Coupled Variational Autoencoder

Xiaoran Hao [1] **Patrick Shafto** [1 2]

Abstract

Variational auto-encoders are powerful probabilistic models in generative tasks but suffer from generating low-quality samples which are caused by the holes in the prior. We propose the Coupled Variational Auto-Encoder (C-VAE), which formulates the VAE problem as one of Optimal Transport (OT) between the prior and data distributions. The C-VAE allows greater flexibility in priors and natural resolution of the prior hole problem by enforcing coupling between the prior and the data distribution and enables flexible optimization through the primal, dual, and semi-dual formulations of entropic OT. Simulations on synthetic and real data show that the C-VAE outperforms alternatives including VAE, WAE, and InfoVAE in fidelity to the data, quality of the latent representation, and in quality of generated samples.

1. Introduction

The combination of variational Bayesian inference and deep latent variable models resulted in one of the most powerful generative models, Variational autoencoders (VAEs) (Kingma & Welling, 2014; Rezende et al., 2014). Scaleable inference and simple, flexible latent representations have enabled VAEs to make substantial progress in fields including image and video generation (Razavi et al., 2019; Yan et al., 2021), audio and music synthesis (Dhariwal et al., 2020; Kim et al., 2021), molecular processes (Lim et al., 2018), semi-supervised learning (Kingma et al., 2014; Izmailov et al., 2020), and unsupervised representation learning (Fortuin et al., 2018; van den Oord et al., 2017).

VAEs assume a latent prior over a generic space, and a variational approximate posterior, which in principle allow flexibility in modeling domains. However, for simplicity, an isotropic multivariate Gaussian is often employed for both prior distribution and variational posterior. A too simplistic prior could lead to over-regularization and a too simplistic family of variational posterior distributions leaves a big gap between true posterior and variational posterior, which limits performance on complex data (Burda et al., 2016; Kingma et al., 2016; Dai & Wipf, 2019). Although people have proposed many methods to solve these problems. i.e enriching the variational family (Rezende & Mohamed, 2015; Kingma et al., 2016), using a more flexible (Tomczak & Welling, 2018; Takahashi et al., 2019; Casale et al., 2018) or non gaussian prior (Davidson et al., 2018; Joo et al., 2019), these approaches remain restricted to parametric distributions. A second limitation is the "prior hole problem" which refers to the mismatch between prior and aggregate posterior that reduces the quality of ancestral samples (Rezende & Viola, 2018; Aneja et al., 2021). The problem arises due to the variational approximation and density estimation, which when imperfect allows "holes"—areas where the aggregate posterior has low density compared to the prior—where the decoder has not been trained but has a high probability of being generated under the prior.

Optimal transport (OT) (Villani, 2008; Kantorovich, 1942; Cuturi, 2013) provides a cogent solution to both the restriction to parametric posterior distributions and the mismatch between the prior and aggregate posterior distributions. Kantorovich OT computes optimal couplings between marginal distributions, and entropic OT allows efficient computation. Computational methods for solving EOT problems do not require parametric distributions, which in principle allows greater flexibility. Moreover, coupling the marginals—here the prior and the empirical distribution of the data—can resolve the prior hole problem.

We propose the Coupled VAE (C-VAE), which generalizes previous VAEs and addresses their limitations. We derive an EOT-based algorithm for training the decoder and encoder. We use the dual and semi-dual OT formulation solving for the approximate posterior in the continuous prior case and the Sinkhorn algorithm in the discrete prior case. We can therefore work with an enriched family of approximate posterior distributions, and any distribution as prior without extra cost. The prior hole problem is resolved naturally via marginal constraints of EOT. We illustrate the flexibility in both prior and posterior, and the resolution of the prior hole

[1]Department of Mathematics and Computer Science, Rutgers University-Newark, New Jersey, USA [2]School of Mathematics, Institute for Advanced Study, New Jersey, USA. Correspondence to: Xiaoran Hao <xh197@rutgers.edu>.

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

problem through detailed simulations.

Notation. For a metric space $\mathcal{X}$, $\mathcal{C}(\mathcal{X})$ denotes the space of all continuous functions on $\mathcal{X}$ and $\mathcal{M}_+^1(\mathcal{X})$ denotes the set of positive Radon probability measures (i.e. of unit mass) on $\mathcal{X}$. Upper cases X denote random variables that take values on $\mathcal{X}$. $X \sim \alpha$ says that a random variable X follows a distribution $\alpha \in \mathcal{M}_+^1(\mathcal{X})$. Capital letters P_X denote probability distribution and $p(x)$ to represent the probability density function. When there is no ambiguity, we use the same notation for distributions and their densities.

2. Background

We start by reviewing VAEs and the generalization to InfoVAE. Then we will introduce Entropy-regularized Optimal transport and the connection between VAEs and EOT.

2.1. Variatonal autoencoders and InfoVAE

Consider two compact metric spaces $\mathcal{X}$ and $\mathcal{Z}$. Latent variable generative models are usually defined in the form of a parametric model of joint distribution that admits density function $p_\theta(x,z) = p_\theta(x|z)p(z)$ over observable variable $x \in \mathcal{X}$ and latent variable $z \in \mathcal{Z}$. $p(z)$ is typically a simple prior distribution such as uniform or Gaussian. $p_\theta(x|z)$ is the conditional distribution parametrized by neural networks with parameters θ. Marginalizing out the latent variable z results in the model distribution $p_\theta(x) = \int p_\theta(x,z)dz = \int p_\theta(x|z)p(z)dz$. The goal of generative modeling is to maximize marginal likelihood:

$$\mathbb{E}_{p_D(x)}[\log p_\theta(x)] = \mathbb{E}_{p_D(x)}[\log \mathbb{E}_{p(z)}[p_\theta(x|z)]], \quad (1)$$

Where $p_D(x)$ represents data distribution. However, because of the parameterization of $p_\theta(x|z)$, the log marginal likelihood is intractable in general. VAEs maximize a surrogate, the evidence lower bound (ELBO) instead. We denote it by $\mathcal{L}_{\mathbf{VAE}}$:

$$\mathbb{E}_{p_D(x)}[\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] - D_{KL}(q_\phi(z|x)||p(z))], \quad (2)$$

where $q_\phi(z|x)$ is the variational posterior that is modeled by deep neural networks with parameter ϕ. In the context of Auto-encoders, $p_\theta(x|z)$ and $q_\phi(z|x)$ are called decoder and encoder respectively. The first term of equation 2 is called the reconstruction term and the second term is called the regularization term. The typical choice of encoder distribution is Gaussian with a diagonal covariance matrix, i.e., $q_\phi(z|x) = \mathcal{N}(z|\mu_\phi(x), \mathrm{diag}(\sigma_\phi^2(x)))$. The decoder distribution depends on the context, e.g. Bernoulli distribution if x is binary and the Gaussian distribution if x is continuous.

