Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)
Survey Track

Optimal Transport for Deep Generative Models: State of the Art and Research Challenges

Viet Huynh[1] , **Dinh Phung**[1,2] , **He Zhao**[1]
[1]Department of Data Science and Artificial Intelligence, Monash University, Australia
[2] VinAI Research, Vietnam
{viet.huynh, dinh.phung, ethan.zhao}@monash.edu

Abstract

Optimal transport has a long history in mathematics which was proposed by Gaspard Monge in the eighteenth century [Monge, 1781]. However, until recently, advances in optimal transport theory pave the way for its use in the AI community, particularly for formulating deep generative models. In this paper, we provide a comprehensive overview of the literature in the field of deep generative models using optimal transport theory with an aim of providing a systematic review as well as outstanding problems and more importantly, open research opportunities to use the tools from the established optimal transport theory in the deep generative model domain.

1 Motivation, Optimal Transport and Deep Generative Models

Optimal transport and deep generative models have attracted substantial attention in the machine learning and artificial intelligence community in recent years. Tremendous efforts have been made to leverage to combine the recent development of the optimal transport to formulating and learning deep generative models. The efforts have been resulting in a rich literature of related publications and methodologies in using optimal transport distance with deep generative models. Therefore, a comprehensive and systematic survey reviewing and categorizing existing approaches and methodologies, recognizing outstanding research questions, and discussing open challenges and future directions is imperative yet missing.

To this end, we would like to provide a systematic review of deep generative models using optimal transport. We categorize models and methods on the way optimal transport distance is used to formulate the problems. The goal is to help interested researchers address outstanding problems and more importantly, identify open research opportunities to use the tools from the established optimal transport theory in the deep generative model domain. To the best of our knowledge, this is the first comprehensive survey on deep generative models using optimal transport theory. The notable contributions of our survey can be summarized as follows: 1) summarizing and categorizing deep generative models based on their optimal transport distance formulation; 2) addressing the limitations of existing methods and suggest several open challenges and potential future research directions in the field of using optimal transport theory for deep generative models.

The rest of this paper is organized as follows. Sections 1.1 and 1.2 introduce the background and definitions of two main classes of deep generative models and optimal transport distances. Section 2 reviews optimal transport based deep generative models categorized by the formulation of optimal distance. Section 3 discusses the challenges and potential future research directions.

Notations. Let $\mathcal{X}$ be a compact metric space. We denote by $\mathcal{P}(\mathcal{X})$ the set of probability measures over $\mathcal{X}$, each of which is referred to as a probability distribution. For any measurable function ϕ from $\mathcal{X}$ to $\mathbb{R}$, and a metric c, $\text{Lip}_c(\phi)$ denotes the Lipschitz constant of ϕ with respect to c and $\mathcal{L}_c \triangleq \{\phi : \text{Lip}_c(\phi) \leq 1\}$. We use capital letters to denote random variables, e.g. X, while bold lower case letters are used for vectors. Probability distribution of random variable X will be denoted as P_X with the probability density as $p_X(\boldsymbol{x})$. When the context is clear the subscript of probability density will be removed, e.g. $p(\boldsymbol{x})$.

1.1 Deep Generative Models

Deep generative model is a deep neural network based framework for estimating a probability distribution that is "close" to empirical data samples $\{\boldsymbol{x}_i\}_{i=1}^n$ which come from an unknown data distribution P_X over the data space $\mathcal{X}$. Model distribution P_θ over $\mathcal{X}$ is usually assumed as a latent variable model

$$p_\theta(\boldsymbol{x}) = \int_{\mathcal{Z}} p(\boldsymbol{x} \mid \boldsymbol{z})\, p(\boldsymbol{z})\, d\boldsymbol{z}, \tag{1}$$

where $p(\boldsymbol{z})$ is some simple well-known distribution, e.g. normal or uniform distributions while the conditional distribution $p(\boldsymbol{x} \mid \boldsymbol{z})$ can be either deterministic (i.e. Dirac) or stochastic. Since the model distribution P_θ is defined via deep neural networks, it may have no analytical form but can be easy to sample from. In the following sub-sections, we review two classes of deep generative models which are considered as dual (Generative Adversarial Network - GAN) and primal formulations (Variational Autoencoder - VAE) of the problem of minimizing the similarity between P_X and P_θ.

Generative Adversarial Network (GAN)

Generative Adversarial Network (GAN) [Goodfellow *et al.*, 2014] is a framework for estimating empirical data samples $\{\boldsymbol{x}_i\}_{i=1}^n$ by using two deep neural networks contesting with each other in a game. The difference between GAN and density estimation methods, i.e. maximum likelihood estimation (MLE), is that we do not need to define an explicit form of the estimator. In MLE, we usually define the model P_θ as an explicit parametric family of densities $\{p_\theta\}_{\theta\in\mathbb{R}^d}$ and try to solve the maximum likelihood problem $\max_{\theta\in\mathbb{R}^d}\sum_{i=1}^n \log p_\theta(x_i)$, which is equivalent to minimizing the Kullback-Leibler divergence between the model p_θ and the empirical distribution of real data $p_x \approx \frac{1}{n}\sum_{i=1}^n \delta_{\boldsymbol{x}_i}$, i.e. $KL(P_X \parallel P_\theta) = \bar{p}_x$.

