Supervised Training of Conditional Monge Maps

Anonymous Author(s)
Affiliation
Address
email

Abstract

Optimal transport theory describes general principles to define and select, among many possible choices, the most efficient way to map a probability measure onto another. Optimizing for such maps presents considerable optimization challenges. Many practical algorithms have been proposed to recover *Monge maps*, i.e., given samples from a pair of probability measures $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, find a parameterized map $T_\theta : \mathbb{R}^d \to \mathbb{R}^d$ such that $T_\theta \sharp \mu \approx \nu$ and that $T_\theta(x)$ be close to x on average w.r.t. μ. Here, we propose a novel generalization of this approach, where the data comes in the form of multiple *pairs* of measures (μ_i, ν_i) to each of which is associated a *context* label c_i. Our goal is to extract from these pairs $\{(c_i, (\mu_i, \nu_i))\}$ a *global* map $\mathcal{T}_\theta$ which fits *all observed pairs* when conditioned with the right context, i.e., $\mathcal{T}_\theta(c_i)\sharp\mu_i \approx \nu_i$, and which is also expected to generalize to unseen contexts c. We implement this approach by harnessing *partially input convex neural network architectures*, and apply it to the challenging problem of inferring the treatment effects of an arbitrary combination of drugs, using only the observations of the effects of said drugs taken in isolation.

1 Introduction

A key challenge in the treatment of cancer is to predict the effect of (a combination of) drugs on cells from a particular patient. Recent techniques enable us to obtain measurements from individual cells, in treated and untreated conditions, which however are not in correspondence. Given such examples of untreated and treated cells under different drugs, can we predict the effect of new drug combinations? We develop a general approach motivated by this and related problems, through the lens of *optimal transport* (OT) *theory*, and, in that process, develop tools that might be of interest for other application domains of OT. Given a collection of N pairs of measures (μ_i, ν_i) over $\mathbb{R}^d$ (cf., cell measurements), along with a context c_i (cf., encoding the treatment), we seek to learn a context-dependent, parameterized transport map $\mathcal{T}_\theta$ such that, on training data, $\mathcal{T}_\theta(c_i)\sharp\mu_i \approx \nu_i$. Additionally, our goal is to expect this parameterized map to be able to generalize to unseen contexts c_{new} in the sense that given a patient's cells described in μ one might predict the effect of context c_{new} on these cells as $\mathcal{T}_\theta(c_{\text{new}})\sharp\mu$.

Related Work From generative adversarial networks to normalizing flows and diffusion models, the problem of learning maps that move points from a source to a target distribution is central to machine learning and its applications. Optimal transport theory (Santambrogio, 2015) has emerged as a principled approach towards estimating such maps. For a pair of measures μ, ν supported in $\mathbb{R}^d$, OT suggests that, among all maps T such that ν can be reconstructed by applying T to every point in the support of μ, summarized with the push-forward notation as $T\sharp\mu \approx \nu$, one should favor maps T that minimize the length of the average displacement $\|T(x) - x\|^p$, where $p \geq 1$. A rich literature, summarized by Peyré and Cuturi (2019), addresses computational challenges of estimating such maps, and has found impactful applications in several areas of science (cf., Hashimoto et al., 2016; Schiebinger et al., 2019; Yang et al., 2020; Janati et al., 2020).

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

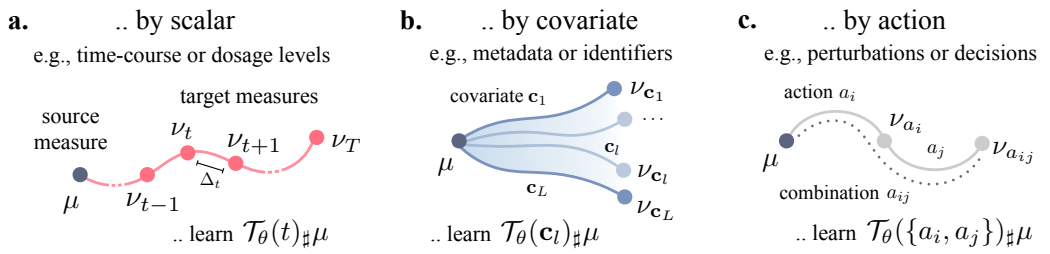

Figure 1: The evolution of a source μ into a target measure ν can be dependent on latent conditions c of various nature. This comprises **a.** scalars such as time or dosage t which determine the magnitude of an optimal transport, **b.** flow of a measure into another one based on additional information stored in vectors $\mathbf{c}_l$, or **c.** discrete and complex actions a_i, possibly in combination a_{ij}. We seek a unified framework for learning a map $\mathcal{T}_\theta$ to predict a target measure ν from any type of condition.

We focus on neural approaches that parameterize the optimal maps T via neural networks. An early approach is the work on Wasserstein GANs (Arjovsky et al., 2017), albeit the transport map is not explicitly estimated. Several recent results have exploited a more explicit connection between OT and NNs, derived from the celebrated Brenier theorem, that is applicable to the case $p = 2$. That theorem states that Monge maps are necessarily gradients of convex functions. As a result, one can rely on the family of input convex neural networks (ICNN) (Amos et al., 2017) to parameterize either the Monge map (Jacob et al., 2018; Yang and Uhler, 2019) or a dual potential (Makkuva et al., 2020; Korotin et al., 2020) as, respectively, the gradient of an ICNN or an ICNN itself. These approaches have been used for instance to model perturbation effects on single-cell populations (Bunne et al., 2021, 2022). In this paper, we build on this line of work, but substantially generalize it, by learning a *parametric family* of context-dependent transport maps, using a collection of labeled measures.

