Meta Optimal Transport

Brandon Amos [1] **Giulia Luise** [2] **Samuel Cohen** [1 3 4] **Ievgen Redko** [5 6]

Abstract

We study the use of amortized optimization to predict optimal transport (OT) maps from the input measures, which we call *Meta OT*. This helps repeatedly solve similar OT problems between different measures by leveraging the knowledge and information present from past problems to rapidly predict and solve new problems. Otherwise, standard methods ignore the knowledge of the past solutions and suboptimally re-solve each problem from scratch. We instantiate Meta OT models in discrete and continuous settings between grayscale images, spherical data, classification labels, and color palettes and use them to improve the computational time of standard OT solvers. Our source code is available at `http://github.com/facebookresearch/meta-ot`.

1. Introduction

Optimal transportation (Villani, 2009; Ambrosio, 2003; Santambrogio, 2015; Peyré et al., 2019; Merigot and Thibert, 2021) is thriving in domains including economics (Galichon, 2016), reinforcement learning (Dadashi et al., 2021; Fickinger et al., 2022), style transfer (Kolkin et al., 2019), generative modeling (Arjovsky et al., 2017; Seguy et al., 2018; Huang et al., 2021; Rout et al., 2022), geometry (Solomon et al., 2015; Cohen et al., 2021), domain adaptation (Courty et al., 2017; Redko et al., 2019), signal processing (Kolouri et al., 2017), fairness (Jiang et al., 2019), and cell reprogramming (Schiebinger et al., 2019). These settings couple two measures (α, β) supported on domains $(\mathcal{X}, \mathcal{Y})$ by solving a transport optimization problem such as the *primal Kantorovich problem* defined by

$$\pi^\star(\alpha,\beta,c) \in \operatorname*{arg\,min}_{\pi\in\mathcal{U}(\alpha,\beta)} \int_{\mathcal{X}\times\mathcal{Y}} c(x,y)\mathrm{d}\pi(x,y), \qquad (1)$$

[1]Meta AI [2]Microsoft Research [3]University College London [4]Fairgen [5]Noah's Ark Lab, Huawei [6]Aalto University. Correspondence to: Brandon Amos <bda@meta.com>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

where the *optimal coupling* $\pi^\star$ is a joint distribution over the product space, $\mathcal{U}(\alpha,\beta)$ is the set of admissible couplings between α and β, and $c : \mathcal{X}\times\mathcal{Y}\to\mathbb{R}$ is the *ground cost*, that represents a notion of distance between elements in $\mathcal{X}$ and elements in $\mathcal{Y}$.

Challenges. Unfortunately, solving eq. (1) *once* is computationally expensive between general measures and computationally cheaper alternatives are an active research topic: *Entropic optimal transport* (Cuturi, 2013) smooths the transport problem with an entropy penalty, and *sliced distances* (Kolouri et al., 2016; 2019b;a; Deshpande et al., 2019) solve OT between 1-dimensional projections of the measures, where eq. (1) can be solved easily.

When an optimal transport method is deployed in practice, eq. (1) is not just solved once, but is *repeatedly* solved for new scenarios between different input measures (α, β). For example, the measures could be representations of images we care about optimally transporting between and in deployment we would receive a stream of new images to couple. Repeatedly solving optimal transport problems also comes up in the context of comparing seismic signals (Engquist and Froese, 2013) and in single-cell perturbations (Bunne et al., 2021; 2022b;a). Standard optimal transport solvers deployed in this setting re-solve the optimization problems from scratch and ignore the shared structure and information present between different coupling problems.

Overview. We study the use of amortized optimization and machine learning methods to rapidly solve multiple optimal transport problems and predict the solution from the input measures (α, β). This setting involves learning a *meta* model to predict the solution to the optimal transport problem, which we will refer to as *Meta Optimal Transport*. We learn Meta OT models to predict the solutions to optimal transport problems and significantly improve the computational time and number of iterations needed to solve eq. (1).

Settings that are not Meta OT. Meta OT is not useful in settings that do *not repeatedly* solve OT problems, e.g. 1) generative modeling settings, such as Arjovsky et al. (2017), that estimate the OT distance between the data and model distributions, and 2) out-of-sample settings (Seguy et al., 2018; Perrot et al., 2016) that couple measures and then extrapolate the map to larger measures.

2. Preliminaries and background

2.1. Entropic OT between discrete measures

We review foundations of OT, following the notation of Peyré et al. (2019) in most places. The discrete setting often favors the entropic regularized version since it can be computed efficiently and in a parallelized way using the Sinkhorn algorithm. While the primal Kantorovich formulation in eq. (1) provides an intuitive problem description, OT problems are rarely solved directly in this form due to the high-dimensionality of the couplings π and the difficulty of satisfying the coupling constraints $\mathcal{U}(\alpha, \beta)$. Instead, most computational OT solvers use the *dual* of eq. (1), which we build our Meta OT solvers on top of.

Let $\alpha := \sum_{i=1}^{m} a_i \delta_{x_i}$ and $\beta := \sum_{i=1}^{n} b_i \delta_{y_i}$ be *discrete* measures, where δ_z is a Dirac at point z and $a \in \Delta_{m-1}$ and $b \in \Delta_{n-1}$ are in the *probability simplex* defined by

$$\Delta_{k-1} := \{x \in \mathbb{R}^k : x \geq 0 \text{ and } \sum_i x_i = 1\}. \tag{2}$$

Discrete OT. Eq. (1) becomes the *linear program*

$$P^\star(\alpha, \beta, c) \in \arg\min_{P \in U(a,b)} \langle C, P \rangle \tag{3}$$

where $U(a,b) := \{P \in \mathbb{R}_+^{n \times m} : P 1_m = a, \quad P^\top 1_n = b\}$, P is a *coupling matrix*, $P^\star(\alpha, \beta)$ is the *optimal* coupling, and the *cost* can be discretized as a matrix $C \in \mathbb{R}^{m \times n}$ with entries $C_{i,j} := c(x_i, y_j)$, and $\langle C, P \rangle := \sum_{i,j} C_{i,j} P_{i,j}$,

Entropic OT. The linear program above can be regularized adding an entropy term to smooth the objective as in Cominetti and Martín (1994); Cuturi (2013), resulting in:

$$P^\star(\alpha, \beta, c, \epsilon) \in \arg\min_{P \in U(a,b)} \langle C, P \rangle - \epsilon H(P) \tag{4}$$

where $H(P) := -\sum_{i,j} P_{i,j}(\log(P_{i,j}) - 1)$ is the discrete entropy of a coupling matrix P.

Entropic OT dual. As presented in Peyré et al. (2019, Prop. 4.4), setting $K \in \mathbb{R}^{m \times n}$ to the *Gibbs kernel* $K_{i,j} := \exp\{-C_{i,j}/\epsilon\}$, the dual of eq. (4) is

$$f^\star, g^\star \in \arg\max_{f \in \mathbb{R}^n, g \in \mathbb{R}^m} \langle f, a \rangle + \langle g, b \rangle - \epsilon e^{f/\epsilon} K e^{g/\epsilon} \tag{5}$$

where the *dual variables* or *potentials* $f \in \mathbb{R}^n$ and $g \in \mathbb{R}^m$ are associated, respectively, with the marginal constraints $P 1_m = a$ and $P^\top 1_n = b$. We omit the dependencies of the duals on the context, e.g. $f^\star$ is shorthand for $f^\star(\alpha, \beta, c, \epsilon)$.

Recovering the primal solution from the duals. Given optimal duals $f^\star$, $g^\star$ that solve eq. (5) the optimal coupling $P^\star$ to the primal problem in eq. (4) can be obtained by

$$P^\star_{i,j}(\alpha, \beta, c, \epsilon) := \exp\{f_i^\star/\epsilon\} K_{i,j} \exp\{g_j^\star/\epsilon\}. \tag{6}$$

The Sinkhorn algorithm. Algorithm 1 summarizes the log-space version, which takes closed-form block coordinate ascent updates on eq. (5) obtained from the first-order optimality conditions (Peyré et al., 2019, Remark 4.21). We will fine-tune Meta OT predictions with Sinkhorn.

Algorithm 1 Sinkhorn($\alpha, \beta, c, \epsilon, f_0 = 0$)

> **for** iteration i = 1 to N **do**
> $\quad g_i \leftarrow \epsilon \log b - \epsilon \log \left(K^\top \exp\{f_{i-1}/\epsilon\}\right)$
> $\quad f_i \leftarrow \epsilon \log a - \epsilon \log \left(K \exp\{g_i/\epsilon\}\right)$
> **end for**
> Compute P_N from f_N, g_N using eq. (6)
> **return** $P_N \approx P^\star$

Algorithm 2 W2GN(α, β, φ_0)

> **for** iteration i = 1 to N **do**
> $\quad$ Sample from (α, β) and estimate $\mathcal{L}(\varphi_{i-1})$ (eq. (13))
> $\quad$ Update φ_i with approximation to $\nabla_\varphi \mathcal{L}(\varphi_{i-1})$
> **end for**
> **return** $T_N(\cdot) := \nabla_x \psi_{\varphi_N}(\cdot) \approx T^\star(\cdot)$

Computing the error. Standard implementations of the Sinkhorn algorithm, such as Flamary et al. (2021); Cuturi et al. (2022), measure the error of a candidate dual solution (f, g) by computing the deviation from the marginals:

$$\mathrm{err}(f, g; \alpha, \beta, c) := \|P 1_m - a\|_1 + \|P^\top 1_n - b\|_1, \tag{7}$$

where P is computed from eq. (6).

Mapping between the duals. The first-order optimality conditions of eq. (5) also provide an equivalence between the optimal dual potentials that we will make use of

$$g(f; b, c) := \epsilon \log b - \epsilon \log \left(K^\top \exp\{f/\epsilon\}\right). \tag{8}$$

2.2. OT between continuous (Euclidean) measures

Let α and β be continuous measures in Euclidean space $\mathcal{X} = \mathcal{Y} = \mathbb{R}^d$ (with α absolutely continuous with respect to the Lebesgue measure) and the ground cost be the squared Euclidean distance $c(x, y) := \|x - y\|_2^2$. Then the minimum of eq. (1) defines the square of the *Wasserstein-2* distance:

$$W_2^2(\alpha, \beta) := \min_{\pi \in \mathcal{U}(\alpha, \beta)} \int_{\mathcal{X} \times \mathcal{Y}} \|x - y\|_2^2 \mathrm{d}\pi(x, y) \tag{9}$$

$$= \min_T \int_{\mathcal{X}} \|x - T(x)\|_2^2 \mathrm{d}\alpha(x), \tag{10}$$

where T is a *transport map* pushing α to β, i.e. $T_\# \alpha = \beta$ with the *pushforward operator* defined by $T_\# \alpha(B) := \alpha(T^{-1}(B))$ for any measurable set B.

