

Inverse Entropic Optimal Transport Solves Semi-supervised Learning via Data Likelihood Maximization

Mikhail Pershiianov^{1,2} Arip Asadulaev^{*3} Nikita Andreev^{*} Nikita Starodubcev⁴ Dmitry Baranchuk⁴
Anastasis Kratsios^{5,6} Evgeny Burnaev^{1,7} Alexander Korotin^{1,7}

Abstract

Learning conditional distributions $\pi^*(\cdot|x)$ is a central problem in machine learning, which is typically approached via supervised methods with paired data $(x, y) \sim \pi^*$. However, acquiring paired data samples is often challenging, especially in problems such as domain translation. This necessitates the development of *semi-supervised* models that utilize both limited paired data and additional unpaired i.i.d. samples $x \sim \pi_x^*$ and $y \sim \pi_y^*$ from the marginal distributions. The usage of such combined data is complex and often relies on heuristic approaches. To tackle this issue, we propose a new learning paradigm called **EBiEOT** that integrates both paired and unpaired data seamlessly using data likelihood maximization techniques. We demonstrate that our approach also connects intriguingly with inverse entropic optimal transport (OT). This finding allows us to apply recent advances in computational OT to establish an *end-to-end* learning algorithm to get $\pi^*(\cdot|x)$. In addition, we derive the universal approximation property, demonstrating that our approach can theoretically recover true conditional distributions with arbitrarily small error. Finally, we demonstrate through empirical tests that our method effectively learns conditional distributions using paired and unpaired data simultaneously. Source code: <https://github.com/MuXauJl111110/EBiEOT>.

¹Applied AI Institute, Moscow, Russia. ²Yandex School of Data Analysis, Moscow, Russia. ³MBZUAI, Abu Dhabi, UAE. ⁴Yandex Research, Moscow, Russia. ⁵Vector Institute, Toronto, Canada. ⁶McMaster University, Hamilton, Canada ⁷AXXX, Moscow, Russia.. Correspondence to: Mikhail Pershiianov <pershiianov.mi@gmail.com>.

Proceedings of the 43rd International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

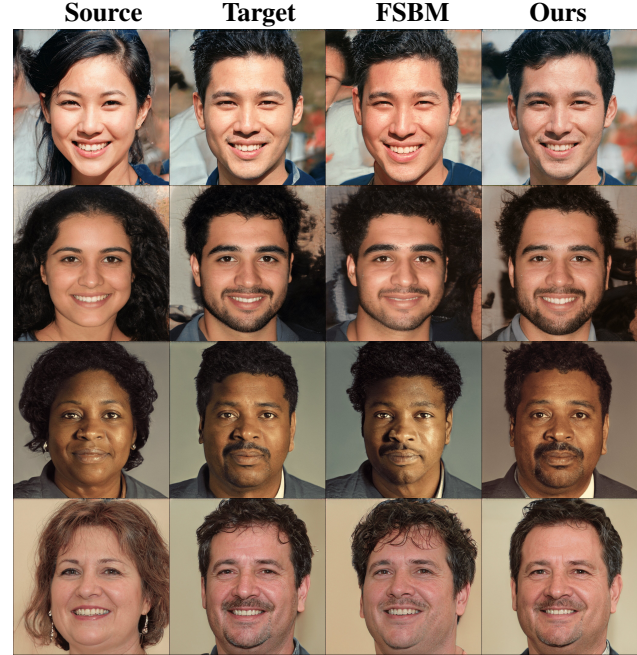


Figure 1. Semi-supervised Woman-to-Man Translation: A comparison between FSBM (Theodoropoulos et al., 2024) and our method. Both methods are applied within the ALAE latent space (Pidhorskyi et al., 2020) using 1024×1024 FFHQ images (Karras et al., 2019). See §5.3 for further experimental details.

1. Introduction

Recovering conditional distributions $\pi^*(y|x)$ from data is one of the fundamental problems in machine learning, which appears both in predictive and generative modeling. In predictive modeling, the standard examples of such tasks are the classification, where $x \in \mathbb{R}^{D_x}$ is a feature vector and $y \in \{y_1, \dots, y_K\}$ is a class label, and regression, in which case x is also a feature vector and $y \in \mathbb{R}$ is a real number. In generative modeling, both x and y are feature vectors in $\mathbb{R}^{D_x}, \mathbb{R}^{D_y}$, respectively, representing complex objects, and the goal is to find a transformation between them.

In this paper, we primarily focus on the continuous setting, where both x and y are multi-dimensional real-valued vectors, and the true joint distribution $\pi^*(x, y)$ is supported on $\mathbb{R}^{D_x} \times \mathbb{R}^{D_y}$. For completeness, we also discuss how the proposed framework can be adapted to classification

problems in Appendix B.2, and provide semi-supervised classification experiments on the MNIST dataset (Bottou et al., 1994) in Appendix C.2. The main focus of the paper is multi-dimensional probabilistic regression, which can also be viewed as a *domain translation* problem, since x and y typically represent feature vectors from different domains. The objective is to perform probabilistic prediction: given a new sample x_{new} from the source domain, we aim to model the corresponding conditional distribution $\pi^*(y|x_{\text{new}})$ over outputs in the target domain.

It is natural to assume that learning the conditional distribution $\pi^*(y|x)$ requires access to input–target data pairs $(x, y) \sim \pi^*$, where π^* denotes the true joint distribution of the data. In such cases, $\pi^*(y|x)$ can be modeled using standard supervised learning approaches, ranging from simple regression to conditional generative models (Mirza & Osindero, 2014; Winkler et al., 2019; Ardizzone et al., 2019; Hagemann et al., 2024). However, acquiring paired data may be costly, while getting unpaired samples $x \sim \pi_x^*$ or $y \sim \pi_y^*$ from two domains may be much easier and cheaper. This fact inspired the development of unsupervised (or unpaired) learning methods, e.g., (Zhu et al., 2017) among many others, which aim to somehow reconstruct the dependencies $\pi^*(y|x)$ with access to unpaired data only.

While both paired (supervised) and unpaired (unsupervised) domain translation approaches are being extremely well developed nowadays, surprisingly, the semi-supervised setup when both paired and unpaired data is available is much less explored. This is due to the challenge of designing learning objective (loss) which can simultaneously take into account both paired and unpaired data. A common approach involves heuristically combining standard paired and unpaired losses (cf. (Tripathy et al., 2019, §3.5), (Jin et al., 2019, §3.3), (Yang & Chen, 2020, §C), (Vasluianu et al., 2021, §3), (Panda et al., 2023, Eq. 8), (Tang et al., 2024, Eq. 8), (Theodoropoulos et al., 2024, §3.2), (Gu et al., 2023, §3)). However, as demonstrated in §5.1, these composite objectives fail to recover the true conditional distribution even in simple cases $D_x = D_y = 2$. This raises the question: *Can we design a simple loss to learn $\pi^*(y|x)$ that naturally integrates both paired and unpaired data?*

In our paper, we positively answer the above-raised question. Our **main contributions** are:

1. We introduce a novel loss function designed to facilitate the learning of conditional distributions $\pi^*(\cdot|x)$ using both paired and unpaired training samples drawn from π^* (§3.1). This loss function is grounded in the well-established principle of likelihood maximization. A key advantage of our approach is its ability to support end-to-end learning, thereby *seamlessly* integrating both paired and unpaired data into the training process.
2. We demonstrate the theoretical equivalence between our

proposed loss function and the *inverse entropic optimal transport* problem (§3.2). This finding enables us to leverage established computational OT methods to address challenges encountered in semi-supervised learning.

3. Building upon recent advances in computational optimal transport, we develop an *end-to-end* algorithm, called EBiEOT-GMM, based on a Gaussian mixture parameterization specifically designed to optimize the proposed likelihood-based objective (§3.3). For completeness, in Appendix A we additionally demonstrate that the same objective can be optimized using a fully neural-network-based parameterization, which we refer to as EBiEOT-NN.
4. We prove that our proposed parameterization satisfies the universal approximation property, namely, the proposed model class can approximate the target conditional distributions arbitrarily well under mild assumptions (§3.4).

Our empirical evaluation in §5 and Appendix C.2 demonstrates the impact of both unpaired and paired data on overall performance for domain translation and classification tasks. In particular, our findings show that the conditional distributions $\pi^*(\cdot|x)$ can be effectively learned even with a modest amount of paired data $(x, y) \sim \pi^*$, provided that sufficient auxiliary unpaired data $x \sim \pi_x^*$ and $y \sim \pi_y^*$ is available.

Notations. Throughout the paper, $\mathcal{X}$ and $\mathcal{Y}$ represent Euclidean spaces, equipped with the standard norm $\|\cdot\|$, induced by the inner product $\langle \cdot, \cdot \rangle$, i.e., $\mathcal{X} \stackrel{\text{def}}{=} \mathbb{R}^{D_x}$ and $\mathcal{Y} \stackrel{\text{def}}{=} \mathbb{R}^{D_y}$. The set of absolutely continuous probability distributions on $\mathcal{X}$ is denoted by $\mathcal{P}_{\text{ac}}(\mathcal{X})$. For simplicity, we use the same notation for both the distributions and their corresponding probability density functions. The joint probability distribution over $\mathcal{X} \times \mathcal{Y}$ is denoted by π with corresponding marginals π_x and π_y . The set of joint distributions with given marginals α and β is represented by $\Pi(\alpha, \beta)$. We use $\pi(\cdot|x)$ for the conditional distribution, while $\pi(y|x)$ represents the conditional density at a specific point y . The differential entropy is given by $H(\beta) = -\int_{\mathcal{Y}} \beta(y) \log \beta(y) dy$.

2. Background

First, we recall the formulation of the domain translation problem (§2.1). We remind the difference between its paired, unpaired, and semi-supervised setups. Next, we recall the basic concepts of the inverse entropic optimal transport, which are relevant to our paper (§2.2).

2.1. Domain Translation Problems

The goal of *domain translation* task is to transform data samples from the source domain to the target domain while maintaining the essential content or structure. This approach

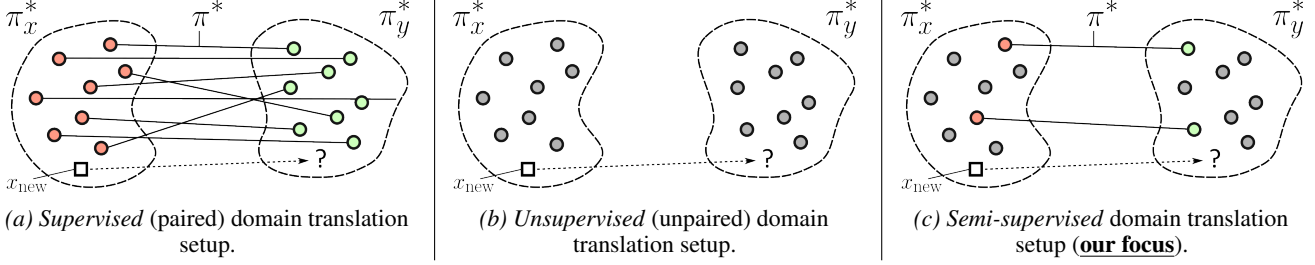


Figure 2. Visualization of domain translation setups. Red and green colors indicate paired training data XY_{paired} , while grey color indicates the unpaired training data $X_{\text{unpaired}}, Y_{\text{unpaired}}$.

is widely used in applications like computer vision (Zhu et al., 2017; Lin et al., 2018; Peng et al., 2023), natural language processing (Jiang et al., 2021; Morishita et al., 2022), audio processing (Du et al., 2022), etc. Domain translation task setups can be classified into supervised (paired), unsupervised (unpaired), and semi-supervised approaches based on the data used for training (Figure 2).

Supervised domain translation relies on matched examples from both the source and target domains, where each input corresponds to a specific output, enabling direct supervision during the learning process. Formally, this setup assumes access to a set of P empirical pairs $XY_{\text{paired}} \stackrel{\text{def}}{=} \{(x_1, y_1), \dots, (x_P, y_P)\} \sim \pi^*$ from some unknown joint distribution. The goal here is to recover the conditional distributions $\pi^*(\cdot|x)$ to generate samples $y|x_{\text{new}}$ for new inputs x_{new} that are not present in the training data. While this task is relatively straightforward to solve, obtaining such paired training datasets can be challenging, as it often involves significant time, cost, and effort.

Unsupervised domain translation, in contrast, does not require direct correspondences between the source and target domains (Zhu et al., 2017, Figure 2). Instead, it involves learning to translate between domains using unpaired data, which offers greater flexibility but demands more advanced techniques to achieve accurate translation. Formally, we are given Q unpaired samples $X_{\text{unpaired}} \stackrel{\text{def}}{=} \{x_1, \dots, x_Q\} \sim \pi_x^*$ from the source distribution, and R unpaired samples $Y_{\text{unpaired}} \stackrel{\text{def}}{=} \{y_1, \dots, y_R\} \sim \pi_y^*$ from the target distribution. Our objective is to learn the conditional distributions $\pi^*(\cdot|x)$ of the unknown joint distribution π^* , whose marginals are π_x^* and π_y^* , respectively. The unsupervised setup is inherently ill-posed, often yielding ambiguous solutions (Morikow et al., 2020). Accurate translation requires constraints and regularization (Yuan et al., 2018). Still, it is highly relevant due to the prevalence of unpaired data in practice.

Semi-supervised domain translation integrates both paired and unpaired data to enhance the translation process (Tripathy et al., 2019; Jiang et al., 2023a). This approach leverages the precision of paired data to guide the model while exploiting the abundance of unpaired data to improve performance and generalization. Formally, the setup as-

sumes access to paired data $XY_{\text{paired}} \sim \pi^*$ as well as additional unpaired samples $X_{\text{unpaired}} \sim \pi_x^*$ and $Y_{\text{unpaired}} \sim \pi_y^*$. Note that paired samples can also be used in an unpaired manner. By convention, we assume $P \leq Q, R$, where the first P unpaired samples are identical to the paired ones. The goal remains to learn the true conditional mapping $\pi^*(\cdot|x)$ using the available data. For extended discussion of real-world applications in which the semi-supervised setting arises naturally, see Appendix B.4.

2.2. Optimal Transport (OT)

We emphasize that this section is included primarily to clarify the connection between our proposed loss function (§3.1) and inverse entropic optimal transport (Dupuy et al., 2019), established in §3.2. Understanding this connection is not necessary for following the derivation of the loss itself, which is presented constructively in order to remain accessible to a broader audience.

For a more detailed discussion of entropic, weak, and inverse optimal transport, we refer the reader to Appendix B.1. Comprehensive introductions to the theoretical foundations of optimal transport can be found in (Villani et al., 2009; Santambrogio, 2015; Peyré & Cuturi, 2019).

Entropic OT (Genevay, 2019). Given source and target distributions $\alpha \in \mathcal{P}_{\text{ac}}(\mathcal{X})$ and $\beta \in \mathcal{P}_{\text{ac}}(\mathcal{Y})$, and a cost function $c^* : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, the *entropic optimal transport* (EOT) problem is defined as:

$$\text{EOT}_{c^*, \varepsilon}(\alpha, \beta) \stackrel{\text{def}}{=} \min_{\pi \in \Pi(\alpha, \beta)} \mathbb{E}_{x, y \sim \pi} [c^*(x, y)] - \varepsilon \mathbb{E}_{x \sim \alpha} \mathbb{H}(\pi(\cdot|x)), \quad (1)$$

where $\varepsilon > 0$ is the regularization parameter; setting $\varepsilon = 0$ recovers the classic OT formulation (Villani et al., 2009) originally proposed by (Kantorovich, 1942). Under mild assumptions, a unique minimizer $\pi^* \in \Pi(\alpha, \beta)$ exists and is known as the *entropic optimal transport plan*. We note that in the literature, the entropy regularization term in (1) is typically written as either $-\varepsilon \mathbb{H}(\pi)$ or $+\varepsilon \text{KL}(\pi \| \alpha \otimes \beta)$. These formulations are equivalent up to additive constants; see the discussion in (Mokrov et al., 2024, §2) or (Gushchin et al., 2023b, §1). In this paper, we adopt the *weak* formulation of

entropic OT (1) see (Gozlan et al., 2017; Backhoff-Veraguas et al., 2019; Backhoff-Veraguas & Pammer, 2022).

