x}|\boldsymbol{y}) = \sum_i \boldsymbol{x}_i \log(\frac{\boldsymbol{x}_i}{\boldsymbol{y}_i}) - \boldsymbol{x}_i + \boldsymbol{y}_i$) are particular instances of such divergence. Consider two positive distributions $\alpha, \beta \in \mathcal{M}_+(\mathcal{X})$. The UOT program between distributions and cost c is defined as

$$\mathrm{OT}_\phi^{\tau, \varepsilon}(\alpha, \beta, c) = \min_{\pi \in \mathcal{M}_+(\mathcal{X}^2)} \int c d\pi + \varepsilon \mathrm{KL}(\pi | \alpha \otimes \beta)$$
$$+ \tau(D_\phi(\pi_1 \| \alpha) + D_\phi(\pi_2 \| \beta)), \quad (2)$$

where π is the transport plan, π_1 and π_2 the plan's marginals, τ is the marginal penalization and $\varepsilon \geq 0$ is the regularization coefficient. Note that the marginals of π are no longer equal

to (α, β) in general. The considered formulation is computable via a generalized Sinkhorn algorithm (Chizat et al., 2018; Séjourné et al., 2019) which is proved to converge. Its complexity for $D_\phi = \mathtt{KL}$ is $\tilde{O}(n^2/\epsilon)$ (Pham et al., 2020). Balanced OT is recovered for inputs (α, β) with equal mass, when $\tau \to \infty$ (hence we note it $\mathrm{OT}^{\infty,0}$). When distributions are discrete, UOT can be expressed as $\mathrm{OT}_\phi^{\tau,\varepsilon}(\boldsymbol{a}, \boldsymbol{b}, C)$ where $\boldsymbol{a}$ and $\boldsymbol{b}$ are two positive vectors, $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}_+^n$ and C is the ground cost.

A shortcoming of adding entropy is the loss of metric properties since $\mathrm{OT}_\phi^{\tau,\varepsilon}(\beta, \beta, c) \neq 0$. It motivated (Séjourné et al., 2019) to introduce an unbalanced generalization of the Sinkhorn divergence (Genevay et al., 2018):

$$S_\phi^{\tau,\varepsilon}(\alpha, \beta, c) = \mathrm{OT}_\phi^{\tau,\varepsilon}(\alpha, \beta, c) + \frac{\varepsilon}{2}(m_\alpha - m_\beta)^2 \quad (3)$$
$$- \frac{1}{2}\mathrm{OT}_\phi^{\tau,\varepsilon}(\alpha, \alpha, c) - \frac{1}{2}\mathrm{OT}_\phi^{\tau,\varepsilon}(\beta, \beta, c),$$

Computing the unbalanced sinkhorn divergence above is of the same order of complexity as the UOT loss. When $e^{-C/\epsilon}$ is a positive definite kernel, $S_\phi^{\tau,\varepsilon}$ is a convex, symmetric, positive definite loss function which metrizes the convergence in law (Séjourné et al., 2019). Thus it allows to mitigate between accelerated computations and conservation of key theoretical guarantees. Regarding empirical estimation, OT suffers from the curse of dimension which means that it is hard to estimate when data lie in high dimension d. Its sample complexity, *i.e.,* its convergence in population, is proven to be in $O\left(\frac{1}{\sqrt{n}}\left(1 + \frac{1}{\varepsilon^{\lfloor d/2 \rfloor}}\right)\right)$ both for OT and UOT (Genevay et al., 2019; Séjourné et al., 2019).

Optimal Transport and robustness in machine learning. UOT is known to be more robust to outliers than OT as it does not need to meet the marginals. Several other formulations make optimal transport robust for practical and statistical reasons. Partial OT can be adapted for partial matchings problem with applications for positive-unlabeled learning (Chapel et al., 2020). A line of work proposes 'distributionnally robust' models, where models are trained in a Wasserstein ball around the empirical distribution in the space of probabilities (Mohajerin Esfahani & Kuhn, 2018; Kuhn et al., 2019). In a similar approach, several variants relax the OT marginal constraints with a ball constraint, and consider several penalties such as integral probability metrics (Nath, 2020), total variation or Csiszàr divergences for outlier detection (Mukherjee et al., 2020; Balaji et al., 2020). Such relaxations allow to derive statistical guarantees w.r.t. noise and outliers. Another idea to ensure robustness consists in learning the cost adversarially, and is formulated as a max-min problem where the cost is modeled by an Euclidean embedding (Genevay et al., 2018), a compact space of matrices (Dhouib et al., 2020) or a projection on a lower dimensional subspace (Paty & Cuturi, 2019).

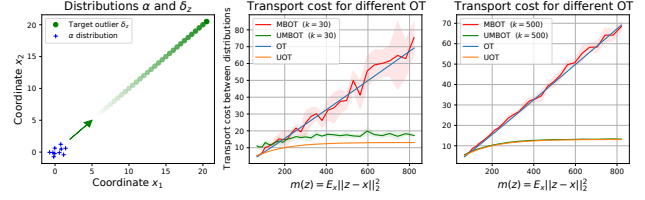In the next section we discuss OT sensitivities in more de-

Figure 2. Several OT costs between 2D distributions with $n = 10$ samples and $m = 5$. Target distribution is equal to the source distribution tainted with a moving outlier (green dot). The shaded area represent the variance of subsample MBOT on 5 run (see section 4).

tails and highlight their exacerbation by minibatch strategy.

3. Minibatch OT and robustness to sampling

In this section, we discuss the limitations of combining balanced OT with the minibatch framework. OT is sensitive to the distributions geometry. When those distributions are tainted by outliers, OT is forced to transport them due to the marginal constraints, inducing an undesirable extra transportation cost. Minibatch OT averages several OT terms related to subsamples of the original distributions, thus sharing this sensitivity. The problem is even worse as two minibatches do not necessarily share samples that would lie in the support of the full OT plan, hence forced to match samples that could be, at the level of a minibatch, considered as outliers. Take as an example two distributions with clustered samples. While in the full OT plan clusters can be matched exactly, those clusters are likely to appear as imbalanced in the minibatches, especially if the size of the minibatch is small and does not respect the statistics of the original distribution. Due to the marginal constraints, samples from one cluster are likely to be matched to unrelated clusters, as depicted in Figure 1. This explains why in practice previous works relied on large minibatches to mitigate this sampling effect (Damodaran et al., 2018). To overcome this issue, we propose the natural solution of relaxing the marginal constraints at the minibatch level. The expected outcome is twofold: *i)* mitigating the effect of subsampling in the minibatch strategy and *ii)* providing a natural and scalable robust optimal transport computation strategy at the global level. We discuss in the following some theoretical considerations to support this claim.

