Information and Inference: A Journal of the IMA (2019) **00**, 1–28
doi:10.1093/imaiai/iaz008

Adaptive optimal transport

Montacer Essid[†]
Courant Institute of Mathematical Sciences, New York University, 251 Mercer St,
New York, NY 10012, USA
[†]Corresponding author. Email: essid@cims.nyu.edu

Debra F. Laefer
NYU Center of Urban Science and Progress, 370 Jay St, Brooklyn, NY 11201, USA

and

Esteban G. Tabak
Courant Institute of Mathematical Sciences, New York University, 251 Mercer St,
New York, NY 10012, USA

[Received on 29 June 2018; revised on 17 February 2019; accepted on 2 March 2019]

An adaptive, adversarial methodology is developed for the optimal transport problem between two distributions μ and ν, known only through a finite set of independent samples $(x_i)_{i=1..n}$ and $(y_j)_{j=1..m}$. The methodology automatically creates features that adapt to the data, thus avoiding reliance on *a priori* knowledge of the distributions underlying the data. Specifically, instead of a discrete point-by-point assignment, the new procedure seeks an optimal map $T(x)$ defined for all x, minimizing the Kullback–Leibler divergence between $(T(x_i))$ and the target (y_j). The relative entropy is given a sample-based, variational characterization, thereby creating an adversarial setting: as one player seeks to push forward one distribution to the other, the second player develops features that focus on those areas where the two distributions fail to match. The procedure solves local problems that seek the optimal transfer between consecutive, intermediate distributions between μ and ν. As a result, maps of arbitrary complexity can be built by composing the simple maps used for each local problem. Displaced interpolation is used to guarantee global from local optimality. The procedure is illustrated through synthetic examples in one and two dimensions.

Keywords: optimal transport; entropy; minimax.

1. Introduction

The optimal transport (OT) problem consists of finding, from among all transformations $y = T(x)$ that push forward a source distribution $\mu(x)$ to a target $\nu(y)$, the map that minimizes the expected transportation cost

$$\min_T \int c\,(x, T(x))\,\mu(x)\,\mathrm{d}x, \qquad T_\#\mu = \nu, \tag{1.1}$$

where $c(x, y)$ is the externally provided cost of moving a unit of mass from x to y [20]. The application for which Monge formulated the OT problem was the actual transportation of material between two sites at minimal cost [9]. Two centuries later, starting with Kantorovich and Koopmans [5], the problem was relaxed from maps to couplings, and applied to more general matching problems, such as matching supply and demand or positions and employees. More recently, the OT problem has become a central tool in many computer and data science applications, as well as in analysis and partial differential equations. Among the many applications for which OT could be used, the particular one that drove the methodology proposed in this article is change detection, for which one seeks a correspondence between two point clouds (from remote sensing data—either imagery or laser scanning) in order to identify differences between them.

The numerical solution of OT problems has been an active area of research for some years. When the two measures μ and ν have discrete support, the relaxation of OT due to Kantorovich [5] becomes a linear programming problem, which can be solved effectively for problems of small and medium size. When the size of the problem grows, its solution can be accelerated significantly through the addition of an entropic regularization and a Sinkhorn-type iterative algorithm [2,13]. This regularized problem, both in the discrete and the continuous versions, is equivalent to the Schrödinger bridge [1,7]. When the space underlying the two measures μ and ν is continuous and the distributions are known in closed form, one can—in small dimensional problems—discretize them on a grid or a graph before applying these techniques. Then their solution provides a point-by-point assignment between the source and the target measures.

However, in most data science applications, the distributions underlying the source and/or target samples are unknown. Moreover, those samples are often embedded in a high-dimensional space, and the data are relatively scarce. Density estimation techniques using this scarce data will yield a poor representation of the source and target measures. Hence, the transport map or transference plan provided by these techniques will be either inaccurate or highly overfitted, which leads to a very poor predictive power for the target of new sample points from the source.

In order to provide a more flexible framework for data science applications, sample-based techniques to solve the OT problem were developed in [6,18,19]. A central question to address when posing sample-based OT problems is the meaning of the push-forward condition $T_{\#}\mu = \nu$ when μ and ν are only known through samples $\left\{x_i\right\}$, $\{y_j\}$. In the formulations in [6,18,19], this condition was relaxed to the equality of the empirical means of a predetermined set of functions or 'features' over the two sample sets; a relaxation that appears naturally in the dual formulation of the problem. This raises the feature selection problem of finding the set of features best suited to each application. The associated challenges are particularly apparent in the change detection problem, where elements in two point clouds may differ for instance in size, colour, shape, data distribution or location, may be large or small, may have appeared, disappeared, have been displaced, deformed, broken, consolidated. . .Thus, the development of a robust, application-independent feature selection methodology is far from trivial.

The methodology proposed in this article incorporates feature selection into the formulation of the OT problem itself, through an adversarial approach. This involves three main steps:

1. Borrowing from the methodology developed in [6], we subdivide the transportation problem between μ and ν into finding N local maps T_t pushing forward ρ_{t-1} to ρ_t, with $\rho_0 = \mu$ and $\rho_N = \nu$. The global map T results from the composition of these local maps: $T = T_N \circ T_{N-1} \circ \ldots \circ T_1$, and global optimality is guaranteed by requiring that the ρ_t are McCann's displacement interpolants [8] between μ and ν. This decomposition achieves two goals:

   - Because every pair of successive ρ_t are close to each other, the corresponding maps T_t are close to the identity, which is the gradient of the strictly convex function $\frac{1}{2}\|x\|^2$. This permits relaxing the requirement that ϕ_t be convex in the optimality condition $T_t = \nabla \phi_t$ for the standard quadratic transportation cost.

- Arbitrarily complex maps T can be built through the composition of quite simple maps T_t. Thus, the maps over which to optimize each local problem can be reduced to a suitable family, depending on just a handful of parameters.

2. We formulate the push-forward condition $T_{t\#}\rho_{t-1} = \rho_t$ not in terms of the empirical expectation of features, but as the minimization of the relative entropy between $T_{t\#}\rho_{t-1}$ and ρ_t. One advantage of this formulation is that it is a natural relaxation of the push-forward condition when T_t is restricted to a small family of maps, which renders impossible the achievement of a perfect match between $T_{t\#}\rho_{t-1}$ and ρ_t.

3. We use a variational characterization of the relative entropy, as the maximizer of a suitable functional over functions $g(x)$. This formulation has three critical properties:

   (a) Since the variational characterization involves expected values of functions over ρ_{t-1} and ρ_t, it can be immediately extended to a sample-based scenario, thereby, replacing those expected values by empirical means.

   (b) Replacing 'all' functions $g(x)$ by a suitable family of functions provides a natural relaxation in the presence of finite sample sets. We show that, unlike the maps T_t, which produce the global map T via composition, it is the sum of the functions g_t that approximates the global g. Moreover, we prove that, if the families of T_t and g_t are built through the linear superposition of a predetermined set of functions, we recover the solution in [6].

   (c) Each local problem has now been given a minimax formulation (minimize over T, maximize over g.) This has a natural adversarial interpretation: while the 'player' with strategy T seeks to minimize the discrepancies between $T_{\#}\rho_{t-1}$ and ρ_t, its adversary with strategy g develops features to prove that the two distributions have not been perfectly matched. This provides the desired adaptability: the user does not need to provide features adapted to the problem in hand, as these will emerge automatically from its solution. This facilitates applications across a broad range of problems, including problems with significant features at various, possibly unknown scales.

This paper is organized as follows. After this introduction, Section 2 describes the methodology and its theoretical underpinning. Section 2.1 introduces the variational characterization of the relative entropy that the algorithm uses and concludes with the sample-based minimax formulation of the local OT problem. Section 2.2 shows that, when the functions g and potentials ϕ are drawn from finite-dimensional linear functional spaces, the solution to the problem agrees with the one obtained in [6] with predetermined features. Section 2.3 proves that the order of minimization and maximization does not matter—that is, that there is no duality gap—and explains the intuition behind the adversarial nature of the game, by detailing how each player reacts to the other's strategy. Section 2.4 integrates the local algorithm just described into a global algorithm for the full OT between μ and ν.

Section 3 details the algorithm further. Section 3.1 specifies the functional spaces chosen for g and ϕ, Section 3.2 the procedure used for solving the minimax problem and Section 3.3 the additional penalization terms required for the nonlinear components of the functional spaces. Finally, Section 4 performs some illustrative numerical experiments, applying the new methodology to synthetic low-dimensional data. The focus of these experiments is to display in action, in easy to visualize scenarios, the adversarial nature of the formulation.

2. Adaptive OT

2.1 *Formulation of the problem: an adversarial approach*

We are given two sample sets $(x_i)_{i=1,\ldots,n}$, $(y_j)_{j=1,\ldots,m} \subset \mathbb{R}^d$ with n and m sample points, respectively, independent realizations of two random variables with unknown distributions μ and ν. Both distributions are assumed to be absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}^d$ and have finite second-order moments. By a slight abuse of notation, we will identify the measures and their densities.

In this case, Brenier's theorem [20, p. 66] guarantees the existence of a map T pushing forward μ to ν and minimizing the transportation cost

$$\int \|T(x) - x\|^2 \, \mu(x) \, \mathrm{d}x. \tag{2.1}$$

From the samples provided, we seek a map T that would perform the transport well when applied to other independent realizations of the unknown distributions μ, ν. We can assume that the source and target distribution are close.

Remark Solving the problem for nearby distributions is the building block of a general procedure for arbitrary distributions and for finding the Wasserstein barycenter of distributions [6]. This more general procedure is presented in Section 2.4.

The OT problem has two main ingredients: the push-forward condition that $(T(x_i))$ and (y_j) have the same distribution and the minimization of the cost.

Remark For the quadratic cost, the optimal solution is the gradient of a convex function $\phi(x)$, $y = T(x) = \nabla\phi(x)$, a convenient characterization. More general cost functions of the type $\ell(x-y)$ would only require modifying $\nabla\phi$ into $x - \nabla\ell^*(\nabla\phi)$, where ℓ^* represents the Legendre–Fenchel transform of the strictly convex function ℓ, in the algorithm presented below.