Semi-dual EOT. Under mild assumptions on c^* , α , β , the further EOT formulation of (1) in semi-dual form holds:

$$\text{EOT}_{c^*, \varepsilon}(\alpha, \beta) = \max_f \left\{ \mathbb{E}_{x \sim \alpha} f^{c^*}(x) + \mathbb{E}_{y \sim \beta} f(y) \right\}, \quad (2)$$

where f ranges over a subset of continuous functions (dual potentials) subject to mild boundedness conditions; see (Backhoff-Veraguas & Pammer, 2022, Eq. 3.3) for details. The term f^c denotes the so-called *weak entropic c -transform* of f , defined as:

$$f^{c^*}(x) \stackrel{\text{def}}{=} \min_{\mu \in \mathcal{P}(\mathcal{Y})} \left\{ \mathbb{E}_{y \sim \mu} [c^*(x, y)] - \varepsilon \text{H}(\mu) - \mathbb{E}_{y \sim \mu} f(y) \right\}. \quad (3)$$

It has closed-form (Mokrov et al., 2024, Eq. 14) given by:

$$f^c(x) = -\varepsilon \log \int_{\mathcal{Y}} \exp \left(\frac{f(y) - c(x, y)}{\varepsilon} \right) dy. \quad (4)$$

Inverse EOT. The classical forward EOT problem (1) seeks an optimal transport plan π^* between two given marginal distributions α and β under a fixed cost function c^* . In contrast, the *inverse* EOT problem considers the reverse setting (Chan et al., 2025, §5.1): given a joint distribution π^* with marginals π_x^* and π_y^* , the goal is to recover a cost function c^* such that π^* is the EOT plan for c^* .

This inverse formulation is not uniquely defined in the literature – each version is typically tailored to specific applications (Stuart & Wolfram, 2020; Ma et al., 2020; Galichon & Salanié, 2022; Andrade et al., 2023). In this work, we adopt a version that aligns with our learning objective described in §3.1. This choice enables us, in §3.2, to formally relate our proposed loss to the inverse EOT framework. We further conjecture that this connection could potentially enable the application of advanced EOT solvers (e.g., diffusion Schrödinger bridges (Vargas et al., 2021; De Bortoli et al., 2021; Gushchin et al., 2023a; Shi et al., 2024; Gushchin et al., 2024b)) to enhance performance in semi-supervised learning scenarios, which we leave for future work.

With this motivation, we consider the *inverse EOT problem* as the following minimization problem:

$$c^* \in \arg \min_c \left[\underbrace{\mathbb{E}_{x, y \sim \pi^*} [c(x, y)]}_{\geq \text{EOT}_{c, \varepsilon}(\pi_x^*, \pi_y^*)} - \overbrace{\varepsilon \mathbb{E}_{x \sim \pi_x^*} \text{H}(\pi^*(\cdot|x))}^{\text{not depend on } c} - \text{EOT}_{c, \varepsilon}(\pi_x^*, \pi_y^*) \right], \quad (5)$$

where $c : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ ranges over measurable cost functions. Consider the term $\text{EOT}_{c, \varepsilon}(\pi_x^*, \pi_y^*)$: due to entropic regularization and under mild assumptions, this expression

admits a unique optimal transport plan π_c^* for every quadruple $(c, \varepsilon, \pi_x^*, \pi_y^*)$. While π_c^* matches the marginals of π^* , its internal structure – i.e., the conditional distributions – may differ. The term $\mathbb{E}_{x, y \sim \pi^*} [c(x, y)] - \varepsilon \mathbb{E}_{x \sim \pi_x^*} \text{H}(\pi^*(\cdot|x))$ represents the *transportation cost* of using c to transport mass according to π^* (cf. the minimization objective in (1)). If the “inner” part of π_c^* differs from that of π^* , this cost exceeds $\text{EOT}_{c, \varepsilon}(\pi_x^*, \pi_y^*)$. Therefore, the minimum of the full objective is achieved only when π^* coincides with the optimal transport plan for some cost c^* , in which case the objective value is zero. Notably, the term $-\varepsilon \mathbb{E}_{x \sim \pi_x^*} \text{H}(\pi^*(\cdot|x))$ is independent of c and can be omitted from the optimization. Additionally:

- **Invariance to ε .** Unlike the forward EOT problem (1), the inverse formulation (5) is invariant to the choice of the regularization parameter $\varepsilon > 0$, up to a rescaling of the cost function. Indeed, let $\varepsilon' > 0$ and define $c'(x, y) = \frac{\varepsilon'}{\varepsilon} c(x, y)$. Then, $\frac{1}{\varepsilon'} c'(x, y) = \frac{1}{\varepsilon} c(x, y)$, so the factor $\exp\left(-\frac{c(x, y)}{\varepsilon}\right)$ remains unchanged. Consequently, the corresponding entropic OT plan is identical: $\pi_{c', \varepsilon'}^* = \pi_{c, \varepsilon}^*$. Moreover, substituting c' into (5) multiplies the entire objective by the constant factor ε'/ε , which does not affect the minimizers. Therefore, the inverse problem depends only on the ratio c/ε , and different choices of ε lead to equivalent solutions after rescaling the cost.
- **Multiple solutions.** The inverse problem (5) generally admits *many* valid cost functions. For instance, $c^*(x, y) = -\varepsilon \log \pi^*(x, y)$ achieves the minimum by construction. More generally, any function of the form $c'(x, y) = -\varepsilon \log \pi^*(x, y) + u(x) + v(y)$ is also valid, since additive terms depending only on x or y do not affect the resulting EOT plan. In particular, setting $u(x) = \varepsilon \log \pi_x^*(x)$ and $v(y) = 0$ yields $c^*(x, y) = -\varepsilon \log \pi^*(y|x)$.

In practice, π^* is known only through samples and not via its density. Therefore, closed-form expressions like $-\varepsilon \log \pi^*(x, y)$ or $-\varepsilon \log \pi^*(y|x)$ cannot be computed directly. This necessitates learning a parametric estimator π^θ to approximate the unknown conditional distributions.

3. Semi-supervised Domain Translation via Inverse EOT

In §3.1, we introduce a novel likelihood-based objective, called EBiEOT (Energy-Based inverse Entropic Optimal Transport), derived from the minimization of the KL divergence. In §3.2, we demonstrate the proposed loss is equivalent to solving the inverse EOT problem (5), thereby connecting optimal transport theory with our practical framework. To implement this approach, §3.3 introduces a tailored parameterization. We subsequently prove in §3.4 that this parameterization, when combined with our loss minimization, guarantees arbitrarily accurate reconstruction of

the true conditional plan under mild assumptions. For an extension to fully neural parameterizations, see Appendix A. Complete proofs for all results are located in Appendix E.

3.1. Loss Derivation

Part I. Data likelihood maximization and its limitation.

Our goal is to approximate the true distribution π^* by some parametric model π^θ , where θ represents the parameters of the model. To achieve this, we would like to employ the standard KL-divergence minimization framework, also known as data likelihood maximization. Namely, we aim to minimize:

$$\text{KL}(\pi^* \parallel \pi^\theta) = \mathbb{E}_{x,y \sim \pi^*} \log \frac{\pi_x^*(x) \pi^*(y|x)}{\pi_x^\theta(x) \pi^\theta(y|x)} = \quad (6)$$

$$\mathbb{E}_{x \sim \pi_x^*} \log \frac{\pi_x^*(x)}{\pi_x^\theta(x)} + \mathbb{E}_{x,y \sim \pi^*} \log \frac{\pi^*(y|x)}{\pi^\theta(y|x)} = \quad (7)$$

$$\text{KL}(\pi_x^* \parallel \pi_x^\theta) + \mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^*(\cdot|x)} \log \frac{\pi^*(y|x)}{\pi^\theta(y|x)} = \quad (8)$$

$$\underbrace{\text{KL}(\pi_x^* \parallel \pi_x^\theta)}_{\text{Marginal}} + \underbrace{\mathbb{E}_{x \sim \pi_x^*} \text{KL}(\pi^*(\cdot|x) \parallel \pi^\theta(\cdot|x))}_{\text{Conditional}}. \quad (9)$$

It is clear that objective (9) splits into two independent components: the *marginal* and the *conditional* matching terms. Our focus will be on the conditional component $\pi^\theta(\cdot|x)$, as it is the necessary part for the domain translation. Note that the marginal part π_x^θ is not actually needed. The conditional part of (9) can further be divided into the following terms:

$$\mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^*(\cdot|x)} [\log \pi^*(y|x) - \log \pi^\theta(y|x)] = -\mathbb{E}_{x \sim \pi_x^*} \text{H}(\pi^*(\cdot|x)) - \mathbb{E}_{x,y \sim \pi^*} \log \pi^\theta(y|x). \quad (10)$$

The first term is independent of θ , so we obtain the following minimization objective:

$$\mathcal{L}(\theta) \stackrel{\text{def}}{=} -\mathbb{E}_{x,y \sim \pi^*} \log \pi^\theta(y|x). \quad (11)$$

It is important to note that minimizing (11) is equivalent to maximizing the conditional likelihood, a strategy utilized in conditional normalizing flows (Papamakarios et al., 2021, CondNF). However, a major limitation of this approach is its reliance solely on paired data from π^* , which can be difficult to obtain in real-world scenarios. In the following section, we modify this strategy to incorporate available unpaired data within a semi-supervised learning setup (§2.1).

Part II. Solving the limitations via a tailored parameterization. To address the above-mentioned issue and utilize unpaired data, we first use Gibbs-Boltzmann distribution (LeCun et al., 2006) density parameterization:

$$\pi^\theta(y|x) \stackrel{\text{def}}{=} \frac{\exp(-E^\theta(y|x))}{Z^\theta(x)}, \quad (12)$$

where $E^\theta(\cdot|x) : \mathcal{Y} \rightarrow \mathbb{R}$ is the *Energy function*, and $Z^\theta(x) \stackrel{\text{def}}{=} \int_{\mathcal{Y}} \exp(-E^\theta(y|x)) dy$ is the normalization constant. Substituting (12) into (11), we obtain:

$$\mathcal{L}(\theta) = \mathbb{E}_{x,y \sim \pi^*} E^\theta(y|x) + \mathbb{E}_{x \sim \pi_x^*} \log Z^\theta(x). \quad (13)$$

This objective already provides an opportunity to exploit the unpaired samples from the marginal distribution π_x^* to learn the conditional distributions $\pi^\theta(\cdot|x) \approx \pi^*(\cdot|x)$. Namely, it helps to estimate the part of the objective related to the normalization constant Z^θ . To incorporate independent samples from the second marginal distribution π_y^* , it is crucial to adopt a parameterization that separates the term in the energy function $E^\theta(y|x)$ that depends only on y . Thus, we propose:

$$E^\theta(y|x) \stackrel{\text{def}}{=} \frac{c^\theta(x,y) - f^\theta(y)}{\varepsilon}. \quad (14)$$

In fact, this parameterization allows us to decouple the cost function $c^\theta(x,y)$ and the potential function $f^\theta(y)$. Specifically, changes in $f^\theta(y)$ can be offset by corresponding changes in $c^\theta(x,y)$, resulting in the same energy function $E^\theta(y|x)$. For example, by setting $f^\theta(y) \equiv 0$ and $\varepsilon = 1$, the parameterization of the energy function $E^\theta(y|x)$ remains consistent, as it can be exclusively derived from $c^\theta(x,y)$. Substituting (14) into the energy term of (13), and using the identity $\mathbb{E}_{x,y \sim \pi^*} f^\theta(y) = \mathbb{E}_{y \sim \pi_y^*} f^\theta(y)$, yields our *final objective*, which integrates both paired and unpaired data:

$$\begin{aligned} \mathcal{L}(\theta) = & \underbrace{\varepsilon^{-1} \mathbb{E}_{x,y \sim \pi^*} [c^\theta(x,y)]}_{\text{Joint, requires pairs } (x,y) \sim \pi^*} \\ & - \underbrace{\varepsilon^{-1} \mathbb{E}_{y \sim \pi_y^*} f^\theta(y)}_{\text{Marginal, requires } y \sim \pi_y^*} + \underbrace{\mathbb{E}_{x \sim \pi_x^*} \log Z^\theta(x)}_{\text{Marginal, requires } x \sim \pi_x^*} \end{aligned} \quad (15)$$

In Appendix E.1, we present a rigorous, step-by-step derivation starting from (6) and arriving at (15), using only *formal mathematical* transitions. Throughout this derivation, we assume that paired samples are drawn from the full joint distribution π^* . However, in practice the paired data may be restricted to a subset of π^* , discussed in Appendix B.3.

At this point, a reader may come up with 2 reasonable questions regarding (15):

1. How to perform the optimization of the proposed objective? This question is not straightforward due to the existence of the (typically intractable) normalizing constant Z_θ in the objective.
2. To which extent do the separate terms in (15) (paired, unpaired data) contribute to the objective, and which type of data is the most important to get the correct solution?

We answer these questions in §3.3 and §5. Before doing that, we show a surprising finding that our proposed objective actually solves the inverse entropic OT problem (5).

3.2. Relation to Inverse EOT

To show that (5) is equivalent to (15), we begin by substituting the semi-dual EOT formulation (2) into (5). After dropping the constant entropy term and utilizing the identity $\min(-g) = -\max g$, the expression simplifies to:

$$\min_{c,f} \left\{ \mathbb{E}_{x,y \sim \pi^*} [c(x,y)] - \mathbb{E}_{x \sim \pi_x^*} f^c(x) - \mathbb{E}_{y \sim \pi_y^*} f(y) \right\}. \quad (16)$$

Let c^θ and f^θ be parameterized by θ . Using (4) and the energy function in (14), we obtain $(f^\theta)^{c^\theta}(x) = -\varepsilon \log Z^\theta(x)$. This shows that the (5) formulation is equivalent to our proposed loss (15) up to scaling factor ε .

This result shows that *inverse entropic OT can be viewed as a likelihood maximization problem*, enabling the use of established techniques like ELBO and EM (Barber, 2012; Alemi et al., 2018; Bishop & Bishop, 2023). It also reframes inverse EOT as a semi-supervised domain translation task. Notably, prior work on inverse OT has largely focused on discrete, fully paired settings (see §4).

3.3. Practical Parameterization

The most computationally intensive aspect of optimizing the loss function in (15) lies in calculating the integral for the normalization constant Z^θ . To tackle this challenge, we propose a tailored parameterization, called EBiEOT-GMM, that yields closed-form expressions for each term in the loss function. Our proposed cost function parameterization c^θ is based on the log-sum-exp function (Murphy, 2012, §3.5.3):

$$c^\theta(x, y) = -\varepsilon \log \sum_{m=1}^M v_m^\theta(x) \exp \left(\frac{\langle a_m^\theta(x), y \rangle}{\varepsilon} \right), \quad (17)$$

where $\{v_m^\theta(x) : \mathbb{R}^{D_x} \rightarrow \mathbb{R}_+, a_m^\theta(x) : \mathbb{R}^{D_x} \rightarrow \mathbb{R}^{D_y}\}_{m=1}^M$ are arbitrary parametric functions, e.g., *neural networks*, with learnable parameters denoted by θ_c . The parametric form of the cost is motivated by (Korotin et al., 2024), from which we derived a more general functional form appropriate for our setting. Therefore, we adopt a Gaussian mixture parameterization for the dual potential f^θ :

$$f^\theta(y) = \varepsilon \log \sum_{n=1}^N w_n^\theta \mathcal{N}(y | b_n^\theta, \varepsilon B_n^\theta), \quad (18)$$

where $\theta_f \stackrel{\text{def}}{=} \{w_n^\theta, b_n^\theta, B_n^\theta\}_{n=1}^N$ are learnable parameters of the potential, with $w_n^\theta \geq 0$, $b_n^\theta \in \mathbb{R}^{D_y}$, and $B_n^\theta \in \mathbb{R}^{D_y \times D_y}$ being a symmetric positive definite matrix. Thereby, our framework comprises a total of $\theta \stackrel{\text{def}}{=} \theta_f \cup \theta_c$ learnable parameters. For clarity and to avoid notation overload, we will omit the superscript $^\theta$ associated with learnable parameters and functions in the subsequent formulas.

