Spherical Sliced-Wasserstein

Anonymous Author(s)
Affiliation
Address
email

Abstract

Many variants of the Wasserstein distance have been introduced to reduce its original computational burden. In particular the Sliced-Wasserstein distance (SW), which leverages one-dimensional projections for which a closed-form solution of the Wasserstein distance is available, has received a lot of interest. Yet, it is restricted to data living in Euclidean spaces, while the Wasserstein distance has been studied and used recently on manifolds. We focus more specifically on the sphere, for which we define a novel SW discrepancy, which we call spherical Sliced-Wasserstein, making a first step towards defining SW discrepancies on manifolds. Our construction is notably based on closed-form solutions of the Wasserstein distance on the circle, together with a new spherical Radon transform. Along with efficient algorithms and the corresponding implementations, we illustrate its properties in several machine learning use cases where spherical representations of data are at stake: density estimation on the sphere, variational inference or hyperspherical auto-encoders.

1 Introduction

Optimal transport (OT) [101] has received a lot of attention in machine learning in the past few years. As it allows to compare distributions with metrics, it has been used for different tasks such as domain adaptation [25] or generative models [8], to name a few. The most classical distance used in OT is the Wasserstein distance. However, calculating it can be computationally expensive. Hence, several variants were proposed to alleviate the computational burden, such as the entropic regularization [27, 97], minibatch OT [36] or the sliced-Wasserstein distance (SW) for distributions supported on Euclidean spaces [90].

Although embedded in larger dimensional Euclidean spaces, data generally lie in practice on manifolds [37]. A simple manifold, but with lots of practical applications, is the hypersphere S^{d-1}. Several types of data are by essence spherical: a good example is found in directional data [73, 87] for which dedicated machine learning solutions are being developed [98], but other applications concern for instance geophysical data [33], meteorology [11], cosmology [86] or extreme value theory for the estimation of spectral measures [45]. Remarkably, in a more abstract setting, considering hyperspherical latent representations of data is becoming more and more common (*e.g.* [29, 72, 109]). For example, in the context of variational autoencoders [59], using priors on the sphere has been demonstrated to be beneficial [29]. Also, in the context of self-supervised learning (SSL), where one wants to learn discriminative representations in an unsupervised way, the hypersphere is usually considered for the latent representation [20, 21, 44, 104, 108]. It is thus of primary importance to develop machine learning tools that accommodate well with this specific geometry.

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

The OT theory on manifolds is well developed [39, 75, 101] and several works started to use it in practice, with a focus mainly on the approximation of OT maps. For example, Cohen et al. [24], Rezende and Racanière [91] approximate the OT map to define normalizing flows on Riemannian manifolds, Cui et al. [26], Hamfeldt and Turnquist [46, 47] derive algorithms to approximate the OT map on the sphere, Alvarez-Melis et al. [5], Hoyos-Idrobo [52] learn the transport map on hyperbolic spaces. However, the computational bottleneck to compute the Wasserstein distance on such spaces remains, and, as underlined in the conclusion of [76], defining SW distances on manifolds would be of much interest.

Contributions. Therefore, by leveraging properties of the Wasserstein distance on the circle [89], we define the first, to the best of our knowledge, SW discrepancy on a non trivial manifold, namely the sphere S^{d-1}, and hence we make a first step towards defining SW distances on Riemannian manifolds. We make connections with a new spherical Radon transform and analyze some of its properties. We discuss the underlying algorithmic procedure, and notably provide an efficient implementation when computing the discrepancy against a uniform distribution. Then, we show that we can use this discrepancy on different tasks such as density estimation, variational inference or generative modeling.

2 Background

The aim of this paper is to define a Sliced-Wasserstein discrepancy on the hypersphere $S^{d-1} = \{x \in \mathbb{R}^d,\ \|x\|_2 = 1\}$. Therefore, in this section, we introduce the Wasserstein distance on manifolds and the classical SW distance on $\mathbb{R}^d$.

2.1 Wasserstein distance

Since we are interested in defining a SW discrepancy on the sphere, we start by introducing the Wasserstein distance on a Riemannian manifold M endowed with the Riemannian distance d. We refer to [39, 101] for more details.

Let $p \geq 1$ and $\mu, \nu \in \mathcal{P}_p(M) = \{\mu \in \mathcal{P}(M),\ \int_M d^p(x, x_0)\, \mathrm{d}\mu(x) < \infty$ for some $x_0 \in M\}$. Then, the p-Wasserstein distance between μ and ν is defined as

$$W_p^p(\mu, \nu) = \inf_{\gamma \in \Pi(\mu,\nu)} \int_{M \times M} d^p(x, y)\, \mathrm{d}\gamma(x, y), \tag{1}$$

where $\Pi(\mu, \nu) = \{\gamma \in \mathcal{P}(M \times M),\ \forall A \subset M,\ \gamma(M \times A) = \nu(A)$ and $\gamma(A \times M) = \mu(A)\}$ denotes the set of couplings.

For discrete probability measures, the Wasserstein distance can be computed using linear programs [88]. However, these algorithms have a $O(n^3 \log n)$ complexity *w.r.t.* the number of samples n which is computationally intensive. Therefore, a whole literature consists of defining alternative discrepancies which are cheaper to compute. On Euclidean spaces, one of them is the Sliced-Wasserstein distance.

2.2 Sliced-Wasserstein distance

On $M = \mathbb{R}^d$ with $d(x, y) = \|x - y\|_p^p$, a more attractive distance is the Sliced-Wasserstein (SW) distance. This distance relies on the appealing fact that for one dimensional measures $\mu, \nu \in \mathcal{P}(\mathbb{R})$, we have the following closed-form [88, Remark 2.30]

$$W_p^p(\mu, \nu) = \int_0^1 \left|F_\mu^{-1}(u) - F_\nu^{-1}(u)\right|^p \mathrm{d}u, \tag{2}$$

where F_μ^{-1} (resp. F_ν^{-1}) is the quantile function of μ (resp. ν). From this property, Bonnotte [16], Rabin et al. [90] defined the SW distance as

$$\forall \mu, \nu \in \mathcal{P}_p(\mathbb{R}^d),\ SW_p^p(\mu, \nu) = \int_{S^{d-1}} W_p^p(P_\#^\theta \mu, P_\#^\theta \nu)\, \mathrm{d}\lambda(\theta), \tag{3}$$

where $P^\theta(x) = \langle x, \theta \rangle$, λ is the uniform distribution on S^{d-1} and for any Borel set $A \in \mathcal{B}(\mathbb{R}^d)$, $P^\theta_\# \mu(A) = \mu((P^\theta)^{-1}(A))$.

This distance can be approximated efficiently by using a Monte-Carlo approximation [77], and amounts to a complexity of $O(Ln \log n)$ where L denotes the number of projections used for the Monte-Carlo approximation and n the number of samples.