Theoretical analysis: impact of an outlier We start by examining the impact of an outlier in the behaviors of OT and UOT. The following lemma illustrates the relations between those two quantities.

Lemma 1. *Take (α, β) two probability distributions. For $\zeta \in [0,1]$, write $\tilde{\alpha} = \zeta\alpha + (1-\zeta)\delta_{\boldsymbol{z}}$ a distribution perturbed by a Dirac outlier located at some z outside of the support of (α, β). Take the unregularized OT loss $\mathrm{OT}_{KL}^{\tau,0}$ with KL*

entropy and cost C. Write $m(\boldsymbol{z}) = \int C(\boldsymbol{z}, \boldsymbol{y}) d\beta(\boldsymbol{y})$. One has:

$$\mathrm{OT}_{KL}^{\tau,0}(\tilde{\alpha}, \beta, C) \leq \zeta \, \mathrm{OT}_{KL}^{\tau,0}(\alpha, \beta, C) \quad (4)$$
$$+ 2\tau(1 - \zeta)(1 - e^{-m(\boldsymbol{z})/2\tau})$$

Now take the unregularized, balanced OT loss $\mathrm{OT}_\phi^{\infty,0}$ with cost C. Write (f, g) the optimal dual potentials (i.e. functions) of $\mathrm{OT}_\phi^{\infty,0}(\alpha, \beta)$, and y^* in β's support. Then:

$$\mathrm{OT}_\phi^{\infty,0}(\tilde{\alpha}, \beta) \geq \zeta \, \mathrm{OT}_\phi^{\infty,0}(\alpha, \beta) \quad (5)$$
$$+ (1 - \zeta)\Big(C(\boldsymbol{z}, y^*) - g(y^*) + \int g \, d\beta\Big)$$

Equation (5) shows that when $\boldsymbol{z}$ gets further from the supports of (α, β), the OT loss increases. However for UOT the upper bound (4) tends to saturate as $\boldsymbol{z}$ gets further away. What remains is the UOT loss between distributions whose outliers are removed, with a cost of removing the outlier proportional to its mass.

We first illustrate Lemma 1 with a toy example in Figure 2. We consider a probability distribution α tainted with an outlier (green dot) to get a target probability distribution $\alpha' = \frac{1}{n+1}(n.\alpha + \delta_{\boldsymbol{z}})$. We then move away the outlier from α's support, as shown with the green arrow, and we calculate several OT costs. The minibatch size is set to $m = 5$ and the total cost is the average of k OT costs between those minibatches. We see that OT variants are not robust to the outlier as their loss increases along the outlier displacement unlike UOT variants which reach a plateau as predicted by Lemma 1. Each computation is done 5 times to show that variance is lower for bigger k and that UMBOT has a lower variance than MBOT for $k = 30$ and $k = 500$.

We consider now an example in 2D, akin to Figure 1, where our goal is to illustrate the OT plan between two empirical distributions of 10 samples in Figure 3. We use two 2D empirical distributions where the samples belong to a certain cluster depending on a related class (color information). The source data are equally distributed between classes while the target data have different proportions, 3 samples belong to the red class while 7 samples belong to the green class. Different proportions between domains are ubiquitous for real world data. We compare unbalanced minibatch OT, minibatch OT, entropic OT and UOT. For UOT, the divergence D_ϕ equals to KL divergence and for the minibatch variant, the minibatch size is $m = 2$. We can see from the OT plans in the first row of the figure that the cluster structure is more or less recovered. However OT and minibatch OT tend to connect samples from different classes. This configuration would lead, for instance, to negative transfer in a context of domain adaptation applications, *i.e.,* matching of samples between different domains. This is less true for UOT, where the pairings between different classes is diminished and tend to disappear when we reduce the penalty τ.

4. Unbalanced Minibatch Optimal Transport

In this section we express some mathematical properties at the heart of this work. In (Fatras et al., 2020), authors described some properties of the minibatch OT. We provide here extensions of those results to Unbalanced OT. We start by defining minibatch estimators as done in (Fatras et al., 2020), then we generalize the concentration bounds and optimization properties from (Fatras et al., 2020). To derive concentration bounds we prove that the UOT cost is finite and the transport plan is bounded. Without loss of generality, we consider n-tuples $\boldsymbol{X}$ and $\boldsymbol{Y}$ with uniform vectors $\boldsymbol{u}_n$, to form empirical distributions encountered in the different applications and the associated ground cost matrix C.

4.1. Minibatch Unbalanced OT estimation

Estimators. To build minibatches, we select m samples from $\boldsymbol{X}$ and $\boldsymbol{Y}$. We rely on a generic element of indices $I = (i_1, \ldots, i_m) \in [\![n]\!]^m$, which is called an index m-tuple. I represents the selected samples from the n-data tuple $\boldsymbol{X}$ or $\boldsymbol{Y}$. In this work, we only focus on m-tuples without replacement I, whose their set is denoted $\mathcal{P}^m$.

Definition 1 (Minibatch UOT). *Let $C = C^n(\boldsymbol{X}, \boldsymbol{Y})$ be a square matrix of size n. Given an unbalanced OT loss $h \in \{\mathrm{OT}_\phi^{\tau,\varepsilon}, S_\phi^{\tau,\varepsilon}\}$ and an integer $m \leq n$, we define the following quantity:*

$$\bar{h}_C^m(\boldsymbol{X}, \boldsymbol{Y}) := \frac{(n-m)!^2}{n!^2} \sum_{I,J \in \mathcal{P}^m} h(\boldsymbol{u}_m, \boldsymbol{u}_m, C_{I,J}) \quad (6)$$

where for I, J two m-tuples, $C_{(I,J)}$ is the matrix extracted from C by keeping the rows and columns corresponding to I and J respectively. We denote the optimal plan $\Pi_{I,J}$, lifted as a $n \times n$ matrix where all entries are zero except those indexed in $I \times J$. We define the averaged minibatch transport plan: $\overline{\Pi}^m(\boldsymbol{X}, \boldsymbol{Y}) := \binom{n}{m}^{-2} \sum_{I,J \in \mathcal{P}^m} \Pi_{I,J}$.

We omit C when clear from context. A simple combinatorial argument provided in appendix assures that the sum of $\boldsymbol{u}_m$ over all m-tuples I gives $\boldsymbol{u}_n$. Note that this formalism was first proposed in (Fatras et al., 2020). In the formulation above, we no longer compute UOT between the full distributions but instead we compute the expectation of UOT over all minibatches drawn from $\alpha^{\otimes m} \otimes \beta^{\otimes m}$:

$$E_h := \mathop{\mathbb{E}}_{(\boldsymbol{X}, \boldsymbol{Y}) \sim \alpha^{\otimes m} \otimes \beta^{\otimes m}} [h(\boldsymbol{u}_m, \boldsymbol{u}_m, C^m(\boldsymbol{X}, \boldsymbol{Y}))], \quad (7)$$

The combinatorial number of terms is prohibitive to compute, fortunately we can rely on subsample quantities.

