Flow-Based Distributionally Robust Optimization

Chen Xu, Jonghyeok Lee, Xiuyuan Cheng, and Yao Xie, *Member, IEEE*

***Abstract*—We present a computationally efficient framework, called `FlowDRO`, for solving flow-based distributionally robust optimization (DRO) problems with Wasserstein uncertainty sets while aiming to find continuous worst-case distribution (also called the Least Favorable Distribution, LFD) and sample from it. The requirement for LFD to be continuous is so that the algorithm can be scalable to problems with larger sample sizes and achieve better generalization capability for the induced robust algorithms. To tackle the computationally challenging infinitely dimensional optimization problem, we leverage flow-based models and continuous-time invertible transport maps between the data distribution and the target distribution and develop a Wasserstein proximal gradient flow type algorithm. In theory, we establish the equivalence of the solution by optimal transport map to the original formulation, as well as the dual form of the problem through Wasserstein calculus and Brenier theorem. In practice, we parameterize the transport maps by a sequence of neural networks progressively trained in blocks by gradient descent. We demonstrate its usage in adversarial learning, distributionally robust hypothesis testing, and a new mechanism for data-driven distribution perturbation differential privacy, where the proposed method gives strong empirical performance on high-dimensional real data.**

***Index Terms*—Flow-based generative models, distributionally robust optimization.**

I. Introduction

DISTRIBUTIONALLY Robust Optimization (DRO) is a fundamental problem in optimization, serving as a basic model for decision-making under uncertainty and in statistics for addressing general minimax problems. It aims to identify a minimax optimal solution that minimizes an expected loss over the worst-case distribution within a pre-determined set of distributions (i.e., an uncertainty set). DRO arises from various applications, including robust hypothesis testing [23], [53], boosting [9], semi-supervised learning [7], fair classification [49], clustering [58], and so on; see [33] for a more complete review. Inherently, DRO leads to an infinite dimensional problem, and thus, it faces a significant computational challenge in most general settings. Despite the existing efforts to solve DRO that allow analytic or approximate solutions, current approaches still have limited scalability in solving high-dimensional, large-sample problems with general risk functions. In this work, we aim to address the computational challenge using a new neural network flow-based approach; the connection with existing approaches is further discussed in Section II-C.

The basic setup for DRO is given below. Let $\mathcal{X} = \mathbb{R}^d$ be the data domain. Assume a real-valued *risk function* $\mathcal{R}(P; \phi)$ taking as inputs a d-dimensional distribution P (with a finite second moment) and a measurable decision function $\phi \in \Phi$ in a certain function class (problem specific and possibly parametric). Assume a pre-specified scalar loss function $r : \mathcal{X} \times \Phi \to \mathbb{R}$ so that

$$\mathcal{R}(P; \phi) = \mathbb{E}_{x\sim P}[r(x; \phi)]. \tag{1}$$

Some examples of the decision function ϕ and loss function r include ϕ being a multi-class classifier and r being the cross-entropy loss, and ϕ being a scalar test function and r being the logistic loss. We are interested in solving the following minimax problem:

$$\min_{\phi\in\Phi} \max_{Q\in\mathcal{B}} \mathcal{R}(Q; \phi). \tag{2}$$

In (2), $\mathcal{B}$ is a pre-defined uncertainty set that contains a set of (possibly continuous) distributions that are variations from a *reference distribution* P; this is known as the distributionally robust optimization (DRO) problem [44]. In particular, we are interested in Wasserstein DRO or WDRO (see, e.g., the original contribution [38]), where the $\mathcal{B}$ is the Wasserstein uncertainty set centered around the reference distribution induced by Wasserstein distance. WDRO receives popularity partly due to its data-driven uncertainty sets and no parametric restriction on the distributional forms considered.

The worst-case distribution that achieves the saddle point in (2) is called the Least Favorable Distribution (LFD) (also called the "extreme distributions" in prior works, e.g., [38]). In this work, we consider the problem of finding LFD for a given algorithm ϕ, which is useful in various practical settings such as generating *worst scenarios* to test the algorithm and develop robust algorithms.

A. Proposed: Flow-DRO

In this paper, we propose a *computational* framework, a flow-based neural network called `FlowDRO` to find the worst-case distributions (LFDs) for DRO or solve the inner maximization of minimax problem (2). In particular, `FlowDRO` can efficiently compute worst-case distributions for

Manuscript received 29 October 2023; accepted 19 January 2024. Date of publication 27 February 2024; date of current version 26 March 2024. This work was supported in part by NSF CAREER under Grant CCF-1650913, Grant NSF DMS-2134037, Grant CMMI-2015787, Grant CMMI-2112533, Grant DMS-1938106, and Grant DMS-1830210; and in part by the Coca-Cola Foundation. The work of Xiuyuan Cheng was supported in part by NSF under Grant DMS-2237842, and in part by the Simons Foundation. *(Corresponding author: Yao Xie.)*

Chen Xu, Jonghyeok Lee, and Yao Xie are with the School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: yao.xie@isye.gatech.edu).

Xiuyuan Cheng is with the Department of Mathematics, Duke University, Durham, NC 27708 USA.

This article has supplementary downloadable material available at https://doi.org/10.1109/JSAIT.2024.3370699, provided by the authors.

Digital Object Identifier 10.1109/JSAIT.2024.3370699

various high-dimensional problems, thanks to the strong representation power of neural network-based generative models. The main idea is to connect the WDRO problem through Lagrangian duality to a function optimization problem with Wasserstein proximal regularization. This connection enables us to adapt the recently developed computationally efficient Wasserstein proximal gradient flow [14], [56] to develop computationally efficient *flow-based models* parameterized by neural networks. Our framework can be viewed as a generative model for LFDs. It is thus suitable for many statistical and machine learning tasks, including adversarial learning, robust hypothesis testing, and differential privacy, leading to computationally efficient solutions and performance gain, as we demonstrated using numerical examples.

Our main contributions are:

- Develop a new Wasserstein proximal gradient descent approach to find worst-case distributions (or Least Favorable Distributions, LFDs) in WDRO by reformulating the problem into its Wasserstein proximal form using Lagrangian duality. We introduce an alternative way to represent the LFDs through the *optimal transport maps* from a continuous reference measure to induce continuous LFD and use data to estimate.
- Algorithm-wise, we adopt a new neural-network generative model approach to find LFD, called `FlowDRO`. The proposed neural network-based method can be scalable to larger sample sizes and high dimensionality, overcoming the computational challenges of previous WDRO methods. `FlowDRO` parameterize LFD by a transport map represented by a neural network, which can learn from training samples and automatically generalizes to unseen samples and efficiently generate samples from the LFDs; we demonstrate its versatility in various applications and demonstrate the effectiveness of `FlowDRO` on multiple applications with high-dimensional problems (including images) from adversarial attack and differential privacy using numerical results.
- Theoretically, we approach the problem in a different route, relying on the tools of optimal transport: we derive the equivalence between the original $\mathcal{W}_2$-proximal problem and the transport-map-search problem making use of Brenier theorem enabled by considering continuous distributions. Our theory also shows that the first-order condition of our $\mathcal{W}_2$-proximal problem using Wasserstein calculus leads to an optimality condition of solving the Moreau envelope without assuming the convexity of the objective. As a by-product, we recover the closed-form expression of the dual function involving the Moreau envelope of the (negative) loss, consistent with existing work, and highlight the computational advantages of using our alternative optimal transport map reformulation.

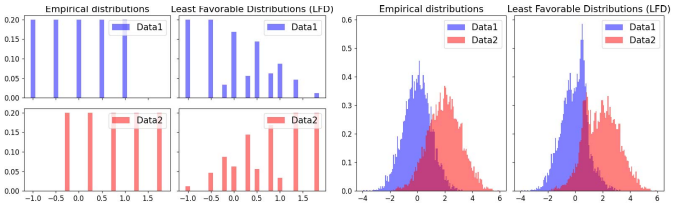
To the best of our knowledge, `FlowDRO` is the first work that finds the worst-case distributions in DRO using flow-based models. However, we would like to emphasize that our approach is general and does not rely on neural networks; one can potentially use an alternative representation of the transport maps (e.g., [29]) in low-dimensional and small sample settings for stronger learning guarantees. In the context of minimizing an objective functional in probability space, [31] proposed an infinite-dimensional Frank-Wolfe procedure. The work leveraged the strong duality result in DRO [8] (see more in Section II-A, Eqn. (8)) to compute the Wasserstein gradient descent steps. Our work focuses on the sub-problem of finding LFD in DRO, and our algorithm uses neural networks to tackle distributions in high dimensional space.

B. Motivating Example: Why Continuous Density for LFD?

One may quickly realize that finding LFD is an infinite-dimensional optimization problem that is particularly challenging in high dimensions and general risk functions. A useful observation that occurs in such infinite-dimensional optimization problem is that the worst-case distribution solution of the WDRO problem (2) turns out to be discrete, as shown in the original paper [38] and various follow-up works including [53] for the distributionally robust hypothesis test. This particular solution structure does help to overcome the computational challenge caused by the infinite-dimensional optimization problem.

However, the discrete nature of LFD, as coming from the WDRO formulation, is not desirable in practice, as explained in the following. First, there is a significant computational challenge. The method is not scalable to large datasets: the discrete WDRO formulation will require solving a Linear Program (LP) with the number of decision variables to be $\mathcal{O}(n^2)$, where n is the total number of training data points and the complexity of solving an LP is typically quadratic on the number of the decision variable. Such computational complexity for problems with thousands of training data points can be prohibitive (e.g., the MNIST handwritten digit example in our later section uses $\sim$5000 samples per class). So typically, the current WDRO formulation usually can only be used to find discrete LFDs for small sample settings (e.g., [53], [58]). Second, the discrete LFD will limit the *generalization* capability of the resulting algorithm. In machine learning applications, when we develop a robust detector (binary classifier) using DRO [23], [53], the LFD is discrete with a support on the training data set, as shown in Fig. 1(a). As a result, the optimal detector is also *only defined* on the support of training data points. Such an optimal detector does not generalize in that, given a new test sample, we cannot directly apply it to the test data if it does not coincide completely with one of the training data points. An ad-hoc approach could be to "interpolate" the optimal detector on the training samples by convolving with a smoothing kernel (such as a Gaussian kernel); however, this will lose the property of the original minimax optimality of the detector. It would be better to seek continuous worst-case distributions (LFDs) when we solve the minimax problem. Thus, we may want to add a constraint in the formulation and consider the uncertainty set as the intersection of the Wasserstein uncertainty set and the set of continuous functions.

Suppose we would like to find *continuous worst-case distribution* instead for the above consideration. However, if one restricts $\mathcal{P}$ in the minimax problem (2) to be the Wasserstein uncertainty set *intersecting all continuous distribution functions*, that will lead to an even more difficult



(a) WDRO on training samples (b) FlowDRO on test samples

Fig. 1. Comparison of WDRO and FlowDRO on the 1D example following [53, Figure 1]. **Left of (a) and (b):** Empirical distributions of two sets of *training* (shown in (a)) and *test* (shown in (b)) samples from $\mathcal{N}(0,1)$ and $\mathcal{N}(2,1.2)$. **Right of (a) and (b):** Least-favorable distributions (LFD) found by WDRO and FlowDRO, where LFDs are within the W_2 ball (4) with radius $\varepsilon = 0.1$. As expected, the LFDs overlap more with each other than the empirical distributions do.Note that WDRO solves a convex problem to obtain the LFD by moving the probability mass on *discrete training samples*. In particular, WDRO is not generalized to a test sample unless it coincides exactly with some of the training samples. In comparison, FlowDRO yields a one-to-one continuous-time transport map that can be directly applied to training and test samples. The resulting LFD is also continuous, as it is the push-forward distribution by the transport map on the underlying continuous data distribution.

infinite dimensional problem involving distribution functions, and the (discrete) solution structure property no longer holds. This brings out the main motivation of our paper: we will introduce *a neural network (NN) approach to solve minimax problem* leveraging the strong approximation power of NN and that they *implicitly regularize* the solution to achieve continuous density. To carry out the plan, we need a carefully designed NN architecture and training scheme leveraging the recent advances in *normalizing flow* to represent distribution functions. Recently, there have also been works considering entropy regularized Wasserstein uncertainty sets, called the Sinkhorn DRO problems [50], which lead to continuous LFDs with kernel-type solutions. Still, it is more suitable for low-dimensional problems due to the nature of the kernel solutions.

C. Flow-Based Generative Models

Recently, diffusion models and closely related flow-based models have drawn much research attention, given their state-of-the-art performance in image generation; see [14] for a complete summary. Flow-based generative models enjoy certain advantages in computing the data generation and the likelihood and have recently shown competitive empirical performance. They can be understood as continuous-time models that gradually transform the input distribution P into a target distribution Q. These models are popularized in the context of normalizing flow, where the target distribution $Q = \mathcal{N}(0, I_d)$, the standard multivariate Gaussian [32]. They can be largely categorized into discrete-time flows [5], [12], [54] and continuous-time flows [26], [40], [55], [56]. The discrete-time flows can be viewed as Euler approximation of the underlying continuous-time probability trajectory, where the continuous-time flows are based on neural ordinary differential equation (NeuralODE) [13] to learn the probability trajectory directly.