In [6], the push-forward condition was formulated in terms of the equality of the empirical expected values of a predetermined set of feature functions. Instead, we propose a broader and adaptive formulation, in terms of the relative entropy between the two distributions. This introduces some significant improvements:

1. Of the two characterizations of equality of distributions, that all test functions within a broad enough class agree and that their relative entropy vanish, the latter is far more succinct and easier to enforce.
2. Replacing 'all' test functions by a finite set, though a sensible approximation in the presence of finite sample sizes leads to questions of robustness and feature selection. To address this, we will use a variational characterization of the relative entropy, which automatically selects the 'best' features within a given class.
3. For finite sample sets, one would expect the empirical expected values of test functions on the two distributions to agree only in a statistical sense, so requiring their strict equality is somewhat artificial. By contrast, in the new formulation, rather than requiring the relative entropy to vanish, which may be unrealistic for finite sample sizes and a limited family of maps T, we seek to minimize it.

Definition 1 For two probability measures $\rho, \nu \in P(\mathbb{R}^d)$, the *Kullback–Leibler divergence* of ρ with respect to ν—also called their relative entropy—is defined as

$$D_{KL}(\rho||\nu) = \int \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}\right) \mathrm{d}\rho \tag{2.2}$$

if ρ is absolutely continuous with respect to ν ($\rho \ll \nu$), and $+\infty$ otherwise.

Solving the OT problem is equivalent to minimizing a Kullback–Leibler divergence, as the following proposition shows.

Proposition 1 Let $\mu, \nu \in P(\mathbb{R}^d)$, with μ absolutely continuous with respect to the Lebesgue measure m on $\mathbb{R}^d$.

Let $\mathcal{C}$ be the set of convex functions from $\mathbb{R}^d \to \mathbb{R}$.

Define the minimization problem

$$\inf_{\phi\in\mathcal{C}} D_{KL}(\nabla\phi_{\#}\mu||\nu), \tag{KLopt}$$

where $\nabla\phi_{\#}\mu(A) = \mu((\nabla\phi)^{-1}(A))$,[1] , for any Borel measurable set A.

Then there exists a unique minimizer ϕ (up to zero measure sets), which coincides with the minimizer of the 2-Wasserstein distance between μ and ν:

$$\phi = \arg\inf_{\psi\in\mathcal{C}} D_{KL}\big(\nabla\psi_{\#}\mu||\nu\big)$$

and

$$W_2^2(\mu,\nu) = \int \left|\nabla\phi(x) - x\right|^2 \mathrm{d}\mu(x).$$

Proof. By Brenier's theorem, there exists a unique minimizer ϕ (up to zero measure sets) for the 2-Wasserstein problem. The potential ϕ is a proper lower semi-continuous convex function and $\nabla\phi_{\#}\mu = \nu$. One easily sees that ϕ is the minimizer for the Kullback–Leibler divergence optimization problem (KLopt), since for any measure $\rho \in P(\mathbb{R}^d)$ one has

$$D_{KL}(\rho||\nu) \geq 0 \tag{2.3}$$

with equality if and only if $\rho = \nu$ almost everywhere (in ν).

[1] $\nabla\phi$ is well-defined m-a.e. by Theorem 25.5 [14] and hence μ-a.e.

Inequality (2.3) is easy to prove; if ρ is not absolutely continuous w.r.t. ν, the Kullback–Leibler divergence is infinite, so the statement is true. Otherwise, we have

$$\begin{aligned} D_{KL}(\rho||\nu) &= \int \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}\right)\,\mathrm{d}\rho \\ &= \int \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}\right)\frac{\mathrm{d}\rho}{\mathrm{d}\nu}\,\mathrm{d}\nu \\ &\geq \left(\int \frac{\mathrm{d}\rho}{\mathrm{d}\nu}\mathrm{d}\nu\right)\log\left(\int \frac{\mathrm{d}\rho}{\mathrm{d}\nu}\,\mathrm{d}\nu\right) = 0, \end{aligned}$$

where we used Jensen's inequality and the convexity of $x \mapsto x\log(x)$.

So the equality $D_{KL}(\rho||\nu) = 0$ will hold if and only if Jensen's inequality becomes an equality, i.e. if and only if $\frac{\mathrm{d}\rho}{\mathrm{d}\nu} \equiv 1$, or $\rho = \nu$.

In particular, the solution to the OT problem satisfies $\nabla\phi_{\#}\mu = \nu$. Hence,

$$D_{KL}(\nabla\phi_{\#}\mu||\nu) = 0,$$

which shows that ϕ is a minimizer of the OT problem and (KLopt).

As for uniqueness, let $\phi_1, \phi_2 \in \mathcal{C}$ be two minimizers. Then $\nabla\phi_{1\#}\mu = \nabla\phi_{2\#}\mu = \nu$ from the statement above. By Brenier's theorem, they both solve the quadratic cost OT problem, which has a unique solution up to zero measure sets. □

Recently, there has been a push in machine learning to replace the Kullback–Leibler divergence by Wasserstein distances in order to penalize differences in data sets [4,13]. Unlike the Kullback–Leibler divergence, the Wasserstein distance defines a proper distance, enjoys regularity and symmetry properties and is computationally tractable. Nonetheless, the Kullback–Leibler divergence is well suited to measure the dissimilarities between measures that we are trying to detect. In particular, the asymmetry between the two measures under the Kullback–Leibler divergence is well within the spirit of the problem, as we seek a convex function ϕ that makes the transported distribution $\nabla\phi_{\#}\mu$ indistinguishable from the target reference ν. Also, as we shall see, the minimization of the relative entropy captures the differences between the two sample sets far more deftly than does a predefined finite set of test functions.

Thus, the biggest drawback in using the Kullback–Leibler divergence appears to be the difficulty in its numerical evaluation, particularly when we do not have access to a closed form expression for μ and ν, but merely to a finite set of independent samples from each of these distributions. One could resort to density estimation techniques [15,16] to approximate μ and ν, and then proceed to numerical integration. Instead, we use a variational characterization of the Kulback–Leibler divergence of ρ with respect to ν, in the form of a sample-friendly expression.

Proposition 2 Let $\rho, \nu \in P(\mathbb{R}^d)$. Then

$$D_{KL}(\rho||\nu) = 1 + \sup_{g}\left\{\int g\,\mathrm{d}\rho - \int \mathrm{e}^{g}\,\mathrm{d}\nu\right\}$$

over all Borel measurable functions $g : \mathbb{R}^d \to \mathbb{R}$.

Proof. If we do not have $\rho \ll \nu$, there exists a set $A \subset \mathbb{R}^d$ such that $\rho(A) > 0$ and $\nu(A) = 0$. Then

$$1 + \sup_g \left\{ \int g \, \mathrm{d}\rho - \int \mathrm{e}^g \, \mathrm{d}\nu \right\}$$

is infinite, as it can be made arbitrarily large by picking functions of the type $g = c\mathbb{1}_A$, $c \in \mathbb{R}$. $D_{KL}(\rho||\nu)$ is also infinite in this case. Hence, their values agree.

When $\rho \ll \nu$, notice that for ν-almost every $x \in \mathbb{R}^d$,

$$g \in \mathbb{R} \mapsto g \frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x) - \mathrm{e}^g$$

is concave and maximized for $g(x) = \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x)\right)$ (note that the Radon–Nikodym derivative $\frac{\mathrm{d}\rho}{\mathrm{d}\nu}$ is non-negative, ν-a.e.).

Thus, for almost every $x \in \mathbb{R}^d$ and any choice of $g(x) \in \mathbb{R}$, we have

$$1 + g(x)\frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x) - \mathrm{e}^{g(x)} \leq 1 + \frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x)\left[\log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x)\right) - 1\right]$$

with equality if and only if $g(x) = \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x)\right)$.

Integrating over the measure ν yields

$$1 + \int_{\mathbb{R}^d} \left(g(x)\frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x) - \mathrm{e}^{g(x)} \right) \mathrm{d}\nu(x) \leq \int_{\mathbb{R}^d} \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}(x)\right) \mathrm{d}\rho(x) = D_{KL}(\rho||\nu)$$

and, thus, one has

$$1 + \sup_g \left\{ \int_{\mathbb{R}^d} g(x)\,\mathrm{d}\rho(x) - \int_{\mathbb{R}^d} \mathrm{e}^{g(y)}\,\mathrm{d}\nu(y) \right\} = D_{KL}(\rho||\nu)$$

since we have equality for

$$g = \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}\right) \qquad \text{on the support of } \nu.$$

□

Remark

1. The variational reformulation of the Kullback–Leibler divergence is a consequence of the convexity of $x \mapsto -\log(x)$. Indeed, computing its Legendre–Fenchel transform twice yields

$$-\log(x) = \sup_{y<0}\left\{ xy + 1 - \log\left(-\frac{1}{y}\right)\right\} = \sup_{g\in\mathbb{R}}\{g - x\,\mathrm{e}^g\} + 1.$$

This approach extends to a broader set of f-divergences, yielding similar variational formulations; see [10] and [11].

2. A very similar variational formulation was developed in [17] to estimate the likelihood of two samples being generated from independent sources.

3. Note that the variational formulation represented above is very similar to the Donsker–Varadhan [3] formula

$$\sup_g \left\{ \int_{\mathbb{R}^d} g(x)\,\mathrm{d}\rho(x) - \log\left(\int_{\mathbb{R}^d} \mathrm{e}^{g(y)}\,\mathrm{d}\nu(y) \right) \right\}.$$

Indeed, $\log(x) \leq x - 1$ yields

$$\sup_g \left\{ \int_{\mathbb{R}^d} g(x)\,\mathrm{d}\rho(x) - \int_{\mathbb{R}^d} \mathrm{e}^{g(y)}\,\mathrm{d}\nu(y) \right\} + 1 \leq \sup_g \left\{ \int_{\mathbb{R}^d} g(x)\,\mathrm{d}\rho(x) - \log\left(\int_{\mathbb{R}^d} \mathrm{e}^{g(y)}\,\mathrm{d}\nu(y) \right) \right\}$$

and equality is achieved for the same maximizer $g = \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\mu}\right)$, if $\rho \ll \nu$ (otherwise, they are both infinite). The formula in Proposition 2 can be considered as a linearization of the Donsker–Varadhan formula, easier to implement numerically.