Definition 2 (Minibatch subsampling). *Consider the notations of definition 1. Pick an integer $k > 0$, we define:*

$$\widetilde{h}_{k,C}^m(\boldsymbol{X}, \boldsymbol{Y}) := k^{-1} \sum_{(I,J) \in D_k} h(\boldsymbol{u}_m, \boldsymbol{u}_m, C_{I,J}) \quad (8)$$
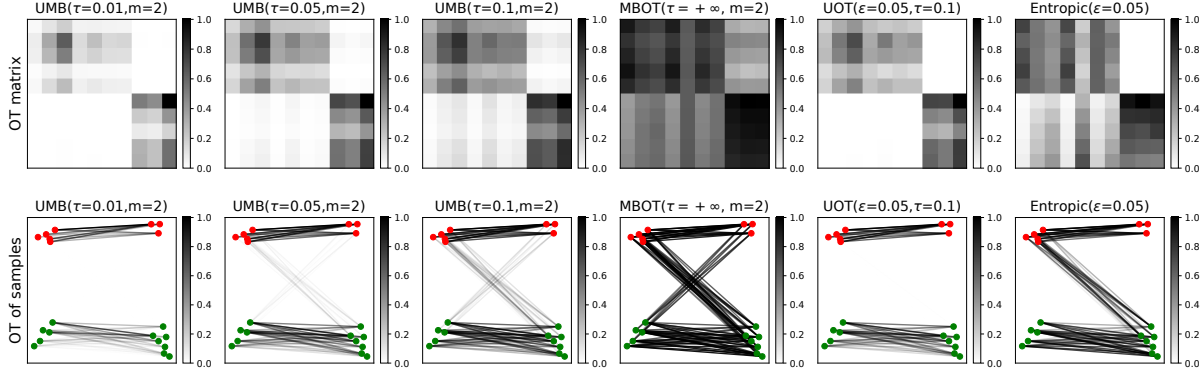
Figure 3. Several OT plans, normalized by their maximum value, between 2D distributions with $n = 10$ samples. The first row shows the minibatch OT plans $\overline{\Pi}^m$ for different values of m, the second row provides an equivalent geometric interpretation of the OT plans, where the mass transportation is depicted as connexions between samples.

where D_k is a set of cardinality k whose elements are drawn at random from the uniform distribution on $\Gamma := \mathcal{P}_m \times \mathcal{P}_m$. A similar construction holds for incomplete minibatch transport plan denoted as $\widetilde{\Pi}_k^m(\boldsymbol{X}, \boldsymbol{Y})$.

Note that $\bar{h}^m$ and $\widetilde{h}_k^m$ are unbiased estimators of E_h as they are, respectively, complete and incomplete U-statistics (J Lee, 2019). The minibatch UOT losses are positive and symmetric, however they are not definites, *i.e.*, $\bar{h}^m(\boldsymbol{X}, \boldsymbol{X}) > 0$ for non trivial $\boldsymbol{X}$ and $1 < m < n$.

4.2. Deviation bounds

To generalize the concentration bounds from (Fatras et al., 2020), we first need to assure that our estimators are bounded. Our first lemma intends to show that the UOT cost is finite and that the optimal transport plan is bounded.

Lemma 2 (Bounded UOT and optimal transport plan). *Let C be a ground cost and $\boldsymbol{a}, \boldsymbol{b}$ two positive vectors in $\mathbb{R}_+^n$ such that $m_{\boldsymbol{a}} = \|\boldsymbol{a}\|_1 > 0$ and $m_{\boldsymbol{b}} = \|\boldsymbol{b}\|_1 > 0$. Assume that $\langle \boldsymbol{a}\boldsymbol{b}^\top, C \rangle < +\infty$. Consider $h = \mathrm{OT}_\phi^{\tau,\varepsilon}$ and assume $\varepsilon > 0$ or $\phi'_\infty > 0$. Then $h(\boldsymbol{a}, \boldsymbol{b}, C)$ is finite and the set of optimal transport plan is a compact set.*

It is straightforward to prove boundedness of $h = S_\phi^{\tau,\varepsilon}$ from Lemma 2. We can now turn to establish concentration bounds for both incomplete estimators $\widetilde{h}_k^m$ and $\widetilde{\Pi}_k^m$.

Theorem 1 (Maximal deviation bound). *Let $\delta \in (0,1)$, three integers $k \geq 1$ and $m \leq n$ be fixed. Consider two n-tuples $\boldsymbol{X} \sim \alpha^{\otimes n}$ and $\boldsymbol{Y} \sim \beta^{\otimes n}$ and a kernel $h \in \{\mathrm{OT}_\phi^{\tau,\varepsilon}, S_\phi^{\tau,\varepsilon}\}$. We have a maximal deviation bound between $\widetilde{h}_k^m(\boldsymbol{X}, \boldsymbol{Y})$ and E_h depending on the number of samples n and the number of batches k. With probability at least $1 - \delta$ on the draw of $\boldsymbol{X}, \boldsymbol{Y}$ and D_k we have:*

$$|\widetilde{h}_k^m(\boldsymbol{X}, \boldsymbol{Y}) - E_h| \leq M \left(\sqrt{\frac{\log(\frac{2}{\delta})}{2\lfloor \frac{n}{m} \rfloor}} + \sqrt{\frac{2\log(\frac{2}{\delta})}{k}} \right),$$

where M is the UOT upper bound. Furthermore, for $h = \mathrm{OT}_\phi^{\tau,\varepsilon}$, let $\mathfrak{M}_{\overline{\Pi}}^\infty$ be the maximum mass of minibatch transport plans. For all $k \geq 1$, all $1 \leq i \leq n$, with probability at least $1 - \delta$ on the draw of $\boldsymbol{X}, \boldsymbol{Y}$ and D_k we have:

$$|\widetilde{\Pi}_k^m(\boldsymbol{X}, \boldsymbol{Y})_{(i)} \mathbf{1}_n - \overline{\Pi}^m(\boldsymbol{X}, \boldsymbol{Y})_{(i)} \mathbf{1}_n| \leq \mathfrak{M}_{\overline{\Pi}}^\infty \sqrt{\frac{2\log(2/\delta)}{k}},$$

where we denote by $\Pi_{(i)}$ the i-th row of matrix Π and by $\mathbf{1}_n \in \mathbb{R}^n$ the vector whose entries are all equal to 1.