SW can also be written through the Radon transform [15]. Let $f \in L^1(\mathbb{R}^d)$, then the Radon transform $R : L^1(\mathbb{R}^d) \to L^1(\mathbb{R} \times S^{d-1})$ is defined as [49]

$$\forall \theta \in S^{d-1},\ \forall t \in \mathbb{R},\ Rf(t, \theta) = \int_{\mathbb{R}^d} f(x) \mathbb{1}_{\{\langle x, \theta \rangle = t\}} \mathrm{d}x. \tag{4}$$

Its dual $R^* : C_0(\mathbb{R} \times S^{d-1}) \to C_0(\mathbb{R}^d)$ (also known as back-projection operator), where C_0 denotes the set of continuous functions that vanish at infinity, satisfies for all f, g, $\langle Rf, g \rangle_{\mathbb{R} \times S^{d-1}} = \langle f, R^* g \rangle_{\mathbb{R}^d}$ and can be defined as [13, 15]

$$\forall g \in C_0(\mathbb{R} \times S^{d-1}), \forall x \in \mathbb{R}^d,\ R^* g(x) = \int_{S^{d-1}} g(\langle x, \theta \rangle, \theta)\, \mathrm{d}\theta. \tag{5}$$

Therefore, by duality, we can define the Radon transform of a measure $\mu \in \mathcal{M}(\mathbb{R}^d)$ as the measure $R\mu \in \mathcal{M}(\mathbb{R} \times S^{d-1})$ such that for all $g \in C_0(\mathbb{R} \times S^{d-1})$, $\langle R\mu, g \rangle_{\mathbb{R} \times S^{d-1}} = \langle \mu, R^* g \rangle_{\mathbb{R}^d}$. Since $R\mu$ is a measure on the product space $\mathbb{R} \times S^{d-1}$, we can disintegrate it *w.r.t.* λ, the uniform measure on S^{d-1} [6], as $R\mu = \lambda \otimes K$ with K a probability kernel on $S^{d-1} \times \mathcal{B}(\mathbb{R})$, *i.e.* for all $\theta \in S^{d-1}$, $K(\theta, \cdot)$ is a probability on $\mathbb{R}$, for any Borel set $A \in \mathcal{B}(\mathbb{R})$, $K(\cdot, A)$ is measurable, and

$$\forall \phi \in C(\mathbb{R} \times S^{d-1}),\ \int_{\mathbb{R} \times S^{d-1}} \phi(t, \theta) \mathrm{d}(R\mu)(t, \theta) = \int_{S^{d-1}} \int_{\mathbb{R}} \phi(t, \theta) K(\theta, \mathrm{d}t) \mathrm{d}\lambda(\theta). \tag{6}$$

By Proposition 6 in [15], we have that for λ-almost every $\theta \in S^{d-1}$, $(R\mu)^\theta = P^\theta_\# \mu$ where we denote $K(\theta, \cdot) = (R\mu)^\theta$. Therefore, we have

$$\forall \mu, \nu \in \mathcal{P}_p(\mathbb{R}^d),\ SW_p^p(\mu, \nu) = \int_{S^{d-1}} W_p^p\big((R\mu)^\theta, (R\mu)^\theta\big)\, \mathrm{d}\lambda(\theta). \tag{7}$$

Variants of SW have been defined in recent works, either by integrating *w.r.t.* different distributions [32, 81], by projecting on $\mathbb{R}$ using different projections [79, 80] or Radon transforms [22, 61], or by projecting on subspaces of higher dimensions [53, 68, 69, 85].

3 A Sliced-Wasserstein discrepancy on the sphere

Our goal here is to define a sliced-Wasserstein distance on the sphere S^{d-1}. To that aim, we proceed analogously to the classical Euclidean space. We first rely on the nice properties of the Wasserstein distance on the circle [89] and then propose to project distributions lying on the sphere to great circles. Hence, circles play the role of the real line for the hypersphere. In this section, we first describe the OT problem on the circle, then we define a sliced-Wasserstein discrepancy on the sphere and discuss some of its properties. Notably, we derive a new spherical Radon transform which is linked to our newly defined spherical SW. We refer to Appendix A for the proofs.

3.1 Optimal transport on the circle

On the circle $S^1 = \mathbb{R}/\mathbb{Z}$ equipped with the geodesic distance d_{S^1}, an appealing formulation of the Wasserstein distance is available [31]. First, let us parametrize S^1 by $[0, 1[$, then the geodesic distance can be written as [89], for all $x, y \in [0, 1[$, $d_{S^1}(x, y) = \min(|x - y|, 1 - |x - y|)$. Then, for the cost function $c(x, y) = h(d(x, y))$ with $h : \mathbb{R} \to \mathbb{R}^+$ an increasing convex function, the Wasserstein distance between $\mu \in \mathcal{P}(S^1)$ and $\nu \in \mathcal{P}(S^1)$ can be written as

$$W_c(\mu, \nu) = \inf_{\alpha \in \mathbb{R}} \int_0^1 h\big(|F_\mu^{-1}(t) - (F_\nu - \alpha)^{-1}(t)|\big)\, \mathrm{d}t, \tag{8}$$

where $F_\mu : [0,1[\to [0,1]$ denotes the cumulative distribution function (cdf) of μ, F_μ^{-1} its quantile function and α is a shift parameter. The optimization problem over the shifted cdf $F_\nu - \alpha$ can be seen as looking for the best "cut" (or origin) of the circle into the real line because of the 1-periodicity. Indeed, the proof of this result for discrete distributions in [89] consists in cutting the circle at the optimal point and wrapping it around the real line, for which the optimal transport map is the increasing rearrangement $F_\nu^{-1} \circ F_\mu$ which can be obtained for discrete distributions by sorting the points [88].

Rabin et al. [89] showed that the minimization problem is convex and coercive in the shift parameter and Delon et al. [31] derived a binary search algorithm to find it. For the particular case of $h = \mathrm{Id}$, it can further be shown [19, 106] that

$$W_1(\mu,\nu) = \inf_{\alpha \in \mathbb{R}} \int_0^1 |F_\mu(t) - F_\nu(t) - \alpha| \, \mathrm{d}t. \tag{9}$$

In this case, we know exactly the minimum which is attained at the level median [54]. For $f : [0,1[\to \mathbb{R}$,

$$\mathrm{LevMed}(f) = \min \left\{ \operatorname*{argmin}_{\alpha \in \mathbb{R}} \int_0^1 |f(t) - \alpha| \mathrm{d}t \right\} = \inf \left\{ t \in \mathbb{R},\ \beta(\{x \in [0,1[,\ f(x) \leq t\}) \geq \frac{1}{2} \right\}, \tag{10}$$

where β is the Lebesgue measure. Therefore, we also have

$$W_1(\mu,\nu) = \int_0^1 |F_\mu(t) - F_\nu(t) - \mathrm{LevMed}(F_\mu - F_\nu)| \, \mathrm{d}t. \tag{11}$$

Since we know the minimum, we do not need the binary search and we can approximate the integral very efficiently as we only need to sort the samples to compute the level median and the cdfs.