Our Contributions We propose a framework that can leverage *labeled* pairs of measures $\{(c_i, (\mu_i, \nu_i))\}_i$ to infer a *global* parameterized map $\mathcal{T}_\theta$. Hereby, the context c_i belongs to an arbitrary set $\mathcal{C}$. We construct $\mathcal{T}_\theta$ so that it should be able, given a possibly unseen context label $c \in \mathcal{C}$, to output a map from $\mathbb{R}^d$ onto itself, $\mathcal{T}_\theta(c) : \mathbb{R}^d \to \mathbb{R}^d$, ensuring that itself is the gradient of a convex function. To that end, we propose to learn these parameterized Monge maps $\mathcal{T}_\theta$ as the gradients of partially input convex neural networks (PICNN), which we borrow from the foundational work of Amos et al. (2017). In our framework, a PICNN can be alternatively interpreted as a hypernetwork (Ha et al., 2016) that produces an ICNN from a context vector c. Because both ICNN —and to a greater extent PICNN— are notoriously difficult to train (Richter-Powell et al., 2021), we use closed-form solutions between Gaussian approximations to derive generic initializations for (P)ICNN training. These initializations ensure the gradients of (P)ICNNs initially mimick the identity map or an affine Monge map between Gaussian approximations of measures μ_i, ν_i. We demonstrate the usefulness of our framework on three crucial scenarios: parameterization of transport through a real variable (time or drug dosage), through an auxiliary informative variable (cell covariates) and through action variables (genetic perturbations in combination). Our results demonstrate the ability of our architectures to better capture on out-of-sample observations the effects due to these variables in various settings, even when considering never-seen, composite context labels. These results suggest potential applications of conditional OT to model personalized medicine outcomes or to guide novel experiments, where OT could serve as a predictor for never tested context labels.

2 Background on Neural Solvers for the 2-Wasserstein Problem

Optimal Transport The Monge problem between two measures $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$, here restricted to measures supported on $\mathbb{R}^d$ and compared with the squared Euclidean metric, reads

$$T^\star_{\mu,\nu} := \arg\inf_{T\sharp\mu=\nu} \int_{\mathbb{R}^d} \|x - T(x)\|^2 d\mu(x)\,. \tag{1}$$

The existence of $T^\star_{\mu,\nu}$ is guaranteed under fairly general conditions (Santambrogio, 2015, Theorem 1.17). This can be proved with the celebrated Brenier theorem (1987), which states that there must exist a unique (up to the addition of a constant) potential $\psi^\star_{\mu,\nu} : \mathbb{R}^d \to \mathbb{R}$ such that $T^\star_{\mu,\nu} = \nabla\psi^\star_{\mu,\nu}$.

This theorem has far-reaching impact: it is sufficient, when seeking optimal transport maps, to restrict the computational effort to seek a "good" convex potential such that its gradient pushes μ towards ν. This result has been exploited to propose OT solvers that rely on input convex neural networks (ICNNs) (Amos et al., 2017). The practical estimation of such maps can be carried out using various tricks (Makkuva et al., 2020; Korotin et al., 2020; Bunne et al., 2022; Alvarez-Melis et al., 2021; Mokrov et al., 2021). For instance, a related reformulation of Problem (1) is to recover the following identy linking Monge and dual optimal solutions (writing f^* for the Legendre transform of f):

$$f^\star_{\mu,\nu} := \arg \sup_{f \text{ convex}} \int_{\mathbb{R}^d} f^* \mathrm{d}\mu + \int_{\mathbb{R}^d} f \mathrm{d}\nu \; = \psi^\star_{\mu,\nu} \,. \tag{2}$$

Convex Neural Architectures Input convex neural networks (ICNN) are neural networks ψ_θ that admit certain constraints on their architecture and parameters θ, such that their output $\psi_\theta(x)$ is a convex function of their input x (Amos et al., 2017). As a result, they have been increasingly used as drop-in replacements to the set of admissible functions in (2). Practically speaking, an ICNN is a K-layer, fully connected network such that, at each layer index k from 0 to $K-1$, one has

$$z_{k+1} = \sigma_k(W^x_k x + W^z_k z_k + b_k) \text{ and } \psi_\theta(x) = z_K \,, \tag{3}$$

where, by convention, z_0 and W^z_0 are 0, and σ_k are *convex* non-decreasing activation functions, $\theta = \{b_k, W^z_k, W^x_k\}_{k=0}^{K-1}$ are the weights and biases of the neural network. While ample flexibility is provided to choose dimensions for intermediate hidden states z_k, the last layer must necessarily produce a scalar, hence W^x_{K-1} and W^z_{k-1} are line vectors and $b_{K-1} \in \mathbb{R}$. ICNNs are characterized by the fact that all weight matrices W^z_k associated to latent representations z must have non-negative entries, which ensures their convexity w.r.t. inputs x. We encode this identifying these matrices as the softplus of other matrices of the same size. Since the work by Amos et al. (2017), convex neural architectures have been used within the context of OT to model convex dual functions (Makkuva et al., 2020), or normalizing flows derived from convex potentials (Huang et al., 2021). Their expressivity and universal approximation properties have been studied by Chen et al. (2019), who show that any convex function over a compact convex domain can be approximated in sup norm by an ICNN.

3 Supervised Training of Conditional Monge Maps

We assume we are given N pairs of measures $(\mu_i, \nu_i) \in \mathcal{P}(\mathbb{R}^d)^2$, each labeled by a context value $c_i \in \mathcal{C}$. We make minimal assumptions on $\mathcal{C}$: For instance, $\mathcal{C}$ could be $\mathbb{R}$: This can be useful to encode a time-span Δt that occurred between the observation of the same population on two timestamps, i.e., μ_i, then ν_i, or a quantification of strength of dosage of a single treatment of interest. More generally, context c can describe metadata or identifiers provided as additional inputs to describe the mass transfer from μ_i to ν_i (capturing, e.g., the effect of different experimental conditions). Finally, they can encode more complex actions, possibly described through a learned representation, and, as we present later in§3.2, allowing for combinations of such actions.

3.1 Integrating Context in Convex Architectures

Towards Parameterized Monge Maps Building on Brenier's theorem and following Korotin et al. (2020); Makkuva et al. (2020), we retain the idea that OT maps between two measures μ_i and ν_i should be estimated as the gradients of a convex Brenier potential $\nabla\psi$, and that such convex functions should be defined using neural architectures. Unlike previous approaches, however, our goal is to *(i.)* avoid the estimation of a new ICNN for each available pair (μ_i, ν_i), and, instead, pool efficiently these samples to train in a multi-task manner an overall neural architecture whose weights can be shared efficiently, while *(ii.)* leveraging as much as possible context values c_i to facilitate joint training and improve generalization. To reach these two goals, we propose to learn a *unique* parameterized map $\mathcal{T}_\theta$ that outputs, for any context value c (either known and seen in the dataset, or unknown), a $\mathbb{R}^d \to \mathbb{R}^d$ map $\mathcal{T}_\theta(c)$ that has the property that if $c \approx c_i$ then $\mathcal{T}_\theta(c)\#\mu_i \approx \nu_i$. This goal can be realized by defining a parameterized potential function $\psi_\theta : \mathbb{R}^d \times \mathcal{C}$ that takes that context as an additional input, to define next our parameterized families of OT maps as

$$\mathcal{T}_\theta(c) := x \mapsto \nabla_1 \psi_\theta(x, c).$$

A crucial observation is that the resulting parameterized potential $\psi_\theta(x, c)$ only needs to be input convex in x, *but not in context value* c to ensure that $\mathcal{T}_\theta(c)$ is an OT map. Thus, to obtain this property, we build conditional Monge maps using *partially input convex neural networks* (PICNN).