This deviation bound shows that if we increase the number of data n and batches k while keeping the minibatch size m fixed, we get closer to the expectation. The rate $\frac{m}{n}$ is almost optimal and is the same as in (Fatras et al., 2020). The main difference is the upper bound M which bounds UOT. Note that the bound does not depend on the dimension of $\mathcal{X}$ unlike original unbalanced OT (Séjourné et al., 2019). Regarding OT plans, the constant $\mathfrak{M}_{\Pi}$ represents the maximum mass of minibatch transport plans which would be 1 for OT.

4.3. Unbiased gradients and optimization

The Wasserstein distance is known to suffer from biased gradients (Bellemare et al., 2017), meaning that minimizing the estimator of the Wasserstein distance with empirical distributions does not lead to the minimum of the Wasserstein distance between full distribution. In (Fatras et al., 2020), authors showed that minibatch entropic OT does not suffer from these biased gradients. In this section we show that this property can be generalized for minibatch UOT, including unregularized UOT. We achieve this point by relying on Clarke regularity as proposed in (Fatras et al., 2021).

We study a standard parametric data fitting problem. Given some discrete samples $(x_i)_{i=1}^n \subset \mathcal{X}$ from an unknown distribution α, our goal is to fit a parametric model $\lambda \mapsto \beta_\lambda \in \mathcal{M}_+(\mathcal{X})$ to α for $\lambda \in \Lambda$ in an Euclidian space. To do so, we use minibatch UOT and its incomplete estimators
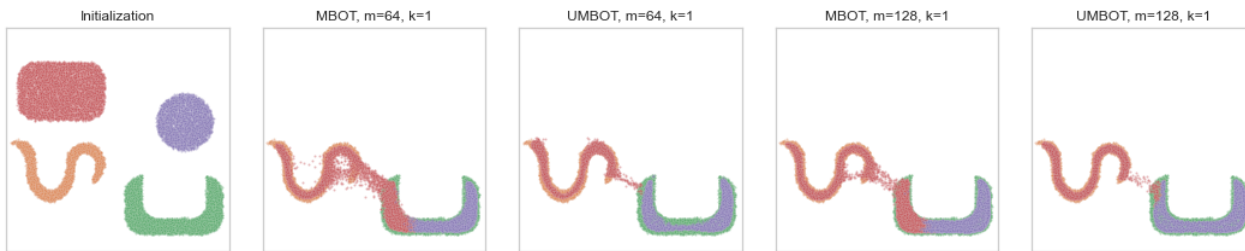
Figure 4. (Best viewed in colors) Minibatch UOT gradient flow on a 2D dataset. Source data and target data are divided in 2 imbalanced clusters, source left (red) and target right (green) shapes have 6400 samples while source right (purple) and target left (orange) shapes have 3600 samples. The batch size m is set to $\{64, 128\}$ and the number of minibatch k is set to 1, meaning that the explicit Euler integration step is conducted for each batch. Results are computed with the (unbalanced) minibatch Sinkhorn divergence losses.

$\widetilde{h}_k^m$ as a contrast loss. U-statistics as contrast loss have been studied in (Papa et al., 2015). Stochastic gradients (SGD) has also proven to be really efficient at optimizing neural network parameters λ even if they are non convex (Bottou, 2010). To justify the convergence of SGD, we need to exchange expectation and gradients and use that $\widetilde{h}_k^m$ is an unbiased estimator of E_h.

The former is not immediate because UOT is not differentiable as we do not have a unique optimal transport plan when $\varepsilon = 0$. Thus, we use the notion of Clarke generalized derivatives. They define a regularity for nonsmooth but locally Lipschitz and semi-continuous function. It is a close concept to subgradients for convex functions since when a convex function is locally Lipshitz at x the two notions are equivalent. An intuitive geometric interpretation is that a function is Clarke regular if it doesn't have "upwards dashes" in its graph, for a total survey see (Clarke, 1990).

Theorem 2. *Let $\hat{X}, \{\hat{Y}_\theta\}_{\theta \in \Theta}$ be two m-tuples of random vectors compactly supported, $h \in \{OT^{\tau,\varepsilon}, S^{\tau,\varepsilon}\}$ and C^m a $\mathbf{C}^1$ cost. Under an additional integrability assumption, we have:*

$$\partial_\theta \mathbb{E}[h(\boldsymbol{u}, \boldsymbol{u}, C^m(\hat{\boldsymbol{X}}, \hat{\boldsymbol{Y}}_\theta))] = \mathbb{E}[\partial_\theta h(\boldsymbol{u}, \boldsymbol{u}, C^m(\hat{\boldsymbol{X}}, \hat{\boldsymbol{Y}}_\theta))],$$

with both expectation being finite. Furthermore the function $\theta \mapsto -\mathbb{E}[h(\boldsymbol{u}, \boldsymbol{u}, C^m(\hat{\boldsymbol{X}}, \hat{\boldsymbol{Y}}_\theta))]$ is also Clarke regular.

Note that Theorem 2 implies that if we use the Minibatch UOT loss with $h \in \{OT_\phi^{\tau,\varepsilon}, S_\phi^{\tau,\varepsilon}\}$ as a loss function, then the minus objective function is Clarke regular. Furthermore, Stochastic gradient with decreasing step sizes converges almost surely to the set of critical points of Clarke generalized derivative (Davis et al., 2020; Majewski et al., 2018). As a consequence, it is justified to use SGD with minibatch UOT.

5. Experiments

In this section, we illustrate the practical behavior of unbalanced minibatch OT for gradient flow and for domain adaptation experiments. We relied on the POT package

(Flamary et al., 2021) to compute the exact OT solver or the entropic UOT loss and the Geomloss package (Feydy et al., 2019) for the Unbalanced Sinkhorn divergence. The experiments were designed in PyTorch (Paszke et al., 2017) and all the code can be found here[*].

5.1. Unbalanced MiniBatch OT gradient flow

Consider a given target distribution α, the purpose of gradient flows is to model a distribution $\beta(t)$ which at each iteration follows the gradient direction minimizing the loss $\beta_t \mapsto h(\alpha, \beta_t)$ (Peyré, 2015; Liutkus et al., 2019). The gradient flow simulate the non parametric setting of data fitting problem, where the modeled distribution β is parametrized by a vector position x that encodes its support.

We follow the same experimental procedure as in (Feydy et al., 2019). The gradient flow algorithm uses a simple explicit Euler integration scheme. Formally, it starts from an initial distribution at time $t = 0$ and integrates at each iteration a SDE. In our case, we cannot compute the gradient directly from our minibatch OT losses. As the OT loss inputs are empirical distributions, we have an inherent bias when we calculate the gradient from the weights $\frac{1}{m}$ of samples that we correct by multiplying the gradient by the inverse weight m. Finally, we integrate: $\dot{\boldsymbol{X}}(t) = -m\nabla_{\boldsymbol{X}}\widetilde{h}_k^m(\boldsymbol{X}, \boldsymbol{Y})$.

