The Unbalanced Gromov Wasserstein Distance: Conic Formulation and Relaxation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Comparing metric measure spaces (i.e. a metric space endowed with a probability distribution) is at the heart of many machine learning problems. The most popular distance between such metric measure spaces is the Gromov-Wasserstein (GW) distance, which is the solution of a quadratic assignment problem. The GW distance is however limited to the comparison of metric measure spaces endowed with a *probability* distribution. To alleviate this issue, we introduce two Unbalanced Gromov-Wasserstein formulations: a distance and a more tractable upper-bounding relaxation. They both allow the comparison of metric spaces equipped with arbitrary positive measures up to isometries. The first formulation is a positive and definite divergence based on a relaxation of the mass conservation constraint using a novel type of quadratically-homogeneous divergence. This divergence works hand in hand with the entropic regularization approach which is popular to solve large scale optimal transport problems. We show that the underlying non-convex optimization problem can be efficiently tackled using a highly parallelizable and GPU-friendly iterative scheme. The second formulation is a distance between mm-spaces up to isometries based on a conic lifting. Lastly, we provide numerical experiments on synthetic examples and domain adaptation data with a Positive-Unlabeled learning task to highlight the salient features of the unbalanced divergence and its potential applications in ML.

1 Introduction

Comparing data distributions on different metric spaces is a basic problem in machine learning. This class of problems is for instance at the heart of surfaces [Bronstein et al., 2006] or graph matching [Xu et al., 2019] (equipping the surface or graph with its associated geodesic distance), regression problems in quantum chemistry [Gilmer et al., 2017] (viewing the molecules as distributions of points in $\mathbb{R}^3$) and natural language processing [Grave et al., 2019, Alvarez-Melis and Jaakkola, 2018] (where texts in different languages are embedded as points distributions in different vector spaces).

Metric measure spaces. The mathematical way to formalize these problems is to model the data as *metric measure spaces* (mm-spaces). A mm-space is denoted as $\mathcal{X} = (X, d, \mu)$ where X is a complete separable set endowed with a distance d and a positive Borel measure $\mu \in \mathcal{M}_+(X)$. For instance, if $X = (x_i)_i$ is a finite set of points, then $\mu = \sum_i m_i \delta_{x_i}$ (here δ_{x_i} is the Dirac mass at x_i) is simply a set of positive weights $m_i = \mu(\{x_i\}) \geq 0$ associated to each point x_i, which accounts for its mass or importance. For instance, setting some m_i to 0 is equivalent to removing the point x_i. We refer to Sturm [2012] for a mathematical account on the theory of mm-spaces. In all the applications highlighted above, it makes sense to perform the comparisons up to isometric transformations of the data. Two mm-spaces $\mathcal{X} = (X, d_X, \mu)$ and $\mathcal{Y} = (Y, d_Y, \nu)$ are considered to be equal (denoted $\mathcal{X} \sim \mathcal{Y}$) if they are isometric, meaning that there is a bijection $\psi : \mathrm{spt}(\mu) \to \mathrm{spt}(\nu)$

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

(where $\mathrm{spt}(\mu)$ is the support of μ) such that $d_X(x,y) = d_Y(\psi(x), \psi(y))$ and $\psi_\sharp \mu = \nu$. Here $\psi_\sharp$ is the push-forward operator, so that $\psi_\sharp \mu = \nu$ is equivalent to imposing $\nu(A) = \mu(\psi^{-1}(A))$ for any set $A \subset Y$. For discrete spaces where $\mu = \sum_i m_i \delta_{x_i}$, then one should have $\nu = \psi_\sharp \mu = \sum_i m_i \delta_{\psi(x_i)}$. As highlighted by Mémoli [2011], considering mm-spaces up to isometry is a powerful way to formalize and analyze a wide variety of problems such as matching, regression and classification of distributions of points belonging to different spaces. The key to unlock all these problems is the computation of a distance between mm-spaces up to isometry. So far, existing distances (reviewed below) assume that μ is a probability distribution, i.e. $\mu(X) = 1$. This constraint is not natural and sometimes problematic for most of the practical applications to machine learning. The goal of this paper is to alleviate this restriction. We define for the first time a class of distances between unbalanced metric measure spaces, these distances being upper-bounded by divergences which can be approximated by an efficient numerical scheme.

Csiszár divergences The simplest case is when $X = Y$ and one simply ignores the underlying metric. One can then use Csiszár divergences (or φ-divergences), which perform a pointwise comparison (in contrast with optimal transport distances, which perform a displacement comparison). It is defined using an entropy function $\varphi : \mathbb{R}_+ \to [0, +\infty]$, which is a convex, lower semi-continuous, positive function with $\varphi(1) = 0$. The Csiszár φ-divergence reads $\mathrm{D}_\varphi(\mu|\nu) \triangleq \int_X \varphi(\frac{\mathrm{d}\mu}{\mathrm{d}\nu})\mathrm{d}\nu + \varphi'_\infty \int_X \mathrm{d}\mu^\perp$, where $\mu = \frac{\mathrm{d}\mu}{\mathrm{d}\nu}\nu + \mu^\perp$ is the Lebesgue decomposition of μ with respect to ν and $\varphi'_\infty = \lim_{r\to\infty} \varphi(r)/r \in \mathbb{R} \cup \{+\infty\}$ is called the recession constant. This divergence D_φ is convex, positive, 1-homogeneous and weak* lower-semicontinuous, see Liero et al. [2015] for details. Particular instances of φ-divergences are Kullback-Leibler (KL) for $\varphi(r) = r\log(r) - r + 1$ (note that $\varphi'_\infty = \infty$) and Total Variation (TV) for $\varphi(r) = |r - 1|$.

Balanced and unbalanced optimal transport. If the common embedding space X is equipped with a distance $d(x,y)$, one can use more elaborated methods such as optimal transport (OT) distances, which are computed by solving convex optimization problems. This type of methods has proven useful for ML problems as diverse as domain adaptation [Courty et al., 2014], supervised learning over histograms [Frogner et al., 2015] and unsupervised learning of generative models [Arjovsky et al., 2017]. In this case, the extension from probability distributions to arbitrary positive measures $(\mu, \nu) \in \mathcal{M}_+(X)^2$ is now well understood and corresponds to the theory of unbalanced OT. Following Liero et al. [2015], Chizat et al. [2018a], a family of unbalanced Wasserstein distances is defined by solving

$$\mathrm{UW}(\mu,\nu)^q \triangleq \inf_{\pi \in \mathcal{M}(X\times X)} \int \lambda(d(x,y))\mathrm{d}\pi(x,y) + \mathrm{D}_\varphi(\pi_1|\mu) + \mathrm{D}_\varphi(\pi_2|\mu). \tag{1}$$

Here (π_1, π_2) are the two marginals of the joint distribution π, defined by $\pi_1(A) = \pi(A \times Y)$ for $A \subset X$. The mapping $\lambda : \mathbb{R}^+ \to \mathbb{R}$ and exponent $q \geq 1$ should be chosen wisely to ensure for instance that UW defines a distance (see Section 2.2.1). It is frequent to take $\rho\mathrm{D}_\varphi$ instead of D_φ (i.e. take $\psi = \rho\varphi$) to adjust the strength of the marginals' penalization. Balanced OT is retrieved with the convex indicator $\varphi = \iota_{\{1\}}$ or by taking the limit $\rho \to +\infty$, which enforces $\pi_1 = \mu$ and $\pi_2 = \nu$. When $0 < \rho < +\infty$, unbalanced OT operates a trade-off between transportation and creation of mass, which is crucial to be robust to outliers in the data and to cope with mass variations in the modes of the distributions. For supervised tasks, the value of ρ should be cross-validated to obtain the best performances. Its use is gaining popularity in applications, such as medical imaging registration [Feydy et al., 2019a], videos [Lee et al., 2019], generative learning [Balaji et al., 2020] and gradient flow to train neural networks [Chizat and Bach, 2018, Rotskoff et al., 2019]. Furthermore, existing efficient algorithms for balanced OT extend to this unbalanced problem. In particular Sinkhorn's iterations, introduced in ML for balanced OT by Cuturi [2013], extend to unbalanced OT [Chizat et al., 2018b, Séjourné et al., 2019], as detailed in Section 3.