Another interesting setting in practice is to compute W_2, *i.e.* with $h(x) = x^2$, *w.r.t.* a uniform distribution ν on the circle. We derive here the optimal shift $\hat{\alpha}$ for the Wasserstein distance between μ an arbitrary distribution on S^1 and ν. We also provide a closed-form when μ is a discrete distribution.

Proposition 1. *Let $\mu \in \mathcal{P}_2(S^1)$ and $\nu = \mathrm{Unif}(S^1)$. Then,*

$$W_2^2(\mu,\nu) = \int_0^1 |F_\mu^{-1}(t) - t - \hat{\alpha}|^2 \, \mathrm{d}t \quad \textit{with} \quad \hat{\alpha} = \int x \, \mathrm{d}\mu(x) - \frac{1}{2}. \tag{12}$$

In particular, if $x_1 < \cdots < x_n$ and $\mu_n = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$, then

$$W_2^2(\mu_n,\nu) = \frac{1}{n}\sum_{i=1}^n x_i^2 - \Big(\frac{1}{n}\sum_{i=1}^n x_i\Big)^2 + \frac{1}{n^2}\sum_{i=1}^n (n+1-2i)x_i + \frac{1}{12}. \tag{13}$$

This proposition offers an intuitive interpretation: the optimal cut point between an empirical and a uniform distributions is the antipodal point of the circular mean of the discrete samples. Moreover, a very efficient algorithm can be derived from this property, as it solely requires a sorting operation to compute the order statistics of the samples.

3.2 Definition of SW on the sphere

On the hypersphere, the counterpart of straight lines are the great circles, which correspond to the geodesics. Moreover, we can compute the Wasserstein distance on the circle fairly efficiently. Hence, to define a sliced-Wasserstein discrepancy on this manifold, we propose, analogously to the classical SW distance, to project measures on great circles. The most natural way to project points from S^{d-1} to a great circle C is to use the geodesic projection [41, 56] defined as

$$\forall x \in S^{d-1},\ P^C(x) = \operatorname*{argmin}_{y \in C} \ d_{S^{d-1}}(x,y), \tag{14}$$

where $d_{S^{d-1}}(x,y) = \arccos(\langle x,y \rangle)$ is the geodesic distance. See Figure 1 for an illustration of the geodesic projection on a great circle. Note that the projection is unique for almost every x (see [9, Proposition 4.2] and Appendix B.1).

Great circles can be obtained by intersecting S^{d-1} with a 2-dimensional plane [57]. Therefore, to average over all great circles, we propose to integrate over the Grassmann manifold $\mathcal{G}_{d,2} = \{E \subset \mathbb{R}^d,\ \dim(E) = 2\}$ [2, 10] and then to project the distributions onto the intersection with the hypersphere. Since the Grassmannian is not very practical, we consider the identification using the set of rank 2 projectors:

$$\mathcal{G}_{d,2} = \{P \in \mathbb{R}^{d\times d},\ P^T = P,\ P^2 = P,\ \mathrm{Tr}(P) = 2\} = \{UU^T,\ U \in \mathbb{V}_{d,2}\}, \tag{15}$$

where $\mathbb{V}_{d,2} = \{U \in \mathbb{R}^{d\times 2},\ U^T U = I_2\}$ is the Stiefel manifold [10].

Finally, we can define the Spherical Sliced-Wasserstein distance (SSW) as

$$\forall p \geq 1, \forall \mu, \nu \in \mathcal{P}_p(S^{d-1}),\ SSW_p^p(\mu,\nu) = \int_{\mathbb{V}_{d,2}} W_p^p(P^U_\# \mu, P^U_\# \nu)\, \mathrm{d}\sigma(U), \tag{16}$$

where σ is the uniform distribution over the Stiefel manifold $\mathbb{V}_{d,2}$, P^U is the geodesic projection on the great circle generated by U and then projected on S^1, *i.e.*

$$\forall U \in \mathbb{V}_{d,2}, \forall x \in S^{d-1},\ P^U(x) = U^T \underset{y \in \mathrm{span}(UU^T)\cap S^{d-1}}{\mathrm{argmin}} d_{S^{d-1}}(x,y) = \underset{z\in S^1}{\mathrm{argmin}}\ d_{S^{d-1}}(x, Uz), \tag{17}$$

and the Wasserstein distance is defined with the geodesic distance d_{S^1}.

Moreover, we can derive a closed form expression which will be very useful in practice:

Lemma 1. *Let $U \in \mathbb{V}_{d,2}$ and $x \in S^{d-1}$, then*

$$P^U(x) = \frac{U^T x}{\|U^T x\|_2}. \tag{18}$$

Hence, we notice from this expression of the projection that we recover almost the same formula as Lin et al. [68] but with an additional ℓ^2 normalization which projects the data on the circle. As in [68], we could project on a higher dimensional subsphere by integrating over $\mathbb{V}_{d,k}$ with $k \geq 2$. However, we would lose the computational efficiency provided by the properties of the Wasserstein distance on the circle.

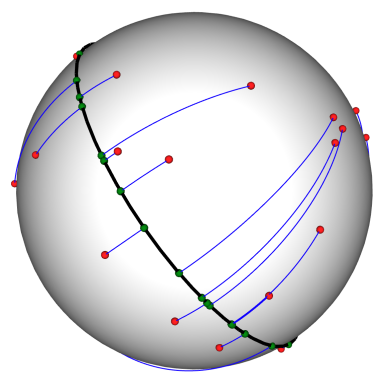


Figure 1: Illustration of the geodesic projections on a great circle (in black). In red, random points sampled on the sphere. In green the projections and in blue the trajectories.

3.3 A Spherical Radon Transform

As for the classical SW distance, we can derive a second formulation using a Radon transform. Let $f \in L^1(S^{d-1})$, we define a spherical Radon transform $\tilde{R} : L^1(S^{d-1}) \to L^1(S^1 \times \mathbb{V}_{d,2})$ as

$$\forall z \in S^1,\ \forall U \in \mathbb{V}_{d,2},\ \tilde{R}f(z,U) = \int_{S^{d-1}} f(x)\mathbb{1}_{\{z = P^U(x)\}} \mathrm{d}x. \tag{19}$$

This is basically the same formulation as the classical Radon transform [49, 78] where we replaced the real line coordinate t by the coordinate on the circle z and the projection is the geodesic one which is well suited to the sphere. This transform is actually new since we integrate over different sets compared to existing works on spherical Radon transforms.