The parameters of the neural networks are obtained by optimizing the ELBO. For continuous latent variables, this could be done efficiently through the re-parameterization trick (Kingma & Welling, 2014).

InfoVAE (Zhao et al., 2019) generalizes the VAE family of models by introducing a scaling parameter to the KL divergence term and including a mutual information term that encourages high mutual information between x and z. Formally, up to an additive constant, we can derive an equivalent expression of ELBO:

$$\begin{aligned}\mathcal{L}_{\mathbf{VAE}} = &-D_{KL}(q_\phi(z)||p(z))] \\ &- \mathbb{E}_{q_\phi(z)}[D_{KL}(q_\phi(x|z)||p_\theta(x|z)] + \text{const.}\end{aligned} \quad (3)$$

where constant is given by $\mathbb{E}_{p_D(x)}[\log p_D(x)]$, $q_\phi(z) = \int q_\phi(z|x)p_D(x)dx$ and $q_\phi(x|z) = \frac{q_\phi(z|x)p_D(x)}{q_\phi(z)}$. Then the loss of InfoVAE is derived as follows:

$$\begin{aligned}\mathcal{L}_{\mathbf{InfoVAE}} = &-\lambda D_{KL}(q_\phi(z)||p(z)) \\ &- \mathbb{E}_{q_\phi(z)}[D_{KL}(q_\phi(x|z)||p_\theta(x|z))] \\ &+ \alpha I_q(x;z) \\ =&\mathbb{E}_{p_D(x)}\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] \\ &- (1-\alpha)\mathbb{E}_{p_D(x)}[D_{KL}(q_\phi(z|x)||p(z))] \\ &- (\alpha+\lambda-1)D_{KL}(q_\phi(z)||p(z)).\end{aligned} \quad (4)$$

When $\alpha = 0$ and $\lambda = 1$, InfoVAE recovers simple VAE. When $\alpha+\lambda-1 = 0$, we have β-VAE (Higgins et al., 2017). If $\alpha = \lambda = 1$ and KL divergence is replaced with Jensen Shannon divergence, and the model becomes adversarial autoencoder (AAE) (Makhzani et al., 2016).

The final optimization problem of InfoVAE is,

$$\min_{p_\theta(x|z)} \min_{q_\phi(z|x)} -\mathcal{L}_{\mathbf{InfoVAE}}. \quad (5)$$

Typically, optimizations over $p_\theta(x|z)$ and $q_\phi(z|x)$ are done jointly and over parameters θ and ϕ. We introduce separated minimizations here in order to make connections in the following sections.

Prior hole problem. The problem refers to the situation when $q_\phi(z) \neq p(z)$ and there exist regions that have high density under $p(z)$ but low, possibly zero, density under $q_\phi(z)$. This does harm to the ancestral sampling process of the VAEs model as the samples drawn from the prior may not be decoded closely to the samples from the data distribution. In other words, the holes in the prior will generate low-quality samples. InfoVAE mitigates but does not solve, this problem to some degree as it has an explicit penalty term with a pre-defined weight parameter.

2.2. Entropy-regularized optimal transport

For two continuous probability measures $\mu \in \mathcal{M}_+^1(\mathcal{X})$ and $\nu \in \mathcal{M}_+^1(\mathcal{Y})$, given a measurable function $c(x,y) : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ which represents the ground cost of moving a unit of mass from x to y, Kantorovich OT (Kantorovich,

1942) seeks the optimal transport plan $\pi(x, y)$ subject to marginals μ, ν with minimum transport loss. Formally, the Kantorovich OT is,

$$OT(\mu, \nu) \stackrel{\text{def}}{=} \min_{\pi \in \mathcal{U}(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x, y) d\pi(x, y). \quad (6)$$

where the feasible set $\mathcal{U}(\mu, \nu)$ consists of all probability measures defined over the product space $\mathcal{X} \times \mathcal{Y}$ with marginal measures μ and ν respectively.

Entropy regularized Optimal Transport (EOT) (Cuturi, 2013; Genevay et al., 2016) includes an entropy regularization term into the original Kantorovich OT objective:

$$OT_\varepsilon(\mu, \nu) \stackrel{\text{def}}{=} \min_{\pi \in \mathcal{U}(\mu,\nu)} \int_{\mathcal{X}\times\mathcal{Y}} c(x, y) d\pi(x, y) + \varepsilon D_{\mathrm{KL}}(\pi \| \mu \otimes \nu). \quad (7)$$

where ϵ is the regularization weight and relative entropy $D_{\mathrm{KL}}(\pi\|\mu \otimes \nu)$ is defined as:

$$D_{\mathrm{KL}}(\pi\|\mu \otimes \nu) \stackrel{\text{def}}{=} \int_{\mathcal{X}\times\mathcal{Y}} \log\left(\frac{d\pi(x, y)}{d\mu(x)d\nu(y)}\right) d\pi(x, y). \quad (8)$$

