Dynamic Flows on Curved Space Generated by Labeled Data

Xinru Hua\textsuperscript{1}, Truyen Nguyen\textsuperscript{2}, Tam Le\textsuperscript{3}, Jose Blanchet\textsuperscript{1} and Viet Anh Nguyen\textsuperscript{4}

\textsuperscript{1} Stanford University, United States
\textsuperscript{2} The University of Akron, United States
\textsuperscript{3} The Institute of Statistical Mathematics / RIKEN AIP, Japan
\textsuperscript{4} Chinese University of Hong Kong, China
{huaxinru, jose.blanchet} @stanford.edu, truyennguyen3@gmail.com, tam@ism.ac.jp, nguyen@se.cuhk.edu.hk

Abstract

The scarcity of labeled data is a long-standing challenge for many machine learning tasks. We propose our gradient flow method to leverage the existing dataset (i.e., source) to generate new samples that are close to the dataset of interest (i.e., target). We lift both datasets to the space of probability distributions on the feature-Gaussian manifold, and then develop a gradient flow method that minimizes the maximum mean discrepancy loss. To perform the gradient flow of distributions on the curved feature-Gaussian space, we unravel the Riemannian structure of the space and compute explicitly the Riemannian gradient of the loss function induced by the optimal transport metric. For practical applications, we also propose a discretized flow, and provide conditional results guaranteeing the global convergence of the flow to the optimum. We illustrate the results of our proposed gradient flow method on several real-world datasets and show our method can improve the accuracy of classification models in transfer learning settings.

1 Introduction

A major challenge in many data science applications is the scarcity of labeled data. Data augmentation methods have been studied in the literature; see for example, the noise injection methods [Moreno-Barea et al., 2018], generative models [Yi et al., 2019], and [Shorten and Khoshgoftaar, 2019] for a survey. We consider a setting where one domain has only a few labeled samples for each class, so we cannot train a well-performing classifier with the available data. To alleviate the data scarcity problem in this setting, we propose to enrich the target dataset by generating additional labeled samples. Using generative models is not possible in our setting because they usually require more than a few samples for each class to learn and generate high-quality new samples [Gao et al., 2018]. In our work, we choose a source dataset with extensive labeled data and then flow the labeled data to the target dataset. Precisely, we introduce a novel data augmentation methodology based on a gradient flow approach that minimizes the maximum mean discrepancy (MMD) distance between the target and the augmented data. Therefore, when minimizing the MMD distance, we are able to obtain an efficient scheme which generates additional labeled data from the target distribution. Our scheme is model-independent and can be applied to any datasets regardless of the number of classes or dimensionality\footnote{Our code and supplementary are available at https://github.com/LucyXH/Dynamic_Flows_Curved_Space/}.

Mathematically, we consider a feature space $X = \mathbb{R}^m$ and a categorical label space $Y$. We have a source domain dataset consisting of $N$ samples $(x_i, y_i) \in X \times Y$ for $i = 1, \ldots, N$, and a target domain dataset of $M$ samples $(\bar{x}_j, \bar{y}_j) \in X \times Y$ for $j = 1, \ldots, M (M \ll N)$. The ultimate goal of this paper is to generate new samples in the target domain, and we aim to generate new samples whose distribution is as close as possible to the distribution that governs the target domain.

We here introduce a gradient flow method [Arbel et al., 2019; Mroueh et al., 2019] to synthesize new, unseen data samples. Gradient flow is a continuous flow along the path where a considered loss function decreases its value. Because we have extensive source domain samples, it is possible to flow each source sample towards the target data while minimizing the loss function. The terminal product of the flow will be new samples that can sufficiently approximate the distribution of the target domain. Thus, gradient flow is an approach to synthesize new target domain samples, and is a complement to data augmentation methods, like adding random noise.

Unfortunately, formulating a gradient flow algorithm for labeled data with categorical set $\bar{Y}$ is problematic. Indeed, there is no clear metric structure on $\bar{Y}$ in order to define the topological neighborhood, this in turn leads to the difficulty of forming the gradients with respect to the categorical component. To overcome this difficulty, we lift each individual label to a richer structure. For example, a label such as “0” is replaced by a mean vector and a covariance matrix based on the whole distribution of the information associated to this particular label. Then it will be much more natural to apply gradient flow algorithms in the space of the lifted representation. A gradient flow on the dataset space with this idea was recently proposed in [Alvarez-Melis and Fusi, 2021] by leveraging a new notion of distance between datasets in [Alvarez-Melis and Fusi, 2020; Courty et al., 2017; Damodaran et al., 2018]. The main idea behind this approach is to reparametrize the categorical space...
was pioneered in [Mroueh (RKHS). In contrast to the Kullback-Leibler divergence flow, recent gradient flow works able to higher-dimensional image data, that is, higher than ImageNet datasets to highlight that our algorithm is scal-

tively augment the target data, and thus can significantly boost the accuracy in the classification task in the few-shot setting. We also compare our method with [Alvarez-Melis and Fusi, 2021], mixup method [Zhang et al., 2017], and traditional data augmentation methods in Supplementary B.7, which show that our method improves the accuracy in transfer learning more than these methods.