The Gromov-Wasserstein distance and its applications. The Gromov-Wasserstein (GW) distance [Mémoli, 2011, Sturm, 2012] generalizes the notion of OT to the setting of mm-spaces up to isometries. It replaces the linear cost $\int \lambda(d)\mathrm{d}\pi$ of OT by a quadratic function. It reads

$$\mathrm{GW}(\mathcal{X},\mathcal{Y})^q \triangleq \min_{\pi \in \mathcal{M}_+(X\times Y)} \left\{ \int \lambda(|d_X(x,x') - d_Y(y,y')|)\mathrm{d}\pi(x,y)\mathrm{d}\pi(x',y') \ : \ \begin{array}{l} \pi_1 = \mu \\ \pi_2 = \nu \end{array} \right\}. \tag{2}$$

It is proved in Mémoli [2011], Sturm [2012] that GW defines with $\lambda(t) = t^q$ a distance up to isometries on balanced mm-spaces (i.e. the measures are probability distributions). The GW distance

is applied successfully in natural language processing for unsupervised translation learning [Grave et al., 2019, Alvarez-Melis and Jaakkola, 2018], in generative learning for objects lying in spaces of different dimensions [Bunne et al., 2019] and to build VAE for graphs [Xu et al., 2020]. It has been adapted for domain adaptation over different spaces [Redko et al., 2020]. It is also a relevant distance to compute barycenters between graphs or shapes [Vayer et al., 2018, Chowdhury and Needham, 2020]. When $(\mathcal{X}, \mathcal{Y})$ are Euclidean spaces, this distance compares distributions up to rigid isometry, and is closely related (but not equal) to metrics defined by procrustes analysis [Grave et al., 2019, Alvarez-Melis et al., 2019]. The problem (2) is non convex because the quadratic form $\int \lambda(|d_X - d_Y|)\mathrm{d}\pi \otimes \pi$ is not positive in general. It is in fact closely related to quadratic assignment problems [Burkard et al., 1998], which are used for graph matching problems, and are known to be NP-hard in general. Nevertheless, non-convex optimization methods have been shown to be successful in practice to use GW distances for ML problems. This includes for instance alternating minimization [Mémoli, 2011, Redko et al., 2020] and entropic regularization [Peyré et al., 2016, Gold and Rangarajan, 1996].

Related works and contributions. The concomitant work of De Ponti and Mondino [2020] extends the L^p transportation distance defined in Sturm et al. [2006] to unbalanced mm-spaces and studies its geometric properties. This distortion distance is not equivalent to the GW distance, and is more difficult to estimate numerically because it explicitly imposes a triangle inequality constraint in the optimization problem. The work of Chapel et al. [2020] relaxes the GW distance to the unbalanced setting by hybridizing GW with partial OT [Figalli, 2010] for unsupervised labeling. It ressembles one particular setting of our formulation, but with some important differences, detailed in Section 2. Our construction is also connected to partial matching methods, which find numerous applications in graphics and vision [Cosmo et al., 2016]. In particular, Rodola et al. [2012] introduces a mass conservation relaxation of the GW problem.

The two main contributions of this paper are the definition of two formulations relaxing the GW distance. The first one is called the Unbalanced Gromov-Wasserstein (UGW) divergence and can be computed efficiently on GPUs. The second one is called the Conic Gromov-Wasserstein distance (CGW). It is proved to be a distance between mm-spaces endowed with positive measures up to isometries, as stated in Theorem 1 which is the main theoretical result of this paper. We also prove in Theorem 1 that UGW can be used as a surrogate upper-bounding CGW. We present those concepts and their properties in Section 2. We also detail in Section 3 an efficient computational scheme for a particular setting of UGW. This method computes an approximate stationary point of a biconvex relaxation of UGW. Even though it is a lower bound of the original problem, we provide in Theorem 2 conditions ensuring the tightness of this relaxation. The algorithm leverages the strength of entropic regularization and the Sinkhorn algorithm, namely that it is GPU-friendly and defines smooth loss functions amenable to back-propagation for ML applications. Section 4 provides some numerical experiments to highlight the qualitative behavior of this algorithm and its ability to cope with outliers and mass variations in the modes of the distributions. We illustrate numerically the tightness of the relation between UGW and CGW, showing that UGW is a reasonnable proxy of a distance, at least locally. We provide an application of our divergence in the positive unlabeled learning setting, using domain adaptation data, and display results which are at par or outperform the computable competitor Chapel et al. [2020].

2 Unbalanced Gromov-Wasserstein formulations

We present in this section our two new formulations and their properties. The first one, called UGW, is exploited in Sections 3 and 4 to derive an efficient algorithm used in numerical experiments. The second one, called CGW, defines a distance between mm-spaces up to isometries. Those results build upon the work of Liero et al. [2015], and a summary of the construction of UOT is detailed in Appendix A. In all what follows, we consider complete separable mm-spaces endowed with a metric and a positive measure.

2.1 The unbalanced Gromov-Wasserstein divergence

This new formulation makes use of quadratic φ-divergences, defined as $\mathrm{D}_\varphi^{\otimes}(\rho|\nu) \triangleq \mathrm{D}_\varphi(\rho \otimes \rho | \nu \otimes \nu)$, where $\rho \otimes \rho \in \mathcal{M}_+(X^2)$ is the tensor product measure defined by $\mathrm{d}(\rho \otimes \rho)(x, y) = \mathrm{d}\rho(x)\mathrm{d}\rho(y)$. Note that $\mathrm{D}_\varphi^{\otimes}$ is not a convex function in general.

Definition 1 (Unbalanced GW)**.** *The Unbalanced Gromov-Wasserstein divergence is defined as* $\mathrm{UGW}(\mathcal{X},\mathcal{Y}) = \inf_{\pi\in\mathcal{M}^+(X\times Y)} \mathcal{L}(\pi) \triangleq \mathcal{G}(\pi) + \mathrm{D}_\varphi^\otimes(\pi_1|\mu) + \mathrm{D}_\varphi^\otimes(\pi_2|\nu)$.

This definition can be understood as an hybridation between (1) and (2) but with a twist: one needs to use the quadratic divergence $\mathrm{D}_\varphi^\otimes$ in place of D_φ. To the best of our knowledge, it is the first time such quadratic divergences are being used and studied. In the TV case, this is the most important distinction between UGW and partial GW [Chapel et al., 2020]. Note also that the balanced GW distance (2) is recovered as a particular case when using $\varphi = \iota_{\{1\}}$ or by letting $\rho \to +\infty$ for an entropy $\psi = \rho\varphi$.