By Fenchel-Rockafellar duality, the Kantorovich problem with entropy regularization admits dual formulation, which can be expressed as the maximization of an expectation:

$$\begin{aligned} OT_\varepsilon(\mu, \nu) &= \max_{\substack{u\in\mathcal{C}(\mathcal{X}) \\ v\in\mathcal{C}(\mathcal{Y})}} \int_{\mathcal{X}} u(x) d\mu(x) + \int_{\mathcal{Y}} v(z) d\nu(y) \\ &\quad - \varepsilon \int_{\mathcal{X}\times\mathcal{Y}} e^{\frac{u(x)+v(y)-c(x,y)}{\varepsilon}} d\mu(x)d\nu(y) \quad (9) \\ &= \max_{u\in\mathcal{C}(\mathcal{X})} \int_{\mathcal{X}} u(x) d\mu(x) + \int_{\mathcal{Y}} u^{c,\varepsilon}(y) d\nu(y) \\ &= \max_{u\in\mathcal{C}(\mathcal{X})} \mathbb{E}_{Y\sim\nu}[\int_{\mathcal{X}} u(x) d\mu(x) + u^{c,\varepsilon}(Y)], \quad (10) \end{aligned}$$

where $u^{c,\varepsilon}(y)$ is the c, ε -transform:

$$u^{c,\varepsilon}(y) \stackrel{\text{def}}{=} -\varepsilon \log \int_{\mathcal{X}} e^{\frac{u(x)-c(x,y)}{\varepsilon}} d\mu(x). \quad (11)$$

Sinkhorn's algorithm solves discrete EOT with linear convergence (Cuturi, 2013). However, EOT does not scale well to measures supported on a large number of points and it also assumes discrete measures. A fundamental property of the dual problem (Equation 10) is stochastic gradient methods are applicable as long as we can sample from the marginal distributions (Genevay et al., 2016; Seguy et al., 2018). Dual variables can be parameterized by neural networks in continuous settings or left as a finite vector in discrete cases. This provides a method to apply optimal transport to large-scale machine learning tasks such as generative modeling.

The relationship between primal and dual problems allows switching between formulations. After solving for the dual variables, we can recover a feasible solution to the primal problem by the first-order optimality condition,

$$d\pi(x, y) = \exp\left(\frac{u(x) + v(y) - c(x, y)}{\varepsilon} - 1\right) d\mu(x)d\nu(y). \quad (12)$$

3. C-VAE formulation

C-VAE generalizes InfoVAE via EOT, which will address the prior hole problem and relax the Gaussian assumption of the approximated posterior distribution.

Framework. Assume all probability measures are absolutely continuous with respect to the Lebesgue measure or counting measure. i.e., $d\pi(x, y) = \pi(x, y)dxdy$. Consider data space $\mathcal{X}$ and latent space $\mathcal{Z}$ as the underlying metric space for EOT. Data distribution P_D and the prior distribution of latent variable P_Z from VAEs serve as marginals for the joint distribution. Let cost function $c(x, z)$ to be negative log likelihood, i.e., $c(x, z) = -\log p_\theta(x|z)$. Plugging in everything above into equation 7, we get:

$$OT_\varepsilon(P_D, P_Z) = \min_{\pi(x,z)} \int_{\mathcal{X}\times\mathcal{Z}} -\log p_\theta(x|z)\pi(x, z)dxdz + \varepsilon D_{\mathrm{KL}}(\pi(x, z)\|p(z)p_D(x)). \quad (13)$$

Decompose the joint distribution into marginal and conditional distribution, i.e., $\pi(x, z) = q(z|x)p_D(x)$. OT requires the joint distribution to satisfy the other marginal distribution also, i.e., $p(z) = q(z) \stackrel{\text{def}}{=} \int q(z|x)p_D(x)dx$. Relaxing this hard constraint with a KL divergence and minimizing the transport loss with respect to the cost function, we get:

$$\begin{aligned} OT_\varepsilon(P_D, P_Z) = \min_{\pi(x,z)} \int_{\mathcal{X}\times\mathcal{Z}} & -\log p_\theta(x|z)\pi(x, z)dxdz \\ & + \varepsilon D_{\mathrm{KL}}(\pi(x, z)\|p(z)p_D(x)) \\ & + \varsigma D_{\mathrm{KL}}(q(z)\|p(z)) \\ = \min_{q(z|x)} \mathbb{E}_{p_D(x)}\mathbb{E}_{q(z|x)}[&-\log p_\theta(x|z)] \\ & + \varepsilon \mathbb{E}_{p_D(x)}[D_{KL}(q(z|x)\|p(z))] \\ & + \varsigma D_{\mathrm{KL}}(q(z)\|p(z)), \quad (14) \end{aligned}$$

where $\varsigma > 0$ is the weight for the penalty term. Let $\varepsilon = 1-\alpha$ and $\varsigma = \alpha + \lambda - 1$, equation 14 coincides with equation 4

except the optimization is only about the posterior $q(z|x)$. With $p_\theta(x|z)$ as a neural network parametrized decoder, optimizing θ by SGD, we arrive at the final objective:

$$\min_{p_\theta(x|z)} \min_{q(z|x)} \mathbb{E}_{p_D(x)}\mathbb{E}_{q(z|x)}[-\log p_\theta(x|z)] + \varepsilon\mathbb{E}_{p_D(x)}[D_{KL}(q(z|x)\|p(z))] + \varsigma D_{\mathrm{KL}}(q(z)\|p(z)). \tag{15}$$

Noting that $p_\theta(x|z)$ is the likelihood function in VAEs, which has a natural explanation as decoding cost such that if x can be successfully decoded from z, then the cost is small. Otherwise, the cost is large. Cost function optimization is problematic in the usual OT setting because the cost function can be arbitrarily small for any pair of x and z. However, this is not the case for our model as the likelihood is assumed to be Gaussian or Bernoulli.

Proposition 3.1. *Let $p_\theta(x|z)$ be selected from the family of all parametric Gaussian distributions, i.e., $p_\theta(x|z) = \mathcal{N}(x|\mu_\theta(z), \mathrm{diag}(\sigma^2_\theta(z)))$. Then for any $z \in \mathcal{Z}$, $c(\cdot, z) = -\log p_\theta(\cdot|z)$ is a positive quadratic function with minimum $\sum_{i=1}^{d_x} \log\sqrt{2\pi\sigma^2_{\theta,i}(z)}$ achieved at $x = \mu_\theta(z)$.*

The proof is a direct calculation of the negative log density function of Gaussians. It shows that given any z, there can be only one minimum at the mean of the distribution. If we want to make all costs equally small, then the minimum $\sum_{i=1}^{d_x} \log\sqrt{2\pi\sigma^2_{\theta,i}(z)}$ will go up as $\sigma^2_{\theta,i}(z)$ will increase. Similar results can be derived for Bernoulli likelihood.