In GAN, we do not need to specify the density p_θ which might not exist or be difficult to characterize in high dimensional data [1]. Instead, we can define a mechanism that can generate samples that are close to those generated from real data distribution P_X (and approximated by empirical distribution $\bar{P}_X$). The generation process is defined through two steps: drawing a random variable Z from a fixed distribution P_Z, e.g., uniform distribution in $(0,1)$ and mapping it to the same space of the real data samples, with a function $G_\theta : \mathcal{Z} \to \mathcal{X}$ (usually a deep neural network) where θ is the parameter of the function. The induced distribution defined above is denoted as P_θ. We can vary θ to change the induced distribution and make it "close" to $\bar{P}_X$. To define closeness, some divergence between distributions is used. In vanilla GAN [Goodfellow *et al.*, 2014], the authors used the Jensen-Shannon divergence between $\bar{P}_X$ and P_θ

$$JS\left(\bar{P}_X \parallel P_\theta\right) = KL\left(\bar{P}_X \parallel P_m\right) + KL\left(P_\theta \parallel P_m\right), \quad (2)$$

where $P_m = \frac{P_\theta + \bar{P}_X}{2}$. A GAN model can be considered as a latent variable model in Eq 1 in which the conditional distribution $P(\boldsymbol{x} \mid \boldsymbol{z})$ is defined using a deterministic map $G_\theta : \mathcal{Z} \to \mathcal{X}$ where $\mathcal{Z}$ is latent space, i.e., $p(\boldsymbol{x} \mid \boldsymbol{z}) = \mathbf{1}_x(G(\boldsymbol{z}))$ where $\mathbf{1}_x$ is the indicator function of x.

Variational Auto-Encoders

Variational Auto-encoders (VAE) is a sub-class of latent models in Eq. 1 in which the latent distributions are (standard) Gaussian distributions, i.e. $p(z) = \mathcal{N}(\boldsymbol{z} \mid \mathbf{0}, \boldsymbol{I})$, and the conditional distribution $p(\boldsymbol{x} \mid \boldsymbol{z}) = p(\boldsymbol{x} \mid f_\theta(z))$ is the likelihood with parameters defined using non-linear functions (typically a deep neural networks) $f_\theta(z)$ parameterized by $\boldsymbol{\theta}$. For instance if the likelihood is a Gaussian distribution, then $p(\boldsymbol{x} \mid \boldsymbol{z}) = \mathcal{N}\left(\boldsymbol{\mu}_\theta(\boldsymbol{z}), \operatorname{diag}\left(\sigma_\theta^2(\boldsymbol{z})\right)\right)$ where $\boldsymbol{\mu}_\theta(\boldsymbol{z})$ and $\sigma_\theta^2(\boldsymbol{z})$ are non-linear functions with parameters $\boldsymbol{\theta}$. Unfortunately, marginal distribution $p(\boldsymbol{x})$ is not tractable due to the non-linearity of deep neural networks used to define the conditional distributions. Variational inference therefore is used to approximate the posterior distribution $P_{Z|X}$ (and the marginal distribution P_X). Typically, the variational conditional distribution $q_{\boldsymbol{\omega}}(\boldsymbol{z} \mid \boldsymbol{x}) = \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\omega}}(\boldsymbol{x}), \Sigma_{\boldsymbol{\omega}}(\boldsymbol{x}))$ is used to approximate the conditional distribution where $\boldsymbol{\mu}_{\boldsymbol{\omega}}$ and $\Sigma_{\boldsymbol{\omega}}$ are two non-linear functions aka deep neural networks[2] with parameters $\boldsymbol{\omega}$. We can use the Kullback-Leibler divergence to measure the difference between empirical distribution P_X and marginal distribution P_θ, i.e. $\inf_\theta KL(P_X, P_\theta)$ which is equivalent to optimize the Evidence Lower BOund (ELBO) $\inf_{\theta,\lambda} - \mathop{\mathbb{E}}_{q(\boldsymbol{z}|\boldsymbol{x})}[\ln p(\boldsymbol{x} \mid \boldsymbol{z})] + \mathop{\mathbb{E}}_{p(\boldsymbol{x})}\left[KL\left(Q_{Z|X}, P_Z\right)\right]$.

Applications of Deep Generative Models

Deep generative models have a wide range of applications real-world problems from image processing such as super resolution (generating images with higher pixel resolution) [Ledig *et al.*, 2017; Vasu *et al.*, 2018], image-to-image translation [Isola *et al.*, 2017], photo inpainting [Pathak *et al.*, 2016]; game simulation [Kim *et al.*, 2020]; healthcare analytics [Frid-Adar *et al.*, 2018; Yoon *et al.*, 2019; Costa *et al.*, 2017; Choi *et al.*, 2017]; drug discovery [Jin *et al.*, 2018; De Cao and Kipf, 2018]; finance [Takahashi *et al.*, 2019]; bioinformatics [Marouf *et al.*, 2020; Anand and Huang, 2018]; and e-commerce [Kumar *et al.*, 2018]. Most of the applications are based on image-based generation tasks. The number of works that can generate data beyond images is still limited.

1.2 Optimal Transport Theory

In this section, we summarize optimal transport theory which provides the theoretical framework for defining problems of deep generative models

Primal Formulation

Given a continuous cost function $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, and two probability distributions $P_X,\ P_Y \in \mathcal{P}(\mathcal{X})$, the optimal transport distance of order p called *Wasserstein-p distance* is defined as the minimal cost for transporting from the density of p_x to that of p_y[Villani, 2008]

$$W_p(P_X, P_Y) = \left(\inf_{\pi\in\Pi}\int\int c^p(x,y)\,\pi(x,y)\,dxdy\right)^{1/p}, \quad (3)$$

where Π is the set of joint distributions with marginal constraints of $\int_{\mathcal{X}} \pi(x,y)\,dy = p_x$ and $\int_{\mathcal{X}} \pi(x,y)\,dy = p_x$.