Partially Convex Neural Architectures PICNNs are neural networks that can be evaluated over a pair of inputs (x, c), but which are only required to be convex w.r.t. x. Such networks were proposed by Amos et al. (2017, Eq. (3)) to address a problem that is in a way the opposite of ours: Their inputs are labeled as (y, x), where y is a label vector of small size respective to a large dimensional input vector x. Their PICNN is convex w.r.t. y, in order to easily recover, given an *input* observation x (e.g., images) the optimal label y aligned with x using gradient descent as a subroutine. Their PICNN is used to form an implicit layer to which classification can be delegated in an amortized way. By contrast, our construction builds upon the property that $\nabla_1 \psi_\theta(x, c)$ describes a parameterized family of OT maps, and the convexity holds on our input space.

Given inputs x (samples of source distribution in $\mathbb{R}^d$) and condition $c \in \mathcal{C}$, we define the PICNN over (x, c). We assume for now that $\mathcal{C} \subset \mathbb{R}^q$ and is encoded as a vector. When $\mathcal{C}$ is not a vector space, we use embeddings or combinators as described in §3.2. A K-layer PICNN is defined as:

$$\begin{aligned} u_{k+1} &= \tau_k \left(V_k u_k + v_k \right) \\ z_{k+1} &= \sigma_k \left(W_k^z \left(z_k \circ [W_k^{zu} u_k + b_k^z]_+ \right) + W_k^x \left(x \circ [W_k^{xu} u_k + b_k^x]_+ \right) + W_k^u u_k + b_k^u \right) \\ \psi_\theta(x, c) &= z_k, \end{aligned} \tag{4}$$

where the PICNN is initialized as $u_0 = c$, $z_0 = \mathbf{0}$, $\circ$ denotes the Hadamard elementwise product, and τ_k is any activation function. The parameters of the PICNN are then given by

$$\theta = \{V_k, W_k^z, W_k^{zu}, W_k^x, W_k^{xu}, W_k^u, v_k, b_k^z, b_k^x, b_k^u\}.$$

Similar to ICNNs, the convexity w.r.t. input variable x is guaranteed as long as activation functions σ_i are convex and non-decreasing, and weight matrices W_k^z have non-negative entries. We parameterize this by storing them as elementwise applications of softplus operations on precursor matrices of the same size. Finally, all matrices at the $K - 1$ layer are line vectors, and their biases scalars.

3.2 Conditional Monge Map Architecture

Using PICNNs as a base module, the CONDOT architecture integrates operations on the contexts $\mathcal{C}$. As seen in Figure 3, context values c may take various forms:

1. A scalar t denoting a strength or a temporal effect. McCann's interpolation and its time parameterization, $\alpha_t = ((1-t)\mathrm{Id} + tT)_\sharp \alpha_0$ (McCann, 1997) can be reinterpreted as a conditional OT model by simply stating, from an OT map ψ_θ that $\mathcal{T}_\theta(t) := x \mapsto \nabla_x \left((1-t)/2 \|x\|^2 + t\psi_\theta(x) \right)$.
2. A covariate vector influencing the nature of the effect that led μ_i to ν_i.
3. One or multiple actions, possibly discrete, representing decisions or perturbations applied onto μ_i.

To provide a flexible architecture capable of modeling different types of conditions as well as conditions appearing in combinations, the more general CONDOT architecture consists of the hypernetwork $\mathcal{T}_\theta$ that is itself given a context vector through embedding and combinator modules. This generic architecture provides a one-size fits all approach to integrate all types of latent conditions c.

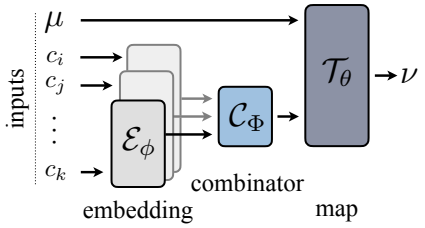

Figure 2: **CONDOT Architecture and Modules**. The embedding module $\mathcal{E}_\phi$ embeds arbitrary conditions c, which are then combined via module $\mathcal{C}_\Phi$. Using the processed conditions summarized in c, the map $\mathcal{T}_\theta(c)$ acts on μ to predict the target measure ν.

Embedding Module To give greater flexibility when setting the context variable c, CONDOT contains an embedding module $\mathcal{E}$ that translates arbitrary contexts into real-valued vectors. Besides simple scalars t (Fig. 3) for which no embedding is required, discrete contexts can be handled with an embedding module $\mathcal{E}_\phi$. When the set $\mathcal{C}$ is small, this can be done effectively using one-hot embeddings $\mathcal{E}_{\text{ohe}}$. For more complicated actions a such as treatments, there is no simple way to vectorize a context c. Similarly to action embeddings in reinforcement learning (Chandak et al., 2019; Tennenholtz and Mannor, 2019), we can learn embeddings for discrete actions into a learned continuous representation. This often requires domain-knowledge on the context values. For molecular drugs, for example, we can learn molecular representations $\mathcal{E}_{\text{mol}}$ such as chemical, motif-based (Rogers and Hahn, 2010) or neural fingerprints (Rong et al., 2020). However, often this domain knowledge is not available. In this work we thus construct so-called *mode-of-action* embeddings, by computing an embedding $\mathcal{E}_{\text{moa}}$ that allows actions a with similar