EOT acts as variational inference. In VAEs, amortized variational inference requires approximated posteriors $q_\phi(z|x)$ to be multivariate gaussian distributions which are modeled by neural networks with parameters ϕ. Furthermore, the covariance matrices of Gaussians are often assumed to be diagonal. Normalizing flow enriches the distributions via composing with invertible transformations. But it is still within a restricted family of distributions. This limits the expressiveness of the variational distributions and hurts the accuracy of the inference. In the OT setting, we can solve for the approximated posterior $q(z|x)$ or more generally joint distribution $\pi(z, x)$ using optimal transport solvers such as the Sinkhorn algorithm in the discrete setting, stochastic gradients method of Dual or Semi-dual in the continuous setting, and unbalanced Sinkhorn for relaxed marginals problem without assuming the form of approximate distributions.

EOT explanations of VAEs. One of the well-known drawbacks of VAEs models is the blurry samples. People originally attributed this issue on the maximum likelihood objective that penalizes differently when $p_\theta(x) > p_D(x)$ and when $p_\theta(x) < p_D(x)$. Zhao et al. (2017) argued that blurriness is not merely because of the objective but the VAE approximation of the maximum likelihood objective, whereas Cai et al. (2017) argued the L_2 distance used in the objective caused the fuzziness in samples. We propose to explain the fuzziness in samples through EOT. The encoder's and decoder's parameter learning in VAEs correspond to the coupling and cost function learning in EOT. The relative entropy term forces the coupling to spread over all possible locations, resulting in a plan $\pi(x, z)$ that has a non-zero density between different x_i's and the same z. Then if we try to optimize the cost function to further reduce the transport cost, based on the proposition we state above, we know that the decoder will send z to the weighted average of all x_i's.

Another common problem of VAEs is the posterior collapse phenomenon, which refers to the situation when $q_\phi(z|x) = p(z)$ for any x. One commonly accepted reason is that the decoder is too flexible. It could ignore the latent representations but achieve a good enough likelihood estimation, i.e., $p_\theta(x|z) = p_D(x)$. This is also easy to explain in EOT as the cost function is no longer a function of z. The optimal plan only depends on the relative entropy term which will achieve minimum when the plan is an independent coupling.

Matching the aggregate posterior with prior. In the following sections, we will assume $\varsigma = +\infty$ which forces aggregated posterior to equal prior distribution (hard constraint). We claim this is not an unreasonable choice because maximizing VAE objective with respect to the prior distribution is equivalent to matching the prior with aggregate posterior, and if the model has learned the data distribution, i.e. $p_\theta(x) = p_D(x)$ and approximated posterior capture the true posterior, i.e., $q_\phi(z|x) = p(z|x)$, then $q_\phi(z) = \int q_\phi(z|x)p_D(x)dx = \int p(z|x)p_\theta(x)dx$ should equal to $p(z)$.

Unlike the primal OT problem which is a constrained optimization problem that is known to be difficult to solve, the Dual and Semi-Dual formulation is unconstrained where SGD methods apply naturally:

$$\min_\theta \max_{\substack{u\in\mathcal{C}(\mathcal{X})\\ v\in\mathcal{C}(\mathcal{Z})}} \mathbb{E}_{p(z)\otimes p_D(x)}\left[u(x) + v(z) - \varepsilon e^{\frac{u(x)+v(z)-c_\theta(x,z)}{\varepsilon}}\right] \tag{16}$$

and

$$\min_\theta \max_{u\in\mathcal{C}(\mathcal{X})} \mathbb{E}_{p(z)}\left[\int_{\mathcal{X}} u(x)p_D(x)dx + u^{c_\theta,\varepsilon}(z)\right]. \tag{17}$$

We keep $c_\theta(x, z) = -\log p_\theta(x|z)$ for readability. Since $p_D(x) \approx \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$ where δ_x is Dirac-delta function, inner optimization is actually a finite-dimensional concave maximization problem:

$$\min_\theta \max_{\mathbf{u}\in\mathbb{R}^n} \mathcal{L}_{\mathbf{C-VAE}}, \tag{18}$$

where

$$\mathcal{L}_{\mathbf{C-VAE}} = \mathbb{E}_{p(z)}\left[\sum_{i=1}^{n}\frac{1}{n}\mathbf{u}_i - \varepsilon\log\sum_{i=1}^{n}\frac{1}{n}e^{\frac{\mathbf{u}_i - c_\theta(x_i,z)}{\varepsilon}}\right].$$

With the primal-dual relationship, we can recover the optimal plan:

$$\pi(x_i,z) = \exp\left(\frac{\mathbf{u}_i + u^{c_\theta,\varepsilon}(z) - c_\theta(x_i,z)}{\varepsilon} - 1\right)p(z)p_D(x_i). \tag{19}$$

Substitute π with $q(z|x)p_D(x_i)$, we can get the expression of posterior :

$$q(z|x_i) = \exp\left(\frac{\mathbf{u}_i + u^{c_\theta,\varepsilon}(z) - c_\theta(x_i,z)}{\varepsilon} - 1\right)p(z). \tag{20}$$

Flexible choice of prior distribution. Our model can work with any prior distribution from which we can sample. There are no differences in training between models with different priors. The data distribution P_D is always discrete as it is approximated by Dirac measures of samples in the training set empirically. If we have a discrete prior over the latent variables, i.e., categorical distribution $P_Z = Cat(K;\mathbf{p})$, we will arrive at a discrete EOT problem that can be solved by Sinkhorn algorithm efficiently.

We will arrive at semi-discrete EOT problems if we have continuous marginal over latent variables like $P_Z = \mathcal{N}(\mathbf{z};\mathbf{0},\mathbf{I})$ or $P_Z = Dir(\mathbf{z};\mathbf{K},\alpha)$. Next, we will come up with two different strategies for learning the latent generative models. The main difference between these two strategies is optimizing the primal or dual formulation of EOT with respect to the likelihood function.