Dual Formulation

Owing to the marginal constraint of the joint distribution, the optimization process in Eq. (3) is hard to employ. In practice, the celebrated dual from of the optimization in Eq. (3) for order $p = 1$ called *Kantorovich-Rubinstein duality* is often used [Villani, 2008]

$$W_1(P_X, P_Y) = \sup_{\phi\in\mathcal{L}_c}\int \phi(x)\,p_x(x)\,dx - \int \phi(y)\,p_y(y)\,dy \quad (4)$$

This dual form leads to a simpler optimization in comparison with the primal form which is used to formulate the Wasserstein GAN models in section 2.1.

Relaxed Formulation

In practice, an entropic regularization version of Wasserstein distance in Eq. 3 which is faster to compute in discrete cases was proposed in [Cuturi, 2013] and called Sinkhorn distance.

[1] Data may be in high dimensional space, but the support for density function may lie on low dimensional manifolds.

[2] called inference networks

The definition of Sinkhorn distance can be extended for continuous cases as follows

$$W_\epsilon(P_X, P_Y) = \inf_{\pi \in \Pi(P_X, P_Y)} \int\int c(x,y)\,\pi(x,y)\,dxdy + \epsilon KL(\pi \mid p_x \otimes p_y), \tag{5}$$

where $KL(\pi \mid p_x \otimes p_y) \triangleq \int\int \ln \frac{\pi(x,y)}{p_x(x)p_y(y)} dxdy$ is the relative entropy between the joint coupling and the marginals. However, the relaxed formulation of Sinkhorn distance in Eq. 5 is not a proper distance since $W_\epsilon(P_X, P_X) \neq 0$ for $\epsilon > 0$. Sinkhorn divergences [Ramdas *et al.*, 2017] are introduced to resolve the drawback

$$S_\epsilon(P_X, P_Y) = W_\epsilon(P_X, P_Y) - \frac{1}{2}W_\epsilon(P_X, P_X) - \frac{1}{2}W_\epsilon(P_Y, P_Y) \tag{6}$$

which becomes to Wasserstein distance with $\epsilon \to 0$ and reaches MMD with $\epsilon \to \infty$ [Ramdas *et al.*, 2017].

Approximated Formulation

Although Wasserstein distance on higher dimensions does not possess closed-form formulation in general, one can compute explicitly Wasserstein distance for one-dimensional measures [Villani, 2008]. This property motivates to approximate Wasserstein distance by projection probability distributions on high-dimensional to one-dimensional space [Bonneel *et al.*, 2015]

$$SW(P_X, P_Y) = \int_{S^d} W(P_{X_\theta}, P_{Y_\theta})\,d\boldsymbol{\theta}, \tag{7}$$

where $S^d = \left\{\boldsymbol{\theta} \in \mathbb{R}^d : \|\boldsymbol{\theta}\| = 1\right\}$ is the unit $d-$dimensional sphere, and $W(P_{X_\theta}, P_{Y_\theta})$ is 1-dimensional Wasserstein distance between 1-dimensional projected measures with $x_\theta = \langle \boldsymbol{x}, \boldsymbol{\theta} \rangle$ and $y_\theta = \langle \boldsymbol{y}, \boldsymbol{\theta} \rangle$ for $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$. The number of projections required to approximate sliced Wasserstein distance with a given accuracy are exponential proportion to data dimension [Kolouri *et al.*, 2019a]. To mitigate the projection complexity, max-sliced Wasserstein distance was proposed [Deshpande *et al.*, 2019]

$$\text{maxSW}(P_X, P_Y) = \max_{\theta \in S^d} W(P_{X_\theta}, P_{Y_\theta}). \tag{8}$$

A generalized version of sliced Wasserstein distance in Eq. 7 was introduced to replace the dot product in the projection with generalized Radon transform [Kolouri *et al.*, 2019a] $g : \mathcal{X} \times \left(\mathbb{R}^d \backslash \{0\}\right) \to \mathbb{R}$, i.e. $x_\theta = g(x, \theta)$ where $\theta \in \Omega \triangleq \mathbb{R}^d \backslash \{0\}$

$$GSW(P_X, P_Y) = \int_\Omega W(P_{X_\theta}, P_{Y_\theta})\,d\boldsymbol{\theta}.$$

Similarly, maximum generalized sliced Wasserstein distances are defined in a similar fashion in Eq. 8. Recently, [Nguyen *et al.*, 2021a] have introduced a slice-based variant called distributional sliced Wasserstein distance which that seeks for an optimal distribution over projections on the unit sphere.

$$DSW(P_X, P_Y) = \sup_{\rho(\theta) \in M_C} \mathop{\mathbb{E}}_{\rho(\theta)} \left[W(P_{X_\theta}, P_{Y_\theta})\right],$$

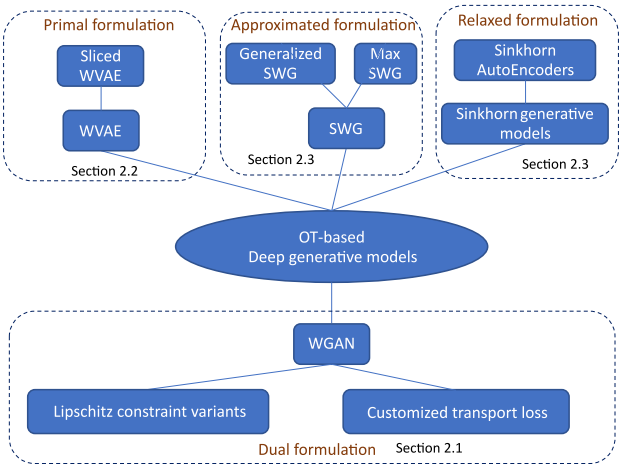

Figure 1: Categorization of optimal transport-based deep generative models. Models are categorized into four groups based on the optimal transport distance formulations.

where $M_C = \left\{\rho \in \mathcal{P}(S^d) \mid \mathop{\mathbb{E}}_{\theta,\theta' \sim \rho}\left[\left|\theta^\mathsf{T}\theta'\right|\right] \leq C\right\}$ is the set probability measures ρ on S^d satisfying $\mathop{\mathbb{E}}_{\theta,\theta' \sim \rho}\left[\left|\theta^\mathsf{T}\theta'\right|\right] \leq C$ and not empty.

