
Sinkhorn AutoEncoders

Giorgio Patrini*
UvA Bosch Delta Lab

Rianne van den Berg*†
University of Amsterdam

Patrick Forré
University of Amsterdam

Marcello Carioni
KFU Graz

Samarth Bhargav
University of Amsterdam

Max Welling
University of Amsterdam
CIFAR

Tim Genewein‡
Bosch Center for
Artificial Intelligence

Frank Nielsen
École Polytechnique
Sony CSL

Abstract

Optimal transport offers an alternative to maximum likelihood for learning generative autoencoding models. We show that minimizing the p -Wasserstein distance between the generator and the true data distribution is equivalent to the unconstrained min-min optimization of the p -Wasserstein distance between the encoder aggregated posterior and the prior in latent space, plus a reconstruction error. We also identify the role of its trade-off hyperparameter as the capacity of the generator: its Lipschitz constant. Moreover, we prove that optimizing the encoder over any class of universal approximators, such as deterministic neural networks, is enough to come arbitrarily close to the optimum. We therefore advertise this framework, which holds for any metric space and prior, as a sweet-spot of current generative autoencoding objectives.

We then introduce the Sinkhorn autoencoder (SAE), which approximates and minimizes the p -Wasserstein distance in latent space via backpropagation through the Sinkhorn algorithm. SAE directly works on samples, i.e. it models the aggregated posterior as an implicit distribution, with no need for a reparameterization trick for gradients estimations. SAE is thus able to work with different metric spaces and priors with minimal adaptations.

We demonstrate the flexibility of SAE on latent spaces with different geometries and priors and compare with other methods on benchmark data sets.

1 INTRODUCTION

Unsupervised learning aims at finding the underlying rules that govern a given data distribution P_X . It can be approached by learning to mimic the data generation process, or by finding an adequate representation of the data. Generative Adversarial Networks (GAN) (Goodfellow et al., 2014) belong to the former class, by learning to transform noise into an implicit distribution that matches the given one. AutoEncoders (AE) (Hinton and Salakhutdinov, 2006) are of the latter type, by learning a representation that maximizes the mutual information between the data and its reconstruction, subject to an information bottleneck. Variational AutoEncoders (VAE) (Kingma and Welling, 2013; Rezende et al., 2014), provide both a generative model — i.e. a *prior* distribution P_Z on the latent space with a decoder $G(X|Z)$ that models the conditional likelihood — and an encoder $Q(Z|X)$ — approximating the *posterior* distribution of the generative model. Optimizing the exact marginal likelihood is intractable in latent variable models such as VAE’s. Instead one maximizes the Evidence Lower BOund (ELBO) as a surrogate. This objective trades off a reconstruction error of the input distribution P_X and a regularization term that aims at minimizing the Kullback-Leibler (KL) divergence from the approximate posterior $Q(Z|X)$ to the prior P_Z .

An alternative principle for learning generative autoencoders comes from the theory of Optimal Transport (OT) (Villani, 2008), where the usual KL-divergence $\text{KL}(P_X, P_G)$ is replaced by OT-cost divergences $W_c(P_X, P_G)$, among which the p -Wasserstein distances W_p are proper metrics. In the papers Tolstikhin et al. (2018); Bousquet et al. (2017) it was shown that the objective $W_c(P_X, P_G)$ can be re-written as the minimization of the reconstruction error of the input P_X over all *probabilistic* encoders

*Equal contributions. •Now at Deeptrace. †Now at Google Brain. ‡Now at DeepMind.

$Q(Z|X)$ constrained to the condition of matching the *aggregated posterior* Q_Z — the average (approximate) posterior $\mathbb{E}_{P_X}[Q(Z|X)]$ — to the prior P_Z in the latent space. In Wasserstein AutoEncoders (WAE) (Tolstikhin et al., 2018), it was suggested, following the standard optimization principles, to softly enforce that constraint via a penalization term depending on a choice of a divergence $D(Q_Z, P_Z)$ in latent space. For any such choice of divergence this leads to the minimization of a lower bound of the original objective, leaving the question about the status of the original objective open. Nonetheless, WAE empirically improves upon VAE for the two choices made there, namely either a Maximum Mean Discrepancy (MMD) (Gretton et al., 2007; Sriperumbudur et al., 2010, 2011), or an adversarial loss (GAN), again both in latent space.

We contribute to the formal analysis of autoencoders with OT. First, using the Monge-Kantorovich equivalence (Villani, 2008), we show that (in non-degenerate cases) the objective $W_c(P_X, P_G)$ can be reduced to the minimization of the reconstruction error of P_X over any class containing the class of all *deterministic* encoders $Q(Z|X)$, again constrained to $Q_Z = P_Z$. Second, when restricted to the p -Wasserstein distance $W_p(P_X, P_G)$, and by using a combination of triangle inequality and a form of data processing inequality for the generator G , we show that the soft and *unconstrained* minimization of the reconstruction error of P_X together with the penalization term $\gamma \cdot W_p(Q_Z, P_Z)$ is actually an *upper* bound to the original objective $W_p(P_X, P_G)$, where the regularization/trade-off hyperparameter γ needs to match at least the *capacity* of the generator G , i.e. its Lipschitz constant. This suggests that using a p -Wasserstein metric $W_p(Q_Z, P_Z)$ in latent space in the WAE setting (Tolstikhin et al., 2018) is a preferable choice.

Third, we show that the minimum of that objective can be approximated from above by any class of universal approximators for $Q(Z|X)$ to arbitrarily small error. In case we choose the L_p -norms $\|\cdot\|_p$ and corresponding p -Wasserstein distances W_p one can use the results of (Hornik, 1991) to show that any class of probabilistic encoders $Q(Z|X)$ that contains the class of *deterministic* neural networks has all those desired properties. This justifies the use of such classes in practice. Note that analogous results for the latter for other divergences and function classes are unknown.

Fourth, as a corollary we get the folklore claim that matching the aggregated posterior Q_Z and prior P_Z

is a *necessary* condition for learning the true data distribution P_X in rigorous mathematical terms. Any deviation will thus be punished with a poorer performance. Altogether, we have addressed and answered the open questions in (Tolstikhin et al., 2018; Bousquet et al., 2017) in detail and highlighted the sweet-spot framework for generative autoencoder models based on Optimal Transport (OT) for *any* metric space and *any* prior distribution P_Z , and with special emphasis on Euclidean spaces, L_p -norms and neural networks.