Using quadratic divergences results in UGW being 2-homogeneous: for $\theta \geq 0$, writing $(\mathcal{X}_\theta, \mathcal{Y}_\theta)$ equiped with $(\theta\mu, \theta\nu)$, one has $\theta^{-2}\mathrm{UGW}(\mathcal{X}_\theta, \mathcal{Y}_\theta) = \mathrm{UGW}(\mathcal{X},\mathcal{Y})$. When using non tensorized φ-divergences, the resulting unbalanced Gromov-Wassertein functional between $\mathcal{X}_\theta$ and $\mathcal{Y}_\theta$ have very different and inconsistent behaviors when $\theta \to 0$ and $\theta \to +\infty$. Indeed, once normalized by θ^{-2} and θ^{-1}, one obtains respectively balanced GW and a Hellinger-type distance. Using tensorized divergences ensure that the behavior does not depends on θ.

We first prove the existence of optimal plans π minimizing $\mathcal{L}$, which holds for the three key settings of Section 2.2.1, namely for KL, TV, and for compact metric spaces (such as finite pointclouds and graphs). All proofs are deferred in Appendix B.

Proposition 1 (Existence of minimizers)**.** *We assume that* (X, Y) *are compact and that either (i)* φ *superlinear, i.e* $\varphi'_\infty = \infty$*, or (ii)* λ *has compact sublevel sets in* $\mathbb{R}_+$ *and* $2\varphi'_\infty + \inf\lambda > 0$*. Then there exists* $\pi \in \mathcal{M}_+(X \times Y)$ *such that* $\mathrm{UGW}(\mathcal{X},\mathcal{Y}) = \mathcal{L}(\pi)$.

The following proposition ensures that the functional UGW can be used to compare mm-spaces.

Proposition 2 (Definiteness of UGW)**.** *Assume that* $\varphi^{-1}(\{0\}) = \{1\}$ *and* $\lambda^{-1}(\{0\}) = \{0\}$*. Then* $\mathrm{UGW}(\mathcal{X},\mathcal{Y}) \geq 0$ *and is* 0 *if and only if* $\mathcal{X} \sim \mathcal{Y}$.

We end this section with a reformulation of UGW which is important to make the connection with the second formulation of the following section. Its proof is deferred to Appendix B. It splits UGW into two parts: the term $\varphi(0)(|(\mu\otimes\mu)^\perp| + |(\nu\otimes\nu)^\perp|)$ accounts for the pure creation/destruction of mass and a new transport cost L_c accounts for the remaining part (partial/pure transport and partial creation/destruction of mass).

Lemma 1. *Defining* $L_c(a,b) \triangleq c + a\varphi(1/a) + b\varphi(1/b)$*, and writing* $(f \triangleq \frac{\mathrm{d}\mu}{\mathrm{d}\pi_1}, g \triangleq \frac{\mathrm{d}\nu}{\mathrm{d}\pi_2})$ *the Lebesgue densities of* (μ,ν) *w.r.t.* (π_1,π_2) *such that* $\mu = f\pi_1 + \mu^\perp$ *and* $\nu = g\pi_2 + \nu^\perp$*, one has*

$$\mathcal{L}(\pi) = \int_{X^2\times Y^2} L_{\lambda(|d_X - d_Y|)}(f\otimes f, g\otimes g)\mathrm{d}\pi\mathrm{d}\pi + \varphi(0)(|(\mu\otimes\mu)^\perp| + |(\nu\otimes\nu)^\perp|). \tag{3}$$

2.2 The conic Gromov-Wasserstein distance

We introduce a second "conic" formulation of unbalanced GW, which is connected to UGW, and whose construction is inspired by the conic formulation of UOT (see Appendix A for an overview).

2.2.1 Background on cone sets and distances

The conic formulation lifts a point $x \in X$ to a couple $(x, r) \in X \times \mathbb{R}^+$ where r encodes some (power of a) mass. Then we seek optimal transport plans defined over $\mathfrak{C}[X] \triangleq X \times \mathbb{R}_+/(X \times \{0\})$, where coordinates $(x, r = 0)$ with no mass are merged into a single point $\mathfrak{o}_X$ called the apex of the cone. In the sequel, points of $X \times \mathbb{R}_+$ are noted (x, r), while $[x, r]$ are quotiented points of $\mathfrak{C}[X]$.

While transport plans depend on variables $([x,r],[y,s])$ and $([x',r'],[y',s'])$ in $\mathfrak{C}[X]\times\mathfrak{C}[Y]$, the transportation cost involved in our conic formulation only makes use of the 2-D cone $\mathfrak{C}[\mathbb{R}_+]$ over $\mathbb{R}_+$ endowed with the distance $|u-v|$ (note that any other distance on $\mathbb{R}$ could be used as well). More specifically, we consider coordinates of the form $([u,a],[v,b]) = ([d_X(x,x'), rr'], [d_Y(y,y'), ss']) \in \mathfrak{C}[\mathbb{R}_+] \times \mathfrak{C}[\mathbb{R}_+]$. Thus we now describe conic discrepancies $\mathcal{D}$ on $\mathfrak{C}[\mathbb{R}_+]$, which are defined for $(p,q) \geq 1$ as $\mathcal{D}([u,a],[v,b])^q \triangleq H_{\lambda(|u-v|)}(a^p, b^p)$, where $H_c(a^p, b^p) \triangleq \inf_{\theta\geq 0} \theta L_c(\frac{a^p}{\theta}, \frac{b^p}{\theta})$ is the perspective transform of L_c introduced in Lemma 1. The intuition underpinning the definition of this cost is that the perspective transform accounts for the possibility to rescale a transport plan π by a scalar θ but the scaling is performed pointwise instead of globally. In general $\mathcal{D}$ is not a distance, but it is always definite as stated by this result proved in Appendix A.

Proposition 3. *Assume $\lambda^{-1}(\{0\}) = \{0\}$, $\varphi^{-1}(\{0\}) = \{1\}$ and φ is coercive. Then $\mathcal{D}$ is definite on $\mathfrak{C}[\mathbb{R}^+]$, i.e. $\mathcal{D}([u,a],[v,b]) = 0$ if and only if $(a = b = 0)$ or $(a = b$ and $u = v)$.*

Of particular interest are those φ where $\mathcal{D}$ is a distance, which necessitates a careful choice of λ, p and q. We now detail three examples where this is the case.

Gaussian Hellinger distance (GH). When $\mathrm{D}_\varphi = \mathrm{KL}$, $\lambda(t) = t^2$ and $q = p = 2$, then one has $\mathcal{D}([u,a],[v,b])^2 = a^2 + b^2 - 2abe^{-|u-v|/2}$. This cone distance [Burago et al., 2001] is further generalized by De Ponti [2019] who shows that $\mathcal{D}$ is a distance for power entropies $\varphi(s) = \frac{s^p - p(s-1) - 1}{p(p-1)}$ if $p \geq 1$ (the case $p = 1$ corresponding to $\mathrm{D}_\varphi = \mathrm{KL}$).

Hellinger-Kantorovich (HK) / Wasserstein-Fisher-Rao distance (WFR). When $\mathrm{D}_\varphi = \mathrm{KL}$, $\lambda(t) = -\log\cos^2(t \wedge \frac{\pi}{2})$ and $q = p = 2$, then one has $\mathcal{D}([u,a],[v,b])^2 = a^2 + b^2 - 2ab\cos(\frac{\pi}{2} \wedge |u - v|)$. This construction, which might seem peculiar, corresponds to the one used to make unbalanced OT a geodesic distance, as detailed in [Liero et al., 2015, Chizat et al., 2018a].