We remark that, unlike normalizing flow and flow between arbitrary pre-specified pairs of distributions, our flow model tries to learn the worst-case distribution Q^* that maximizes certain risk functions. Compared with other flow-based generative models, such as the traditional settings of normalizing flow or pre-specified target distributions, FlowDRO does not choose a target distribution *a-priori*, which is learned by maximizing the objective function.

We also note that different from training generative adversarial networks (GAN) [24] that may also generate worst-case samples, our flow-based approach can be more stable during training as it involves neither auxiliary discriminators nor inner loops. Compared with recent works on flow-based generative models [14], [56], where only KL divergence was considered for the loss function, we consider general loss as motivated by various applications.

D. Applications

FlowDRO can also directly benefit several applications, which can be formulated as DRO problems, as we present in more detail in Section V. First, in the case of an adversarial attack, our flow model is an *attacker* that can find the distribution causing the most disruption to existing systems. This is especially important for engineering system design. For example, in power systems, we are interested in understanding the resiliency of a power network. Given limited historical observations, we are interested in discovering whether any unseen scenario may cause a catastrophic consequence to the system. Finding such scenarios can help evaluate engineering systems and improve network resiliency. Second, in the case of differential privacy (DP), our flow model acts as a *distributional perturbation mechanism* to dataset queries. Upon finding the worst-case distribution around the data distribution over queries, we can provide much protection against potential data disclosure and/or privacy loss. This is extremely useful in high-stakes settings where sensitive information must be protected. We also note that the existing DP framework is largely not data-driven. Specifically, DP mechanisms often take the simple approach of adding i.i.d. noise to each dimension of queries, and the noises are unrelated to data. There is growing interest in developing data-driven mechanisms by exploiting the query structure or the data distribution, which may bring potential performance gains. However, finding such optimal perturbation subject to the privacy constraint poses a computational challenge, which we try to address through the proposed FlowDRO.

The rest of the paper is organized as follows. Section II formally introduces the framework of solving for the worst-case distribution, along with theoretical analyses. Section IV describes the FlowDRO method and the concrete training algorithm. Section V considers several important applications for which FlowDRO can be used. Section VI shows numerical results of FlowDRO on high-dimensional problems. Section VII concludes the work with discussions. All proofs are delegated to the appendix of https://arxiv.org/abs/2310.19253.

II. Framework

Below, we focus on Wasserstein-2 ($\mathcal{W}_2$) in this work, and extensions to $\mathcal{W}_p$ with other p are left to future studies. Let $\mathcal{X} = \mathbb{R}^d$, and denote by $\mathcal{P}_2(\mathcal{X})$ the space of all distributions on domain $\mathcal{X}$ that have a finite second moment, that is,

$\mathcal{P}_2(\mathcal{X}) := \{P, \int_{\mathcal{X}} |x|^2 dP(x) < \infty\}$. Define $\mathcal{P}_2^r(\mathcal{X}) := \{P \in \mathcal{P}_2, P \ll \text{Leb}\}$, that is, all distributions in $\mathcal{P}_2(\mathcal{X})$ that also have continuous densities (absolute continuous with respect to the Lebesgue measure). We may omit $(\mathcal{X})$ in the notation $\mathcal{P}_2$ and $\mathcal{P}_2^r$.

A. Dual Formulation and Wasserstein Proximal Problem

The $\mathcal{W}_2$-distance between two distributions in $\mathcal{P}_2$ is defined by

$$\mathcal{W}_2^2(\mu, \nu) := \inf_{\pi \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\|^2 d\pi(x, y), \tag{3}$$

where $\Pi(\mu, \nu)$ denotes the family of all joint distributions with μ and ν as marginal distributions, called the couplings of μ and ν. For any given $\nu \in \mathcal{P}_2$, the functional $\mathcal{W}_2^2(\cdot, \nu)$ maps from $\mathcal{P}_2$ to $[0, \infty)$, by the following lemma.

Lemma 1: For any $\mu, \nu \in \mathcal{P}_2$, $\mathcal{W}_2(\mu, \nu) < \infty$.

Let $\mathcal{B}_\varepsilon(P)$ be the $\mathcal{W}_2$-ball in $\mathcal{P}_2$ around the reference distribution P of radius $\varepsilon > 0$, namely

$$\mathcal{B}_\varepsilon(P) = \{Q \in \mathcal{P}_2, \mathcal{W}_2(Q, P) \le \varepsilon\}. \tag{4}$$

As explained in the introduction, we will focus on the case where P has (continuous) density, that is $P \in \mathcal{P}_2^r$. We focus on the inner loop (the "max") of the min-max problem (2) where the uncertainty set $\mathcal{B} = \mathcal{B}_\varepsilon(P)$. For fixed decision function ϕ, we cast the maximization as a minimization by defining $V(x) := -r(x; \phi)$. The central problem we aim to solve in the paper is to find the LFD, which can be equivalently written as the following:

$$\min_{Q \in \mathcal{B}_\varepsilon(P)} \mathbb{E}_{x\sim Q} V(x), \quad \texttt{\{LFD problem\}}. \tag{5}$$

The idea is to convert the uncertainty set constraint as a regularization term of the original objective function by introducing a Lagrangian multiplier. Then, we can leverage this connection to build a Wasserstein gradient flow type of algorithm to solve the LFD problem.

1) Dual Form and Proximal Problem: The constrained minimization (5) is a trust region problem. It is well known that in vector space, trust-region problem can be solved by a proximal problem where the Lagrangian multiplier defined through $\lambda > 0$ corresponds to the radius ε [42]. Specifically, consider the *dual form* of the LFD problem (5), which can be written as

$$\sup_{\lambda \ge 0} G(\lambda) := \min_{Q \in \mathcal{P}_2} \mathbb{E}_{x\sim Q} V(x) + \lambda\Big(\mathcal{W}_2^2(P, Q) - \varepsilon^2\Big), \quad \texttt{\{dual form\}}. \tag{6}$$

We restrict ourselves to the case when $\lambda > 0$, and introduce the change of variable $\lambda = \frac{1}{2\gamma}$ for $\gamma > 0$. After dropping the constant term $\lambda\varepsilon^2$ in (6), we obtain the following Wasserstein *proximal problem*

$$\min_{Q \in \mathcal{P}_2(\mathcal{X})} \mathbb{E}_{x\sim Q} V(x) + \frac{1}{2\gamma} \mathcal{W}_2^2(P, Q), \quad \texttt{\{proximal problem\}}. \tag{7}$$

The $\mathcal{W}_2$-proximal problem can be viewed as the Moreau envelope (or the Moreau-Yosida regularization) in the Wasserstein space [39]. Similar to the vector-space case, we will have a correspondence between (5) and (7), see Remark 2, which will be introduced in Section III after we derive the first-order optimality conditions of the two problems.

2) Explicit Form of Dual Function: It has been pointed out in several prior works that the dual form can be reformulated using the Moreau envelope of the (negated) loss function under different scenarios [8], [22], [57]. Specifically, the explicit expression of the dual form (6) is written as

$$\sup_{\lambda \ge 0} G(\lambda) := \mathbb{E}_{x\sim P} \inf_z \Big[V(z) + \lambda \|z - x\|^2\Big] - \lambda\varepsilon^2. \tag{8}$$

Assuming $\lambda > 0$, the dual function G in (8) can be equivalently written as

$$G\left(\frac{1}{2\gamma}\right) = \mathbb{E}_{x\sim P}\, u(x, \gamma) - \frac{\varepsilon^2}{2\gamma}, \tag{9}$$

where $u(x, \gamma)$ is the Moreau envelope of V defined as

$$u(x, t) := \inf_z \left[V(z) + \frac{1}{2t}\|z - x\|^2\right], \quad t > 0. \tag{10}$$

This form of the dual function echos the observation that the Wasserstein proximal operator for the functional in the form of $\varphi(\mu) = \int V d\mu$ can be solved by the proximal operator (Moreau envelope) of V, as has been pointed out in the PDE literature, see e.g., [10].

We will recover the same explicit form of the dual function under certain conditions in Section III where the Moreau envelope has unique minimizer z for each x, see Corollary 1. Meanwhile, from the computational perspective, the Moreau envelope $u(x, \gamma)$ may still be challenging to solve in high dimensions, among other algorithmic challenges. We further discuss this and the connections to previous studies of the dual form in Section II-C. Instead of using the dual form (8), we propose to solve the dual problem (equivalently the $\mathcal{W}_2$-proximal problem (7)) by parameterizing a transport map $T : \mathbb{R}^d \to \mathbb{R}^d$ possibly by a neural network, to be detailed in the next section.

B. Solving the Wasserstein Proximal Problem by Transport Map

We show that the problem (7) that minimizes over Q can be solved by minimizing over the transport map T, which will pushforward P to Q. (Recall that for $T : \mathcal{X} \to \mathcal{X}$, the *pushforward* of a distribution P is denoted as $T_\# P$, such that $T_\# P(A) = P(T^{-1}(A))$ for any measurable set A.) This reformulation is rooted in the Monge formulation of the Wasserstein distance.

When $P \in \mathcal{P}_2^r$, the Brenier theorem allows a well-defined and unique optimal transport (OT) map from P to any $\mu \in \mathcal{P}_2$. For completeness, we include the argument as follows. We denote by $T_P^\square$ the OT map from P to $\square \in \mathcal{P}_2$, which is defined P-a.e., and $(T_P^\mu)_\# P = \mu$. Given any $\mu \in \mathcal{P}_2$, for any $T : \mathbb{R}^d \to \mathbb{R}^d$ s.t. $T_\# P = \mu$, $(\mathrm{I_d}, T)_\# P$ is a coupling of P and μ, and thus

$$\mathcal{W}_2^2(\mu, P) \le \mathbb{E}_{x\sim P} \|x - T(x)\|^2. \tag{11}$$

The problem of minimizing the r.h.s. of (11) over all T that pushforwards P to μ is known as the Monge Problem. *By*

Brenier Theorem, when $P \in \mathcal{P}_2^r$, the OT map attains the minimum of the Monge problem, that is,

$$\mathcal{W}_2^2(\mu, P) \ = \mathbb{E}_{x\sim P}\|x - T_P^{\mu}(x)\|^2. \tag{12}$$

We introduce the following transport map minimization problem corresponding to the W_2-proximal problem (7)

$$\min_{T\,:\,\mathcal{X}\to\mathcal{X},\, T_{\#}P\in\mathcal{P}_2(\mathcal{X})} \mathbb{E}_{x\sim P}\left(V\circ T(x) + \frac{1}{2\gamma}\|x - T(x)\|^2\right). \tag{13}$$

The formal statement of the equivalence between (7) and (13) is by applying Proposition 1 with $\varphi(\mu) \coloneqq \mathbb{E}_{x\sim\mu}V(x)$, which is assumed to be finite for any $\mu \in \mathcal{P}_2(\mathcal{X})$, and $\lambda = 1/2\gamma > 0$. The proof follows a similar argument as in [56, Lemma A.1] and is included in Appendix A for completeness.

Proposition 1 (Equivalent Solution by Transport Map): Suppose $\varphi : \mathcal{P}_2(\mathcal{X}) \to (-\infty, \infty)$, $P \in \mathcal{P}_2^r(\mathcal{X})$, and define $L^2(P) \coloneqq \{v : \mathbb{R}^d \to \mathbb{R}^d,\ \mathbb{E}_{x\sim P}\|v(x)\|^2 < \infty\}$. For any $\lambda > 0$, the following two problems

$$\min_{\mu\in\mathcal{P}_2(\mathcal{X})} L_\mu(\mu) = \varphi(\mu) + \lambda\mathcal{W}_2^2(P, \mu), \tag{14}$$

$$\min_{T\in L^2(P)} L_T(T) = \varphi(T_{\#}P) \ + \lambda\mathbb{E}_{x\sim P}\|x - T(x)\|^2, \tag{15}$$

satisfy that

(a) If T^* is a minimizer of (15), then $(T^*)_{\#}P$ is a minimizer of (14).

(b) If μ^* is a minimizer of (14), then the OT map from P to μ^* minimizes (15).

In both cases, the minimum L_μ^* of (14) and the minimum L_T^* of (15) equal.

We will solve (13) by parameterizing the transport map T by a flow network on $[0, \gamma]$ and learn T by setting (13) as the training objective. Details will be introduced in section IV.

C. Connection to Existing Wasserstein DRO

The dual form (8) has been derived in several works under different settings [8], [22], [33], [38], [57] - noting that we define V to be the negative loss, thus (8) is "sup-inf", while the dual of the original LFD problem is "inf-sup". Below, we discuss the connection under our framework.