The theory supports practical innovations. We are now in a position to learn deterministic autoencoders, $Q(Z|X)$, $G(X|Z)$, by minimizing a reconstruction error for P_X and the p -Wasserstein distance on the latent space between samples of the aggregated posterior and the prior $W_p(\hat{Q}_Z, \hat{P}_Z)$. The computation of the latter is known to be difficult and costly (cp. Hungarian algorithm (Kuhn, 1955)). A fast approximate solution is provided by the *Sinkhorn algorithm* (Cuturi, 2013), which uses an entropic relaxation. We follow (Frogner et al., 2015) and (Genevay et al., 2018), by exploiting the differentiability of the Sinkhorn iterations, and unroll it for backpropagation. In addition we correct for the entropic bias of the Sinkhorn algorithm (Genevay et al., 2018; Feydy et al., 2018). Altogether, we call our method the *Sinkhorn AutoEncoder* (SAE).

The Sinkhorn AutoEncoder is agnostic to the analytical form of the prior, as it optimizes a sample-based cost function which is aware of the geometry of the latent space. Furthermore, as a byproduct of using deterministic networks, it models the aggregated posterior as an implicit distribution (Mohamed and Lakshminarayanan, 2016) with no need of the reparametrization trick for learning the encoder (Kingma and Welling, 2013). Therefore, with essentially no change in the algorithm, we can learn models with normally distributed priors and aggregated posteriors, as well as distributions living on manifolds such as hyperspheres (Davidson et al., 2018) and probability simplices.

In our experiments we explore how well the Sinkhorn AutoEncoder performs on the benchmark datasets MNIST and CelebA with different prior distributions P_Z and geometries in latent space, e.g. the Gaussian in Euclidean space or the uniform distribution on a hypersphere. Furthermore, we compare the SAE to the VAE (Kingma and Welling, 2013), to the WAE-MMD (Tolstikhin et al., 2018) and other methods of approximating the p -Wasserstein distance in latent space like the Hungarian algorithm (Kuhn, 1955)

and the Sliced Wasserstein AutoEncoder (Kolouri et al., 2018). We also explore the idea of matching the aggregated posterior Q_Z to a standard Gaussian prior P_Z via the fact that the 2-Wasserstein distance has a closed form for Gaussian distributions: we estimate the mean and covariance of Q_Z on minibatches and use the loss $W_2(\hat{Q}_Z, P_Z)$ for backpropagation. Finally, we train SAE on MNIST with a probability simplex as a latent space and visualize the matching of the aggregate posterior and the prior.

2 PRINCIPLES OF WASSERSTEIN AUTOENCODING

2.1 OPTIMAL TRANSPORT

We follow Tolstikhin et al. (2018) and denote with $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ the sample spaces and with X, Y, Z and P_X, P_Y, P_Z the corresponding random variables and distributions. Given a map $F : \mathcal{X} \rightarrow \mathcal{Y}$ we denote by $F_\#$ the push-forward map acting on a distribution P as $P \circ F^{-1}$. If $F(Y|X)$ is non-deterministic we define the push-forward $F(Y|X)_\# P_X$ of a distribution P as the induced marginal of the joint distribution $F(Y|X)P_X$. For any measurable non-negative cost $c : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^+ \cup \{\infty\}$, one can define the following *OT-cost* between distributions P_X and P_Y via:

$$W_c(P_X, P_Y) = \inf_{\Gamma \in \Pi(P_X, P_Y)} \mathbb{E}_{(X, Y) \sim \Gamma} [c(X, Y)], \quad (1)$$

where $\Pi(P_X, P_Y)$ is the set of all joint distributions that have P_X and P_Y as the marginals. The elements from $\Pi(P_X, P_Y)$ are called *couplings* from P_X to P_Y . If $c(x, y) = d(x, y)^p$ for a metric d and $p \geq 1$ then $W_p := \sqrt[p]{W_c}$ is called the p -Wasserstein distance.

Let P_X denote the true data distribution on \mathcal{X} . We define a *latent variable model* as follows: we fix a latent space \mathcal{Z} and a prior distribution P_Z on \mathcal{Z} and consider the conditional distribution $G(X|Z)$ (the decoder) parameterized by a neural network G . Together they specify a generative model as $G(X|Z)P_Z$. The induced marginal will be denoted by P_G . Learning P_G such that it approximates the true P_X is then defined as:

$$\min_G W_c(P_X, P_G). \quad (2)$$

Because of the infimum over $\Pi(P_X, P_G)$ inside W_c , this is intractable. To rewrite this objective we consider the posterior distribution $Q(Z|X)$ (the encoder) and its *aggregated posterior* Q_Z :

$$Q_Z = Q(Z|X)_\# P_X = \mathbb{E}_{X \sim P_X} Q(Z|X), \quad (3)$$

the induced marginal of the joint $Q(Z|X)P_X$.

2.2 THE WASSERSTEIN AUTOENCODER (WAE)

Tolstikhin et al. (2018) show that if the decoder $G(X|Z)$ is deterministic, i.e. $P_G = G_\# P_Z$, or in other words, if all stochasticity of the generative model is captured by P_Z , then:

$$W_c(P_X, P_G) = \inf_{\substack{Q(Z|X): \\ Q_Z = P_Z}} \mathbb{E}_{X \sim P_X} \mathbb{E}_{Z \sim Q(Z|X)} [c(X, G(Z))]. \quad (4)$$

Learning the generative model G with the WAE amounts to the objective:

$$\begin{aligned} \min_G \min_{Q(Z|X)} & \mathbb{E}_{X \sim P_X} \mathbb{E}_{Z \sim Q(Z|X)} [c(X, G(Z))] \\ & + \beta \cdot D(Q_Z, P_Z), \end{aligned} \quad (5)$$

where $\beta > 0$ is a Lagrange multiplier and D is any divergence measure on probability distributions on \mathcal{Z} . The specific choice for D is left open. WAE uses either MMD (Gretton et al., 2012) or a discriminator trained adversarially for D . As discussed in Bousquet et al. (2017), Eq. 5 is a lower bound of Eq. 4 for any choice of D and any value of $\beta > 0$. Minimizing this lower bound does not ensure a minimization of the original objective of Eq. 4.

2.3 THEORETICAL CONTRIBUTIONS

We improve upon the analysis of Tolstikhin et al. (2018) of generative autoencoders in the framework of Optimal Transport in several ways. Our contributions can be summarized by the following theorem, upon which we will comment directly after.