Partial optimal transport distance (PT). When $\mathrm{D}_\varphi = \mathrm{TV}$, $\lambda(t) = t^q$, $q \geq 1$ and $p = 1$, then $\mathcal{D}([u,a],[v,b])^q = a + b - (a \wedge b)(2 - |u - v|^q)_+$ defines a cone distance [Chizat et al., 2018a].

2.2.2 Definitions and properties

The conic formulation consists in solving a GW problem on the cone, with the addition of two linear constraints. Informally speaking, L_c from Lemma 1 becomes $\mathcal{D}$, the term $(|(\mu \otimes \mu)^\perp| + |(\nu \otimes \nu)^\perp|)$ is taken into account by the constraints (5) below, and the variables (f, g) are replaced by (r^p, s^p). It reads $\mathrm{CGW}(\mathcal{X}, \mathcal{Y}) \triangleq \inf_{\alpha \in \mathcal{U}_p(\mu,\nu)} \mathcal{H}(\alpha)$ where

$$\mathcal{H}(\alpha) \triangleq \int \mathcal{D}([d_X(x,x'), rr'], [d_Y(y,y'), ss'])^q \, \mathrm{d}\alpha([x,r],[y,s]) \mathrm{d}\alpha([x',r'],[y',s']), \tag{4}$$

and $\mathcal{U}_p(\mu,\nu)$ is defined as the set

$$\mathcal{U}_p(\mu,\nu) \triangleq \left\{ \alpha \in \mathcal{M}_+(\mathfrak{C}[X] \times \mathfrak{C}[Y]), \int_{\mathbb{R}_+} r^p \mathrm{d}\alpha_1(\cdot, r) = \mu, \int_{\mathbb{R}_+} s^p \mathrm{d}\alpha_2(\cdot, s) = \nu \right\}. \tag{5}$$

It is similar to the conic formulation of UW, see Appendix A. Note that similarly to the GW formulation (2) – and in sharp contrast with the conic formulation of UW – here the transport plans are defined on the cone $\mathfrak{C}[X] \times \mathfrak{C}[Y]$ but the cost $\mathcal{D}$ is a distance on $\mathfrak{C}[\mathbb{R}_+]$.

We present now the main contributions of this paper, proved in Appendix C. We state that CGW defines a distance under conditions that hold for the settings of Section 2.2.1, and that it is upper-bounded by UGW. The divergene UGW can be approximated with efficient numerical schemes as detailed in Section 3.

Theorem 1. *(i) The divergence* CGW *is symmetric, positive and definite up to isometries. (ii) If $\mathcal{D}$ is a distance on $\mathfrak{C}[\mathbb{R}_+]$, then* $\mathrm{CGW}^{1/q}$ *is a distance on the set of mm-spaces up to isometries. (iii) For any $(\mathrm{D}_\varphi, \lambda, p, q)$ with associated cost $\mathcal{D}$ on the cone, one has* $\mathrm{UGW} \geq \mathrm{CGW}$.

3 Algorithms

We focus in this section on the numerical computation of the upper bound UGW using a bi-convex relaxation and derive an alternate minimization scheme coupled with entropic regularization. We also propose to approximate CGW by doing a similar alternate minimization, as detailed in Appendix E. For both computations we provide guarantees of tightness on the bi-convex relaxation (see Theorem 2). The computation of the distance CGW is heavy in practice because it requires an optimization over a lifted conic space, which needs to be discretized. Thus it does not scale to large problem for CGW, but allows to explore numerically how tight is the upper bound $\mathrm{UGW} \geq \mathrm{CGW}$, see Section 4. The algorithm for UGW is presented on arbitrary measures, the special case of discrete measures being a particular case. The discretized formulas and algorithms are detailed in Appendix D, see also Chizat et al. [2018b], Peyré et al. [2016]. All implementations are available at `https://github.com/anonymous-conference-submission`.

3.1 Bi-convex relaxation and tightness

In order to derive a simple numerical approximation scheme, following Mémoli [2011], we introduce a lower bound obtained by introducing two transportation plans. To further accelerate the method and enable GPU-friendly iterations, similarly to Gold et al. [1996], Solomon et al. [2016], we consider an entropic regularization. It reads, for any $\varepsilon \geq 0$,

$$\mathrm{UGW}_\varepsilon(\mathcal{X},\mathcal{Y}) \triangleq \inf_\pi \mathcal{L}(\pi) + \varepsilon \mathrm{KL}^\otimes(\pi|\mu\otimes\nu) \geq \inf_{\pi,\gamma} \mathcal{F}(\pi,\gamma) + \varepsilon \mathrm{KL}(\pi\otimes\gamma|(\mu\otimes\nu)^{\otimes 2}), \tag{6}$$

$$\text{and} \quad \mathcal{F}(\pi,\gamma) \triangleq \int_{X^2\times Y^2} \lambda(|d_X - d_Y|)\mathrm{d}\pi\otimes\gamma + \mathrm{D}_\varphi(\pi_1\otimes\gamma_1|\mu\otimes\mu) + \mathrm{D}_\varphi(\pi_2\otimes\gamma_2|\nu\otimes\nu)\,,$$

where (γ_1,γ_2) denote the marginals of the plan γ. In the sequel we write $\mathcal{F}_\varepsilon = \mathcal{F} + \varepsilon\mathrm{KL}^\otimes$. Note that in contrast to the entropic regularization of GW Peyré et al. [2016], here we use a tensorized entropy to maintain the overall homogeneity of the energy. A simple method to approximate this lower bound is to perform an alternate minimization on π and γ, which is known to converge for smooth φ to a stationary point since the coupling term in the functional is smooth [Tseng, 2001].

Note that if $\pi\otimes\gamma$ is optimal then so is $(s\pi)\otimes(\frac{1}{s}\gamma)$ with $s\geq 0$. Thus without loss of generality we optimize under the constraint $m(\pi) = m(\gamma)$ by setting $s = \sqrt{m(\gamma)/m(\pi)}$. Such assumption allows to prove that the bi-convex relaxation is tight. We provide in Appendix D a general result holding for a wide range of quadratic programs, but we state here its application in the setting of UGW_ε.

Theorem 2 (Tightness for (U)GW$_\varepsilon$)**.** *For any $\varepsilon \geq 0$, for Balanced-GW or for KL-UGW under the constraint $m(\pi) = m(\gamma)$, if the kernel $\lambda(|d_X - d_Y|)$ is negative semi-definite, then minimizing the lower bound* (6) *is equivalent to minimizing the functional defining* $\mathrm{UGW}_\varepsilon(\mathcal{X},\mathcal{Y})$*, i.e. the relaxation is tight. More precisely, all minimizers (π_*,γ_*) of* (6) *satisfy $\mathcal{F}_\varepsilon(\pi_*,\gamma_*) = \mathcal{F}_\varepsilon(\pi_*,\pi_*) = \mathcal{F}_\varepsilon(\gamma_*,\gamma_*)$.*

The proof of Theorem 2 is in Appendix D. It is based on a generalization of Konno's result [Konno, 1976] combined with a decomposition formula of $\mathrm{KL}^\otimes$. Konno's result applies for unregularized, Balanced-GW. What is new is its extension to both GW and UGW, in presence of an entropic regularization.