Some works study nonparametric gradient flows using the 2-Wasserstein distance between distributions [Ambrosio et al., 2008; Jordan et al., 1998; Otto, 2001; Villani, 2008; Santambrogio, 2015; Santambrogio, 2017; Frogn and Poggio, 2020], but only for distributions on Euclidean spaces and different metrics. Nonparametric gradient flows with other metrics include Sliced-Wasserstein Descent [Liutkus et al., 2019], Stein Descent [Liu, 2017; Liu and Wang, 2016], and Sobolev Descent [Mroueh et al., 2019], but only for distributions on Euclidean spaces. In particular, [Liu, 2017] introduce Riemannian structures for the Stein geometry on flat spaces, while ours is on a curved space. Parametric flows for training GANs are studied in [Chizat and Bach, 2018; Arbel et al., 2020; Mroueh and Nguyen, 2021].

Contributions. We study a gradient flow approach to synthe-
size new labeled samples related to the target domain. To con-
struct this flow, we consider the space of probability distribu-
tions on the feature-Gaussian manifold, and we are metrizing this space with an optimal transport distance. We summarize the contributions of this paper as follows.

• We study in details the Riemannian structure of the feature-
Gaussian manifold in Section 3, as well as the Riemannian structure of the space of probability measures supported on this manifold in Supplementary A.1.

• We consider a gradient flow that minimizes the squared MMD loss function to the target distribution. We describe explicitly the (Riemannian) gradient of the squared MMD in Lemma 5, and we provide a partial differential equation describing the evolution of the gradient flow that follows the (Riemannian) steepest descent direction.

• We propose two discretized schemes to approximate the continuous gradient flow equation in Section 4.1 and 4.2. We provide conditions guaranteeing the global convergence of our gradient flows to the optimum in both schemes.

• In Section 5, we demonstrate numerical results with our method on real-world image datasets. We show that our method can generate high-fidelity images and improve the classification accuracy in transfer learning settings.

Notations. We use $\mathbb{S}^n$ to denote the set of $n \times n$ real and symmetric matrices, and $\mathbb{S}^n_+ \subset \mathbb{S}^n$ consists of all positive definite matrices. For $A \in \mathbb{S}^n$, $\text{tr}(A) := \sum_i A_{ii}$. We use $(\cdot, \cdot)$ and $\| \cdot \|_2$ to denote the standard inner product and norm on Euclidean spaces. Let $P(X)$ be the collection of all probability distributions with finite second moment on metric space $X$. If $\varphi : X \to Y$ is a Borel map and $\nu \in P(X)$, then the push-
forward $\varphi_* \nu$ is the distribution on $Y$ given by $\varphi_* \nu(E) = \nu(\varphi^{-1}(E))$ for all Borel sets $E \subset Y$. For a function $f$ of the continuous time variable $t$, $f_t$ denotes the value of $f$ at $t$ while $\partial_z f$ denotes the standard derivative of $f$ w.r.t. $t$. Also, $\delta_z$ denotes the Dirac delta measure at $z$.

All proofs are provided in the Supplementary material.
2 Labeled Data Synthesis via Gradient Flows of Lifted Distributions

In this section, we describe our approach to synthesize target domain samples using gradient flows. A holistic view of our method is presented in Fig. 1.

In the first step, we would need to lift the feature-label space $\mathcal{X} \times \mathcal{Y}$ to a higher dimensional space where a metric can be defined. Consider momentarily the source data samples $(x_i, y_i)_{i=1}^N$. Notice that this data can be represented as an empirical distribution $\nu$ on $\mathcal{X} \times \mathcal{Y}$. More precisely, we have $\nu = N^{-1} \sum_{i=1}^N \delta_{(x_i, y_i)}$. As $\mathcal{Y}$ is discrete, the law of conditional probabilities allows us to dis-integrate $\nu$ into the conditional distributions $\nu_y$ of $X|Y = y$ satisfying $\nu(E \times F) = \int_E \nu_y(E) \nu^2(dy)$ for every $E \subset \mathcal{X}$ and $F \subset \mathcal{Y}$, where $\nu^2 := N^{-1} \sum_{i=1}^N \delta_{y_i}$ is the second marginal of $\nu$ [Ambrosio et al., 2008, Theorem 5.3.1]. The lifting procedure is obtained by employing a pre-determined mapping $\phi : \mathcal{X} \to \mathbb{R}^n$, and any categorical value $y \in \mathcal{Y}$ can now be represented as an $n$-dimensional distribution $\phi_y \nu_y$. Using this lifting, any source sample $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ is lifted to a point $(x_i, \phi_y y_i) \in \mathcal{X} \times \mathcal{P}(\mathbb{R}^n)$ and the source dataset is representable as an empirical distribution of the form $N^{-1} \sum_{i=1}^N \delta_{(x_i, \phi_y y_i)}$.