Theorem 2.1. *Let \mathcal{X}, \mathcal{Z} be endowed with any metrics and $p \geq 1$. Let P_X be a non-atomic distribution¹ and $G(X|Z)$ be a deterministic generator/decoder that is γ -Lipschitz. Then we have the equality:*

$$W_p(P_X, P_G) = \inf_{Q \in \mathcal{F}} \sqrt[p]{\mathbb{E}_{X \sim P_X} \mathbb{E}_{Z \sim Q(Z|X)} [d(X, G(Z))^p]} + \gamma \cdot W_p(Q_Z, P_Z), \quad (6)$$

where \mathcal{F} is any class of probabilistic encoders that at least contains a class of universal approximators. If \mathcal{X}, \mathcal{Z} are Euclidean spaces endowed with the L_p -norms $\|\cdot\|_p$ then a valid minimal choice for \mathcal{F} is the class of all deterministic neural network encoders Q

¹A probability measure is non-atomic if every point in its support has zero measure. It is important to distinguish between the *empirical* data distribution \hat{P}_X , which is always atomic, and the underlying *true* distribution P_X , only which we need to assume to be non-atomic.

(here written as a function), for which the objective reduces to:

$$W_p(P_X, P_G) = \inf_{Q \in \mathcal{F}} \sqrt[p]{\mathbb{E}_{X \sim P_X} [\|X - G(Q(X))\|_p^p]} + \gamma \cdot W_p(Q_Z, P_Z).$$

The proof of Theorem 2.1 can be found in Appendix A, B and C. It uses the following three arguments:

i.) It is the Monge-Kantorovich equivalence (Villani 2008) for non-atomic P_X that allows us to restrict to *deterministic* encoders $Q(Z|X)$. This is a first theoretical improvement over the Eq. 4 from Tolstikhin et al. (2018).

ii.) The *upper bound* can be achieved by a simple *triangle inequality*:

$$W_p(P_X, P_G) \leq W_p(P_X, \tilde{P}_X) + W_p(\tilde{P}_X, P_G),$$

where $\tilde{P}_X := G_\# Q(Z|X)_\# P_X = G_\# Q_Z$ is the reconstruction of P_X . Note that the triangle inequality is not available for other general cost functions or divergences. This might be a reason for the difficulty of getting upper bounds in such settings. On the other hand, if a divergence satisfies the triangle inequality then one can use the same argument to arrive at new variational optimization objectives and principles.

iii.) We then prove the *data processing inequality* for the W_p -distance:

$$W_p(G_\# Q_Z, G_\# P_Z) \leq \gamma \cdot W_p(Q_Z, P_Z),$$

with any $\gamma \geq \|G\|_{\text{Lip}}$, the Lipschitz constant of G . Such an inequality is available and known for several other divergences usually with $\gamma = 1$.

Putting all three pieces together we immediately arrive at the *equality* (upper and lower bound) of the first part of Theorem 2.1. This insight directly suggests that using the divergence $W_p(Q_Z, P_Z)$ in latent space with a hyperparameter $\gamma \geq \|G\|_{\text{Lip}}$ in the WAE setting is a preferable choice. These are two further improvements over Tolstikhin et al. (2018). Note that if G is a neural network with activation function g with $\|g'\|_\infty \leq 1$ (e.g. ReLU, sigmoid, tanh, etc.) and weight matrices $(B_\ell)_{\ell=1,\dots,L}$, then G is γ -Lipschitz for any $\gamma \geq \|B_1\|_p \cdots \|B_L\|_p$, where the latter is the product of the L_p -matrix norms (cp. Balan et al. (2017)).

iv.) For the second part of Theorem 2.1 we use the universal approximator property of neural networks (Hornik 1991) and the compatibility of the L_p -norm $\|\cdot\|_p$ -norm with the p -Wasserstein distance W_p . Proving such statements for other divergences seems to require much more effort (if possible at all).

When the encoders are restricted to be neural networks of limited capacity, e.g. if their architecture is fixed, then enforcing $Q_Z \approx P_Z$ might not be feasible in the general case of dimensionality mismatch between \mathcal{X} and \mathcal{Z} (Rubenstein et al., 2018). In fact, since the class of deterministic neural networks (of limited capacity) is much smaller than the class of deterministic measurable maps, one might consider adding noise to the output, i.e. use stochastic networks instead. Nonetheless, neural networks can approximate any measurable map up to arbitrarily small error (Hornik 1991). Furthermore, in practice the encoder $Q(Z|X)$ maps from the high dimensional data space \mathcal{X} to the much lower dimensional latent space \mathcal{Z} , suggesting that the task of matching distributions in the lower dimensional latent space \mathcal{Z} should be feasible. Also, in view of Theorem 2.1 it follows that learning deterministic autoencoders is sufficient to approach the theoretical bound and thus will be our empirical choice.

Theorem 2.1 certifies that, *failing* to match aggregated posterior and prior makes learning the data distribution impossible. Matching in latent space should be seen as fundamental as minimizing the reconstruction error, a fact known about the performance of VAE (Hoffman and Johnson, 2016; Higgins et al., 2017; Alemi et al., 2018; Rosca et al., 2018). This necessary condition for learning the data distribution turns out to be also sufficient assuming that the set of encoders is expressive enough to nullify the reconstruction error.

With the help of Theorem 2.1 we arrive at the following *unconstrained min-min-optimization* objective over deterministic decoder and encoder neural networks (Q written as a function here):

$$\min_G \min_Q \sqrt[p]{\mathbb{E}_{X \sim P_X} [\|X - G(Q(X))\|_p^p]} + \gamma \cdot W_p(Q_Z, P_Z),$$

with $\gamma \geq \|G\|_{\text{Lip}}$ for all occurring G .

3 THE SINKHORN AUTOENCODER

3.1 ENTROPY REGULARIZED OPTIMAL TRANSPORT

Even though the theory supports the use of the p -Wasserstein distance $W_p(Q_Z, P_Z)$ in latent space, it is notoriously hard to compute or estimate. In practice, we will need to approximate $W_p(Q_Z, P_Z)$ via samples from Q_Z (and P_Z). The sample version $W_p(\hat{Q}_Z, \hat{P}_Z)$

with $\hat{P}_Z = \frac{1}{M} \sum_{m=1}^M \delta_{z_m}$ and $\hat{Q}_Z = \frac{1}{M} \sum_{m=1}^M \delta_{\tilde{z}_m}$ has an exact solution, which can be computed using the Hungarian algorithm (Kuhn, 1955) in near $O(M^3)$ time (time complexity). Furthermore, $W_p(\hat{Q}_Z, \hat{P}_Z)$ will differ from $W_p(Q_Z, P_Z)$ in size of about $O(M^{-\frac{1}{k}})$ (sample complexity), where k is the dimension of Z (Weed and Bach, 2017). Both complexity measures are unsatisfying in practice, but they can be improved via *entropy regularization* (Cuturi, 2013), which we will explain next.