Convex dual potentials. The primal in eq. (10) is difficult to solve due to the constraints and many computational methods (Makkuva et al., 2020; Taghvaei and Jalali, 2019; Korotin et al., 2021a;c; 2022; Amos, 2023) solve the dual

$$\psi^\star(\,\cdot\,; \alpha, \beta) \in \arg\min_{\psi \in \text{convex}} \int_{\mathcal{X}} \psi(x) \mathrm{d}\alpha(x) + \int_{\mathcal{Y}} \overline{\psi}(y) \mathrm{d}\beta(y), \tag{11}$$

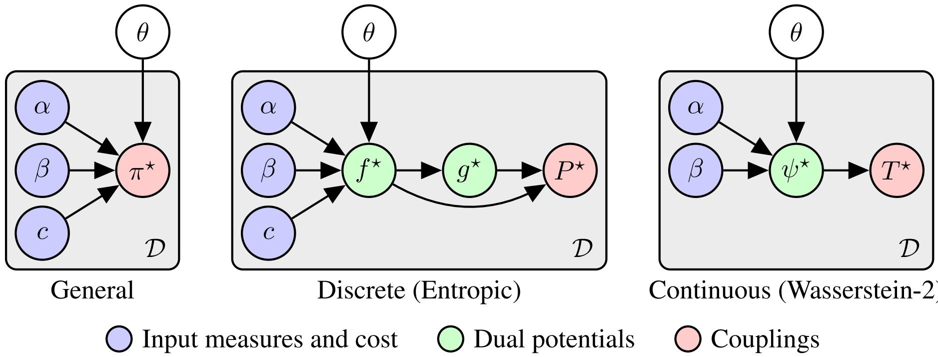

Figure 1. Meta OT uses objective-based amortization for optimal transport. In the general formulation, the *parameters* θ capture shared structure in the *optimal couplings* $\pi^\star$ between multiple input measures and costs over some *distribution* $\mathcal{D}$. In practice, we learn this shared structure over the *dual potentials* which map back to the coupling: $f^\star$ in discrete settings and $\psi^\star$ in continuous ones.

where ψ is a convex function referred to as a *potential*, and $\overline{\psi}(y) := \max_{x\in\mathcal{X}} \langle x, y\rangle - \psi(x)$ is the *Legendre-Fenchel transform* or *convex conjugate* of ψ (Fenchel, 1949; Rockafellar, 2015). The potential may be approximated with an input-convex neural network (ICNN) (Amos et al., 2017).

Recovering the primal solution from the dual. Given an optimal dual $\psi^\star$ for eq. (11), Brenier (1991) shows that an optimal map $T^\star$ for eq. (10) can be obtained with

$$T^\star(x) = \nabla_x \psi^\star(x). \tag{12}$$

Wasserstein-2 Generative Networks (W2GNs). Korotin et al. (2021a) model ψ_φ and $\overline{\psi_\varphi}$ in eq. (11) with two separate ICNNs parameterized by φ. The separate model for $\overline{\psi_\varphi}$ is useful because the conjugate operation in eq. (11) becomes computationally expensive. They optimize the loss

$$\mathcal{L}(\varphi) := \underbrace{\mathop{\mathbb{E}}_{x\sim\alpha}[\psi_\varphi(x)] + \mathop{\mathbb{E}}_{y\sim\beta}\left[\langle\nabla\overline{\psi_{\not\varphi}}(y), y\rangle - \psi_\varphi(\nabla\overline{\psi_{\not\varphi}}(y))\right]}_{\text{Cyclic monotone correlations (dual objective)}} + \gamma \underbrace{\mathop{\mathbb{E}}_{y\sim\beta}\|\nabla\psi_\varphi \circ \nabla\overline{\psi_\varphi}(y) - y\|_2^2}_{\text{Cycle-consistency regularizer}}, \tag{13}$$

where $\not\varphi$ is a detached copy of the parameters and γ is a hyper-parameter. The first term are the *cyclic monotone correlations* (Chartrand et al., 2009; Taghvaei and Jalali, 2019), that optimize the dual objective in eq. (11), and the second term provides *cycle consistency* (Zhu et al., 2017) to estimate the conjugate $\overline{\psi}$. Algorithm 2 shows how $\mathcal{L}$ is typically optimized using samples from the measures, which we use to fine-tune Meta OT predictions.

2.3. Amortized optimization and learning to optimize

Our paper is an application of amortized optimization methods that predict the solutions of optimization problems, as surveyed in, e.g., Chen et al. (2021); Amos (2022). We use the setup from Amos (2022), which considers unconstrained continuous optimization problems

$$z^\star(\phi) \in \arg\min_z J(z; \phi), \tag{14}$$

where J is the objective, $z \in \mathcal{Z}$ is the *domain*, and $\phi \in \Phi$ is some *context* or *parameterization*. In other words, the context conditions the objective but is not optimized over. Given a *distribution over contexts* $\mathcal{P}(\phi)$, we learn a model $\hat{z}_\theta$ parameterized by θ to approximate eq. (14), i.e. $\hat{z}_\theta(\phi) \approx z^\star(\phi)$. J will be differentiable, so we optimize the parameters using *objective-based learning* with

$$\min_\theta \mathop{\mathbb{E}}_{\phi\sim\mathcal{P}(\phi)} J(\hat{z}_\theta(\phi); \phi), \tag{15}$$

which does *not* require ground-truth solutions $z^\star$ and can be optimized with a gradient-based solver.

3. Meta Optimal Transport

We refer to Meta Optimal Transport as the setting when amortized optimization (sect. 2.3) is used for predicting solutions to optimal transport problems such as eq. (1). We refer to the distribution over the OT problems (measures and costs) as the *meta-distribution* and denote it as $\mathcal{D}(\alpha, \beta, c)$, which we call *meta* to distinguish it from the measures α, β. For example, sects. 4.1.1 and 4.1.2 considers meta-distributions over the weights of the atoms, i.e. $(a, b) \sim \mathcal{D}$, where $\mathcal{D}$ is a distribution over $\Delta_{m-1} \times \Delta_{n-1}$. While a model could directly predict the primal solution to eq. (1), i.e. $P_\theta(\alpha, \beta, c) \approx P^\star(\alpha, \beta, c)$ for $(\alpha, \beta, c) \sim \mathcal{D}$, this is difficult due to the coupling constraints. We instead opt to predict the dual variables. Figure 1 illustrates Meta OT in discrete and continuous settings.

3.1. Meta OT between discrete measures

We build on standard methods for entropic OT reviewed in sect. 2.1 between discrete measures $\alpha := \sum_{i=1}^m a_i\delta_{x_i}$ and $\beta := \sum_{i=1}^n b_i\delta_{x_i}$ with $a \in \Delta_{m-1}$ and $b \in \Delta_{n-1}$ coupled using a cost c. In the Meta OT setting, the measures and cost are the contexts for amortization and sampled from a *meta-distribution*, i.e. $(\alpha, \beta, c) \sim \mathcal{D}(\alpha, \beta, c)$. For example, sects. 4.1.1 and 4.1.2 considers meta-distributions over the weights of the atoms, i.e. $(a, b) \sim \mathcal{D}$, where $\mathcal{D}$ is a distribution over $\Delta_{m-1} \times \Delta_{n-1}$.

Amortization objective. We will seek to predict the *optimal* potential. At optimality, the pair of potentials are related to each other via eq. (8), i.e. $g(f; \alpha, \beta, c) := \epsilon \log b - \epsilon \log \left(K^\top \exp\{f/\epsilon\}\right)$ where $K \in \mathbb{R}^{m \times n}$ is the *Gibbs kernel* from eq. (5). Hence, it is sufficient to predict one of the potentials, e.g. f, and recover the other. We thus re-formulate eq. (5) to just optimize over f with

$$f^\star(\alpha, \beta, c, \epsilon) \in \operatorname*{arg\,min}_{f \in \mathbb{R}^n} J(f; \alpha, \beta, c), \qquad (16)$$

where $-J(f; \alpha, \beta, c) := \langle f, a \rangle + \langle g, b \rangle - \epsilon \langle \exp\{f/\epsilon\}, K \exp\{g/\epsilon\} \rangle$ is the (negated) dual objective. Even though most solvers optimize over f and g jointly as in eq. (16), amortizing over these would likely need to have a higher capacity than a model just predicting f and learn how f and g are connected through eq. (8) while in eq. (16) we explicitly provide this knowledge.

Amortization model. We predict the solution to eq. (16) with $\hat{f}_\theta(\alpha, \beta, c)$ parameterized by θ, resulting in a computationally efficient approximation $\hat{f}_\theta \approx f^\star$. Here we use the notation $\hat{f}_\theta(\alpha, \beta, c)$ to mean that the model $\hat{f}_\theta$ depends on *representations* of the input measures and cost. In our settings, we define $\hat{f}_\theta$ as a fully-connected MLP mapping from the atoms of the measures to the duals.

Amortization loss. Applying objective-based amortization from eq. (15) to the dual in eq. (16) completes the learning setup. The model should optimize the expected dual value:

$$\min_\theta \mathop{\mathbb{E}}_{(\alpha, \beta, c) \sim \mathcal{D}} J(\hat{f}_\theta(\alpha, \beta, c); \alpha, \beta, c), \qquad (17)$$

which is appealing as it does not require ground-truth solutions $f^\star$. The ground-truth solutions may be expensive to obtain, but if they are available, a regression term can also be added (Amos, 2022). Algorithm 3 shows a basic training loop for eq. (17) using a gradient-based optimizer such as Adam (Kingma and Ba, 2015).

Sinkhorn fine-tuning. The dual prediction made by $\hat{f}_\theta$ with an associated $\hat{g}$ can be used to initialize a standard Sinkhorn solver. This allows for the predicted solution to be refined to an optimality threshold.

On accelerated solvers. While we have considered fine-tuning the Meta OT prediction with a log-Sinkhorn solver, Meta OT can also be combined with accelerated variants of entropic OT solvers such as Thibault et al. (2017); Altschuler et al. (2017); Alaya et al. (2019); Lin et al. (2019) that otherwise solve every problem from scratch.

Algorithm 3 Training Meta OT

```
Initialize amortization model with θ_0
for iteration i do
    Sample (α, β, c) ~ D
    Predict duals f̂_θ or φ̂_θ on the sample
    Estimate the loss in eq. (17) or eq. (18)
    Update θ_{i+1} with a gradient step
end for
```

Table 1. Sinkhorn runtime (seconds) to reach a marginal error of 10^{-2}. Meta OT's initial prediction takes $\approx 5 \cdot 10^{-5}$ seconds. We report the mean and std across 10 test instances.

Initialization	MNIST	Spherical
Zeros (t_{zeros})	$4.5 \cdot 10^{-3}$ $\pm 1.5 \cdot 10^{-3}$	0.88 ± 0.13
Gaussian	$4.1 \cdot 10^{-3}$ $\pm 1.2 \cdot 10^{-3}$	0.56 $\pm 9.9 \cdot 10^{-2}$
Meta OT (t_{Meta})	$2.3 \cdot 10^{-3}$ $\pm 9.2 \cdot 10^{-6}$	$7.8 \cdot 10^{-2}$ $\pm 3.4 \cdot 10^{-2}$
Improvement ($t_{\text{zeros}}/t_{\text{Meta}}$)	1.96	11.3

3.2. Meta OT between continuous measures

We take an analogous approach to predicting the Wasserstein-2 map between continuous measures for Wasserstein-2 as reviewed in sect. 2.2. Here the measures α, β are supported in continuous space $\mathcal{X} = \mathcal{Y} = \mathbb{R}^d$ and we focus on computing Wasserstein-2 couplings from instances sampled from a *meta-distribution* $(\alpha, \beta) \sim \mathcal{D}(\alpha, \beta)$. The cost c is not included in $\mathcal{D}$ as it remains fixed to the squared Euclidean cost everywhere here.