1) Reduction in the Case of Discrete Reference Measure: We show a connection of our problem to the known result in the literature (see, e.g., [33]): when the reference distribution P is discrete (rather than having a density, i.e., a continuous distribution considered in our setting), [33] proved a "strong duality" result (16). Here, we show that the dual form (6) will end up being in the same as the dual form therein (which is equivalent to (8)), and the argument is via (13) which illustrates the role played by the transport map T. This is an interesting connection because the dual form in [33, Th. 7] plays a role in reducing the original complex infinite-dimensional problem to a finite-dimensional problem to solve the discrete LFD [33], [38]. However, such reduction only happens when the center of the uncertainty set P is discrete; when P is not discrete rather than continuous, the case considered in our paper, we need to develop an alternative computational scheme.

When P is an empirical distribution (thus discrete), we denote $P = \hat{P}$ and $\hat{P} = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$, for a dataset $\{x_i\}_{i=1}^n$. We first restate [33, Th. 7] using our notations ($p = 2$ in $\mathcal{W}_p$):

$$\sup_{Q\in\mathcal{B}_\varepsilon(\hat{P})} \mathbb{E}_{x\sim Q} r(x, \phi) = \inf_{\lambda\geq 0}\left\{\mathbb{E}_{x\sim\hat{P}}\sup_z\left[r(z;\phi) - \lambda\|z - x\|^2\right] + \lambda\varepsilon^2\right\}. \tag{16}$$

Note that the dual form (the r.h.s. of (16)) is equivalent to (8) replacing P to be $\hat{P}$ (and swapping to "sup-inf").

Recall the dual form (6) where we take $P = \hat{P}$. After dropping the constant term $\lambda\varepsilon^2$, the following proposition gives the explicit expression of the dual function $G(\lambda)$. We believe similar arguments have appeared in the literature, and we include proof for completeness.

Proposition 2 (Dual Form for Discrete P): Given $\lambda > 0$, suppose $\forall i = 1, \ldots, n$, $\inf_z [V(z) + \lambda\|x_i - z\|^2]$ attains its minimum at some point $z_i \in \mathbb{R}^d$, then

$$\min_{Q\in\mathcal{P}_2} \mathbb{E}_{x\sim Q}V(x) + \lambda\mathcal{W}_2^2\left(\hat{P}, Q\right) = \mathbb{E}_{x\sim\hat{P}}\inf_z\left[V(z) + \lambda\|x - z\|^2\right]. \tag{17}$$

We thus have $G(\lambda) = \mathbb{E}_{x\sim\hat{P}}\inf_z [V(z) + \lambda\|x - z\|^2] - \lambda\varepsilon^2$. This dual function is equivalent to the dual form on the r.h.s. of (16), recall that $V(x) = -r(x; \phi)$.

It will be illustrative to derive the r.h.s. of (17) formally from the transport-map-search problem (13): with $P = \hat{P}$, we obtain

$$\min_{T\,:\,\mathbb{R}^d\to\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^n\left(V\circ T(x_i) + \lambda\|x_i - T(x_i)\|^2\right). \tag{18}$$

Since x_i are discrete points, the effective variable are $z_i \coloneqq T(x_i)$, that is, the minimization problem is equivalent to $\min_{\{z_i\}_{i=1}^n,\, z_i\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^n (V(z_i) + \lambda\|x_i - z_i\|^2)$. This minimization is decoupled for the n points z_i, and the minimization of each z_i This gives that $\min_{T\,:\,\mathbb{R}^d\to\mathbb{R}^d} \mathbb{E}_{x\sim\hat{P}}[V(T(x)) + \lambda\|x - T(x)\|^2] = \mathbb{E}_{x\sim\hat{P}}\inf_z [V(z) + \lambda\|x - z\|^2]$.

2) Connection to the Dual Formulation of WDRO: Prior works have also attempted to use the dual formulation to evaluate the *objective value* under the worst-case distribution $L \coloneqq \max_{Q\in\mathcal{B}} \mathcal{R}(Q; \phi)$. For example, the strong duality was obtained in a general setting in [8]. This approach is helpful to evaluate the objective function value under the worst-case distribution directly and, thus, can help to develop a robust algorithm $\phi(\theta)$ with respect to its parameter θ. However, the approaches along this line of thought may encounter certain limitations in practice. First, it is well understood that general functions do not admit explicit formulas for their proximal operators, that is, finding the Moreau envelop, namely finding the inner pointwise supermum problem $\sup_{z\in\mathcal{X}}\{r(z; \phi) - \lambda\|z - x\|^2\}$ does not have a closed-form solution. In cases where this inner-loop optimization is convex and differentiable, one can use standard iterative solvers to find the supremum z, yet this calls for a solution of z for each x point-wisely. When the objective r is non-linear and non-convex, there can be other algorithmic complications; see a recent discussion in [41].

In addition, computational challenges arise in evaluating the expectation $\mathbb{E}_{x\sim P}$ in (8). When the reference distribution P is not discrete, the expectation may not have a closed-form expression, or one may have to rely on sampling from P and perform a Sample Average Approximation (SAA), and the accuracy of SAA in high dimension relies on processing a large number of data samples. At last, even the formulation (8) can be useful for finding robust algorithms that minimize the worst-case loss, the LFD cannot be identified using the formulation, and one cannot sample from the LFD, which is desirable for applications such as adversarial scenario generation.

Theoretically, our analysis in this work obtains the dual form in a different route, primarily relying on the theoretical tools of optimal transport. We will derive the dual form (8) in Section III by showing that the first-order condition of the $\mathcal{W}_2$-proximal problem in Wasserstein calculus leads to an optimality condition of solving the Moreau envelope (Corollary 1); To justify the algorithm based on parameterizing the transport map, we derive the equivalence between the distribution-search problem (the original $\mathcal{W}_2$-proximal problem) and the transport-map-search problem in Proposition 1 making use of the Brenier theorem. These theoretical analyses utilize that the LFD has a continuous density.

III. Theory

In this section, we derive the first-order optimality condition for the `LFD problem` (5) and the `proximal problem` (7), when considering the primal formulation to find LFD. Although the derivation is elementary, such characterization seems to not exist in the literature as far as we know, and the characterization may shed some insights into the Wasserstein space nature of the problem. Moreover, the first-order conditions also help establish the dual form of the LFD problem.

A. Preliminaries

1) Notations: To state the main result, we first introduce some necessary notations. For a distribution P on $\mathbb{R}^d$, define the second moment $M_2(P) := \int_{\mathbb{R}^d} \|x\|^2 dP(x)$. Given $\mu \in \mathcal{P}_2$, the L^2 space denoted by $L^2(\mu)$ is for the vector fields $v : \mathbb{R}^d \to \mathbb{R}^d$. For $u, v : \mathbb{R}^d \to \mathbb{R}^d$, the inner-product $\langle u, v\rangle_\mu := \int_{\mathbb{R}^d} u(x)^T v(x) d\mu(x)$, and the L^2-norm is defined as $\|u\|_\mu^2 = \int_{\mathbb{R}^d} \|u(x)\|^2 d\mu(x)$.

We will use $v \in L^2(\mu)$ as a (small) displacement field; that is, we will consider the perturbation of μ to $(\mathrm{I_d} + v)_\# \mu$. By Lemma 2, if $v \in L_2(\mu)$, then $(\mathrm{I_d} + v)_\# \mu$ remains in $\mathcal{P}_2$.

Lemma 2: If $\mu \in \mathcal{P}_2$, $T \in L^2(\mu)$, then $T_\# \mu \in \mathcal{P}_2$.

We introduce notations of the following key functionals on $\mathcal{P}_2$,

$$\varphi(\mu) := \int_{\mathbb{R}^d} V(x) d\mu(x), \quad \psi(\mu) := \frac{1}{2}\mathcal{W}_2^2(\mu, P). \tag{19}$$

Then the `LFD problem` can be written as $\min_{Q\in\mathcal{P}_2,\, \psi(Q)\le\varepsilon^2/2} \varphi(Q)$. Because $\mathcal{P}_2$ lies inside the manifold of all distributions over $\mathbb{R}^d$, the notion of calculus and convexity of φ and ψ in $\mathcal{P}_2$ are very different from the case in vector space. However, it is reasonable to expect certain optimization results in vector space to find the analog here. The analysis here centers around the (sub)differential of φ and ψ in $\mathcal{P}_2$, which has been systematically studied in the analysis literature, see [2, Secs. IX and 10]. Our argument follows the constructions in [2], simplifying the notions and making the theoretical argument self-contained.

B. $\mathcal{W}_2$-Differentials

Recall that φ defined in (19) is a linear function of μ; however, being linear generally does not imply that the functional is "convex" on $\mathcal{P}_2$. Specifically, the convexity in $\mathcal{P}_2$ needs to be defined along geodesics (or general geodesics). As a simple example, $\mu_0 = \delta_{x_0}$ and $\mu_1 = \delta_{x_1}$, then the geodesic from μ_0 to μ_1 in $\mathcal{P}_2(\mathbb{R}^d)$ will consists the Dirac measure $\mu_t = \delta_{x_t}$, $t \in [0, 1]$ where x_t lies on the geodesic from x_0 to x_1 in $\mathbb{R}^d$ namely the line connecting the two points. For any $t \in [0, 1]$, $\varphi(\mu_t) = V(x_t)$. Then, unless the function V is convex, the functional $\varphi(\mu)$ will not be convex along the geodesic from μ_0 to μ_1.

We first introduce a lemma concerning the behavior of φ when the distribution is perturbed in $\mathcal{P}_2$. For $\varphi(\mu) = \int V d\mu$, we introduce the following assumption on the potential V (without assuming its convexity).

Assumption 1 (L-Smooth Loss): V is L-smooth on $\mathbb{R}^d$ for some $L > 0$, meaning that V is C^1 on $\mathbb{R}^d$ and ∇V is L-Lipschitz.

Lemma 3 (Strong Differential of φ): Under Assumption 1, φ:$\mathcal{P}_2 \to (-\infty, \infty)$. At any $\mu \in \mathcal{P}_2$, $\nabla V \in L^2(\mu)$ and φ has strong $\mathcal{W}_2$-differential $\nabla_{\mathcal{W}_2}\varphi(\mu) = \nabla V$, μ-a.e., in the sense that $\forall v \in L^2(\mu)$, $\|v\|_\mu = 1$, and $\delta \to 0+$,

$$\varphi((\mathrm{I_d} + \delta v)_\# \mu) = \varphi(\mu) + \delta\langle \nabla V, v\rangle_\mu + o(\delta). \tag{20}$$

For $\psi(\mu) = \frac{1}{2}\mathcal{W}_2^2(\mu, P)$, where $P \in \mathcal{P}_2^r$ is fixed, the $\mathcal{P}_2$ calculus is more conveniently derived in a neighborhood of $\mu \in \mathcal{P}_2^r$. It is known that the $\mathcal{W}_2$ differential (both sub- and super-differential) of ψ at $\mu \in \mathcal{P}_2^r$ has the expression as $(\mathrm{I_d} - T_\mu^P)$, see, e.g., [2, Corollary 10.2.7] where the subdifferential is defined not in the "strong" sense. Here, we give a lemma on the strong super-differential of ψ (i.e., strong subdifferential of $-\psi$), which suffices for our purpose.

Lemma 4 (Strong Super-Differential of ψ): Let $P \in \mathcal{P}_2$ be fixed, for any $\mu \in \mathcal{P}_2^r$, the optimal transport map T_μ^P is defined μ-a.e., and the functional $-\psi$ has strong $\mathcal{W}_2$-subdifferential at μ, $-(\mathrm{I_d} - T_\mu^P) \in \partial_{\mathcal{W}_2}(-\psi)(\mu)$, in the sense that $\forall v \in L^2(\mu)$, $\|v\|_\mu = 1$, and $\delta \to 0+$,

$$\psi((\mathrm{I_d} + \delta v)_\# \mu) \le \psi(\mu) + \delta\langle \mathrm{I_d} - T_\mu^P, v\rangle_\mu + o(\delta). \tag{21}$$

One remark is that, in the above lemma, we only need $P \in \mathcal{P}_2$ and no need to have density. The unique existence of the optimal transport map T_μ^P needs μ to have density.

C. First-Order Condition of `LFD Problem`

We will analyze the first-order condition around a local minimum of the `LFD problem` based on the relations (20) and (21). While (20) holds at any $\mu \in \mathcal{P}_2$, (21) requires $\mu \in$

$\mathcal{P}_2^r$, Thus, we assume the minimizer Q of the TR problem has density.

Assumption 2 (Minimizer of `LFD Problem` in $\mathcal{P}_2^r$): The problem (5) attains a (local) minimum at $Q \in \mathcal{P}_2^r$.