Given two random variables $Z \sim \rho$ and $Y \sim \nu$ with $\rho \ll \nu$, we can equivalently express the formula in Proposition 2 as

$$D_{KL}(\rho||\nu) = 1 + \max_g \left\{ \mathbb{R}[g(Z)] - \mathbb{R}[\mathrm{e}^{g(Y)}] \right\}.$$

If instead, we are given *independent samples* $z_1, \ldots, z_n$ of Z, and $y_1, \ldots, y_m$ of Y, we can approximate the above reformulation by its empirical counterpart

$$D_{KL}(\rho||\nu) \approx 1 + \max_g \left\{ \frac{1}{n} \sum_i g\left(z_i\right) - \frac{1}{m} \sum_j \mathrm{e}^{g(y_j)} \right\},$$

where the maximization is sought over a suitable class of functions g. Theorem 1 of [10] shows that if this class of functions

1. contains the optimizer $g^* = \log\left(\frac{\mathrm{d}\rho}{\mathrm{d}\nu}\right)$,
2. satisfies the envelope conditions [10] [16a, 16b] (e.g. g is bounded) and
3. satisfies the entropy conditions [10] [17a, 17b] (e.g. Sobolev spaces $\mathcal{W}^{k,2}$ on a compact space),

then we have Hellinger consistency of this estimator, that is

$$\int \left(\sqrt{\exp(g^*)} - \sqrt{\exp(g_{n,m})} \right)^2 \mathrm{d}\nu \xrightarrow[n,m \to +\infty]{} 0, \tag{2.4}$$

where $g_{n,m} = \arg\max \left\{ \frac{1}{n} \sum_i g\left(z_i\right) - \frac{1}{m} \sum_j \mathrm{e}^{g(y_j)} \right\}$.

We deduce from Propositions 1 and 2 the following reformulation of the OT problem between μ and ν, under a quadratic cost, expressed as a minimax problem.

Problem 1 (Minimax reformulation).

$$\min_\phi \max_g L[\phi, g] \equiv \mathbb{R}\left[g(\nabla\phi(X))\right] - \mathbb{R}\left[\mathrm{e}^{g(Y)}\right].$$

Note that the *Lagrangian L* is concave in the maximization variable g, but not necessarily convex in the minimization variable ϕ.

The sample-based version of Problem 1 is given by the following.

PROBLEM 2 (Sample-based minimax reformulation).

$$\min_{\phi}\max_{g} L[\phi, g] \approx \min_{\phi}\max_{g}\left\{\frac{1}{n}\sum_{i} g\left(\nabla\phi(x_i)\right) - \frac{1}{m}\sum_{j} \mathrm{e}^{g(y_j)}\right\}$$

over *suitable* function spaces for $\phi(x)$ and $g(y)$, as detailed in Section 3.

This is an adversarial setting, in which the player with strategy ϕ attempts to minimize the discrepancies between the distributions underlying the sample sets $\left\{\nabla\phi(x_i)\right\}$ and $\left\{y_j\right\}$, while the player with strategy g attempts to show that the two distributions are in fact different. Thus, g would point to those areas where the two distributions differ the most, and ϕ would correct those discrepancies. We will see this competition in action in the examples in Section 4.

This saddle point optimization problem is reminiscent of the ones encountered in the Generative Adversarial Networks (GANs) literature [11]. Broadly speaking, a GAN learns how to generate a sample from an unknown distribution. To do so, a two-player game is introduced; a parameterized *generator Q* aims to produce samples as 'close' as possible to the samples in the training set. This is quantified by the use of an f-divergence (e.g. Kullback–Leibler, Jensen–Shannon or 'GAN' divergence), which is given a variational formulation in the exact same way as it is done in Proposition 2. This in turn introduces a *discriminator*, whose role is to prove that the generator has not done the right job.

Formulated as such, our optimization problem is quite similar to a GAN. Indeed, the generator Q is a distribution that is usually induced by the pushforward of a generic distribution (e.g. standard Gaussian) by a map T. This map, as well as the discriminator, are calibrated using neural networks. This is well within the spirit of the method we use to generate the OT map, as well as the function g (see Section 2.4).

The main differences with the algorithm presented in [11] and ours are the following:

1. Our map T is restricted to the form $\nabla\phi$, where ϕ is convex, in order to solve the quadratic Wasserstein problem. To our knowledge, there are no restrictions on the map in the GAN problem.
2. We use a variational formulation of the Kullback–Leibler divergence instead of the 'GAN' divergence.
3. Instead of using a batch gradient descent for the optimization algorithm, we proceed to what we call 'implicit gradient descent', which is described in Section 3.2.
4. Although our method of generating the map T and 'discriminant' g proceed to a sum or composition of many nonlinear maps, we do not directly use neural networks.

2.2 *Connection with the predetermined features case*

In [6], a set of 'features' $f_1, \ldots, f_K$ serve as test functions to evaluate the statement $\rho = \nu$ for $\rho, \nu \in P(\mathbb{R}^d)$, when we only have sample points $(z_i)_{i=1,\ldots,n}$ and $(y_j)_{j=1,\ldots,m}$ generated from $Z \sim \rho$, $Y \sim \nu$. As in [6], we will assume that μ, ν are 'close'. The general case with more distant measures can be reduced to the solution of many local problems, as shown in Algorithm 1 below, also borrowed from [6].

Definition 2 The samples $(z_i)_{i=1,\ldots,n}$ and $(y_j)_{j=1,\ldots,m}$ generated from random variables $Z \sim \rho$, $Y \sim \nu$ are equivalent for the set of features $f_1, \ldots, f_K$ if

$$\frac{1}{n}\sum_{i=1}^{n} f_k(z_i) = \frac{1}{m}\sum_{j=1}^{m} f_k(y_j), \quad \forall\, k = 1, \ldots, K.$$

The definition above is a relaxation of the equivalence $\mu = \nu \Leftrightarrow \mathbb{R}[f(Z)] = \mathbb{R}[f(Y)]$ for all test functions $f \in C_b(\mathbb{R}^d)$. Then solving the transport problem between the samples (x_i) and (y_j) is reduced to finding a map T such that $(T(x_i))_i$ is equivalent to (y_j), for the features $f_1, \ldots, f_K$.

In [6], T is chosen to be of the type

$$T(x) = \nabla\phi(x) = x + \sum_k \alpha_k \nabla\phi_k(x)$$

for some predetermined functions $\phi_1, \ldots, \phi_K$ and constants $\alpha_1, \ldots, \alpha_K$. In fact, the potentials ϕ_k adopted in [6] agree with the features f_k, but our proposition below applies to more general choices. It shows that the procedure to solve the sample-based OT problem with predetermined features is a particular instance of Problem 2. A specific choice of functional space for g will yield this result. Before introducing it, we need a set of compatibility conditions for the choices of possible ϕ and g.

Definition 3 The features f_k, $k = 1, \ldots, K$ are said to be compatible with the potentials ϕ_k, $k = 1, \ldots, K$ for the sample $(x_i)_{i=1,\ldots,n}$, if the matrix $C \in \mathbb{R}^{K\times K}$ defined as

$$C_{kk'} = \frac{1}{n}\sum_{i=1}^{n} \nabla\phi_k(x_i) \cdot \nabla f_{k'}(x_i)$$

is non-singular.

This compatibility assumption essentially guarantees the non-degeneracy of the choice of functions, as it restricts the average displacement to affect the features in an independent fashion. It can be summarized by the requirement that $C = \mathbb{R}[J_\phi J_f^\top]$ is non-singular, where J_ϕ, J_f are the Jacobian matrices of ϕ, f.

Proposition 3 Given a compatible set of features $f_1, \ldots, f_K$ and potentials $\phi_1, \ldots, \phi_K$ for the sample $(x_i)_{i=1,\ldots,n}$, consider Problem 2 using the functional spaces

$$g(z) = \sum_{k=1}^{K} \beta_k f_k(z), \quad \phi(x) = \frac{|x|^2}{2} + \sum_{k=1}^{K} \alpha_k \phi_k(x)$$

for $\beta \in \mathbb{R}^K$, $\alpha \in \mathbb{R}^K$ in a small enough neighbourhood of zero.

Then the optimizer ϕ of Problem 2 for two sample sets close to each other solves the sample-based OT problem with predetermined features, meaning that $(\nabla\phi(x_i))$ is equivalent to (y_j) for the features $f_1, \ldots, f_K$

$$\frac{1}{n}\sum_{i=1}^{n} f_k(\nabla\phi(x_i)) = \frac{1}{m}\sum_{j=1}^{m} f_k(y_j), \quad \forall\, k = 1, \ldots, K.$$

Proof. The Lagrangian L as a function of α, β is given by

$$\frac{1}{n}\sum_i \left[\sum_{k=1}^{K} \beta_k f_k\left(x_i + \sum_{l=1}^{K} \alpha_l \nabla\phi_l(x_i)\right)\right] - \left[\frac{1}{m}\sum_j \mathrm{e}^{\sum_{k=1}^{K}\beta_k f_k(y_j)}\right].$$

Taking the first-order conditions at optimality yields

$$\nabla_\alpha L = C(\alpha)\beta, \quad \text{where} \quad C(\alpha)_{kk'} = \frac{1}{n}\sum_i \left[\nabla\phi_k(x_i)\cdot\nabla f_k\left(x_i + \sum_{l=1}^{K}\alpha_l\nabla\phi_l(x_i)\right)\right].$$

Since α is in a neighbourhood of zero, the matrix $C(\alpha)$ is a small perturbation of the non-singular matrix C. Since features and potentials are compatible, the matrix C is non-singular, and, thus, $C(\alpha)$ is non-singular itself. Hence,

$$\nabla_\alpha L = 0 \Rightarrow \beta = 0.$$

Moreover, the second optimality condition evaluated at $\beta = 0$ yields $\forall\, k$

$$\partial_{\beta_k} L = \frac{1}{n}\sum_i f_k\left(x_i + \sum_{l=1}^{K}\alpha_l\nabla\phi_l(x_i)\right) - \frac{1}{m}\sum_j f_k(y_j).$$

Hence, $\nabla_\beta L = 0$ at $\beta = 0$ implies that

$$\frac{1}{n}\sum_i f_k\left(x_i + \sum_{l=1}^{K}\alpha_l\nabla\phi_l(x_i)\right) = \frac{1}{m}\sum_j f_k(y_j).$$

Notice that the closeness of the two sample sets and the compatibility between the potential and features guarantee that this problem has a solution with a small α (in fact, this can be taken as a feature-dependent characterization of what it means for two sample sets to be close to each other). This result means that the empirical expected values of the f_k agree on $\{T(x_i)\}$ and $\{y_j\}$, i.e. the samples are equivalent for the features $f_1, ..., f_K$. Hence, $T = \nabla\phi$ solves the sample-based OT problem with predetermined features. □

Note that we are restricting the maps $\nabla\phi$ to be 'small' perturbations of the identity, by choosing α in a neighbourhood of 0. This is because the OT procedure will only be applied to measures or samples that are 'close' to each other.