Wasserstein Distance for Structured Objects

Gromov-Wasserstein distance: let $P_X \in \mathcal{P}\left(\mathbb{R}^{d_x}\right)$ and $P_Y \in \mathcal{P}\left(\mathbb{R}^{d_y}\right)$ be two probability measures on Euclidean spaces of different dimension d_x and d_y. Distance function $c_x : \mathbb{R}^{d_x} \times \mathbb{R}^{d_x} \to \mathbb{R}^+$(resp. $c_y : \mathbb{R}^{d_y} \times \mathbb{R}^{d_y} \to \mathbb{R}^+$) defines the similarity between samples in P_X (resp. P_Y) . The Gromov-Wasserstein distance of order p, $GW_p(P_X, P_Y)$, is defined as [Mémoli, 2011]

$$GW_p^p(P_X, P_Y) = \inf_{\pi \in \Pi(P_X, P_Y)} \mathop{\mathbb{E}}_{\pi}\left[\left|\left|c_x\left(x, x'\right) - c_y\left(y, y'\right)\right|\right|^p\right], \tag{9}$$

where Π is the set of joint distributions with marginal constraints defined similarly in Eq. (3).

2 Generative Models Using Optimal Transport

In this section we review and categorize deep generative models based on how optimal transport distance is used to formulate the objective functions. Figure 1 denotes the categorization of models based on the formulations of optimal transport described in Sec 1.2.

2.1 Dual Formulation

Wasserstein Generative Adversarial Network (WGAN)

One of the first works that had tried to use Wasserstein distance for learning deep generative models is the work by [Arjovsky *et al.*, 2017]. Wasserstein distance was proposed to use for learning since it has the weak*-topology property that allows defining the distance between two distributions with non-overlapped supports and is expected to alleviate the mode-collapsing problem of GANs. In Wasserstein GAN,

the authors suggested using the dual form Wasserstein distance of order 1 in Eq (4) instead of Jensen-Shannon divergence. The function $\phi_\omega(\cdot)$ in Eq (4) is parameterized using a (deep) neural networks of parameters ω. However, one of the important constraints for ϕ is the Lipschitz condition, i.e. $|\phi(x) - \phi(y)| \leq c(x,y)$[3] for all $x, y \in \mathcal{X}$, which is not always satisfied by an arbitrary deep network. To endow the constraint, Arjovsky et al. suggest a heuristic to clip weights of network to some certain ranges, e.g. the clipped box of $\omega \in [-0.01, 0.01]$. Despite the heuristically constraining the Lipschitz condition, Wasserstein GANs showed stability improvement in training processes. However, the weight clipping trick may lead to poor samples or fail to converge as observed in [Gulrajani *et al.*, 2017]. There are the following works [Gulrajani *et al.*, 2017; Petzka *et al.*, 2018; Wei *et al.*, 2018; Miyato *et al.*, 2018] that solve the 1-Lipschitz constraint with principled approaches.

Lipschitz Constraint Variants

[Gulrajani *et al.*, 2017, Corollary 1] have proved that the optimal critic function, $\phi*$, has gradients norm 1 almost everywhere. Hence, they proposed to train Wasserstein GAN with gradient penalty instead of weight clipping which results with a more stable and improved performance called WGAN gradient penalty (WGAN-GP) $R_{GP} = \mathop{\mathbb{E}}_{P_{\bar{X}}}\left[\left(\|\nabla_\omega \phi(x)\|_2 - 1\right)^2\right]$ where $\bar{X} = tX + (t-1)Y$ for $t \sim \text{Uni}(0,1)$ and $X \sim P_X$, $Y \sim P_\theta$ which respectively are real and generated samples. This is motivated by the fact that the optimal critic function $\phi*$ contains straight lines with gradient norm 1 connecting coupled points from P_X, and P_θ(cf. [Gulrajani *et al.*, 2017, Proposition 1]). [Petzka *et al.*, 2018] then proposed an alternative to gradient penalty called Lipschitz penalty which replace a l_2 norm of gradient in GP to l_1, $R_{LP} = \mathop{\mathbb{E}}_{P_{\bar{X}}}\left[\left(\max\{0, \|\nabla_w f(x)\| - 1\}\right)^2\right]$. Gradient and Lipschitz penalty regularizers only enforce Lipschitz constraint on a local data domain (not $\forall x, y \in \mathcal{X}$), therefore [Wei *et al.*, 2018] further improved the WGAN-GP by incorporating a consistency term to the objective function of WGAN-GP to enforce the (global) Lipschitz constraints of Wasserstein GAN. They defined a practical form of term CT as follows $R_{CT} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim P_X}[\max(0, c(x_1, x_2) + 0.1 \cdot d(x_{1-}, x_{2-}) - K)]$, where x_1, x_2 and x_{1-}, x_{2-} are perturbed embedded real data via critic ϕ, i.e. $x_{1,2} = \phi_{\text{drop}}(x)$ where ϕ_{drop} is a hidden layers dropout of ϕ [4]. Two vectors $x_{1-}$and x_{2-} are outputs of the last layer of ϕ_{drop} corresponding to x_1,x_2. Spectral normalization technique proposed by [Miyato *et al.*, 2018] allows to set the upper bound of the Lipschitz constant of ϕ. In this setting, the weight matrix in each layer is normalized by the spectral norm of that matrix which is equivalent to the largest singular value. A recent work by [Avraham *et al.*, 2019] has introduced an additional low dimensional representation (latent space) in parallel with original data and used Wasserstein distances on both latent space and original space[5].