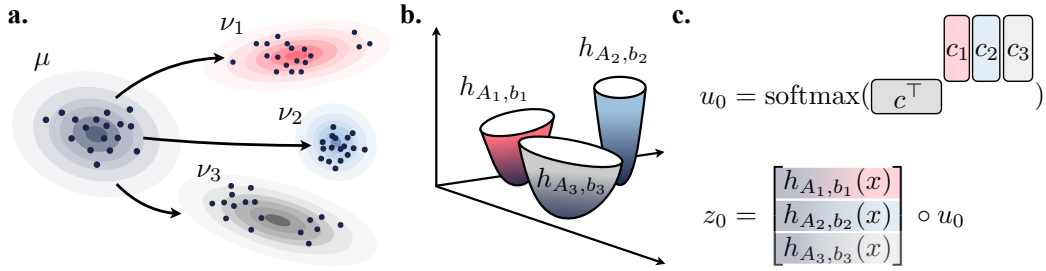

Figure 3: **a.** From a measure μ to several target measures ν_1, ν_2, ν_3 provided with labels c_1, c_2, c_3 we can extract three Gaussian (quadratic) potentials in closed form, **b.** whose gradients transport on a first approximation μ to areas in space that cover the three targets. **c.** Given a new label vector c, we compare it to known labels to modulate the magnitude of each of the three potentials.

effect on target population ν to have a similar representation. In § 5, we analyze several embedding types for different use-cases.

Combinator Module While we often have access to contexts c in isolation, it is crucial to infer the effect of contexts applied in combination. A prominent example are cancer combination therapies, in which multiple treatment modalities are administered in combination to enhance treatment efficacy (Kummar et al., 2010). In these settings, the mode of operation between individual contexts c is often not known, and can thus not be directly modeled via simple arithmetic operations such as `min`, `max`, `sum`, `mean`. While we test as a baseline the case, applicable to one-hot-embeddings, where simple additions are used to model these combinations, we propose to augment the CONDOT architecture with a parameterized combinator module $\mathcal{C}_\Phi$. If the order in which the actions are applied is irrelevant or unknown, the corresponding network $\mathcal{C}_\Phi$ needs to be permutation-invariant, which can be achieved by using a deep set architecture (Zaheer et al., 2017). Receiving a flexible number of inputs from the embedding module $\mathcal{E}_\phi$, CONDOT allows for a joint training of the PICNN parameters θ, combinator parameters Φ, and embedding parameters ϕ in a single, end-to-end differentiable architecture.

4 Initialization Strategies for Neural Convex Architectures

We address the problem of initializing the parameters of (P)ICNNs to ensure their gradient evaluated at every point is (initially) meaningful in the context of OT, namely that it is able to approximately map a measure μ into a target measure ν. The initializers we propose build heavily on closed-form solutions available for Gaussian approximations of measures (Gelbrich, 1990).

Closed-Form Potentials for Gaussians Given two Gaussian distributions $\mathcal{N}_1, \mathcal{N}_2$ with means respectively $\mathbf{m}_1, \mathbf{m}_2$ and covariance matrices Σ_1, Σ_2 (where Σ_1 is assumed to be full rank), the Brenier potential solving the OT problem from the first to the second Gaussian reads:

$$f^\star_{\mathcal{N}_1,\mathcal{N}_2} = \tfrac{1}{2} x^T A^T A x + b^T x + t(A,b) = \tfrac{1}{2}\|Ax\|_2^2 + b^T x + t(A,b), \text{ where,}$$
$$A := \left(\Sigma_1^{-1/2}\left(\Sigma_1^{1/2}\Sigma_2\Sigma_1^{1/2}\right)^{1/2}\Sigma_1^{-1/2}\right)^{1/2}, \quad b := \mathbf{m}_2 - A^T A \mathbf{m}_1, \tag{5}$$

define both quadratic and linear terms and $t(A,b)$ can be any constant. Importantly, note that we write the quadratic term in factorized form AA^T, not as usually done in the literature with a single psd matrix (Peyré and Cuturi, 2019, Remark 2.31). We favor this convention because it allows to parameterize directly the quadratic term in a manner that guarantees its positive-definiteness.

Quadratic Potentials Lower Bounded by 0 Naturally, for any choice of $t(A,b)$ one recovers the property that $\nabla f^\star_{\mu,\nu} \sharp \mathcal{N}_1 = \mathcal{N}_2$. When used in deep architectures, the level of that constant does, however, play a role, since convex functions in ICNN are typically thresholded or modulated using rectifying functions. To remove this ambiguity, we settle on a choice for $t(A,b)$ that is such that the lowest value reached by $f^\star_{\mathcal{N}_1,\mathcal{N}_2}$ is 0. This can be obtained by setting

$$t(A,b) := b^T (A^T A)^{-1} b\,, \tag{6}$$

Table 1: Evaluation of drug effect predictions from control cells to cells treated with drug Givinostat when conditioning on various covariates influencing cellular responses such as drug dosage and cell type. Results are reported based on MMD and the ℓ_2 distance between drug scores in a 1000 dimensional gene expression space.

Method	**Conditioned on Drug Dosage**				**Conditioned on Cell Line**	
	In-Sample		Out-of-Sample		In-Sample	
	MMD	ℓ_2(DS)	MMD	ℓ_2(DS)	MMD	ℓ_2(DS)
CPA (Lotfollahi et al., 2021)	0.1502 ± 0.0769	2.47 ± 2.89	0.1568 ± 0.0729	2.65 ± 2.75	0.2551 ± 0.006	2.71 ± 1.51
ICNN OT (Makkuva et al., 2020)	0.0365 ± 0.0473	2.37 ± 2.15	0.0466 ± 0.0479	2.24 ± 2.39	0.0206 ± 0.0109	1.16 ± 0.75
CondOT (Identity initialization)	0.0111 ± 0.0055	0.63 ± 0.09	0.0374 ± 0.0052	2.02 ± 0.10	0.0148 ± 0.0078	0.39 ± 0.06
CondOT (Gaussian initialization)	0.0128 ± 0.0081	0.60 ± 0.11	0.0325 ± 0.0062	1.84 ± 0.14	0.0146 ± 0.0074	0.41 ± 0.07