Proposition 3.1 (Tractable normalization constant). *Our parameterization of the cost function (17) and dual potential*

(18) *delivers* $Z^\theta(x) \stackrel{\text{def}}{=} \sum_{m=1}^M \sum_{n=1}^N z_{mn}(x)$, *where*

$$z_{mn}(x) \stackrel{\text{def}}{=} w_n v_m(x) \exp \left(\frac{a_m^\top(x) B_n a_m(x) + 2b_n^\top a_m(x)}{2\varepsilon} \right).$$

The proposition offers a closed-form expression for $Z^\theta(x)$, which is essential for optimizing (15). Furthermore, our following proposition supplements the previous one and provides a method for sampling y given a new sample x_{new} .

Proposition 3.2 (Tractable conditional distributions). *From our parameterization of the cost function (17) and dual potential (18) it follows that the $\pi^\theta(\cdot|x)$ are Gaussian mixtures:*

$$\pi^\theta(y|x) = \frac{1}{Z^\theta(x)} \sum_{m=1}^M \sum_{n=1}^N z_{mn}(x) \mathcal{N}(y | d_{mn}(x), \varepsilon B_n), \quad (19)$$

where $d_{mn}(x) \stackrel{\text{def}}{=} b_n + B_n a_m(x)$ and $z_{mn}(x)$ defined in Proposition 3.1.

TRAINING. As stated in §2.1, since we only have access to samples from the distributions, we minimize the empirical counterpart of (15) via the gradient descent w.r.t. θ :

$$\begin{aligned} \mathcal{L}(\theta) \approx \hat{\mathcal{L}}(\theta) \stackrel{\text{def}}{=} & -\varepsilon^{-1} \frac{1}{P} \sum_{p=1}^P c^\theta(x_p, y_p) \\ & -\varepsilon^{-1} \frac{1}{R} \sum_{r=1}^R f^\theta(y_r) + \frac{1}{Q} \sum_{q=1}^Q \log Z^\theta(x_q). \end{aligned} \quad (20)$$

INFERENCE. According to our Proposition 3.2, the conditional distributions $\pi^\theta(\cdot|x)$ are Gaussian mixtures (19). As a result, sampling y given x is fast and straightforward.

3.4. Universal Approximation of the Proposed Parameterization

One may naturally wonder how expressive is our proposed parameterization of π_θ in §3.3. Below we show that this parameterization allows approximating any distribution π^* that satisfies mild assumptions on boundness and regularity assumptions, see the details in Appendix E.4.

Theorem 3.3 (Proposed parameterization guarantees universal conditional distributions). *Under mild assumptions on the joint distribution π^* , for all $\delta > 0$ there exists (a) an integer $N > 0$ and a Gaussian mixture f^θ (18) with N components, (b) an integer $M > 0$ and cost c^θ (17) defined by fully-connected neural networks $a_m : \mathbb{R}^{D_x} \rightarrow \mathbb{R}^{D_y}$, $v_m : \mathbb{R}^{D_x} \rightarrow \mathbb{R}_+$ with ReLU activations such that π^θ defined by (12) and (14) satisfies $\text{KL}(\pi^* \parallel \pi^\theta) < \delta$.*

4. Related Works

In this section, we briefly summarize the most relevant prior work; a more detailed discussion appears in Appendix B.5.

Existing semi-supervised domain translation approaches typically combine ad hoc objectives based on GAN losses and paired-data regularization (Chen et al., 2023; Panda et al., 2023), or use *keypoint-guided OT* (Gu et al., 2022), later extended to diffusion-based models (Gu et al., 2023; Theodoropoulos et al., 2024). Importantly, the paradigms outlined above do not offer any theoretical guarantees for reconstructing the conditional distribution $\pi^*(y|x)$, as they depend on heuristic loss constructions. We show that such approaches actually fail to recover the true plan even in toy 2-dimensional cases, refer to experiments in §5 for an illustrative example. **Inverse OT solvers:** works (Dupuy et al., 2019; Stuart & Wolfram, 2020) focuses on reconstructing cost functions (often in discrete settings), whereas our aim is to learn conditional distribution $\pi^\theta(\cdot|x)$. **Forward OT solvers:** Building on (Mokrov et al., 2024) and Gaussian-mixture parameterizations (Korotin et al., 2024; Gushchin et al., 2024a), our solver extends forward OT methods to general cost functions (Eq. (17)) and incorporates paired data through likelihood-based cost learning. Full details and additional discussion of **metric-learning** (Cuturi & Avis, 2014) provided in the Appendix B.5.

5. Experimental Illustrations

We evaluate the proposed solver on several tasks, including synthetic datasets (§5.1 and Appendix C.1), real-world data (§5.2), image translation (§5.3), and semi-supervised classification (Appendix C.2). Our PyTorch implementation is publicly available at the following repository¹. Additional experimental details are provided in Appendix D.

5.1. Gaussian to Swiss Roll Mapping

Setup. For illustration, we adapt the experimental setup from (Korotin et al., 2024) to our purposes. We consider the task of learning conditional distributions from a Gaussian distribution π_x^* to a Swiss Roll distribution π_y^* (Figure 3a), guided by paired samples (Figure 3b) drawn from the ground-truth plan π^* . The ground-truth plan π^* is obtained from a mini-batch OT plan after solving the *forward* OT problem with a specially designed cost that induces bi-modal conditionals $\pi^*(\cdot|x)$. Specifically, the cost matrix is defined as $C = \min(C^{+\varphi}, C^{-\varphi})$, where $C^{\pm\varphi}$ contains pairwise ℓ_2 distances between x and $-y^{\pm\varphi}$, with $-y^{\pm\varphi}$ denoting the vector $-y$ rotated by an angle of $\varphi = \pm 90^\circ$. In other words, each $x \sim \pi_x^*$ is mapped to a point y on the opposite side of the Swiss Roll, rotated either by $+\varphi$ or $-\varphi$ (Figure 3c). Additional details on the generation of paired data are provided in Appendix D.4. We also consider an alternative analytically defined ground-truth plan of the form $y = T(x) + \xi$, where ξ is a noise term, described in Appendix C.1.3. We evaluate each method by its ability to

recover these multi-modal conditional distributions. Unless stated otherwise, training uses $P = 128$ paired samples together with $Q = R = 1024$ unpaired samples. In Appendix C.1.2, we study the effect of varying the amounts of paired and unpaired data, while Appendix C.1.4 analyzes the sensitivity of the method to the choice of the parameters N and M .

Baselines. We evaluate our method against several baselines (see Appendix D.1 for details):

1. **Semi-supervised log-likelihood** methods (Atanov et al., 2019; Izmailov et al., 2020): CNF (SS) and CGMM (SS).
2. **Semi-supervised methods:** Neural OT with pair-guided cost (Asadulaev et al., 2024, GNOT, Appendix E), Differentiable Cost-Parameterized Entropic Mapping Estimator (Howard et al., 2024, DCPME), (Panda et al., 2023, parOT), (Gu et al., 2023, OTCS), Feedback Schrödinger Bridge Matching (Theodoropoulos et al., 2024, FSBM).
3. **Standard generative & predictive models:** MLP regression with ℓ^2 loss, Unconditional GAN with ℓ^2 loss supplement (Goodfellow et al., 2014, UGAN+ ℓ^2), Conditional GAN (Mirza & Osindero, 2014, CGAN), Conditional Normalizing Flow (Winkler et al., 2019, CondNF).

Note that some baselines can fully utilize both paired and unpaired data during training, while others rely solely on paired data. Refer to Table 7 for specifics on data usage.

Metrics. We evaluate the generated target distributions π_y^* and conditional distributions $\pi^*(\cdot|x)$ using the Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) and Sinkhorn divergence (Feydy et al., 2019) metrics approximating $\mathcal{W}_2$. Definitions of these metrics together with implementation details are provided in Appendix D.2. The corresponding results are reported in Table 5 for the setting considered in this section ($P = 128$, $Q = R = 1024$), as well as for the almost infinite amount of samples setting with $P = Q = R = 16k$ discussed in Appendix C.1.1.

Discussion. The results of the aforementioned methods are depicted in Figure 3. Clearly, the Regression model simply predicts the conditional mean $\mathbb{E}_{y \sim \pi^*(\cdot|x)} y$, failing to capture the full distribution. The CGAN is unable to accurately learn the target distribution π_y^* , while the UGAN+ ℓ^2 fails to capture the underlying conditional distribution, resulting in suboptimal performance. The CondNF model suffers from overfitting, likely due to the limited availability of paired data XY_{paired} . Methods GNOT, DCPME, parOT learn deterministic mapping and therefore are unable to capture the conditional distribution. Similar to parOT, both OTCS and FSBM build on the idea of key-points but are designed for stochastic setup. However, these methods fail to capture bi-modal conditional mappings, presumably due to a biased objective introduced by the artificial cost function that enforces alignment with key-points. The CondNF (SS)

¹ <https://github.com/MuXauJI11110/EBiEOT>

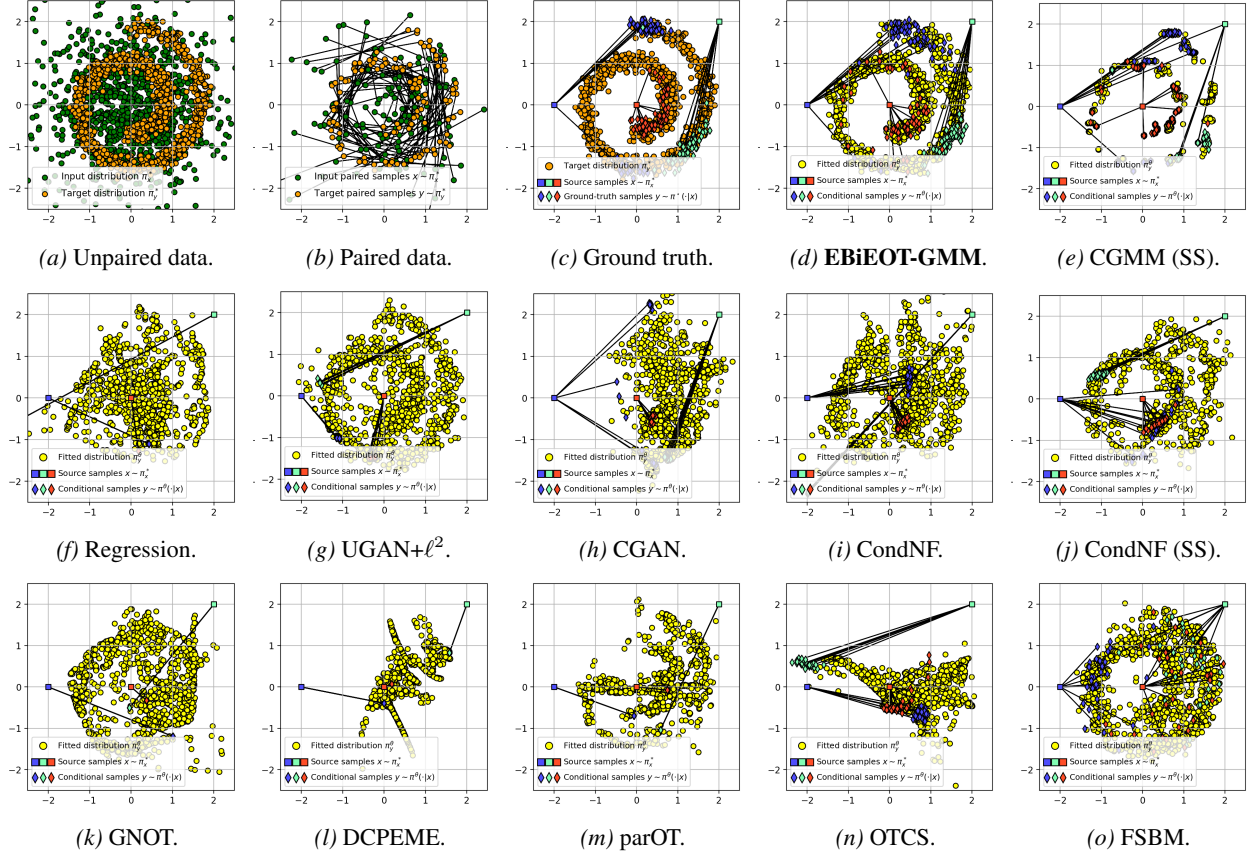


Figure 3. Learned mapping on the *Gaussian* $\rightarrow$ *Swiss Roll* task for $P = 128$ and $Q = R = 1024$.

does not provide improvement compared to CondNF, and CGMM (SS) model learns a degenerate solution, which is presumably due to the overfitting. As a sanity check, we evaluate all baselines using a large amount of paired data. Details are given in Appendix C.1.1. In fact, even in this case, almost all the methods fail to learn true $\pi^*(\cdot|x)$.

5.2. Weather Prediction

Here we aim to evaluate our proposed approach on real-world data. We consider the *weather prediction* dataset (Mallin et al., 2021; Rubachev et al., 2025). The data is collected from weather stations across the world and weather forecast physical models. It consists of 94 meteorological features, e.g., pressure, wind, humidity, etc., which are measured over a period of one year at different spatial locations.

Setup. Initially, the problem was formulated as the prediction and uncertainty estimation of the air temperature at a specific time and location. We expand this task to the probabilistic prediction of all meteorological features, thereby reducing reliance on measurement equipment in remote and difficult-to-access locations, e.g. the Polar regions.

Metrics and baselines. We evaluate the performance of our approach by calculating the *log-likelihood* (LL) on the

test target features. A natural baseline for this task is a probabilistic model that maximizes the likelihood of the target data. Thus, we implement an MLP that learns to predict the parameters of a mixture of Gaussians and is trained on the paired data only via the LL optimization (11). We also compare with semi-supervised LL methods CGMM (SS) and CondNF (SS). For completeness, we also add standard generative models. These models are trained using the available paired and unpaired data. Note that GAN models do not provide the density estimation and log-likelihood can not be computed for them. Therefore, we report Conditional Fréchet Distance (CFD): for each test x , we compute the FID (Heusel et al., 2017, Eq. 6) between predicted and true features y , then average over x .

Discussion. Tables 1 and 2 summarize our findings. From Table 1, the main observation is that even a small amount of unpaired data leads to substantial performance gains, underscoring the effectiveness of our semi-supervised formulation. Furthermore, Table 2 shows that our method obtains the best LL and is statistically competitive on CFD; CGMM (SS) attains a slightly lower CFD within overlapping uncertainty. For more detailed discussion regarding low-data regimes, see Appendix D.6.

# Paired \ # Unpaired	Baseline	Ours					
	0	5	10	50	100	250	500
5	diverged	9.4 ±.1	14.2 ±1.7	15.47 ±.02	16.6 ±.0	17.91 ±.07	9.40 ±.03
10	0.4 ±.2	9.48 ±.02	17.9 ±.3	18.5 ±.4	18.4 ±.2	18.8 ±.2	19.2 ±.3
25	3.5 ±.09	9.40 ±.03	18.3 ±.06	18.7 ±.2	18.8 ±.07	19.5 ±.1	19.8 ±.1
50	6.4 ±.05	9.47 ±.01	18.7 ±.2	18.9 ±.04	19.2 ±.2	19.8 ±.03	20.3 ±.4
90	6.5 ±.1	9.30 ±.05	19 ±.01	19.4 ±.05	19.4 ±.2	20.3 ±.05	20.5 ±.09

Table 1. The values of the test *log-likelihood* $\uparrow$ on the *weather prediction* dataset obtained for a different number of paired and unpaired training samples.

	Ours	CGAN	UGAN+ ℓ^2	CondNF	Regression	CGMM (SS)	CondNF (SS)
LL $\uparrow$	20.5 ±.09	N/A	N/A	1.29 ±.03	N/A	0.32 ±.03	0.52 ±.02
CFD $\downarrow$	7.21 ±.04	15.79 ±1.11	15.44 ±1.89	18.72 ±.09	8.29 ±.04	7.17 ±.07	28.5 ±.5

Table 2. The values of the test *Log-Likelihood* (LL) and *Conditional Fréchet distance* (CFD) on the *weather prediction* dataset of our approach and baselines (500 unpaired and 90 paired samples).