For α and $\beta(0)$ we generate 10000 2D points divided in 2 imbalanced clusters with number of samples in each cluster provided in Figure 4. We consider the (unbalanced) sinkhorn divergence, a squared euclidean cost, a learning rate of 0.02, 5000 iterations, m equals 64 or 128 and $k = 1$. We show the gradient flow of the upper clusters to the lower clusters in Figure 4. From the experiment, we can see that the minibatch OT is not robust to imbalanced classes on the contrary to the minibatch UOT. Indeed there are data from the upper left cluster which converge to the down right cluster and we can also see an overlap between the classes. Due to OT marginal constraints, the loss forces to transport all data in the batch which results in breaking the target

---

[*]https://github.com/kilianFatras/JUMBOT

shapes. This is not the case for minibatch UOT, which better respects the shape of target distributions.

5.2. Domain adaptation

We now follow the settings of unsupervised DA where both domains share the same labels $\mathcal{Y}_s = \mathcal{Y}_t$.

JUMBOT. Optimal Transport has been proposed in (Courty et al., 2017) as a way to solve the domain adaptation problem. Our method is based on (Damodaran et al., 2018) and aims at finding a joint distribution map between a source and a target distribution by taking into account a term on a neural network embedding space and on the label space. Formally, let g_θ be an embedding function where the input is mapped into the latent space and f_λ which maps the latent space to the label space on the target domain. The embedding space is in our experiment the before last layer of a neural network. For a given minibatch, embedding g_θ and classification map f_λ, the transfer term is:

$$\bar{h}^m_{C_{\theta,\lambda}}((\boldsymbol{X}^s, \boldsymbol{Y}^s), (\boldsymbol{X}^t, f_\lambda(g_\theta(\boldsymbol{X}^t)))), \text{with} \quad (9)$$

$$(C_{\theta,\lambda})_{i,j} = \eta_1 \|g_\theta(\boldsymbol{x}^s_i) - g_\theta(\boldsymbol{x}^t_j)\|^2_2 + \eta_2 \mathcal{L}(\boldsymbol{y}^s_i, f_\lambda(g_\theta(\boldsymbol{x}^t_j))),$$

where $\mathcal{L}(.,.)$ is the cross-entropy loss and η_1, η_2 are positive constants. Basically, this specific transportation cost combines a distance between the representation of the data through the neural network g_θ and a loss function between the associated labels (Courty et al., 2017). Taking $k = 1$ led to state of the art results. The Csiszàr divergence ϕ is the Kullback-Leibler divergence KL. We also add a cross entropy term on the source data. Hence our optimization problem is:

$$\min_{\theta,\lambda} \sum_i \mathcal{L}(f_\lambda(g_\theta(\boldsymbol{x}^s_i)), \boldsymbol{y}^s_i) \quad (10)$$
$$+ \eta_3 \widetilde{h}^m_{k,C_{\theta,\lambda}}((\boldsymbol{X}^s, \boldsymbol{Y}^s), (\boldsymbol{X}^t, f_\lambda(g_\theta(\boldsymbol{X}^t)))).$$

Our method is called **JUMBOT** and stands for Joint Unbalanced MiniBatch OT. It is related to DEEPJDOT (Courty et al., 2017; Damodaran et al., 2018) at the notable exceptions that we use minibatch UOT, which can also handle partial domain adaptation as suggested by our experiments.

Datasets. We start with **digits** datasets. Following the evaluation protocol of (Damodaran et al., 2018) we experiment on three adaptation scenarios: USPS to MNIST (U↦M), MNIST to M-MNIST (M↦MM), and SVHN to MNIST (S↦M). MNIST (LeCun & Cortes, 2010) contains 60,000 images of handwritten digits, M-MNIST contains the 60,000 MNIST images with color patches (Ganin et al., 2016) and USPS (Hull, 1994) contains 7,291 images. Street View House Numbers (SVHN)(Netzer et al., 2011) consists of 73, 257 images with digits and numbers in natural scenes. We report the evaluation results on the test target datasets. **Office-Home** (Venkateswara et al., 2017) is a difficult dataset for
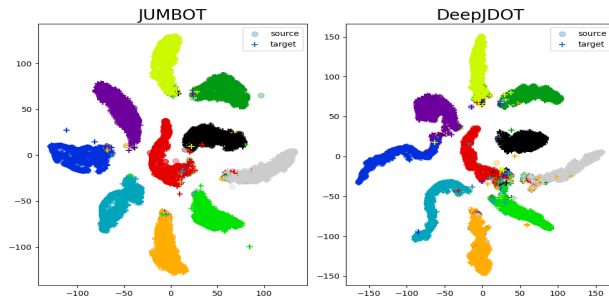

Figure 5. T-SNE embeddings of 10000 test samples for MNIST (source) and MNIST-M(target) for DEEPJDOT and our method. It shows the ability of the methods to discriminate classes (samples are colored w.r.t. their classes).

unsupervised domain adaptation (UDA), it has 15,500 images from four different domains: Artistic images (A), Clip Art (C), Product images (P) and Real-World (R). For each domain, the dataset contains images of 65 object categories that are common in office and home scenarios. We evaluate all methods in 12 adaptation scenarios. **VisDA-2017** (Peng et al., 2017) is a large-scale dataset for UDA from simulation to real. VisDA contains 152,397 synthetic images as the source domain and 55,388 real-world images as the target domain. 12 object categories are shared by these two domains. Following (Long et al., 2018; Chen et al., 2020), we evaluate all methods on VisDA validation set.

Results. We compare our method against state of the art methods: DANN(Ganin et al., 2016), CDAN-E(Long et al., 2018), DEEPJDOT(Damodaran et al., 2018), ALDA(Chen et al., 2020), ROT(Balaji et al., 2020). Details about training procedure and architectures can be found in supplementary.

We reported the test score at the end of optimization for all benchmarks and methods. The results on digit datasets can be found in Table 2 where (*) denotes reproduced results. We conducted each experiment three times and report their average results and variance. For fair comparisons, we only resize and normalize the image without data augmentation. We see that our method performs best with a margin of at least 1.5 points. Furthermore, we see an important 4% increase of the performance compared to DEEPJDOT. A deeper analysis of this difference is considered in the next paragraph. Office-Home results are gathered in Table 1 and VisDA are reported in Table 3. For fair comparison with previous work, we used a similar data pre-processing and we used the ten-crop technique (Long et al., 2018; Chen et al., 2020) for testing our methods. Experiments in (Balaji et al., 2020) follow a different setup explaining the difference of performance between their reported score and our reproduced score. JUMBOT achieves the best accuracy on average and on 11 of the 12 scenarios on the Office-Home dataset and achieves the best accuracy on VisDA at the end of training with a margin of 4% and 2% respectively.