The lifted representation of a categorical value $y \in \mathcal{Y}$ as an $n$-dimensional distribution $\phi_y \nu_y \in \mathcal{P}(\mathbb{R}^n)$ is advantageous because $\mathcal{P}(\mathbb{R}^n)$ is metrizable, for example, using the $2$-Wasserstein distance. The downside is that $\mathcal{P}(\mathbb{R}^n)$ is infinite dimensional, and encoding the lifted distributions in this lifted representation is not efficient. To resolve this issue, we assume that $\phi_y \nu_y$ is Gaussian for all $y \in \mathcal{Y}$, and any distribution $\phi_y \nu_y$ can be characterized by the mean vector $\mu_y \in \mathbb{R}^n$ and covariance matrix $\Sigma_y \in \mathbb{S}^n_+$ defined as $\mu_y = \int_{\mathcal{X}} \phi(x) \nu_y(dx)$ and $\Sigma_y = \int_{\mathcal{X}} [\phi(x) - \mu_y] [\phi(x) - \mu_y]^T \nu_y(dx)$ for all $y \in \mathcal{Y}$, where $^T$ denotes the transposition of a vector. In real-world settings, the conditional moments of $\phi(X)|Y$ are sufficiently different for $y \neq y'$, and thus the representations using $(\mu_y, \Sigma_y)$ will likely lead to any loss of label information. With this lifting, the source data thus can be represented as an empirical distribution $\rho^0$ on $\mathbb{R}^m \times \mathbb{R}^n \times \mathbb{S}^n_+$ via $\rho^0 = N^{-1} \sum_{i=1}^N \delta_{(x_i, \mu_{y_i}, \Sigma_{y_i})}$. By an analogous construction to compute $\tilde{\mu}_y$ and $\tilde{\Sigma}_y$ using the target data, the target domain data $(x_j, y_j)_{j=1}^M$ can be represented as another empirical distribution $\varrho = M^{-1} \sum_{j=1}^M \delta_{(x_j, \tilde{\mu}_{y_j}, \tilde{\Sigma}_{y_j})}$. Let us denote the shorthand $\mathcal{Z} = \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{S}^n_+$, then $\rho^0$ and $\varrho$ are both probability measures on $\mathcal{Z}$. We refer to $\rho^0$ and $\varrho$ as the feature-Gaussian representations of the source and target datasets.

We now consider the gradient flow associated with the optimization problem

$$\min_{\rho \in \mathcal{P} (\mathcal{Z})} \{ F (\rho) := \frac{1}{2} \text{MMD}(\rho, \varrho)^2 \}$$

under the initialization $\rho = \rho^0$. The objective function $F(\rho)$ quantifies how far an incumbent solution $\rho$ is from the target distribution $\varrho$, measured using the MMD distance. In Sections 3 and 4, we will provide the necessary ingredients to construct this flow.

Suppose that after $T$ iterations of the discretized gradient flow algorithm, we obtain a distribution $\rho^T \in \mathcal{P}(\mathcal{Z})$ that is sufficiently close to $\varrho$, i.e., $F(\rho^T)$ is close to zero. Then we can recover new target labels by projecting the samples of the distribution $\rho^T$ to the locations on $\mathcal{X} \times \mathcal{Y}$. This projection can be computed efficiently by solving a linear optimization problem, as discussed in Supplementary B.3.

**Remark 1** (Reduction of dimensions). If $m = n$ and $\phi$ is the identity map, then our lifting procedure coincides with that proposed in [Alvarez-Melis and Fusi, 2020]. However, a large $n$ is redundant, especially when the cardinality of $\mathcal{Y}$ is low. If $n \ll m$, then $\phi$ offers significant reduction in the number of dimensions, and will speed up the gradient flow algorithms.

**Remark 2** (Generalization to elliptical distributions). Our framework can be extended to the symmetric elliptical distributions because the Bures distance for elliptical distributions admits the same closed-form as for the Gaussian distributions [Gembicki, 1980]. In this paper, we use $\phi$ as the t-SNE embedding. According to [van der Maaten and Hinton, 2008], t-SNE’s low-dimensional embedded space forms a Student-t distribution, which is an elliptical distribution.

3 Riemannian Geometry of $\mathcal{Z}$ and $\mathcal{P}(\mathcal{Z})$

If we opt to measure the distance between two Gaussian distributions using the 2-Wasserstein metric, then this choice would induce a natural distance $d$ on the space $\mathcal{Z} = \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{S}^n_+$ prescribed as

$$d((x_1, \mu_1, \Sigma_1), (x_2, \mu_2, \Sigma_2)) := \|x_1 - x_2\|_2^2 + \|\mu_1 - \mu_2\|_2^2 + B(\Sigma_1, \Sigma_2)^2 \right)^{\frac{1}{2}},$$

(3.1)

where $B$ is the Bures metric on $\mathbb{S}^n_+$, given by $B(\Sigma_1, \Sigma_2) := [\text{tr}(\Sigma_1 + \Sigma_2 - 2\Sigma_1^{\frac{1}{2}} \Sigma_2^{\frac{1}{2}})]^{\frac{1}{2}}$.