Then, analogously to the classical Radon transform, we can define the back-projection operator $\tilde{R}^* : C_0(S^1 \times \mathbb{V}_{d,2}) \to C_b(S^{d-1})$, $C_b(S^{d-1})$ being the space of continuous bounded functions, for $g \in C_0(S^1 \times \mathbb{V}_{d,2})$ as

$$\forall x \in S^{d-1},\ \tilde{R}^* g(x) = \int_{\mathbb{V}_{d,2}} g(P^U(x), U)\, \mathrm{d}\sigma(U). \tag{20}$$

Proposition 2. *$\tilde{R}^*$ is the dual operator of $\tilde{R}$,* i.e. *for all $f \in L^1(S^{d-1})$, $g \in C_0(S^1 \times \mathbb{V}_{d,2})$,*

$$\langle \tilde{R}f, g\rangle_{S^1\times\mathbb{V}_{d,2}} = \langle f, \tilde{R}^* g\rangle_{S^{d-1}}. \tag{21}$$

Now that we have a dual operator, we can also define the Radon transform of a measure $\mu \in \mathcal{M}(S^{d-1})$ by duality [13, 15] as the measure $\tilde{R}\mu$ satisfying

$$\forall g \in C_0(S^1 \times \mathbb{V}_{d,2}),\ \int_{S^1\times\mathbb{V}_{d,2}} g(z,U)\,\mathrm{d}(\tilde{R}\mu)(z,U) = \int_{S^{d-1}} \tilde{R}^* g(x)\,\mathrm{d}\mu(x). \tag{22}$$

Since $\tilde{R}\mu$ is a measure on the product space $S^1 \times \mathbb{V}_{d,2}$, $\tilde{R}\mu$ can be disintegrated [6, Theorem 5.3.1] *w.r.t.* σ as $\tilde{R}\mu = \sigma \otimes K$ where K is a probability kernel on $\mathbb{V}_{d,2} \times \mathcal{S}^1$ with $\mathcal{S}^1$ the Borel σ-field of S^1. We will denote for σ-almost every $U \in \mathbb{V}_{d,2}$, $(\tilde{R}\mu)^U = K(U,\cdot)$ the conditional probability.

Proposition 3. *Let $\mu \in \mathcal{M}(S^{d-1})$, then for σ-almost every $U \in \mathbb{V}_{d,2}$, $(\tilde{R}\mu)^U = P^U_{\#}\mu$.*

Finally, we can write SSW (16) using this Radon transform:

$$\forall \mu,\nu \in \mathcal{P}_p(S^{d-1}),\ SSW_p^p(\mu,\nu) = \int_{\mathbb{V}_{d,2}} W_p^p\big((\tilde{R}\mu)^U, (\tilde{R}\nu)^U\big)\,\mathrm{d}\sigma(U). \tag{23}$$

Note that a natural way to define SW distances can be through already known Radon transforms using the formulation (23). It is for example what was done in [61] using generalized Radon transforms [35, 51] to define generalized SW distances, or in [22] with the spatial Radon transform. However, for known spherical Radon transforms [1, 7] such as the Minkowski-Funk transform [28] or more generally the geodesic Radon transform [95], there is no natural way that we know of to integrate over some product space and allowing to define a SW distance using disintegration.

As observed by Kolouri et al. [61] for the generalized SW distances (GSW), studying the injectivity of the related Radon transforms allows to study the set on which SW is actually a distance. While the classical Radon transform integrates over hyperplanes of $\mathbb{R}^d$, the generalized Radon transform over hypersurfaces [61] and the Minkowski-Funk transform over "big circles", *i.e.* the intersection between a hyperplane and S^{d-1} [96], the set of integration here is a half of a big circle. Hence, $\tilde{R}$ is related to the hemispherical transform [94] on S^{d-2}. We refer to Appendix A.6 for more details on the links with the hemispherical transform. Using these connections, we can derive the kernel of $\tilde{R}$ as the set of even measures which are null over all hyperplanes intersected with S^{d-1}.

Proposition 4. $\ker(\tilde{R}) = \{\mu \in \mathcal{M}_{\text{even}}(S^{d-1}),\ \forall H \in \mathcal{G}_{d,d-1},\ \mu(H \cap S^{d-1}) = 0\}$ *where $\mu \in \mathcal{M}_{\text{even}}$ if for all $f \in C(S^{d-1})$, $\langle \mu, f\rangle = \langle \mu, f_+\rangle$ with $f_+(x) = (f(x) + f(-x))/2$ for all x.*

We leave for future works checking whether this set is null or not. Hence, we conclude here that SSW is a pseudo-distance, but a distance on the sets of injectivity of $\tilde{R}$ [4].

Proposition 5. *Let $p \geq 1$, SSW_p is a pseudo-distance on $\mathcal{P}_p(S^{d-1})$.*

4 Implementation

In practice, we approximate the distributions with empirical approximations and, as for the classical SW distance, we rely on the Monte-Carlo approximation of the integral on $\mathbb{V}_{d,2}$. We first need to sample from the uniform distribution $\sigma \in \mathcal{P}(\mathbb{V}_{d,2})$. This can be done by first constructing $Z \in \mathbb{R}^{d\times 2}$ by drawing each of its component from the standard normal distribution $\mathcal{N}(0,1)$ and then applying the QR decomposition [69]. Once we have $(U_\ell)_{\ell=1}^L \sim \sigma$, we project the samples on the circle S^1 by applying Lemma 1 and we compute the coordinates on the circle using the $\mathrm{atan2}$ function. Finally, we can compute the Wasserstein distance on the circle by either applying the binary search algorithm of [31] or the level median formulation (11) for SSW_1. In the particular case in which we want to compute SSW_2 between a measure μ and the uniform measure on the sphere $\nu = \mathrm{Unif}(S^{d-1})$, we can use the appealing fact that the projection of ν on the circle is uniform, *i.e.* $P^U_{\#}\nu = \mathrm{Unif}(S^1)$ (particular case of Theorem 3.1 in [56], see Appendix B.3). Hence, we can use the Proposition 1 to compute W_2, which allows a very efficient implementation either by the closed-form (13) or approximation by rectangle method of (12). This will be of particular interest for applications in Section 5 such as autoencoders. We sum up the procedure in Algorithm 1.