The kernel $|d_X - d_Y|^2$ is negative semi-definite when both (d_X, d_Y) are conditionally negative kernels. Such property does not hold in general (e.g. for graph geodesic distances) and the tightness of the relaxation remains open in this setting. It holds for instance with tree metrics in the case of graphs, or for Euclidean, spherical, hyperbolic distances over their respective manifolds Feragen et al. [2015]. In practice we observed in the indefinite setting that the relaxation could fall into spurious minima when ε is small, while it happens much less frequently in the negative semi-definite setting.

3.2 Alternate Sinkhorn minimization

Minimizing the lower bound (6) with respect to either π or γ is non-trivial for an arbitrary φ. We restrict our attention to the Kullback-Leibler case $\mathrm{D}_\varphi = \rho\mathrm{KL}$ with $\rho > 0$, which can be addressed by solving a regularized and convex unbalanced problem as studied in Chizat et al. [2018b], Séjourné et al. [2019]. It is explained in the following proposition.

Proposition 4. *For a fixed γ, the optimal $\pi \in \arg\min_\pi \mathcal{F}(\pi,\gamma) + \varepsilon\mathrm{KL}(\pi\otimes\gamma|(\mu\otimes\nu)^{\otimes 2})$ is the solution of*

$$\min_\pi \int c^\varepsilon_\gamma(x,y)\mathrm{d}\pi(x,y) + \rho m(\gamma)\mathrm{KL}(\pi_1|\mu) + \rho m(\gamma)\mathrm{KL}(\pi_2|\nu) + \varepsilon m(\gamma)\mathrm{KL}(\pi|\mu\otimes\nu),$$

where $m(\gamma) \triangleq \gamma(X\times Y)$ is the mass of γ, and where we define the cost associated to γ as

$$c^\varepsilon_\gamma(x,y) \triangleq \int \lambda(|d_X(x,\cdot) - d_Y(y,\cdot)|)\mathrm{d}\gamma + \rho\int \log(\frac{\mathrm{d}\gamma_1}{\mathrm{d}\mu})\mathrm{d}\gamma_1 + \rho\int \log(\frac{\mathrm{d}\gamma_2}{\mathrm{d}\nu})\mathrm{d}\gamma_2 + \varepsilon\int \log(\frac{\mathrm{d}\gamma}{\mathrm{d}\mu\mathrm{d}\nu})\mathrm{d}\gamma.$$

Computing the cost c^ε_γ for spaces X and Y of n points has in general a cost $O(n^4)$ in time and memory. However, as explained for instance in Peyré et al. [2016], for the special case $\lambda(t) = t^2$, this cost is reduced to $O(n^3)$ in time and $O(n^2)$ in memory. This is the setting we consider in the numerical simulations. This makes the method applicable for scales of the order of 10^4 points. For larger datasets one should use approximation schemes such as hierarchical approaches [Xu et al., 2019] or Nyström compression of the kernel [Altschuler et al., 2018].

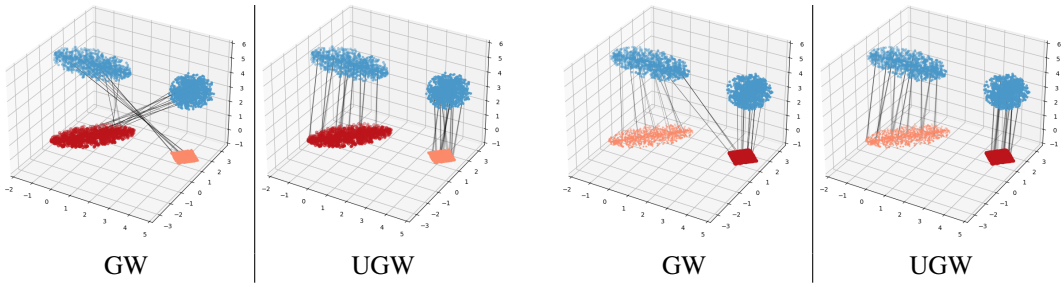

Figure 1: GW vs. UGW transportation plan, using $\nu = 0.3\mathcal{E}_2 + 0.7\mathcal{C}$ on the left, and $\nu = 0.7\mathcal{E}_2 + 0.3\mathcal{C}$ on the right. The 2D mm-spaces is lifted into $\mathbb{R}^3$ by padding the third coordinate to zero.

The resulting alternate minimization method is detailed in Algorithm 1, see Appendix D for a discretized version. It uses the unbalanced Sinkhorn algorithm of Chizat et al. [2018b], Séjourné et al. [2019] as sub-iterations and is initialized using $\pi = \mu \otimes \nu / \sqrt{m(\mu)m(\nu)}$. This Sinkhorn algorithm operates over a pair of continuous functions (so-called Kantorovitch potentials) $f(x)$ and $g(y)$. For discrete spaces X and Y of size n, these functions are stored in vectors of size n, and that integral involved in the updates becomes a sum. Each iteration of Sinkhorn thus has a cost n^2, and all the involved operation can be efficiently mapped to parallelizable GPU routines as detailed in Chizat et al. [2018b], Séjourné et al. [2019]. Another advantage of using an unbalanced Sinkhorn algorithm is its complexity $O(n^2/\varepsilon)$ to compute an ε-approximation, as stated in Pham et al. [2020], which should be compared to $O(n^2/\varepsilon^2)$ operations for balanced Sinkhorn.

Algorithm 1 – UGW($\mathcal{X}$, $\mathcal{Y}$, ρ, ε)

Input: mm-spaces $(\mathcal{X}, \mathcal{Y})$, relax. ρ, regul. ε
Output: π, γ solving (6)

Init. $\pi = \gamma = \mu \otimes \nu / \sqrt{m(\mu)m(\nu)}$, $g = 0$.
while (π, γ) has not converged **do**
  Update $\pi \leftarrow \gamma$,
  then $c \leftarrow c^{\varepsilon}_{\pi}$, $\tilde{\rho} \leftarrow m(\pi)\rho$, $\tilde{\varepsilon} \leftarrow m(\pi)\varepsilon$
  while (f, g) has not converged **do**
    $f \leftarrow -\frac{\tilde{\varepsilon}\tilde{\rho}}{\tilde{\varepsilon}+\tilde{\rho}} \log \int e^{(g(y)-c(\cdot,y))/\tilde{\varepsilon}} \mathrm{d}\nu(y)$
    $g \leftarrow -\frac{\tilde{\varepsilon}\tilde{\rho}}{\tilde{\varepsilon}+\tilde{\rho}} \log \int e^{(f(x)-c(x,\cdot))/\tilde{\varepsilon}} \mathrm{d}\mu(x)$
  end while
  Upd. $\gamma(x,y) \leftarrow e^{\frac{f(x)+g(y)-c(x,y)}{\tilde{\varepsilon}}} \mu(x)\nu(y)$
  Rescale $\gamma \leftarrow \sqrt{m(\pi)/m(\gamma)}\gamma$
end while
Return (π, γ).

Note also that balanced GW is recovered as a special case when setting $\rho \to +\infty$, so that $\tilde{\rho}/(\tilde{\varepsilon} + \tilde{\rho}) \to 1$ should be used in the iterations. In order to speed up Sinkhorn inner-loops, especially for small values of ε, one can use linear extrapolation [Thibault et al., 2017] or non-linear Anderson acceleration [Scieur et al., 2016].