Another line of work that aims to remove the Lipschitz condition in the dual form of Wasserstein distance is to reformulate it. The work of [Liu *et al.*, 2018] redefined the dual form in Eq. (3) as two-step of optimization: solving linear programming to approximate discretized critic function T and then regress the critic ϕ to fit T. Recent work of [Dam *et al.*, 2019] introduced a new function called mover to get rid of the Lipschitz constraint but a new function to optimize. Their new objective function becomes a *min-max-min* loss.

Customized Transport Loss Wasserstein GAN

As Wasserstein distance is defined based on cost function $c(\cdot,\cdot)$ which is reflected in the Lipschitz condition in the dual formulation. WGAN-GP used the l_2 norm of the gradient in the loss function which corresponds to the l_2 norm used for cost function c. Similarly, WGAN-LP implies the l_1 norm used. The underlying cost function reflects the geometry of the space generated data lying on. The work of [Adler and Lunz, 2018] has generalized the cost function of norms on Euclidean space to norms on Banach spaces such as Sobolev norms and L^p norms. These extensions allow practitioners more ranges of the cost functions to choose from to emphasize features they wish for generated data. The quadratic cost function was proposed to use in [Liu *et al.*, 2019] in which they used two-step optimization in [Liu *et al.*, 2018] for learning. A recent work by [Korotin *et al.*, 2021] also used the quadratic cost in their Wasserstein distance and used the input convex neural networks (ICNN) [Amos *et al.*, 2017] for approximating the distance. As Wasserstein distance is known as an effective distance in differentiating images [Rubner *et al.*, 2000], it was used as a transport loss for Wasserstein GAN models [Dukler *et al.*, 2019]. Though three-player WGAN [Dam *et al.*, 2019] was developed to overcome the Lipschitz constraint, their model is not limited to any specific transport loss as long as it is lower semicontinuous and bounded [Villani, 2008].

2.2 Primal Formulation

Wasserstein Auto-Encoder (WAE)

[Tolstikhin *et al.*, 2018] have introduced to use Wasserstein distance to formulate auto-encoder instead of KL divergence used in VAE in Sec 1.1. In the paper, they consider a deterministic conditional distribution $p_\theta(\boldsymbol{x} \mid \boldsymbol{z})$ via a map $G_\theta : \mathcal{Z} \to \mathcal{X}$. It turns out that the Wasserstein distance of order 1 between the empirical data distribution P_X and the generated data distribution P_θ under cost function c can be represented as $W_1(P_X, P_\theta) = \inf_{Q:\, Q_Z = P_Z} \mathop{\mathbb{E}}_{Q_{Z|X},\, P_X}[c(\boldsymbol{x}, G_\theta(\boldsymbol{z}))]$ where $q(\boldsymbol{z}) = \int q(\boldsymbol{z} \mid \boldsymbol{x}) p(\boldsymbol{x})\, dx$ is the marginal distribution of Z. In practical, they relax the constraint of $Q_Z = P_Z$ by using Lagrangian multiplier and define the WAE objective

$$D_{\text{WVAE}} \triangleq \inf_{\omega} \mathop{\mathbb{E}}_{q_\omega(\boldsymbol{z}|\boldsymbol{x}),\, p(\boldsymbol{x})}[c(\boldsymbol{x}, G_\theta(\boldsymbol{z}))] + \lambda D_Z(q(z), p(z)),$$

[3] In [Arjovsky *et al.*, 2017], authors concretely chose the cost function $c(x,y) = \|x - y\|$ which allows weight clipping inducing Lipschitz condition.

[4] Note that since dropout is stochastic,ϕ_{drop} produces two different networks for $x_1$and x_2.

[5] In fact, they used Monge formulation [Villani, 2003, p.4] on latent space, a strengthened version of Wasserstein distance in which the constraint of one-to-one mapping between P_X and P_θ is endowed.)

where $q_\omega(\boldsymbol{z} \mid \boldsymbol{x})$ is an encoder parameterized by ω, and D_Z is an arbitrary divergence between Q_Z and P_Z. Authors suggested using variants of generative models such as GAN [Goodfellow *et al.*, 2014] or the maximum mean discrepancy (MMD) [Gretton *et al.*, 2012; Li *et al.*, 2015] as divergence D_Z. Recent works have been exploring to use difference divergences for D_z, for instance, [Zhang *et al.*, 2019] approximate Q_Z as a Gaussian and use the prior P_Z as a Gaussian distribution which leads a analytical form of Wasserstein distance between two Gaussians for D_z. [Kolouri *et al.*, 2019b] used sliced Wasserstein distance in Eq. (7) to approximate the difference between Q_Z and P_Z. Sinkhorn divergence is also used to characterize distinction in [Patrini *et al.*, 2020].