3.1. Optimizing $q(z|x)$ with dual but $p_\theta(x|z)$ with primal

Building on the foundation from the above section, we now define the first training strategy that is established on the primal formulation of EOT. We call this Primal Strategy (see Algorithm 1):

1. Solve for the optimal $q(z|x)$ in dual EOT with a fixed log likelihood $\log p_\theta(x|z)$ as cost function.
2. Optimize the primal EOT objective with respect to likelihood function $p_\theta(x|z)$ using posterior $q(z|x)$ got from the first step.
3. Repeat these two steps until convergence or desired number of steps.

Step one is to find the optimal plan or optimal conditional plan given the marginals. In this step, we will use dual or semi-dual formulation depending on the context. By primal-dual relationship, we can get $q(z|x)$ from dual variables. Then, we are solving for the likelihood function which needs to evaluate the expectation with respect to $q(z|x)$. This is particularly difficult as the integral does not have a closed form. Monte Carlo estimate is also hard because samples from a nontrivial $q(z|x)$ are difficult. Fortunately, the expression of $q(z|x)$ derived above could be seen as a weighted prior which is exactly what we need in importance sampling. The computationally expensive parts of the weight can be cached during the optimization of the optimal plan. Calculating importance weight will not cost much to generate samples from the posterior.

Algorithm 1 Primal Strategy

Require: Input distributions P_Z, P_D; cost function or decoder network $p_\theta(x|z)$; sample size n; learning rate λ_1, λ_2; number of iterations K;
Semi-Dual: Dual variable $\mathbf{u}$ or u_ϕ if parametrized by neural network;
Dual: Dual variables $\mathbf{u}$, $\mathbf{v}$ or u_ϕ, v_ψ if parametrized by neural network;
Ensure: θ, ϕ (and ψ if dual formulation)
repeat
  for $k = 1, 2, ...K$ **do**
    Sample $(z_j)_{j=1}^n \sim P_Z$;
    $\mathcal{L}_\mathbf{u} \leftarrow \frac{1}{n}\sum_{j=1}^{n}[\sum_{i=1}^{m}\frac{1}{m}\mathbf{u}_i - \varepsilon\log\sum_{i=1}^{m}\frac{1}{m}\exp\left(\frac{\mathbf{u}_i - c_\theta(x_i,z_j)}{\varepsilon}\right)]$
    Update $\mathbf{u}$ using $\frac{\partial\mathcal{L}_\mathbf{u}}{\partial\mathbf{u}}$ and λ_1 to maximize $\mathcal{L}_\mathbf{u}$;
  end for
  Sample $(z_j)_{j=1}^n \sim Q_{Z|X}$;
  $\mathcal{L}_\theta \leftarrow \frac{1}{m}\sum_{i=1}^{m}[\frac{1}{n}\sum_{j=1}^{n} - \log p_\theta(x_i|z_j)]$
  Update θ using $\frac{\partial\mathcal{L}_\theta}{\partial\theta}$ and λ_2 to minimize $\mathcal{L}_\theta$;
until converged or reach the max number of epochs

3.2. Optimizing $q(z|x)$ with dual and $p_\theta(x|z)$ with dual

An alternative training strategy is to calculate $q(z|x)$ by the primal-dual relationship, it is possible to optimize dual objective with $p_\theta(x|z)$ directly (see Algorithm 2). The relationship between optimal coupling and dual variables only holds at optimum. When we update only for a fixed number of steps instead of convergence, the coupling can be very biased. Direct optimizing dual can improve this problem. Based on Genevay et al. (2016), we could derive the gradients with respect to dual variable and cost function analytically.

Proposition 3.2. *When $\epsilon > 0$, the gradient of $\mathcal{L}_{\mathbf{C-VAE}}$ with respect to c_θ is given by*

$$\left(\frac{\partial\mathcal{L}_{\mathbf{C-VAE}}}{\partial c_\theta}\right)_j = \mathbb{E}_{p(z)}\left[\frac{\exp\left(\frac{\mathbf{u}_j - c_\theta(x_j,z)}{\varepsilon}\right)]}{\sum_{i=1}^{n}\exp\left(\frac{\mathbf{u}_i - c_\theta(x_i,z)}{\varepsilon}\right)]}\right]$$

Proof is trivial. For details we refer to Genevay et al. (2016)

Algorithm 2 Dual Strategy

Require: Input distributions P_Z, P_D; cost function or decoder network$p_\theta(x|z)$; sample size n; learning rate λ_1, λ_2; number of iterations K;
Semi-Dual: Dual variable $\mathbf{u}$ or u_ϕ if parametrized by neural network;
Dual: Dual variables $\mathbf{u}$, $\mathbf{v}$ or u_ϕ, v_ψ if parametrized by neural network;
Ensure: θ, ϕ (and ψ if dual formulation)
repeat
for $k = 1, 2, ...K$ **do**
Sample $(z_j)_{j=1}^n \sim P_Z$;
$\mathcal{L}_\mathbf{u} \leftarrow \frac{1}{n}\sum_{j=1}^n[\sum_{i=1}^m \frac{1}{m}\mathbf{u}_i - \varepsilon \log \sum_{i=1}^m \frac{1}{m} \exp\left(\frac{\mathbf{u}_i - c_\theta(x_i, z_j)}{\varepsilon}\right)]$
Update $\mathbf{u}$ using $\frac{\partial \mathcal{L}_\mathbf{u}}{\partial \mathbf{u}}$ and λ_1 to maximize $\mathcal{L}_\mathbf{u}$;
end for
Sample $(z_j)_{j=1}^n \sim P_Z$;
$\mathcal{L}_\theta \leftarrow \frac{1}{n}\sum_{j=1}^n[\sum_{i=1}^m \frac{1}{m}\mathbf{u}_i - \varepsilon \log \sum_{i=1}^m \frac{1}{m} \exp\left(\frac{\mathbf{u}_i - c_\theta(x_i, z_j)}{\varepsilon}\right)]$
Update θ using $\frac{\partial \mathcal{L}_\theta}{\partial \theta}$ and λ_2 to minimize $\mathcal{L}_\theta$;
until converged or reach the max number of epochs

where gradients of dual variables are given. By the chain rule, the decoder can be trained through back-propagation.