Algorithm 1 SSW

Input: $(x_i)_{i=1}^n \sim \mu$, $(y_j)_{j=1}^m \sim \nu$, L the number of projections, p the order
for $\ell = 1$ **to** L **do**
  Draw a random matrix $Z \in \mathbb{R}^{d\times 2}$ with for all $i, j,\ Z_{i,j} \sim \mathcal{N}(0,1)$
  $U = \mathrm{QR}(Z) \sim \sigma$
  Project on S^1 the points: $\forall i, j,\ \hat{x}_i^\ell = \frac{U^T x_i}{\|U^T x_i\|_2}$, $\hat{y}_j^\ell = \frac{U^T y_j}{\|U^T y_j\|_2}$
  Compute the coordinates on the circle S^1: $\forall i, j,\ \tilde{x}_i^\ell = (\pi + \mathrm{atan2}(-x_{i,2}, -x_{i,1}))/(2\pi)$, $\tilde{y}_j^\ell = (\pi + \mathrm{atan2}(-y_{j,2}, -y_{j,1}))/(2\pi)$
  Compute $W_p^p(\frac{1}{n}\sum_{i=1}^n \delta_{\tilde{x}_i^\ell}, \frac{1}{m}\sum_{j=1}^m \delta_{\tilde{y}_j^\ell})$ by binary search or (11) for $p = 1$
end for
Return $SSW_p^p(\mu,\nu) \approx \frac{1}{L}\sum_{\ell=1}^L W_p^p(\frac{1}{n}\sum_{i=1}^n \delta_{\tilde{x}_i^\ell}, \frac{1}{m}\sum_{j=1}^m \delta_{\tilde{y}_j^\ell})$

Complexity. Let us note n (resp. m) the number of samples of μ (resp. ν), and L the number of projections. First, we need to compute the QR factorization of L matrices of size $d \times 2$. This can be done in $O(Ld)$ by using *e.g.* Householder reflections [43, Chapter 5.2] or the Scharwz-Rutishauser algorithm [42]. Projecting the points on S^1 by Lemma 1 is in $O((n+m)dL)$ since we need to compute $L(n+m)$ products between $U_\ell^T \in \mathbb{R}^{2\times d}$ and $x \in \mathbb{R}^d$. For the binary search or particular case formula (11) and (13), we need first to sort the points. But the binary search also adds a cost of $O((n+m)\log(\frac{1}{\epsilon}))$ to approximate the solution with precision ϵ [31] and the computation of the level median requires to sort $(n+m)$ points. Hence, for the general SSW_p, the complexity is $O(L(n+m)(d+\log(\frac{1}{\epsilon})) + Ln\log n + Lm\log m)$ versus $O(L(n+m)(d+\log(n+m)))$ for SSW_1 with the level median and $O(Ln(d+\log n))$ for SSW_2 against a uniform with the particular advantage that we do not need uniform samples in this case.

Runtime Comparison. We perform here some runtime comparisons. Using Pytorch [83], we implemented the binary search algorithm of [31] and used it with $\epsilon = 10^{-6}$. We also implemented SSW_1 using the level median formula (11) and SSW_2 against a uniform measure (12). All experiments are conducted on GPU.

On Figure 2, we compare the runtime between two distributions on S^2 between SSW, the Wasserstein distance and the entropic approximation using the Sinkhorn algorithm [27] with the geodesic distance as cost function. The distributions were approximated using $n \in \{10^2, 10^3, 10^4, 5\cdot 10^4, 10^5\}$ samples of each distribution and we report the mean over 20 computations. We use the Python Optimal Transport (POT) library [40] to compute the Wasserstein distance and the entropic approximation. For large enough batches, we observe that SSW is much faster than its Wasserstein counterpart, and it also scales better in term of memory because of the need to store the $n \times n$ cost matrix. For small batches, the computation of SSW actually takes longer because of the computation of the QR factorizations and of the projections. For bigger batches, it is bounded by the sorting operation and we recover the quasi-linear slope. Furthermore, as expected, the fastest algorithms are SSW_1 with the level median and SSW_2 against a uniform as they have a quasilinear complexity. We report in Appendix C.2 other runtimes experiments *w.r.t.* to *e.g.* the number of projections or the dimension.

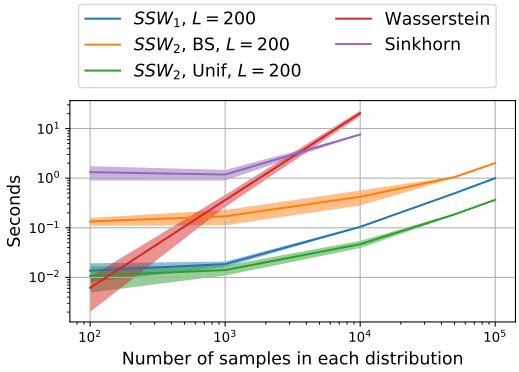
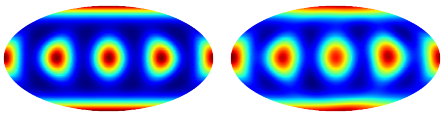
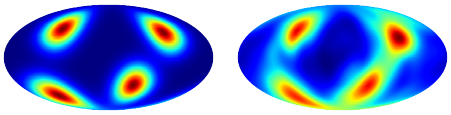


Figure 2: Runtime comparison in log-log scale between W, Sinkhorn with the geodesic distance, SSW_2 with the binary search (BS) and uniform distribution (12) and SSW_1 with formula (11) between two distributions on S^2. The time includes the calculation of the distance matrices.



(a) Target (b) KDE estimate

Figure 3: Minimization of SSW with respect to a mixture of vMF.



(a) Target distribution (b) Density learned

Figure 4: Amortized SSWVI with a normalizing flow *w.r.t.* a mixture of vMF.

5 Applications

In this section, we first illustrate the ability to approximate different distributions by minimizing SSW *w.r.t.* some target distributions on S^2. We first use distributions from which we can draw samples. Then, we use target distributions from which we know the density only up to a constant. Finally, we apply SSW for generative modeling tasks using the framework of Sliced-Wasserstein autoencoder and we show that we obtain competitive results with other Wasserstein autoencoder based methods using a prior on the hypersphere. We also add in Appendix C.6 some experiments where we use SSW in order to enforce uniformity in a contrastive self-supervised learning context.

5.1 SSW as a loss

We verify on the two first experiments that we can learn some target distribution $\nu \in \mathcal{P}(S^{d-1})$ by minimizing SSW, *i.e.* we consider the minimization problem $\text{argmin}_\mu \; SSW_p^p(\mu, \nu)$.

Gradient flow. First, we suppose that we have access to the target distribution ν through samples, *i.e.* through $\hat{\nu}_m = \frac{1}{m}\sum_{j=1}^m \delta_{y_j}$ where $(y_j)_{j=1}^m$ are i.i.d samples of ν. We choose as target distribution a mixture of 6 well separated von Mises-Fisher distributions [74]. This is a fairly challenging distribution since there are 6 modes which are not connected. We show on Figure 3 the Mollweide projection of the density approximated by a kernel density estimator for a distribution with 500 particles. To optimize directly over particles, we can either perform a Riemannian gradient descent on the sphere [3] or a projected gradient descent. We report in Appendix C.3 additional details and experiments.