Remark 1: In our theory, we do not use the assumption $P \in \mathcal{P}_2^r$ explicitly, however, if P does not have density, then usually the minimizer Q will not have density, e.g., in the discrete LFD considered in [33], [38]. Thus, we assume P has a density so that Assumption 2 can be reasonable.

Theorem 1 (First-Order Condition of `LFD Problem`): Let $P \in \mathcal{P}_2$ be fixed, under Assumptions 1 and 2, at a local minimizer Q of (5) which is in $\mathcal{P}_2^r$,

(i) $\mathcal{B}_\varepsilon$ constraint not tight: If $\mathcal{W}_2(Q,P) < \varepsilon$, then $\nabla V = 0$, Q-a.e.

(ii) $\mathcal{B}_\varepsilon$ constraint tight: If $\mathcal{W}_2(Q,P) = \varepsilon$, then either $\nabla V = 0$, Q-a.e. or $\exists \lambda > 0$, s.t.,

$$\nabla V + \lambda\left(\mathrm{I_d} - T_Q^P\right) = 0, \quad Q\text{-a.e.} \tag{22}$$

Note that the statement of the proposition implies that $T_Q^P = \mathrm{I_d} + \frac{1}{\lambda}\nabla V$, when $\lambda > 0$, and otherwise $\nabla V = 0$, which takes the form of *complementarity condition*.

D. First-Order Condition of `Proximal Problem`

For any $\gamma > 0$, the first-order condition of the Wasserstein `proximal problem` (7) is derived in the following proposition.

Theorem 2: (First-Order Condition of `Proximal Problem`): Let $P \in \mathcal{P}_2$ be fixed, under Assumption 1, for $\gamma > 0$, suppose the problem (7) attains a (local) minimum at $Q \in \mathcal{P}_2^r$, then

$$0 = \nabla V + \frac{1}{\gamma}\left(\mathrm{I_d} - T_Q^P\right), \quad Q\text{-a.e.} \tag{23}$$

Remark 2 (Correspondence Between of `LFD Problem` and `Proximal Problem`): We can see that the condition (23) matches the first order condition (22) (when Wasserstein ball constraint is tight) by setting $\gamma = 1/\lambda$.

The $\mathcal{W}_2$-proximal problem has been studied in [2, Sec. X-A], and in particular, Lemma 10.1.2 derived a first-order condition (in terms of strong subdifferential of φ) at a minimizer. In our case, the strong $\mathcal{W}_2$-differential of φ exists at Q and thus the subdifferential uniquely exists, i.e., $\partial_{\mathcal{W}_2}\varphi = \{\nabla V\}$. Then the conclusion of [2, Lemma 10.1.2] directly implies (23). We include a direct proof of the proposition for completeness.

The first-order condition of the $\mathcal{W}_2$-proximal problem allows us to prove the explicit expression of the dual form (8), technically with small enough γ s.t. the Moreau envelope of V has unique minimizer in the $\inf_z$.

Corollary 1 Dual Form Let $P \in \mathcal{P}_2$ be fixed, under Assumption 1, for $0 < \gamma < \frac{1}{L}$, suppose the `proximal problem` (7) attains a local minimum at $Q \in \mathcal{P}_2^r$. Then, the Moreau envelope $u(x,\gamma)$ defined in (10) is solved at an unique minimizer z^* for each x, Q is a global minimum of the `priximal problem` (7), and the dual function G defined in `dual form` (6) has the expression as in (9).

Remark 3 (Interpretation of the Optimal Transport Map): When the optimal transport map from P to Q also exists, it can be interpreted as the map from x to z^*, which solves (the unique minimizer of) the Moreau envelope as well as a Backward Euler scheme to solve the continuous-time gradient flow. Specifically, when $P \in \mathcal{P}_2^r$, the optimal transport map T_P^Q is defined P-a.e., and $T_Q^P \circ T_P^Q = \mathrm{I_d}$, P-a.e. By Theorem 2, we have (23), which implies that

$$T_P^Q = \mathrm{I_d} - \gamma \nabla V \circ T_P^Q, \quad P\text{-a.e.} \tag{24}$$

By a similar argument as in the proof of Corollary 1, $z = T_P^Q(x)$ solves the unique minimizer of the Moreau envelope $u(x,\gamma) = \inf_z [V(z) + \frac{1}{2\gamma}\|z - x\|^2]$. To view the map T_P^Q as a Backward Euler scheme to solve the $\mathcal{W}_2$-proximal gradient descent: Suppose we use T_P^Q to pushforward from the current distribution $P_k = P$ to the next distribution $P_{k+1} = Q$, then each point x_k is moved to x_{k+1} by T_P^Q, i.e., $x_{k+1} = T_P^Q(x_k)$, then (24) gives that

$$x_{k+1} = x_k - \gamma \nabla V(x_{k+1}), \tag{25}$$

which is a Backward Euler scheme to integrate the continuous-time gradient descent ODE $\dot{x}(t) = -\nabla V(x(t))$ with step size γ.

IV. Algorithm: Flow-DRO

This section presents a neural network flow-based approach to solve the LFD problem by representing the optimal transport maps by ResNet blocks [27]. Our framework does not need to rely on neural networks; there can be other ways to represent the transport map (e.g., kernel representation). For high-dimensional data, with sufficient training data, neural networks tend to have competitive performance due to their expressiveness power. Below, in Section IV-A, we first parameterize the transport map T in (13) as the solution map of a NeuralODE [13]. In Section IV-B, we present the block-wise progressive training algorithm of the proposed flow model. In Section IV-C, we explain how `FlowDRO` can be used as an adversarial generative sampler. In Section IV-D, we propose an iterative algorithm to solve the original min-max DRO problem (2) with $\mathcal{B}$ being the $\mathcal{W}_2$ ball around P.

A. Flow-Based Neural Network Parameterization of Transport Map

Consider a density evolution (i.e., flow) $\rho(x,t)$ such that $\rho(x,0) = P$ at $t = 0$, and $\rho(x,t)$ approaches Q^* as t increases, where Q^* is the minimizer of (7) (unknown *a priori*). Below, we interchangeably refer $\rho(x,t)$ both as the marginal distribution of $x(t)$ and its corresponding density function. Given the initial distribution $\rho(x,0) = P$, such a flow is typically non-unique. We consider when the flow is induced by an ODE of $x(t)$ in $\mathbb{R}^d$:

$$\dot{x}(t) = f(x(t),t), \tag{26}$$

where $x(0) \sim P$. Note that by the Liouville equation (the continuity equation) (see, e.g., [14]), the marginal distribution $\rho(x,t)$ of $x(t)$ satisfies $\partial_t \rho + \nabla \cdot (\rho f) = 0$.

We choose to parameterize $f(x(t),t)$ in (26) by a neural network $f(x(t),t;\theta)$ with trainable parameters $\theta \in \Theta$ (using

continuous-time NeuralODE [13]). Assuming the flow map is within the unit interval $t \in [0, 1)$, the θ-parameterized solution map T can be expressed as

$$T(x;\theta) = x + \int_0^1 f\big(x(s'), s'; \theta\big)ds', x(s) = x. \tag{27}$$

Using (27), the problem of finding T in (13) thus reduces to training θ in the following problem:

$$\min_{\theta\in\Theta} \mathbb{E}_{x\sim P}\bigg(V \circ T(x;\theta) + \frac{1}{2\gamma}\|x - T(x;\theta)\|^2\bigg). \tag{28}$$

There are two main benefits of parameterizing T as a flow model with parameters θ. First, flow models are continuous in time so that we can obtain gradually transformed samples by integrating $f(x(s), s; \theta)$ over a smaller interval $[0, t)$ for $t < 1$. In practice, these gradually transformed samples can be directly compared against those obtained by other baselines, where numerical results are presented in Section VI. Second, compared to other popular generative models such as GAN [24], the proposed flow model based on NeuralODE can be simpler and easier to train. This is because our objective (28) involves no additional discriminators to guide the training of $T(\cdot;\theta)$, and therefore no additional inner loops are required.

We also note a close connection between training θ in (28) and training continuous normalizing flow (CNF) models with transport-cost regularization [21], [40], [54]. In CNF, the problem is to train θ so that $T(\cdot;\theta)_\# P$ is close to the isotropic Gaussian distribution $P_Z = \mathcal{N}(0, I_d)$. To do so, the CNF objective minimizes the KL-divergence $\mathrm{KL}(T(\cdot;\theta)_\# P||P_Z)$ up to constants, upon utilizing the instantaneous change-of-variable formula [13]. To ensure a smooth and regularized flow trajectory, the transport cost $\frac{1}{2\gamma}\|T(x;\theta) - x\|_2^2$ is also commonly used as a regularization term. Hence, the only difference between training our FlowDRO and a transport-regularized CNF model lies in the expression of the *first term* in (28): our FlowDRO minimizes $\mathbb{E}_{x\sim P}(V \circ T(x;\theta))$, which is guided by V dependent on the loss function r and decision function ϕ, while CNF minimizes the KL-divergence between $T(\cdot;\theta)_\# P$ and P_Z.

B. Block-Wise Progressive Training Algorithm

We propose a block-wise progressive training algorithm of minimizing (28) with respect to the network parameters θ. We build on the JKO-iFlow method in [56], originally developed for training normalizing flows. The convergence of JKO-type $\mathcal{W}_2$ proximal GD for learning a generative model (a special case when the loss function is the KL divergence between the data density and the multi-variate Gaussian distribution) is shown in [14].

Specifically, we would learn K optimal transports block-wise, where the k-th transport $T(\cdot,\theta_k)$ is parameterized by θ_k. After training, the final optimal transport map T_{final} is approximated by $T_K \circ \cdots \circ T_1$ for $T_k := T(\cdot;\hat{\theta}_k)$ with trained parameters $\hat{\theta}_k$; here for two mappings T_1, $T_2{:}\mathcal{X} \to \mathcal{X}$, $T_2 \circ T_1(x) = T_2(T_1(x))$. To perform block-wise progressive training, we first train θ_1 using (28) with the penalty parameter $\gamma = \gamma_1$. The expectation is taken over $x(0) \sim P$, the data distribution. Using the trained parameters $\hat{\theta}_1$, we could thus compute the push-forward distribution $P(1) = (T_1)_\# P$. This push-forward operation is done empirically by computing $x(1) = T_1(x(0)), x(0) \sim P$ using the first trained flow block. Then, we continue training θ_2 using (28) with $\gamma = \gamma_2$, where the expectation is taken over $x(1) \sim P(1)$. In general, starting at $P(0) = P$, we are able to train the $(k+1)$-th block parameters θ_{k+1} with $\gamma = \gamma_{k+1}$ given previous k blocks, where the expectation is taken over $x(k) \sim P(k)$.

Algorithm 1 Block-Wise Progressive Training of FlowDRO

Require: Regularization parameters $\{\gamma_k\}_{k=1}^K$, training data $\{x_i\} \sim P$

1: **for** $k = 1, \ldots, K$ **do**
2: Optimize parameters θ_k of $T(\cdot;\theta_k)$ by minimizing the sample average approximation (SAA) version of (28) using samples mapped through previous maps $\{T(\cdot;\hat{\theta}_i)\}_{i=1}^{k-1}$, and regularization parameter γ_k by setting $\gamma = \gamma_k$.
3: **end for**

Ensure: K trained flow blocks $\{T(\cdot;\hat{\theta}_k)\}_{k=1}^K$.

This leads to a block-wise progressive training scheme of the proposed FlowDRO, as summarized in Algorithm 1. Note that the regularization parameters $\{\gamma_k\}$ indirectly control the amount of perturbation, which is represented by the radius ε in the uncertainty set (4). Smaller choices of γ induce greater regularization and hence allow less perturbation of P by the flow model, whereas larger choices of γ impose less regularization on the amount of transport. Regarding the specification of these regularization parameters, we note that the desired specification varies across different problems, but setting an even choice (i.e., $\gamma_k = \gamma$) or changing by a constant factor (i.e., $\gamma_k = c\gamma_{k-1}$ for $c > 0$) typically work well in practice. To further improve the empirical performance, one can also consider adaptive step size using the time reparameterization technique [56], which is an attempt to encourage a more even amount of $\mathcal{W}_2$ transport cost by individual blocks.

The motivation of the progressive training scheme is to improve the end-to-end training of a single complicated block, especially when we allow a large $\mathcal{W}_2$ ball around P (e.g., due to a large γ in (28).) Specifically, compared to training a single large model, Algorithm 1 with multiple blocks helps reduce the memory and computational load because each small block has simpler architecture and is easier to train. This allows larger batch sizes and more accurate numerical ODE integrators when integrating $f(x(t), t; \theta_k)$ at block k. Furthermore, we note that the proposed FlowDRO is adaptive: one can always terminate after training a specific number of blocks, where termination depends on the current performance measured against some application-specific metric; in the case of assuming a small $\mathcal{W}_2$ ball around P, it could also be sufficient to terminate after training a single block.