5.3. Image Translation via ALAE

Setup. In this section, following the setup from (Theodoropoulos et al., 2024), we demonstrate our method capabilities for image translation in the 512-dimensional latent space of the ALAE encoder (Pidhorskyi et al., 2020), pre-trained on the 1024×1024 FFHQ dataset (Karras et al., 2019). We consider two translation tasks: (i) Woman-to-Man translation and (ii) Old-to-Young translation. Similarly, we generate 2K paired samples using (Korotin et al., 2024).

Baselines. We used the publicly available FSBM (Theodoropoulos et al., 2024) implementation from GitHub². However, due to reproducibility issues in the repository, we generated 2K paired samples ourselves via the procedure described in Appendix C.3 of the original paper.

Metrics. Metrics were computed using `TorchMetrics` (Falcon et al., 2020) with a batch size of 128 and averaged over three trainings with different seeds (LPIPS (Zhang et al., 2018), FID (Heusel et al., 2017), SSIM (Wang et al., 2004)). All metrics measure similarity between the generated and target distributions and are averaged across three independent runs with different seeds. Results are reported rounded to the first significant digit.

Discussion. Qualitative results are presented in Figure 1. Quantitative results, averaged across three independent runs, are reported in Table 3 and Table 4. Additional visual examples are provided in Figure 10 and Figure 11. Notably, our method achieves comparable performance to FSBM while being significantly more efficient: it requires only 3 minutes of training on an A100 GPU, whereas FSBM requires 5 hours on the same hardware. Additional implementation and training details are provided in Appendix D.5.

²<https://github.com/panostheo98/FSBM>

Method	FID $\downarrow$	SSIM $\uparrow$	LPIPS $\downarrow$
FSBM	10.2 ± 0.6	0.5237 ± 0.0005	0.5625 ± 0.0003
Ours	9.3 ± 0.1	0.5315 ± 0.0002	0.5531 ± 0.0006

Table 3. Metrics for Woman-to-Man translation.

Method	FID $\downarrow$	SSIM $\uparrow$	LPIPS $\downarrow$
FSBM	11.5 ± 0.6	0.5285 ± 0.0008	0.5628 ± 0.0004
Ours	9.4 ± 0.2	0.5361 ± 0.0004	0.5560 ± 0.0005

Table 4. Metrics for Old-to-Young translation.

6. Discussion

Contributions & Potential impact. Our framework offers a simple, non-minimax objective that naturally integrates both paired and unpaired data. We expect that these advantages, together with the connection to entropic optimal transport, will encourage adoption in more advanced semi-supervised methods, including approaches based on diffusion Schrödinger bridges (Vargas et al., 2021; De Bortoli et al., 2021; Shi et al., 2024) and flow matching (Chen et al., 2025; Balcerak et al., 2025).

The primary focus of this paper is semi-supervised learning for domain translation with continuous targets $y \in \mathbb{R}^{D_y}$. For completeness, Appendix B.2 discusses an extension of the proposed loss to discrete targets $y \in \mathcal{Y}_K^{D_y}$, where $\mathcal{Y}_K = \{y_1, \dots, y_K\}$ denotes a finite set of categories. We view this discrete setting as an important direction for future research. In addition, Appendix C.2 demonstrates the applicability of our method to semi-supervised classification tasks.

Limitations & Future Work. A limitation of our method is its reliance on Gaussian Mixture parameterization (§3.3), which may affect scalability. To address this, we provide a proof of concept for fully neural parameterizations of the cost and potential functions below, with a detailed discussion in Appendix A. These parameterizations can be integrated into our loss via energy-based modeling (Song & Kingma, 2021) and could, in principle, scale to large image domains (Schröder et al., 2023; Yu et al., 2023; Zhu et al., 2024). A full investigation of such large-scale applications, however, lies beyond the scope of our methodological work.

Acknowledgments

The work was supported by the grant for research centers in the field of AI provided by the Ministry of Economic Development of the Russian Federation in accordance with the agreement 000000C313925P4F0002 and the agreement №139-10-2025-033.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

- Acciaio, B., Kratsios, A., and Pammer, G. Designing universal causal deep learning models: The geometric (hyper) transformer. *Mathematical Finance*, 34(2):671–735, 2024.
- Agarap, A. F. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.
- Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.
- Alemi, A., Poole, B., Fischer, I., Dillon, J., Saourous, R. A., and Murphy, K. Fixing a broken elbo. In *International conference on machine learning*, pp. 159–168. PMLR, 2018.
- Altschuler, J., Niles-Weed, J., and Rigollet, P. Near-linear time approximation algorithms for optimal transport via sinkhorn iteration. *Advances in neural information processing systems*, 30, 2017.
- Andrade, F., Peyré, G., and Poon, C. Sparsistency for inverse optimal transport. *arXiv preprint arXiv:2310.05461*, 2023.
- Andrade, F., Peyré, G., and Poon, C. Learning from samples: Inverse problems over measures via sharpened fenchel-young losses. *arXiv preprint arXiv:2505.07124*, 2025.
- Andrieu, C., De Freitas, N., Doucet, A., and Jordan, M. I. An introduction to mcmc for machine learning. *Machine learning*, 50:5–43, 2003.
- Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., Bao, B., Bell, P., Berard, D., Burovski, E., Chauhan, G., Chourdia, A., Constable, W., Desmaison, A., DeVito, Z., Ellison, E., Feng, W., Gong, J., Gschwind, M., Hirsh, B., Huang, S., Kalambarkar, K., Kirsch, L., Lazos, M., Lezcano, M., Liang, Y., Liang, J., Lu, Y., Luk, C., Maher, B., Pan, Y., Puhersch, C., Reso, M., Saroufim, M., Siraichi, M. Y., Suk, H., Suo, M., Tillet, P., Wang, E., Wang, X., Wen, W., Zhang, S., Zhao, X., Zhou, K., Zou, R., Mathews, A., Chanan, G., Wu, P., and Chintala, S. Pytorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, April 2024. doi: 10.1145/3620665.3640366. URL <https://pytorch.org/assets/pytorch2-2.pdf>.
- Arabpour, R., Armstrong, J., Galimberti, L., Kratsios, A., and Livieri, G. Low-dimensional approximations of the conditional law of volterra processes: a non-positive curvature approach. *arXiv preprint arXiv:2405.20094*, 2024.
- Ardizzone, L., Lüth, C., Kruse, J., Rother, C., and Köthe, U. Guided image generation with conditional invertible neural networks. *arXiv preprint arXiv:1907.02392*, 2019.
- Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.
- Asadulaev, A., Korotin, A., Egiastian, V., Mokrov, P., and Burnaev, E. Neural optimal transport with general cost functionals. In *The Twelfth International Conference on Learning Representations*, 2024.
- Atanov, A., Volokhova, A., Ashukha, A., Sosnovik, I., and Vetrov, D. Semi-conditional normalizing flows for semi-supervised learning. *arXiv preprint arXiv:1905.00505*, 2019.
- Auffenberg, J., Bresch, J., Melnyk, O., and Steidl, G. Unsupervised ground metric learning. *arXiv preprint arXiv:2507.13094*, 2025.
- Austin, J., Johnson, D. D., Ho, J., Tarlow, D., and Van Den Berg, R. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.
- Backhoff-Veraguas, J. and Pammer, G. Applications of weak transport theory. *Bernoulli*, 28(1):370–394, 2022.
- Backhoff-Veraguas, J., Beiglböck, M., and Pammer, G. Existence, duality, and cyclical monotonicity for weak transport costs. *Calculus of Variations and Partial Differential Equations*, 58(6):203, 2019.
- Balcerak, M., Amiranashvili, T., Shit, S., Terpin, A., Kaltenbach, S., Koumoutsakos, P., and Menze, B. Energy matching: Unifying flow matching and energy-based models for generative modeling. *arXiv preprint arXiv:2504.10612*, 2025.
- Barber, D. *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.
- Bishop, C. M. and Bishop, H. *Deep learning: Foundations and concepts*. Springer Nature, 2023.
- Bottou, L., Cortes, C., Denker, J. S., Drucker, H., Guyon, I., Jackel, L. D., LeCun, Y., Muller, U. A., Sackinger, E., Simard, P., et al. Comparison of classifier methods: a case study in handwritten digit recognition. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition, Vol. 3-Conference C: Signal Processing (Cat. No. 94CH3440-5)*, volume 2, pp. 77–82. IEEE, 1994.

- Campbell, A., Benton, J., De Bortoli, V., Rainforth, T., Deligiannidis, G., and Doucet, A. A continuous time framework for discrete denoising models. Advances in Neural Information Processing Systems, 35:28266–28279, 2022.
- Carbone, D. Hitchhiker’s guide on energy-based models: a comprehensive review on the relation with other generative models, sampling and statistical physics. arXiv preprint arXiv:2406.13661, 2024.
- Carbone, D., Hua, M., Coste, S., and Vanden-Eijnden, E. Efficient training of energy-based models using jarzynski equality. Advances in Neural Information Processing Systems, 36:52583–52614, 2023.
- Chan, T. C., Mahmood, R., and Zhu, I. Y. Inverse optimization: Theory and applications. Operations Research, 73(2):1046–1074, 2025.
- Chen, C., Liu, W., Tan, X., and Wong, K.-Y. K. Semi-supervised cycle-gan for face photo-sketch translation in the wild. Computer Vision and Image Understanding, 235:103775, 2023.
- Chen, H., Zhang, K., Tan, H., Xu, Z., Luan, F., Guibas, L., Wetzstein, G., and Bi, S. Gaussian mixture flow matching models. arXiv preprint arXiv:2504.05304, 2025.
- Chizat, L., Peyré, G., Schmitzer, B., and Vialard, F.-X. Scaling algorithms for unbalanced optimal transport problems. Mathematics of computation, 87(314):2563–2609, 2018.
- Cui, T., Dai, Q., Zhang, M., Li, K., and Ji, X. Scl-dehaze: Toward real-world image dehazing via semi-supervised codebook learning. Electronics, 13(19):3826, 2024.
- Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. Advances in neural information processing systems, 26, 2013.
- Cuturi, M. and Avis, D. Ground metric learning. The Journal of Machine Learning Research, 15(1):533–564, 2014.
- Cuturi, M., Klein, M., and Abblin, P. Monge, bregman and oc-cam: Interpretable optimal transport in high-dimensions with feature-sparse maps. In International Conference on Machine Learning, pp. 6671–6682. PMLR, 2023.
- De Bortoli, V., Thornton, J., Heng, J., and Doucet, A. Diffusion schrödinger bridge with applications to score-based generative modeling. Advances in Neural Information Processing Systems, 34:17695–17709, 2021.
- Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real NVP. In International Conference on Learning Representations, 2017. URL <https://openreview.net/forum?id=HkpbhH9lx>.
- Du, Y. and Mordatch, I. Implicit generation and modeling with energy based models. Advances in Neural Information Processing Systems, 32, 2019.
- Du, Y., Li, S., Tenenbaum, B. J., and Mordatch, I. Improved contrastive divergence training of energy based models. In Proceedings of the 38th International Conference on Machine Learning (ICML-21), 2021.
- Du, Y., Wang, W., Zhang, Z., Chen, B., Xu, T., Xie, J., and Chen, E. Non-parametric domain adaptation for end-to-end speech translation. In Conference on Empirical Methods in Natural Language Processing (EMNLP), 2022.
- Dupuy, A., Galichon, A., and Sun, Y. Estimating matching affinity matrices under low-rank constraints. Information and Inference: A Journal of the IMA, 8(4):677–689, 2019.
- Dvurechensky, P., Gasnikov, A., and Kroshnin, A. Computational optimal transport: Complexity by accelerated gradient descent is better than by sinkhorn’s algorithm. In International conference on machine learning, pp. 1367–1376. PMLR, 2018.
- Falcon, W., Skafté, N., Schock, J., et al. Torchmetrics: Machine learning metrics for pytorch, 2020. URL <https://github.com/Lightning-AI/metrics>. Version: latest.
- Feydy, J., Séjourné, T., Vialard, F.-X., Amari, S.-i., Trounev, A., and Peyré, G. Interpolating between optimal transport and mmd using sinkhorn divergences. In The 22nd International Conference on Artificial Intelligence and Statistics, pp. 2681–2690, 2019.
- Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boissunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., et al. Pot: Python optimal transport. Journal of Machine Learning Research, 22(78):1–8, 2021.
- Galichon, A. and Salanié, B. Cupid’s invisible hand: Social surplus and identification in matching models. The Review of Economic Studies, 89(5):2600–2629, 2022.
- Gao, R., Song, Y., Poole, B., Wu, Y. N., and Kingma, D. P. Learning energy-based models by diffusion recovery likelihood. In International Conference on Learning Representations, 2021.
- Gat, I., Remez, T., Shaul, N., Kreuk, F., Chen, R. T., Synnaeve, G., Adi, Y., and Lipman, Y. Discrete flow matching. Advances in Neural Information Processing Systems, 37:133345–133385, 2024.

- Genevay, A. Entropy-regularized optimal transport for machine learning. PhD thesis, Université Paris sciences et lettres, 2019.
- Genevay, A., Chizat, L., Bach, F., Cuturi, M., and Peyré, G. Sample complexity of sinkhorn divergences. In The 22nd international conference on artificial intelligence and statistics, pp. 1574–1583. PMLR, 2019.
- Geng, C., Han, T., Jiang, P.-T., Zhang, H., Chen, J., Hauberg, S., and Li, B. Improving adversarial energy-based model via diffusion process. In Forty-first International Conference on Machine Learning, 2024.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. Advances in neural information processing systems, 27, 2014.
- Gozlan, N., Roberto, C., Samson, P.-M., and Tetali, P. Kantorovich duality for general transport costs and applications. Journal of Functional Analysis, 273(11):3327–3405, 2017.
- Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., Schölkopf, B., et al. Covariate shift by kernel mean matching. Dataset shift in machine learning, 3(4): 5, 2009.
- Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. The journal of machine learning research, 13(1):723–773, 2012.
- Groppe, M. and Hundrieser, S. Lower complexity adaptation for empirical entropic optimal transport. Journal of Machine Learning Research, 25(344):1–55, 2024.
- Gu, X., Yang, Y., Zeng, W., Sun, J., and Xu, Z. Keypoint-guided optimal transport with applications in heterogeneous domain adaptation. Advances in Neural Information Processing Systems, 35:14972–14985, 2022.
- Gu, X., Yang, L., Sun, J., and Xu, Z. Optimal transport-guided conditional score-based diffusion model. Advances in Neural Information Processing Systems, 36: 36540–36552, 2023.
- Gushchin, N., Kolesov, A., Korotin, A., Vetrov, D. P., and Burnaev, E. Entropic neural optimal transport via diffusion processes. Advances in Neural Information Processing Systems, 36:75517–75544, 2023a.
- Gushchin, N., Kolesov, A., Mokrov, P., Karpikova, P., Spiridonov, A., Burnaev, E., and Korotin, A. Building the bridge of schrödinger: A continuous entropic optimal transport benchmark. Advances in Neural Information Processing Systems, 36:18932–18963, 2023b.
- Gushchin, N., Kholkin, S., Burnaev, E., and Korotin, A. Light and optimal schrödinger bridge matching. In Forty-first International Conference on Machine Learning, 2024a.
- Gushchin, N., Selikhanovych, D., Kholkin, S., Burnaev, E., and Korotin, A. Adversarial schrödinger bridge matching. Advances in Neural Information Processing Systems, 37: 89612–89651, 2024b.
- Hagemann, P., Hertrich, J., Altekruiger, F., Beinert, R., Chemseddine, J., and Steidl, G. Posterior sampling based on gradient flows of the MMD with negative distance kernel. In The Twelfth International Conference on Learning Representations, 2024. URL <https://openreview.net/forum?id=YrXHEb2qMb>.
- Heitz, M., Bonneel, N., Coeurjolly, D., Cuturi, M., and Peyré, G. Ground metric learning on graphs. Journal of Mathematical Imaging and Vision, 63(1):89–107, 2021.
- Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems, 30, 2017.
- Holderrieth, P., Albergo, M. S., and Jaakkola, T. LEAPS: A discrete neural sampler via locally equivariant networks. In Forty-second International Conference on Machine Learning, 2025. URL <https://openreview.net/forum?id=Hq2RniQAET>.
- Hou, Q., Wang, S., and Liu, J. Semi-supervised dehazing method based on image enhancement and multi-negative contrastive auxiliary learning. International Journal of Machine Learning and Cybernetics, pp. 1–14, 2025.
- Howard, S., Deligiannidis, G., Rebeschini, P., and Thornton, J. Differentiable cost-parameterized monge map estimators. arXiv preprint arXiv:2406.08399, 2024.
- Huizing, G.-J., Cantini, L., and Peyré, G. Unsupervised ground metric learning using wasserstein singular vectors. In International Conference on Machine Learning, pp. 9429–9443. PMLR, 2022.
- Izmailov, P., Kirichenko, P., Finzi, M., and Wilson, A. G. Semi-supervised learning with normalizing flows. In International conference on machine learning, pp. 4615–4630. PMLR, 2020.
- Jawanpuria, P., Shi, D., Mishra, B., and Gao, J. A riemannian approach to ground metric learning for optimal transport. In ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1–5. IEEE, 2025.