2.3 Relaxed and Approximated Formulation

Instead of using the original version of Wasserstein (in both dual and primal forms), there is a line of work that dedicated to using variants of sliced Wasserstein distance in Sec 1.2 for learning deep generative models These works used the primal form of sliced Wasserstein distance as signals to update the generators. However, since data lies on high dimensional spaces, the number of projections for a good approximation is huge. [Deshpande *et al.*, 2018; 2019; Kolouri *et al.*, 2019a] have suggested to use a learnable function, which is learned to most discriminate two distributions, to map data before projecting to compute the (max/generalized) sliced Wasserstein distance. In [Wu *et al.*, 2019], authors used sliced Wasserstein distance in both primal and dual form to learn Wasserstein GAN and Wasserstein Autoencoder.

As Sinkhorn divergence in Eq. 6 is computed using the Sinkhorn algorithm which is amenable to automatic differentiation framework, it was used to learn deep generative models [Frogner *et al.*, 2015]. However, when dealing with high-dimensional data, choosing cost function c is a critical task owing to the curse of dimensionality. In [Genevay *et al.*, 2018], authors proposed to learn cost function c via a deep neural network ϕ to map data from original space onto Euclidean space $\mathbb{R}^d$ which has a similar role as critics in WGAN, i.e. $c_\omega(x, y) = (\|\phi_\omega(x) - \phi_\omega(y)\|)$. Typically, embedding function from data space $\mathcal{X}$ to lower dimension is learned with a maximization problem in a similar fashion with MMD models [Li *et al.*, 2017] which leads to the optimization problem $\min_\theta \max_\omega S_\epsilon(P_X, P_\theta)$. One of the benefits of learning deep generative models with Sinkhorn divergences is that its natural gradient can be approximated accurately with low complexity [Shen *et al.*, 2020]. This will help the training process converge faster.

3 Challenges and Discussions

Since optimal transport in machine learning is a fast-developing and promising area, there are potential open questions and challenges in applying optimal transport for deep generative models particularly. In this section we would like to highlight a number of open challenges for future research of using optimal transport for learning deep generative models research in both practical and theoretical aspects.

Cost function. Wasserstein distance has the benefit of taking into account the geometry of data space of distributions which is reflected through the cost function c. Therefore *choosing the appropriate cost function* for each data type is one of the key factors in designing successful learning models. As described in Sec 2, the popular cost functions used in major classes of models are l_p $(p = 1, 2)$ norms which imply that data lie on Euclidean spaces. Some others used norms on Banach spaces or Wasserstein distance. However, *which cost function is suitable* for use with a given data type/dataset is still an open question. Some works do not specify the cost function but design deep neural network to learn to map from data spaces to Euclidean spaces and use norms on them. These methodologies are merely heuristic and do not possess any *theoretical guarantee*.

Approximation formulation comparison. There are different forms to approximate Wasserstein distance however it is not clear which approximation is tighter than the others. For instance, all formulations are approximated using the mini-batch approximation of intractable expectation computation. The approximation in the dual formulation of WGAN also comes from deep neural network approximation of the critic, and soft constraint of 1-Lipschitz condition for the critics. From the experimental perspective, it is imperative to have benchmarks on real-world datasets for comparison between variants of Wasserstein distance. From the theoretical perspective, there are bound between Wasserstein distances and sliced variants [Bonneel *et al.*, 2015] but how good a mini-batch approximation of these formulations in terms of asymptotic and gradient properties is raised as potential challenges.

Better Wasserstein approximation and extensions. Beyond four approximated formulations which have presented in Sec 1.2, the community is actively proposing some other approximation methods which are probably amenable for learning with deep generative models. For instance, recent work on minibatch Wasserstein [Fatras *et al.*, 2020; Nguyen *et al.*, 2021b] with the theoretical bound on the approximation of intractable Wasserstein distance has been proposed. Applying these new approximations for learning deep generative models is a fruitful direction. There are several extensions of optimal transport to more general settings: multiple measures called multi-marginal OT, unnormalized ones called unbalanced OT [Chizat *et al.*, 2018]. Investigations on these extensions are still an open direction with few recently published works [Balaji *et al.*, 2020; Cao *et al.*, 2019].

Beyond Euclidean data generation. Many data in real-world applications have an underlying structure beyond Euclidean data such as social networks, molecular graph (graph data), financial data, sentences (sequence data). Most of the existing works of deep generative models focused and demonstrated on synthetic and image data. Data generation using deep neural networks on graph or sequence data is an open problem with few works published [Yoon *et al.*, 2019; Golany *et al.*, 2020; De Cao and Kipf, 2018; Jin *et al.*, 2018; Liao *et al.*, 2019]. Optimal transport based deep generative models can provide a flexible framework for learning these

complex structured data. If one can define a suitable cost function c for a corresponding data type, models in Sec 2 can be used to generate the desired data with some considerations. For instance, we can use (soft) dynamic time wrapping (DWT) [Staib *et al.*, 2017; Blondel *et al.*, 2020] for real-valued time series, and Gromov-Wasserstein distance in Eq. (9) for graphs. When the dual form models such as WGAN are used, the challenge is to enforce the Lipschitz constraint under the new cost function[6]. Gradient penalty or Lipschitz penalty in WGAN is not valid as they have been defined on l_p norms. If we use primal form models such as WAE, the new cost function can be applied directly. However, the challenge may come from the complexity of computing these metrics which can be a burden when training with a large mini-batch or dataset.

4 Conclusion

Our work provides a systematic review of using optimal transport for deep generative models. We present a taxonomy of deep generative models based on variants of optimal transport distance including primal, dual, relaxed, and approximated formulations, followed by a detailed introduction, comparison, and discussion connections and differences of them. We also discuss the current challenges of the current models in terms of cost function awareness and comparison of different approximations of optimal transport distance. We finally describe visions and directions of data generation with non-Euclidean data.