	Method	A-C	A-P	A-R	C-A	C-P	C-R	P-A	P-C	P-R	R-A	R-C	R-P	avg
DA	RESNET-50	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
	DANN (*)	44.3	59.8	69.8	48.0	58.3	63.0	49.7	42.7	70.6	64.0	51.7	78.3	58.3
	CDAN-E(*)	52.5	71.4	76.1	59.7	69.9	71.5	58.7	50.3	77.5	70.5	57.9	**83.5**	66.6
	DEEPJDOT (*)	50.7	68.6	74.4	59.9	65.8	68.1	55.2	46.3	73.8	66.0	54.9	78.3	63.5
	E-DEEPJDOT (*)	50.6	68.9	74.4	59.3	65.1	69.0	56.2	46.5	74.5	65.1	54.7	78.1	63.5
	ALDA (*)	52.2	69.3	76.4	58.7	68.2	71.1	57.4	49.6	76.8	70.6	57.3	82.5	65.8
	ROT (*)	47.2	71.8	76.4	58.6	68.1	70.2	56.5	45.0	75.8	69.4	52.1	80.6	64.3
	JUMBOT	**55.2**	**75.5**	**80.8**	**65.5**	**74.4**	**74.9**	**65.2**	**52.7**	**79.2**	**73.0**	**59.9**	83.4	**70.0**
PDA	RESNET-50	46.3	67.5	75.9	59.1	59.9	62.7	58.2	41.8	74.9	67.4	48.2	74.2	61.4
	DEEPJDOT(*)	48.2	66.2	76.6	56.1	57.8	64.5	58.3	42.7	73.5	65.7	48.2	73.7	60.9
	E-DEEPJDOT(*)	47.6	67.0	77.3	57.1	57.9	65.4	58.1	41.3	74.4	66.4	47.7	75.1	61.3
	PADA	51.9	67.0	78.7	52.2	53.8	59.0	52.6	43.2	78.8	73.7	56.6	77.1	62.1
	ETN	59.2	77.0	79.5	62.9	65.7	75.0	68.3	55.4	84.4	75.7	57.7	**84.5**	70.4
	BA3US(*)	56.7	76.0	**84.8**	73.9	67.8	**83.7**	72.7	56.5	84.9	77.8	64.5	83.8	73.6
	JUMBOT	**62.7**	**77.5**	84.4	**76.0**	**73.3**	80.5	**74.7**	**60.8**	**85.1**	**80.2**	**66.5**	83.9	**75.5**

Table 1. Summary table of DA and Partial DA results on Office-Home (ResNet-50). (*) denotes the reproduced methods.

Methods/DA	$U \mapsto M$	$M \mapsto MM$	$S \mapsto M$	Avg
DANN(*)	92.2 ± 0.3	96.1 ± 0.6	88.7 ± 1.2	92.3
CDAN-E(*)	$\mathbf{99.2 \pm 0.1}$	95.0 ± 3.4	90.9 ± 4.8	95.0
ALDA(*)	97.0 ± 1.4	96.4 ± 0.3	96.1 ± 0.1	96.5
DEEPJDOT(*)	96.4 ± 0.3	91.8 ± 0.2	95.4 ± 0.1	94.5
E-DEEPJDOT(*)	97.1 ± 0.3	94.2 ± 0.1	97.6 ± 0.1	96.3
JUMBOT	98.2 ± 0.1	$\mathbf{97.0 \pm 0.3}$	$\mathbf{98.9 \pm 0.1}$	**98.0**

Table 2. Summary table of DA results on digit datasets. Experiments were run three times. (*) denotes the reproduced methods.

Methods	Accuracy (in %)
CDAN-E(*)	70.1
ALDA(*)	70.5
DEEPJDOT(*)	68.0
E-DEEPJDOT(*)	69.2
ROBUST OT(*)	66.3
JUMBOT	**72.5**

Table 3. Summary table of DA results on VisDA datasets. (*) denotes the reproduced methods.

Ablation. The main difference between JUMBOT and DEEPJDOT is the use of a different OT solver. JUMBOT uses entropic regularized unbalanced OT and DEEPJDOT uses original OT. We have conducted an ablation study in all our domain adaptation experiments to measure the effect of the entropic regularization and the unbalanced formulation. The use of entropic regularized OT in place of original OT in DEEPJDOT is denoted E-DEEPJDOT. We can see that for the digits experiments, the entropic regularization helps to get better performances of 0.6% on the DA task USPS ↦ MNIST and more than 2% on SVHN ↦ MNIST. For the last digit task MNIST ↦ M-MNIST, the grid search on ε needed to be reduced due to some numerical instabilities. Overall, the performances of E-DEEPJDOT are still lower than the entropic regularized Unbalanced OT of JUMBOT by more than 1%. A similar performance gain was observed on the VisDA dataset, where E-DEEPJDOT was 1.2% higher than DEEPJDOT but still more than 3% below JUMBOT. On

the Office-Home experiments, the entropic regularization alone did not get better results.

Analysis. In this paragraph, we study the difference of behavior between DEEPJDOT and JUMBOT. Along JUMBOT's training on the DA task USPS to MNIST, we measured the percentage of mass between data with different labels at each iteration. In average along training about 7% of DEEPJDOT connections are between data with different labels while this percentage decreases to 0.7% for JUMBOT. So DEEPJDOT transfers wrong labels to target data which will decrease the overall accuracy. We also plot a TSNE embedding of our method and DEEPJDOT (see Figure 5), we can see that there are some overlaps between clusters for DEEPJDOT unlike our method. This is probably due to the minibatch smoothing effect which would tend to bring clusters of different classes closer.