We also discuss the computational complexity of Algorithm 1. We do so in terms of the number of function evaluations of the network $f(x(t), t; \theta)$ when computing (28),

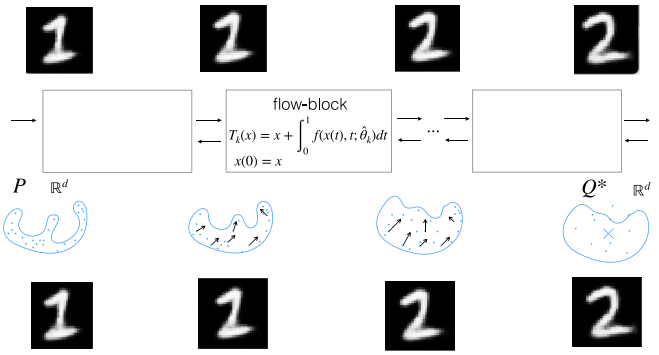


Fig. 2. An illustration of the FlowDRO framework, which learns a sequence of invertible optimal transport maps that pushes the underlying population density P to a target LFD Q^*; the maps are learned from finite training samples. The handwritten digits represent samples in each stage that show the gradual (continuous) transition of samples.

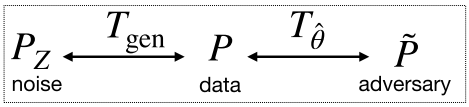


Fig. 3. Construction of the proposed sampler from LFD. After training the proposed FlowDRO $T_{\hat{\theta}}$, we train a separate generic flow model T_{gen} to map between the noise distribution P_Z (a multivariate Gaussian $\mathcal{N}(0, I_d)$) and the data distribution P. The full sampler $T_{\text{adv}} = T_{\hat{\theta}} \circ T_{\text{gen}}$.

as this is the most expensive step. Suppose at block k, we break the integral of $f(x(t), t; \theta_k)$ over $[0, 1)$ into $S \geq 1$ smaller pieces $\{[t_i, t_{i+1})\}_{i=0}^{S-1}$. Let the integral on each piece be numerically estimated by the fixed-stage Runge-Kutta fourth-order (RK4) method [47]. As a result, it takes $\mathcal{O}(4NS)$ evaluation of $f(x(t), t; \theta_k)$ on N samples per block. The total computation is on the order of $\mathcal{O}(4SKN)$ when training K blocks. Note that the overall computation is linear in the number of samples and thus scalable to large datasets.

C. Generative Model for Sampling From LFD

We now show how FlowDRO can be conveniently used to generate samples from LFD. This means that we can generate samples from the worst-case distribution Q^*, which is found as the push-forward distribution by FlowDRO. Specifically, let $T_{\hat{\theta}}$ be a trained FlowDRO model composed of K small blocks. Recall that $Q^* = (T_{\hat{\theta}})_{\#}P$ where P is the data distribution. Therefore, generating samples from the LFD Q^* is straightforward: one first obtains a new sample from $X \sim P$ and then computes $\tilde{X} = T_{\hat{\theta}}(X) \sim Q^*$. It remains to build a sampler for $X \sim P$. To do so, one can train an alternative generic flow model [26], [56] T_{gen} between P and P_Z, where P_Z is the standard multivariate Gaussian $\mathcal{N}(0, I_d)$, which is easy to sample from.

As a result, we can build the sampler from LFD as $T_{\text{adv}} = T_{\hat{\theta}} \circ T_{\text{gen}}$. This means we can first sample from multivariate Gaussian $Z \sim P_Z$, propagate it through the generic generative model T_{gen} to obtain a sample from P, and then propagate the sample through the map $T_{\hat{\theta}}$ to obtain a sample from the LFD Q^*, i.e., $\tilde{X} = T_{\hat{\theta}}(T_{\text{gen}}(Z)) \sim Q^*$. Figure 3 illustrates the idea.

Algorithm 2 Solving Min-Max Problem Using FlowDRO

Require: Regularization parameter γ, training data $\{x_i\} \sim P$, total iteration N, number of inner loops N_{inner}.

1: **for** $i = 1, \ldots, N$ **do**
2: Optimize T by minimizing the SAA of (28) for N_{inner} steps.
3: Optimize ϕ by minimizing the SAA of $\mathcal{R}(Q; \phi)$ defined in (1) over the LFD $Q = T_{\#}P$ for 1 step.
4: **end for**

Ensure: Trained models $(\hat{\phi}, \hat{T})$.

Meanwhile, we can also perform conditional generation, which is useful for classification problems. Suppose $X = (X_{\text{sub}}, Y)$ where $Y \in [C]$ is a discrete label for X_{sub}. To generate X_{sub} with its corresponding Y, we can follow the suggestion in [54] to train T_{gen}: let P_{sub} be the distribution of X_{sub}. Then, train a flow model to map between $P_{\text{sub}}|Y$ and $H|Y$, where $H|Y$ is a pre-specified Gaussian mixture in $\mathcal{P}_2^r(\mathcal{X})$. Hence, we can sample X_{sub} with label $c \in [C]$ by sampling from the corresponding $H|Y = c$ and mapping through T_{gen}. The sample X_{sub} can then be passed through $T_{\hat{\theta}}$ to get the sample with its corresponding label c from LFD.

D. Iterative Approach to Solve Min-Max Problem

While in this paper we focus on finding the LFD within the $\mathcal{W}_2$ ball around P (i.e., solve problem (5)), we hereby propose an iterative scheme that solves the min-max problem (2) that leads a pair of estimates $(\hat{\phi}, \hat{Q})$. In the context of supervised learning (e.g., classification), the solution $\hat{\phi}$ denotes a predictor robust against unobserved perturbation over input data to $\hat{\phi}$.

The high-level idea is as follows. We start from the samplable data distribution P and randomly initialized decision function ϕ and flow map T. We first update T by minimizing (28) to find the LFD $Q = T_{\#}P$. We then update ϕ by minimizing the risk $\mathcal{R}(Q; \phi)$ where $\mathcal{R}$ is defined in (1). We finally iterate the training of T and ϕ for some number of steps until the training converges. The procedure is summarized in Algorithm (2).

We further note the similarity and difference between Algorithm 2 and existing iterative DRO solvers (e.g., [46, Algorithm 1]). Both approaches iterate between finding the LFD and updating ϕ on samples from the LFD until convergence. The main difference lies in how the LFD is found. Our approach trains a continuous flow model T whose push-forward distribution $T_{\#}P$ is the LFD. In contrast, [46] solves the sample-wise LFD by iteratively moving inputs x_i along the gradient $\nabla_x[r(x; \phi) - \frac{1}{2\gamma}\|x - x_i\|^2]$. We empirically show the benefit of our proposed flow-based approach in Section VI-A2.

V. Applications

We consider several applications that can be formulated as DRO problems so that our proposed FlowDRO can be used to find the worst-case distribution.

A. Adversarial Learning With Distributional Attack

It has been widely known that state-of-the-art machine learning models are often adversarially vulnerable. Under small but carefully crafted perturbations to the inputs, the models can make severely wrong predictions on the adversarial examples [3], [48]. Adversarial training thus refers to the defense strategy in the clean training dataset augmented with adversarial examples, upon which retraining increases the robustness of the model on new adversarial examples.

Finding suitable adversarial examples before retraining is a critically important step. Most methods, such as the widely-used FGSM [25] and PGD [36], are based on the *point-wise* attack. We can show that the solution of the W_2 trust-region problem (5) more effectively "disrupts" a fixed decision function ϕ than the solution induced by the transport map of point-wise attack. Specifically, let $\phi \in \Phi$ be a fixed decision function. Given $x \in \mathcal{X}$, we define $T_{\text{point}} : \mathbb{R}^d \to \mathbb{R}^d$ as the transport port associated with the following point-wise perturbation problem:

$$T_{\text{point}}(x) := x + \delta_x^*,\ \ \delta_x^* = \arg\max_{\|\delta_x\|_2 \le \varepsilon} r(x + \delta_x, \phi). \tag{29}$$

Denote $Q^*_{\text{point}} = (T_{\text{point}})_{\#}P$ as the push-forward distribution by T_{point} on the data distribution P. Denote Q^*_{dist} as the solution of the W_2 trust-region problem (5), which is our objective of interest. We thus have the following result in terms of reaching *higher risk* under Q^*_{dist}, the proof is in Appendix A.

Proposition 3: For a fixed decision function ϕ, we have $\mathcal{R}(Q^*_{\text{dist}}, \phi) \ge \mathcal{R}(Q^*_{\text{point}}, \phi)$.

Several works have also considered *distributional* attacks on the input distribution to extend beyond point-wise attacks. For example, [46] uses the Wasserstein distance to measure the difference between input and adversarial distribution. It then proposes to solve a Lagrangian penalty formulation of the distributional attack problem by stochastic gradient methods with respect to the inputs x. Additionally, [11] shows the generality of such distributional attack methods by subsuming different point-wise attack methods under the distributional attack framework under a new Wasserstein cost function. While these works share the similar goal of solving for adversarial distributions, the proposed solutions do not solve for a continuous-time transport map as we intend to do, whose push-forward distribution of P yields the worst-case distribution.

We now formally introduce the adversarial learning problem under the current DRO framework, using image classification as a canonical example [25]. Let $X = (X_{\text{img}}, Y), X \sim P$ be an image-label pair with raw image X_{img} and its label $Y \in [C]$. The decision function ϕ is typically chosen as a C-class classifier taking X_{img} as the input, and the loss function $r(X, \phi) = -\log(\phi(X_{\text{img}})_Y)$ is the cross-entropy loss. To find an alternative distribution Q^* on which the risk is high, it is conventional to keep Y the same and perturb the corresponding X_{img}. Thus, for a given image-label distribution P, let $P_{\text{img}} = \{X_{\text{img}} : X = (X_{\text{img}}, Y), X \sim P\}$. As a result, the W_2 ball $\mathcal{B}_\varepsilon(P)$ with radius ε around the data distribution P is defined as

$$\mathcal{B}_\varepsilon(P) = \{Q \in \mathcal{P}_2(\mathcal{X}) : W_2^2\big(Q_{\text{img}}, P_{\text{img}}\big) \le \varepsilon^2\}. \tag{30}$$

Let Φ be the set of C-class classifiers on images X_{img}. The DRO problem under $\mathcal{B}_\varepsilon(P)$ in (30) is

$$\min_{\phi \in \Phi} \max_{Q \in \mathcal{B}_\varepsilon(P)} \mathbb{E}_{X \sim Q}\big[-\log\big(\phi(X_{\text{img}})_Y\big)\big]. \tag{31}$$

B. Robust Hypothesis Testing

The goal of hypothesis testing is to develop a detector which, given two hypotheses H_0 and H_1, discriminates between the hypotheses using input data while reaching a small error probability. In practice, true data distribution often deviates from the assumed nominal distribution, so one needs to develop robust hypothesis testing procedures to improve the detector's performance. The seminal work by [28] considers the problem of using ϵ-contamination sets, which are all distributions close to the base distributions in total variation. Later, [35] considers uncertainty sets under the KL-divergence and develops robust detectors for one-dimensional problems. More recently, [23] developed data-driven robust minimax detectors for non-parametric hypothesis testing, assuming the uncertainty set is a Wasserstein ball around the empirical distributions. In addition, [51] derives the optimal detector by considering Sinkhorn uncertainty sets around the empirical distributions. Compared to robust detectors under Wasserstein uncertainty sets, the Sinkhorn-based method is applicable even if the test samples do not have the same support as the training samples.

We follow the notations in [23] to introduce the problem. Given data $X \in \Omega$, we test between $H_0 : X \sim Q_0, Q_0 \in \mathcal{B}_{0,\varepsilon}(P_0)$ and $H_1 : X \sim Q_1, Q_1 \in \mathcal{B}_{1,\varepsilon}(P_1)$, where $\mathcal{B}_{i,\varepsilon}(P_i)$ denotes the W_2 ball of radius ε as in (4) around the corresponding data distribution P_i. Then, we find a measurable scalar-valued detector $\phi : \Omega \to \mathbb{R}$ to perform the hypothesis test. Specifically, for a given observation $X \in \Omega$, ϕ accepts H_0 and rejects H_1 whenever $\phi(X) < 0$ and otherwise rejects H_0 and accepts H_1. In this problem, the risk function $\mathcal{R}((Q_0, Q_1), \phi)$ is defined to provide a convex upper bound on the sum of type-I and type-II errors. Specifically, consider a so-called *generating function* f that is non-negative, non-decreasing, and convex. The risk is thus defined as

$$\begin{aligned}\mathcal{R}((Q_0, Q_1), \phi) = \mathbb{E}_{x \sim Q_0}\big[f \circ (-\phi)(x)\big] \\ + \mathbb{E}_{x \sim Q_1}\big[f \circ \phi(x)\big].\end{aligned} \tag{32}$$

Examples of the generating function f to defined (32) include $f(x) = \exp(t), f(x) = \log(1 + \exp(t)), f(x) = (t+1)_+^2$, and so on. As a result of $\mathcal{R}$ in (32), the robust hypothesis testing can be formulated as the following DRO problem

$$\begin{aligned}\min_{\phi\ :\ \Omega \to \mathbb{R}}\ \max_{Q_i \in \mathcal{B}_{i,\varepsilon}(P_i), i=0,1}\ & \mathbb{E}_{x \sim Q_0}\big[f \circ (-\phi)(x)\big] \\ & + \mathbb{E}_{x \sim Q_1}\big[f \circ \phi(x)\big].\end{aligned} \tag{33}$$

Solving the inner maximization of (33) requires finding a pair of worst-case distributions Q_0^* and Q_1^*. However, using the change-of-measure technique [23, Th. 2], we can solve an equivalent problem of finding Q^* within a W_2 ball round the data distribution $P = P_1 + P_2$ to fit our original formulation (2).