Sliced-Wasserstein variational inference on the sphere. Another setting of interest is when we have access to some target distribution only up to a constant. For example in Bayesian inference, we want to sample from a posterior distribution $p(\cdot|x)$ for which the normalizing constant is costly to compute, *i.e.* we can only evaluate some function π such that $p(\cdot|x) \propto \pi$. Popular methods to solve these types of problems are MCMCs [93] or variational inference [12, 55].

Variational inference aims at approximating the target by a distribution q in some family of distributions $\mathcal{Q}$. The classical way of doing it is to minimize the Kullback-Leibler (KL) divergence. However, the KL divergence suffers from some drawbacks such as under estimating the target distribution and not being a distance. Recently, Yi and Liu [110] proposed to use the SW distance instead. The method is called Sliced-Wasserstein Variation Inference (SWVI) and relies on running at each iteration few MCMC steps and then performing gradient descent to learn the variational distribution. We refer to Appendix C.4 and Algorithm 2 for further details on the method.

In the following, we replace SW by SSW in SWVI, which we denote SSWVI, and we perform amortized variational inference on the sphere by using exponential map normalizing flows (see [92] and Appendix B.4) to learn the distribution and the Geodesic Langevin algorithm [105] as MCMC method. We use the same target as Rezende et al. [92] and we report on Figure 4 the Mollweide projection of the learned density. Since we learn to sample from a noise distribution, here the uniform distribution on the sphere, we do not have directly access to the density and we report a kernel density estimate with a Gaussian kernel. On Figure 5, we plot the evolution of the effective samples size (ESS) [34, 71] through the iterations. This indicates how well the flow matches the target. We observe that using SSW gives slightly better results, or at least comparable, than SWVI with SW.

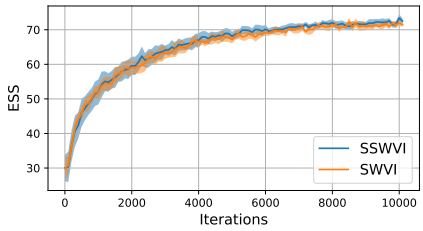
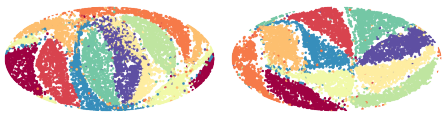


Figure 5: Comparison of the ESS between SWVI et SSWVI with the mixture target (mean over 10 runs).



(a) SWAE (b) SSWAE

Figure 6: Latent space of SWAE and SSWAE for a uniform prior on S^2.

5.2 SSW autoencoders

In this section, we use SSW to learn the latent space of autoencoders (AE). We rely on the SWAE framework introduced by Kolouri et al. [60]. Let f be some encoder and g be some decoder, denote p_Z a prior distribution, then the loss minimized in SWAE is

$$\mathcal{L}(f,g) = \int c\big(x, g(f(x))\big)\mathrm{d}\mu(x) + \lambda SW_2^2(f_{\#}\mu, p_Z), \quad (24)$$

Table 1: FID (Lower is better).

Method / Prior	Unif(S^{10})
SSWAE	**14.91 ± 0.32**
SWAE	15.18 ± 0.32
WAE-MMD IMQ	18.12 ± 0.62
WAE-MMD RBF	20.09 ± 1.42
SAE	19.39 ± 0.56
Circular GSWAE	15.01 ± 0.26

where μ is the distribution of the data for which we have access to samples. One advantage of this framework over more classical VAEs [59] is that no parametrization trick is needed here and therefore the choice of the prior is more free.

In several concomitant works, it was shown that using a prior on the hypersphere can improve the results [29, 109]. Hence, we propose in the same fashion as [60, 61, 84] to replace SW by SSW, which we denote SSWAE, and to enforce a prior on the sphere. In the following, we use the MNIST dataset [66] and we put an ℓ^2 normalization at the output of the encoder. As a prior, we use the uniform distribution on S^{10} and we compare in Table 1 the Fréchet Inception Distance (FID) [50], for 10000 samples and averaged over 5 trainings, obtained with the Wasserstein Autoencoder (WAE) [99], the classical SWAE [60], the Sinkhorn Autoencoder (SAE) [84] and circular GSWAE [61]. We observe that we obtain fairly competitive results. We add on Figure 6 the latent space obtained with a uniform prior on S^2. We refer to appendix C.5 for more details and additional experiments.

6 Conclusion and discussion

In this work, we derive a new sliced-Wasserstein discrepancy on the hypersphere, that comes with practical advantages when computing optimal transport distances on hyperspherical data. We notably showed that it is competitive or even sometimes better than other metrics defined directly on $\mathbb{R}^d$ on a variety of machine learning tasks, including density estimation, variational inference or generative models. Our work is, up to our knowledge, the first to adapt the sliced Wasserstein framework to non-trivial manifolds. The three main ingredients are: *i)* a closed-form for Wasserstein on the circle, *ii)* a closed-form solution to the projection onto great circles, and *iii)* a novel Radon transform on the Sphere. An immediate extension of this work would be to consider sliced-Wasserstein discrepancy in hyperbolic spaces, where geodesics are circular arcs as in the Poincaré disk. Beyond the generalization to other, possibly well behaved, manifolds, statistical aspects need to be examined, such as sample complexity or dependence to the hypersphere dimension. While we postulate that results comparable to the Euclidean case might be reached, the fact that the manifold is closed might bring interesting differences and justify further use of this type of discrepancies rather than their Euclidean counterparts.

References

[1] Ahmed Abouelaz and Radouan Daher. Sur la transformation de radon de la sphère S^d. *Bulletin de la Société Mathématique de France*, 121(3):353–382, 1993.

[2] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. Riemannian geometry of grassmann manifolds with a view on algorithmic computation. *Acta Applicandae Mathematica*, 80(2): 199–220, 2004.

[3] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. Optimization algorithms on matrix manifolds. In *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2009.

[4] Mark L Agranovskyt and Eric Todd Quintott. Injectivity of the spherical mean operator and related problems. *Complex analysis, harmonic analysis and applications*, 347:12, 1996.

[5] David Alvarez-Melis, Youssef Mroueh, and Tommi Jaakkola. Unsupervised hierarchy matching with optimal transport over hyperbolic spaces. In *International Conference on Artificial Intelligence and Statistics*, pages 1606–1617. PMLR, 2020.

[6] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré. *Gradient flows: in metric spaces and in the space of probability measures*. Springer Science & Business Media, 2005.

[7] Yuri A Antipov, Ricardo Estrada, and Boris Rubin. Inversion formulas for the spherical means in constant curvature spaces. *arXiv preprint arXiv:1107.5992*, 2011.