There is an extra scaling step after computing γ involving the mass $m(\pi)$. It corresponds to the scaling s of $\pi \otimes \gamma$ such that $m(\pi) = m(\gamma)$, and we observe that this scaling is key not only to impose this mass equality but also to stabilize the algorithm. Otherwise we observed that $m(\gamma) < 1 < m(\pi)$ and underflows whenever $m(\gamma) \to 0$ and $m(\pi) \to \infty$.

4 Numerical experiments

This section presents simulations on synthetic examples to highlight the qualitative behavior of UGW and the tightness of the bound UGW $\geq$ CGW. Other illustrations on UGW are available in Appendix E. We end the section with a learning application of UGW in a positive-unlabeled setting, using domain adaptation data so as to compare with PGW Chapel et al. [2020]. In the synthetic experiments, μ and ν are probability distributions, which allows us to compare GW with UGW.

Robustness to imbalanced classes. In this first example, we take $X = \mathbb{R}^3$, $Y = \mathbb{R}^2$ and consider $\mathcal{E}_2$, $\mathcal{E}_3$, $\mathcal{C}$ and $\mathcal{S}$ to be uniform distributions on a 2D and 3D ellipse, a square and a sphere. We consider mm-spaces of different dimensions to emphasize the ability of (U)GW to compare different spaces. Figure 1 contrasts the transportation plan obtained by GW and UGW for a fixed $\mu = 0.5\mathcal{E}_3 + 0.5\mathcal{S}$ and ν obtained using two different mixtures of $\mathcal{E}_2$ and $\mathcal{C}$. The black segments show the largest entries of the transportation matrix π, for a sub-sampled set of points (to ease visibility), thus effectively displaying the matching induced by the plan. Furthermore, the width of the dots are scaled according to the mass of the marginals $\pi_1 \approx \mu$ and $\pi_2 \approx \nu$, i.e. the smaller the point, the smaller is the amount

of transported mass. This figure shows that the exact conservation of mass imposed by GW leads to a poor geometrical matching of the shapes which have different global mass. As this should be expected, UGW recovers coherent matchings. We suspect the alternate minimization algorithm is able to find the global minimum in these cases.

Tightness of the bound CGW≤UGW We propose to approximate CGW by doing a similar alternate minimization as for UGW, as detailed in Appendix E. This numerical scheme does not scale to large problems, but allows us to explore numerically how tight is the upper bound UGW $\geq$ CGW. Figure 2 highlights the fact that in Euclidean space $X = Y = \mathbb{R}^d$, this bound seems to be tight when the two measures are sufficiently close. We consider discrete measures $\mu = \frac{1}{n}\sum_i \delta_{x_i}$ in $X = Y = \mathbb{R}^d$ and $\nu_t = \frac{1}{n}\sum_i \delta_{y_i}$ where $y_i = x_i + t\Delta_i$ where Δ_i are random perturbations and denote $(\mathcal{X}, \mathcal{Y}_t)$ the two mm-spaces associated to the Euclidean distance. As $t \to 0$, μ and ν_t get closer, and numerically UGW≈CGW. Figure 3 considers random points $(x_i)_i$ and $(y_i)_i$ and displays the histograms of the ratio CGW/UGW for $n = 3$. This shows that while the bound CGW≤UGW seems not tight, the ratio appears to be bounded even for points not being close. This numerical experiment suggests that UGW and CGW are locally equivalent and that UGW is in practice an acceptable proxy of the distance CGW.

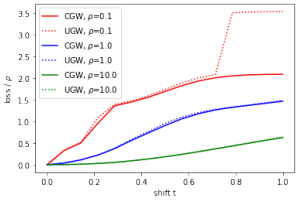

Figure 2: Comparison of UGW$(\mathcal{X}, \mathcal{Y}_t)$ and CGW$(\mathcal{X}, \mathcal{Y}_t)$ as the support gets shifted by a perturbation.

Positive unlabeled learning experiments Positive Unlabeled (PU) learning is a semi-supervised classification problem, where instead of learning from positive and negative samples $(x_i, \ell_i)_i$ with labels $\ell_i \in \{-1, 1\}$ we only learn from one class labeled with positives, i.e. only those $X \triangleq \{x_i \ : \ \ell_i = 1\}$. The task is to leverage X to predict the classes $\ell = \ell(y) \in \{-1, +1\}$ of unlabelled $y \in Y$ belong to a separate space. We consider here that X, Y are embedded in Euclidean space, and denote $\mathcal{X}, \mathcal{Y}$ the associated labelled and unlabelled mm-spaces, equipped with the uniform distribution. Our experiments are adapted from Partial-GW (PGW) Chapel et al. [2020], which used partial GW to solve PU-learning. The rationale of using unbalanced OT methods for PU learning stems from the fact that positive samples should be matched with positive due to their similar features, while negative samples would be ignored due to dissimilar features that induce a laziness to transport mass and match them.

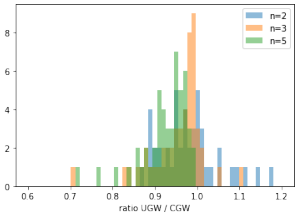

Figure 3: Histograms of the ratio CGW / UGW for random spaces with $n \in \{2, 3, 5\}$ samples. Ratios over 1 are due to local minima.

We consider PU learning over the Caltech office dataset used for domain adaptation tasks (with domains Caltech (C) Griffin et al. [2007], Amazon (A), Webcam (W) and DSLR (D) Saenko et al. [2010]). The Caltech datasets are represented with two embeddings based on Surf and Decaf features Saenko et al. [2010], Donahue et al. [2014]. On the latter datasets, we perform PU learning over similar features (e.g. surf-C $\to$ surf-* or decaf-C $\to$ decaf-*) and from one feature format to the other (e.g. surf-C $\to$ decaf-* or surf-C $\to$ decaf-*). Those features are projected via PCA to subspaces of dimension 10 for surf features and 40 for decaf features. In the last task, one cannot use standard PU-method, and to the best of our knowledge, Unbalanced-GW methods are the only approaches for PU learning across different domains/features.

The procedure is the following. We solve the PU learning problem by computing the optimal plan π for UGW$(\mathcal{X}, \mathcal{Y})$. We compute its first marginal π_2 on $\mathcal{Y}$, and predict the labels of some $y \in Y$ as $\ell(y) \triangleq \text{sign}(\pi_2(y) - q)$ where q is the quantile of π_2 corresponding to the proportion r of positives samples in Y. Following Chapel et al. [2020] which is adapted from Kato et al. [2018], Hsieh et al. [2019], this proportion r is assumed to be known. We report the accuracy of the prediction over the same 20 folds of the datasets, and use 20 other folds to validate the parameters of UGW. We consider 100 random samples for each fold of (X, Y), a ratio of positive samples $r = 0.1$ for domains (C,A,W,D), and a ratio $r = 0.2$ for domains (C,A,W).