C. Differential Privacy

Established by [17], [18], differential privacy (DP) offers a structured method to measure how well individual privacy is secured in a database when collective data insights are shared as answers to the query. In short, DP upholds robust privacy assurances by ensuring that it is nearly impossible to determine an individual's presence or absence in the database from the disclosed information. These can be realized by introducing random perturbations to the query function output before release.

To be more precise, consider datasets $D, D' \in \mathcal{D}^n$ where each consists of n rows, and $\mathcal{D}$ is the space where each datum lies. We say D and D' are neighboring datasets if they differ in exactly a single element (i.e., in the record of one individual), and we denote $D \simeq D'$. An output of the query function $q : \mathcal{D}^n \to \Omega$ is given based on the dataset. A randomized mechanism $M : \mathcal{D}^n \to \Omega$, which maps a dataset to a random output under the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, imparts randomness to the answer to the query by perturbing $q(D)$. Differentially private randomized mechanisms secure privacy by ensuring that the outputs of M from neighboring datasets are nearly indistinguishable.

The most represented standard for DP is (ϵ, δ)-DP [17] (without causing confusion, here ϵ is not related to the radius of the uncertainty set ε). Given $\epsilon, \delta \geq 0$, a randomized mechanism M is (ϵ, δ)-differentially private, or (ϵ, δ)-DP, if $\mathbb{P}(M(D) \in A) \leq e^{\epsilon}\mathbb{P}(M(D') \in A) + \delta$ for any $D \simeq D' \in \mathcal{D}^n$ and $A \in \mathcal{F}$. When $\delta = 0$, we simply say that M is ϵ-DP. Besides, numerous variants of DP with rigorous definitions such as f-DP [16], Renyi DP [37], and Concentrated DP [20] have been established and studied; for a comprehensive overview, see [15].

The randomized mechanisms exhibit a clear trade-off: the more they secure privacy, the more they sacrifice statistical utility [1]. Therefore, the constant focus of research has been to design mechanisms that minimize the perturbation and thus the loss of utility (based on specific criteria such as l_p cost) while ensuring a certain level of privacy. Below, we borrow the notion of DP to conceptualize the design of a privacy protection mechanism as a DRO problem and propose the potential applicability of our `FlowDRO` as a data-dependent distributional perturbation mechanism.

DP can be understood as a hypothesis-testing problem [4], [16], [30], [52]. Consider an adversary trying to differentiate between neighboring datasets D and D' based on the mechanism output. In this context, the hypothesis testing problem of interest is

$$H_0 : X \overset{d}{=} M(D) \sim Q_0 \quad \text{vs.} \quad H_1 : X \overset{d}{=} M\big(D'\big) \sim Q_1 \tag{34}$$

where $X \in \Omega$ is a single perturbed observation. The harder this test is, the more difficult it is to distinguish between neighboring datasets, which implies that strong privacy is ensured. Consider testing (34) with a decision function $\phi:\Omega \to [0, 1]$, and denote the type-I and type-II errors as $\alpha_\phi = \mathbb{E}_{X\sim Q_0}\phi(X)$ and $\beta_\phi = \mathbb{E}_{X\sim Q_1}(1-\phi(X))$. Then, a mechanism is (ϵ, δ)-DP if and only if $\alpha_\phi + e^{\epsilon}\beta_\phi \geq 1-\delta$ and $e^{\epsilon}\alpha_\phi + \beta_\phi \geq 1-\delta$ for any $D \simeq D'$ and decision function ϕ that is a deterministic function of X [[52, Th. 2.4]; [30, Th. 2.1]].

Now, we first set up an optimization problem with the risk function measuring indistinguishability between Q_0 and Q_1 in (34), given the restricted level of perturbation and the neighboring datasets D and D'. Consider a risk function $\mathcal{R}((Q_0, Q_1), \phi)$ representing the ease of (34) with a decision function $\phi : \Omega \to [0, 1]$. To ensure strong privacy with a randomized mechanism, even in the "worst-case scenario" with a powerful discriminator, one should make it difficult to distinguish Q_0 and Q_1 by bringing the two distributions closely together, thereby reducing the risk function. Hence, finding such a pair of indistinguishable distributions with perturbation levels controlled by the Wasserstein-2 distance reduces to

$$\min_{Q_i\in\mathcal{B}_{i,\varepsilon}(P_i),i=0,1} \max_{\phi\in\Phi} \mathcal{R}((Q_0, Q_1), \phi) \tag{35}$$

where $\mathcal{B}_{i,\varepsilon}(P_i)$ denotes the $\mathcal{W}_2$ ball of radius ε as in (4) around the corresponding data distribution P_i.

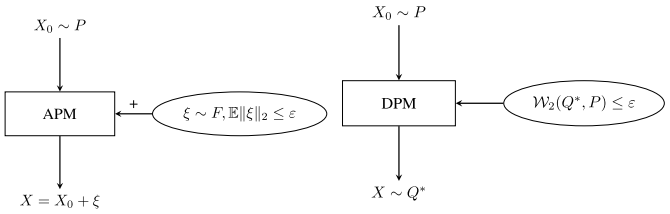
In this context, the risk function can be chosen based on which measure reflects the indistinguishability of outputs from neighboring datasets. For instance, under the f-DP criterion, one must first consider the most powerful test for a given level α: the decision function that minimizes β_ϕ. The corresponding problem is formulated as finding $\min_\phi \beta_\phi$ subject to $\alpha_\phi \leq \alpha$. Therefore, using the Lagrange multiplier and the change-of-measure technique, our DRO formulation (35) becomes $\max_{Q_i\in\mathcal{B}_{i,\varepsilon}(P_i),i=0,1} \min_\phi \max_{\lambda\geq 0} -\mathbb{E}_{x\sim Q_0+Q_1}[(dQ_1/[d(Q_0 + Q_1)])[x]\phi(x) - \lambda([dQ_0/(d(Q_0 + Q_1))][x]\phi(x) - \alpha)]$. In our experiments, we will use α_ϕ and β_ϕ as performance measures by replacing them with sample average approximations.

The conventional and straightforward method to privatize a query function is to apply a calibrated additive noise. In this case, the i-th uncertainty set in (35) is $\mathcal{B}_{i,\varepsilon}(P_i) = \{Q_i{:}M(D) \sim Q_i, M(D) = q(D) + \xi_i, q(D) \sim P_i, D \in \mathcal{D}^n\}$, where ξ_i with $\mathbb{E}\|\xi_i\|_2 \leq \varepsilon$ is a random noise following certain distributions from a specific family. We call such a mechanism that adds noise of a certain distribution an *additive perturbation mechanism* (APM). Typical noise distributions used in APM include the Laplace [18] and Gaussian distributions [19].

In contrast, based on the formulation (35), we aim to introduce distributional perturbation with our `FlowDRO` to provide a more flexible mechanism. Consequently, we want to ensure the mechanism outputs are indistinguishable with less perturbation than additive mechanisms. We refer to the corresponding mechanism as the *distributional perturbation mechanism* (DPM) and illustrate its comparison with APM in Figure 4. We remark that the proposed `FlowDRO` allows the DPM to apply an arbitrary amount of perturbation to the original distribution of queries. Thus, we can apply DPM at arbitrary precision by controlling the perturbation to satisfy the privacy constraints with reasonable utility.

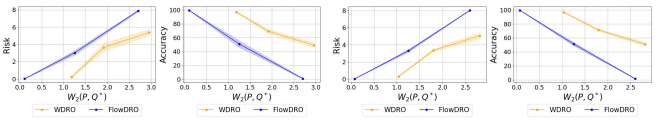
VI. Numerical Examples

We conduct experiments to examine the effectiveness of `FlowDRO` on high-dimensional data. First, in Section VI-A, we compare our proposed `FlowDRO` with existing DRO methods to solve robust hypothesis testing problems and train



(a) Additive perturbation mechanism (APM) (b) Distributional perturbation mechanism (DPM)

Fig. 4. Comparison between APM and DPM for differential privacy. APM adds random noises ξ *independently* to queries, whereas DPM (through the use of proposed `FlowDRO`) considers the data distribution P defined over all queries to find a worst-case distribution Q^* within $\mathcal{B}_\varepsilon(P)$.



(a) Training size $n = 200$ (b) Training size $n = 500$

Fig. 5. Test performance of a pre-trained MNIST classifier ϕ on LFDs Q^* of binary MNIST digits; the higher the risk, and the lower the accuracy, the better, meaning we have achieved a more effective attack for the same amount of Wasserstein-2 perturbation from the nominal distribution. WDRO and `FlowDRO` find the LFDs within different W_2 balls around P, which consists of n training data from MNIST. The empirical W_2 distances upon solving the earth moving distance between P and Q^* are shown on the x-axis.

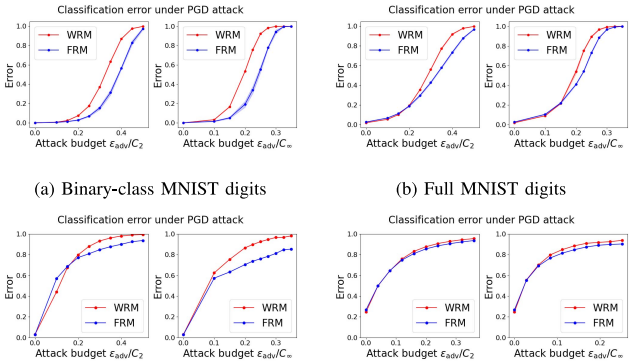
robust classifiers. Then, in Section VI-B, we use `FlowDRO` to perform the adversarial attack on pre-trained image classifiers and compare against existing point-wise attack methods. In Section VI-C, we use `FlowDRO` as the DPM in differential privacy settings and compare it against APM under different noise distribution specifications. In all examples of finding the LFD, we assume the decision function ϕ is pre-trained on the data distribution P and fixed, so the goal is to find the worst-case distribution $Q^* \in \mathcal{B}_\varepsilon(P)$ defined in (4) and compare what `FlowDRO` found against that by other methods. Code is available on https://github.com/hamrel-cxu/FlowDRO.

A. Comparison With Existing DRO Methods

We first compare the proposed `FlowDRO` in Algorithm 1 against WDRO [53] in finding LFD. We then compare the DRO solver 2 against the Wasserstein Robust Method (WRM) [46]. Further details of the experiments are in Appendix B.

1) Finding LFD: We consider binary MNIST digits from classes 0 and 8 as an example. Given a pre-trained CNN classifier, the goal is to find the LFD around the original digits. We measure the effectiveness of the LFD according to how the pre-trained classifier performs: the found LFD is more effective if, at the same level of W_2 perturbation, the classifier reaches a lower test prediction accuracy and a higher test risk on samples from that LFD.

Figure 5 shows the test risk and accuracy of the pre-trained classifier on the LFDs obtained by WDRO and `FlowDRO`. We see that compared to WDRO, our proposed `FlowDRO` finds more effective LFDs with the same or even smaller *budget*, which is measured as the empirical W_2 distance between P and Q^*, the found LFD. The benefit of `FlowDRO` holds both small ($n = 200$) and large ($n = 500$) sample sizes.



(a) Binary-class MNIST digits (b) Full MNIST digits

(c) Binary-class CIFAR10 images (d) Full CIFAR10 images

Fig. 6. Test classification error of robust classifiers on test data attacked by PGD under ℓ_2 and ℓ_∞ norm. The lower the error, the better, meaning we have achieved a more robust algorithm at the same attack budget. The robust classifiers are trained via solving the DRO problem using WRM [46] and FRM (ours in Algorithm 2). The binary classification results are for two randomly selected classes out of ten. The attack budget on the x-axis denotes the ℓ_p norm between raw and PGD-attacked test data as a fraction of C_p, the ℓ_p norm of raw test data.