4. Related work

There are several examples of OT-based generative modeling in the past few years. The majority focus on classical optimal transport (Arjovsky et al., 2017; Tomczak & Welling, 2018; Patrini et al., 2018; Seguy et al., 2018; Deshpande et al., 2018; An et al., 2020; Rout et al., 2021). WGAN (Arjovsky et al., 2017) replaced the Jensen-Shannon divergence optimized in the original GAN framework with the Wasserstein-1 distance. Deshpande et al. (2018) proposed to further modify the GAN by approximating Wasserstein-1 distance by sliced Wasserstein distance. People also tried to apply OT on auto-encoder models. WAE (Tomczak & Welling, 2018) aims to minimize the penalized Wasserstein distance between model distribution and target distribution. SAE (Patrini et al., 2018) replaced the MMD or GAN penalty term in WAE with Sinkhorn divergence. Sinkhorn generative model (Genevay et al., 2018) minimizes the Sinkhorn divergence between data distribution and generative distribution in a mini-batch manner. LSOT (Seguy et al., 2018), AE-OT (An et al., 2020) and OTM (Rout et al., 2021) are computing the OT maps instead of the plans in generative modeling. LSOT considers continuous OT with regularization. AE-OT solves the semi-discrete OT between a noise distribution and encoded data distribution captured by an autoencoder. OTM is trying to find the OT maps in observation space that is different from AE-OT which happens in latent space.

Our methods are different from the above methods as we compute the EOT loss between two distributions supported on different spaces. We unify the training of the encoder in VAEs with the EOT problem and treat the decoder as a learnable cost function. The recast of VAE as EOT provides a new perspective of generative modeling and new explanations for problems that occurred during training.

5. Experiments

In this section, we explore various properties of C-VAE on a selection of synthetic and real datasets. And We also compare it with other autoencoder models including VAE, VAE-NF[1], WAE[2] and InfoVAE[3]. All of the models are trained with the exact same architecture (if they have the same components) across all experiments. We use Dual strategy for C-VAE training in all experiments.

Mixture of Gaussians. We first test our model on a two-dimensional synthetic dataset consisting of a mixture of 25 isotropic Gaussian distributions laid out on a grid (Dumoulin et al., 2016). The means are on the grids $\mu \in \{-2, 1, 0, 1, 2\}$ and we used a standard deviation $\sigma = 0.05$. Despite being a 2D toy example, it is not an easy task since the distribution defined there exhibits many modes that are separated by large low-density areas. In all experiments, we generated 300 samples from each Gaussian and synthesize a dataset with a total of 7500 data points. For all encoder and decoder neural networks, we used four fully connected layers with ReLU activations with Batch normalizations in between. All hidden layers have 256 neurons and we choose the latent dimension $d_z = 2$ to plot the latent space. The prior is a 2D Gaussian with a mean of 0 and a standard deviation of 1, i.e., $p(z) = \mathcal{N}(\mathbf{0}, \mathbf{I_d})$. We pick the best model based on hyperparameter searching.

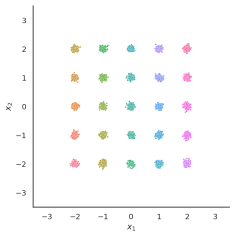


Figure 2. Mixture of 25 Gaussians in $\mathbb{R}^2$.

Visualization. To better understand how our model performs in generative tasks and latent representation learning, we visualize the random samples from models along with the training reconstructions. The samples are obtained by

[1]VAE-NF means VAE with normalizing flow. In particular, We used 10 layers of planar flow.

[2]WAE has penalty parameter equaling 1 for Mixture of Gaussians and 2 for MNIST.

[3]InfoVAE has hyperparameters $\alpha = 0$, $\lambda = 2$ for Mixture of Gaussians and $\alpha = 0$, $\lambda = 3$ for MNIST.

Figure 1. Columns from left to right represent (1) random samples, (2) reconstructions, (3) latent representations, (4) samples from aggregated posteriors and (5) posterior, respectively. Each row represents a model. Colors correspond to the classes. We can see that C-VAE has the best sample quality (1), the prior hole problem is resolved in C-VAE (4), C-VAE and VAE-NF can have non-isotropic-Gaussian posterior in contrast to Gaussians posteriors in others but VAE-NF is still approximately Gaussian-like (5).

generating a latent code from $p(z)$ first and then decoding it by $p_\theta(x|z)$. We also show the latent representation learned from the whole dataset, the aggregate posterior of the model, and the posterior distribution for the individual data point. In order to visualize the encoding distribution, we choose the mean of the encoding distribution as latent representations. Aggregated posterior is produced by ancestral sampling of $q(z|x)p_D(x)$. Figure 1 displays the results of all models in this experiment. We observe:

1. For the data reconstruction (column 2), all models do well. But VAE, VAE-NF, WAE, and InfoVAE all generate samples (column 1) lying on the edges and inside between different components of mixture Gaussians, which implies the learned model distribution P_θ fails to match the data distribution P_D, which lives only on the grids.
2. All models have learned well-separated representations (column 3). Prior holes—mismatch of aggregated posterior and prior—are visible for VAE, VAE-NF, WAE, and InfoVAE (column 4). Compared with VAE, InfoVAE has smaller holes and better random sample quality. The difference between two models is the explicit penalty term on the discrepancy of $q(z)$ and $p(z)$, which is consistent with our hypothesis about the importance of closing the holes inside the prior. C-VAE outperforms InfoVAE by implicitly having an infinite penalty on the gap. WAE ends with an almost-deterministic encoder as the $\sigma^2_\phi(x) \approx \mathbf{0}$ for all x. This makes the holes even larger than VAE.