[8] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.

[9] Eleonora Bardelli and Andrea Carlo Giuseppe Mennucci. Probability measures on infinite-dimensional stiefel manifolds. *Journal of Geometric Mechanics*, 9(3):291, 2017.

[10] Thomas Bendokat, Ralf Zimmermann, and P-A Absil. A grassmann manifold handbook: Basic geometry and computational aspects. *arXiv preprint arXiv:2011.13699*, 2020.

[11] Camille Besombes, Olivier Pannekoucke, Corentin Lapeyre, Benjamin Sanderson, and Olivier Thual. Producing realistic climate data with generative adversarial networks. *Nonlinear Processes in Geophysics*, 28(3):347–370, 2021.

[12] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

[13] Jan Boman and Filip Lindskog. Support theorems for the radon transform and cramér-wold theorems. *Journal of theoretical probability*, 22(3):683–710, 2009.

[14] Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229, 2013.

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

[16] Nicolas Bonnotte. *Unidimensional and evolution methods for optimal transportation*. PhD thesis, Paris 11, 2013.

[17] Nicolas Boumal. An introduction to optimization on smooth manifolds. To appear with Cambridge University Press, Apr 2022. URL http://www.nicolasboumal.net/book.

[18] Simon Byrne and Mark Girolami. Geodesic monte carlo on embedded manifolds. *Scandinavian Journal of Statistics*, 40(4):825–845, 2013.

[19] Carlos A Cabrelli and Ursula M Molter. The kantorovich metric for probability measures on the circle. *Journal of Computational and Applied Mathematics*, 57(3):345–361, 1995.

[20] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9912–9924. Curran

Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/70feb62b69f16e0238f741fab228fec2-Paper.pdf.

[21] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/chen20j.html.

[22] Xiongjie Chen, Yongxin Yang, and Yunpeng Li. Augmented sliced wasserstein distances. *arXiv preprint arXiv:2006.08812*, 2020.

[23] Adam Coates, A. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011.

[24] Samuel Cohen, Brandon Amos, and Yaron Lipman. Riemannian convex potential maps. In *International Conference on Machine Learning*, pages 2028–2038. PMLR, 2021.

[25] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9): 1853–1865, 2016.

[26] Li Cui, Xin Qi, Chengfeng Wen, Na Lei, Xinyuan Li, Min Zhang, and Xianfeng Gu. Spherical optimal transportation. *Computer-Aided Design*, 115:181–193, 2019.

[27] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[28] Susanna Dann. On the minkowski-funk transform. *arXiv preprint arXiv:1003.5565*, 2010.

[29] Tim R. Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M. Tomczak. Hyperspherical variational auto-encoders. In Amir Globerson and Ricardo Silva, editors, *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, pages 856–865. AUAI Press, 2018. URL http://auai.org/uai2018/proceedings/papers/309.pdf.

[30] Nicola De Cao and Wilker Aziz. The power spherical distribution. *arXiv preprint arXiv:2006.04437*, 2020.

[31] Julie Delon, Julien Salomon, and Andrei Sobolevski. Fast transport optimization for monge costs on the circle. *SIAM Journal on Applied Mathematics*, 70(7):2239–2258, 2010.

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

[33] Marco Di Marzio, Agnese Panzera, and Charles C Taylor. Nonparametric regression for spherical data. *Journal of the American Statistical Association*, 109(506):748–763, 2014.

[34] Arnaud Doucet, Nando De Freitas, Neil James Gordon, et al. *Sequential Monte Carlo methods in practice*, volume 1. Springer, 2001.

[35] Leon Ehrenpreis. *The universality of the Radon transform*. OUP Oxford, 2003.

[36] Kilian Fatras, Younes Zine, Rémi Flamary, Remi Gribonval, and Nicolas Courty. Learning with minibatch wasserstein : asymptotic and gradient properties. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 2131–2141. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/fatras20a.html.

[37] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.

[38] Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouve, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2681–2690, 2019.

[39] Alessio Figalli and Cédric Villani. Optimal transport and curvature. In *Nonlinear PDE's and Applications*, pages 171–217. Springer, 2011.

[40] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021. URL http://jmlr.org/papers/v22/20-451.html.

[41] P Thomas Fletcher, Conglin Lu, Stephen M Pizer, and Sarang Joshi. Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE transactions on medical imaging*, 23(8):995–1005, 2004.

[42] Walter Gander. Algorithms for the qr decomposition. *Res. Rep*, 80(02):1251–1268, 1980.

[43] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

[44] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21271–21284. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/f3ada80d5c4ee70142b17b8192b2958e-Paper.pdf.

[45] Armelle Guillou, Philippe Naveau, and Alexandre You. A folding methodology for multivariate extremes: estimation of the spectral probability measure and actuarial applications. *Scandinavian Actuarial Journal*, 2015(7):549–572, 2015.

[46] Brittany Froese Hamfeldt and Axel GR Turnquist. A convergence framework for optimal transport on the sphere. *arXiv preprint arXiv:2103.05739*, 2021.

[47] Brittany Froese Hamfeldt and Axel GR Turnquist. A convergent finite difference method for optimal transport on the sphere. *arXiv preprint arXiv:2105.03500*, 2021.

[48] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[49] Sigurdur Helgason et al. *Integral geometry and Radon transforms*. Springer, 2011.

[50] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[51] Andrew Homan and Hanming Zhou. Injectivity and stability for a generic class of generalized radon transforms. *The Journal of Geometric Analysis*, 27(2):1515–1529, 2017.

[52] Andrés Hoyos-Idrobo. Aligning hyperbolic representations: an optimal transport-based approach. *arXiv preprint arXiv:2012.01089*, 2020.

[53] Minhui Huang, Shiqian Ma, and Lifeng Lai. A riemannian block coordinate descent method for computing the projection robust wasserstein distance. In *International Conference on Machine Learning*, pages 4446–4455. PMLR, 2021.

[54] Shayan Hundrieser, Marcel Klatt, and Axel Munk. The statistics of circular optimal transport. *arXiv preprint arXiv:2103.15426*, 2021.

[55] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[56] Sungkyu Jung. Geodesic projection of the von mises–fisher distribution for projection pursuit of directional data. *Electronic Journal of Statistics*, 15(1):984–1033, 2021.

[57] Sungkyu Jung, Ian L Dryden, and James Stephen Marron. Analysis of principal nested spheres. *Biometrika*, 99(3):551–568, 2012.

[58] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[59] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[60] Soheil Kolouri, Phillip E Pope, Charles E Martin, and Gustavo K Rohde. Sliced wasserstein auto-encoders. In *International Conference on Learning Representations*, 2018.

[61] Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. Generalized sliced wasserstein distances. *Advances in Neural Information Processing Systems*, 32, 2019.