One challenge here is that the optimal dual potential $\psi^\star(\,\cdot\,; \alpha, \beta)$ in eq. (11) is a convex function and not simply a finite-dimensional real vector. The dual potentials in this setting are approximated by, e.g., an ICNN. We thus propose a *Meta ICNN* that predicts the *parameters* φ of an ICNN ψ_φ that approximates the optimal dual potentials, which can be seen as a hypernetwork (Stanley et al., 2009; Ha et al., 2017). The dual prediction made by $\hat{\varphi}_\theta$ can easily be input as the initial value to a standard W2GN solver. App. D.2 discusses other modeling choices we considered: we tried models based on MAML (Finn et al., 2017) and neural processes (Garnelo et al., 2018b;a).

Amortization objective. We build on the W2GN formulation (Korotin et al., 2021a) and seek parameters $\varphi^\star$ optimizing the dual ICNN potentials ψ_φ and $\overline{\psi_\varphi}$ with $\mathcal{L}(\varphi; \alpha, \beta)$ from eq. (13). We chose W2GN due to the stability, but could also easily use other losses optimizing ICNN potentials.

Amortization model: the Meta ICNN. We predict the solution to eq. (13) with $\hat{\varphi}_\theta(\alpha, \beta)$ parameterized by θ, resulting in a computationally efficient approximation to the optimum $\hat{\varphi}_\theta \approx \varphi^\star$. Figure 11 instantiates a convolutional Meta ICNN model using a ResNet-18 (He et al., 2016) architecture for coupling image-based measures. We again emphasize that α, β used with the model here are *represen-*

Figure 2. Interpolations between MNIST test digits using couplings obtained from (left) solving the problem with Sinkhorn, and (right) Meta OT model's initial prediction, which is $\approx$**100** times computationally cheaper and produces a nearly identical coupling.

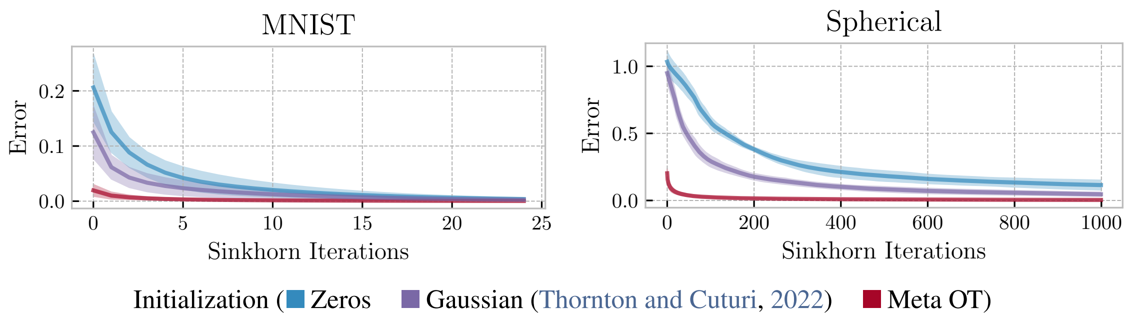

Figure 3. Meta OT successfully predicts warm-start initializations that significantly improve the convergence of Sinkhorn iterations on test data. The error is the marginal error defined in eq. (7).

tations of measures, which in our cases are simply images.

Amortization loss. Applying objective-based amortization from eq. (15) to the W2GN loss in eq. (13) completes our learning setup. We optimize the loss

$$\min_{\theta} \mathop{\mathbb{E}}_{(\alpha,\beta)\sim\mathcal{D}} \mathcal{L}(\hat{\varphi}_\theta(\alpha,\beta);\alpha,\beta). \tag{18}$$

As in the discrete setting, this loss does not require ground-truth solutions $\varphi^\star$ and we find the solution with Adam.

4. Experiments

We demonstrate how Meta OT models improve the convergence of the state-of-the-art solvers in settings where solving multiple OT problems naturally arises. We implemented our code in JAX (Bradbury et al., 2018) as an extension to the the Optimal Transport Tools (OTT) package (Cuturi et al., 2022). App. B covers further experimental and implementation details, and shows that all of our experiments take a few hours to run on our single Quadro GP100 GPU. The source code to reproduce all of our experiments is available at `http://github.com/facebookresearch/meta-ot`.

4.1. Discrete OT

4.1.1. Grayscale image transport

Images provide a natural setting for Meta OT where the distribution over images provide the meta-distribution $\mathcal{D}$ over OT problems. Given a pair of images α_0 and α_1, each grayscale image is cast as a discrete measure in 2-dimensional space where the intensities define the probabilities of the atoms. The goal is to compute the optimal transport interpolation between the two measures as in, e.g., Peyré et al. (2019, §7). Formally, this means computing the optimal coupling $P^\star$ by solving the entropic optimal transport problem between α_0 and α_1 and computing the interpolates as $\alpha_t = (t\,\mathrm{proj}_y + (1-t)\,\mathrm{proj}_x)_\# P^\star$, for $t \in [0,1]$, where $\mathrm{proj}_x(x,y) := x$ and $\mathrm{proj}_y(x,y) := y$. We selected $\epsilon = 10^{-2}$ as app. A shows that it gives interpolations that are not too blurry or sharp.

Our Meta OT model $\hat{f}_\theta$ (sect. 3) is an MLP that predicts the transport map between pairs of MNIST digits. We train on every pair from the standard training dataset. Figure 2 shows that even without fine-tuning, Meta OT's predicted Wasserstein interpolations between the measures are close to the ground-truth interpolations obtained from running the Sinkhorn algorithm to convergence. We then fine-tune Meta OT's prediction with Sinkhorn. Figure 3 shows that the near-optimal predictions can be quickly refined in fewer iterations than running Sinkhorn with the default initialization, and table 1 shows the runtime required to reach an error threshold of 10^{-2}, showing that the Meta OT initialization help solve the problems faster by an order of magnitude. We compare our learned initialization to the standard zero initialization, as well as the Gaussian initialization proposed in Thornton and Cuturi (2022), which takes a continuous Gaussian approximation of the measures and initializes the potentials to be the known coupling between the Gaussians. This Gaussian initialization assumes the squared Euclidean cost, which is not the case in our spherical transport problem, but we find it is still helpful over the zero initialization.

Out-of-distribution generalization We now test the abil-

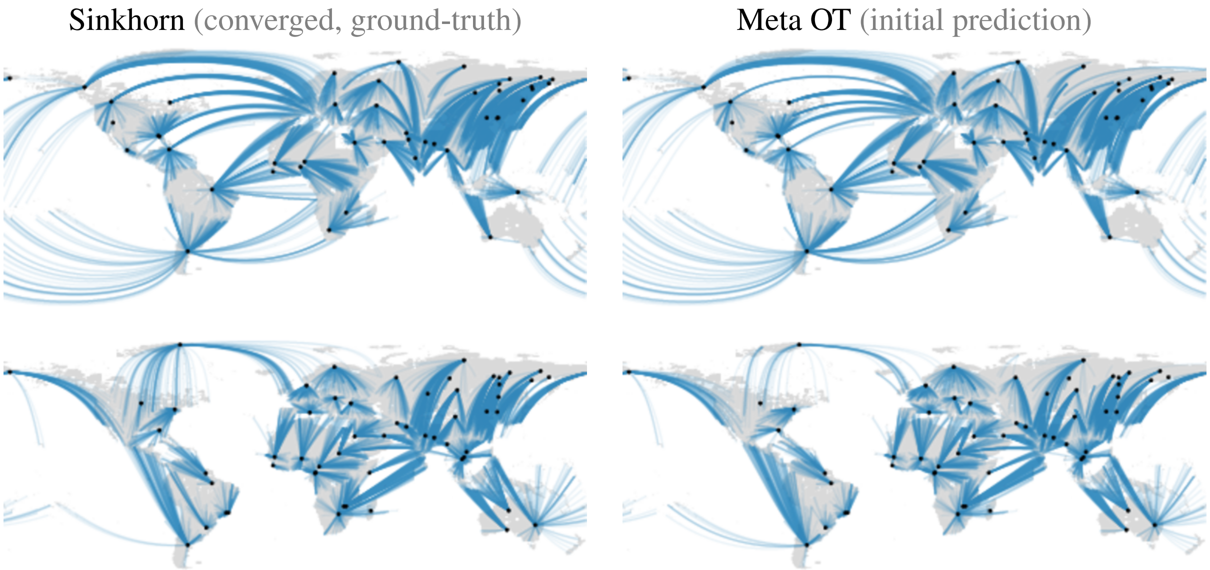

Figure 4. Test set coupling predictions of the spherical transport problem. Meta OT's initial prediction is $\approx\mathbf{37500}$ times faster than solving Sinkhorn to optimality. Supply locations are shown as black dots and the blue lines show the spherical transport maps T going to demand locations at the end. The sphere is visualized with the Mercator projection.

ity of Meta OT to predict potentials for out-of-distribution input data. We consider the pairwise training and evaluation on the following datasets: 1) MNIST; 2) USPS (Hull, 1994) (upscaled to have the same size as the MNIST); 3) Google Doodles dataset* with classes Crab, Cat and Faces; 4) sparsified random uniform data in [0,1] where sparsity (zeroing values below 0.95) is used to mimic the sparse signal in black-and-white images. For each pair, eg, MNIST-USPS, we train on one dataset and use the other to predict the potentials. The comparison is done using the same metric as before, i.e., the deviation from the marginal constraints defined in eq. (7). App. C shows how well the learned models are capable of transferring to new domains.

4.1.2. Supply-demand transportation on spherical data

We next set up a synthetic transport problem between supply and demand locations where the supply and demands may change locations or quantities frequently, creating another Meta OT setting to be able to rapidly solve the new instances. We specifically consider measures living on the 2-sphere defined by $\mathcal{S}_2 := \{x \in \mathbb{R}^3 : \|x\| = 1\}$, i.e. $\mathcal{X} = \mathcal{Y} = \mathcal{S}_2$, with the transport cost given by the spherical distance $c(x, y) = \arccos(\langle x, y\rangle)$. We then randomly sample supply locations uniformly from Earth's landmass and demand locations from Earth's population density to induce a class of transport problems on the sphere obtained from the CC-licensed dataset from Doxsey-Whitfield et al. (2015). Figure 4 shows that the predicted transport maps on test instances are close to the optimal maps obtained from Sinkhorn to convergence. Similar to the MNIST setting, fig. 3 and table 1 show improved convergence and runtime.