- Jiang, Q., Wang, M., Cao, J., Cheng, S., Huang, S., and Li, L. Learning kernel-smoothed machine translation with retrieved examples. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2021.
- Jiang, Y., Jiang, L., Yang, S., and Loy, C. C. Scenimefy: Learning to craft anime scene via semi-supervised image-to-image translation. In *IEEE International Conference on Computer Vision (ICCV)*, 2023a.
- Jiang, Y., Jiang, L., Yang, S., and Loy, C. C. Scenimefy: Learning to craft anime scene via semi-supervised image-to-image translation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7357–7367, 2023b.
- Jin, C.-B., Kim, H., Liu, M., Jung, W., Joo, S., Park, E., Ahn, Y. S., Han, I. H., Lee, J. I., and Cui, X. Deep ct to mr synthesis using paired and unpaired data. *Sensors*, 19(10):2361, 2019.
- Joblove, G. H. and Greenberg, D. Color spaces for computer graphics. In *Proceedings of the 5th annual conference on Computer graphics and interactive techniques*, pp. 20–25, 1978.
- Kantorovich, L. V. On the translocation of masses. In *Dokl. Akad. Nauk. USSR (NS)*, volume 37, pp. 199–201, 1942.
- Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.
- Kerdoncuff, T., Emonet, R., and Sebban, M. Metric learning in optimal transport for domain adaptation. In *International joint conference on artificial intelligence*, pp. 2162–2168. IJCAI, 2020.
- Kingma, D. P. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Korotin, A., Gushchin, N., and Burnaev, E. Light schrödinger bridge. In *The Twelfth International Conference on Learning Representations*, 2024.
- Ksenofontov, G. and Korotin, A. Categorical schrödinger bridge matching. In *Forty-second International Conference on Machine Learning*, 2025. URL <https://openreview.net/forum?id=RBly0nOr2h>.
- LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., Huang, F., et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.
- Léonard, C. A survey of the schrödinger problem and some of its connections with optimal transport. *Discrete & Continuous Dynamical Systems-A*, 34(4):1533–1574, 2014.
- Li, L., Dong, Y., Ren, W., Pan, J., Gao, C., Sang, N., and Yang, M.-H. Semi-supervised image dehazing. *IEEE Transactions on Image Processing*, 29:2766–2779, 2019a.
- Li, R., Ye, X., Zhou, H., and Zha, H. Learning to match via inverse optimal transport. *Journal of machine learning research*, 20(80):1–37, 2019b.
- Li, W. and Chang, C. Semi-supervised image-dehazing network based on a trusted library. *Electronics*, 14(15): 2956, 2025.
- Lin, J., Xia, Y., Qin, T., Chen, Z., and Liu, T.-Y. Conditional image-to-image translation. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.
- Liu, J., Hou, Q., Wang, S., and Zhang, X. Semi-supervised single image dehazing based on dual-teacher-student network with knowledge transfer. *Signal, Image and Video Processing*, 18(6):5073–5087, 2024.
- Ma, S., Sun, H., Ye, X., Zha, H., and Zhou, H. Learning cost functions for optimal transport. *arXiv preprint arXiv:2002.09650*, 2020.
- Malinin, A., Band, N., Chesnokov, G., Gal, Y., Gales, M. J., Noskov, A., Ploskonosov, A., Prokhorenkova, L., Provilkov, I., Raina, V., et al. Shifts: A dataset of real distributional shift across multiple large-scale tasks. *arXiv preprint arXiv:2107.07455*, 2021.
- Marino, S. D. and Gerolin, A. An optimal transport approach for the schrödinger bridge problem and convergence of sinkhorn algorithm. *Journal of Scientific Computing*, 85(2):27, 2020.
- Mirza, M. and Osindero, S. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014. URL <http://arxiv.org/abs/1411.1784>.
- Mo, T., Zheng, S., Chan, W.-Y., and Yang, R. Review of ai image enhancement techniques for in-vehicle vision systems under adverse weather conditions. *World Electric Vehicle Journal*, 16(2):72, 2025.
- Mokrov, P., Korotin, A., Kolesov, A., Gushchin, N., and Burnaev, E. Energy-guided entropic neural optimal transport. In *The Twelfth International Conference on Learning Representations*, 2024.
- Monge, G. Mémoire sur la théorie des déblais et des remblais. *Mem. Math. Phys. Acad. Royale Sci.*, pp. 666–704, 1781.
- Moriakov, N., Adler, J., and Teuwen, J. Kernel of cyclegan as a principal homogeneous space. In *International Conference on Learning Representations*,

2020. URL <https://openreview.net/forum?id=B1eWOJHKvB>.
- Morishita, M., Suzuki, J., and Nagata, M. Domain adaptation of machine translation with crowdworkers. In Conference on Empirical Methods in Natural Language Processing (EMNLP), 2022.
- Murphy, K. P. Machine learning: a probabilistic perspective. MIT press, 2012.
- Mustafa, A. and Mantiuk, R. K. Transformation consistency regularization—a semi-supervised paradigm for image-to-image translation. In Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16, pp. 599–615. Springer, 2020.
- Nagumo, R. and Fujisawa, H. Density ratio estimation with doubly strong robustness. In Forty-first International Conference on Machine Learning, 2024.
- Niles-Weed, J. and Rigollet, P. Estimation of wasserstein distances in the spiked transport model. Bernoulli, 28(4): 2663–2688, 2022.
- Oza, M., Vaghela, H., and Bagul, S. Semi-supervised image-to-image translation. In 2019 International Conference of Artificial Intelligence and Information Technology (ICAIIIT), pp. 16–20. IEEE, 2019.
- Paavilainen, P., Akram, S. U., and Kannala, J. Bridging the gap between paired and unpaired medical image translation. In MICCAI Workshop on Deep Generative Models, pp. 35–44. Springer, 2021.
- Panda, N., Klein, N., Yang, D., Gasda, P., and Oyen, D. Semi-supervised learning of pushforwards for domain translation & adaptation. arXiv preprint arXiv:2304.08673, 2023.
- Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. Normalizing flows for probabilistic modeling and inference. Journal of Machine Learning Research, 22(57):1–64, 2021.
- Peng, D., Hu, P., Ke, Q., and Liu, J. Diffusion-based image translation with label guidance for domain adaptive semantic segmentation. In IEEE International Conference on Computer Vision (ICCV), 2023.
- Peyré, G. and Cuturi, M. Computational Optimal Transport: With Applications to Data Science. Foundations and trends in machine learning. Now Publishers, 2019. ISBN 9781680835502. URL <https://books.google.ru/books?id=J0BiwgEACAAJ>.
- Pidhorskyi, S., Adjero, D. A., and Doretto, G. Adversarial latent autoencoders. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 14104–14113, 2020.
- Pinkus, A. Approximation theory of the mlp model in neural networks. Acta numerica, 8:143–195, 1999.
- Ren, Y., Wu, J., Zhang, P., Zhang, M., Xiao, X., He, Q., Wang, R., Zheng, M., and Pan, X. Ugc: Unified gan compression for efficient image-to-image translation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 17281–17291, 2023.
- Roberts, G. O. and Tweedie, R. L. Exponential convergence of langevin distributions and their discrete approximations. Bernoulli, pp. 341–363, 1996.
- Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, October 5–9, 2015, proceedings, part III 18, pp. 234–241. Springer, 2015.
- Rubachev, I., Kartashev, N., Gorishniy, Y., and Babenko, A. Tabred: Analyzing pitfalls and filling the gaps in tabular deep learning benchmarks. In The Thirteenth International Conference on Learning Representations, 2025. URL <https://openreview.net/forum?id=L14sqcrUC3>.
- Santambrogio, F. Optimal transport for applied mathematicians. Birkhäuser, NY, 55(58-63):94, 2015.
- Sason, I. On reverse pinsker inequalities. arXiv preprint arXiv:1503.07118, 2015.
- Scarvelis, C. and Solomon, J. Riemannian metric learning via optimal transport. In International Conference on Learning Representations. OpenReview, 2023.
- Schröder, T., Ou, Z., Lim, J., Li, Y., Vollmer, S., and Duncan, A. Energy discrepancies: a score-independent loss for energy-based models. Advances in Neural Information Processing Systems, 36:45300–45338, 2023.
- Shi, L., Zhang, G., Zhen, H., Fan, J., and Yan, J. Understanding and generalizing contrastive learning from the inverse optimal transport perspective. In International conference on machine learning, pp. 31408–31421. PMLR, 2023.
- Shi, Y., De Bortoli, V., Campbell, A., and Doucet, A. Diffusion schrödinger bridge matching. Advances in Neural Information Processing Systems, 36, 2024.

- Skibbe, H., Watakabe, A., Rachmadi, F., Gutierrez, C. E., Nakae, K., and Yamamori, T. Semi-supervised image-to-image translation for robust image registration. Medical Imaging with Deep Learning (MIDL), 2021. URL <https://openreview.net/forum?id=GOhAoJdaLg>.
- Song, Y. and Kingma, D. P. How to train your energy-based models. arXiv preprint arXiv:2101.03288, 2021.
- Stuart, A. M. and Wolfram, M.-T. Inverse optimal transport. SIAM Journal on Applied Mathematics, 80(1):599–619, 2020.
- Sugiyama, M., Suzuki, T., and Kanamori, T. Density ratio estimation in machine learning. Cambridge University Press, 2012.
- Tang, X., Hu, X., Gu, X., and Sun, J. Residual-conditioned optimal transport: Towards structure-preserving unpaired and paired image restoration. In Forty-first International Conference on Machine Learning, 2024. URL <https://openreview.net/forum?id=irBHPlknxP>.
- Theodoropoulos, P., Komianos, N., Pacelli, V., Liu, G.-H., and Theodorou, E. A. Feedback schrödinger bridge matching. arXiv preprint arXiv:2410.14055, 2024.
- Tripathy, S., Kannala, J., and Rahtu, E. Learning image-to-image translation using paired and unpaired training samples. In Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part II 14, pp. 51–66. Springer, 2019.
- Vargas, F., Thodoroff, P., Lamacraft, A., and Lawrence, N. Solving schrödinger bridges via maximum likelihood. Entropy, 23(9):1134, 2021.
- Vasluianu, F.-A., Romero, A., Van Gool, L., and Timofte, R. Shadow removal with paired and unpaired learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 826–835, 2021.
- Villani, C. et al. Optimal transport: old and new, volume 338. Springer, 2009.
- Wang, K., Gharbi, M., Zhang, H., Xia, Z., and Shechtman, E. Semi-supervised parametric real-world image harmonization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5927–5936, 2023.
- Wang, M., Huang, W., Gong, M., and Zhang, Z. Projection pursuit density ratio estimation. In Forty-second International Conference on Machine Learning, 2025. URL <https://openreview.net/forum?id=MgNeJO0PcF>.
- Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing, 13(4):600–612, 2004.
- Weed, J. and Bach, F. Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance. Bernoulli, 25(4A):2620–2648, 2019.
- Winkler, C., Worrall, D. E., Hoozeboom, E., and Welling, M. Learning likelihoods with conditional normalizing flows. CoRR, abs/1912.00042, 2019. URL <http://arxiv.org/abs/1912.00042>.
- Yang, Z. and Chen, Z. Learning from paired and unpaired data: Alternately trained cyclegan for near infrared image colorization. In 2020 IEEE International Conference on Visual Communications and Image Processing (VCIP), pp. 467–470. IEEE, 2020.
- Yu, P., Zhu, Y., Xie, S., Ma, X. S., Gao, R., Zhu, S.-C., and Wu, Y. N. Learning energy-based prior model with diffusion-amortized mcmc. Advances in Neural Information Processing Systems, 36:42717–42747, 2023.
- Yuan, Y., Liu, S., Zhang, J., Zhang, Y., Dong, C., and Lin, L. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 701–710, 2018.
- Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 586–595, 2018.
- Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE international conference on computer vision, pp. 2223–2232, 2017.
- Zhu, Y., Xie, J., Wu, Y. N., and Gao, R. Learning energy-based models by cooperative diffusion recovery likelihood. In The Twelfth International Conference on Learning Representations, 2024. URL <https://openreview.net/forum?id=AyzkDpuqcl>.

Appendix

A Neural Parameterization	16
A.1 Algorithm Derivation	17
A.2 Illustrative Example	18
A.3 Colored Images Example	18
A.4 Conclusion	19
B Additional Discussions	19
B.1 Entropic/Weak/Inverse Optimal Transport	19
B.2 Discrete Spaces Extension	21
B.3 Partially Paired Data	22
B.4 Examples of Semi-supervised Domain Translation Setups	22
B.5 Related Works	23
C Additional Experiments	24
C.1 Gaussian to Swiss Roll Mapping	24
C.2 Semi-supervised Classification	27
D Experimental Details	28
D.1 Baseline Details	28
D.2 Metric Computation Details	30
D.3 General Implementation Details	32
D.4 Gaussian to Swiss Roll Mapping Details	32
D.5 Image Translation via ALAE Details	33
D.6 Weather Prediction Details	33
E Proofs	33
E.1 Loss Derivation	33
E.2 Expressions for the Gaussian Parameterization	34
E.3 Gradient of our Loss for Energy-Based Modeling	35
E.4 Universal Approximation	35

A. Neural Parameterization

Throughout the main text, we parameterized the cost c^θ and potential f^θ using log-sum-exp functions and Gaussian mixtures (see §3.3), resulting in the method we refer to as EBiEOT-GMM. A natural question is whether more general parameterizations can be used within our framework, for example by directly parameterizing both c^θ and f^θ with neural networks. In this section, we answer this question affirmatively and introduce EBiEOT-NN, a neural-network-based version

of our method for optimizing the objective $\mathcal{L}(\theta)$ in (15).

A.1. Algorithm Derivation

We note that a key advantage of our chosen parameterization (see §3.3) is that the normalizing constant Z_θ appearing in $\mathcal{L}(\theta)$ is available in the closed form. Unfortunately, this is not the case with general parameterizations of c^θ and f^θ , necessitating the use of more advanced optimization techniques. While the objective $\mathcal{L}(\theta)$ itself may be intractable, we can derive its gradient, which is essential for optimization. The following proposition is derived in a manner similar to (Mokrov et al., 2024), who proposed methods for solving forward entropic OT problems with neural nets.