In this paper, we will allow g to be more general than a simple linear combination of features, thus greatly expanding the procedure in [6]. This added flexibility yields better adaptability to the most important characteristics of the data.

2.3 *Duality*

2.3.1 *No duality gap*

Given the Lagrangian L introduced in Problem 1, the primal objective functional to minimize is, according to Proposition 2,

$$D[\phi] = \max_g L[\phi, g] = D_{KL}\left(\nabla\phi_{\#}\mu||\nu\right) - 1. \tag{2.5}$$

The proof in Proposition 4 shows that the dual objective functional to be maximized is

$$d[g] = \min_\phi L[\phi, g] = \left(\min_{y\in\mathbb{R}^d} g(y)\right) - \mathbb{E}\left[\mathrm{e}^{g(Y)}\right]. \tag{2.6}$$

A desired property of the adversarial game, defined by the formulation in Problem 1, is the absence of an irreversible advantage or penalty a player gets from playing first. In other words we do not want a duality gap. This is the content of the following proposition.

Proposition 4 (Absence of duality gap).

$$\min_\phi \max_g L[\phi, g] = \min_\phi D[\phi] = \max_g d[g] = \max_g \min_\phi L[\phi, g].$$

Proof. From Proposition 1, we know that

$$\min_\phi D_{KL}(\nabla\phi_{\#}\mu||\nu) = 0$$

with the minimizer reached for the solution of the transport problem.

Hence, we get in Equation (2.5)

$$\min_\phi \max_g L[\phi, g] = \min_\phi D[\phi] = -1.$$

On the other hand, maximizing Equation (2.6) yields

$$\max_g \min_\phi L[\phi, g] = \max_g \left\{\min_\phi \mathbb{E}[g(\nabla\phi(X))] - \mathbb{E}\left[\mathrm{e}^{g(Y)}\right]\right\}.$$

Note that the inner minimum is reached for the convex function $\phi(x) = y_{min}\cdot x$, where $\min_y g(y) = g(y_{min}) \equiv g_{min}$.

In the case where the minimum of g is not reached, take a minimizing sequence y^n_{min} such that $g(y^n_{min}) \to \inf_{y\in\mathbb{R}^d} g(y) \equiv g_{min}$. Then a minimizing sequence for the inner minimum in ϕ is given by $\phi^n(x) = y^n_{min}\cdot x$.

In both cases,

$$\min_{\phi} \mathbb{E}[g(\nabla\phi(X))] = g_{min}.$$

We are, thus, left with maximizing the dual problem

$$\max_{g} d[g] = \max_{g} \left\{ g_{min} - \mathbb{E}\left[\mathrm{e}^{g(Y)} \right] \right\}.$$

Since $\mathbb{E}\left[\mathrm{e}^{g(Y)}\right] \geq \mathrm{e}^{g_{min}}$, we can always choose g to be the constant function g_{min}. We are then left with maximizing

$$\max_{g_{min}} g_{min} - \mathrm{e}^{g_{min}},$$

which is achieved for $g \equiv g_{min} = 0$. Hence, we also have that

$$\max_{g} d[g] = \max_{g} \min_{\phi} L(\phi, g) = -1.$$

□

2.3.2 *An adversarial view of duality* The optimality conditions for the minimax problem are given by

$$\begin{cases} \nabla\phi \text{ moves mass to where } g \text{ is smallest} \\ g(y) = \log\left(\frac{\nabla\phi_{\#}\mu(y)}{\nu(y)}\right). \end{cases}$$

Examining the primal and dual problems in light of these conditions explains the behaviour of the competing players ϕ and g.

- Given a function g, ϕ will try to move mass from the areas where g is large (i.e. $\nabla\phi_{\#}\mu(y) \geq \nu(y)$) to those where g is small (i.e. $\nabla\phi_{\#}\mu(y) \leq \nu(y)$). Following this strategy allows this player to minimize the impact of g on the Lagrangian.
- Given a function ϕ, g will adapt to get closer to the function $\log\left(\frac{\nabla\phi_{\#}\mu(y)}{\nu(y)}\right)$, which is large where mass is lacking ($\nabla\phi_{\#}\mu(y) \geq \nu(y)$) and vice versa. Following this strategy allows the second player to increase the Lagrangian by focusing on those areas where the push-forward condition has not been fully achieved.

The game concludes when g becomes constant (necessarily 0) on the support of the distributions. Then ϕ does not need to move mass anymore, as it then receives no new directive from g.

2.4 *Global algorithm*

One could attempt to directly use a procedure based on Problem 2 to solve the OT problem for any samples $(x)_i$ and $(y)_j$. Such direct approach, however, would not be universally efficient for the following reasons:

- If the distributions underlying $(x)_i$ and $(y)_j$ are considerably different, one would require a very rich family of potentials to build a ϕ that can perform an accurate transfer.

- One would also require a rich functional space from which to draw g in order to properly characterize all significant differences in the two data samples.
- Depending on the parametrization of ϕ and g, the Lagrangian can be non-convex in the variables parametrizing ϕ and non-concave in the variables parametrizing g. With distributions that are far apart, this could make the numerical solution depend on the initialization of those parameters.
- The condition that ϕ is a convex function is typically hard to enforce. For nearby distributions, on the other hand, it is satisfied automatically, as $\phi(x)$ is close to the convex potential $\frac{1}{2}\|x\|^2$ corresponding to the identity map.

Algorithm 1 Theoretical Global Optimal Transport Algorithm (TGOT)

```
procedure TGOT($\mu, \nu$)
  ▷ Step 1: initialize intermediate nodes
  $N \leftarrow$ number of intermediary steps
  $\rho_0 \leftarrow \mu, \quad \rho_T \leftarrow \nu$
  for $t = 1, \ldots, N-1$ do
    $\rho_t \leftarrow \frac{N-t}{N}\mu + \frac{t}{N}\nu$                ▷ or any arbitrary measure
  end for
  while not converged do
    ▷ Step 2: forward step
    for $t = 1, \ldots, N$ do
      Solve the OT problem between $\rho_{t-1}$ and $\rho_t$, as defined in Problem 1. It yields a 'local' optimal map $\nabla\phi_t$.
    end for
    $\nabla\phi \leftarrow \nabla\phi_N \circ \nabla\phi_{N-1} \circ \cdots \circ \nabla\phi_1$
    ▷ Step 3: backward step
    for $t = 1, \ldots, N-1$ do
      $\rho_t \leftarrow \left(\frac{N-t}{N} Id + \frac{t}{N}\nabla\phi\right)_\# \mu$
    end for
  end while
  return $\nabla\phi$
end procedure
```

Algorithm 2 Sample-Based Global Optimal Transport Algorithm (SBGOT)

```
procedure SBGOT($(x_i), (y_j)$)
  ▷ Step 1: initialize intermediate nodes
  $N \leftarrow$ number of intermediary steps
  $z_0 \leftarrow x, \quad z_N \leftarrow y$
  for $t = 1, \ldots, N-1$ do
    $z_{t,i} \leftarrow \frac{N-t}{N} x_i + \frac{t}{N} y_{\sigma(i)}$    ▷ for some $\sigma : \{1, \ldots, n\} \to \{1, \ldots, m\}$ (or any arbitrary samples)
  end for
  while not converged do
    ▷ Step 2: forward step
    for $t = 1, \ldots, N$ do
      $z_t \leftarrow SBLOT(z_{t-1}, z_t)$    ▷ see Algorithm 3
    end for
    ▷ Step 3: backward step
    for $t = 1, \ldots, N-1$ do
      $z_t \leftarrow \frac{N-t}{N} x + \frac{t}{N} z_N$
    end for
  end while
  return $z_N$
end procedure
```

For these reasons, we will solve multiple local OT problems, instead of one global one. More precisely, we will apply Algorithm 1, adapted from Algorithms 2 and 7 in [6].

Theorem 2.4 in [6] proves the convergence of Algorithm 1 to the solution of the OT problem.

In Algorithm 1, the forward step consists of solving multiple, small, OT problems, addressed in Section 3.2. The backward step back-propagates the final sample computed in the forward pass to all the intermediate samples using McCann's displacement interpolants.

This procedure, reminiscent of the neural networks of machine learning with their 'hidden layers' replaced by local OT problems, introduces several advantages:

- The global solution will be obtained by composition of the local maps

$$\nabla\phi = \nabla\phi_N \circ \nabla\phi_{N-1} \circ \cdots \circ \nabla\phi_1. \tag{2.7}$$

  Hence, one can choose a small family of maps to solve each local OT problem, and still span a rich family of maps for the global displacement.

- Note that in our two-player game, we would theoretically have at optimality $T_{\#}\mu = \nu$ and hence the optimal g would be equal to $\log(T_{\#}\mu/\nu) = 0$.
- If ρ_t and ρ_{t+1} are close, the local OT problem has a solution $\nabla\phi$ that is a small perturbation of the identity, i.e. the gradient of a strictly convex potential. Starting from the identity, the numerical algorithm will explore a small neighbourhood around it. If the solution that we seek is in this neighbourhood, convexity will be preserved.

The global algorithm for finding the optimal map between two distributions known through the samples (x_i) and (y_j) is summarized in Algorithm 2.

Algorithm 3 in Section 3 further details the procedure to solve the sample-based local OT problem.

3. Algorithm

In order to complete the description of the algorithm proposed, we need to specify the functional spaces from which g and ϕ are drawn, and the procedure used for solving the minimax problem of the Lagrangian $L(g, \phi)$.

3.1 *Choice of functional spaces*

Since any two consecutive distributions μ, ν in the procedure are close to each other, the optimal map is a perturbation of the identity. The potential ϕ will, thus, be chosen in the form

$$\phi(x) = \frac{1}{2}\|x\|^2 + \psi(x), \tag{3.1}$$

where ψ has a Hessian with a spectral radius less than 1. No such centering is required for $g(x)$, as at optimality $g(x) = \log(1) = 0$.