We also provide a *sensitivity analysis* to the batch size, ε and τ parameters. All results are gathered in Figure 6. When τ is too small, JUMBOT creates negative transfer because of the entropic regularization. When τ increases, we see that JUMBOT accuracy increases and it reaches its maximum around $\tau = 1$. However when τ is too high, we recover entropic OT. When ε varies for JUMBOT and the entropic variant of DEEPJDOT, we see that entropy helps getting slightly better results. However when the entropic regularization coefficient is too big, the accuracy falls. We conjecture that entropic regularized OT regularizes the neural network because the target prediction is matched to a smoothed source label (see a similar discussion in (Damodaran et al., 2019)). And it is well known that label smoothing creates class clusters in the penultimate layer of the neural network (Müller et al., 2019). We now discuss the minibatch size. While JUMBOT has a constant accuracy along all batch size, the DEEPJDOT accuracy falls of 4% for SVHN ↦ MNIST and 6% for USPS ↦ MNIST. The benefits of our method over DEEPJDOT are twofold, it is more robust to small batch sizes
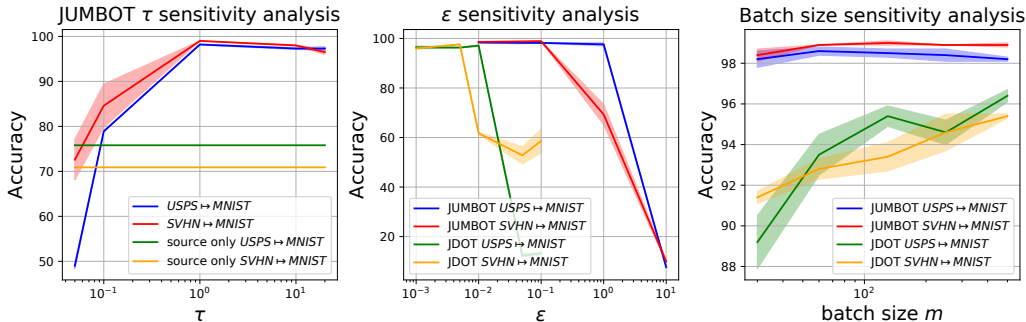
Figure 6. (Best viewed in colors) DEEPJDOT and JUMBOT sensitivity analysis. We report the classification accuracy of DEEPJDOT and JUMBOT on the DA tasks USPS $\mapsto$ MNIST and SVHN $\mapsto$ MNIST for several hyperparameter variations. We consider the marginal coeffifient τ, the entropic coefficient ε and the batch size m.

and it is performant for small computation budget unlike DEEPJDOT. Finally, we provide in appendix a classification accuracy along training which demonstrates the network overfitting with DEEPJDOT and not with JUMBOT.

5.3. Partial DA

Finally we consider the Partial DA (PDA) application. In PDA, the target labels are a subset of the source labels, *i.e.*, $\mathcal{Y}_t \subset \mathcal{Y}_s$. Samples belonging to these missing classes become outliers which can produce negative transfer. We want to investigate the robustness of our method in such an extreme scenario. We evaluate our method on partial Office-Home, where we follow (Cao et al., 2018) to select the first 25 categories (in alphabetic order) in each domain as a partial target domain. We compare our method against state of the art PDA methods: PADA (Cao et al., 2018), ETN (Cao et al., 2019) and BA3US (Jian et al., 2020). For fair comparison we followed the experimental setting of PADA, ETN and BA3US and we report to the supplementary material for more details. The final performances are gathered in the lower part of table 1. Note that we do not use the ten-crop technics for evaluating the methods, as we were not able to reproduce PADA and ETN. We can see that JUMBOT is state of the art on 9 out of the 12 domain adaptation tasks and is on average 2% above competitors. Finally, we also evaluate DEEPJDOT (Damodaran et al., 2018) and its entropic variant E-DEEPJDOT on the PDA task to compare the performances of regularized unbalanced OT against regularized and unregularized OT. E-DEEPJDOT gives similar results to DEEPJDOT and they both produce negative transfer. JUMBOT is on average 15% higher on this problem, showing the clear advantages of our strategy in the presence of unbalanced classes.

6. Conclusion

Computing minibatches is a common practice to accommodate large quantities of data in deep learning, and can be

used in synergy with OT. However it amplifies the shortcomings of OT due to its marginal constraints combined with subsampling effects, which is detrimental to learning application performances. To mitigate this issue, we propose to relax those constraints and use UOT at the minibatch level. We showed that not only theoretical properties are preserved with such loss, but it also dampens negative coupling effects, yielding a more efficient measure of comparison between data distributions. We notably showed it can reach state-of-the-art performances on challenging domain adaptation problems. We believe those results will encourage the use of minibatch Unbalanced OT in machine learning applications.

References

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017. 1

Balaji, Y., Chellappa, R., and Feizi, S. Robust optimal transport with applications in generative modeling and

domain adaptation. In *Advances in Neural Information Processing Systems*, 2020. 3, 7

Bassetti, F., Bodini, A., and Regazzini, E. On minimum kantorovich distance estimators. *Statistics & Probability Letters*, 76, 2006. 1

Bellemare, M. G., Danihelka, I., Dabney, W., Mohamed, S., Lakshminarayanan, B., Hoyer, S., and Munos, R. The cramer distance as a solution to biased wasserstein gradients. *CoRR*, abs/1705.10743, 2017. 5

Bernton, E., Jacob, P. E., Gerber, M., and Robert, C. P. On parameter estimation with the wasserstein distance. *Information and Inference: A Journal of the IMA*, 8(4): 657–676, 2019. 1

Bertsekas, D. P. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

Bottou, L. Large-scale machine learning with stochastic gradient descent. In *in COMPSTAT*, 2010. 6

Cao, Z., Ma, L., Long, M., and Wang, J. Partial adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 9

Cao, Z., You, K., Long, M., Wang, J., and Yang, Q. Learning to transfer examples for partial domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 9

Chapel, L., Alaya, M. Z., and Gasso, G. Partial optimal transport with applications on positive-unlabeled learning. In *Advances in Neural Information Processing Systems*, 2020. 3

Chen, M., Zhao, S., Liu, H., and Cai, D. Adversarial-learned loss for domain adaptation. *arXiv*, abs/2001.01046, 2020. 7

Chizat, L., Peyré, G., Schmitzer, B., and Vialard, F. Scaling algorithms for unbalanced optimal transport problems. *Math. Comput.*, 87(314):2563–2609, 2018. doi: 10.1090/mcom/3303. 3

Clarke, F. H. *Optimization and nonsmooth analysis*. SIAM, 1990. 6

Clarke, H. F. Generalized gradients and applications. *Transactions of The American Mathematical Society*, pp. 247–247, 1975.