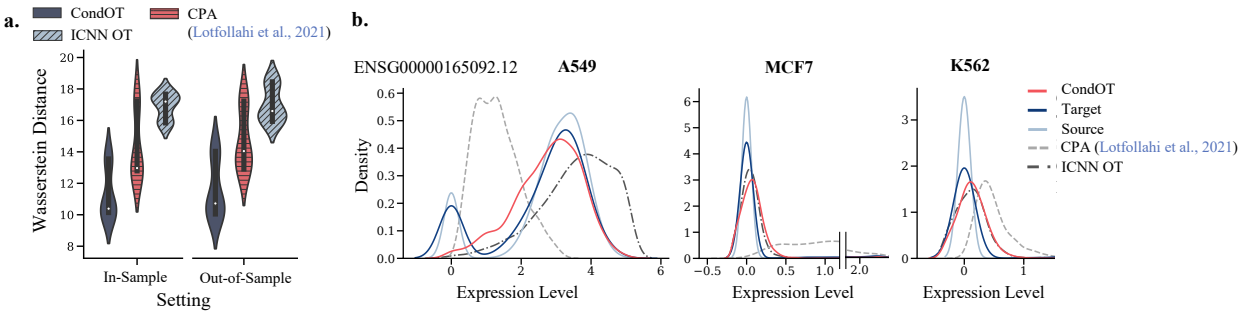

Figure 4: **a.** Predictive performance of CondOT and baselines w.r.t. the entropy-regularized Wasserstein distance on drug dosages *in-sample*, i.e., seen during training, and *out-of-sample*, i.e., unseen during training. **b.** Marginal distributions of observed source and target distributions, as well as predictions on perturbed distributions by CondOT and baselines of an exemplary gene across different cell lines. Predicted marginals of each method should match the marginal of the target population.

which results in the following choice, writing $\omega = \mathbf{m}_1 - (A^T A)^{-1}\mathbf{m}_2$,

$$f^{\star}_{\mathcal{N}_1,\mathcal{N}_2}(x) = \tfrac{1}{2}\|A\left(x + (A^T A)^{-1} b\right)\|_2^2 = \tfrac{1}{2}\|A\left(x - \omega\right)\|_2^2 \,. \tag{7}$$

To mimic these potential functions, we introduce a quadratic *layer* parameterized by a weight matrix M and a "bias" vector m, defined as $q_{M,m}(x) = \frac{1}{2}\|M(x-m)\|_2^2$. By design, $q_{M,m}(x)$ is a convex quadratic, non-negative layer. Additionally, one has the following relationships:

$$\nabla q_{I,\mathbf{0}_d} = \mathrm{Id}, \quad \nabla q_{A,\omega} \sharp \mathcal{N}_1 = \mathcal{N}_2 \,. \tag{8}$$

ICNN Initialization We explore two possible ICNN (3) initializers for OT.

Identity Initialization. The first approach ensures that upon initialization the ICNN's gradient mimics the *identity* map, i.e., $\nabla\psi_\theta(x) = x$ for any x. We do so by injecting in the initial hidden state z_0 the norm of the input vector $\frac{1}{2}\|x\|^2$, cast as a trainable layer $q_{M,m}$ initialized with $M = I$ and $m = \mathbf{0}_d$, see (8). The remaining parameters are chosen to propagate that norm throughout layers using averages. This amounts to the following choices:

1. Set all σ_i to be activations such that $\sigma_i'(u) \approx 1$ for u large enough, e.g. (leaky) ReLU or softplus.
2. Introduce an initialization layer, $z_0 = q_{M,m}(x)\mathbf{1}$, itself initialized with $M = I$ and $m = \mathbf{0}_d$.
3. Initialize all matrices W_i^z to $\approx \mathbf{1}_{d_2,d_1}/d_1$, where d_1, d_2 are the dimensions of these matrices.
4. Initialize all matrices W_i^x to ≈ 0.
5. Initialize biases b_i to $s\mathbf{1}$, where s is a large enough value s so that $\sigma_i'(s) \approx 1$.

Gaussian Initialization. The second approach can be used to initialize an ICNN so that its gradient mimics the affine transport between the Gaussian approximations of μ and ν. To this end, we follow all of the steps outlined above, except for step 2 where the quadratic layer $q_{M,m}$ is initialized instead with $M = A$ and $m = \mathbf{m}_1 - (A^T A)^{-1}\mathbf{m}_2$ using notations in (5), (6) (7), where $\mathbf{m}_1, \mathbf{m}_2, \Sigma_1, \Sigma_2$ are replaced by the empirical mean and covariances of μ and ν.

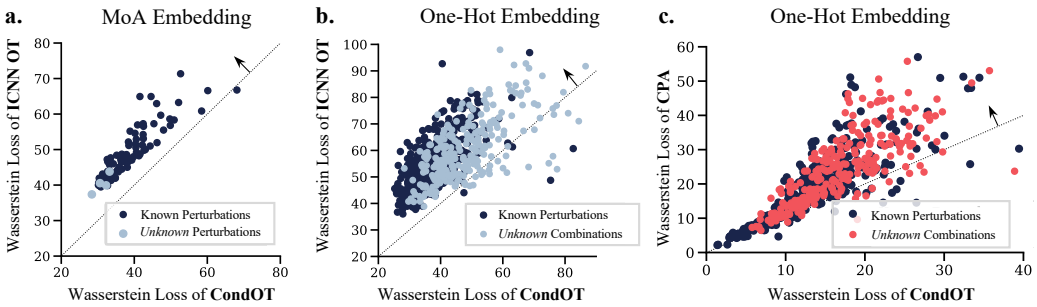

Figure 5: Comparison between **a.** CONDOT and ICNN OT (Makkuva et al., 2020) based on embedding $\mathcal{E}_{\text{moa}}$ **b.** as well as $\mathcal{E}_{\text{ohe}}$, and **c.** CONDOT and CPA (Lotfollahi et al., 2021) based on embedding $\mathcal{E}_{\text{ohe}}$ on *known* and *unknown* perturbations or combinations. Results above the diagonal suggest higher predictive performance of CONDOT.