Following Genevay et al. (2018, 2019); Feydy et al. (2018) we define the *entropy regularized OT cost* with $\varepsilon \geq 0$:

$$\begin{aligned} \tilde{S}_{c,\varepsilon}(P_X, P_Y) := & \inf_{\Gamma \in \Pi(P_X, P_Y)} \mathbb{E}_{(X,Y) \sim \Gamma} [c(X, Y)] \\ & + \varepsilon \cdot \text{KL}(\Gamma, P_X \otimes P_Y). \end{aligned} \quad (7)$$

This is in general not a divergence due to its entropic bias. When we remove this bias we arrive at the *Sinkhorn divergence*:

$$\begin{aligned} S_{c,\varepsilon}(P_X, P_Y) := & \tilde{S}_{c,\varepsilon}(P_X, P_Y) \\ & - \frac{1}{2} \left(\tilde{S}_{c,\varepsilon}(P_X, P_X) + \tilde{S}_{c,\varepsilon}(P_Y, P_Y) \right). \end{aligned} \quad (8)$$

The Sinkhorn divergence has the following limiting behaviour:

$$\begin{aligned} S_{c,\varepsilon}(P_X, P_Y) & \xrightarrow{\varepsilon \rightarrow 0} W_c(P_X, P_Y), \\ S_{c,\varepsilon}(P_X, P_Y) & \xrightarrow{\varepsilon \rightarrow \infty} \text{MMD}_{-c}(P_X, P_Y). \end{aligned}$$

This means that the Sinkhorn divergence $S_{c,\varepsilon}$ interpolates between OT-divergences and MMDs (Gretton et al., 2012). On the one hand, for small ε it is known that $S_{c,\varepsilon}$ deviates from the initial objective W_c by about $O(\varepsilon \log(1/\varepsilon))$ (Genevay et al., 2019). On the other hand, if ε is big enough then $S_{c,\varepsilon}$ will have the more favourable sample complexity of $O(M^{-\frac{1}{2}})$ of MMDs, which is independent of the dimension, and was proven in Genevay et al. (2019). Furthermore, the *Sinkhorn algorithm* (Cuturi, 2013), which will be explained in the section 3.3, allows for faster computation of the Sinkhorn divergence $S_{c,\varepsilon}$ with time complexity close to $O(M^2)$ (Altschuler et al., 2017). Therefore, if we balance ε well, we are close to our original objective and at the same time have favourable computational and statistical properties.

3.2 THE SINKHORN AUTOENCODER OBJECTIVE

Guided by the theoretical insights, we can restrict the WAE framework (Tolstikhin et al., 2018) to Sinkhorn

divergences with cost \tilde{c} in latent space and c in data space to arrive at the objective:

$$\begin{aligned} \min_G \min_{Q(Z|X)} & \mathbb{E}_{X \sim P_X} \mathbb{E}_{Z \sim Q(Z|X)} [c(X, G(Z))] \\ & + \beta \cdot S_{\tilde{c},\varepsilon}(Q_Z, P_Z), \end{aligned} \quad (9)$$

with hyperparameters $\beta \geq 0$ and $\varepsilon \geq 0$.

Restricting further to p -Wasserstein distances, corresponding Sinkhorn divergences and deterministic encoder/decoder neural networks, we arrive at the *Sinkhorn AutoEncoder (SAE)* objective:

$$\begin{aligned} \min_G \min_Q & \sqrt[p]{\mathbb{E}_{X \sim P_X} [\|X - G(Q(X))\|_p^p]} \\ & + \gamma \cdot S_{\|\cdot\|_p^p, \varepsilon}(Q_Z, P_Z)^{\frac{1}{p}}, \end{aligned} \quad (10)$$

which is then up to the ε -terms close to the original objective. Note that for computational reasons it is sometimes convenient to remove the p -th roots again. The inequality $\sqrt[p]{a} + \sqrt[p]{b} \leq 2\sqrt[p]{a+b}$ shows that the additional loss is small, while still minimizing an upper bound (using $\beta := \gamma^p$).

3.3 THE SINKHORN ALGORITHM

Now that we have the general Sinkhorn AutoEncoder optimization objective, we need to review how the Sinkhorn divergence $S_{\tilde{c},\varepsilon}(Q_Z, P_Z)$ can be estimated in practice by the *Sinkhorn algorithm* (Cuturi, 2013) using samples.

If we take M samples each from Q_Z and P_Z , we get the corresponding empirical (discrete) distributions concentrated on M points: $\hat{P}_Z = \frac{1}{M} \sum_{m=1}^M \delta_{z_m}$ and $\hat{Q}_Z = \frac{1}{M} \sum_{m=1}^M \delta_{\tilde{z}_m}$. Then, the optimal coupling of the (empirical) entropy regularized OT-cost $\tilde{S}_{\tilde{c},\varepsilon}(\hat{Q}_Z, \hat{P}_Z)$ with $\varepsilon \geq 0$ is given by the matrix:

$$R^* := \arg \min_{R \in S_M} \frac{1}{M} \langle R, \tilde{C} \rangle_F - \varepsilon \cdot H(R), \quad (11)$$

where $\tilde{C}_{ij} = \tilde{c}(\tilde{z}_i, z_j)$ is the matrix associated to the cost \tilde{c} , R is a doubly stochastic matrix as defined in $S_M = \{R \in \mathbb{R}_{\geq 0}^{M \times M} \mid R\mathbf{1} = \mathbf{1}, R^T\mathbf{1} = \mathbf{1}\}$, and $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product; $\mathbf{1}$ is the vector of ones and $H(R) = -\sum_{i,j=1}^M R_{i,j} \log R_{i,j}$ is the entropy of R .

Cuturi (2013) shows that the SINKHORN Algorithm 1 (Sinkhorn, 1964) returns its ε -regularized optimum R^* (see Eq. 11) in the limit $L \rightarrow \infty$, which is also unique due to strong convexity of the entropy. The Sinkhorn algorithm is a fixed point algorithm that is much faster than the Hungarian algorithm: it runs in nearly $O(M^2)$ time (Altschuler et al., 2017) and

Algorithm 1 SINKHORN

Input: $\{\tilde{z}_i\}_{i=1}^M \sim Q_Z$, $\{z_j\}_{j=1}^M \sim P_Z$, ε , L
 $\forall i, j : \tilde{C}_{ij} = \tilde{c}(\tilde{z}_i, z_j)$
 $K = \exp(-\tilde{C}/\varepsilon)$, $u \leftarrow \mathbf{1}$ # elem-wise exp
repeat until convergence, but at most L times:
 $v \leftarrow \mathbf{1}/(K^\top u)$ # elem-wise division
 $u \leftarrow \mathbf{1}/(Kv)$
 $R^* \leftarrow \text{Diag}(u) K \text{Diag}(v)$ # plus rounding step
Output: R^* , \tilde{C} .

can be efficiently implemented with matrix multiplications; see Algorithm 1. For better differentiability properties we deviate from Eq. 8 and use the *unbiased sharp Sinkhorn loss* (Luise et al., 2018) Genevay et al. (2018) by dropping the entropy terms (only) in the evaluations:

$$\begin{aligned} \bar{S}_{\tilde{c}, \varepsilon}(\hat{Q}_Z, \hat{P}_Z) := & \frac{1}{M} \langle R^*, \tilde{C} \rangle_F \\ & - \frac{1}{2M} \left(\langle R_{\hat{Q}_Z}^*, \tilde{C}_{\hat{Q}_Z} \rangle_F + \langle R_{\hat{P}_Z}^*, \tilde{C}_{\hat{P}_Z} \rangle_F \right), \end{aligned} \quad (12)$$

where the indices \hat{Q}_Z , \hat{P}_Z refer to Eq. 11 applied to the samples from Q_Z in both arguments and then P_Z in both arguments, respectively.