Since the GW objective is non-convex, the initialization of the minimization algorithms is key to obtain good performances. In Chapel et al. [2020] and our experiments, for tasks where Y and Y belong to the same Euclidean space, (e.g. surf-C $\to$ surf-*) we initialize π with the Partial-Wasserstein

Dataset	prior	Init (PW)	PGW	**UGW**	Dataset	prior	Init (FLB)	PGW	**UGW**
surf-C → surf-C	0.1	**89.9**	84.9	83.9	surf-C → decaf-C	0.1	85.0	85.1	**85.6**
surf-C → surf-A	0.1	81.8	82.2	**83.5**	surf-C → decaf-A	0.1	84.2	**87.1**	83.6
surf-C → surf-W	0.1	**81.9**	81.3	80.3	surf-C → decaf-W	0.1	86.2	**88.6**	86.8
surf-C → surf-D	0.1	80.0	81.4	**83.2**	surf-C → decaf-D	0.1	84.7	**91.1**	90.7
surf-C → surf-C	0.2	**79.7**	75.7	75.4	surf-C → decaf-C	0.2	74.8	75.6	**75.9**
surf-C → surf-A	0.2	65.6	66.0	**76.4**	surf-C → decaf-A	0.2	76.2	**87.9**	82.4
surf-C → surf-W	0.2	65.1	64.3	**67.3**	surf-C → decaf-W	0.2	81.5	88.4	**89.9**
decaf-C → decaf-C	0.1	**93.9**	83.0	86.8	decaf-C → surf-C	0.1	**81.7**	81.0	81.1
decaf-C → decaf-A	0.1	80.1	81.4	**85.6**	decaf-C → surf-A	0.1	80.9	81.2	**82.4**
decaf-C → decaf-W	0.1	80.1	82.7	**86.1**	decaf-C → surf-W	0.1	82.0	81.3	**83.5**
decaf-C → decaf-D	0.1	80.6	**83.8**	83.4	decaf-C → surf-D	0.1	80.0	80.8	**81.5**
decaf-C → decaf-C	0.2	**90.6**	76.7	80.5	decaf-C → surf-C	0.2	**66.6**	63.7	65.2
decaf-C → decaf-A	0.2	62.5	68.7	**74.7**	decaf-C → surf-A	0.2	62.9	62.4	**69.3**
decaf-C → decaf-W	0.2	65.7	75.9	**79.2**	decaf-C → surf-W	0.2	65.1	61.4	**83.3**

Table 1: Accuracy for all tasks. The left block are domain adaptation experiments with similar features, where both PGW and UGW are initialised with PW. The right block are domain adaptation experiments with different features, and the reported init is FLB (see Appendix E) used for UGW.

(PW) solution with a squared Euclidean cost. For cross-domain prediction (e.g. surf-C → decaf-*), following Chapel et al. [2020], PGW is initialized with a list of plans built using a coarsened representation of the data with k-NN. While Chapel et al. [2020] makes use in an oracle manner of the plan providing the best accuracy, we modified their protocol and keep the plan which has the lowest PGW cost, which seems fairer, hence the difference in performance with Chapel et al. [2020]. To initialize UGW when X and Y do not belong to the same Euclidean space, we use a UOT solution of a matching between distance histograms called FLB Mémoli [2011], as detailed in Appendix E.

In the experiments we slightly generalize UGW and use two different marginal penalties $\rho_1 \mathrm{KL}^{\otimes}(\pi_1|\mu) + \rho_2 \mathrm{KL}^{\otimes}(\pi_2|\nu)$ with two parameters (ρ_1, ρ_2) to take into account shifts between domains/features. Note that PGW has a single parameter (which plays a role similar to (ρ_1, ρ_2)) which controls the cost of mass creation/destruction. We set $\varepsilon = 2^{-9}$, which avoids introducing an extra parameter in the method. The value $(\rho_1, \rho_2) \in \{2^{-k},\ k \in [\![5, 10]\!]\}^2$ are cross validated for each task on the validation folds, and we report the average accuracy on the testing folds. We discuss in Appendix E the impact of reducing the number of parameters on the performance. Comparison with other methods – PU and PUSB Kato et al. [2018], Du Plessis et al. [2014] – are provided in Chapel et al. [2020] and we focus here on the comparison with PGW only.

The results are reported in Table 3. We display the performance of PGW, UGW and the initialization used for UGW to guarantee that using UGW does improve the performance. We observe that when the source and target dataset is the same (C→C tasks), the PW initialization performs better and PGW/UGW degrade the performance, so that in this setting Optimal Transport should be preferred over GW, which is to be expected. However when the domains are different, applying UGW improves the performance over the initialization (which is FLB) in almost all tasks. Note that in that case the methods PU, PUSB or PW cannot be used. Overall, this shows that GW methods are able to solve to some extent the PU learning problem across different spaces, and that using a "softer" KL penalties in UGW is at least at par with Partial GW, and performs better in some settings.

5 Conclusion and perspectives

This paper defines two Unbalanced Gromov-Wasserstein formulations: CGW and UGW. We prove that they are both positive and definite. We provide a scalable, GPU-friendly algorithm to compute UGW illustrate its applicability in learning tasks, and show that CGW is a distance between mm-spaces up to isometry. These divergences and distances allow for the first time to blend in a seamless way the transportation geometry of GW with creation and destruction of mass. This hybridization is the key to unlock both theoretical and practical issues. This work opens new questions for future works, for instance removing the bias introduced by the use of entropic regularization, which is important for applications to ML. Note that such a debiasing was successfully applied for Balanced-GW in Bunne et al. [2019] and is shown to lead to a valid divergence for balanced OT in Feydy et al. [2019b] and UW in Séjourné et al. [2019]. The design of efficient numerical solvers for the the conic formulation is also an interesting avenue for future works.

References

Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. Generalized multidimensional scaling: a framework for isometry-invariant partial surface matching. *Proceedings of the National Academy of Sciences*, 103(5):1168–1172, 2006.

Hongteng Xu, Dixin Luo, and Lawrence Carin. Scalable gromov-wasserstein learning for graph partitioning and matching. In *Advances in neural information processing systems*, pages 3046–3056, 2019.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1263–1272. JMLR. org, 2017.

Edouard Grave, Armand Joulin, and Quentin Berthet. Unsupervised alignment of embeddings with wasserstein procrustes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1880–1890, 2019.

David Alvarez-Melis and Tommi S Jaakkola. Gromov-wasserstein alignment of word embedding spaces. *arXiv preprint arXiv:1809.00013*, 2018.

Karl-Theodor Sturm. The space of spaces: curvature bounds and gradient flows on the space of metric measure spaces. *arXiv preprint arXiv:1208.0434*, 2012.

Facundo Mémoli. Gromov–wasserstein distances and the metric approach to object matching. *Foundations of computational mathematics*, 11(4):417–487, 2011.

Matthias Liero, Alexander Mielke, and Giuseppe Savaré. Optimal entropy-transport problems and a new hellinger–kantorovich distance between positive measures. *Inventiones mathematicae*, pages 1–149, 2015.

Nicolas Courty, Rémi Flamary, and Devis Tuia. Domain adaptation with regularized optimal transport. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 274–289. Springer, 2014.

Charlie Frogner, Chiyuan Zhang, Hossein Mobahi, Mauricio Araya, and Tomaso A Poggio. Learning with a Wasserstein loss. In *Advances in Neural Information Processing Systems*, pages 2053–2061, 2015.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv preprint arXiv:1701.07875*, 2017.

Lénaïc Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. Unbalanced optimal transport: Dynamic and kantorovich formulations. *Journal of Functional Analysis*, 274(11): 3090–3123, 2018a.

Jean Feydy, Pierre Roussillon, Alain Trouvé, and Pietro Gori. Fast and scalable optimal transport for brain tractograms. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 636–644. Springer, 2019a.