References

[Adler and Lunz, 2018] Jonas Adler and Sebastian Lunz. Banach wasserstein gan. In *Advances in Neural Information Processing Systems*, pages 6754–6763, 2018.

[Amos *et al.*, 2017] Brandon Amos, Lei Xu, and J Zico Kolter. Input convex neural networks. In *International Conference on Machine Learning*, pages 146–155. PMLR, 2017.

[Anand and Huang, 2018] Namrata Anand and Po-Ssu Huang. Generative modeling for protein structures. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 7505–7516, 2018.

[Arjovsky *et al.*, 2017] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223, 2017.

[Avraham *et al.*, 2019] Gil Avraham, Yan Zuo, and Tom Drummond. Parallel optimal transport gan. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4411–4420, 2019.

[Balaji *et al.*, 2020] Yogesh Balaji, Rama Chellappa, and Soheil Feizi. Robust optimal transport with applications in generative modeling and domain adaptation. In *Advances in Neural Information Processing Systems Foundation (NeurIPS)*, 2020.

[Blondel *et al.*, 2020] Mathieu Blondel, Arthur Mensch, and Jean-Philippe Vert. Differentiable divergences between time series. *arXiv preprint arXiv:2010.08354*, 2020.

[Bonneel *et al.*, 2015] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and radon wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015.

[Cao *et al.*, 2019] Jiezhang Cao, Langyuan Mo, Yifan Zhang, Kui Jia, Chunhua Shen, and Mingkui Tan. Multi-marginal wasserstein gan. In *Advances in Neural Information Processing Systems*, pages 1776–1786, 2019.

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

[Choi *et al.*, 2017] Edward Choi, Siddharth Biswal, Bradley Malin, Jon Duke, Walter F Stewart, and Jimeng Sun. Generating multi-label discrete patient records using generative adversarial networks. In *Machine learning for healthcare conference*, pages 286–305. PMLR, 2017.

[Costa *et al.*, 2017] Pedro Costa, Adrian Galdran, Maria Ines Meyer, Meindert Niemeijer, Michael Abràmoff, Ana Maria Mendonça, and Aurélio Campilho. End-to-end adversarial retinal image synthesis. *IEEE transactions on medical imaging*, 37(3):781–791, 2017.

[Cuturi, 2013] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300, 2013.

[Dam *et al.*, 2019] Nhan Dam, Quan Hoang, Trung Le, Tu Dinh Nguyen, Hung Bui, and Dinh Phung. Threeplayer wasserstein gan via amortised duality. In *Proceedings of the 28th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2019.

[De Cao and Kipf, 2018] Nicola De Cao and Thomas Kipf. Molgan: An implicit generative model for small molecular graphs. *arXiv preprint arXiv:1805.11973*, 2018.

[Deshpande *et al.*, 2018] Ishan Deshpande, Ziyu Zhang, and Alexander G Schwing. Generative modeling using the sliced wasserstein distance. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3483–3491, 2018.

[Deshpande *et al.*, 2019] Ishan Deshpande, Yuan-Ting Hu, Ruoyu Sun, Ayis Pyrros, Nasir Siddiqui, Sanmi Koyejo, Zhizhen Zhao, David Forsyth, and Alexander G Schwing. Max-sliced wasserstein distance and its use for gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10648–10656, 2019.

[Dukler *et al.*, 2019] Yonatan Dukler, Wuchen Li, Alex Lin, and Guido Montúfar. Wasserstein of wasserstein loss for

[6]Recall that the definition of Lipschitz condition depends on the cost function.

learning generative models. In *International Conference on Machine Learning*, pages 1716–1725. PMLR, 2019.

[Fatras *et al.*, 2020] Kilian Fatras, Younes Zine, Rémi Flamary, Rémi Gribonval, and Nicolas Courty. Learning with minibatch wasserstein: asymptotic and gradient properties. In *the 23nd International Conference on Artificial Intelligence and Statistics*, volume 108, 2020.

[Frid-Adar *et al.*, 2018] Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *Neurocomputing*, 321:321–331, 2018.

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

[Genevay *et al.*, 2018] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pages 1608–1617. PMLR, 2018.

[Golany *et al.*, 2020] Tomer Golany, Kira Radinsky, and Daniel Freedman. Simgans: Simulator-based generative adversarial networks for ecg synthesis to improve deep ecg classification. In *International Conference on Machine Learning*, pages 3597–3606. PMLR, 2020.

[Goodfellow *et al.*, 2014] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[Gretton *et al.*, 2012] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012.

[Gulrajani *et al.*, 2017] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5769–5779, 2017.

[Isola *et al.*, 2017] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[Jin *et al.*, 2018] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International Conference on Machine Learning*, pages 2323–2332. PMLR, 2018.

[Kim *et al.*, 2020] Seung Wook Kim, Yuhao Zhou, Jonah Philion, Antonio Torralba, and Sanja Fidler. Learning to simulate dynamic environments with gamegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1231–1240, 2020.

[Kolouri *et al.*, 2019a] Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and K Gustavo. Generalized sliced wasserstein distances. In *NeurIPS 2019*, 2019.

[Kolouri *et al.*, 2019b] Soheil Kolouri, Phillip E Pope, Charles E Martin, and Gustavo K Rohde. Sliced wasserstein auto-encoders. In *International Conference on Learning Representations*, 2019.