Since this only deviates from Eq. 8 in ε -terms we still have all the mentioned properties, e.g. that the optimum of this Sinkhorn distance approaches the optimum of the OT-cost with the stated rate (Genevay et al., 2018; Cominetti and San Martín, 1994; Weed, 2018). Furthermore, for numerical stability we use the Sinkhorn algorithm in log-space (Chizat et al., 2016; Schmitzer, 2016). In order to round the R that results from a finite number L of Sinkhorn iterations to a doubly stochastic matrix, we use the procedure described Algorithm 2 of (Altschuler et al., 2017).

The smaller the ε , the smaller the entropy and the better the approximation of the OT-cost. At the same time, a larger number of steps $O(L)$ is needed to converge, while the rate of convergence remains linear in L (Genevay et al., 2018). Note that all Sinkhorn operations are differentiable. Therefore, when the distance is used as a cost function, we can unroll $O(L)$ iterations and backpropagate (Genevay et al., 2018). In conclusion, we obtain a differentiable surrogate for OT-cost between empirical distributions; the approximation arises from sampling, entropy regularization and the finite amount of steps in place of convergence.

Algorithm 2 SAE TRAINING ROUND

Input: encoder weights A , decoder weights B , ε , L , β
Minibatch: $x = \{x_i\}_{i=1}^M \sim P_X$, $z = \{z_j\}_{j=1}^M \sim P_Z$
 $\tilde{z} \leftarrow Q_A(x)$, $\tilde{x} \leftarrow G_B(\tilde{z})$
 $D = \frac{1}{M} \|x - \tilde{x}\|_p^p$
 $S = \text{SINKHORNLOSS}(\tilde{z}, z, \varepsilon, L)$, # 3 x Alg. 1 + Eq. 12
Update: A, B with gradient $\nabla_{(A, B)}(D + \beta \cdot S)$.

3.4 TRAINING THE SINKHORN AUTOENCODER

To train the Sinkhorn AutoEncoder with encoder Q_A , decoder G_B and with weights A, B , resp., we sample minibatches $x = \{x_i\}_{i=1}^M$ from the data distribution P_X and $z = \{z_j\}_{j=1}^M$ from the prior P_Z . After encoding x we then run the SINKHORN Algorithm 1 three times (for (x, z) , (x, x) and (z, z)) to find the optimal couplings and then compute the unbiased SINKHORN-Loss via Eq. 12. Note that the L SINKHORN steps in Algorithm 1 are differentiable. The weights can then be updated via (auto-)differentiation through the SINKHORN steps (together with the gradient of the reconstruction loss). One training round is summarized in Algorithm 2. Small ε and large L worsen the numerical stability of the Sinkhorn. In most experiments, both c and \tilde{c} will be $\|\cdot\|_2^2$. Experimentally we found that the re-calculation of the three optimal couplings at each iteration is not a significant overhead.

SAE can in principle work with arbitrary priors. The only requirement coming from the Sinkhorn is the ability to generate samples. The choice should be motivated by the desired geometric properties of the latent space.

4 CLOSED FORM OF THE 2-WASSERSTEIN DISTANCE

The 2-Wasserstein distance $W_2(Q_Z, P_Z)$ has a closed form in Euclidean space if both Q_Z and P_Z are Gaussian (Peyré and Cuturi (2018) Rem. 2.31):

$$\begin{aligned} W_2^2(\mathcal{N}(\mu_1, \Sigma_1), \mathcal{N}(\mu_2, \Sigma_2)) = & \|\mu_1 - \mu_2\|_2^2 \\ & + \text{tr} \left(\Sigma_1 + \Sigma_2 - 2 \left(\Sigma_2^{\frac{1}{2}} \Sigma_1 \Sigma_2^{\frac{1}{2}} \right)^{\frac{1}{2}} \right), \end{aligned} \quad (13)$$

which will further simplify if P_Z is standard Gaussian. Even though the aggregated posterior Q_Z might not be Gaussian we use the above formula for matching and backpropagation, by estimating μ_1 and Σ_1 on minibatches of Q_Z via the standard formulas: $\hat{\mu}_1 := \frac{1}{M} \sum_{i=1}^M \tilde{z}_i$ and $\hat{\Sigma}_1 := \frac{1}{M-1} \sum_{i=1}^M (\tilde{z}_i - \hat{\mu}_1)(\tilde{z}_i - \hat{\mu}_1)^T$. We refer to this method as W2GAE (Wasserstein Gaussian AutoEncoder). We will compare this

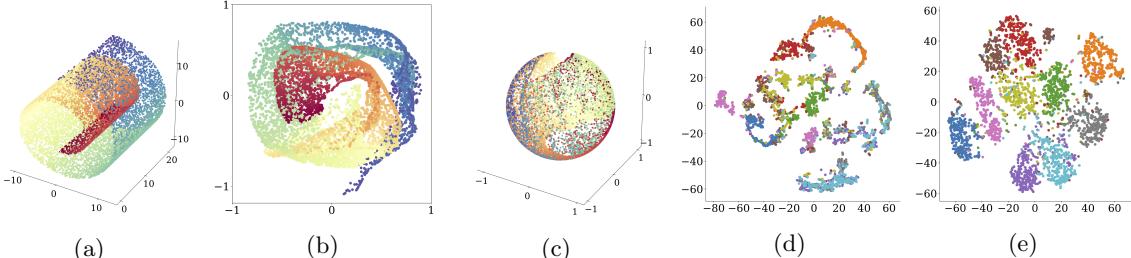


Figure 1: a) Swiss Roll and its b) squared and c) spherical embeddings learned by Sinkhorn encoders. MNIST embedded onto a 10D sphere viewed through t-SNE, with classes by colours: d) encoder only or e) encoder + decoder.

method against SAE and other baselines as discussed next in the related work section.