As $B$ is a metric on $\mathbb{S}^n_+$ [Bhatia et al., 2019, p.167], $d$ is hence a product metric on $\mathcal{Z}$. In this section, first, we study the non-Euclidean geometry of $\mathcal{Z}$ under the ground

<table>
<thead>
<tr>
<th>Paper</th>
<th>Dataset</th>
<th>On curved Riemannian space</th>
<th>Gradient has mean and covariance component</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Alvarez-Melis and Fusi, 2021]</td>
<td>synthetic, *NIST, and CIFAR10</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>[Arbel et al., 2019]</td>
<td>synthetic</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Ours</td>
<td>synthetic, *NIST, and TinyImageNet</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 1: To the best of our knowledge, we provide the first results on the full gradient of the features and lifted labels on a curved Riemannian space. We also conduct numerical experiments on the highest-dimension real-world datasets.
where $t$ is of the form $z\Sigma S$, flow for the squared $P$ metric. Figure 1: Schematic view of our approach: The source and target datasets are first lifted to distributions tangent vector $nS(Z)$: is a product Riemannian manifold, any geodesic in $Z$. As we shall see, $Z$ is a curved space as its geodesics are not straight lines and involve solutions to the Lyapunov equation. For any positive definite matrix $Σ \in S_{++}^n$ and any symmetric matrix $V \in S^n$, the Lyapunov equation

$$HΣ + ΣH = V \quad (3.2)$$

has a unique solution $H \in S^n$ [Bhatia, 1997, Theorem VII.2.1]. Let $Σ[V]$ denote this unique solution $H$.

The space $S^n_{++}$ is a Riemannian manifold with the Bures metric $B$ as the associated distance function, see [Takatsu, 2011, Proposition A]. Since $Z$ is the product of two Euclidean spaces and $S^n_{++}$, this gives rise to the following geometric structure for $Z$.

**Proposition 3** (Geometry of $Z$). The space $Z$ is a Riemannian manifold: at each point $z = (x, μ, Σ) ∈ Z$, the tangent space is $T_z Z = \mathbb{R}^m \times \mathbb{R}^n × S^n$, and the Riemannian metric is

$$\left\langle (w_1, v_1, V_1), (w_2, v_2, V_2) \right\rangle_z := \langle w_1, w_2 \rangle + \langle v_1, v_2 \rangle + \langle V_1, V_2 \rangle_Σ \quad (3.3)$$

for two tangent vectors $(w_1, v_1, V_1)$ and $(w_2, v_2, V_2)$ in $\mathbb{R}^m × \mathbb{R}^n × S^n$, where $\langle V_1, V_2 \rangle_Σ := tr(Σ[V_1] Σ[V_2])$. Moreover, the distance function corresponding to this Riemannian metric coincides with the $d$ distance given by (3.1).

As $Z$ is a product Riemannian manifold, any geodesic in $Z$ is of the form $(θ, γ, Γ)$ with $θ, γ$ being the Euclidean geodesics (straight lines) and $Γ$ being a geodesic in the Riemannian manifold $S^n_{++}$. More precisely, for each $Σ ∈ S^n_{++}$ and each tangent vector $V ∈ S^n$, the geodesic in the manifold $S^n_{++}$ emanating from $Σ$ with direction $V$ is given by

$$Γ(t) = (I + tLΣ[V]Σ(I + tLΣ[V])) \quad \text{for } t ∈ J^*, \quad (3.4)$$

where $J^*$ is the open interval about the origin given by $J^* = \{t ∈ \mathbb{R} : t + tLΣ[V] ∈ S^n_{++}\}$ [Malagò et al., 2018]. As a consequence, for each point $(x, μ, Σ) ∈ Z$ and each tangent vector $(w, v, V) ∈ \mathbb{R}^m × \mathbb{R}^n × S^n$, the Riemannian exponential map in $Z$ for $t ∈ J^*$ is given by

$$\exp_{(x, μ, Σ)}(t(w, v, V)) := (θ(t), γ(t), Γ(t)) \quad (3.5)$$

where $θ(t) := x + tw$, $γ(t) := μ + tv$, and $Γ(t)$ is defined by (3.4). By definition, $t ↦ \exp_{(x, μ, Σ)}(t(w, v, V))$ is the geodesic emanating from $(x, μ, Σ)$ with direction $(w, v, V)$.

Given the Riemannian metric (3.3), one can define the corresponding notion of gradient and divergence [Lee, 2003]. For a differentiable function $φ : Z \rightarrow \mathbb{R}$, its gradient $\nabla_dφ(z)$ w.r.t. the metric $d$ defined by (3.1) is the unique element in the tangent space $\mathbb{R}^m × \mathbb{R}^n × S^n$ satisfying

$$\left\langle \nabla_dφ(z), (w, v, V) \right\rangle_z = Dφ_z(w, v, V)$$

for all $(w, v, V) ∈ \mathbb{R}^m × \mathbb{R}^n × S^n$ with $Dφ_z(w, v, V)$ denoting the standard directional derivative of $φ$ at $z$ in the direction $(w, v, V)$. By exploiting the special form of $\langle -, - \rangle_z$ in (3.3), we can compute $\nabla_dφ(z)$ explicitly:

**Lemma 4** (Gradintes). For a differentiable function $φ : Z \rightarrow \mathbb{R}$, we have for $z = (x, μ, Σ)$ that

$$\nabla_dφ(z) = (\nabla_zφ(z), \nabla_μφ(z), 2[\nabla_Σφ(z)]Σ + 2Σ[\nabla_Σφ(z)]) \quad (3.6)$$

where $(\nabla_z, \nabla_μ, \nabla_Σ)$ are the standard (Euclidean) gradients of the respective components.