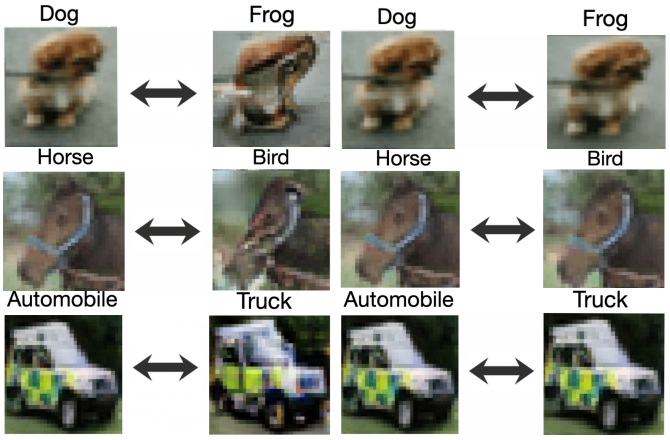
2) Training Robust Classifiers: We consider both MNIST digits and CIFAR10 images as examples. The goal is to train a robust classifier ϕ so that when test images are attacked by PGD under ℓ_p norms, the classifier can defend against such attacks by incurring a small classification error. Hence, one classifier is more robust than another when it reaches a smaller classification error on the same set of attacked test images.

Figure 6 shows test errors by robust classifiers trained via WRM [46] and via our proposed *flow robust method* (FRM) in Algorithm 2. We can see that for small attacks (for instance, when the attack budget is below 0.2), our method is slightly better than WRM (except for one case of CIFAR-10 binary class). However, for "higher" attacks, our FRM shows significantly better performance. The experiments show the effectiveness of our methods in obtaining an overall more robust classifier.

B. Adversarial Distributional Attack

We consider two sets of experiments in this section. The first example finds the distributional perturbation of CIFAR-10 images by `FlowDRO`, where we compare the effectiveness of our distributional attack against the widely-used projected gradient descent (PGD) baselines under ℓ_2 and ℓ_∞ perturbation [36]. The second example finds the distributional perturbation of MNIST digits by `FlowDRO`. Further details of the experiments are in Appendix C.

1) CIFAR10 Against Point-Wise Attacks: The goal is to show that `FlowDRO` can yield more effective attacks than point-wise attack baselines. Table I quantitatively compares the risk and accuracy of the pre-trained classifier ϕ on CIFAR10. We notice that under the same amount of ℓ_2 perturbation between raw and perturbed images, ϕ on the adversarial distribution found by `FlowDRO` yields significantly larger risk and lower accuracy. Hence, we conclude that `FlowDRO`



(a) Raw (left) & adversarial (right) samples by FlowDRO (b) Raw (left) & adversarial (right) samples by PGD-ℓ_2

Fig. 7. Raw and adversarial samples found by FlowDRO and by PGD-ℓ_2. Captions show prediction by the pre-trained classifier ϕ on raw input images $X_{\text{test, img}}$ before attack and adversarial samples $\tilde{X}_{\text{test, img}}$ after attack. FlowDRO results in more meaningful contextual changes in the raw images.

TABLE I
RISK AND ACCURACY OF A PRE-TRAINED VGG-16 CLASSIFIER ϕ ON CLEAN TEST DATA DISTRIBUTION P_{TEST} AND ADVERSARIALLY PERTURBED DATA DISTRIBUTION Q^*_{TEST} BY FLOWDRO AND BY PGD UNDER ℓ_2 AND ℓ_∞ PERTURBATION. FOR A FAIR COMPARISON, WE CONTROL THE SAME AMOUNT OF ℓ_2 PERTURBATION ON THE TEST DISTRIBUTION BY DIFFERENT ATTACKERS

	Clean data	Attack by FlowDRO	Attack by PGD-ℓ_2	Attack by PGD-ℓ_∞
Risk of ϕ in (A.25)	2.03	32.32	6.22	10.51
Accuracy of ϕ in (A.26)	87.02	24.22	61.44	41.57

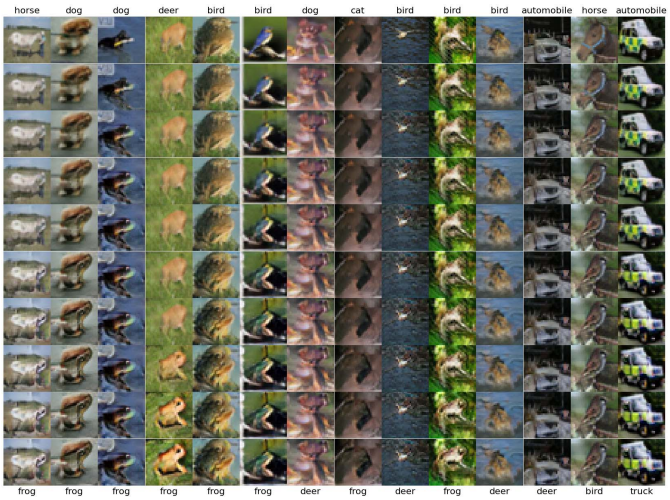
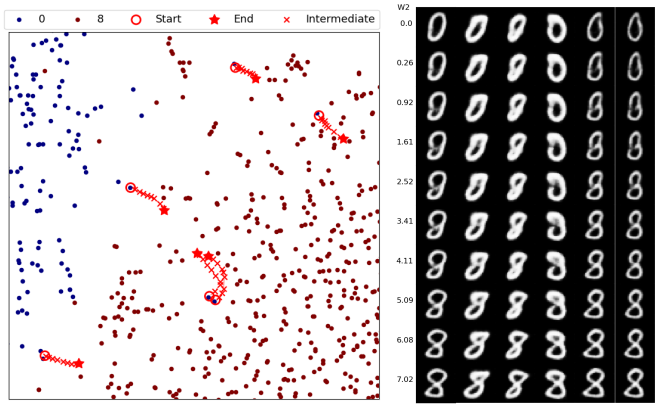


Fig. 8. Trajectory of FlowDRO adversarial attacks on different $X_{\text{test, img}}$ (shown as columns) to $\tilde{X}_{\text{test, img}}$. We visualize the changes as rows over three FlowDRO blocks, each of which breaks [0, 1) into three evenly spaced sub-intervals, resulting in nine integration steps along the perturbation trajectory. Captions on the top and bottom indicate predictions by the pre-trained ϕ on raw $X_{\text{test, img}}$ and final perturbed adversarial $\tilde{X}_{\text{test, img}}$.

performs much more effective attacks than the PDG baselines. Meanwhile, Figure 7 visualizes the qualitative changes to test images $X_{\text{test, img}}$ by FlowDRO and PGD, where the proposed FlowDRO also induces more meaningful contextual changes to the input image. Lastly, Figure 8 visualizes the gradual



(a) Digit deformation trajectories under T-SNE embedding (b) Transition of $X_{\text{test,img}}$ in pixel space

Fig. 9. FlowDRO perturbation of MNIST digits over blocks and integration steps. Figure (a) visualizes the perturbation trajectories from digits 0 to 8 under 2D T-SNE embedding. Figure (b) shows the trajectory in pixel space, along with the corresponding W_2 distance between original and perturbed images over integration steps.

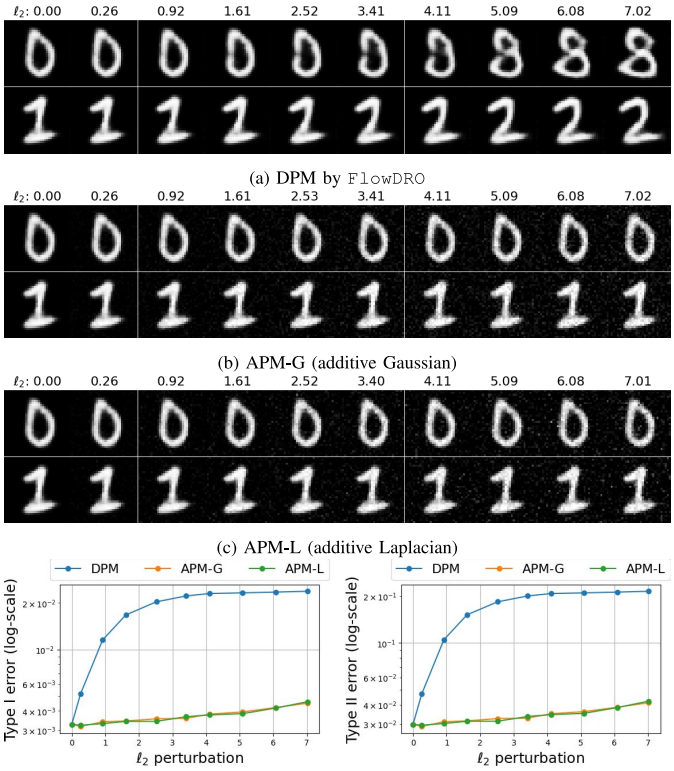
changes of $X_{\text{test, img}}$ over blocks and their integration steps by FlowDRO, demonstrating the continuous deformation by our trained flow model on test images $X_{\text{test, img}}$.

2) MNIST Trajectory Illustration: We now apply FlowDRO on finding the worst-case distribution, given a pre-trained LeNet classifier [34] ϕ. In this example, we focus on providing more insights into the behavior of FlowDRO without comparing it against other baselines. Figure 9 visualizes the gradual and smooth perturbation of test images $X_{\text{test, img}}$ by FlowDRO. We notice the cost-effectiveness and interpretability of FlowDRO. First, the T-SNE embedding in Figure 9 shows that FlowDRO tends to push digits around the *boundary* of certain digit clouds to that of other digit clouds, as such changes take the least amount of transport cost but can likely induce a great increase of the classification loss by ϕ. Second, changes in the pixel space in Figure 9 show that visible perturbation is mostly applied to the foreground of the image (i.e., actual digits), as changes in the foreground tend to have a higher impact on the classification by ϕ.

C. Data-Driven Differential Privacy

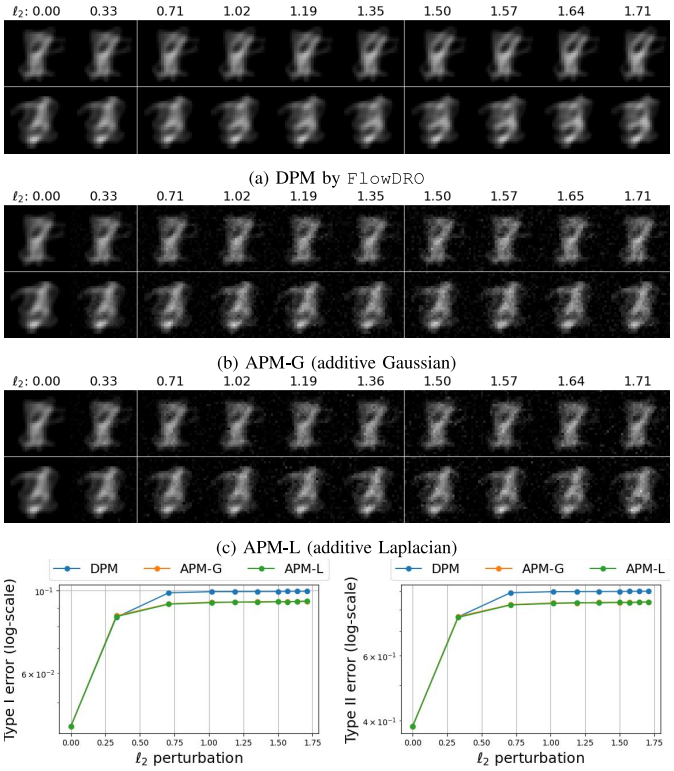
This section demonstrates the benefit of our FlowDRO DPM in privacy protection. We specifically focus on the examples of image recognition based on MNIST, where the decision function ϕ is specified as pre-trained classifiers. We mainly compare DPM against two APM baselines: APM under Gaussian noise (APM-G) and APM under Laplacian noise (APM-L). Further details of the experiments are in Appendix D.

1) MNIST Raw Digit Classification: We show that our DPM is a more effective mechanism than APM-G and APM-L when the dataset contains raw MNIST digits. Figure 11 shows the comparative results by the proposed FlowDRO DPM against the APM-G and APM-L baselines. Qualitatively, we notice in (a)-(c) that under the same amount of ℓ_2 perturbation ε, DPM induces meaningful contextual changes



(d) Type-I and type-II errors by DPM, APM-G, and APM-L.

Fig. 10. Differential privacy example of raw MNIST digit recognition. We control the ℓ_2 perturbation amount by DPM, APM-G, and APM-L to be identical for a fair comparison. Figures (a)-(c) visualize privacy-protected queries $M_\varepsilon(D)$ by DPM, APM-G, and APM-L over different ε. Figure (d) examines the corresponding type-I and type-II errors defined in (A.30) by these mechanisms.



(d) Type-I and type-II errors by DPM, APM-G, and APM-L.