5 RELATED WORK

The Gaussian prior is common in VAE’s for the reason of tractability. In fact, changing the prior and/or the approximate posterior distributions requires the use of tractable densities and the appropriate reparametrization trick. A hyperspherical prior is used by [Davidson et al. \(2018\)](#) with improved experimental performance; the algorithm models a Von Mises-Fisher posterior, with a non-trivial posterior sampling procedure and a reparametrization trick based on rejection sampling. Our implicit encoder distribution sidesteps these difficulties. Recent advances on variable reparametrization can also simplify these requirements [Figurnov et al. \(2018\)](#). We are not aware of methods embedding on probability simplices, except the use of Dirichlet priors by the same [Figurnov et al. \(2018\)](#).

[Hoffman and Johnson \(2016\)](#) showed that the objective of a VAE does not force the aggregated posterior and prior to match, and that the mutual information of input and codes may be minimized instead. Just like the WAE, SAE avoids this effect by construction. [Makhzani et al. \(2015\)](#) and WAE improve latent matching by GAN/MMD. With the same goal, [Alemi et al. \(2017\)](#) and [Tomczak and Welling \(2017\)](#) introduce learnable priors in the form of a mixture of posteriors, which can be used in SAE as well.

The [Sinkhorn \(1964\)](#) algorithm gained interest after [Cuturi \(2013\)](#) showed its application for fast computation of Wasserstein distances. The algorithm has been applied to ranking [\(Adams and Zemel, 2011\)](#), domain adaptation [\(Courty et al., 2014\)](#), multi-label classification [\(Frogner et al., 2015\)](#), metric learning [\(Huang et al., 2016\)](#) and ecological inference [\(Muzeil et al., 2017\)](#). [Santa Cruz et al. \(2017\); Linderman et al. \(2018\)](#) used it for supervised combinatorial losses. Our use of the Sinkhorn for generative modeling is akin to that of [Genevay et al. \(2018\)](#), which

matches data and model samples with adversarial training, and to [Ambrogioni et al. \(2018\)](#), which matches samples from the model joint distribution and a variational joint approximation. WAE and WGAN objectives are linked respectively to primal and dual formulations of OT [\(Tolstikhin et al., 2018\)](#).

Our approach for training the encoder alone qualifies as self-supervised representation learning [\(Donahue et al., 2017; Noroozi and Favaro, 2016; Noroozi et al., 2017\)](#). As in noise-as-target (NAT) [\(Bojanowski and Joulin, 2017\)](#) and in contrast to most other methods, we can sample pseudo labels (from the prior) independently from the input. In Appendix D we show a formal connection with NAT.

Another way of estimating the 2-Wasserstein distance in Euclidean space is the Sliced Wasserstein AutoEncoder (SWAE) [\(Kolouri et al., 2018\)](#). The main idea is to sample one-dimensional lines in Euclidean space and exploit the explicit form of the 2-Wasserstein distance in terms of cumulative distribution functions in the one-dimensional setting. We will compare our methods to SWAE as well.

6 EXPERIMENTS

6.1 REPRESENTATION LEARNING WITH SINKHORN ENCODERS

We demonstrate qualitatively that the Sinkhorn distance is a valid objective for unsupervised feature learning by training the encoder in isolation. The task consists of embedding the input distribution in a lower dimensional space, while preserving the local data geometry and minimizing the loss function $L = \frac{1}{M} \langle R^*, \tilde{C} \rangle_F$, with $\tilde{c}(z, z') = \|z - z'\|_2^2$. Here M is the minibatch size.

We display the representation of a 3D Swiss Roll and MNIST. For the Swiss Roll we set $\epsilon = 10^{-3}$, while for MNIST it is set to 0.5, and L is picked to ensure convergence. For the Swiss roll (Figure 1a), we use a 50-50 fully connected network with ReLUs.

method	prior	cost	MNIST				CelebA			
			β	MMD	RE	FID	β	MMD	RE	FID
VAE	\mathcal{N}	KL	1	0.28	12.22	11.4	1	0.20	94.19	55
β -VAE	\mathcal{N}	KL	0.1	2.20	11.76	50.0	0.1	0.21	67.80	65
WAE	\mathcal{N}	MMD	100	0.50	7.07	24.4	2000*	0.21	65.45	58
SWAE	\mathcal{N}	SW	100	0.32	7.46	18.8	100	0.21	65.28	64
W2GAE (ours)	\mathcal{N}	W_2^2	1	0.67	7.04	30.5	1	0.20	65.55	58
HAE (ours)	\mathcal{N}	Hungarian	100	5.79	11.84	16.8	100	32.09	84.51	293
SAE (ours)	\mathcal{N}	Sinkhorn	100	5.34	12.81	17.2	100	4.82	90.54	187
HVAE [†]	\mathcal{H}	KL	1	0.25	12.73	21.5	-	-	-	-
WAE	\mathcal{H}	MMD	100	0.24	7.88	22.3	2000*	0.25	66.54	59
SWAE	\mathcal{H}	SW	100	0.24	7.80	27.6	100	0.41	63.64	80
HAE (ours)	\mathcal{H}	Hungarian	100	0.23	8.69	12.0	100	0.26	63.49	58
SAE (ours)	\mathcal{H}	Sinkhorn	100	0.25	8.59	12.5	100	0.24	63.97	56

Table 1: Results of the autoencoding task. Top 3 results for the FID scores are indicated with boldface numbers. We compute MMD in latent space to evaluate the matching between the aggregated posterior and prior. MMD results are reported times 10^2 . Note that MMD scores are not comparable for different priors. For SAE and the Gaussian prior, we used $\epsilon = 10$ as lower values led to numerical instabilities. For the hypersphere we set $\epsilon = 0.1$. *The value of $\beta = 2000$ is similar to the value $\lambda = 100$ as used in (Tolstikhin et al., 2018), as a prefactor of 0.05 was used there for the reconstruction cost. [†]Comparing with Davidson et al. (2018) in high-dimensional latent spaces turned out to be unfeasible, due to CPU-based computations.

Figures 1b, 1c show that the local geometry of the Swiss Roll is conserved in the new representational spaces — a square and a sphere. Figure 1d shows the t-SNE visualization (Maaten and Hinton, 2008) of the learned representation of the MNIST test set. With neither labels nor reconstruction error, we learn an embedding that is aware of class-wise clusters. Minimization of the Sinkhorn distance achieves this by encoding onto a d -dimensional hypersphere with a uniform prior, such that points are encouraged to map far apart. A contractive force is present due to the inductive prior of neural networks, which are known to be Lipschitz functions. On the one hand, points in the latent space disperse in order to fill up the sphere; on the other hand, points close on image space cannot be mapped too far from each other. As a result, local distances are conserved while the overall distribution is spread. When the encoder is combined with a decoder G the contractive force is enlarged: they collaborate in learning a latent space which makes reconstruction possible despite finite capacity; see Figure 1e.