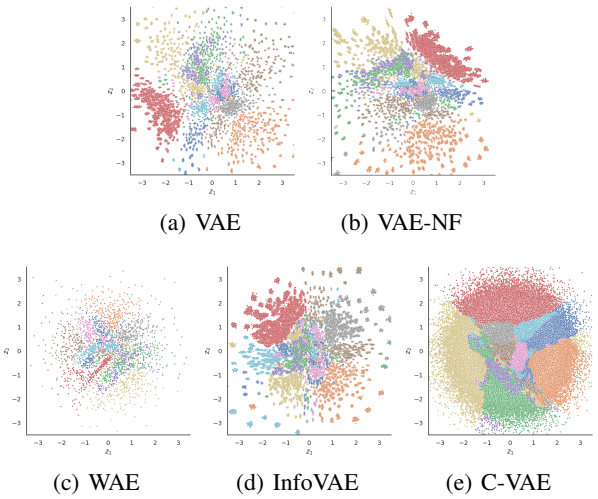
3. The last column of figure 1 is the density plots of posteriors of a single data point. VAE, WAE, and InfoVAE all have Gaussian posteriors by assumption. WAE's posterior collapsed to a point. VAE-NF shows a non-isotropic-Gaussian posterior. It's clear that the posterior has non-zero covariance between z_1 and z_2. But overall it's still close to Gaussian with single mode. C-VAE presents a very different posterior which has multiple modes.

Table 1. Quantitative results on the mixture of Gaussians

MODEL	HIGH DENSITY RATIO	STD W/I MODES
VAE	71.5± 0.005	0.0778± 0.0004
VAE-NF	53.6± 0.006	0.0833± 0.0006
WAE	60.8± 0.006	0.0764± 0.0005
INFOVAE	86.0± 0.004	0.0673± 0.0004
C-VAE	**98.5± 0.001**	**0.0478± 0.0003**

Table 2. MMD between prior and aggregate posterior

MODEL	MMD
VAE	0.0106± 0.0004
VAE-NF	0.0161± 0.0005
WAE	0.0060± 0.0002
INFOVAE	0.0045± 0.0002
C-VAE	**0.0032± 0.0005**



(a) VAE (b) VAE-NF

(c) WAE (d) InfoVAE (e) C-VAE

Figure 3. Samples from aggregated posterior $q(z)$. Each color represents a digit. C-VAE matches $p(z)$ best among the 5 models.

Quantitative results. To measure performance quantitatively, we follow the metrics used in (Azadi et al., 2020). We will assign each sample to its closest mixture component. A sample is considered as a high-density sample if it is within four standard deviations of its closest mixture component. The fraction of high-density samples in all samples will be calculated. Table 1 shows that C-VAE has attained the best high density ratio and the standard deviation of the learned model is 0.0478 which is closest to the ground truth of 0.05. We reported the Maximum Mean Discrepancy (MMD) between prior and aggregate posterior for all models in Table 2. C-VAE has the smallest MMD.

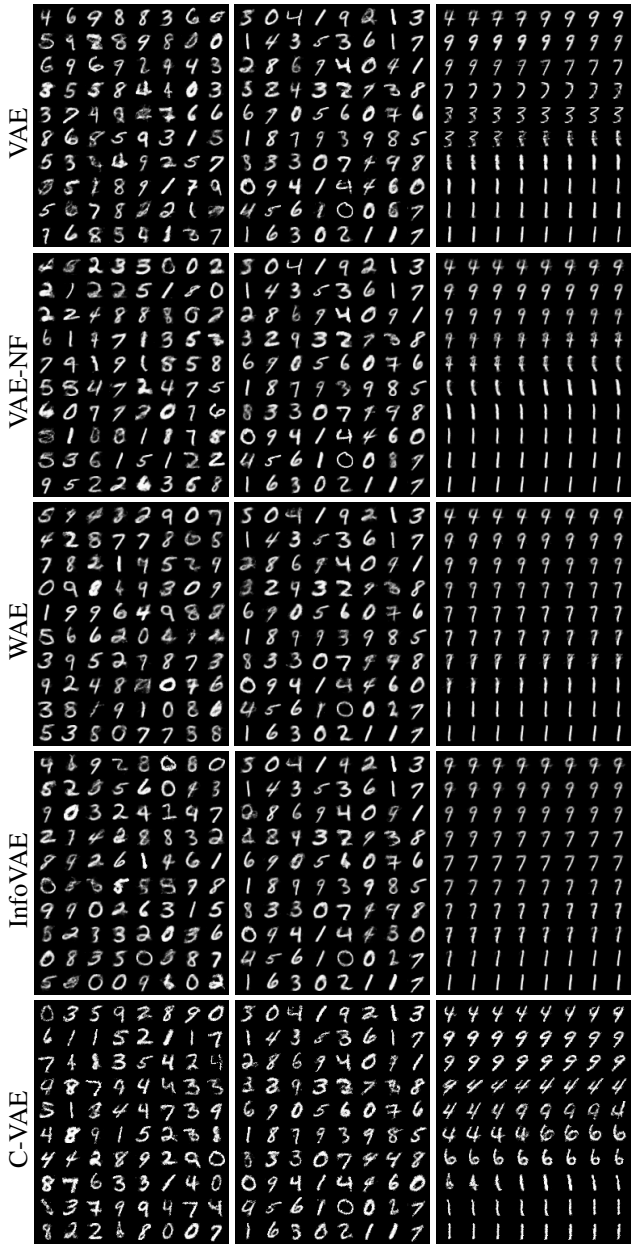


Figure 4. columns from left to right represent (1) random samples, (2) reconstructions, (3) interpolations. Each row represents a model. The quality of random samples generated by C-VAE is better in the sense that samples are sharper and there are fewer samples that look like an average of digits. In the right column, we could see the decoded images of linear interpolation in the latent space. The C-VAE interpolation is smoother and more realistic.

MNIST. We then run our model on a subset of MNIST which consists of 2560 different hand-written digits images from 10 classes. We again compare our model with VAE, VAE-NF, WAE, and InfoVAE on similar tasks of synthetic examples. We also examine the interpolation in the latent space of all models. Latent dimension $d_z = 2$. All the autoencoder networks have the same architecture. We use convolutional layers paired with ReLU as the building block for the encoding/decoding networks. We choose Bernoulli likelihood in the experiments. Models are trained through Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

The results are reported in figure 3 and figure 4. As expected, the quality of random samples and interpolation in latent space depends on how accurately $q(z)$ matches $p(z)$. figure 3 shows that C-VAE eliminates holes in the prior without hurting the performance of the model. WAE cannot match $p(z)$ perfectly because the encoder is collapsed to a deterministic map, which means $q(z)$ will have support on a finite number of z. The number is given by the number of data in the training set. InfoVAE mitigates but does not resolve the hole problem. Matching $q(z)$ to $p(z)$ better will need a larger penalty on the discrepancy, which will trade off against generative capacity. Interpolations, reconstructions, and samples are in figure 4.