John Lee, Nicholas P Bertrand, and Christopher J Rozell. Parallel unbalanced optimal transport regularization for large scale imaging problems. *arXiv preprint arXiv:1909.00149*, 2019.

Yogesh Balaji, Rama Chellappa, and Soheil Feizi. Robust optimal transport with applications in generative modeling and domain adaptation. *Advances in Neural Information Processing Systems*, 33, 2020.

Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in neural information processing systems*, pages 3036–3046, 2018.

Grant Rotskoff, Samy Jelassi, Joan Bruna, and Eric Vanden-Eijnden. Global convergence of neuron birth-death dynamics. *arXiv preprint arXiv:1902.01843*, 2019.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Adv. in Neural Information Processing Systems*, pages 2292–2300, 2013.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. Scaling algorithms for unbalanced transport problems. *to appear in Mathematics of Computation*, 2018b.

Thibault Séjourné, Jean Feydy, François-Xavier Vialard, Alain Trouvé, and Gabriel Peyré. Sinkhorn divergences for unbalanced optimal transport. *arXiv preprint arXiv:1910.12958*, 2019.

Charlotte Bunne, David Alvarez-Melis, Andreas Krause, and Stefanie Jegelka. Learning generative models across incomparable spaces. *arXiv preprint arXiv:1905.05461*, 2019.

Hongteng Xu, Dixin Luo, Ricardo Henao, Svati Shah, and Lawrence Carin. Learning autoencoders with relational regularization. *arXiv preprint arXiv:2002.02913*, 2020.

Ivegen Redko, Titouan Vayer, Rémi Flamary, and Nicolas Courty. Co-optimal transport. *arXiv preprint arXiv:2002.03731*, 2020.

Titouan Vayer, Laetita Chapel, Rémi Flamary, Romain Tavenard, and Nicolas Courty. Fused gromov-wasserstein distance for structured objects: theoretical foundations and mathematical properties. *arXiv preprint arXiv:1811.02834*, 2018.

Samir Chowdhury and Tom Needham. Gromov-wasserstein averaging in a riemannian framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 842–843, 2020.

David Alvarez-Melis, Stefanie Jegelka, and Tommi S Jaakkola. Towards optimal transport with global invariances. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1870–1879. PMLR, 2019.

Rainer E Burkard, Eranda Cela, Panos M Pardalos, and Leonidas S Pitsoulis. The quadratic assignment problem. In *Handbook of combinatorial optimization*, pages 1713–1809. Springer, 1998.

Gabriel Peyré, Marco Cuturi, and Justin Solomon. Gromov-wasserstein averaging of kernel and distance matrices. In *International Conference on Machine Learning*, pages 2664–2672, 2016.

Steven Gold and Anand Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Transactions on pattern analysis and machine intelligence*, 18(4):377–388, 1996.

Nicoló De Ponti and Andrea Mondino. Entropy-transport distances between unbalanced metric measure spaces. *arXiv preprint arXiv:2009.10636*, 2020.

Karl-Theodor Sturm et al. On the geometry of metric measure spaces. *Acta mathematica*, 196(1): 65–131, 2006.

Laetitia Chapel, Mokhtar Z Alaya, and Gilles Gasso. Partial gromov-wasserstein with applications on positive-unlabeled learning. *arXiv preprint arXiv:2002.08276*, 2020.

Alessio Figalli. The optimal partial transport problem. *Archive for rational mechanics and analysis*, 195(2):533–560, 2010.

Luca Cosmo, Emanuele Rodolà, Michael M Bronstein, Andrea Torsello, Daniel Cremers, and Y Sahillioglu. Shrec'16: Partial matching of deformable shapes. *Proc. 3DOR*, 2(9):12, 2016.

Emanuele Rodola, Alex M Bronstein, Andrea Albarelli, Filippo Bergamasco, and Andrea Torsello. A game-theoretic approach to deformable shape matching. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 182–189. IEEE, 2012.

Dmitri Burago, Iu D Burago, Yuri Burago, Sergei A Ivanov, and Sergei Ivanov. *A course in metric geometry*, volume 33. American Mathematical Soc., 2001.

Nicolò De Ponti. Metric properties of homogeneous and spatially inhomogeneous f-divergences. *IEEE Transactions on Information Theory*, 66(5):2872–2890, 2019.

Cedric Villani. *Topics in C. Transportation*. Graduate studies in Math. AMS, 2003.

Steven Gold, Anand Rangarajan, et al. Softmax to softassign: Neural network algorithms for combinatorial optimization. *Journal of Artificial Neural Networks*, 2(4):381–399, 1996.

J. Solomon, G. Peyré, V. Kim, and S. Sra. Entropic metric alignment for correspondence problems. *ACM Transactions on Graphics (TOG)*, 35(4), 2016.

Paul Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3):475–494, 2001.

Hiroshi Konno. Maximization of a convex quadratic function under linear constraints. *Mathematical programming*, 11(1):117–127, 1976.

Aasa Feragen, Francois Lauze, and Soren Hauberg. Geodesic exponential kernels: When curvature and linearity conflict. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3032–3042, 2015.

Jason Altschuler, Francis Bach, Alessandro Rudi, and Jonathan Weed. Massively scalable Sinkhorn distances via the nystr\" om method. *arXiv preprint arXiv:1812.05189*, 2018.

Khiem Pham, Khang Le, Nhat Ho, Tung Pham, and Hung Bui. On unbalanced optimal transport: An analysis of sinkhorn algorithm. *arXiv preprint arXiv:2002.03293*, 2020.

Alexis Thibault, Lénaïc Chizat, Charles Dossal, and Nicolas Papadakis. Overrelaxed Sinkhorn-Knopp algorithm for regularized optimal transport. *arXiv preprint arXiv:1711.01851*, 2017.

Damien Scieur, Alexandre d'Aspremont, and Francis Bach. Regularized nonlinear acceleration. In *Advances In Neural Information Processing Systems*, pages 712–720, 2016.

Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. 2007.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655. PMLR, 2014.

Masahiro Kato, Takeshi Teshima, and Junya Honda. Learning from positive and unlabeled data with a selection bias. In *International Conference on Learning Representations*, 2018.

Yu-Guan Hsieh, Gang Niu, and Masashi Sugiyama. Classification from positive, unlabeled and biased negative data. In *International Conference on Machine Learning*, pages 2820–2829. PMLR, 2019.

Marthinus C Du Plessis, Gang Niu, and Masashi Sugiyama. Analysis of learning from positive and unlabeled data. *Advances in neural information processing systems*, 27:703–711, 2014.

Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouvé, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2681–2690, 2019b.

Matthias Liero, Alexander Mielke, and Giuseppe Savaré. Optimal transport in competition with reaction: The hellinger–kantorovich distance and geodesic curves. *SIAM Journal on Mathematical Analysis*, 48(4):2869–2911, 2016.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. An interpolating distance between optimal transport and fisher–rao metrics. *Foundations of Computational Mathematics*, 18(1):1–44, 2018c.

Rémi Flamary and Nicolas Courty. Pot python optimal transport library. *GitHub: https://github.com/rflamary/POT*, 2017.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Due to the scalability of the method, it might have a limited immediate impact with respect to individual and societal aspects.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix B and D.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4 and Appendix E.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See standard deviation table in Appendix E.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix E.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A] Datasets are open source.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See the supplementary material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] The datasets used involve no individual data.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] The datasets used involve no individual data.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]