6.2 AUTOENCODING EXPERIMENTS

For the autoencoding task we compare SAE against (β) -VAE, HVAE, SWAE and WAE-MMD. We furthermore denote the model that matches the samples in latent space with the Hungarian algorithm with HAE. Where compatible, all methods are evaluated both on the hypersphere and with a standard normal prior. Results from our proposed W2GAE method as discussed in section 4 for Gaussian priors are shown as well. We compute FID scores (Heusel et al., 2017) on CelebA and MNIST. For MNIST we use LeNet as

proposed in (Bińkowski et al., 2018). For details on the experimental setup, see Appendix E. The results for MNIST and CelebA are shown in Table I. Extrapolations, interpolations and samples of WAE and SAE for CelebA are shown in Fig. 2. Visualizations for MNIST are shown in Appendix D. Interpolations on the hypersphere are defined on geodesics connecting points on the hypersphere. FID scores of SAE with a hyperspherical prior are on par or better than the competing methods. Note that although the FID scores for the VAE are slightly better than that of SAE/HAE, the reconstruction error of the VAE is significantly higher. Surprisingly, the simple W2GAE method is on par with WAE on CelebA.

For the Gaussian prior on CelebA, both HAE and SAE perform very poorly. In appendix F we analyzed the behaviour of the Hungarian algorithm in isolation for two sets of samples from high-dimensional Gaussian distributions. The Hungarian algorithm finds a better matching between samples from a smaller variance Gaussian with samples from the standard normal distribution. This behaviour gets worse for higher dimensions, and also occurs for the Sinkhorn algorithm. This might be due to the fact that most probability mass of a high-dimensional isotropic Gaussian with standard deviation σ lies on a thin annulus at radius $\sigma\sqrt{d}$ from its origin. For a finite number of samples the L_2^2 cost function can lead to a lower matching cost for samples between two annuli of different radii. This effect leads to an encoder with a variance lower than one. When sampling from the prior after training, this yields saturated sampled images. See Appendix D for reconstructions and samples for HAE with a Gaussian prior on CelebA. Note that neither SWAE and W2GAE suffer from this



Figure 2: From left to right: CelebA extrapolations, interpolations, and samples. Models from Table 1: WAE with a Gaussian prior (top) and SAE with a uniform prior on the hypersphere (bottom).

problem in our experiments, even though these methods also provide an estimate of the 2-Wasserstein distance. For W2GAE this problem does start at even higher dimensions (Appendix F).

6.3 DIRICHLET PRIORS

We further demonstrate the flexibility of SAE by using Dirichlet priors on MNIST. The prior draws samples on the probability simplex; hence we constrain the encoder by a final softmax layer. We use priors that concentrate on the vertices with the purpose of clustering the digits. A 10-dimensional $\text{Dir}(1/2)$ prior (Figure 3a) results in an embedding qualitatively similar to the uniform sphere (1e). With a more skewed prior $\text{Dir}(1/5)$, the latent space could be organized such that each digit is mapped to a vertex, with little mass in the center. We found that in dimension 10 this is seldom the case, as multiple vertices can be taken by the same digit to model different styles, while other digits share the same vertex. We therefore experiment with a 16-dimensional $\text{Dir}(1/5)$, which yields more disconnected clusters

(3b); the effect is evident when showing the prior and the aggregated posterior that tries to cover it (3c). Figure 3d (leftmost and rightmost columns) shows that every digit 0 – 9 is indeed represented on one of the 16 vertices, while some digits are present with multiple styles, e.g. the 7. The central samples in the Figure are the interpolations obtained by sampling on edges connecting vertices – no real data is autoencoded. Samples from the vertices appear much crisper than other prior samples (3e), a sign of mismatch between prior and aggregated posterior on areas with lower probability mass. Finally, we could even learn the Dirichlet hyperparameter(s) with a reparametrization trick (Figurnov et al., 2018) and let the data inform the model on the best prior.

7 CONCLUSION

We introduced a generative model built on the principles of Optimal Transport. Working with empirical Wasserstein distances and deterministic networks provides us with a flexible likelihood-free framework for latent variable modeling.

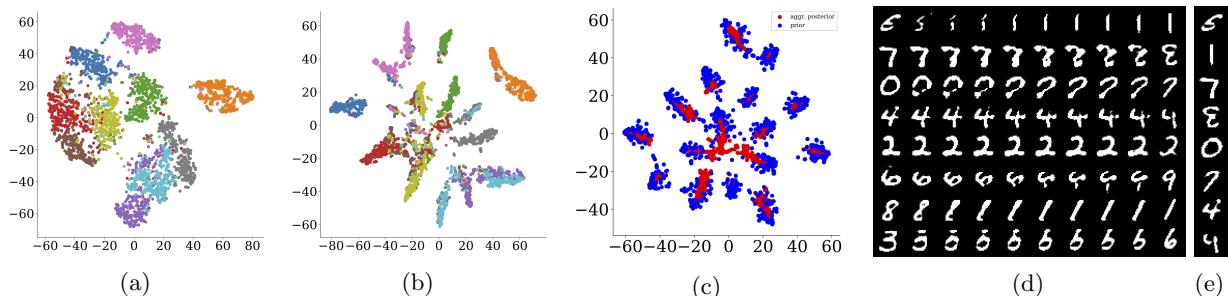


Figure 3: t-SNEs of SAE latent spaces on MNIST: a) 10-dim $\text{Dir}(1/2)$ and b) 16-dim $\text{Dir}(1/5)$ priors. For the latter: c) aggregated posterior (red) vs. prior (blue), d) vertices interpolation and e) samples from the prior.