*https://quickdraw.withgoogle.com/data

4.1.3. Wasserstein adversarial regularization

Wasserstein losses has recently attracted a considerable attention in the field of multi-label (Frogner et al., 2015; Yang et al., 2018; Jawanpuria et al., 2021; Toyokuni et al., 2021) and multi-class classification (Liu et al., 2020a;b; 2019; Han et al., 2020; Fatras et al., 2021) as they both require finding an informative way of comparing discrete distributions given by the true labeling of the data points and those predicted by the classification model. In this experiment, we aim to show that meta OT model can be learned alongside the training of the multi-class classification model and used to make predictions for the Wasserstein loss term appearing in the objective function of the latter. For this, we consider as an example the setup of Fatras et al. (2021) where the authors define an adversarial loss term (called WAR) aiming at limiting the effect of label noise on the generalization capacity of deep vision neural networks. In particular, given a neural network p_θ predicting a vector of class memberships in $\mathbb{R}^c$, the regularization term is

$$\begin{aligned} R_{\text{WAR}}(x_i) &= W_C^\epsilon(p_\theta(x_i + r_i^a), p_\theta(x_i)) \\ r_i^a &= \operatorname*{arg\,max}_{r_i, ||r_i|| \leq \varepsilon} W_C^\epsilon(p_\theta(x_i + r_i), p_\theta(x_i)). \end{aligned} \tag{19}$$

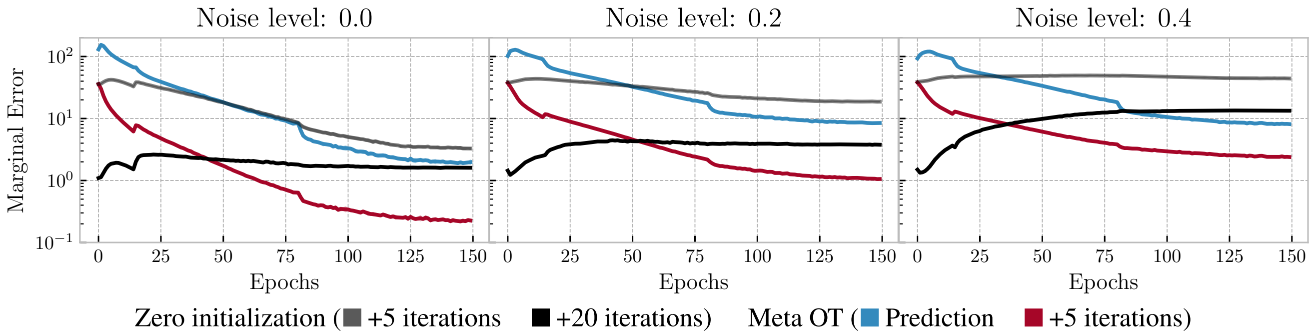

Figure 5. Marginal errors throughout WAR training for CIFAR-100 classification. The bumps correspond to when the loss and learning rate are updated during training as described in Fatras et al. (2021)

Dataset	Noise level	WAR (5 iter.)	Methods WAR (20 iter.)	Meta OT + 5 iter.
Fashion MNIST	0%	94.75 $\pm$0.05	94.16 $\pm$0.02	94.70 $\pm$0.05
	20%	93.00 $\pm$0.15	93.41 $\pm$0.08	93.53 $\pm$0.01
	40%	88.67 $\pm$0.10	88.08 $\pm$0.10	89.08 $\pm$0.61
Cifar-10	0%	91.96 $\pm$0.13	91.76 $\pm$0.25	91.98 $\pm$0.16
	20%	88.80 $\pm$0.11	90.59 $\pm$0.05	90.13 $\pm$0.21
	40%	81.09 $\pm$0.06	87.07 $\pm$0.08	83.57 $\pm$1.13
Cifar-100	0%	70.93 $\pm$0.23	69.79 $\pm$0.34	70.25 $\pm$0.04
	20%	66.23 $\pm$0.29	66.18 $\pm$0.18	66.59 $\pm$0.38
	40%	52.69 $\pm$0.12	61.63 $\pm$0.33	61.13 $\pm$0.17

Table 2. Comparison of the original WAR implementation with WAR implementation using only 5 Sinkhorn iterations and our Meta OT model with 5 Sinkhorn iterations on top of initial predictions. We report the mean and std across 3 random seeds.

where W_C^ϵ is the Wasserstein distance with entropic regularization introduced in eq. (4) with a cost matrix $C \in R^{c\times c}$. Learning p_θ is done by optimizing the cross entropy loss together with $R_{\text{WAR}}(x_i)$ using stochastic optimization. This means that OT problems in eq. (19) are solved repeatedly, for every batch in the input dataset and during multiple epochs thus making the meta OT warm-starts particularly computationally attractive in this context. For this task, we optimize a meta OT model defined as a MLP with 3 hidden layers over the same data alongside the main optimization procedure. We use meta OT model to predict the solutions to both OT problems in eq. (19) and use only 25% of iterations in the Sinkhorn loop to compute W_C^ϵ. As in Fatras et al. (2021), we evaluate the efficiency of such learning strategy on three computer vision datasets, namely: Fashion MNIST, Cifar-10 and Cifar-100. For each of them, we consider the clean version of the data (0% noise), and two variations with 20% and 40% of noise in labels. The authors of Fatras et al. (2021) experiment with two cost matrices: one is defined based on the distances between the class centroids of 30000 samples from the original dataset when embedded with ResNet18; second one is defined as the Euclidean distance between the word2vec embeddings of the classes of the original dataset. To show the versatility of our approach with respect to different geometries, we use the first cost matrix for Fashion MNIST dataset, and the second one for Cifar-10 and Cifar-100 datasets.

We evaluate meta OT for this task based on three criteria. First, we want to make sure that reducing the number of iterations in the Sinkhorn loop is not detrimental for the overall performance of the learned classification model. These results are presented in table 2, where we can see that meta OT leads to the same performance as the original WAR model while doing only 5 iterations of Sinkhorn on top of the initial predictions. Second, we show in sect. 4.1.2 that our meta OT model predicts warm-start initializations that have a low marginal error so that even its initial predictions are become at least as qualitative as the solution obtained using 20 iterations of the Sinkhorn algorithm. Finally, we show in table 3, that training a meta OT model alongside the main model doesn't introduce any additional overhead in terms of computational time. In this table, we compare the average runtime of each of the considered baselines and account for the time needed to make a backward pass for the meta OT model. This result is important as once a meta OT model is trained, it can be further used to make predictions without any finetuning for other training runs with different hyperparameters leading to an important reduction in terms of computational time.

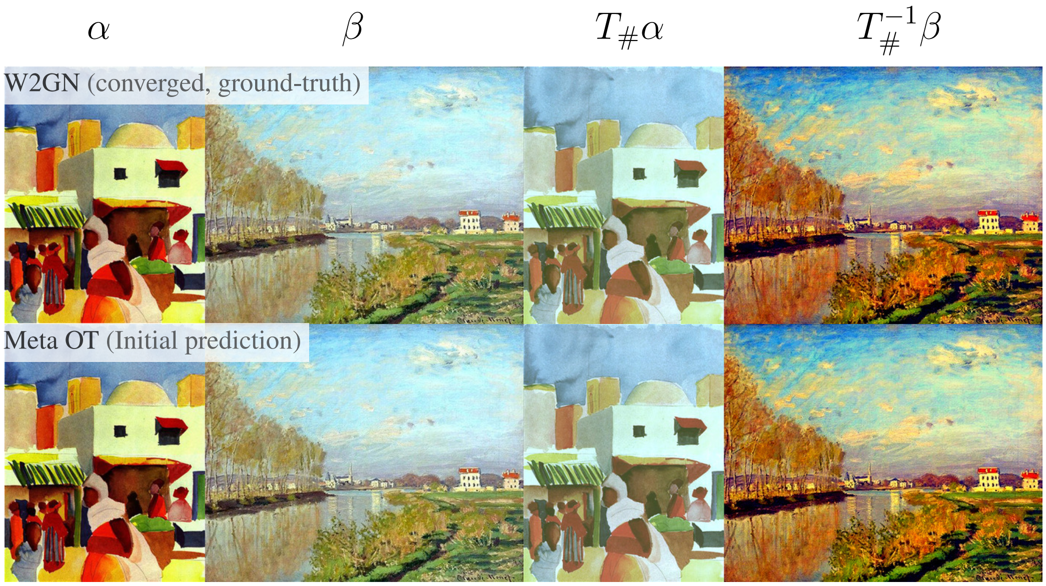

Figure 6. Color transfers with a Meta ICNN on test pairs of images. The objective is to optimally transport the continuous RGB measure of the first image α to the second β, producing an invertible transport map T. Meta OT's prediction is $\approx$**1000** times faster than training W2GN (Korotin et al., 2021a) from scratch. The image generating α is Market in Algiers by August Macke (1914) and β is Argenteuil, The Seine by Claude Monet (1872), obtained from WikiArt.

Table 3. Runtime (s) per epoch of training of the WAR model.

	Iter	Fashion MNIST	Cifar-10	Cifar-100
Zero Init	5	11.51 $\pm$0.07	12.88 $\pm$0.02	13.13 $\pm$0.02
Meta OT	5	14.37 $\pm$0.04	14.07 $\pm$0.15	14.17 $\pm$0.02
Zero Init	20	17.04 $\pm$0.12	14.36 $\pm$0.02	14.37 $\pm$0.02

Table 4. Color transfer runtimes and values. We report the mean and std across 10 test instances.

	Iter	Runtime (s)	Dual Value
Meta OT	None	$3.5 \cdot 10^{-3} \pm 2.7 \cdot 10^{-4}$	$0.90 \pm 6.08 \cdot 10^{-2}$
+ W2GN	1k	$0.93 \pm 2.27 \cdot 10^{-2}$	$1.0 \pm 2.57 \cdot 10^{-3}$
	2k	$1.84 \pm 3.78 \cdot 10^{-2}$	$1.0 \pm 5.30 \cdot 10^{-3}$
W2GN	1k	$0.90 \pm 1.62 \cdot 10^{-2}$	$0.96 \pm 2.62 \cdot 10^{-2}$
	2k	$1.81 \pm 3.05 \cdot 10^{-2}$	$0.99 \pm 1.14 \cdot 10^{-2}$

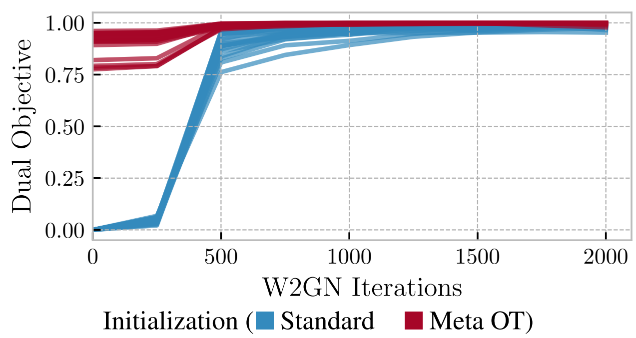

Figure 7. Convergence on color transfer test instances using W2GN. Meta ICNNs predicts warm-start initializations that significantly improve the (normalized) dual objective values.