One basic capability that one should require of the functional spaces for g and ϕ is that of detecting and correcting global displacements and scaling—not necessarily isotropic—between two distributions. Thus, one should have

$$\phi(x) = \frac{1}{2}x^\top(I + A_0)x + a_1 \cdot x + \phi_{nl}(x)$$

and

$$g(z) = \frac{1}{2}z^\top B_0 z + b_1 \cdot z + b_2 + g_{nl}(z),$$

where A_0, B_0 are symmetric matrices in $\mathbb{R}^{d\times d}$, a_1, b_1 are vectors in $\mathbb{R}^d$, $b_2 \in \mathbb{R}$ is a scalar and ϕ_{nl} and g_{nl} stand for additional nonlinear features discussed below. The quadratic polynomial in ϕ allows for global translations and dilations. Correspondingly, the quadratic polynomial in g allows for the detection of any mismatch in the mean and covariance of the two distributions. One can easily check that, with these basic functions available, the procedure yields the exact solution to the OT problem between arbitrary Gaussians.

If these are the only features available, then there is no advantage in dividing the global problem into local ones, as the composition of linear maps is also linear, thereby providing no additional richness to the single-step scenario. The natural element to add is an adaptive feature that could perform—and detect the need of—local mass displacements. In one dimension, a natural choice is provided by one or more Gaussians of the form

$$\phi_{nl}^k = \alpha_k \exp\left(-\frac{[v_k(x-\bar{x}_k)]^2}{2}\right), \quad g_{nl}^k = \beta_k \exp\left(-\frac{[s_k(z-\bar{z}_k)]^2}{2}\right),$$

where the index k labels the Gaussian feature when more than one is used. The Gaussians in ϕ allow for local stretching/compression around m with scale $|v|^{-1}$ and amplitude α, while each Gaussian in g detects local discrepancies between the two distributions, as opposed to the global scale and positioning provided by its quadratic component. The parameters v, $\bar{x}$, s and $\bar{z}$ appear nonlinearly in ϕ and g, moving us away from the linear feature spaces of [6] and into the realm of adaptability, as the parameters automatically select the location and scale of the changes required by the data.

There are at least four alternative ways to bring these Gaussian features to higher dimensions:

1. Adopt general Gaussians of the form

$$\phi_{nl} = \alpha \exp\left(-\frac{\|V(x-\bar{x})\|^2}{2}\right),$$

with $\bar{x}$ a vector and V a matrix (it is more convenient to write the Gaussian in terms of a general matrix V in this way, rather than in terms of the inverse covariance matrix $C^{-1} = V^T V$, as we would need to require this to be positive definite);

2. adopt isotropic Gaussians

$$\phi_{nl} = \alpha \exp\left(-\frac{v\|x-\bar{x}\|^2}{2}\right),$$

with v a scalar;

3. adopt one-dimensional Gaussians along arbitrary directions

$$\phi_{nl} = \alpha \exp\left(-\frac{\|v\cdot(x-\bar{x})\|^2}{2}\right),$$

with v a vector and

4. adopt a Gaussian with diagonal covariance

$$\phi_{nl} = \alpha \exp\left(-\frac{\|D(x-\bar{x})\|^2}{2}\right),$$

with D a diagonal matrix,

and similarly for g_{nl} in all four cases. The first choice has the advantage of generality, but may be prone to overfitting in high dimensions, unless it is severely penalized. The second approximates a general function ϕ by the composition of isotropic bumps, an appropriate image is that of hammering a sheet of metal into any desired shape. Yet, it would resolve poorly local, one-dimensional changes. The third choice excels at these, but will fare poorly for more isotropic local changes. Finally, the fourth choice is attached to the coordinate axes, which would make sense only if these correspond to variables that are assumed to change independently.

A natural question is how many Gaussians to include in the functional space proposed. We have used two in the examples below, but one Gaussian would have sufficed; in the adversarial multistep method proposed, it is enough that the player with strategy $g(y)$ has a 'lens'(the Gaussian) to identify the area where the two distributions least agree, and the player with strategy $\phi(x)$ has the capability to perform local moves to correct this misfit. Since the center and width of the Gaussian are free parameters, both assertions hold. With a single Gaussian feature, both players can focus only on one local misfit at a time. However, the algorithm has multiple steps, so effectively the total number of features available is the product of the features per step times the number of steps.

3.2 *Local Algorithm*

We will use vectors $\alpha \in \mathbb{R}^a, \beta \in \mathbb{R}^b$ to parametrize $\phi(x) = \phi_\alpha(x)$ and $g(y) = g_\beta(y)$. We are seeking to solve the minimax problem in $\alpha \in \mathbb{R}^a, \beta \in \mathbb{R}^b$ for the Lagrangian

$$L[\alpha, \beta] = \frac{1}{n}\sum_{i=1}^{n} g_\beta(\nabla\phi_\alpha(x_i)) - \frac{1}{m}\sum_{j=1}^{m} \mathrm{e}^{g_\beta(y_j)} + P(\alpha, \beta),$$

where P is a penalization function that will be described in Section 3.3.

In practice, one could use any available minimax solver to find a critical point of the above Lagrangian. Yet, to our knowledge, there is no available efficient method suitable for a non-convex/non-concave landscape.

A naive algorithm would simultaneously implement gradient descent in α and gradient ascent in β, with updates given at each step s by

$$\begin{aligned}\alpha^{s+1} &= \alpha^s - \eta\nabla_\alpha L[\alpha^s, \beta^s]\\ \beta^{s+1} &= \beta^s + \eta\nabla_\beta L[\alpha^s, \beta^s],\end{aligned}$$

with a step size η that may change at each iteration. From a game theory perspective, this corresponds to two myopic players that plan their next move based only on their current position, without anticipating what the other player might do.

Instead, more insightful players will choose their next move based on the future position of their opponents. This yields a second-order algorithm, that we will refer to as *implicit* gradient descent, with updates given by

$$\begin{aligned}\alpha^{s+1} &= \alpha^s - \eta\nabla_\alpha L[\alpha^{s+1}, \beta^{s+1}]\\ \beta^{s+1} &= \beta^s + \eta\nabla_\beta L[\alpha^{s+1}, \beta^{s+1}].\end{aligned}$$

A simple Taylor expansion gives

$$\begin{aligned}\nabla_\alpha L[\alpha^{s+1}, \beta^{s+1}] &\approx \nabla_\alpha L^s + \nabla^2_{\alpha\alpha}L^s \cdot (\alpha^{s+1} - \alpha^s) + \nabla^2_{\alpha\beta}L^s \cdot (\beta^{s+1} - \beta^s)\\ \nabla_\beta L[\alpha^{s+1}, \beta^{s+1}] &\approx \nabla_\beta L^s + \nabla^2_{\alpha\beta}L^s \cdot (\alpha^{s+1} - \alpha^s) + \nabla^2_{\beta\beta}L^s \cdot (\beta^{s+1} - \beta^s).\end{aligned}$$

Defining the *twisted* gradient G^s and *twisted* Hessian H^s by

$$G^s = \begin{pmatrix} \nabla_\alpha L^s \\ -\nabla_\beta L^s \end{pmatrix}, \quad H^s = \begin{pmatrix} \nabla^2_{\alpha\alpha} L^s & \nabla^2_{\alpha\beta} L^s \\ -\nabla^2_{\alpha\beta} L^s & -\nabla^2_{\beta\beta} L^s \end{pmatrix}$$

and $\gamma^s = \begin{pmatrix} \alpha^s \\ \beta^s \end{pmatrix}$, one obtains the second-order updating scheme

$$\gamma^{s+1} = \gamma^s - \eta \left(I + \eta H^s\right)^{-1} G^s. \tag{3.2}$$

Notice that as $\eta \to 0$, the scheme is equivalent to a classical gradient descent. On the other hand, as $\eta \to +\infty$, the scheme converges to Newton iterations.

At each iteration, we are allowed to update η in order to accelerate convergence. Ongoing research (subject to work in preparation) addresses the correct rules to update η, as well as the convergence of the algorithm to a critical point of the Lagrangian. This minimax solver is robust in two senses: it guarantees both convergence to a local minimax point and constant improvement. The latter has to do with the subtlety of minimax problems, as opposed to regular minimization where enforcing a decrease of the objective function is enough. In each step of our implicit procedure to $\min_x \max_y L(x, y)$, if $L[x^{s+1}, y^{s+1}]$ is either bigger than $L[x^s, y^{s+1}]$ or smaller than $L[x^{s+1}, y^s]$, we reject the step and adopt a smaller learning rate. Because of this, the solution will always improve over the starting identity map. If computing the twisted Hessian H becomes too costly, one can resort to Hessian approximation techniques such as Broyden-Fletcher-Goldfarb-Shanno algorithm or its variations [12,21].

To conclude, the algorithm for finding the optimal match between two consecutive distributions, which we denote SBLOT, is summarized in Algorithm 3.

Algorithm 3 Sample-Based Local Optimal Transport Algorithm (SBLOT)

```
procedure SBLOT($(x_i), (y_j)$)
  Initialize $\gamma$
  Compute the twisted gradients and Hessians $G, H$
  for $n = 1, \dots, MaxIter$ do
    if $||G|| < tolerance$ then
      break
    end if
    $\gamma \leftarrow \gamma - \eta(I + \eta H)^{-1} G$
    Recompute the twisted gradients and Hessians $G, H$ at $\gamma$
    Update $\eta$
  end for
  return $\nabla\phi_{\gamma[1:a]}(x)$
end procedure
```

3.3 *Penalization*

Transforming Problem 1 into Problem 2 amounts to replacing the theoretical measures with their empirical estimates

$$\rho \approx \hat{\rho} = \frac{1}{n}\sum_{i=1}^{n} \delta_{\nabla\phi(x_i)}, \quad \nu \approx \hat{\nu} = \frac{1}{m}\sum_{j=1}^{m} \delta_{y_j}.$$

Even if $\rho \ll \nu$, this will not hold for their estimates. Allowing maximum freedom for the function g will result in an infinite Kullback–Leibler divergence. For instance, if one allows functions g with support including some $\nabla\phi(x_i)$, but none of the y_j, the Lagrangian will grow unboundedly, since the exponential term that regularly inhibits this growth is now constant. One way to avoid this problem is to use the relative entropy not between $T(X)$ and Y, but between $T(X)$ and $(1-\epsilon)Y + \epsilon T(X)$, as then the law of $T(X)$ is always absolutely continuous w.r.t. the law of $(1-\epsilon)Y + \epsilon T(X)$, eliminating the possibility of blowup in g, and the minimum is still reached when $T(X) = Y$. Another general simple way to avoid this kind of scenario is through the addition to the Lagrangian of terms that penalize overfitting. For our particular choice of functional spaces, it is only the coefficients in the argument of the exponentials that require penalization, as those are the only ones than involve spatial scales. In particular, for a component of g or ϕ of the form

$$a\,\mathrm{e}^{-(b\cdot(x-c))^2},$$

we add penalization terms proportional to

$$\mathrm{e}^{(\epsilon\|b\|)^2},$$

with ϵ as defined above, to avoid resolving scales smaller than ϵ, to

$$\frac{1}{(D\|b\|)^2},$$

where D measures the diameter of the support of the data, to avoid having Gaussians so broad that they are indistinguishable from the quadratic components of the functional space, to

$$\left\|\frac{c}{D}\right\|^2,$$

to avoid centering the Gaussian away from the data, and, when more that one Gaussian is used, to

$$\frac{\epsilon^2}{\|c_i - c_j\|^2},$$

for every pair (i,j) of Gaussians, to avoid possible degeneracies in the functional space when two Gaussians become nearly indistinguishable.