References

- Adams, R. P. and Zemel, R. S. (2011). Ranking via Sinkhorn Propagation. *arXiv preprint arXiv:1106.1925*.
- Alemi, A., Poole, B., Fischer, I., Dillon, J., Saurous, R. A., and Murphy, K. (2018). Fixing a Broken ELBO. In *ICML*.
- Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. (2017). Deep variational information bottleneck. In *ICLR*.
- Altschuler, J., Weed, J., and Rigollet, P. (2017). Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration. In *NIPS*.
- Ambrogioni, L., Güçlü, U., GüclüTÜRK, Y., Hinne, M., van Gerven, M. A., and Maris, E. (2018). Wasserstein Variational Inference. In *NIPS*.
- Balan, R., Singh, M., and Zou, D. (2017). Lipschitz properties for deep convolutional networks. *arXiv preprint arXiv:1701.05217*.
- Bińkowski, M., Sutherland, D. J., Arbel, M., and Gretton, A. (2018). Demystifying MMD GANs. *arXiv preprint arXiv:1801.01401*.
- Bojanowski, P. and Joulin, A. (2017). Unsupervised Learning by Predicting Noise. In *ICML*.
- Bousquet, O., Gelly, S., Tolstikhin, I., Simon-Gabriel, C.-J., and Schoelkopf, B. (2017). From optimal transport to generative modeling: the VEGAN cookbook. *arXiv preprint arXiv:1705.07642*.
- Chizat, L., Peyré, G., Schmitzer, B., and Vialard, F.-X. (2016). Scaling Algorithms for Unbalanced Transport Problems. *arXiv preprint arXiv:1607.05816*.
- Cominetti, R. and San Martín, J. (1994). Asymptotic analysis of the exponential penalty trajectory in linear programming. *Math. Programming*, 67(2, Ser. A):169–187.
- Courty, N., Flamary, R., and Tuia, D. (2014). Domain adaptation with regularized optimal transport. In *KDD*.
- Cuturi, M. (2013). Sinkhorn Distances: Lightspeed Computation of Optimal Transport. In *NIPS*.
- Davidson, T. R., Falorsi, L., De Cao, N., Kipf, T., and Tomczak, J. M. (2018). Hyperspherical Variational Auto-Encoders. In *UAI*.
- Donahue, J., Krähenbühl, P., and Darrell, T. (2017). Adversarial feature learning. In *ICLR*.
- Feydy, J., Séjourné, T., Vialard, F.-X., Amari, S.-i., Trouvé, A., and Peyré, G. (2018). Interpolating between Optimal Transport and MMD using Sinkhorn Divergences. *arXiv preprint arXiv:1810.08278*.
- Figurnov, M., Mohamed, S., and Mnih, A. (2018). Implicit Reparameterization Gradients. In *NIPS*.
- Frogner, C., Zhang, C., Mobahi, H., Araya, M., and Poggio, T. A. (2015). Learning with a Wasserstein Loss. In *NIPS*.
- Genevay, A., Chizat, L., Bach, F., Cuturi, M., Peyré, G., et al. (2019). Sample Complexity of Sinkhorn Divergences. In *AISTATS*.
- Genevay, A., Peyré, G., Cuturi, M., et al. (2018). Learning Generative Models with Sinkhorn Divergences. In *AISTATS*.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative Adversarial Nets. In *NIPS*.
- Gretton, A., Borgwardt, K. M., Rasch, M., Schölkopf, B., and Smola, A. J. (2007). A kernel method for the two-sample-problem. In *NIPS*.
- Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773.
- Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*.
- Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2017). β -VAE: Learning basic visual concepts with a constrained variational framework. In *ICLR*.
- Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507.
- Hoffman, M. D. and Johnson, M. J. (2016). ELBO surgery: yet another way to carve up the variational evidence lower bound. In *Workshop in Advances in Approximate Bayesian Inference, NIPS*.
- Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257.
- Huang, G., Guo, C., Kusner, M. J., Sun, Y., Sha, F., and Weinberger, K. Q. (2016). Supervised word mover’s distance. In *NIPS*.
- Kingma, D. P. and Welling, M. (2013). Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*.

- Kolouri, S., Martin, C. E., and Rohde, G. K. (2018). Sliced-Wasserstein Autoencoder: An Embarrassingly Simple Generative Model. *arXiv preprint arXiv:1804.01947*.
- Kuhn, H. W. (1955). The Hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.
- Linderman, S. W., Mena, G. E., Cooper, H., Paninski, L., and Cunningham, J. P. (2018). Reparameterizing the Birkhoff Polytope for Variational Permutation Inference. *AISTATS*.
- Luise, G., Rudi, A., Pontil, M., and Ciliberto, C. (2018). Differential Properties of Sinkhorn Approximation for Learning with Wasserstein Distance. In *NIPS*.
- Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605.
- Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., and Frey, B. (2015). Adversarial autoencoders. *ICLR*.
- Mohamed, S. and Lakshminarayanan, B. (2016). Learning in implicit generative models. In *ICML*.
- Muzellec, B., Nock, R., Patrini, G., and Nielsen, F. (2017). Tsallis Regularized Optimal Transport and Ecological Inference. In *AAAI*.
- Noroozi, M. and Favaro, P. (2016). Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*.
- Noroozi, M., Pirsiavash, H., and Favaro, P. (2017). Representation learning by learning to count. *CVPR*.
- Peyré, G. and Cuturi, M. (2018). Computational Optimal Transport. *arXiv preprint arXiv:1803.00567*.
- Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. *ICML*.
- Rosca, M., Lakshminarayanan, B., and Mohamed, S. (2018). Distribution Matching in Variational Inference. *arXiv preprint arXiv:1802.06847*.
- Rubenstein, P. K., Schoelkopf, B., and Tolstikhin, I. (2018). Wasserstein Auto-Encoders: Latent Dimensionality and Random Encoders. In *ICLR workshop*.
- Santa Cruz, R., Fernando, B., Cherian, A., and Gould, S. (2017). Deeppermnet: Visual permutation learning. In *CVPR*.
- Schmitzer, B. (2016). Stabilized sparse scaling algorithms for entropy regularized transport problems. *arXiv preprint arXiv:1610.06519*.
- Sinkhorn, R. (1964). A relationship between arbitrary positive matrices and doubly stochastic matrices. *Ann. Math. Statist.*, 35.
- Sriperumbudur, B. K., Fukumizu, K., and Lanckriet, G. R. G. (2011). Universality, characteristic kernels and RKHS embedding of measures. *Journal of Machine Learning Research*, 12(Jul):2389–2410.
- Sriperumbudur, B. K., Gretton, A., Fukumizu, K., Schölkopf, B., and Lanckriet, G. R. G. (2010). Hilbert space embeddings and metrics on probability measures. *Journal of Machine Learning Research*, 11(Apr):1517–1561.
- Tolstikhin, I., Bousquet, O., Gelly, S., and Schoelkopf, B. (2018). Wasserstein Auto-Encoders. In *ICLR*.
- Tomczak, J. M. and Welling, M. (2017). VAE with a VampPrior. In *AISTATS*.
- Villani, C. (2008). *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg.
- Weed, J. (2018). An explicit analysis of the entropic penalty in linear programming. *arXiv preprint arXiv:1806.01879*.
- Weed, J. and Bach, F. (2017). Sharp asymptotic and finite-sample rates of convergence of empirical measures in Wasserstein distance. In *NIPS*.