Courty, N., Flamary, R., Habrard, A., and Rakotomamonjy, A. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems*, 2017. 1, 7

Courty, N., Flamary, R., Tuia, D., and Rakotomamonjy, A. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 1

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems 26*, 2013. 1

Damodaran, B. B., Kellenberger, B., Flamary, R., Tuia, D., and Courty, N. DeepJDOT: Deep Joint Distribution Optimal Transport for Unsupervised Domain Adaptation. In *ECCV 2018 - 15th European Conference on Computer Vision*. Springer, 2018. 1, 3, 7, 9

Damodaran, B. B., Flamary, R., Seguy, V., and Courty, N. An Entropic Optimal Transport Loss for Learning Deep Neural Networks under Label Noise in Remote Sensing Images. In *Computer Vision and Image Understanding*, 2019. 8

Davis, D., Drusvyatskiy, D., Kakade, S., and Lee, J. D. Stochastic subgradient method converges on tame functions. *Foundations of computational mathematics*, 20(1): 119–154, 2020. 6

Dhouib, S., Redko, I., Kerdoncuff, T., Emonet, R., and Sebban, M. A swiss army knife for minimax optimal transport. In *International Conference on Machine Learning*, pp. 2504–2513, 2020. 3

Fatras, K., Zine, Y., Flamary, R., Gribonval, R., and Courty, N. Learning with minibatch wasserstein : asymptotic and gradient properties. In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2131–2141. PMLR, 26–28 Aug 2020. 1, 4, 5

Fatras, K., Zine, Y., Majewski, S., Flamary, R., Gribonval, R., and Courty, N. Minibatch optimal transport distances; analysis and applications. *CoRR*, 2021. 1, 5

Feydy, J., Séjourné, T., Vialard, F.-X., Amari, S.-i., Trouve, A., and Peyré, G. Interpolating between optimal transport and mmd using sinkhorn divergences. In *Proceedings of Machine Learning Research*, 2019. 1, 6

Figalli, A. The optimal partial transport problem. *Archive for Rational Mechanics and Analysis*, 195:533–560, 02 2010. 2

Figalli, A. and Gigli, N. A new transportation distance between non-negative measures, with applications to gradients flows with dirichlet boundary conditions. *Journal de mathématiques pures et appliquées*, 94(2):107–130, 2010. 2

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Bois-bunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D. J., Tavenard, R., Tong, A., and Vayer, T. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021. URL http://jmlr.org/papers/v22/20-451.html. 6

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., March, M., and Lempitsky, V. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016. 7

Genevay, A., Peyre, G., and Cuturi, M. Learning generative models with sinkhorn divergences. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, 2018. 1, 3

Genevay, A., Chizat, L., Bach, F., Cuturi, M., and Peyré, G. Sample complexity of sinkhorn divergences. In *Proceedings of Machine Learning Research*, 2019. 3

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems 30*. 2017. 1

Hanin, L. Kantorovich-rubinstein norm and its application in the theory of lipschitz spaces. 1992. 2

Hoeffding, W. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 1963.

Hull, J. Database for handwritten text recognition research. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16, 1994. doi: 10.1109/34.291440. 7

J Lee, A. U-statistics : theory and practice / a. j. lee. *SERBIULA (sistema Librum 2.0)*, 2019. 5

Jian, L., Yunbo, W., Dapeng, H., Ran, H., and Jiashi, F. A balanced and uncertainty-aware approach for partial domain adaptation. In *European Conference on Computer Vision (ECCV)*, August 2020. 9

Kolouri, S., Zou, Y., and Rohde, G. K. Sliced wasserstein kernels for probability distributions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1

Kuhn, D., Mohajerin Esfahani, P., Nguyen, V. A., and Shafieezadeh Abadeh, S. Wasserstein distributionally robust optimization: Theory and applications in machine learning. *INFORMS TutORials in Operations Research*, 2019. 3

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010. 7

Liero, M., Mielke, A., and Savaré, G. Optimal entropy-transport problems and a new hellinger–kantorovich distance between positive measures. *Inventiones mathematicae*, 211(3):969–1117, Dec 2017. ISSN 1432-1297. doi: 10.1007/s00222-017-0759-8. 2

Liutkus, A., Simsekli, U., Majewski, S., Durmus, A., and Stöter, F.-R. Sliced-Wasserstein flows: Nonparametric generative modeling via optimal transport and diffusions. In *Proceedings of the 36th International Conference on Machine Learning*, 2019. 1, 6

Long, M., Cao, Z., Wang, J., and Jordan, M. I. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pp. 1645–1655, 2018. 7

Majewski, S., Miasojedow, B., and Moulines, E. Analysis of nonsmooth stochastic approximation: the differential inclusion approach. *arXiv preprint arXiv:1805.01916*, 2018. 6

Mohajerin Esfahani, P. and Kuhn, D. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Math. Program.*, 171(1–2):115–166, September 2018. ISSN 0025-5610. 3

Mukherjee, D., Guha, A., Solomon, J., Sun, Y., and Yurochkin, M. Outlier-robust optimal transport. *CoRR*, 2020. 3

Müller, R., Kornblith, S., and Hinton, G. E. When does label smoothing help? In *Advances in Neural Information Processing Systems*, 2019. 8

Muzellec, B., Josse, J., Boyer, C., and Cuturi, M. Missing data imputation using optimal transport. *arXiv preprint arXiv:2002.03860*, 2020. 1

Nath, J. S. Unbalanced optimal transport using integral probability metric regularization. *CoRR*, 2020. 2, 3

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. 7

Papa, G., Clémençon, S., and Bellet, A. Sgd algorithms based on incomplete u-statistics: Large-scale minimization of empirical risk. In *Advances in Neural Information Processing Systems 28*, 2015. 6

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017. 6

Paty, F.-P. and Cuturi, M. Subspace robust wasserstein distances. In *International Conference on Machine Learning*, pp. 5072–5081, 2019. 3

Peng, X., Usman, B., Kaushik, N., Hoffman, J., Wang, D., and Saenko, K. Visda: The visual domain adaptation challenge. *CoRR*, abs/1710.06924, 2017. 7

Peyré, G. Entropic approximation of wasserstein gradient flows. *SIAM Journal on Imaging Sciences*, 2015. 6

Peyré, G. and Cuturi, M. Computational optimal transport. *Foundations and Trends® in Machine Learning*, 2019. 1

Pham, K., Le, K., Ho, N., Pham, T., and Bui, H. On unbalanced optimal transport: An analysis of Sinkhorn algorithm. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7673–7682. PMLR, 13–18 Jul 2020. 3

Piccoli, B. and Rossi, F. On properties of the generalized wasserstein distance, 2014. 2

Schmitzer, B. and Wirth, B. A framework for wasserstein-1-type metrics. *ArXiv*, abs/1701.01945, 2017. 2

Séjourné, T., Feydy, J., Vialard, F.-X., Trouvé, A., and Peyré, G. Sinkhorn divergences for unbalanced optimal transport. *arXiv preprint arXiv:1910.12958*, 2019. 3, 5

Shen, J., Qu, Y., Zhang, W., and Yu, Y. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 1

Sommerfeld, M., Schrieber, J., Zemel, Y., and Munk, A. Optimal transport: Fast probabilistic approximation with exact solvers. *Journal of Machine Learning Research*, 2019. 1

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *(IEEE) Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 7