All these terms are added and multiplied by a tunable parameter λ. Yet, one more consideration is required for the penalization of the parameters of the potential ϕ: since in the Lagrangian, ϕ appears only as an argument of g, for a fixed λ, the penalization terms and the core Lagrangian can easily become unbalanced. In particular, at the exact solution, g is zero, so only the penalization terms will remain. To correct for such imbalance, we multiply the corresponding penalization terms by the average value of $\|\nabla g\|$ over all current $\nabla\phi(x_i)$.

4. Experiments

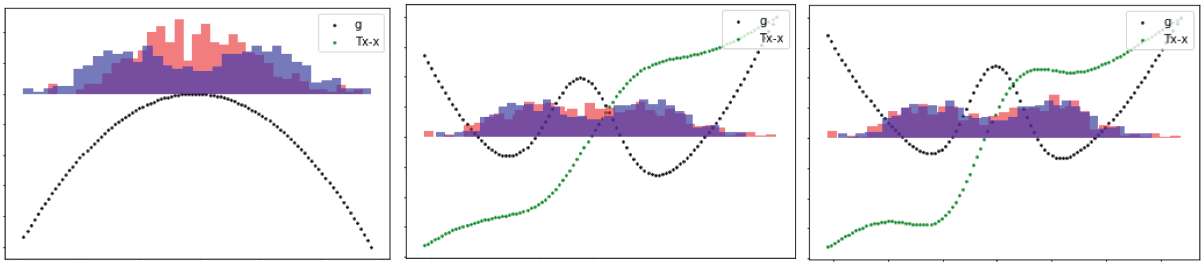
This section illustrates the algorithm through some simple examples. First we use a one-dimensional example—simplest for visualization—and a direct solver between initial and final distributions to display the way in which the function g adapts, creating features that point to those areas where transport in still deficient, thus guiding ϕ to correct them. The two distributions in the first example are relatively close, so that they can be matched without involving intermediate distributions. A second set of one-dimensional examples follows, involving more significant changes and hence requiring the use of interpolated distributions. Then we perform some two-dimensional examples, involving Gaussians, Gaussian mixtures and a distribution uniform within an annulus. Finally, we use an example built so that we know the exact answer to perform an empirical analysis of convergence. All the examples presented are intended for illustration and use synthetic data; applications to real data, particularly to change detection, will be presented in field-specific articles currently under development.

4.1 *Adversarial behaviour of ϕ and g*

This section shows, through a simple experiment, the competitive behaviour exhibited by the two players ϕ and g in the local algorithm (Algorithm 3). To this end, we create data where the initial and final distribution are not very far from each other, so that the local algorithm can be used as a stand alone routine. More specifically, we map one single Gaussian distribution to a Gaussian mixture, where the two components of the mixture overlap significantly, so that they do not differ too markedly from the source.

Figure 1 shows steps in the solution to the corresponding sample-based OT problem, with the source samples $(x)_i$ from a Gaussian—and their transforms—in red and the samples $(y)_j$ from a mixture of two Gaussians in blue. Point samples are represented through histograms. The figure on the left represents the initial configuration, the one in the middle the configuration after 10 iterations of Algorithm 3 and the one on the right the final configuration.

On top of the histograms, we display the function $g(x)$ in black, scaled vertically to be in the interval $[-1; 1]$ for easier comparison with the data, and the displacement $\nabla\phi(x) - x$ in green, representing the map that sends the initial sample (in red, in the left figure) to the current sample (in red, in the middle or right figure).



Iteration 0

After 8 iterations

After 17 iterations

FIG. 1. Plot at three different iteration times of Algorithm 3. Histograms of the source samples (middle Gaussian) and their transforms are in red and of the target samples in (two symmetrically decentered Gaussians). The darker curve corresponds to $g(x)$, vertically rescaled for visualization. The lighter curve represents the displacement $T(x) - x$.

The initial displacement, being 0, was not represented at initialization, but we initialize the function $g(z)$ at the purely quadratic function

$$\frac{1}{2}z^T\left(\hat{\Sigma}_y^{-1}-\hat{\Sigma}_x^{-1}\right)z+\left(\hat{\Sigma}_x^{-1}\hat{x}-\hat{\Sigma}_y^{-1}\hat{y}\right)^T z+\frac{1}{2}\left(\hat{y}^T\hat{\Sigma}_y^{-1}\hat{y}-\hat{x}^T\hat{\Sigma}_x^{-1}\hat{x}\right), \tag{4.1}$$

where $\hat{x}, \hat{y}$ are the empirical means of the samples $(x)_i, (y)_j$ and $\hat{\Sigma}_x, \hat{\Sigma}_y$ their empirical covariance matrices. Equation 4.1 represents the optimal g for two Gaussian measures. More generally, starting with this expression as the initial guess for g instructs ϕ to shift the samples as well as to stretch/compress them, in order to match the first and second moments of the two distributions.

The left image of Fig. 1 shows how g highlights the lack of variance in $(x)_i$; its maximum is at 0, and it has smaller values at the edges. This forces ϕ to adapt accordingly, by applying a linear map to stretch $(x)_i$. When the variance of the $(\nabla\phi(x))_i$ exceeds the variance of the $(y)_j$, the shape of g is inverted.

In the middle image of Fig. 1, we can see that $\nabla\phi$ corrected the mismatches highlighted by g and even started to slightly separate the mass in the middle. However, there is still too much red mass around 0 and too little red mass around the two peaks of the blue Gaussian mixture. This is well detected by g, which has a local maximum within the area of red mass excess and two local minima within the area of red mass default. In the right image of Fig. 1, we observe that $\nabla\phi$ adapted accordingly and starts yielding satisfactory results. At this point, g is very close to 0 ($||g||_\infty \sim 10^{-5}$), although this is not apparent in the figure due to the normalization we applied for plotting.

4.2 *The global algorithm in dimension one*

Figures 2 and 3 represent inputs and outputs of Algorithm 2, where $(x)_i$ is sampled from a Gaussian and $(y)_j$ from a mixture of two and three Gaussians, respectively.

These results were obtained by generating ∼200 samples for the source and target measures, and using the functional spaces defined in Section 3.1 in the local algorithm (Algorithm 3), with a general quadratic form for both ϕ and g, plus one adaptive Gaussian for ϕ and two for g. A total of $N = 10$ and $N = 20$ intermediary measures were adopted for the first and second example, respectively. As one can see, even though each local map can only perform one local deformation, the composition of many creates all the complexity required to move one single Gaussian to a mixture of two or three.

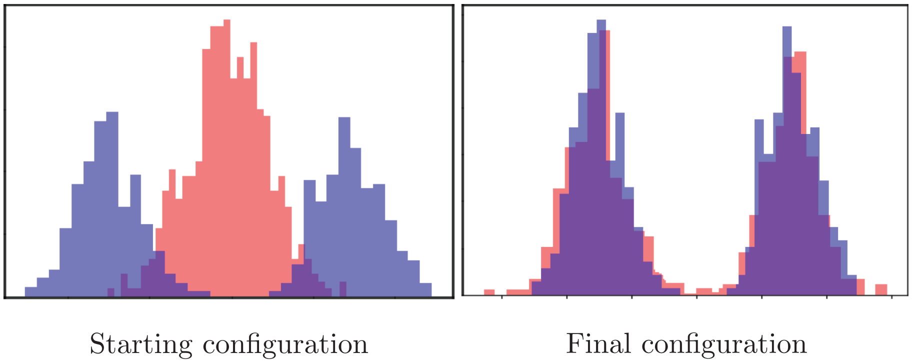
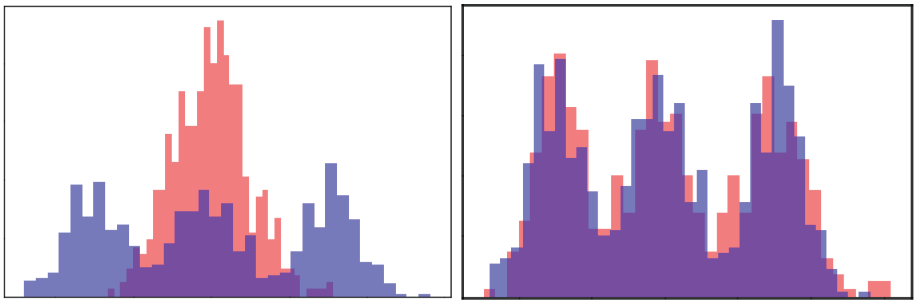


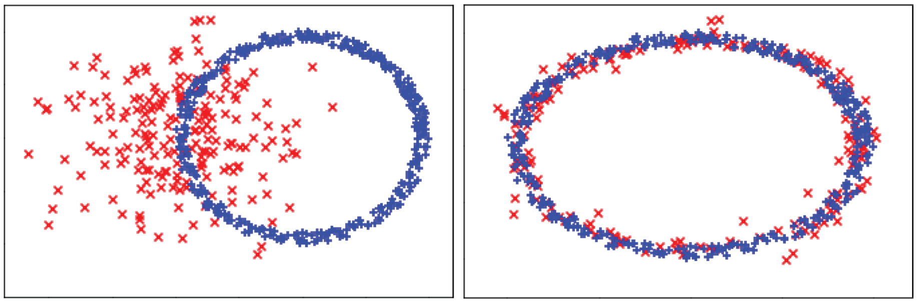
FIG. 2. Algorithm 2 pushing forward a Gaussian to a mixture of two Gaussians in one dimension. The source samples (middle in the left picture) and the target samples (two symmetrically decentered Gaussians) are depicted through histograms.