Proposition A.1 (Gradient of our main loss (15)). *It holds that*

$$\begin{aligned} \frac{\partial}{\partial \theta} \mathcal{L}(\theta) = \varepsilon^{-1} & \left\{ \mathbb{E}_{x,y \sim \pi^*} \left[\frac{\partial}{\partial \theta} c^\theta(x, y) \right] - \mathbb{E}_{y \sim \pi_y^*} \left[\frac{\partial}{\partial \theta} f^\theta(y) \right] \right. \\ & \left. + \mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^\theta(\cdot|x)} \left[\frac{\partial}{\partial \theta} (f^\theta(y) - c^\theta(x, y)) \right] \right\}. \end{aligned} \quad (21)$$

The gradient formula eliminates the need for the intractable normalizing constant Z_θ , but computing it still requires sampling from the current model $y \sim \pi^\theta(\cdot|x)$. Unlike the Gaussian mixture case in §3.3, we now only access the unnormalized density defined by c^θ and f^θ , which is not necessarily a Gaussian mixture. To address this, we rely on standard methods for sampling from unnormalized densities, such as Markov Chain Monte Carlo (MCMC) (Andrieu et al., 2003). This enables practical gradient estimation and motivates the training procedure in Algorithm 1, where the conditional distribution is modeled as $\pi^\theta(y|x) \propto \exp\left(\frac{f^\theta(y) - c^\theta(x, y)}{\varepsilon}\right)$, with energy $\varepsilon^{-1}(c^\theta(x, y) - f^\theta(y))$.

Algorithm 1: EBiEOT-NN

Input : Paired samples $XY_{\text{paired}} \sim \pi^*$; unpaired samples $X_{\text{unpaired}} \sim \pi_x^*$, $Y_{\text{unpaired}} \sim \pi_y^*$;
 potential network $f^\theta : \mathbb{R}^{D_y} \rightarrow \mathbb{R}$, cost network $c^\theta(x, y) : \mathbb{R}^{D_x} \times \mathbb{R}^{D_y} \rightarrow \mathbb{R}$;
 number of Langevin steps $K > 0$, Langevin discretization step size $\eta > 0$;
 basic noise std $\sigma_0 > 0$; batch sizes $\hat{P}, \hat{Q}, \hat{R} > 0$.

Output : trained potential network f^{θ^*} and cost network c^{θ^*} recovering $\pi^{\theta^*}(y|x)$ from (12).

for $i = 1, 2, \dots$ **do**

Derive batches $\{\hat{x}_p, \hat{y}_p\}_{p=1}^{\hat{P}} = XY \sim \pi^*$, $\{\hat{x}_n\}_{n=1}^{\hat{Q}} = X \sim \pi_x^*$, $\{\hat{y}_r\}_{r=1}^{\hat{R}} = Y \sim \pi_y^*$;

Sample basic noise $Y^{(0)} \sim \mathcal{N}(0, \sigma_0)$ of size $\hat{Q}$;

for $k = 1, 2, \dots, K$ **do**

Sample $Z^{(k)} = \{z_q^{(k)}\}_{q=1}^{\hat{Q}}$, where $z_q^{(k)} \sim \mathcal{N}(0, 1)$;

Obtain $Y^{(k)} = \{y_q^{(k)}\}_{q=1}^{\hat{Q}}$ with Langevin step:

$y_q^{(k)} \leftarrow y_q^{(k-1)} + \frac{\eta}{2\varepsilon} \cdot \text{stop\_grad}\left(\frac{\partial}{\partial y} [f^\theta(y) - c^\theta(x_q, y)] \Big|_{y=y_q^{(k-1)}}\right) + \sqrt{\eta} z_q^{(k)}$

$\hat{\mathcal{L}} \leftarrow \frac{1}{\hat{P}} \left[\sum_{x_p, y_p \in XY} c^\theta(x_p, y_p) \right] + \frac{1}{\hat{Q}} \left[\sum_{x_q \in X, y_q^{(K)} \in Y^{(K)}} (f^\theta(y_q^{(K)}) - c^\theta(x_q, y_q^{(K)})) \right] - \frac{1}{\hat{R}} \left[\sum_{y_r \in Y} f_\theta(y_r) \right];$

Perform a gradient step over θ by using $\frac{\partial \hat{\mathcal{L}}}{\partial \theta}$;

In Algorithm 1, we use the Unadjusted Langevin Algorithm (ULA) (Roberts & Tweedie, 1996), a standard MCMC method. For an in-depth discussion on EBM training methods, see the recent surveys (Song & Kingma, 2021; Carbone, 2024).

Our proposed *inverse* OT algorithm is closely related to the *forward* OT framework in (Mokrov et al., 2024, Algorithm 1), with key distinctions: (1) it learns the cost function c^θ during training, and (2) it leverages both paired and unpaired data.

Below, we demonstrate a proof-of-concept performance of Algorithm 1 on two setups: a 2D example and colored images.

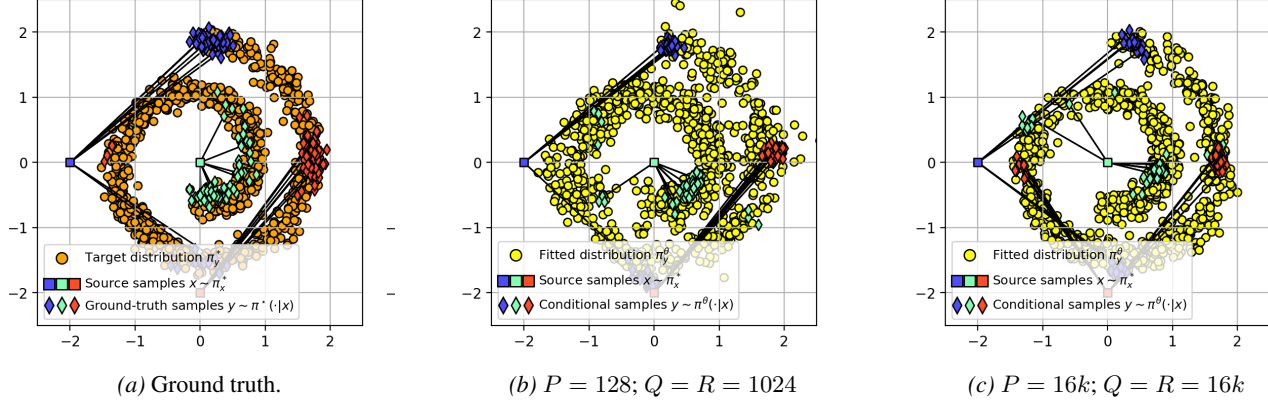


Figure 4. Performance of the proposed neural parameterization in the *Gaussian* $\rightarrow$ *Swiss Roll* mapping task (§5.1). We use MLPs to parametrize both the potential function f^θ and the cost function c^θ .

A.2. Illustrative Example

Setup. We begin with a 2D example to showcase the capability of EBiEOT-NN to learn conditional plans using a fully neural network-based parameterization. Specifically, we conduct experiments on the *Gaussian* $\rightarrow$ *Swiss Roll* mapping problem (see §5.1) using two datasets: one containing 128 paired samples (described in §5.1) and another with 16k paired samples (see Appendix C.1.1).

Discussion. It is worth noting that the model’s ability to fit the target distribution is influenced by the amount of labeled data used during training. When working with partially labeled samples (as shown in Figure 4b), the model’s fit to the target distribution is less accurate compared to using a larger dataset. However, even with limited labeled data, the model still maintains good accuracy in terms of the paired samples. On the other hand, when provided with fully labeled data (see Figure 4c), the model generates more consistent results and achieves a better approximation of the target distribution. A comparison of the results obtained using EBiEOT-NN with neural network parameterization and those achieved using EBiEOT-GMM with Gaussian parameterization (Figure 3d) reveals that EBiEOT-NN exhibits greater instability. This observation aligns with the findings of (Mokrov et al., 2024, Section 2.2), which emphasize the instability and mode collapse issues commonly encountered when working with EBMs.

Implementation Details. We employ MLPs with hidden layer configurations of [128, 128] and [256, 256, 256], using *LeakyReLU*(0.2) for the parameterization of the potential f^θ and the cost c^θ , respectively. The learning rates are set to $lr_{\text{paired}} = 5 \times 10^{-4}$ and $lr_{\text{unpaired}} = 2 \times 10^{-4}$. The sampling parameters follow those specified in (Mokrov et al., 2024).

A.3. Colored Images Example

Setup. We adapted an experiment from (Mokrov et al., 2024) using the colored MNIST dataset (Arjovsky et al., 2019). While the original task involved translating digit 2 into digit 3 using unpaired images, we modified the setup to demonstrate our method’s ability to perform translations according to paired data. Namely, we created pairs by shifting the hue (Joblove & Greenberg, 1978) of the source images by 120° . Specifically, for a source image with a hue h in the range $0^\circ \leq h < 360^\circ$, the target image’s hue was set to $(h + 120^\circ) \bmod 360^\circ$.

Discussion. The results of this experiment are shown in Figure 5. Notably, the model successfully learned the color transformation using only 10 pairs (third row). Increasing the number of pairs to 200 further improved the quality of the translation (fourth row).

Implementation Details. We adopt the same parameters as in (Mokrov et al., 2024), except for the cost function:

$$c^\theta(x, y) = \frac{1}{D_y} \|U_{\text{net}}^\theta - y\|_2^2.$$

Here, the dimensions of source and target spaces are $D_x = D_y = 3 \times 32 \times 32$ and $U_{\text{net}}^\theta : \mathbb{R}^{D_x} \rightarrow \mathbb{R}^{D_y}$ is a neural net function with U-Net architecture (Ronneberger et al., 2015) with 16 layers. The first layer has 64 filters, and the number of filters doubles in each subsequent layer. The experiment was run for 10,000 iterations on a 2080 Ti GPU, completing in approximately 40 minutes.

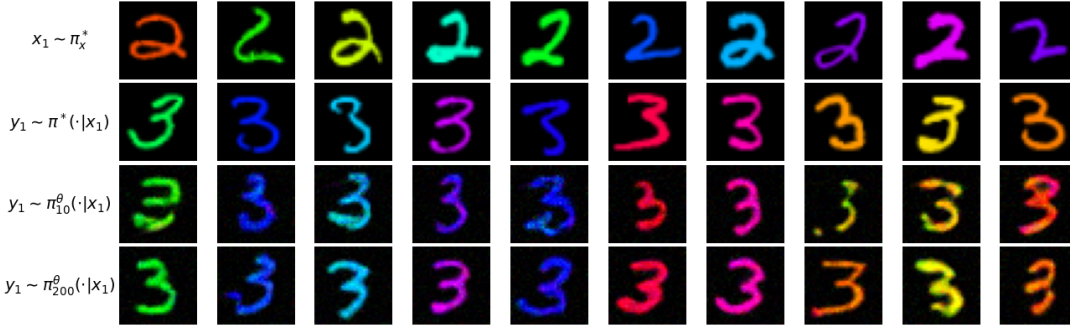


Figure 5. Performance of the proposed EBiEOT-NN method

on the colored MNIST mapping task. Each pair consists of digits 2 and 3 with a hue shift of 120° . The first row shows the source images, the second row displays target images with ground-truth colors, the third row presents the mapping results for 10 pairs in the train data, and the fourth row shows results for 200 pairs.

A.4. Conclusion

It is important to recognize that the field of Energy-Based Models has undergone significant advancements in recent years, with the development of numerous scalable approaches. For examples of such progress, we refer readers to recent works by (Geng et al., 2024; Carbone et al., 2023; Du et al., 2021; Gao et al., 2021) and other the references therein. Additionally, we recommend the comprehensive tutorial by (Song & Kingma, 2021; Carbone, 2024) for an overview of training methods for EBMs. Given these advancements, it is reasonable to expect that by incorporating more sophisticated techniques into our basic Algorithm 1, it may be possible to scale the method to handle high-dimensional setups, such as image data. However, exploring these scaling techniques is beyond the scope of the current paper, which primarily focuses on the general methodology for semi-supervised domain translation. We leave the development of methods for further scaling our approach as a promising direction for future research.

B. Additional Discussions

B.1. Entropic/Weak/Inverse Optimal Transport

In this section, we explain our motivation for adopting the *entropic* OT formulation rather than the standard OT formulation (OT). Specifically, we focus on the *weak semi-dual* formulation of the entropic OT problem (Mokrov et al., 2024, §3.1), as opposed to its standard *semi-dual* form (Genevay, 2019, §4.3), and highlight its connections to the existing inverse entropic optimal transport frameworks in the literature.

Classic OT. Given source and target distributions $\alpha \in \mathcal{P}_{ac}(\mathcal{X})$ and $\beta \in \mathcal{P}_{ac}(\mathcal{Y})$, and a cost function $c^* : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$, the *primal* optimal transport problem (Villani et al., 2009) is defined as:

$$\text{OT}_{c^*}(\alpha, \beta) \stackrel{\text{def}}{=} \min_{\pi \in \Pi(\alpha, \beta)} \mathbb{E}_{x, y \sim \pi} [c^*(x, y)]. \quad (\text{OT})$$

This formulation was originally introduced by Kantorovich (Kantorovich, 1942) as a relaxation of Monge’s original problem (Monge, 1781), which is more restrictive because it does not allow mass to be split, resulting in *deterministic* solutions called optimal transport *maps*. However, as is well-known in optimal transport theory (see (Villani et al., 2009, §9)), solutions to problem (OT), called optimal transport *plans*, can still be deterministic. For example, when the cost is quadratic and the measures are absolutely continuous on the same space $\mathbb{R}^{D_x} = \mathbb{R}^{D_y}$, Brenier’s theorem (Peyré & Cuturi, 2019, Remark 2.24) guarantees that the optimal transport plan is deterministic. Specifically, each x is mapped deterministically to $y = T^*(x)$ for some optimal map T^* , meaning that the conditional distribution $\pi^*(y|x)$ collapses to a single point mass $\delta_{T^*(x)}$.

Such deterministic plans, however, are unsuitable for our semi-supervised domain translation setup, where a multimodal transport behavior of $\pi^*(y|x)$ may be necessary. Our synthetic experiments in §5.1 (Figure 3) illustrate these cases. To enforce mapping uniqueness while allowing *stochastic* (i.e., non-deterministic) mappings, a common approach is to regularize (OT) with an entropy term, which makes the objective strictly convex with respect to π , as discussed below.

Entropic OT. The work of (Cuturi, 2013) proposed regularizing (OT) with an entropy term, known as entropic OT (EOT), to improve computational tractability of OT (Genevay, 2019). Moreover, besides the computational advantages, the EOT

problem has a connection to the *Static Schrödinger Bridge* (SB) problem (Léonard, 2014):

$$\pi^* = \arg \min_{\pi \in \Pi(\alpha, \beta)} \text{KL}(\pi \| \pi^{\text{ref}}), \quad (\text{SB})$$

where the aim of the problem is to find the transport plan $\pi \in \Pi(\alpha, \beta)$ closest to π^{ref} in terms of the Kullback-Leibler (KL) divergence. Observe that EOT and the static SB problem are equivalent:

$$\min_{\pi \in \Pi(\alpha, \beta)} \text{KL}(\pi \| \pi^{\text{ref}}) = \min_{\pi \in \Pi(\alpha, \beta)} \mathbb{E}_{x, y \sim \pi} \log \frac{\pi(x, y)}{\pi^{\text{ref}}(x, y)} = \quad (22)$$

$$\min_{\pi \in \Pi(\alpha, \beta)} \left\{ \mathbb{E}_{x, y \sim \pi} \underbrace{[-\log \pi^{\text{ref}}(x, y)]}_{\stackrel{\text{def}}{=} c^*(x, y)} - H(\pi) \right\} = \min_{\pi \in \Pi(\alpha, \beta)} \left\{ \mathbb{E}_{x, y \sim \pi} [c^*(x, y)] - H(\pi) \right\}. \quad (23)$$

Using the equivalence of the following formulations (see (Mokrov et al., 2024, Eq. 2–4) and (Gushchin et al., 2023b, Eq. 3–5) for details):

$$\begin{cases} \text{EOT}_{c^*, \varepsilon}^{(1)}(\alpha, \beta) \\ \text{EOT}_{c^*, \varepsilon}^{(2)}(\alpha, \beta) \\ \text{EOT}_{c^*, \varepsilon}(\alpha, \beta) \end{cases} = \min_{\pi \in \Pi(\alpha, \beta)} \mathbb{E}_{x, y \sim \pi} [c^*(x, y)] + \begin{cases} +\varepsilon \text{KL}(\pi \| \alpha \otimes \beta), \\ -\varepsilon H(\pi), \\ -\varepsilon \mathbb{E}_{x \sim \alpha} H(\pi(\cdot | x)), \end{cases}$$

we conclude that equation (23) is equivalent to (1) for $\varepsilon = 1$.