4.2. Continuous OT for color transfer

The problem of color transfer between two images consists in mapping the color palette of one image into the other one. The images are required to have the same number of channels, for example RGB images. The continuous formulation that we use from Korotin et al. (2021a), takes i.e. $\mathcal{X} = \mathcal{Y} = [0, 1]^3$ with c being the squared Euclidean distance. We collected $\approx$200 public domain images from WikiArt and trained a Meta ICNN model from sect. 3.2 to predict the color transfer maps between every pair of them. Figure 6 shows the predictions on test pairs and fig. 7 shows the convergence in comparison to the standard W2GN learning. Table 4 reports runtimes. App. D.3 show additional color transfer App. D.2

5. Related work

Efficiently estimating OT maps. To compute OT maps with fixed cost between pairs of measures efficiently, neural OT models (Korotin et al., 2021a;b; Mokrov et al., 2021; Korotin et al., 2021c) leverage ICNNs to estimate maps between continuous high-dimensional measures given samples from these, and Litvinenko et al. (2021); Scetbon et al. (2021); Forrow et al. (2019); Sommerfeld et al. (2019); Scetbon et al. (2022); Muzellec and Cuturi (2019); Bonet et al. (2021) leverage structural assumptions on coupling and cost matrices to reduce the computational and memory complexity. In the meta-OT setting, we consider learning to rapidly compute OT mappings between new pairs measures. All these works can hence benefit from an acceleration effect with amortization.

Embedding measures where OT distances are discriminative. Effort has been invested in learning encodings/projections of measures through a nested optimization problem, which aims to find discriminative embeddings of the measures to be compared (Genevay et al., 2018; Deshpande et al., 2019; Nguyen and Ho, 2022). While these works share an encoder and/or a projection across task with the aim of leveraging more discriminative alignments (and hence an OT distance with a metric different from the Euclidean metric), our work differs in the sense that we find good initializations to solve the OT problem itself with fixed cost more efficiently across tasks.

Optimal transport and amortization. Courty et al. (2018) learn a latent space in which the Wasserstein distance between the measure's embeddings is equivalent to the Euclidean distance. Nguyen and Ho (2022) amortizes the estimation of the optimal projection in the max-sliced objective, which differs from our work where we instead amortize the estimation of the optimal coupling directly. Lacombe et al. (2021) learns to predict Wasserstein barycenters of pixel images by training a convolutional networks that, given images as input, outputs their barycenters. Our work is hence a generalization of this pixel-based work to general measures – both discrete and continuous. One limitation is that the barycenter predictions do not provide the optimal couplings. Gracyk and Chen (2022) learn a neural operator, e.g. from Kovachki et al. (2021); Li et al. (2020) to amortize the solution to the PDE from the dynamic OT formulation. Bunne et al. (2022a) predict the solutions to continuous neural OT problems.

6. Conclusions

We have presented foundations for modeling and learning to solve OT problems with Meta OT by using amortized optimization to predict optimal transport plans. This works best in applications that require solving multiple OT problems with shared structure. We instantiated it to speed up entropic regularized optimal transport and unregularized optimal transport with squared cost by multiple orders of magnitude. We envision extensions of the work in: 1) **Continuous settings**. Learning solutions continuous OT problems is a budding topic in the community: Gracyk and Chen (2022) amortize solutions to dynamic OT problems between continuous measures, and Bunne et al. (2022a) uses a partially input-convex neural network (PICNN) from Amos et al. (2017) to predict continuous OT solutions from contextual information. Related to these, app. D presents a more general extension of Meta OT and provides a demonstration on transferring color palettes, which is shown in fig. 6. Future directions for amortizing continuous OT problems include exploring modeling (PICNN vs. a hypernetwork), loss, and fine-tuning choices. 2) **Meta OT models**. While we mostly consider models based on hypernetworks, other meta-learning paradigms can be connected in. In the discrete setting, we only considered settings where the cost remains fixed, but the Meta OT model can also be conditioned on the cost by considering the entire cost matrix as an input (which may be too large for most models to handle), or considering a lower-dimensional parameterization of the cost that changes between the Meta OT problem instances. Another modeling dimension is the ability to capture variable-length input measures. Design decisions for this can be inspired from by VeLO (Metz et al., 2022), which learns a generic optimizer for large-scale machine learning models that can predict updates to models with 500M parameters. 3) **OT algorithms**. While we instantiated models on top of log-Sinkhorn, Meta OT could be built on top of other methods, and 4) **OT applications** that are computationally expensive and repeatedly solved, e.g. in multi-marginal and barycentric settings, or for Gromov-Wasserstein distances between metric-measure spaces.

Limitations. While we have illustrated successful applications of Meta OT, it is also important to understand the limitations that also arise in more general amortization settings: 1) **Meta OT does not make previously intractable problems tractable.** All of the baseline OT solvers we consider solve our problems within milliseconds or seconds. 2) **Out-of-distribution generalization.** Meta OT may not generate good predictions on instances that are not close to the training OT problems from the meta-distribution $\mathcal{D}$ over the measures and cost. If the model makes a bad prediction, one fallback option is to re-solve the instance from scratch.

Acknowledgments

We would like to thank Eugene Vinitsky, Mark Tygert, Mathieu Blondel, Maximilian Nickel, and Muhammad Izzatullah for insightful comments and discussions. The core set of tools in Python (Van Rossum and Drake Jr, 1995; Oliphant, 2007) enabled this work, including Hydra (Yadan, 2019), JAX (Bradbury et al., 2018), Matplotlib (Hunter, 2007), numpy (Oliphant, 2006; Van Der Walt et al., 2011), Optimal Transport Tools (Cuturi et al., 2022), and pandas (McKinney, 2012).

References

Mokhtar Z. Alaya, Maxime Berar, Gilles Gasso, and Alain Rakotomamonjy. Screening sinkhorn algorithm for regularized optimal transport. In *Advances in Neural Information Processing Systems*, pages 12169–12179, 2019. Cited on page 4.

Jason Altschuler, Jonathan Weed, and Philippe Rigollet. Near-linear time approximation algorithms for optimal transport via sinkhorn iteration. In *Advances in Neural Information Processing Systems*, pages 1964–1974, 2017. Cited on page 4.

Luigi Ambrosio. Lecture notes on optimal transport problems. In *Mathematical aspects of evolving interfaces*, pages 1–52. Springer, 2003. Cited on page 1.

Brandon Amos. Tutorial on amortized optimization for learning to optimize over continuous domains. *ArXiv preprint*, abs/2202.00665, 2022. Cited on pages 3 and 4.

Brandon Amos. On amortizing convex conjugates for optimal transport. In *The Eleventh International Conference on Learning Representations*, 2023. Cited on page 2.

Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 146–155. PMLR, 2017. Cited on pages 3, 9, and 19.

Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223. PMLR, 2017. Cited on page 1.

Clément Bonet, Titouan Vayer, Nicolas Courty, François Septier, and Lucas Drumetz. Subspace detours meet gromov–wasserstein. *Algorithms*, 14(12):366, 2021. Cited on page 8.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs. *GitHub*, 2018. Cited on pages 5 and 9.

Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on pure and applied mathematics*, 44(4):375–417, 1991. Cited on page 3.

Charlotte Bunne, Stefan G Stark, Gabriele Gut, Jacobo Sarabia del Castillo, Kjong-Van Lehmann, Lucas Pelkmans, Andreas Krause, and Gunnar Ratsch. Learning single-cell perturbation responses using neural optimal transport. *bioRxiv*, 2021. Cited on page 1.

Charlotte Bunne, Andreas Krause, and Marco Cuturi. Supervised training of conditional monge maps. *ArXiv preprint*, abs/2206.14262, 2022a. Cited on pages 1, 9, 18, and 19.

Charlotte Bunne, Laetitia Papaxanthos, Andreas Krause, and Marco Cuturi. Proximal optimal transport modeling of population dynamics. In *International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 151 of *Proceedings of Machine Learning Research*, pages 6511–6528. PMLR, 2022b. Cited on page 1.

Rick Chartrand, Brendt Wohlberg, Kevin Vixie, and Erik Bollt. A gradient descent solution to the monge-kantorovich problem. *Applied Mathematical Sciences*, 3(22):1071–1080, 2009. Cited on page 3.

Tianlong Chen, Xiaohan Chen, Wuyang Chen, Howard Heaton, Jialin Liu, Zhangyang Wang, and Wotao Yin. Learning to optimize: A primer and a benchmark. *ArXiv preprint*, abs/2103.12828, 2021. Cited on page 3.

Samuel Cohen, Brandon Amos, and Yaron Lipman. Riemannian convex potential maps. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 2028–2038. PMLR, 2021. Cited on page 1.

Roberto Cominetti and J San Martín. Asymptotic analysis of the exponential penalty trajectory in linear programming. *Mathematical Programming*, 67(1):169–187, 1994. Cited on page 2.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *Advances in Neural Information Processing Systems*, pages 3730–3739, 2017. Cited on page 1.

Nicolas Courty, Rémi Flamary, and Mélanie Ducoffe. Learning wasserstein embeddings. In *6th International Conference on Learning Representations*. OpenReview.net, 2018. Cited on page 9.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, pages 2292–2300, 2013. Cited on pages 1 and 2.

Marco Cuturi, Laetitia Meng-Papaxanthos, Yingtao Tian, Charlotte Bunne, Geoff Davis, and Olivier Teboul. Optimal transport tools (ott): A jax toolbox for all things wasserstein. *ArXiv preprint*, abs/2201.12324, 2022. Cited on pages 2, 5, and 9.

Robert Dadashi, Léonard Hussenot, Matthieu Geist, and Olivier Pietquin. Primal wasserstein imitation learning. In *9th International Conference on Learning Representations*. OpenReview.net, 2021. Cited on page 1.

Ishan Deshpande, Yuan-Ting Hu, Ruoyu Sun, Ayis Pyrros, Nasir Siddiqui, Sanmi Koyejo, Zhizhen Zhao, David A. Forsyth, and Alexander G. Schwing. Max-sliced wasserstein distance and its use for gans. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 10648–10656. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.01090. Cited on pages 1 and 9.

Erin Doxsey-Whitfield, Kytt MacManus, Susana B Adamo, Linda Pistolesi, John Squires, Olena Borkovska, and Sandra R Baptista. Taking advantage of the improved availability of census data: a first look at the gridded population of the world, version 4. *Papers in Applied Geography*, 1(3):226–234, 2015. Cited on page 6.

Bjorn Engquist and Brittany D Froese. Application of the wasserstein metric to seismic signals. *arXiv preprint arXiv:1311.4581*, 2013. Cited on page 1.

Kilian Fatras, Bharath Bhushan Damodaran, Sylvain Lobry, Remi Flamary, Devis Tuia, and Nicolas Courty. Wasserstein adversarial regularization for learning with label noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. Cited on pages 6 and 7.

Werner Fenchel. On conjugate convex functions. *Canadian Journal of Mathematics*, 1(1):73–77, 1949. Cited on page 3.

Arnaud Fickinger, Samuel Cohen, Stuart Russell, and Brandon Amos. Cross-domain imitation learning via optimal transport. In *International Conference on Learning Representations*, 2022. Cited on page 1.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017. Cited on pages 4, 18, and 19.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, et al. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78): 1–8, 2021. Cited on page 2.

Aden Forrow, Jan-Christian Hütter, Mor Nitzan, Philippe Rigollet, Geoffrey Schiebinger, and Jonathan Weed. Statistical optimal transport via factored couplings. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 89 of *Proceedings of Machine Learning Research*, pages 2454–2465. PMLR, 2019. Cited on page 8.