The last component in formula (3.6) for $\nabla_dφ$ reflects the curved geometry of $Z$, and can be interpreted as the Riemann gradient of the function $Σ → φ(x, μ, Σ)$ w.r.t. the Bures distance $B$.

For a continuous vector field $Φ : Z \rightarrow \mathbb{R}^m × \mathbb{R}^n × S^n$ and a distribution $ρ ∈ P(Z)$, the divergence $\text{div}_d(Φ)$ is the signed measure on $Z$ satisfying the integration by parts formula

$$\int_Z φ(z) \text{div}_d(Φ)(dz) = -\int_Z (Φ(z), \nabla_dφ(z))_z ρ(dz)$$

for every differentiable function $φ : Z \rightarrow \mathbb{R}$ with compact support. In case $ρ$ has a density w.r.t. the Riemannian volume form on $Z$, then this definition coincides with the standard divergence operator induced by Riemannian metric (3.3). The optimal transport distance and its induced Riemannian metric on the space $P(Z)$ are relegated to Supplementary A.1.

4 Gradient Flow for Maximum Mean Discrepancy

As $P(Z)$ is an infinite dimensional curved space, many machine learning methods based on finite dimensional or linear
structure cannot be directly applied to this manifold. To circumvent this problem, we use a positive definite kernel to map \( \mathcal{P}(Z) \) to a RKHS and then perform our analysis on it. Let \( k \) be a positive definite kernel on \( Z \), and let \( H \) be the RKHS generated by \( k \). The inner product on \( H \) is denoted by \( \langle \cdot, \cdot \rangle_H \), and the kernel mean embedding \( \rho \in \mathcal{P}(Z) \rightarrow m_\rho(z) \in H \) is given by \( m_\rho(z) := \int_Z k(z, w) \rho(dw) \) for \( z \in Z \). The MMD [Gretton et al., 2012] between \( \rho \in \mathcal{P}(Z) \) and the target \( \rho \) is defined as the maximum of the mean difference between the two distributions over all test functions in the unit ball of \( H \) (see Supplementary A.3). Moreover, it can be expressed by MMD(\( \rho, \mu \)) = \| m_\rho - m_\mu \|_H. When \( k \) is characteristic, the kernel mean embedding \( \rho \rightarrow m_\rho \) is injective and therefore, MMD(\( \rho, \mu \)) = 0 if and only if \( \rho = \mu \).

Consider the loss function \( F[\rho] := \frac{1}{2} \text{MMD}(\rho, \mu)^2 = \frac{1}{2} \| m_\rho - m_\mu \|_H^2 \). As explained in the introduction, there are three advantages of MMD over Kullback-Leibler divergence: its associated gradient flow can employ a sample approximation for the target distribution, the input distribution \( \rho \) does not have to be absolutely continuous w.r.t. the target distribution \( \mu \), and the squared MMD possesses unbiased sample gradients. For each \( \rho \), the Riemannian gradient \( \nabla F[\rho] \) is defined as the unique element in \( T_\rho \mathcal{P}(Z) \) satisfying \( g_\rho(\nabla F[\rho], \zeta) = \frac{d}{dt} F[\rho_t] \) for every differentiable curve \( t \rightarrow \rho_t \in \mathcal{P}(Z) \) passing through \( \rho \) at \( t = 0 \) with tangent vector \( \partial_t \rho_t|_{t=0} = \zeta \). By using the Riemannian metric tensor (eq. A.3), we can compute explicitly this gradient.

**Lemma 5 (Gradient formula).** The Riemannian gradient of \( F \) satisfies \( \nabla F[\rho] = - \nabla_\rho (\rho \nabla_d(m_\rho - m_\mu)) \).

The Riemannian gradient \( \nabla F \) on \( \mathcal{P}(Z) \) depends not only on the gradient operator \( \nabla_d \) but also on the divergence operator. Using Lemma 5, we can rewrite the gradient flow equation \( \partial_t \rho_t = - \nabla F[\rho_t] \) explicitly as

\[
\partial_t \rho_t = \text{div}_d (\rho \nabla_d (m_\rho - m_\mu)) \quad \text{for} \quad t \geq 0.
\]

(4.1)

The next result exhibits the rate at which \( F \) decreases its value along the flow.

**Proposition 6 (Rate of decrease).** Along the gradient flow \( t \rightarrow \rho_t \in \mathcal{P}(Z) \) given by (4.1), we have

\[
\frac{d}{dt} F[\rho_t] = - \int_Z \| \nabla_d (m_\rho - m_\mu) \|_2^2 \rho_t(dz) \quad \text{for} \quad t \geq 0.
\]

Proposition 6 implies that \( \frac{d}{dt} F[\rho_t] = 0 \) if and only if \( \nabla_d (m_\rho - m_\mu)(z) = 0 \) for every \( z \) in the support of the distribution \( \rho_t \). Thus, the objective function will decrease whenever the gradient \( \nabla_d (m_\rho - m_\mu) \) is not identically zero.