[Korotin *et al.*, 2021] Alexander Korotin, Vage Egiazarian, Arip Asadulaev, Alexander Safin, and Evgeny Burnaev. Wasserstein-2 generative networks. 2021.

[Kumar *et al.*, 2018] A Kumar, A Biswas, and S Sanyal. Ecommercegan: A generative adversarial network for e-commerce. In *6th International Conference on Learning Representations, ICLR 2018-Workshop Track Proceedings*. International Conference on Learning Representations, ICLR, 2018.

[Ledig *et al.*, 2017] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[Li *et al.*, 2015] Yujia Li, Kevin Swersky, and Rich Zemel. Generative moment matching networks. In *International Conference on Machine Learning*, pages 1718–1727, 2015.

[Li *et al.*, 2017] Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. Mmd gan: Towards deeper understanding of moment matching network. In *Advances in Neural Information Processing Systems*, pages 2200–2210, 2017.

[Liao *et al.*, 2019] Renjie Liao, Yujia Li, Yang Song, Shenlong Wang, Charlie Nash, William L. Hamilton, David Duvenaud, Raquel Urtasun, and Richard Zemel. Efficient graph generation with graph recurrent attention networks. In *NeurIPS*, 2019.

[Liu *et al.*, 2018] Huidong Liu, GU Xianfeng, and Dimitris Samaras. A two-step computation of the exact gan wasserstein distance. In *International Conference on Machine Learning*, pages 3159–3168. PMLR, 2018.

[Liu *et al.*, 2019] Huidong Liu, Xianfeng Gu, and Dimitris Samaras. Wasserstein gan with quadratic transport cost. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4832–4841, 2019.

[Marouf *et al.*, 2020] Mohamed Marouf, Pierre Machart, Vikas Bansal, Christoph Kilian, Daniel S Magruder, Christian F Krebs, and Stefan Bonn. Realistic in silico generation and augmentation of single-cell rna-seq data using generative adversarial networks. *Nature communications*, 11(1):1–12, 2020.

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

[Miyato *et al.*, 2018] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

[Monge, 1781] Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences de Paris*, 1781.

[Nguyen *et al.*, 2021a] Khai Nguyen, Nhat Ho, Tung Pham, and Hung Bui. Distributional sliced-wasserstein and applications to generative modeling. In *International Conference on Learning Representations*, 2021.

[Nguyen *et al.*, 2021b] Khai Nguyen, Quoc Nguyen, Nhat Ho, Tung Pham, Hung Bui, Dinh Phung, and Trung Le. Bomb-ot: On batch of mini-batches optimal transport. *arXiv preprint arXiv:2102.05912*, 2021.

[Pathak *et al.*, 2016] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.

[Patrini *et al.*, 2020] Giorgio Patrini, Rianne van den Berg, Patrick Forre, Marcello Carioni, Samarth Bhargav, Max Welling, Tim Genewein, and Frank Nielsen. Sinkhorn autoencoders. In *Uncertainty in Artificial Intelligence*, pages 733–743. PMLR, 2020.

[Petzka *et al.*, 2018] Henning Petzka, Asja Fischer, and Denis Lukovnikov. On the regularization of wasserstein GANs. In *International Conference on Learning Representations*, 2018.

[Ramdas *et al.*, 2017] Aaditya Ramdas, Nicolás García Trillos, and Marco Cuturi. On wasserstein two-sample testing and related families of nonparametric tests. *Entropy*, 19(2):47, 2017.

[Rubner *et al.*, 2000] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000.

[Shen *et al.*, 2020] Zebang Shen, Zhenfu Wang, Alejandro Ribeiro, and Hamed Hassani. Sinkhorn natural gradient for generative models. In *Thirty-fourth Conference on Neural Information Processing Systems*, 2020.

[Staib *et al.*, 2017] Matthew Staib, Sebastian Claici, Justin Solomon, and Stefanie Jegelka. Parallel streaming Wasserstein barycenters. In *Advances in Neural Information Processing Systems 31*, 2017.

[Takahashi *et al.*, 2019] Shuntaro Takahashi, Yu Chen, and Kumiko Tanaka-Ishii. Modeling financial time-series with generative adversarial networks. *Physica A: Statistical Mechanics and its Applications*, 527:121261, 2019.

[Tolstikhin *et al.*, 2018] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. In *International Conference on Learning Representations*, 2018.

[Vasu *et al.*, 2018] Subeesh Vasu, Nimisha Thekke Madam, and AN Rajagopalan. Analyzing perception-distortion tradeoff using enhanced perceptual super-resolution network. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018.

[Villani, 2003] Cédric Villani. *Topics in optimal transportation*. Number 58. American Mathematical Soc., 2003.

[Villani, 2008] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

[Wei *et al.*, 2018] Xiang Wei, Zixia Liu, Liqiang Wang, and Boqing Gong. Improving the improved training of wasserstein GANs. In *International Conference on Learning Representations*, 2018.

[Wu *et al.*, 2019] Jiqing Wu, Zhiwu Huang, Dinesh Acharya, Wen Li, Janine Thoma, Danda Pani Paudel, and Luc Van Gool. Sliced wasserstein generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3713–3722, 2019.

[Yoon *et al.*, 2019] Jinsung Yoon, Daniel Jarrett, and Mihaela van der Schaar. Time-series generative adversarial networks. In *Advances in Neural Information Processing Systems*, 2019.

[Zhang *et al.*, 2019] Shunkang Zhang, Yuan Gao, Yuling Jiao, Jin Liu, Yang Wang, and Can Yang. Wasserstein-wasserstein auto-encoders. *arXiv preprint arXiv:1902.09323*, 2019.