Charlie Frogner, Chiyuan Zhang, Hossein Mobahi, Mauricio Araya-Polo, and Tomaso A. Poggio. Learning with a wasserstein loss. In *Advances in Neural Information Processing Systems*, pages 2053–2061, 2015. Cited on page 6.

Alfred Galichon. Optimal transport methods in economics. In *Optimal Transport Methods in Economics*. Princeton University Press, 2016. Cited on page 1.

Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shanahan, Yee Whye Teh, Danilo Jimenez Rezende, and S. M. Ali Eslami. Conditional neural processes. In *Proceedings of the 35th International Conference on Machine Learning, ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 1690–1699. PMLR, 2018a. Cited on pages 4, 18, and 19.

Marta Garnelo, Jonathan Schwarz, Dan Rosenbaum, Fabio Viola, Danilo J Rezende, SM Eslami, and Yee Whye Teh. Neural processes. *ArXiv preprint*, abs/1807.01622, 2018b. Cited on pages 4, 18, and 19.

Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 84 of *Proceedings of Machine Learning Research*, pages 1608–1617. PMLR, 2018. Cited on page 9.

Andrew Gracyk and Xiaohui Chen. Geonet: a neural operator for learning the wasserstein geodesic. *ArXiv preprint*, abs/2209.14440, 2022. Cited on page 9.

David Ha, Andrew M. Dai, and Quoc V. Le. Hypernetworks. In *5th International Conference on Learning Representations, ICLR*. OpenReview.net, 2017. Cited on page 4.

Yuzhuo Han, Xiaofeng Liu, Zhenfei Sheng, Yutao Ren, Xu Han, Jane You, Risheng Liu, and Zhongxuan Luo. Wasserstein loss based deep object detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4299–4305, 2020. Cited on page 6.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016. Cited on page 4.

Chin-Wei Huang, Ricky T. Q. Chen, Christos Tsirigotis, and Aaron C. Courville. Convex potential flows: Universal probability distributions with optimal transport and convex optimization. In *9th International Conference on Learning Representations, ICLR*. OpenReview.net, 2021. Cited on page 1.

J. J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994. doi: 10.1109/34.291440. Cited on page 6.

John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3):90, 2007. Cited on page 9.

Pratik Jawanpuria, N.T.V. Satyadev, and Bamdev Mishra. Efficient robust optimal transport with application to multi-label classification. In *CDC*, page 1490–1495, 2021. Cited on page 6.

Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang, and Silvia Chiappa. Wasserstein fair classification. In *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI*, volume 115 of *Proceedings of Machine Learning Research*, pages 862–872. AUAI Press, 2019. Cited on page 1.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR*, 2015. Cited on page 4.

Nicholas I. Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 10051–10060. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.01029. Cited on page 1.

Soheil Kolouri, Yang Zou, and Gustavo K. Rohde. Sliced wasserstein kernels for probability distributions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 5258–5267. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.568. Cited on page 1.

Soheil Kolouri, Se Rim Park, Matthew Thorpe, Dejan Slepcev, and Gustavo K Rohde. Optimal mass transport: Signal processing and machine-learning applications. *IEEE signal processing magazine*, 34(4):43–59, 2017. Cited on page 1.

Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo K. Rohde. Generalized sliced wasserstein distances. In *Neural Information Processing Systems*, pages 261–272, 2019a. Cited on page 1.

Soheil Kolouri, Phillip E. Pope, Charles E. Martin, and Gustavo K. Rohde. Sliced wasserstein auto-encoders. In *7th International Conference on Learning Representations*. OpenReview.net, 2019b. Cited on page 1.

Alexander Korotin, Vage Egiazarian, Arip Asadulaev, Alexander Safin, and Evgeny Burnaev. Wasserstein-2 generative networks. In *International Conference on Learning Representations*. OpenReview.net, 2021a. Cited on pages 2, 3, 4, and 8.

Alexander Korotin, Lingxiao Li, Aude Genevay, Justin M. Solomon, Alexander Filippov, and Evgeny Burnaev. Do neural optimal transport solvers work? A continuous wasserstein-2 benchmark. In *NeurIPS*, pages 14593–14605, 2021b. Cited on page 8.

Alexander Korotin, Lingxiao Li, Justin Solomon, and Evgeny Burnaev. Continuous wasserstein-2 barycenter estimation without minimax optimization. In *International Conference on Learning Representations*. OpenReview.net, 2021c. Cited on pages 2 and 8.

Alexander Korotin, Daniil Selikhanovych, and Evgeny Burnaev. Neural optimal transport. *ArXiv preprint*, abs/2201.12220, 2022. Cited on page 2.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. *ArXiv preprint*, abs/2108.08481, 2021. Cited on page 9.

Julien Lacombe, Julie Digne, Nicolas Courty, and Nicolas Bonneel. Learning to generate wasserstein barycenters, 2021. Cited on page 9.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *ArXiv preprint*, abs/2003.03485, 2020. Cited on page 9.

Tianyi Lin, Nhat Ho, and Michael I Jordan. On the acceleration of the sinkhorn and greenkhorn algorithms for optimal transport. *ArXiv preprint*, abs/1906.01437, 2019. Cited on page 4.

Alexander Litvinenko, Youssef Marzouk, Hermann G Matthies, Marco Scavino, and Alessio Spantini. Computing f-divergences and distances of high-dimensional probability density functions–low-rank tensor approximations. *ArXiv preprint*, abs/2111.07164, 2021. Cited on page 8.

Xiaofeng Liu, Yang Zou, Tong Che, Ping Jia, Peng Ding, Jane You, and B. V. K. Vijaya Kumar. Conservative wasserstein training for pose estimation. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV*, pages 8261–8271. IEEE, 2019. doi: 10.1109/ICCV.2019.00835. Cited on page 6.

Xiaofeng Liu, Yuzhuo Han, Song Bai, Yi Ge, Tianxing Wang, Xu Han, Site Li, Jane You, and Jun Lu. Importance-aware semantic segmentation in self-driving with discrete wasserstein training. In *Conference on Artificial Intelligence, AAAI*, pages 11629–11636. AAAI Press, 2020a. Cited on page 6.

Xiaofeng Liu, Wenxuan Ji, Jane You, Georges El Fakhri, and Jonghye Woo. Severity-aware semantic segmentation with reinforced wasserstein training. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 12563–12572. IEEE, 2020b. doi: 10.1109/CVPR42600.2020.01258. Cited on page 6.

Ashok Vardhan Makkuva, Amirhossein Taghvaei, Sewoong Oh, and Jason D. Lee. Optimal transport mapping via input convex neural networks. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 6672–6681. PMLR, 2020. Cited on page 2.

Wes McKinney. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. " O'Reilly Media, Inc.", 2012. Cited on page 9.

Quentin Merigot and Boris Thibert. Optimal transport: discretization and algorithms. In *Handbook of Numerical Analysis*, volume 22, pages 133–212. Elsevier, 2021. Cited on page 1.

Luke Metz, James Harrison, C Daniel Freeman, Amil Merchant, Lucas Beyer, James Bradbury, Naman Agrawal, Ben Poole, Igor Mordatch, Adam Roberts, et al. Velo: Training versatile learned optimizers by scaling up. *ArXiv preprint*, abs/2211.09760, 2022. Cited on page 9.

Petr Mokrov, Alexander Korotin, Lingxiao Li, Aude Genevay, Justin M. Solomon, and Evgeny Burnaev. Large-scale wasserstein gradient flows. In *NeurIPS*, pages 15243–15256, 2021. Cited on page 8.

Boris Muzellec and Marco Cuturi. Subspace detours: Building transport plans that are optimal on subspace projections. In *Advances in Neural Information Processing Systems*, pages 6914–6925, 2019. Cited on page 8.

Khai Nguyen and Nhat Ho. Amortized projection optimization for sliced wasserstein generative models. *ArXiv preprint*, abs/2203.13417, 2022. Cited on page 9.

Travis E Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006. Cited on page 9.

Travis E Oliphant. Python for scientific computing. *Computing in Science & Engineering*, 9(3):10–20, 2007. Cited on page 9.

Michaël Perrot, Nicolas Courty, Rémi Flamary, and Amaury Habrard. Mapping estimation for discrete optimal transport. In *Advances in Neural Information Processing Systems*, pages 4197–4205, 2016. Cited on page 1.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019. Cited on pages 1, 2, and 5.

Ievgen Redko, Nicolas Courty, Rémi Flamary, and Devis Tuia. Optimal transport for multi-source domain adaptation under target shift. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS*, volume 89 of *Proceedings of Machine Learning Research*, pages 849–858. PMLR, 2019. Cited on page 1.

Ralph Tyrell Rockafellar. Convex analysis. In *Convex analysis*. Princeton university press, 2015. Cited on page 3.

Litu Rout, Alexander Korotin, and Evgeny Burnaev. Generative modeling with optimal transport maps. In *ICLR*. OpenReview.net, 2022. Cited on page 1.

Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *7th International Conference on Learning Representations, ICLR 2019*. OpenReview.net, 2019. Cited on page 19.

Filippo Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, 55(58-63):94, 2015. Cited on page 1.

Meyer Scetbon, Marco Cuturi, and Gabriel Peyré. Low-rank sinkhorn factorization. In *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 9344–9354. PMLR, 2021. Cited on page 8.

Meyer Scetbon, Gabriel Peyré, and Marco Cuturi. Linear-time gromov wasserstein distances using low rank couplings and costs. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pages 19347–19365. PMLR, 2022. Cited on page 8.

Geoffrey Schiebinger, Jian Shu, Marcin Tabaka, Brian Cleary, Vidya Subramanian, Aryeh Solomon, Joshua

Gould, Siyan Liu, Stacie Lin, Peter Berube, Lia Lee, Jenny Chen, Justin Brumbaugh, Philippe Rigollet, Konrad Hochedlinger, Rudolf Jaenisch, Aviv Regev, and Eric S. Lander. Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell*, 176(4):928–943.e22, 2019. ISSN 0092-8674. doi: https://doi.org/10.1016/j.cell.2019.01.006. Cited on page 1.

Vivien Seguy, Bharath Bhushan Damodaran, Rémi Flamary, Nicolas Courty, Antoine Rolet, and Mathieu Blondel. Large scale optimal transport and mapping estimation. In *6th International Conference on Learning Representations, ICLR 2018*. OpenReview.net, 2018. Cited on page 1.

Justin Solomon, Fernando De Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas. Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Transactions on Graphics (TOG)*, 34(4):1–11, 2015. Cited on page 1.

Max Sommerfeld, Jörn Schrieber, Yoav Zemel, and Axel Munk. Optimal transport: Fast probabilistic approximation with exact solvers. *J. Mach. Learn. Res.*, 20:105–1, 2019. Cited on page 8.

Kenneth O Stanley, David B D'Ambrosio, and Jason Gauci. A hypercube-based encoding for evolving large-scale neural networks. *Artificial life*, 15(2):185–212, 2009. Cited on page 4.

Amirhossein Taghvaei and Amin Jalali. 2-wasserstein approximation via restricted convex potentials with application to improved training for gans. *ArXiv preprint*, abs/1902.07197, 2019. Cited on pages 2 and 3.

Matthew Tancik, Ben Mildenhall, Terrance Wang, Divi Schmidt, Pratul P Srinivasan, Jonathan T Barron, and Ren Ng. Learned initializations for optimizing coordinate-based neural representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2846–2855, 2021. Cited on page 19.