[62] Alex Krizhevsky. Learning multiple layers of features from tiny images, 2009.

[63] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

[64] Gerhard Kurz and Uwe D Hanebeck. Stochastic sampling of the hyperspherical von mises–fisher distribution without rejection methods. In *2015 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6. IEEE, 2015.

[65] Shiwei Lan, Bo Zhou, and Babak Shahbaba. Spherical hamiltonian monte carlo for constrained target distributions. In *International Conference on Machine Learning*, pages 629–637. PMLR, 2014.

[66] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

[67] Mufan Bill Li and Murat A Erdogdu. Riemannian langevin algorithm for solving semidefinite programs. *arXiv preprint arXiv:2010.11176*, 2020.

[68] Tianyi Lin, Chenyou Fan, Nhat Ho, Marco Cuturi, and Michael Jordan. Projection robust wasserstein distance and riemannian optimization. *Advances in neural information processing systems*, 33:9383–9397, 2020.

[69] Tianyi Lin, Zeyu Zheng, Elynn Chen, Marco Cuturi, and Michael I Jordan. On projection robust optimal transport: Sample complexity and model misspecification. In *International Conference on Artificial Intelligence and Statistics*, pages 262–270. PMLR, 2021.

[70] Chang Liu, Jun Zhu, and Yang Song. Stochastic gradient geodesic mcmc methods. *Advances in neural information processing systems*, 29, 2016.

[71] Jun S Liu and Rong Chen. Blind deconvolution via sequential imputations. *Journal of the american statistical association*, 90(430):567–576, 1995.

[72] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[73] Kanti V Mardia, Peter E Jupp, and KV Mardia. *Directional statistics*, volume 2. Wiley Online Library, 2000.

[74] Kantilal Varichand Mardia. Statistics of directional data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 37(3):349–371, 1975.

[75] Robert J McCann. Polar factorization of maps on riemannian manifolds. *Geometric & Functional Analysis GAFA*, 11(3):589–608, 2001.

[76] Kimia Nadjahi. *Sliced-Wasserstein distance for large-scale machine learning: theory, methodology and extensions*. PhD thesis, Institut polytechnique de Paris, 2021.

[77] Kimia Nadjahi, Alain Durmus, Umut Simsekli, and Roland Badeau. Asymptotic guarantees for learning generative models with the sliced-wasserstein distance. *Advances in Neural Information Processing Systems*, 32, 2019.

[78] Frank Natterer. *The mathematics of computerized tomography*. SIAM, 2001.

[79] Khai Nguyen and Nhat Ho. Amortized projection optimization for sliced wasserstein generative models. *arXiv preprint arXiv:2203.13417*, 2022.

[80] Khai Nguyen and Nhat Ho. Revisiting sliced wasserstein on images: From vectorization to convolution. *arXiv preprint arXiv:2204.01188*, 2022.

[81] Khai Nguyen, Nhat Ho, Tung Pham, and Hung Bui. Distributional sliced-wasserstein and applications to generative modeling. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=QYjO70ACDK.

[82] George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

[83] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

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

[85] François-Pierre Paty and Marco Cuturi. Subspace robust wasserstein distances. In *International conference on machine learning*, pages 5072–5081. PMLR, 2019.

[86] Nathanaël Perraudin, Michaël Defferrard, Tomasz Kacprzak, and Raphael Sgier. Deepsphere: Efficient spherical convolutional neural network with healpix sampling for cosmological applications. *Astronomy and Computing*, 27:130–146, 2019.

[87] Arthur Pewsey and Eduardo García-Portugués. Recent advances in directional statistics. *Test*, 30(1):1–58, 2021.

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

[89] Julien Rabin, Julie Delon, and Yann Gousseau. Transportation distances on the circle. *Journal of Mathematical Imaging and Vision*, 41(1):147–167, 2011.

[90] Julien Rabin, Gabriel Peyré, Julie Delon, and Marc Bernot. Wasserstein barycenter and its application to texture mixing. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 435–446. Springer, 2011.

[91] Danilo J Rezende and Sébastien Racanière. Implicit riemannian concave potential maps. *arXiv preprint arXiv:2110.01288*, 2021.

[92] Danilo Jimenez Rezende, George Papamakarios, Sébastien Racaniere, Michael Albergo, Gurtej Kanwar, Phiala Shanahan, and Kyle Cranmer. Normalizing flows on tori and spheres. In *International Conference on Machine Learning*, pages 8083–8092. PMLR, 2020.

[93] Christian P Robert, George Casella, and George Casella. *Monte Carlo statistical methods*, volume 2. Springer, 1999.

[94] Boris Rubin. Inversion and characterization of the hemispherical transform. *Journal d'Analyse Mathématique*, 77(1):105–128, 1999.

[95] Boris Rubin. Inversion formulas for the spherical radon transform and the generalized cosine transform. *Advances in Applied Mathematics*, 29(3):471–497, 2002.

[96] Boris Rubin. Notes on radon transforms in integral geometry. *Fractional Calculus and Applied Analysis*, 6(1):25–72, 2003.

[97] Meyer Scetbon, Marco Cuturi, and Gabriel Peyré. Low-rank sinkhorn factorization. In *International Conference on Machine Learning*, pages 9344–9354. PMLR, 2021.

[98] Suvrit Sra. Directional statistics in machine learning: a brief review. *Applied Directional Statistics: Modern Methods and Case Studies*, 225:6, 2018.

[99] Ilya O. Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schölkopf. Wasserstein auto-encoders. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=HkL7n1-0b.

[100] Gary Ulrich. Computer generation of distributions on the m-sphere. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 33(2):158–163, 1984.

[101] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

[102] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[103] Dilin Wang and Qiang Liu. Learning to draw samples: With application to amortized mle for generative adversarial learning. *arXiv preprint arXiv:1611.01722*, 2016.

[104] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020.

[105] Xiao Wang, Qi Lei, and Ioannis Panageas. Fast convergence of langevin dynamics on manifold: Geodesics meet log-sobolev. *Advances in Neural Information Processing Systems*, 33:18894–18904, 2020.

[106] Michael Werman, Shmuel Peleg, and Azriel Rosenfeld. A distance metric for multidimensional histograms. *Computer Vision, Graphics, and Image Processing*, 32(3):328–336, 1985.

[107] Andrew TA Wood. Simulation of the von mises fisher distribution. *Communications in statistics-simulation and computation*, 23(1):157–164, 1994.

[108] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[109] Jiacheng Xu and Greg Durrett. Spherical latent spaces for stable variational autoencoders. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium,*

October 31 - November 4, 2018, pages 4503–4513. Association for Computational Linguistics, 2018.

[110] Mingxuan Yi and Song Liu. Sliced wasserstein variational inference. In *Fourth Symposium on Advances in Approximate Bayesian Inference*, 2021.

Checklist

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