PICNN Initialization Recall for convenience that a K-layer PICNN architecture reads:

$$\begin{aligned} u_{k+1} &= \tau_k\left(V_k u_k + v_k\right) \\ z_{k+1} &= \sigma_k\left(W_k^z\left(z_k \circ [W_k^{zu} u_k + b_k^z]_+\right) + W_k^x\left(x \circ [W_k^{xu} u_k + b_k^x]_+\right) + W_k^u u_k + b_k^u\right) \\ \psi_\theta(x, c) &= z_K. \end{aligned}$$

In their original form (Amos et al., 2017, Eq.3), PICNNs are initialized by setting $u_0 = c$ and $z_0 = \mathbf{0}$ to a zero vector of suitable size. Intuitively, the hidden states u_k act as context-dependent modulators, whereas vectors z_k propagate, layer after layer, a collection of convex functions in x that are iteratively refined, while retaining the property that they are each convex in x. A reasonable initialization for a PICNN that is provided a context vector c is that if $c \approx c_j$ (where j is in the training set), one has that $\nabla_1 \psi_{\theta_0}(\cdot, c)$ maps approximately μ_j to ν_j, which can be obtained by having $\psi_{\theta_0}(\cdot, c)$ mimic the closed-form Brenier potential between the Gaussian approximations of μ_j, ν_j. Alternatively, one may also default to an identity initialization as discusses above. To obtain either behavior, we make the following modifications, and refer to the illustration in Fig. 3:

1. The modulator $u_0(c) = \text{softmax}(c^T M)$, where C is initialized as $M = [c_j]_j$, and $V_i = I, v_i = \mathbf{0}$.
2. $z_0 = [q_{M_j, m_j}(x)]_j$, where weight matrices and bias (M_j, m_j) are either initialized to $(I, \mathbf{0})$ or as (A_j, ω_j) recovered by solving the Gaussian affine map from μ_j to ν_j using (7).
3. Modulator u_0 is passed directly to hidden state upon first iteration $W_0^{zu} = I, b_0^z = 0$.
4. All subsequent matrices W_k^z are initialized to $\approx \mathbf{1}_{d_2, d_1}/d_1$ where d_1, d_2 are their dimensions,
5. W_k^x and W_k^{xu} are ≈ 0, the biases $b_k^z \approx \mathbf{1}$, $b_k^u \approx \mathbf{0}$.

5 Evaluation

In the field of single-cell biology, cells undergo changes in their molecular profiles upon chemical, genetic, or mechanical perturbations. These changes can be measured using recent technological advancements in high-resolution multivariate single-cell data. Measuring single cells in their unperturbed or perturbed state requires, however, to destroy them, resulting in populations μ and ν that are unpaired. The relevance of OT to that comes from its ability to resolve such ambiguities through OT maps, that hold the promises of a better understanding of human health and disease. We consider various high-dimensional problems arising from this scenario to evaluate the performance of CONDOT § 3 versus other baselines.

5.1 Population Dynamics Conditioned on *Scalars*

Upon application of a molecular drug, the state of each cell x_i of the unperturbed population is altered, and observed in population ν. Molecular drugs are often applied at different dosage levels t, and the magnitude of changes in the gene expression profiles of single cells highly correlates with that dosage. We seek to learn a global, parameterized, transport map $\mathcal{T}_\theta$ sensitive to dosage.We evaluate our method on the task of inferring single-cell perturbation responses to the cancer drug Givinostat, a histone deacetylase inhibitor with potential anti-inflammatory, anti-angiogenic, and antineoplastic activities (Srivatsan et al., 2020), applied at different dosage levels, i.e., $t \in \{10, 100, 1000, 10000\}$.

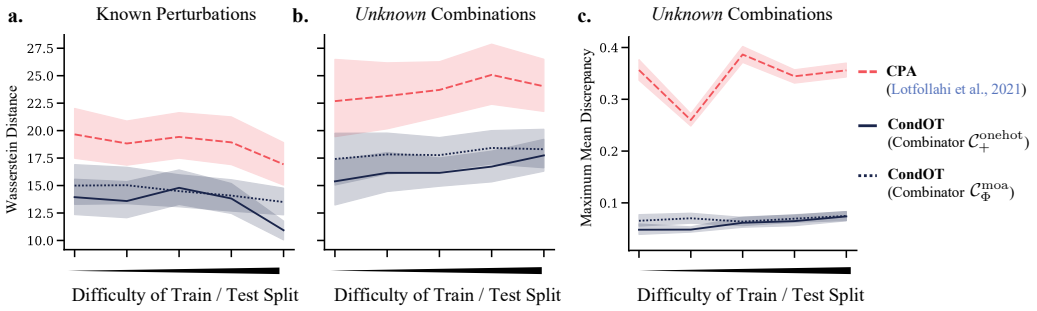

Figure 6: Predictive performance for **a.** known perturbations, **b.** unknown perturbations in combination w.r.t. regularized Wasserstein distance and **c.** MMD over different train / test splits of increasing difficulty for baseline CPA as well as CONDOT with different combinators $\mathcal{C}_{+}^{\text{onehot}}$ and $\mathcal{C}_{\Phi}^{\text{moa}}$.

The dataset contains $3,541$ cells described with the gene expression levels of $1,000$ highly-variable genes. In a first experiment, we measure how well CONDOT captures the drug effects at different dosage levels via distributional distances such as MMD (Gretton et al., 2012) and the ℓ_2-norm between the corresponding drug signatures (DS), as well as the entropy-regularized Wasserstein distance (Cuturi, 2013). We compute the metrics on 50 marker genes, i.e., genes mostly affected upon perturbation. For more details on evaluation metrics, see § E.2. To put CONDOT's performance into perspective, we compare it to current state-of-the-art baselines (Lotfollahi et al., 2021) as well as parameterized Monge maps without context variables (ICNN OT), see § E.1. As visible in Table 1 and Fig. 4a, CONDOT achieves consistently more accurate predictions of the target cell populations at different dosage levels than OT approaches that cannot rely on context information, as seen in a lower average loss and a smaller variance. This becomes even more evident when moving to the setting where the population has been trained only on a subset of dosages and we test CONDOT on *out-of-sample* dosages. Table 1 and Fig. 4a demonstrate that CONDOT is able to generalize to previously *unseen* dosages, thus learning to interpolate the perturbation effects from dosages seen during training. For further analysis, we refer the reader to § E (see Fig. 9 and 10).

5.2 Population Dynamics Conditioned on *Covariates*

Molecular processes are often highly dependent on additional covariates that steer experimental conditions, and which are not present in the features measures in population μ or ν. This can be, for instance, factors such as different cell types clustered within the populations. When the model can only be conditioned w.r.t. a small and *fixed* set of metadata information, such as cell types, it is sufficient to encode these contexts using a one-hot embedding module $\mathcal{E}_{\text{onehot}}$. To illustrate this problem, we consider cell populations comprising three different cell lines (A549, MCF7, and K562). As visible in Table 1, CONDOT outperforms current baselines which equally condition on covariate information such as CPA (Lotfollahi et al., 2021) on various evaluation metrics. Studying a gene showing highly various responses towards the drug Givinostat dependent on the cell line, CONDOT captures the distribution shift from control to target populations and consistently captures the perturbed cell state across different cell lines (see Fig. 4b).