Alexis Thibault, Lenaic Chizat, Charles Dossal, and Nicolas Papadakis. Overrelaxed sinkhorn-knopp algorithm for regularized optimal transport. *ArXiv preprint*, abs/1711.01851, 2017. Cited on page 4.

James Thornton and Marco Cuturi. Rethinking initialization of the sinkhorn algorithm. *ArXiv preprint*, abs/2206.07630, 2022. Cited on pages 5 and 18.

Ayato Toyokuni, Sho Yokoi, Hisashi Kashima, and Makoto Yamada. Computationally efficient Wasserstein loss for structured labels. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 1–7, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-srw.1. Cited on page 6.

Stefan Van Der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13 (2):22, 2011. Cited on page 9.

Guido Van Rossum and Fred L Drake Jr. *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995. Cited on page 9.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009. Cited on page 1.

Omry Yadan. Hydra - a framework for elegantly configuring complex applications. Github, 2019. Cited on page 9.

Yang Yang, Yi-Feng Wu, De-Chuan Zhan, Zhi-Bin Liu, and Yuan Jiang. Complex object classification: A multi-modal multi-instance multi-label deep network with optimal transport. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD*, pages 2594–2603. ACM, 2018. doi: 10.1145/3219819.3220012. Cited on page 6.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision, ICCV*, pages 2242–2251. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.244. Cited on page 3.

Luisa M. Zintgraf, Kyriacos Shiarlis, Vitaly Kurin, Katja Hofmann, and Shimon Whiteson. Fast context adaptation via meta-learning. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 7693–7702. PMLR, 2019. Cited on page 19.

A. Selecting ϵ for MNIST

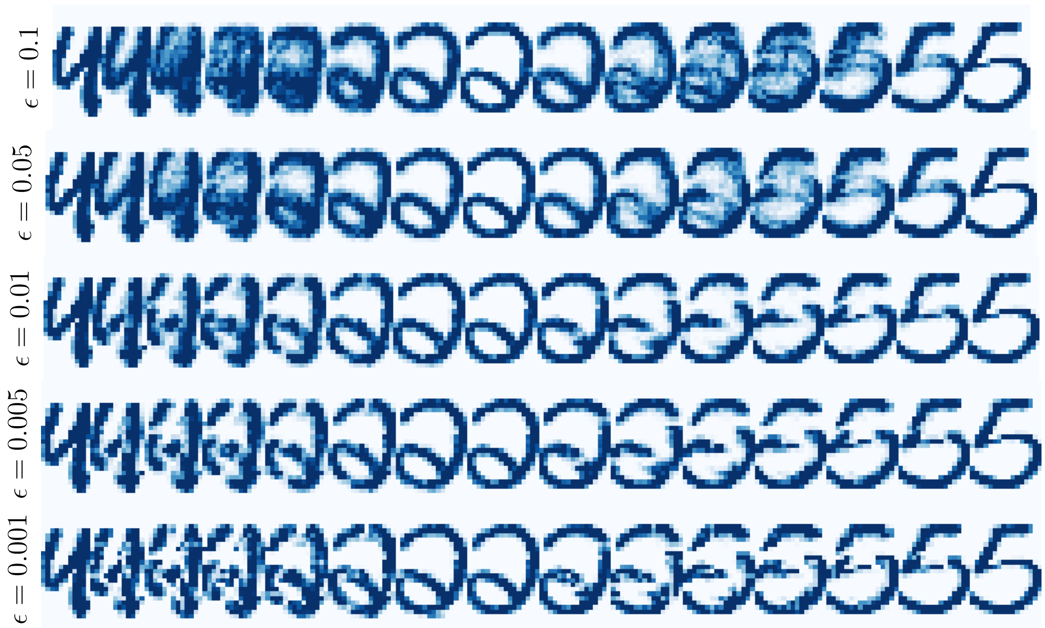

Figure 8. We selected $\epsilon = 10^{-2}$ for our MNIST coupling experiments as it results in transport maps that are not too blurry or sharp.

B. Additional experimental and implementation details

Our Jax source code is available at `http://github.com/facebookresearch/meta-ot` and contains:

- `meta_ot` Meta OT Python library code
  - `conjugate.py` Exact conjugate solver for the continuous setting
  - `data.py`
  - `models.py`
  - `utils.py`
- `config` Hydra configuration for the experiments (containing hyper-parameters)
- `train_discrete.py` Train Meta OT models for discrete OT
- `train_color_single.py` Train a single ICNN with W2GN between 2 images (for debugging)
- `train_color_meta.py` Train a Meta ICNN with W2GN
- `plot_mnist.py` Visualize the MNIST couplings
- `plot_world_pair.py` Visualize the spherical couplings
- `eval_color.py` Evaluate the Meta ICNN in the continuous setting
- `eval_discrete.py` Evaluate the Meta ICNN for the discrete tasks

Connecting to the data is one difficulty in running the experiments. The easiest experiment to re-run is the MNIST one, which will automatically download the dataset:

```
./train_discrete.py # Train the model, outputting to <exp_dir>
./eval_discrete.py <exp_dir> # Evaluate the learned models
./plot_mnist.py <exp_dir> # Produce further visualizations
```

B.1. Hyper-parameters

We briefly summarize the hyper-parameters we used for training, which we did not extensively tune. In the discrete setting, we use the same hyper-parameters for the MNIST and spherical settings.

Table 5. Discrete OT hyper-parameters.

Name	Value
Batch size	128
Number of training iterations	50000
MLP Hidden Sizes	[1024, 1024, 1024]
Adam learning rate	1e-3

Table 6. Continuous OT hyper-parameters.

Name	Value
Meta batch size (for α, β)	8
Inner batch size (to estimate $\mathcal{L}$)	1024
Cycle loss weight (γ)	3.
Adam learning rate	1e-3
ℓ_2 weight penalty	1e-6
Max grad norm (for clipping)	1.
Number of training iterations	200000
Meta ICNN Encoder	ResNet18
Encoder output size (both measures)	256×2
Meta ICNN Decoder Hidden Sizes	[512]

B.2. Sinkhorn convergence times, varying thresholds

In the main paper, table 1 reports the runtime of Sinkhorn to reach a convergence threshold of the marginal error being below a tolerance of 10^{23}. Tables 7 and 8 report the results from sweeping over other thresholds and show that Meta OT's initialization is consistently able to help.

Table 7. Sinkhorn runtime to reach a thresholded marginal error on MNIST.

Initialization	Threshold=10^{-2}	Threshold=10^{-3}	Threshold=10^{-4}	Threshold=10^{-5}
Zeros	$4.5 \cdot 10^{-3}$ $1.5 \cdot 10^{-3}$	$7.7 \cdot 10^{-3}$ $1.2 \cdot 10^{-3}$	$1.1 \cdot 10^{-2}$ $1.8 \cdot 10^{-3}$	$1.5 \cdot 10^{-2}$ $2.3 \cdot 10^{-3}$
Gaussian	$4.1 \cdot 10^{-3}$ $1.2 \cdot 10^{-3}$	$7.7 \cdot 10^{-3}$ $1.4 \cdot 10^{-3}$	$1.1 \cdot 10^{-2}$ $1.7 \cdot 10^{-3}$	$1.4 \cdot 10^{-2}$ $2.4 \cdot 10^{-3}$
Meta OT	$2.3 \cdot 10^{-3}$ $9.2 \cdot 10^{-6}$	$3.9 \cdot 10^{-3}$ $1.6 \cdot 10^{-3}$	$6.7 \cdot 10^{-3}$ $1.4 \cdot 10^{-3}$	$1.0 \cdot 10^{-2}$ $2.4 \cdot 10^{-3}$

Table 8. Sinkhorn runtime to reach a thresholded marginal error on the spherical transport problem.

Initialization	Threshold=10^{-2}	Threshold=10^{-3}	Threshold=10^{-4}	Threshold=10^{-5}
Zeros	$8.8 \cdot 10^{-1}$ $\pm 1.3 \cdot 10^{-1}$	1.4 $\pm 1.9 \cdot 10^{-1}$	2.1 $\pm 3.6 \cdot 10^{-1}$	2.8 $\pm 5.6 \cdot 10^{-1}$
Gaussian	$5.6 \cdot 10^{-1}$ $\pm 9.9 \cdot 10^{-2}$	1.1 $\pm 2.0 \cdot 10^{-1}$	1.7 $\pm 3.5 \cdot 10^{-1}$	2.4 $\pm 5.4 \cdot 10^{-1}$
Meta OT	$7.8 \cdot 10^{-2}$ $\pm 3.4 \cdot 10^{-2}$	0.44 $\pm 1.5 \cdot 10^{-1}$	0.97 $\pm 3.2 \cdot 10^{-1}$	1.7 $\pm 6.8 \cdot 10^{-1}$

B.3. Experimental runtimes and convergence

App. B.3 shows the convergence during training of Meta OT models in the discrete and continuous settings over 10 trials on our single Quadro GP100 GPU. The MNIST models are consistently trained to optimality within 2 minutes (!) while the continuous model takes a few hours to train.