Fig. 11. Differential privacy example of MNIST missing digit detection. We present similar sets of figures as in Figure 10, where the main difference lies in the definition of dataset D and query function $q(D)$, which returns an average image of images in D.

to the queries $q(D)$ (i.e., changing a digit 0 to a digit 8). In contrast, the additive mechanisms only blur the queries slightly. Quantitatively, as shown in (d), such difference helps protect privacy against the decision function ϕ: the type-I and type-II errors of ϕ under our proposed DPM are much higher than those of ϕ under the additive perturbation mechanisms. As a result, our DPM is an empirically more effective privacy-protecting mechanism under the same amount of average perturbation as measured in ε.

2) MNIST Missing Digit Detection: We consider an alternative setting that is a type of *membership inference attack* problem [45] and can be viewed as a more natural DP task. In short, we construct *average* images from digits of 9 classes, where the goal of the decision function ϕ, which is still a 10-class classifier, is to determine the class of the *missing* digit based on a given average image.

Figure 11 shows both qualitative and quantitative comparisons of our proposed DPM against APM-G and APM-L in this more challenging setting. The interpretations of results are similar to those in Section VI-C1. Specifically, we notice more contextual changes by DPM in subfigure (a) than APMs in subfigures (b) and (c), and the higher type-I and type-II errors in subfigure (d) demonstrate the benefit of DPM at protecting privacy against a pre-trained decision function ϕ.

VII. Summary and Discussion

In this paper, we have presented a computational framework called FlowDRO for solving the worst-case distribution, the Least Favorable Distributions (LFD), in Wasserstein Distributionally Robust Optimization (WDRO). Specifically, the worst-case distribution is found as the push-forward distribution induced by our FlowDRO model on the data distribution, and the entire probability trajectory is continuous and invertible due to the use of flow models. We demonstrate the utility of FlowDRO in various applications of DRO, including adversarial attacks of pre-trained image classifiers and differential privacy protection through our distributional perturbation mechanism. FlowDRO demonstrates strong improvement against baseline methods on high-dimensional data.

There are a few future directions to extend the work. Here, we set aside the min-max exchange issue for the following reasons. It has been shown in the original contribution [38] that when the reference measure (i.e., the center of the uncertainty set) is empirical distribution and thus discrete, the problem (2) has *strong duality*: one can exchange the min and max in the formulation and the solutions for the primal and the dual problems are the same when the loss function is convex-concave in the vector space. The results are shown leveraging the fact that the worst-case distributions for the Wasserstein DRO problem are discrete when the reference measure is discrete, thus reducing the infinite-dimensional optimization

problem to a finite-dimensional minimax problem. Thus, one can invoke the standard minimax theorem (see, e.g., [6]). Here, since later on we restrict the LFD to be a continuous function, the strong duality proof in [38] no longer carries through, and one has to extend the minimax theorem (e.g., [43] and [6] using Kakutani theorem) for the most general version involving functionals that are geodesic convex on the manifold of distribution functions; the proof is rather technical, and we leave it for future work. Second, theoretically, how to formalize our distributional perturbation mechanism on high-dimensional queries to make it satisfy a DP criterion is also an important question. Lastly, our approach is general and does not rely on neural networks. In future work, one can potentially extend to other alternative representations of the optimal transport maps that work particularly well for low-dimensional and small sample settings.

ACKNOWLEDGMENT

The authors would like to thank the helpful discussion with Dr. Daniel Kuhn, Dr. Jose Blanchet, Dr. Arkadi Nemirovski, Dr. Alexander Shapiro, and Dr. Georgia-Ann Klutke.

REFERENCES

[1] M. S. Alvim, M. E. Andrés, K. Chatzikokolakis, P. Degano, and C. Palamidessi, "Differential privacy: On the trade-off between utility and information leakage," in *Proc. 8th Int. Workshop Formal Asp. Secur. Trust, (FAST)*, 2012, pp. 39–54.

[2] L. Ambrosio, N. Gigli, and G. Savaré, *Gradient Flows: In Metric Spaces and in the Space of Probability Measures*. Berlin, Germany: Springer, 2005.

[3] T. Bai, J. Luo, J. Zhao, B. Wen, and Q. Wang, "Recent advances in adversarial training for adversarial robustness," in *Proc. 13th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2021, pp. 4312–4321.

[4] B. Balle, G. Barthe, M. Gaboardi, J. Hsu, and T. Sato, "Hypothesis testing interpretations and renyi differential privacy," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 2496–2506.

[5] J. Behrmann, W. Grathwohl, R. T. Q. Chen, D. Duvenaud, and J. Jacobsen, "Invertible residual networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 573–582.

[6] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. Philadelphia, PA, USA: SIAM, 2001.

[7] J. Blanchet and Y. Kang, "Semi-supervised learning based on distributionally robust optimization," in *Data Analysis and Applications 3: Computational, Classification, Financial, Statistical and Stochastic Methods*, Hoboken, NJ, USA: Wiley, 2020, pp. 1–33.

[8] J. Blanchet and K. Murthy, "Quantifying distributional model risk via optimal transport," *Math. Oper. Res.*, vol. 44, no. 2, pp. 565–600, 2019.

[9] J. Blanchet, F. Zhang, Y. Kang, and Z. Hu, "A distributionally robust boosting algorithm," in *Proc. Winter Simul. Conf. (WSC)*, 2019, pp. 3728–3739.

[10] M. Bowles and M. Agueh, "Weak solutions to a fractional Fokker–Planck equation via splitting and Wasserstein gradient flow," *Appl. Math. Lett.*, vol. 42, pp. 30–35, Ap. 2015.

[11] T. A. Bui, T. Le, Q. H. Tran, H. Zhao, and D. Phung, "A unified Wasserstein distributional robustness framework for adversarial training," in *Proc. Int. Conf. Learn. Represent.*, 2022, pp. 1–25.

[12] R. T. Q. Chen, J. Behrmann, D. K. Duvenaud, and J. H. Jacobsen, "Residual flows for invertible generative modeling," in *Proc. 33rd Adv. Neural Inf. Process. Syst.*, 2019, pp. 1–11.

[13] R. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, "Neural ordinary differential equations," in *Proc. 32nd Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–13.

[14] X. Cheng, J. Lu, Y. Tan, and Y. Xie, "Convergence of flow-based generative models via proximal gradient descent in Wasserstein space," 2023, *arXiv:2310.17582*.

[15] D. Desfontaines and B. Pejó, "SoK: Differential privacies," *Proc. Privacy Enhanc. Technol.*, vol. 2, no. 2, pp. 288–313, 2020.

[16] J. Dong, A. Roth, and W. J. Su, "Gaussian differential privacy," *J. Roy. Stat. Ser. B, (Stat. Methodol.)*, vol. 84, no. 1, pp. 3–37, 2022.

[17] C. Dwork, K. Kenthapadi, F. McSherry, I. Mironov, and M. Naor, "Our data, ourselves: Privacy via distributed noise generation," in *Proc. 24th Annu. Int. Conf. Theory Appl. Cryptogr. Techn.*, 2006, pp. 486–503.

[18] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. 3rd Theory Cryptogr. Conf.*, 2006, pp. 265–284.

[19] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends® Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2014.

[20] C. Dwork and G. N. Rothblum, "Concentrated differential privacy," 2016, *arXiv:1603.01887*.

[21] C. Finlay, J. Jacobsen, L. Nurbekyan, and A. Oberman, "How to train your neural ODE: The world of jacobian and kinetic regularization," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 3154–3164.

[22] R. Gao and A. Kleywegt, "Distributionally robust stochastic optimization with Wasserstein distance," *Math. Oper. Res.*, vol. 48, no. 2, pp. 603–655, 2023.

[23] R. Gao, L. Xie, Y. Xie, and H. Xu, "Robust hypothesis testing using Wasserstein uncertainty sets," in *Proc. 32nd Adv. Neural Inf. Process. Syst.*, 2018, pp. 1–11.

[24] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 1–9.

[25] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–11.

[26] W. Grathwohl, R. T. Q. Chen, J. Bettencourt, and D. Duvenaud, "Scalable reversible generative models with free-form continuous dynamics," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–13.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[28] P. J. Huber, "A robust version of the probability ratio test," *Ann. Math. Statist.*, vol. 36, no. 6, pp. 1753–1758, 1965.

[29] J. Hütter and P. Rigollet, "Minimax estimation of smooth optimal transport maps," *Ann. Statist.*, vol. 49, no. 2, pp. 1166–1194, 2021.

[30] P. Kairouz, S. Oh, and P. Viswanath, "The composition theorem for differential privacy," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 1376–1385.

[31] C. Kent, J. Li, J. Blanchet, and P. W. Glynn, "Modified frank Wolfe in probability space," in *Proc. 35th Adv. Neural Inf. Process. Syst.*, 2021, pp. 14448–14462.

[32] I. Kobyzev, S. Prince, and M. A. Brubaker, "Normalizing flows: An introduction and review of current methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 3964–3979, Nov. 2021.

[33] D. Kuhn, P. M. Esfahani, V. A. Nguyen, and S. Shafieezadeh-Abadeh, "Wasserstein distributionally robust optimization: Theory and applications in machine learning," in *Operations Research & Management Science in the Age of Analytics*. Catonsville, MD, USA: Informs, Oct. 2019, pp. 130–166.

[34] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[35] B. C. Levy, "Robust hypothesis testing with a relative entropy tolerance," *IEEE Trans. Inf. Theory*, vol. 55, no. 1, pp. 413–421, Jan. 2009.

[36] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–23.

[37] I. Mironov, "Rényi differential privacy," in *Proc. IEEE 30th Comput. Security Found. Symp. (CSF)*, 2017, pp. 263–275.

[38] P. M. Esfahani and D. Kuhn, "Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations," *Math. Program.*, vol. 171, nos. 1–2, pp. 115–166, 2018.

[39] J. Moreau, "Proximité et dualité dans un espace hilbertien," *Bull. De La Société MathÉMatique De France*, vol. 93, pp. 273–299, 1965.

[40] D. Onken, S. W. Fung, X. Li, and L. Ruthotto, "OT-flow: Fast and accurate continuous normalizing flows via optimal transport," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 9223–9232.

[41] S. Osher, H. Heaton, and S. Wu Fung, "A Hamilton–Jacobi-based proximal operator," *Proc. Nat. Acad. Sci.*, vol. 120, no. 14, 2023, Art. no. e2220469120.

[42] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends® Optim.*, vol. 1, no. 3, pp. 127–239, 2014.

[43] R. Rockafellar, *Convex Analysis*, vol. 11. Princeton, NJ, USA: Princeton Univ. Press, 1997.
[44] A. Shapiro, D. Dentcheva, and A. Ruszczynski, *Lectures on Stochastic Programming: Modeling and Theory*. Philadelphia, PA, USA: SIAM, 2021.
[45] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Proc. IEEE Symp. Security Privacy (SP)*, 2017, pp. 3–18.
[46] A. Sinha, H. Namkoong, and J. Duchi, "Certifiable distributional robustness with principled adversarial training," in *Proc. Int. Conf. Learn. Represent.*, 2018, pp. 1–49.
[47] E. Süli and D. F. Mayers, *An Introduction to Numerical Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[48] C. Szegedy et al., "Intriguing properties of neural networks," in *Proc. 2nd Int. Conf. Learn. Represent.*, 2014, pp. 1–10.
[49] B. Taskesen, V. A. Nguyen, D. Kuhn, and J. Blanchet, "A distributionally robust approach to fair classification," 2020, *arXiv:2007.09530*.
[50] J. Wang, R. Gao, and Y. Xie, "Sinkhorn distributionally robust optimization," 2023, *arXiv:2109.11926*.
[51] J. Wang and Y. Xie, "A data-driven approach to robust hypothesis testing using sinkhorn uncertainty sets," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2022, pp. 3315–3320.
[52] L. Wasserman and S. Zhou, "A statistical framework for differential privacy," *J. Amer. Stat. Assoc.*, vol. 105, no. 489, pp. 375–389, 2010.
[53] L. Xie, R. Gao, and Y. Xie, "Robust hypothesis testing with Wasserstein uncertainty sets," 2021, *arXiv:2105.14348*.
[54] C. Xu, X. Cheng, and Y. Xie, "Invertible neural networks for graph prediction," *IEEE J. Select. Areas Inf. Theory*, vol. 3, no. 3, pp. 454–467, Sep. 2022.
[55] C. Xu, X. Cheng, and Y. Xie, "Computing high-dimensional optimal transport by flow neural networks," 2024, *arXiv:2305.11857*.
[56] C. Xu, X. Cheng, and Y. Xie, "Normalizing flow neural networks by JKO scheme," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, pp. 1–27.
[57] L. Zhang, J. Yang, and R. Gao, "A simple duality proof for Wasserstein distributionally robust optimization," 2023, *arXiv:2205.00362*.
[58] S. Zhu, L. Xie, M. Zhang, R. Gao, and Y. Xie, "Distributionally robust weighted k-nearest neighbors," in *Proc. 36th Adv. Neural Inf. Process. Syst.*, 2022, pp. 29088–29100.