Starting configuration

Final configuration

FIG. 3. Same as Fig. 2, but with a mixture of three Gaussians as target.



Starting configuration

Final configuration

FIG. 4. Algorithm 2 from a displaced Gaussian to an annulus in two dimensions.

4.3 *Two-dimensional examples*

Switching to two dimensions, Fig. 4 represents the results of mapping a Gaussian distribution to a uniform distribution within an annulus.

An isotropic Gaussian was used for ϕ_{nl} and two for g_{nl} in the functional space of Algorithm 3, and $N = 30$ intermediary distributions were used in Algorithm 2. Figure 5 represents the displacement interpolants at $t = k/5$ for $k = 1, \ldots, 5$, obtained from running Algorithm 2 on the example in Fig. 4. In addition to mass spreading from the isotropic Gaussian, the linear and quadratic part of ϕ translated and stretched the red sample accordingly.

Similarly, Fig. 6 represents the initial and final configurations obtained from running Algorithm 2 to transport a two-dimensional Gaussian distribution to a mixture of two Gaussians. A diagonal covariance was used in the nonlinearity ϕ_{nl} for the functional space in Algorithm 3, and $N = 30$ intermediary steps were used in Algorithm 2. This type of nonlinearity is well adapted to separate samples along the horizontal and vertical axes.

Figure 7 represents the displacement interpolants at $t = k/5$ for $k = 1, \ldots, 5$, obtained from running Algorithm 2 on the example in Fig. 6.

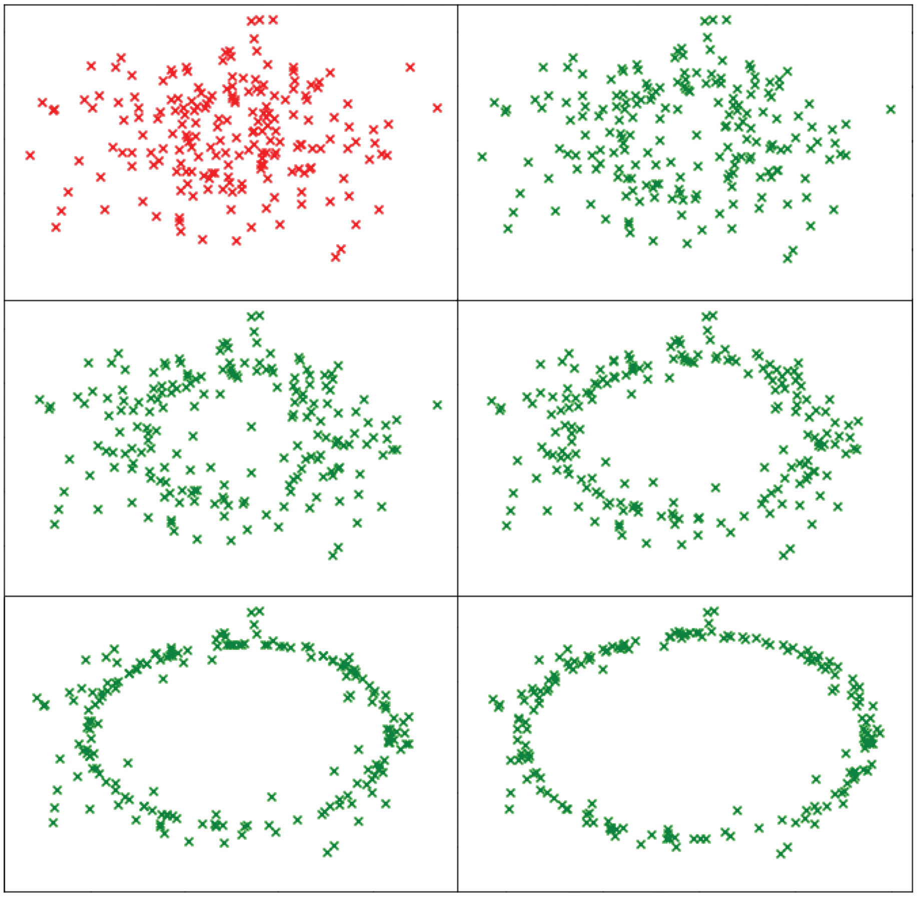


FIG. 5. Interpolants given by Algorithm 2 from a Gaussian to an annulus in two dimensions. The top left figure corresponds to the original sample. Time flows from left to right and from top to bottom. Subsequently represented are the interpolants at time $t = k/5$ for $k = 1, \ldots, 5$.

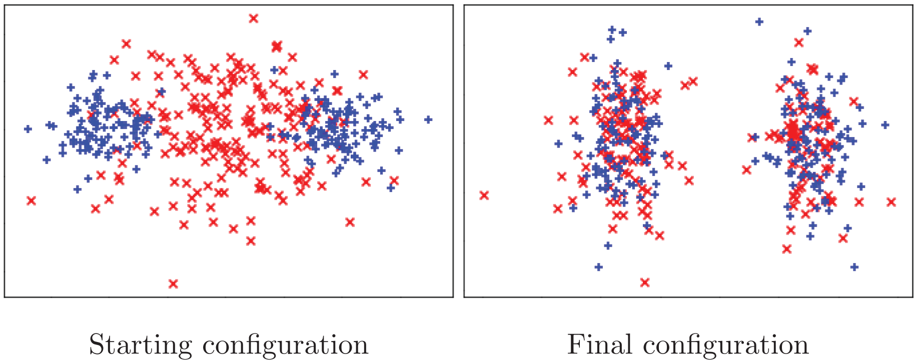


FIG. 6. Algorithm 2 from a Gaussian to a mixture of 2 Gaussians in two dimensions.

4.4 *Empirical analysis of convergence*

In this subsection, we empirically analyse the convergence of the algorithm in a situation where the generating distributions, as well as the optimal map, are known: $(x_i)_{i=1,\ldots,n}$ are i.i.d. samples of a

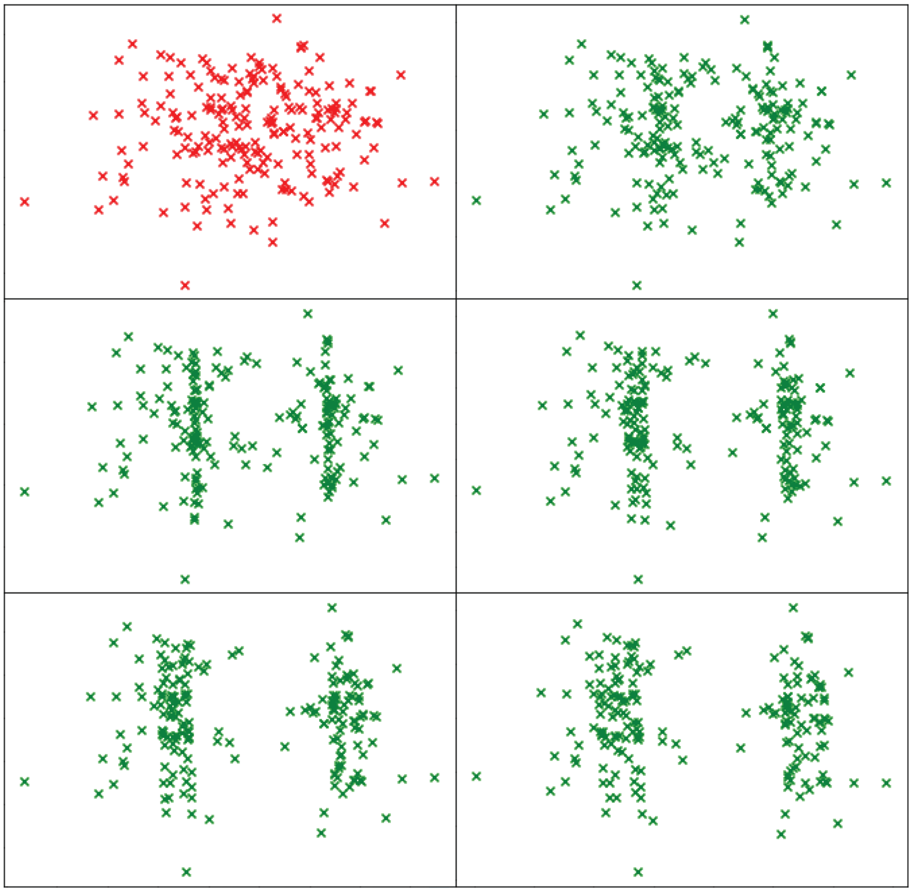


FIG. 7. Interpolants given by Algorithm 2 from a Gaussian to a mixture of two Gaussians in two dimensions. The top left figure (red) corresponds to the original sample. Time flows from left to right and from top to bottom. Subsequently represented are the interpolants at time $t = k/5$ for $k = 1, \dots, 5$.

standard Gaussian distribution, and $(y_j)_{j=1,\dots,m}$ are obtained through $y_i = \phi'(x_i)$ for $\phi(x) = |x|^{1+\epsilon}$ ($\epsilon = 1/4$). Brenier's theorem guarantees that, since ϕ is convex, ϕ' is the optimal map for the quadratic Wasserstein problem.