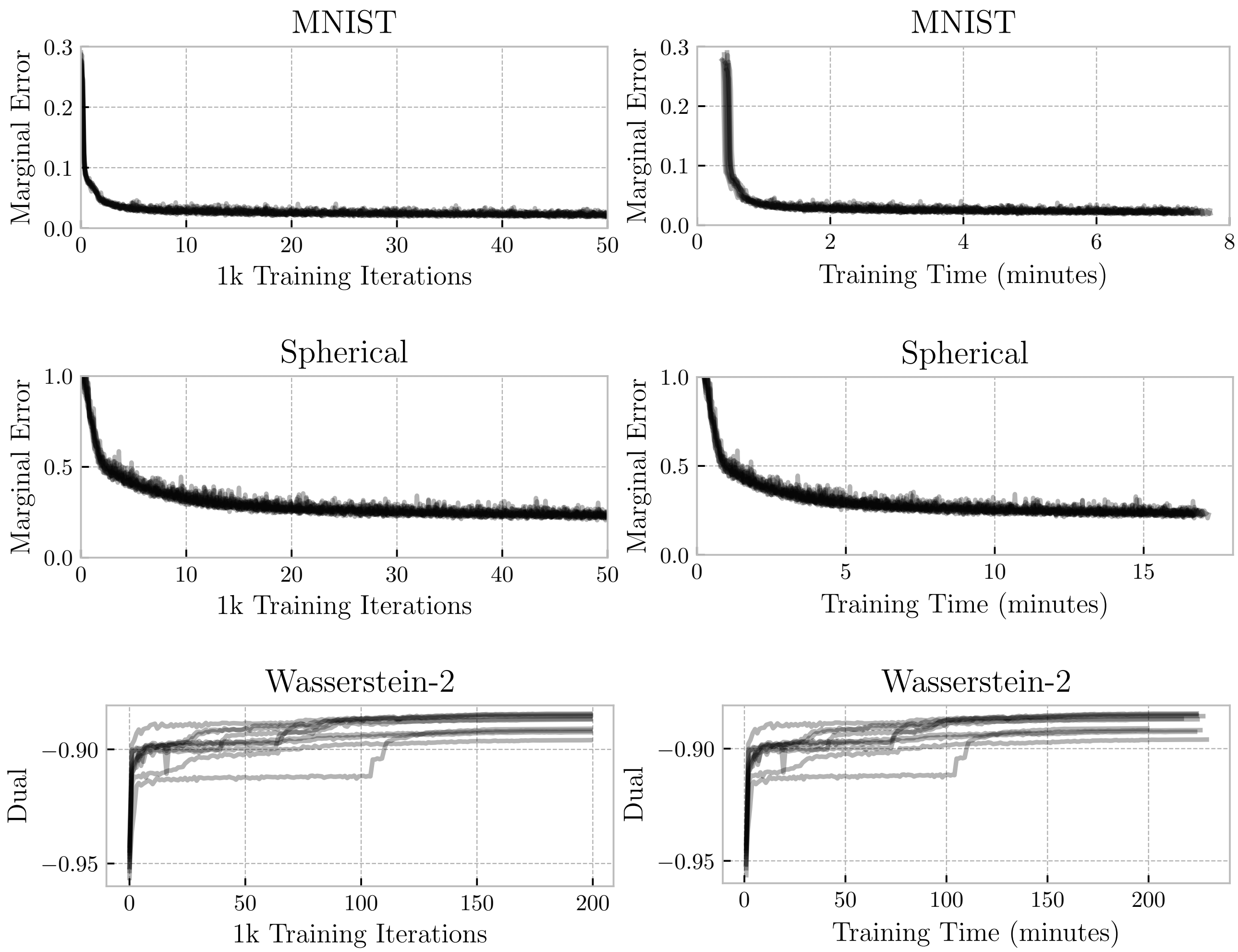

Figure 9. Convergence of Meta OT models during training, reported over iterations and wall-clock time. We run each experiment for 10 trials with different seeds and report each trial as a line.

C. Cross-domain experimental results

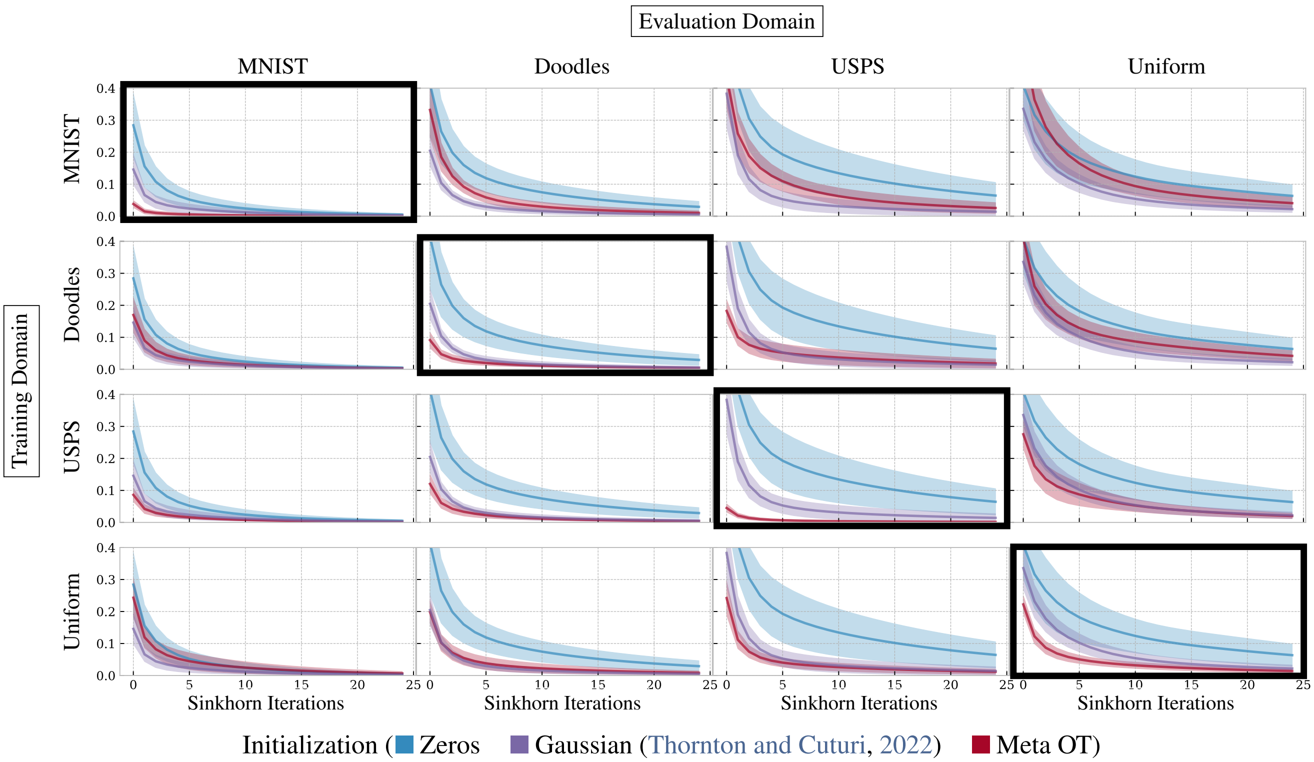

Figure 10. Cross-domain experiments evaluating how well a model trained on one dataset generalizes to another dataset. Notably, we are able to train only on a uniform distribution and transfer reasonable initializations to the image datasets. This indicates that training larger-scale Meta OT models for more general classes of discrete OT problems may be able to provide a fast and reasonable initialization.

D. More information: Meta OT between continuous measures

D.1. Meta ICNN Diagram

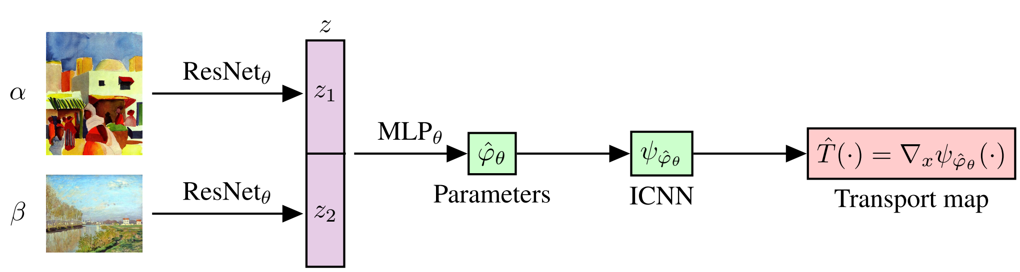

Figure 11. A Meta ICNN for image-based input measures. A shared ResNet processes the input measures α and β into latents z that are decoded with an MLP into the parameters φ of an ICNN dual potential ψ_φ. The derivative of the ICNN provides the transport map $\hat{T}$.

D.2. Other models for continuous OT

We explored a hyper-network model because it is conceptually the most similar to predicting the optimal dual variables in the continuous setting and results in rapid predictions. However, it may not scale well to predicting high-dimensional parameters of ICNNs. This section presents two alternatives based on MAML (Finn et al., 2017) and neural processes (Garnelo et al., 2018b;a), and conditional OT maps (Bunne et al., 2022a).

D.2.1. Optimization-based meta-learning (MAML-inspired)

The model-agnostic meta-learning setup proposed in MAML (Finn et al., 2017) could also be applied in the Meta OT setting to learn an adaptable initial parameterization. In the continuous setting, one initial version would take a parameterized dual potential model $\psi_\varphi(x)$ and seek to learn an initial parameterization φ_0 so that optimizing a loss such as the W2GN loss $\mathcal{L}$ from eq. (13) results in a minimal $\mathcal{L}(\varphi_K)$ after adapting the model for K steps. Formally, this would optimize:

$$\arg\min_{\varphi_0} \mathcal{L}(\varphi_K) \quad \text{where} \quad \varphi_{t+1} = \varphi_t - \nabla_\varphi \mathcal{L}(\varphi_t) \tag{20}$$

Tancik et al. (2021) explores similar learned initializations for coordinate-based neural implicit representations for 2D images, CT scan reconstruction, and 3d shape and scene recovery from 2D observations.

Challenges for Meta OT. The transport maps given by $T = \nabla\psi$ can significantly vary depending on the input measures α, β. We found it difficult to learn an initialization that can be rapidly adapted, and optimizing eq. (20) is more computationally expensive than eq. (18) as it requires unrolling through many evaluations of the transport loss $\mathcal{L}$. And, we found that *only* learning to predict the optimal parameters with eq. (18), conditional on the input measures, and then fine-tuning with W2GN to be stable.

Advantages for Meta OT. Exploring MAML-inspired methods could further incorporate the knowledge that the model's prediction is going to be fine-tuned into the learning process. One promising direction we did not try could be to integrate some of the ideas from LEO (Rusu et al., 2019) and CAVIA (Zintgraf et al., 2019), which propose to learn a latent space for the parameters where the initialization is also conditional on the input.

D.2.2. Neural process and conditional Monge maps

The (conditional) neural process models considered in Garnelo et al. (2018b;a) can also be adapted for the Meta OT setting, and is similar to the model proposed in Bunne et al. (2022a). In the continuous setting, this would result in a dual potential that is also conditioned on a representation of the input measures, e.g. $\psi_\varphi(x; z)$ where $z := f_\varphi^{\rm emb}(\alpha, \beta)$ is a learned embedding of the input measures that is learned with the parameters of ψ. This could be formulated as

$$\arg\min_{\varphi} \mathop{\mathbb{E}}_{(\alpha,\beta)\sim\mathcal{D}} \mathcal{L}(\varphi, f_\varphi^{\rm emb}(\alpha, \beta)), \tag{21}$$

where $\mathcal{L}$ modifies the model used in the loss eq. (13) to also be conditioned on the context extracted from the measures.

Challenges for Meta OT. This raises the issue on best-formulating the model to be conditional on the context. One way could be to append z to the input point x in the domain. Bunne et al. (2022a) proposes to use the Partially Input-Convex Neural Network (PICNN) from (Amos et al., 2017) to make the model convex with respect to x and not z.

Advantages for Meta OT. A large advantage is that the representation z of the measures α, β would be significantly lower-dimensional than the parameters φ that our Meta OT models are predicting.

D.3. Continuous Wasserstein-2 color transfer

The following public domain images are from WikiArt:

- Distant View of the Pyramids by Winston Churchill (1921)
- Charing Cross Bridge, Overcast Weather by Claude Monet (1900)
- Houses of Parliament by Claude Monet (1904)
- October Sundown, Newport by Childe Hassam (1901)
- Landscape with House at Ceret by Juan Gris (1913)
- Irises in Monet's Garden by Claude Monet (1900)
- Crystal Gradation by Paul Klee (1921)
- Senecio by Paul Klee (1922)
- Váza s květinami by Josef Capek (1914)
- Sower with Setting Sun by Vincent van Gogh (1888)
- Three Trees in Grey Weather by Claude Monet (1891)
- Vase with Daisies and Anemones by Vincent van Gogh (1887)

Figure 12. Meta ICNN (initial prediction). The sources are given in the beginning of app. D.3.

Figure 13. Meta ICNN + W2GN fine-tuning. The sources are given in the beginning of app. D.3.

Figure 14. W2GN (final). The sources are given in the beginning of app. D.3.