### 4.1 Riemannian Forward Euler Scheme

We propose the Riemannian version of the forward Euler scheme to discretize continuous flow (4.1):

\[
\rho^{t+1} = \exp(s_t \Phi^t) \# \rho^t \quad \text{with} \quad \Phi^t := - \nabla_d (m_\rho - m_\mu),
\]

(4.2)

where \( s_t > 0 \) is the step size. Here, for a vector field \( \Phi = (\Phi_1, \Phi_2, \Phi_3) : Z \rightarrow \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{S}^n \) and for \( \varepsilon \geq 0 \), \( \exp(\varepsilon \Phi) : Z \rightarrow Z \) is the Riemannian exponential map induced by (3.5), i.e., for \( z = (x, \mu, \Sigma) \in Z \):

\[
\exp_\Sigma (\varepsilon \Phi(z)) := \left( \frac{x + \varepsilon \Phi_1(z)}{\mu + \varepsilon \Phi_2(z)}, (I + \varepsilon L_{\Sigma}[\Phi_3(z)]) \Sigma (I + \varepsilon L_{\Sigma}[\Phi_3(z)]) \right).
\]

Notice in the above equation that the input \( z \) affects simultaneously the bases of the exponential map \( \exp_\Sigma \) as well as the direction \( \Phi(z) \). This map is the \( \varepsilon \)-perturbation of the identity map along geodesics with directions \( \Phi \).

When \( \rho^t = \mathcal{N}^{-1} \sum_{k=1}^N \delta_{z_k^t} \) is an empirical distribution, scheme (4.2) flows each particle \( z_k^t \) to the new position \( z_k^{t+1} = \exp_\Sigma (s_t \Phi(z_k^t)) \). The next lemma shows that \( \Phi^t \) is the steepest descent direction for \( F \) w.r.t. the exponential map among all directions in the space \( \mathbb{L}^2(\rho^t) \), which is the collection of all vector fields \( F \) on \( Z \) satisfying \( \| F(\rho^t) \|_{\mathbb{L}^2(\rho^t)} < \infty \).

**Lemma 7 (Steepest descent direction).** Fix a distribution \( \rho^t \in \mathcal{P}(Z) \). For any vector field \( \Phi : Z \rightarrow \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{S}^n \), we have

\[
\frac{d}{dz} F[\exp(\varepsilon \Phi \# \rho^t)] = - \int_Z \nabla_d (m_\rho - m_\mu)(z) \rho^t(dz).
\]

If \( \Phi^t \) is the unit vector field (w.r.t. the \( \| \cdot \|_{\mathbb{L}^2(\rho^t)} \) norm) in the direction of \( \Phi^t \) given in (4.2), then

\[
\frac{d}{dt} \| \exp(\varepsilon \Phi^t) \# \rho^t \|_{\mathbb{L}^2(\rho^t)} = - \| \nabla_d (m_\rho - m_\mu) \|_{\mathbb{L}^2(\rho^t)}
\]

and this is the fastest decay rate among all unit directions \( \Phi \) in \( \mathbb{L}^2(\rho^t) \).

It follows from Lemma 7 that the discrete scheme (4.2) satisfies the Riemannian gradient descent property: if \( \nabla_d (m_\rho - m_\mu) \) is nonzero and if \( s_t > 0 \) is chosen sufficiently small, then \( F[\rho^{t+1}] < F[\rho^t] \). In Proposition 14 in the Supplementary, we quantify the amount of decrease of \( F \) at each iteration. Algorithm 1 implements the flows (4.2) iteratively. Each iteration in Algorithm 1 has complexity \( O(N(Nm + n^3)) \), where \( m \) is the feature’s dimension, \( n \) is the reduced dimension (\( n \ll m \)), \( N \) is the number of particles.

**Convergence.** We now study the (weak) convergence of the solution \( \rho_t \) of the continuous gradient flow (4.1), as well as the discretized counterpart \( \rho^t \) of flow (4.2), to the target distribution \( \rho \). When the kernel \( k \) is characteristic, this convergence is equivalent to \( \lim_{t \rightarrow \infty} \text{MMD}(\rho_t, \rho) = 0 \). Because the objective function \( F \) is not displacement convex [Arbel et al., 2019, Section 3.1], the convergent theory for gradient flows in [Ambrosio et al., 2008] does not apply even in the case of Euclidean spaces. In general, there is a possibility that \( \text{MMD}(\rho_t, \rho) \) does not decrease to zero as \( t \rightarrow \infty \). In view of Proposition 6, this happens if the solutions \( \rho_t \) are trapped inside the set \( \{ \rho : \int_Z \| \nabla_d (m_\rho - m_\mu) \|_2^2 \rho(dz) = 0 \} \). For each distribution \( \rho \) on \( Z \), we define in Supplementary A.3 a symmetric linear and positive operator \( K_\rho : H \rightarrow H \) with the property that \( \langle K_\rho (m_\rho - m_\mu), m_\rho - m_\mu \rangle_H = \int_Z \| \nabla_d (m_\rho - m_\mu) \|_2^2 \rho(dz) \).

Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence (IJCAI-23)

3807
The analysis in Section 4.1 reveals that the gradient flows et al. and the noise level \( \beta \) that scheme (4.3) achieves the global minimum of \( d \) cost on the Euclidean space to the nonstandard cost function. Specifically, we modify algorithm (4.2) by injecting Gaussian noise into the exponential map at each iteration. Here \( \rho \) belongs to the null space of the operator \( K \) operatorally converges in \( m \) steps. Our next result extends algorithm (4.2) by injecting Gaussian noise into the exponential map at each iteration \( t \) to obtain

\[
\rho^{t+1} = \exp(s_t \Phi^T) \# \rho^{t, \beta_t},
\]

with \( f^{\beta_t}(z, u) \rightarrow \exp_z(\beta_t u), \rho^{t, \beta_t} := f^{\beta_t}(\rho^{t} \otimes g) \).

Here \( g \) is a Gaussian measure with distribution \( \mathcal{N}_{\mathbb{R}^n}(0, 1) \otimes \mathcal{N}_{\mathbb{R}^n}(0, 1) \) on the tangent space and \( \mathcal{N}_{\mathbb{Z}^n}(0, 1) \) anisotropic distribution whose upper triangular elements are i.i.d. standard Gaussian random variables. When \( \rho^t = N^{-1} \sum_{i=1}^N \delta_{z_i^t} \), scheme (4.3) flows each particle \( z_i^t \) first to \( z_i^{t, \beta_t} = \exp_{z_i^t}(\beta_t U) \) with noise \( U \sim \mathcal{N}(0, \sigma^2) \) and then to \( z_i^{t+1} = \exp_{z_i^{t, \beta_t}}(s_t \Phi(z_i^{t, \beta_t})) \). Our next result extends Proposition 8 in [Arbel et al., 2019] for the standard quadratic cost on the Euclidean space to the nonstandard cost function \( \mathcal{F} \) on the curved Riemannian manifold \( \mathbb{Z}^n \). It demonstrates that scheme (4.3) achieves the global minimum of \( \mathcal{F} \) provided that \( k \) is a Lipschitz-gradient kernel when and the step size \( s_t \) is well controlled. The proof of Proposition 8 is given in Supplementary A.3 and relies on arguments that are different from that of [Arbel et al., 2019].

Proposition 8 (Objective value decay for noisy scheme). Suppose that \( k \) is a Lipschitz-gradient kernel with constant \( L \) and the noise level \( \beta_t \) satisfies

\[
\lambda \beta_t^2 \mathcal{F}[\rho^t] \leq \int_{\mathbb{Z}} \| \Phi^T(z) \|^2 \rho^t, \beta_t (dz) \tag{4.4}
\]

for some constant \( \lambda > 0 \). Then for \( \rho^{t+1} \) obtained from scheme (4.3), we have

\[
\mathcal{F}[\rho^{t+1}] \leq \mathcal{F}[\rho^t] \exp \left( -\lambda \sum_{i=0}^{t} [s_i (1 - 2Ls_i) \beta_i^2] \right).
\]

In particular, \( \mathcal{F}[\rho^t] \) tends to zero if the sequence \( \sum_{i=0}^{t} s_i (1 - 2Ls_i) \beta_i^2 \) goes to positive infinity. For an adaptive step size \( s_t \leq 1/4L \), this condition is met if, for example, \( \beta_t \) is chosen of the form \((\tau s_t)^{-\gamma} \) while still satisfying (4.4). The noise perturbs the direction of descent, whereas the step size determines how far to move along this perturbed direction. The noise level needs to be adjusted so that the gradient is not too blurred, but it does not necessarily decrease at each iteration. When the incumbent distribution \( \rho^t \) is close to a local optimum, it is helpful to increase the noise level to escape the local optimum. We demonstrate in Lemma 13 in the Supplementary that any positive definite kernel \( k \) with bounded Hessian w.r.t. distance \( d \) is a Lipschitz-gradient kernel. Algorithm 2 in the Supplementary describes (4.3) in details.

5 Numerical Experiments

We evaluate the proposed gradient flow on real-world datasets and then illustrate its applications in transfer learning. We augment samples for the target dataset, where only a few samples in the dataset are available. We consider three datasets: the MNIST (M) [LeCun and Cortes, 2010], Fashion-MNIST (F) [Xiao et al., 2017], Kuzushiji-MNIST (K) [Clanuwat et al., 2018]. To satisfy the Gaussianity assumption of the conditional distributions, we cluster all the images from each class of the datasets and keep the largest cluster for each class. To demonstrate the scalability of our algorithm to higher-dimensional images, we run experiments on Tiny ImageNet (TIN) [Russakovsky et al., 2015] and upscaled SVHN [Netzer et al., 2011] datasets, where images are of \( 3 \times 64 \times 64 \) size.