From the equations (22)–(23), we see that the cost function $c^*(x, y)$ defines a reference measure that determines the mapping we aim to reconstruct in the forward problem (1). Furthermore, since KL minimization is equivalent to maximum likelihood estimation, EOT is theoretically consistent with standard probabilistic modeling principles.

Weak OT. Following (Mokrov et al., 2024), we provide more details regarding *weak* OT. For a more rigorous treatment, see (Gozlan et al., 2017; Backhoff-Veraguas et al., 2019). Given a *weak* transport cost $C^* : \mathcal{X} \times \mathcal{P}(\mathcal{Y}) \rightarrow \mathbb{R}$, which penalizes the displacement of a point $x \in \mathcal{X}$ into a distribution $\pi(\cdot | x) \in \mathcal{P}(\mathcal{Y})$, the weak OT problem is defined as:

$$\text{WOT}_{C^*}(\alpha, \beta) \stackrel{\text{def}}{=} \min_{\pi \in \Pi(\alpha, \beta)} \mathbb{E}_{x \sim \alpha} C^*(x, \pi(\cdot | x)). \quad (\text{p-WOT})$$

Just as in the classical OT problem (OT), the weak OT formulation (p-WOT) also enjoys *strong duality* under mild assumptions (see (Gozlan et al., 2017, Theorem 9.5); (Backhoff-Veraguas & Pammer, 2022, Theorem 3.3)). This means that the weak formulation (p-WOT) admits an equivalent *weak semi-dual* representation:

$$\text{WOT}_{C^*}(\alpha, \beta) = \max_{f \in \mathcal{C}(\mathcal{Y})} \left\{ \mathbb{E}_{x \sim \alpha} f^{C^*}(x) + \mathbb{E}_{y \sim \beta} f(y) \right\}, \quad (\text{sd-WOT})$$

where $\mathcal{C}(\mathcal{Y})$ denotes the set of continuous functions over $\mathcal{Y}$ and f^C so-called *weak C-transform*:

$$f^C(x) \stackrel{\text{def}}{=} \min_{\mu \in \mathcal{P}(\mathcal{Y})} \{C(x, \mu) - \mathbb{E}_{y \sim \mu} f(y)\}. \quad (24)$$

Furthermore, note that the EOT formulation in (1) can be seen as a special case of the weak OT problem (p-WOT), corresponding to the following weak transport cost C_{EOT}^* :

$$C_{\text{EOT}}^*(x, \pi(\cdot | x)) \stackrel{\text{def}}{=} \mathbb{E}_{y \sim \pi(\cdot | x)} [c^*(x, y)] - \varepsilon H(\pi(\cdot | x)). \quad (25)$$

Substituting expression above into (24), we obtain equation (3) for the weak *entropic c-transform*:

$$f^c(x) = \min_{\mu \in \mathcal{P}(\mathcal{Y})} \{ \mathbb{E}_{y \sim \mu} [c^*(x, y)] - \varepsilon H(\mu) - \mathbb{E}_{y \sim \mu} f(y) \},$$

which admits a closed-form expression given in (Mokrov et al., 2024, Eq. 14), and which we use in our work (4):

$$f^c(x) = -\varepsilon \log \int_{\mathcal{Y}} \exp \left(\frac{f(y) - c(x, y)}{\varepsilon} \right) dy.$$

Furthermore, Appendix A.1 of (Mokrov et al., 2024) provides a detailed discussion of the relationship between the weak entropic c -transform and the so-called (c, ε) -transform (Genevay et al., 2019, 4.15), (Marino & Gerolin, 2020, Theorem 1.2):

$$v^{c, \varepsilon}(x) = -\varepsilon \log \mathbb{E}_{y \sim \beta} \left[\exp \left(\frac{v(y) - c(x, y)}{\varepsilon} \right) \right], \quad (26)$$

which is used in the *semi-dual* formulation of the EOT problem (Genevay, 2019, §4.3):

$$\text{EOT}_{c^*, \varepsilon}^{\text{semi-dual}}(\alpha, \beta) = \max_{v \in \mathcal{C}(\mathcal{Y})} \left\{ \mathbb{E}_{x \sim \alpha} v^{c^*, \varepsilon}(x) + \mathbb{E}_{y \sim \beta} v(y) \right\}. \quad (\text{sd-EOT})$$

As noted in (Mokrov et al., 2024), the main difference between (4) and (26) lies in the integration measure: (26) integrates with respect to β , while (4) uses the standard Lebesgue measure.

For completeness, we present below the *dual* formulation of EOT with a slightly different regularization term, $+\varepsilon \text{KL}(\pi \| \alpha \otimes \beta)$. As noted above, this is equivalent to our choice of regularization, but it is the version commonly used in inverse problems and will be discussed later:

$$\begin{aligned} \text{EOT}_{c^*, \varepsilon}^{\text{dual}}(\alpha, \beta) = \max_{\substack{u \in \mathcal{C}(\mathcal{X}) \\ v \in \mathcal{C}(\mathcal{Y})}} & \left\{ \mathbb{E}_{x \sim \alpha} u(x) + \mathbb{E}_{y \sim \beta} v(y) \right. \\ & \left. - \mathbb{E}_{x, y \sim \alpha \otimes \beta} \left[\exp \left(\frac{u(x) + v(y) - c(x, y)}{\varepsilon} \right) \right] \right\}, \end{aligned} \quad (\text{d-EOT})$$

where the optimization is performed over two Kantorovich potentials u and v , in contrast to the single potential used in our formulation (15). With that said, we are ready to discuss the existing formulations of inverse entropic optimal transport.

Inverse OT. The use of the entropic formulation for inverse optimal transport was first proposed in (Du & Mordatch, 2019, Eq. 8). Their setup, identical to our formulation (5), restricted attention to bilinear cost functions of the form $c_A(x, y) = x^\top A y$, (Eq. 5), with the goal of recovering the matrix A in a discrete setting. A subsequent work (Ma et al., 2020, Eq. 21) extended this idea to the continuous setting by introducing a loss function for learning cost functions, based on the dual formulation (d-EOT) of the EOT problem. As shown in (Andrade et al., 2023, Appendix A.1), their formulation and ours are equivalent, and both admit a maximum likelihood interpretation, consistent with our derivation in §3.1.

The most directly related approach is that of (Andrade et al., 2025, Lemma 1), which addresses the unbalanced OT framework (Chizat et al., 2018) while still relying on the dual formulation (d-EOT). They employ a linearly parameterized cost function (Eq. 4), but their focus is on establishing bounds for cost recovery, in contrast to our emphasis on semi-supervised domain translation. For further formulations of inverse OT, we refer readers to the works cited in the introduction of (Andrade et al., 2023).

B.2. Discrete Spaces Extension

Our theoretical framework is not limited to continuous spaces $\mathcal{X}, \mathcal{Y}$. For instance, if the target space $\mathcal{Y}$ is discrete and takes values in a finite set $\mathbb{Y}_K = \{y_1, \dots, y_K\}$, such as a set of categories, our method remains directly applicable. In this case, the dual potential f^θ (18) can be represented as a vector of length K , and the cost function $c^\theta(x, y)$ (17) can be implemented with a standard neural network. The partition function $Z^\theta(x)$ can then be computed as a finite sum over the K terms:

$$Z^\theta(x) \stackrel{\text{def}}{=} \int_{\mathcal{Y}} \exp(-E^\theta(y|x)) dy = \sum_{k=1}^K \exp \left(\frac{f^\theta(y_k) - c^\theta(x, y_k)}{\varepsilon} \right), \quad (27)$$

making the implementation straightforward. Note that the input x can be either continuous or discrete - it does not affect the formulation. Challenges arise when y is a more complex discrete object, such as a structured output like a sequence of T tokens drawn from a dictionary of size K , i.e., $\mathbb{Y}_K^T$. In such cases, parameterizing f_θ , computing Z_θ , and sampling from the associated energy-based model become significantly more difficult, requiring advanced inference and training techniques, see (Holderrieth et al., 2025) for details. Discrete domains (Austin et al., 2021; Campbell et al., 2022; Gat et al., 2024; Ksenofontov & Korotin, 2025) have received considerable attention recently, and extending our methodology to such spaces represents a promising direction for future research.

B.3. Partially Paired Data

Equation (15) assumes that paired training samples accurately reflect the true marginal distributions π_x^* and π_y^* . One might argue that if these samples are biased or artificially selected, the objective would no longer align with the KL functional derivation in §3.1. However, we show below that the theoretical framework can be corrected under density-ratio assumptions.

Assume that the observed pairs (x, y) come from a joint distribution π_{subset}^* supported on a limited subset of the support of π^* , with x -marginal μ_x and conditional density $\pi^*(y|x)$. In this setting, the induced y -marginal is $\nu_y(y) = \mathbb{E}_{x \sim \mu_x} \pi^*(y|x)$ and the *ground-truth joint density* becomes

$$\pi_{\text{subset}}^*(x, y) = \mu_x(x) \pi^*(y|x).$$

Applying the same derivation as in §3.1, we obtain:

$$\text{KL}(\pi_{\text{subset}}^* \parallel \pi^\theta) = \underbrace{\text{KL}(\mu_x \parallel \pi_x^\theta)}_{\text{Marginal}} + \underbrace{\mathbb{E}_{x \sim \mu_x} \text{KL}(\pi^*(\cdot|x) \parallel \pi^\theta(\cdot|x))}_{\text{Conditional}}. \quad (28)$$

Analogously to §3.1, we consider only the conditional term:

$$\mathbb{E}_{x \sim \mu_x} \mathbb{E}_{y \sim \pi^*(\cdot|x)} [\log \pi^*(y|x) - \log \pi^\theta(y|x)] = -\mathbb{E}_{x \sim \mu_x} H(\pi^*(\cdot|x)) - \mathbb{E}_{x, y \sim \pi_{\text{subset}}^*} \log \pi^\theta(y|x). \quad (29)$$

Thus, we recover the same conditional log-likelihood structure as in (11). Substituting the EBM parameterizations (12) and (14), we obtain

$$\mathbb{E}_{x, y \sim \pi_{\text{subset}}^*} [c(x, y)] - \mathbb{E}_{y \sim \nu_y} f(y) + \mathbb{E}_{x \sim \mu_x} \log Z^\theta(x) = \quad (30)$$

$$\mathbb{E}_{x, y \sim \pi_{\text{subset}}^*} [c(x, y)] - \mathbb{E}_{y \sim \pi_y^*} \left[\frac{\nu_y(y)}{\pi_y^*(y)} f(y) \right] + \mathbb{E}_{x \sim \pi_x^*} \left[\frac{\mu_x(x)}{\pi_x^*(x)} \log Z^\theta(x) \right]. \quad (31)$$

Introducing the *weights*:

$$w_x(x) = \frac{\mu_x(x)}{\pi_x^*(x)}, \quad w_y(y) = \frac{\nu_y(y)}{\pi_y^*(y)}, \quad (32)$$

we obtain the corrected objective:

$$\mathcal{L}_q(\theta) = \underbrace{\varepsilon^{-1} \mathbb{E}_{x, y \sim \pi_{\text{subset}}^*} [c^\theta(x, y)]}_{\text{Joint, requires pairs } (x, y) \sim \pi_{\text{subset}}^*} - \underbrace{\varepsilon^{-1} \mathbb{E}_{y \sim \pi_y^*} [w_y(y) f^\theta(y)]}_{\text{Marginal, requires } y \sim \pi_y^*} + \underbrace{\mathbb{E}_{x \sim \pi_x^*} [w_x(x) \log Z^\theta(x)]}_{\text{Marginal, requires } x \sim \pi_x^*} \rightarrow \min_\theta. \quad (33)$$

A practical way to estimate the required ratios is classifier-based density ratio estimation, widely used in covariate-shift adaptation (Gretton et al., 2009; Sugiyama et al., 2012). To estimate a marginal ratio such as $w_x(x) = \mu_x(x)/\pi_x^*(x)$, we draw samples from the true marginal π_x^* and the biased marginal μ_x , label them as target (1) and observed (0), and train a probabilistic classifier $s_\varphi(x) = \text{Prob}(\text{target}|x)$. With balanced class priors, $\hat{w}_x(x) = \frac{s_\varphi(x)}{1-s_\varphi(x)}$. The same holds for $w_y(y)$. This method requires no density estimation. For recent advancement in density ratio estimation, please refer to (Nagumo & Fujisawa, 2024; Wang et al., 2025). Thus, even if the paired data are *artificially biased*, the loss remains correct as long as the true marginals are known and appropriate weights are applied.

B.4. Examples of Semi-supervised Domain Translation Setups

In this section we outline some real-world scenarios, where *semi-supervised* setups are very natural.

- **Image Harmonization in Photo Editing** (Wang et al., 2023). Photo compositing often involves placing a foreground object into a new background, but realistic blending (e.g., matching lighting and color tone) is challenging. While only a small set of artist-labeled (paired) composites may be available, large collections of unlabeled (unpaired) composites can be gathered from the web.
- **Scene Stylization (e.g., Anime Rendering)** (Jiang et al., 2023b). Transforming real-world photos into anime-style renderings is popular in gaming and animation but is limited by the scarcity of labeled real-anime image pairs.

- **Image Enhancement for Outdoor Vision** (Li et al., 2019a; Liu et al., 2024; Cui et al., 2024; Li & Chang, 2025; Hou et al., 2025). Adverse weather and low-light conditions can compromise the visual systems of autonomous vehicles, such as self-driving cars and UAVs, leading to challenges in both decision-making and navigation. For a comprehensive overview of these scenarios and existing semi-supervised approaches, see (Mo et al., 2025).
- **Biomedical Image Registration (Microscopy)** (Skibbe et al., 2021). In neuroscience research, aligning images from different modalities (e.g., tracer vs. Nissl stain) is crucial but difficult due to modality shifts. Only a limited number of images can be manually registered (paired data), while many are unregistered (unpaired).

The examples above underscore the importance of developing methods for semi-supervised domain translation, which have applications in rendering, image editing, design, computer graphics, and autonomous driving, while also streamlining existing digital content creation pipelines. At the same time, it is important to recognize that the rapid advancement of generative models may have unintended consequences, potentially impacting certain jobs within these industries.

B.5. Related Works

This section provides the detailed discussion of related work and methods that were only briefly summarized in §4, and includes additional coverage of metric learning.

Semi-supervised models. As discussed in §1, many existing semi-supervised domain translation methods combine paired and unpaired data by introducing multiple loss terms into *ad hoc optimization objectives*. Several works, such as (Jin et al., 2019, §3.3), (Tripathy et al., 2019, §3.5), (Oza et al., 2019, §C), (Paavilainen et al., 2021, §2), (Chen et al., 2023, §3.3), (Ren et al., 2023, §3) and (Panda et al., 2023, Eq. 8), employ GAN-based objectives, which incorporate the GAN losses (Goodfellow et al., 2014) augmented with specific regularization terms to utilize paired data. Although most of these methods were initially designed for the image-to-image translation, their dependence on GAN objectives enables their application to broader domain translation tasks. In contrast, the approaches introduced by (Mustafa & Mantiuk, 2020, §3.2) and (Tang et al., 2024, Eq. 8) employ loss functions specifically tailored for the image-to-image translation, making them unsuitable for the general domain translation problem described in §2.1.

Another line of research explores methods based on *key-point guided OT* (Gu et al., 2022), which integrates paired data information into the discrete transport plan. Building on this concept, (Gu et al., 2023) uses such transport plans as heuristics to train a conditional score-based model on unpaired or semi-paired data. Furthermore, recent work (Theodoropoulos et al., 2024) heuristically incorporates paired data into the cost function $c(x, y)$ in (1) with corresponding dynamical formulation.

Importantly, the paradigms outlined above do not offer any theoretical guarantees for reconstructing the conditional distribution $\pi^*(y|x)$, as they depend on heuristic loss constructions. We show that such approaches actually fail to recover the true plan even in toy 2-dimensional cases, refer to experiments in §5 for an illustrative example. We also note that there exist works addressing the question of incorporating unpaired data to the log-likelihood training (11) by adding an extra likelihood terms, see (Atanov et al., 2019; Izmailov et al., 2020). However, they rely on x being a discrete object (e.g., a class label) and do not easily generalize to the continuous case, see Appendix D.1 for details.