5.3 Population Dynamics Conditioned on *Actions*

In order to recommend personalized medical procedures for patients, or to improve our understanding of the genome, it is key to be able to predict the outcomes of novel perturbations, arising from combinations of drugs or of genetic perturbations. Rather than learning individual maps T_θ^a predicting the effect of individual treatments, we aim at learning a global map $\mathcal{T}_\theta$ which, given as input the unperturbed population μ as well as the action a of interest, predicts the cell state perturbed by a. Thanks to its modularity, CONDOT can not only learn a map T_θ for all actions *seen* during training, but also to generalize to *unseen* actions, as well as potential *combinations* of actions. We will discuss all three scenarios below.

Known Actions In the following, we analyze CONDOT's ability to accurately predict phenotypes of genetic perturbations based on single-cell RNA-sequencing pooled CRISPR screens (Norman et al., 2019; Dixit et al., 2016), comprising $108,497$ single-cell gene expression profiles with 108 different

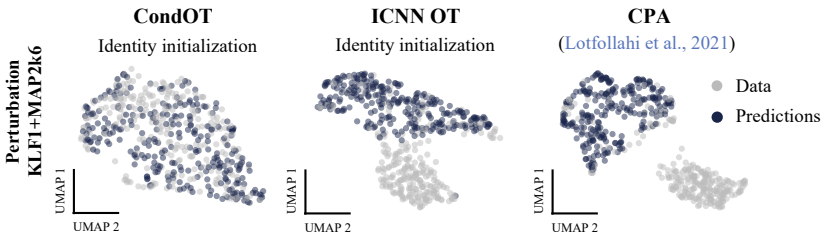

Figure 7: UMAP embeddings of cells perturbed by the combination KLF1+MAP2K6 and predictions of CONDOT (ours), ICNN OT (Makkuva et al., 2020), and CPA (Lotfollahi et al., 2021). While CONDOT covers the space of observed perturbed cells, the baselines fail to capture subpopulations.

genetic perturbations, each cell measured via a $1,000$ highly-variable gene expression vector. As, in a first step, we do not aim at generalizing beyond perturbations encountered during training, we utilize again a one-hot embedding $\mathcal{E}_{\text{onehot}}$ to condition $\mathcal{T}_\theta$ on each perturbation a. We compare our method to other baselines capable of modeling effects of a large set of perturbations such as CPA (Lotfollahi et al., 2021). Often, the effect of genetic perturbations are subtle in the high-dimensional gene expression profile of single cells. Using ICNN-parameterized OT maps without context information, we can thus assess the gain in accuracy of predicting the perturbed target population by incorporating context-awareness over simply predicting an average perturbation effect. Figure 5a and b demonstrate that compared to OT ablation studies as well as the current state-of-the-art CPA (Lotfollahi et al., 2021), CONDOT captures the perturbation responses more accurately w.r.t. the Wasserstein distance.

Unknown Actions With the emergence of new perturbations or drugs, we aim at inferring cellular responses to settings not explored during training. One-hot embeddings, however, do not allow us to model *unknown* perturbations. This requires us to use an embedding $\mathcal{E}$, which can provide us with a representation of an unseen action a'. As genetic perturbations further have no meaningful embeddings such as molecular fingerprints for drugs, we resort to mode-of-action embeddings introduced in § 3.2. Assuming marginal sample access to all individual perturbations, we compute a multidimensional scaling (MDS)-based embedding from pairwise Wasserstein distances between individual target populations, such that perturbations with similar effects are closely represented. For details, see § E. As current state-of-the-art methods are restricted to modeling perturbations via one-hot encodings, we compare to ICNN OT only. As displayed in Fig. 5a, CONDOT accurately captures the response of *unseen* actions (BAK1, FOXF1, MAP2K6, MAP4K3), which were not seen during training, at a similar Wasserstein loss as perturbation effects seen during training. For more details, see § E.

Combinations While experimental studies can often measure perturbation effects in biological systems in isolation, the combinatorial space of perturbations in composition is too large to capture experimentally. Often, however, combination therapies are cornerstones of cancer therapy (Mokhtari et al., 2017). In the following we test different combinator architectures to predict genetic perturbations in combination from single targets. Similarly to Lotfollahi et al. (2021), we can embed combinations by adding individual one-hot encodings of single perturbations (i.e., $\mathcal{C}_+^{\text{onehot}}$). In addition, we parameterize a combinator via a permutation-invariant Deep Set, as introduced in § 3.2 based on mode-of-action embeddings of individual perturbations (i.e., $\mathcal{C}_\Phi^{\text{moa}}$). We split the dataset into train / test splits of increasing difficulty. Initially containing all individual perturbations as well as some combinations, the number of perturbations seen in combination during training decreases over each split. For more details, see § E. We compare different combinators to ICNN OT (Fig. 5b) and CPA (Lotfollahi et al., 2021) (Fig. 5c, Fig. 6). While the performance drops compared to inference on *known* perturbations and decreases with increasing difficulty of the train / test split, CONDOT outperforms all baselines. When embedding these high-dimensional populations in a low-dimensional UMAP space (McInnes et al., 2018), one can see that CONDOT captures the entire perturbed population, while ICNN OT and CPA fail in capturing entire subpopulations in the perturbed state.

6 Conclusion

We have developed the CONDOT framework that able to infer OT maps from not only one pair of measures, but many pairs that come labeled with a context value. The generalization abilities of CONDOT, showcased in the extremely challenging task of predicting outcomes for unseen combinations of treatments, hold several promises, both as an augmentation of the set of tools provided by OT, and for future applications of OT to single cell genomics.

References

D. Alvarez-Melis, Y. Schiff, and Y. Mroueh. Optimizing Functionals on the Space of Probabilities with Input Convex Neural Networks. *arXiv Preprint arXiv:2106.00774*, 2021.

B. Amos, L. Xu, and J. Z. Kolter. Input Convex Neural Networks. In *International Conference on Machine Learning (ICML)*, volume 34, 2017.

M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein Generative Adversarial Networks. In *International Conference on Machine Learning (ICML)*. PMLR, 2017.

Y. Brenier. Décomposition polaire et réarrangement monotone des champs de vecteurs. *CR Acad. Sci. Paris Sér. I Math.*, 305, 1987.