Our mapping \( \phi \) is from \( \mathbb{R}^m \) to \( \mathbb{R}^2 \) in the lifting procedure. To compute the MMD distance using kernel embeddings, we use a tensor kernel \( k \) on \( \mathbb{Z} \) composed from three standard Gaussian kernels corresponding for each component of the feature space \( \mathbb{R}^m \), the mean space \( \mathbb{R}^2 \) and the covariance matrix space \( S_+^2 \). As a consequence, \( k \) is a characteristic kernel by [Szabó and Sriperumbudur, 2018, Theorem 4].

Experiment: Gradient Flow between Datasets. We visualize the path travelled by each sample from the source domain to the target domain, as depicted in Fig. 2. We draw randomly \( N = 200 \) images equally for 10 classes of the source domain, and \( M = 50 \) images equally for 10 classes of the target domain \( (M = 10 \) for the TIN and SVHN datasets). In each subfigure, each column represents a snapshot of a certain time-step and the samples flow from the source (left) to the target (right) as the number of steps increases. The first column in Fig. 2 are the images from the source domain, where the gradient flows start. Empirically, the algorithm converges after step 140 for *NIST datasets and step 6000 for TIN and SVHN. The experiments are run on a C5.4xlarge AWS instance (a CPU instance) and all finish in about one hour.

5.1 Application in Transfer Learning

Our gradient flow can alleviate the problem of insufficient labeled data by synthesizing new samples to augment the target data.
dataset. In this section, we demonstrate that the generated target domain samples can improve the accuracy in one-shot and five-shot transfer learning tasks.

First, we fix a source domain and pretrain a classifier $P$ on this domain. We draw randomly $N$ samples from the source domain to form the source dataset $(x_i, y_i)_{i=1}^N$. Next, we pick a target domain and draw randomly a few samples from this target domain: for example, in 1-shot learning, only 1 image per class from the target domain is selected to form the target dataset $D = (\bar{x}_j, \bar{y}_j)_{j=1}^M$. We then perform a noisy gradient flow scheme (4.3) from the source dataset to the target dataset to get $N$ new samples $S_T = (x_T^i, y_T^i)_{i=1}^N$. With the target dataset $D$ and new samples $S_T$, we can retrain the classifier $P$. Similarly, we can also train new classifiers from scratch using datasets $D$ and $D \cup S_T$. Finally, we test the classifiers on the test set of the target domain.

Fig. 3 presents the accuracy of five transfer learning strategies on four pairs of source and target domain. For the labels above the plot, labels without $P$ mean training a new classifier from scratch, whereas labels with $P$ mean transferring the pre-trained classifier. $D$ and $S_T$ represent the samples in the target domain and our flowed samples. We observe a common trend that the addition of the flowed samples $S_T$ always improves the accuracy of the classifiers, as we compare $D \cup S_T$ with $D$ and compare $P \cup D \cup S_T$ with $P \cup D$. Moreover, the data augmentation with $S_T$ leads to a higher increase of accuracy for the 1-shot learning, where the data scarcity problem is more severe. The transfer learning results for SVHN and TIN datasets are provided in the Supplementary B.6. Although few-shot learning is more challenging due to the high complexity of the datasets, the addition of $S_T$ always improves the accuracy. We also compare with baseline$^3$, mixup method and image augmentation methods in Supplementary B.7.

**Conclusions.** This paper focuses on a gradient flow approach to generate new labeled data samples in the target domain. To overcome the discrete nature of the labels, we represent datasets as distributions on the feature-Gaussian space, and the flow is formulated to minimize the MMD loss function under an optimal transport metric. Contrary to existing gradient flows on linear structure, our flows are developed on the curved Riemannian manifold of Gaussian distributions. We provide explicit formula for the Riemannian gradient of MMD, and analyze in details the flow equations and the convergence properties of both continuous and discretized forms. The numerical experiments demonstrate that our method can efficiently generate high-fidelity labeled training data for real-world datasets, and improve the classification accuracy in few-shot learning. The main limitation exists in the assumption that the data of one label forms an elliptical distribution.

$^3$The only gradient flow work that has experiments on *NIST datasets, but it does not run experiments on TIN and SVHN.
**Ethical Statement**

Our work has positive societal impacts. Our work can enlarge the number of labelled data on any domains, so it alleviates the data scarcity problem in supervised learning settings. As a result, it can help reduce repetitive data collection and labeling work. It does not have possible negative societal impacts in the current stage.

**Acknowledgements**

We thank the anonymous reviewers for their comments. XH is supported by the Air Force Office of Scientific Research under award number FA9550-20-1-0397. Additional support from NSF 1915967, NSF 2118199, and the Ford-Stanford alliance is gratefully acknowledged. The research of TN is supported in part by a grant from the Simons Foundation (#318995). TL gratefully acknowledges the support of JSPS KAKENHI Grants 20K19873 and 23K11243. VAN acknowledges the generous support from the CUHK’s Improvement on Competitiveness in Hiring New Faculties Funding Scheme.

**References**