In a first set of experiments, we keep the number of samples constant at $n = m = 500$, and we vary the number of intermediary steps K in the global algorithm, raging through $K = 1, 2, 3, 5, 10$. In a second set of experiments, we keep the number of intermediary steps in the global algorithm constant at $K = 10$, and vary the number of sample points, using $n = m = 25, 50, 100, 200, 500$. In both sets, we compute the experimental map $\nabla\phi_{exp}$ by (2.7), and compare it to the optimal $\nabla\phi^*$ defined by

$$\nabla\phi^*(x) = (1 + \epsilon)x|x|^{\epsilon-1}.$$

In each experiment, two numerical quantities are computed:

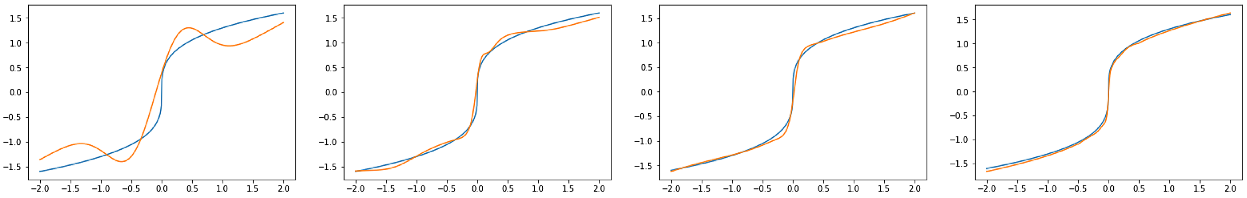
1. the weighted L^2 norm $\int |\nabla\phi_{exp}(x) - \nabla\phi^*(x)|^2 \mu(x)\,\mathrm{d}x \approx \sum_i |\nabla\phi_{exp}(x_i) - \nabla\phi^*(x_i)|^2$ and
2. the L^∞ norm between $\nabla\phi_{exp}$ and $\nabla\phi^*$.

Table 1 *Convergence as a function of the number K of intermediary steps*

$K =$	1	2	3	5	10
$\mathbb{R}[\lvert\nabla\phi^*(X) - \nabla\phi_{exp}(X)\rvert^2]$	0.74	0.55	$8.3 \cdot 10^{-1}$	$1.7 \cdot 10^{-2}$	$8.7 \cdot 10^{-3}$
$\lvert\lvert\nabla\phi^* - \nabla\phi_{exp}\rvert\rvert_{L^\infty}$	0.53	0.22	$9.9 \cdot 10^{-2}$	$8.7 \cdot 10^{-2}$	$6.2 \cdot 10^{-2}$

Table 2 *Convergence as a function of the number of samples n*

$n = m =$	25	50	100	200	500
$\mathbb{R}[\lvert\nabla\phi^*(X) - \nabla\phi_{exp}(X)\rvert^2]$	1.4	0.35	$7.1 \cdot 10^{-2}$	$2.1 \cdot 10^{-2}$	$8.7 \cdot 10^{-3}$
$\lvert\lvert\nabla\phi^* - \nabla\phi_{exp}\rvert\rvert_{L^\infty}$	1.3	0.49	0.16	0.11	$6.2 \cdot 10^{-2}$



$K = 1$ $K = 3$ $K = 5$ $K = 10$

Fig. 8. Comparison between $\nabla\phi^*$ (blue) and $\nabla\phi_{exp}$ (orange) for different values of intermediary steps K.

For illustrative purposes, we show in Fig. 8 the differences between $\nabla\phi_{exp}$ and $\nabla\phi^*$ for various sets of parameters. Tables 1 and 2 summarize the results.

In practice, setting a number of samples less than 15 in this example leads to poor convergence due to the extreme sparsity of data.

Figure 8 compares the optimal map $\nabla\phi^*$ with the computed map $\nabla\phi_{exp}$. Note that the one step algorithm does not provide a monotone solution, i.e. it is not the gradient of a convex function; the source and target distributions are not close enough to guarantee that. This is corrected through the introduction of intermediate steps, which brings the source and target distributions for each step closer to each other via displacement interpolation. For the example under consideration, the optimal solution is convex for any value of K bigger than 4. Notice also that, for $K = 10$ and $n = 500$, the solution approximates the exact one very accurately in the bulk of the distribution, as captured by the density-weighted L_2 norm of their difference. On the other hand, the L_∞ norm is dominated by the behaviour at the tails, where little data are present to guide the algorithm.

5. Discussion and conclusions

We have developed an adaptive methodology for the sample-based OT problem under the standard quadratic cost function. The main advantage of the new procedure is that it does not require any external input on the form of the distributions that one seeks to match or any expert knowledge on the type, location and size of the features in which the source and target distribution may differ.

Even though the map $\nabla\phi$ and test function g used at each step are parametric, by using the composition of many simple maps and having at one's disposal a 'lens' within g that can focus on any individual local mismatch at each step, the resulting procedure can be thought as effectively free of parameters, except for the number of intermediate distributions to use, a stopping criterion and a couple of constants associated with the penalization of the nonlinear features. Thus, it has the potential to form the basis for a universal tool that can be transferred painlessly across fields.

Two main ingredients allow for the procedure to capture arbitrary variability without making use of a huge dictionary of candidate features (in its current version, it uses only three: a linear feature for global displacements, a quadratic feature for global scalings and a Gaussian feature for localized displacements). One ingredient, borrowed from prior work in [6], is the factorization of the potentially quite complex global map into a sequence of much simpler local maps between nearby distributions. The optimality of the composed map is guaranteed through the use of displacement interpolation. The second ingredient is the formulation of the local problem as a two-player game, where the first player seeks to push forward one distribution into the other, while the second player develops features that show where the push-forward condition fails. The variational characterization of the relative entropy between distributions that gives rise to this game-theory formulation has the additional advantage of being sample-friendly, as it involves the two distributions only through the expected values of functions, which can be naturally replaced by empirical means. Because the map between any two consecutive distributions is close to the identity, local optimality is guaranteed by requiring this map to be the gradient of a potential.

Topics for future research include the extension of the algorithm to transportation costs different from the squared distance and, for the purpose of more efficient computability, the optimization of the minimax solver and the parallelization of the computation of the local maps. Most of all, we believe, the use of the new methodology in real applications will shed light on the issues that require further work, which may include the development of features and penalizations suitable for efficiently capturing sharp edges or removed objects.

Acknowledgements

The authors would like to thank Yongxin Chen for connecting the variational formulation of the Kullback–Leibler divergence with the Donsker–Varadhan formula.

Funding

Morse-Sloan Foundation; National Science Foundation (DMS-1715753, DMS-1311833 to M.E.); Office of Naval Research (N00014-15-1-2355).

References

1. Chen, Y., Georgiou, T. T. & Pavon, M. (2016) On the relation between optimal transport and Schrödinger bridges: a stochastic control viewpoint. *J. Optim. Theory Appl.*, **169**, 671–691.
2. Cuturi, M. (2013) Sinkhorn distances: lightspeed computation of optimal transport. *Advances in Neural Information Processing Systems*. (C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani & K. Q. Weinberger eds), vol. 26. Curran Associates, Inc., pp. 2292–2300. http://papers.nips.cc/paper/4927-sinkhorn-distances-lightspeed-computation-of-optimal-transport.pdf.
3. Donsker, M. D. & Varadhan, S. S. (1975) Asymptotic evaluation of certain Markov process expectations for large time, i–iv. *Commun. Pure Appl. Math.*, **28**, 1–47.

4. Frogner, C., Zhang, C., Mobahi, H., Araya-Polo, M. & Poggio, T. A. (2015) Learning with a Wasserstein loss. *Advances in Neural Information Processing Systems* (C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama & R. Garnett), vol. 28. Curran Associates, Inc., pp. 2053–2061. http://papers.nips.cc/paper/5679-learning-with-a-wasserstein-loss.pdf.

5. Kantorovich, L. V. (1942) On the translocation of masses. *C. R. Acad. Bulgare Sci.*, **7**, 199–201.

6. Kuang, M. & Tabak, E. G. (2017) Sample-based optimal transport and barycenter problems. *Commun. Pure Appl. Math.* (in press).

7. Léonard, C. (2012) From the Schrödinger problem to the Monge–Kantorovich problem. *J. Funct. Anal.*, **262**, 1879–1920.

8. McCann, R. J. (1997) A convexity principle for interacting gases. *Adv. Math.*, **128**, 153–179.

9. Monge, G. (1781) *Mémoire sur la Théorie des Déblais et des Remblais*. France: De l'Imprimerie Royale.

10. Nguyen, X., Wainwright, M. J. & Jordan, M. I. (2010) Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Trans. Inform. Theory*, **56**, 5847–5861.

11. Nowozin, S., Cseke, B. & Tomioka, R. (2016) f-GAN: training generative neural samplers using variational divergence minimization. *Advances in Neural Information Processing Systems* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon & R. Garnett eds), vol. 29. Curran Associates, Inc., pp. 271–279. http://papers.nips.cc/paper/6066-f-gan-training-generative-neural-samplers-using-variational-divergence-minimization.pdf.

12. Pavon, M. (2017) A variational derivation of a class of bfgs-like methods. *Optimization*, **67**, 2081–2089. https://doi.org/10.1080/02331934.2018.1522635.

13. Peyre, G. & Cuturi, M. (2019) Computational optimal transport. *Found. Trends Mach. Learn.*, **11**, 355–2607. http://dx.doi.org/10.1561/2200000073.

14. Rockafellar, R. T. (1970) *Convex Analysis*. Princeton Mathematical Series. Princeton, NJ: Princeton University Press.

15. Sheather, S. J. & Jones, M. C. (1991) A reliable data-based bandwidth selection method for kernel density estimation. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, **53**, 683–690.

16. Silverman, B. W. (2018) *Density Estimation for Statistics and Data Analysis*. New York: Routledge.

17. Suzuki, T., Sugiyama, M., Sese, J. & Kanamori, T. (2008) Approximating mutual information by maximum likelihood density ratio estimation. *Proceedings of the Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery at ECML/PKDD 2008* (Y. Saeys, H. Liu, I. Inza, L. Wehenkel & Y. Van de Pee eds). Proceedings of Machine Learning Research. Antwerp, Belgium: PMLR, pp. 5–20. http://proceedings.mlr.press/v4/suzuki08a.html.

18. Tabak, E. G. & Trigila, G. (2018) Conditional expectation estimation through attributable components. *Inf. Inference*, **7**, 727–754. https://doi.org/10.1093/imaiai/iax023.

19. Tabak, E. & Trigila, G. (2018) Explanation of variability and removal of confounding factors from data through optimal transport. *Commun. Pure Appl. Math.*, **71**, 163–199.

20. Villani, C. (2003) *Topics in Optimal Transportation*, vol. **58**. Providence, Rhode Island: American Mathematical Society.

21. Wright, S. & Nocedal, J. (1999) *Numerical Optimization*, vol. 35. New York: Springer Science, pp. 67–68, 7.