C. Bunne, S. G. Stark, G. Gut, J. S. del Castillo, K.-V. Lehmann, L. Pelkmans, A. Krause, and G. Ratsch. Learning Single-Cell Perturbation Responses using Neural Optimal Transport. *bioRxiv*, 2021.

C. Bunne, L. Meng-Papaxanthos, A. Krause, and M. Cuturi. Proximal Optimal Transport Modeling of Population Dynamics. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 25, 2022.

Y. Chandak, G. Theocharous, J. Kostas, S. Jordan, and P. Thomas. Learning Action Representations for Reinforcement Learning. In *International Conference on Machine Learning (ICML)*, 2019.

Y. Chen, Y. Shi, and B. Zhang. Optimal Control Via Neural Networks: A Convex Approach. In *International Conference on Learning Representations (ICLR)*, 2019.

M. Cuturi. Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 26, 2013.

A. Dixit, O. Parnas, B. Li, J. Chen, C. P. Fulco, L. Jerby-Arnon, N. D. Marjanovic, D. Dionne, T. Burks, R. Raychowdhury, et al. Perturb-Seq: Dissecting Molecular Circuits with Scalable Single-Cell RNA Profiling of Pooled Genetic Screens. *Cell*, 167(7), 2016.

M. Gelbrich. On a formula for the l^2 wasserstein metric between measures on euclidean and hilbert spaces. *Mathematische Nachrichten*, 147(1):185–203, 1990.

A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012.

D. Ha, A. Dai, and Q. V. Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016.

T. Hashimoto, D. Gifford, and T. Jaakkola. Learning Population-Level Diffusions with Generative Recurrent Networks. In *International Conference on Machine Learning (ICML)*, volume 33, 2016.

C.-W. Huang, R. T. Q. Chen, C. Tsirigotis, and A. Courville. Convex Potential Flows: Universal Probability Distributions with Optimal Transport and Convex Optimization. In *International Conference on Learning Representations (ICLR)*, 2021.

L. Jacob, J. She, A. Almahairi, S. Rajeswar, and A. Courville. W2GAN: Recovering an Optimal Transport Map with a GAN. 2018.

H. Janati, T. Bazeille, B. Thirion, M. Cuturi, and A. Gramfort. Multi-subject meg/eeg source imaging with sparse multi-task regression. *NeuroImage*, 220:116847, 2020.

A. Korotin, V. Egiazarian, A. Asadulaev, A. Safin, and E. Burnaev. Wasserstein-2 generative networks. In *International Conference on Learning Representations*, 2020.

S. Kummar, H. X. Chen, J. Wright, S. Holbeck, M. D. Millin, J. Tomaszewski, J. Zweibel, J. Collins, and J. H. Doroshow. Utilizing targeted cancer therapeutic agents in combination: novel approaches and urgent requirements. *Nature Reviews Drug discovery*, 9(11), 2010.

M. Lotfollahi, A. K. Susmelj, C. De Donno, Y. Ji, I. L. Ibarra, F. A. Wolf, N. Yakubova, F. J. Theis, and D. Lopez-Paz. Compositional perturbation autoencoder for single-cell response modeling. *bioRxiv*, 2021.

A. Makkuva, A. Taghvaei, S. Oh, and J. Lee. Optimal transport mapping via input convex neural networks. In *International Conference on Machine Learning (ICML)*, volume 37, 2020.

R. J. McCann. A convexity principle for interacting gases. *Advances in mathematics*, 128(1):153–179, 1997.

L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv Preprint*, 2018.

R. B. Mokhtari, T. S. Homayouni, N. Baluch, E. Morgatskaya, S. Kumar, B. Das, and H. Yeger. Combination therapy in combating cancer. *Oncotarget*, 8(23):38022, 2017.

P. Mokrov, A. Korotin, L. Li, A. Genevay, J. Solomon, and E. Burnaev. Large-Scale Wasserstein Gradient Flows. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

T. M. Norman, M. A. Horlbeck, J. M. Replogle, A. Y. Ge, A. Xu, M. Jost, L. A. Gilbert, and J. S. Weissman. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 365(6455), 2019.

G. Peyré and M. Cuturi. Computational optimal transport. *Foundations and Trends in Machine Learning*, 11(5-6), 2019. ISSN 1935-8245.

J. Richter-Powell, J. Lorraine, and B. Amos. Input convex gradient networks. *arXiv preprint arXiv:2111.12187*, 2021.

D. Rogers and M. Hahn. Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5), 2010.

Y. Rong, Y. Bian, T. Xu, W. Xie, Y. Wei, W. Huang, and J. Huang. Self-Supervised Graph Transformer on Large-Scale Molecular Data. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

F. Santambrogio. Optimal Transport for Applied Mathematicians. *Birkhäuser, NY*, 55(58-63):94, 2015.

G. Schiebinger, J. Shu, M. Tabaka, B. Cleary, V. Subramanian, A. Solomon, J. Gould, S. Liu, S. Lin, P. Berube, et al. Optimal-Transport Analysis of Single-Cell Gene Expression Identifies Developmental Trajectories in Reprogramming. *Cell*, 176(4), 2019.

S. R. Srivatsan, J. L. McFaline-Figueroa, V. Ramani, L. Saunders, J. Cao, J. Packer, H. A. Pliner, D. L. Jackson, R. M. Daza, L. Christiansen, et al. Massively multiplex chemical transcriptomics at single-cell resolution. *Science*, 367(6473), 2020.

G. Tennenholtz and S. Mannor. The Natural Language of Actions. In *International Conference on Machine Learning (ICML)*, 2019.

K. D. Yang and C. Uhler. Scalable Unbalanced Optimal Transport using Generative Adversarial Networks. *International Conference on Learning Representations (ICLR)*, 2019.

K. D. Yang, K. Damodaran, S. Venkatachalapathy, A. C. Soylemezoglu, G. Shivashankar, and C. Uhler. Predicting cell lineages using autoencoders and optimal transport. *PLoS Computational Biology*, 16(4), 2020.

M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. Deep Sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The code to reproduce the datasets will be available upon publication of this work. The datasets are public and the sources are referenced in the paper § D. The data processing pipeline and further experimental details such as hyperparameter or algorithmic decisions are outlined in § E.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] The datasets and corresponding splits are described in § D. The hyperparameter setup in all experiments is identical and outlined in § E.3. The baseline configurations are described in § E.1.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]