6. Conclusion

By recasting VAE as EOT, we introduced an EOT-based training scheme for latent variable models, which enables flexible posterior approximation and prior selection. Our model resolved the prior hole problem naturally by EOT. We verify our claims on synthetic mixture of Gaussians dataset and MNIST.

Acknowledgements

This work was supported in part by DARPA Sail-on W911NF2020001, ASIST W912CG22C0001, and NSF MRI 2117429. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the DARPA or ARO, or the U.S. Government.

References

An, D., Guo, Y., Lei, N., Luo, Z., Yau, S.-T., and Gu, X. Ae-ot: a new generative model based on extended semi-discrete optimal transport. In *International Conference on Learning Representations*, 2020.

Aneja, J., Schwing, A. G., Kautz, J., and Vahdat, A. A contrastive learning approach for training variational autoencoder priors. In *Neural Information Processing Systems*, 2021.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, 2017.

Azadi, S., Olsson, C., Darrell, T., Goodfellow, I. J., and Odena, A. Discriminator rejection sampling. In *International Conference on Learning Representations*, 2020.

Burda, Y., Grosse, R. B., and Salakhutdinov, R. Importance weighted autoencoders. *CoRR*, abs/1509.00519, 2016.

Cai, L., Gao, H., and Ji, S. Multi-stage variational auto-encoders for coarse-to-fine image generation. In *SDM*, 2017.

Casale, F. P., Dalca, A. V., Saglietti, L., Listgarten, J., and Fusi, N. Gaussian process prior variational autoencoders. *ArXiv*, abs/1810.11738, 2018.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Dai, B. and Wipf, D. P. Diagnosing and enhancing VAE models. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*, 2019.

Davidson, T. R., Falorsi, L., De Cao, N., Kipf, T., and Tomczak, J. M. Hyperspherical variational auto-encoders. *34th Conference on Uncertainty in Artificial Intelligence (UAI-18)*, 2018.

Deshpande, I., Zhang, Z., and Schwing, A. G. Generative modeling using the sliced wasserstein distance. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3483–3491, 2018.

Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

Dumoulin, V., Belghazi, I., Poole, B., Lamb, A., Arjovsky, M., Mastropietro, O., and Courville, A. C. Adversarially learned inference. *ArXiv*, abs/1606.00704, 2016.

Fortuin, V., Hüser, M., Locatello, F., Strathmann, H., and Rätsch, G. Som-vae: Interpretable discrete representation learning on time series. In *International Conference on Learning Representations*, 2018.

Genevay, A., Cuturi, M., Peyré, G., and Bach, F. Stochastic optimization for large-scale optimal transport. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pp. 3440–3448, 2016.

Genevay, A., Peyre, G., and Cuturi, M. Learning generative models with sinkhorn divergences. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 1608–1617. PMLR, 2018.

Higgins, I., Matthey, L., Pal, A., Burgess, C. P., Glorot, X., Botvinick, M. M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Izmailov, P., Kirichenko, P., Finzi, M., and Wilson, A. G. Semi-supervised learning with normalizing flows. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Joo, W., Lee, W., Park, S., and Moon, I.-C. Dirichlet variational autoencoder, 2019.

Kantorovich, L. On the transfer of masses. *Doklady Akademii Nauk*, 1942.

Kim, J., Kong, J., and Son, J. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In *International Conference on Machine Learning*, 2021.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Kingma, D. P., Rezende, D. J., Mohamed, S., and Welling, M. Semi-supervised learning with deep generative models. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pp. 3581–3589, Cambridge, MA, USA, 2014. MIT Press.

Kingma, D. P., Salimans, T., Józefowicz, R., Chen, X., Sutskever, I., and Welling, M. Improving variational autoencoders with inverse autoregressive flow. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 4736–4744, 2016.

Lim, J., Ryu, S., Kim, J. W., and Kim, W. Y. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *Journal of Cheminformatics*, 10, 2018.

Makhzani, A., Shlens, J., Jaitly, N., and Goodfellow, I. Adversarial autoencoders. In *International Conference on Learning Representations*, 2016. URL http://arxiv.org/abs/1511.05644.

Patrini, G., Carioni, M., Forr'e, P., Bhargav, S., Welling, M., van den Berg, R., Genewein, T., and Nielsen, F. Sinkhorn autoencoders. *ArXiv*, abs/1810.01118, 2018.

Razavi, A., van den Oord, A., and Vinyals, O. Generating diverse high-fidelity images with vq-vae-2. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019.

Rezende, D. and Mohamed, S. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pp. 1530–1538, 2015.

Rezende, D. J. and Viola, F. Taming vaes, 2018. URL https://arxiv.org/abs/1810.00597.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32, pp. 1278–1286, 2014.

Rout, L., Korotin, A., and Burnaev, E. Generative modeling with optimal transport maps. *ArXiv*, abs/2110.02999, 2021.

Seguy, V., Damodaran, B. B., Flamary, R., Courty, N., Rolet, A., and Blondel, M. Large-scale optimal transport and mapping estimation. In *International Conference on Learning Representations*, 2018.

Takahashi, H., Iwata, T., Yamanaka, Y., Yamada, M., and Yagi, S. Variational autoencoder with implicit optimal priors. In *AAAI Conference on Artificial Intelligence*, 2019.

Tomczak, J. and Welling, M. Vae with a vampprior. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84, pp. 1214–1223. PMLR, 2018.

van den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. In *NIPS*, 2017.

Villani, C. Optimal transport: Old and new, 2008.

Yan, W., Zhang, Y., Abbeel, P., and Srinivas, A. Videogpt: Video generation using vq-vae and transformers, 2021.

Zhao, S., Song, J., and Ermon, S. Towards deeper understanding of variational autoencoding models. *ArXiv*, abs/1702.08658, 2017.

Zhao, S., Song, J., and Ermon, S. Infovae: Balancing learning and inference in variational autoencoders. In *AAAI Conference on Artificial Intelligence*, 2019.