Inverse OT solvers. As highlighted in §2.2, the task of inverse optimal transport (IOT) implies learning the cost function from samples drawn from an optimal coupling π^* . Existing IOT solvers (Dupuy et al., 2019; Li et al., 2019b; Stuart & Wolfram, 2020; Galichon & Salanié, 2022; Andrade et al., 2025) focus on reconstructing cost functions primarily from discrete marginal distributions, in particular, using the log-likelihood maximization techniques (Dupuy et al., 2019), see the introduction of (Andrade et al., 2023) for a review. Additionally, the recent work by (Shi et al., 2023) explores the IOT framework in the context of contrastive learning. In contrast, we develop a log-likelihood based approach aimed at learning conditional distributions $\pi^\theta(\cdot|x) \approx \pi^*(\cdot|x)$ using both paired and unpaired data but not the cost function itself.

Forward OT solvers. Our solver is based on the framework of (Mokrov et al., 2024), which proposed a *forward* solver for *unsupervised* domain translation. In contrast, our approach integrates the optimization of the cost function directly into the objective (equation (20)), allowing for effective utilization of paired data. Additionally, we extend the Gaussian Mixture parameterization proposed by (Korotin et al., 2024; Gushchin et al., 2024a), which was originally developed as a forward solver for entropic OT with a quadratic cost function $c^*(x, y) = \frac{1}{2}\|x - y\|_2^2$. Our work generalizes this solver to accommodate a wider variety of cost functions, as specified in equation (17). As a result, our approach also functions as a novel forward solver for these generalized cost functions.

Recent work by (Howard et al., 2024) proposes a framework for learning cost functions to improve the mapping between the

domains. However, it is limited by the use of deterministic mappings, i.e., does not have the ability to model non-degenerate conditional distributions.

Another work by (Asadulaev et al., 2024) introduces a neural network-based OT framework for semi-supervised scenarios, utilizing general cost functionals for OT. However, their method requires *manually* constructing cost functions which can incorporate class labels or predefined pairs. In contrast, our method dynamically adjusts the cost function during training, offering a more flexibility.

Metric-learning and OT. In addition to purely inverse OT approaches, there is a line of work that aims to learn the ground metric used by optimal transport. A seminal work (Cuturi & Avis, 2014) introduced *ground metric learning* in a supervised setting, where they optimize over metric matrices so that OT distances between labeled histograms better reflect the class structure. Building on this, (Huizing et al., 2022) propose *unsupervised ground metric learning* via what they call Wasserstein singular vectors. They jointly learn a ground metric on features and a distance between samples by finding positive singular vectors of the mapping from metric matrices to OT distance matrices. Their method uses stochastic approximation with entropic regularization and is scalable to high-dimensional data. More recently, the work (Auffenberg et al., 2025) analyze this fixed-point problem more deeply: they prove convergence for a stochastic fixed-point iteration (even in scenarios where classical contraction assumptions do not hold) and show that their framework naturally recovers Mahalanobis-type metrics and graph-Laplacian parameterizations as special cases.

In another direction, (Scarvelis & Solomon, 2023) introduce a *Riemannian metric-learning* framework: they parametrize a spatially-varying metric tensor as a neural network over a manifold, and optimize it so that OT distances under this learned geometry better explain meaningful interpolations, such as trajectories in scRNA data or bird migration. In graph-structured domains, (Heitz et al., 2021) learn ground metrics constrained to be geodesic distances on a graph, allowing a structured and efficient metric learning aligned with the graph topology.

Moreover, (Jawanpuria et al., 2025) propose to learn a symmetric positive definite (SPD) ground metric matrix by optimizing over the Riemannian manifold of SPD matrices, enabling the cost metric to adapt flexibly to data while jointly optimizing the OT distance. Finally, in the context of *domain adaptation*, (Kerdoncuff et al., 2020) present MLOT, which learns a global Mahalanobis metric that improves the alignment of source and target distributions under OT.

While these metric-learning works learn a distance function (or cost metric) via OT, they typically assume particular parametric forms (Mahalanobis, SPD matrices, or constructed on manifolds) and focus on matching distributions or aligning domains. In contrast, our approach learns conditional couplings $\pi^\theta(\cdot|x)$ (not just a ground cost), and integrates cost learning dynamically into a likelihood-based solver over paired and unpaired data. Moreover, our cost parameterization extends beyond classical metric forms, enabling more flexible and expressive cost functions (see Eq. (17)).

C. Additional Experiments

C.1. Gaussian to Swiss Roll Mapping

C.1.1. BASELINES WITH THE LARGE AMOUNT OF DATA (16K)

In this section, we show the results of training of the baselines on the large amount of both paired (16k) and unpaired (16k) data (Figure 6). Recall that the ground truth π^* is depicted in Figure 3c.

Metrics. As in §5.1, we evaluate the agreement between the target and learned conditional distributions using Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) and $\mathcal{W}_2$, approximated with the Sinkhorn divergence (Feydy et al., 2019). Definitions of both metrics and implementation details are provided in Appendix D.2. The quantitative results are reported in Table 5. The trends observed in Table 5 are consistent with the qualitative results shown in Figure 3 for $P = 128$ and Figure 6 for 16k samples. In the large-sample regime, most methods successfully reconstruct the marginal distributions, but still have difficulty accurately learning the conditional distribution.

Discussion. Regression fails to learn anything meaningful due to the averaging effect (Figure 6f). In contrast, the unconditional GAN+ ℓ^2 (Figure 6g) nearly succeeds in generating the target data π_y^* , but the learned plan is still incorrect, also due to the averaging effect. Given a sufficient amount of training data, Conditional GAN (Figure 6h) nearly succeeds in learning the true conditional distributions $\pi^*(\cdot|x)$. The same applies to the conditional normalizing flow (Figure 6i), but its results are slightly worse, presumably due to the limited expressiveness of invertible flow architecture.

Experiments using the natural semi-supervised loss function in (36) demonstrate that this loss function can reasonably

recover the conditional mapping with both CondNF (Figure 6j) and CGMM (Figure 6e) parameterizations. However, it requires significantly more training data compared to our proposed loss function (15). This conclusion is supported by the observation that the CGMM model trained with (36) tends to overfit, as shown in Figure 3e. In contrast, our method, which uses the objective (15), achieves strong results, as illustrated in Figure 3d.

Other methods, unfortunately, also struggle to handle this illustrative 2D task effectively, despite their success in large-scale problems. This discrepancy raises questions about the theoretical justification and general applicability of these methods, particularly in scenarios where simpler tasks reveal limitations not evident in more complex settings.

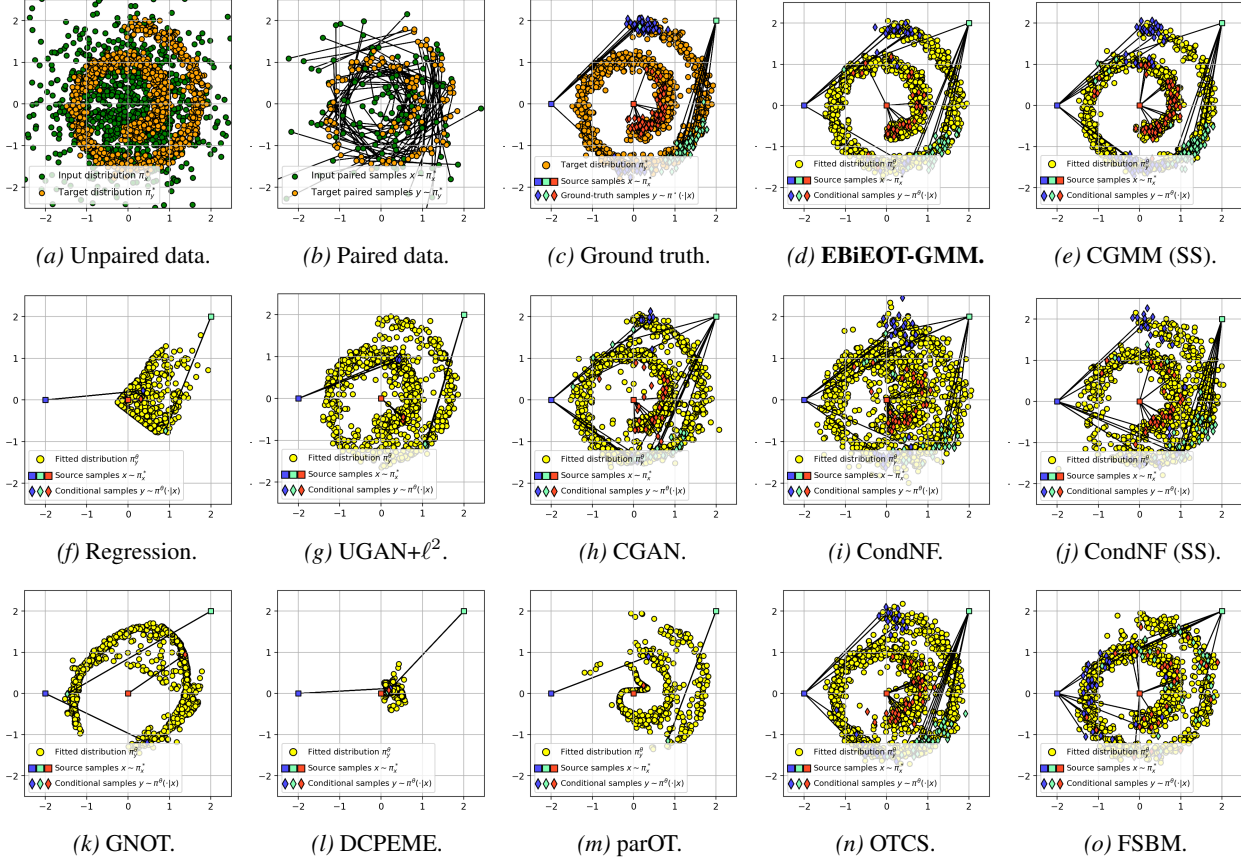


Figure 6. Comparison of the mapping learned by baselines on *Gaussian* $\rightarrow$ *Swiss Roll* task (§5.1). We use $P = 16k$ paired data, $Q = R = 16k$ unpaired data for training.

Method Order	$P = 128, Q = R = 1024$				$P = 16k, Q = R = 16k$			
	Unconditional		Conditional		Unconditional		Conditional	
	MMD $\downarrow$ [10^{-5}]	$\mathcal{W}_2 \downarrow$ [10^{-2}]	MMD $\downarrow$ [10^{-3}]	$\mathcal{W}_2 \downarrow$ [10^{-2}]	MMD $\downarrow$ [10^{-5}]	$\mathcal{W}_2 \downarrow$ [10^{-2}]	MMD $\downarrow$ [10^{-3}]	$\mathcal{W}_2 \downarrow$ [10^{-2}]
Regression	8.25 $\pm$ 5.39	3.77 $\pm$ 0.58	37.61 $\pm$ 0.00	89.76 $\pm$ 0.00	59.90 $\pm$ 4.85	33.40 $\pm$ 0.92	18.35 $\pm$ 0.00	48.42 $\pm$ 0.00
CGAN	666.68 $\pm$ 55.69	21.78 $\pm$ 1.31	41.72 $\pm$ 0.07	101.09 $\pm$ 0.09	15.65 $\pm$ 17.27	0.98 $\pm$ 0.46	22.48 $\pm$ 0.39	96.24 $\pm$ 1.03
UGAN+l ²	46.60 $\pm$ 18.81	1.93 $\pm$ 0.44	40.32 $\pm$ 0.00	94.35 $\pm$ 0.04	23.89 $\pm$ 13.99	2.70 $\pm$ 0.67	9.55 $\pm$ 0.01	30.20 $\pm$ 0.01
CondNF	39.76 $\pm$ 17.11	0.02 $\pm$ 3.00	26.56 $\pm$ 0.10	142.93 $\pm$ 0.04	30.83 $\pm$ 18.58	1.88 $\pm$ 0.48	20.59 $\pm$ 0.29	84.79 $\pm$ 0.61
CondNF (SS)	70.74 $\pm$ 8.70	3.84 $\pm$ 0.10	26.94 $\pm$ 0.22	71.09 $\pm$ 0.36	57.42 $\pm$ 2.80	0.94 $\pm$ 1.19	14.77 $\pm$ 0.16	69.69 $\pm$ 0.28
OTCS	102.52 $\pm$ 14.18	19.61 $\pm$ 0.86	39.38 $\pm$ 0.03	91.73 $\pm$ 0.06	6.05 $\pm$ 3.32	0.69 $\pm$ 0.09	9.13 $\pm$ 0.06	54.11 $\pm$ 0.25
FSBM	15.88 $\pm$ 7.89	0.93 $\pm$ 0.19	33.75 $\pm$ 0.23	126.06 $\pm$ 0.44	4.65 $\pm$ 5.49	0.52 $\pm$ 0.09	30.88 $\pm$ 0.16	126.17 $\pm$ 0.47
EBiEOT-GMM (Ours)	4.19 $\pm$ 0.84	0.43 $\pm$ 0.34	28.35 $\pm$ 0.40	100.59 $\pm$ 0.63	6.18 $\pm$ 3.55	1.00 $\pm$ 0.14	35.72 $\pm$ 0.27	109.16 $\pm$ 0.58

Table 5. Comparison of methods on the 128 and 16k datasets using MMD and $\mathcal{W}_2$ metrics in both unconditional and conditional settings. Results are reported as mean $\pm$ standard deviation over multiple runs. The best results among methods with overlapping confidence intervals are highlighted in bold.

C.1.2. ABLATION STUDY

In this section, we conduct an ablation study to address the question posed in §3.1 regarding how the number of source and target samples influences the quality of the learned mapping. The results, shown in Figure 7, indicate that the quantity of target points R has a greater impact than the number of source points Q (compare Figure 7c with Figure 7b). Additionally, it is evident that the inclusion of unpaired data helps mitigate over-fitting, as demonstrated in Figure 7a.

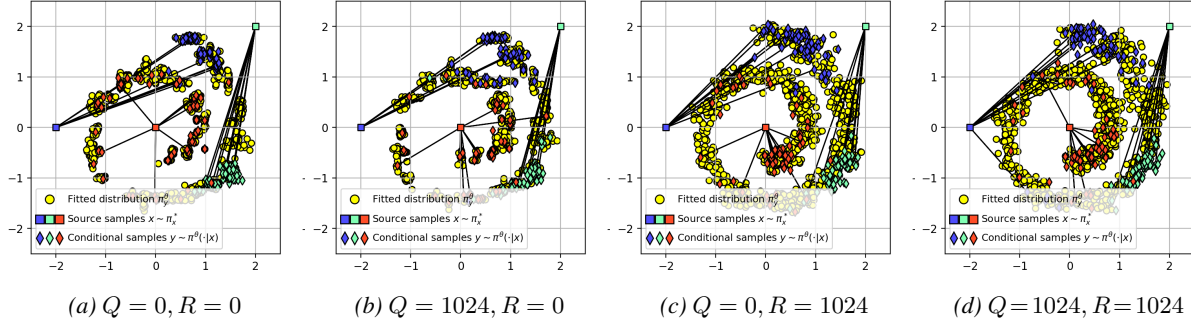


Figure 7. Ablation study analyzing the impact of varying source and target data point quantities on the learned mapping for the *Gaussian* $\rightarrow$ *Swiss Roll* task (using $P = 128$ paired samples).

C.1.3. ADDITIONAL ANALYTICAL CONDITIONAL DISTRIBUTION

Setup. To demonstrate the behavior of our method in scenarios where the ground-truth conditional plan is not induced by optimal transport (Flamary et al., 2021), we consider an experiment in which the conditional distribution $\pi^*(\cdot|x)$ is analytically known. We construct the target distribution by applying a deterministic mapping from the source distribution onto a Swiss-roll manifold, followed by point dependent noise. For each source point x , we first compute $r(x) = \tanh(\|x\|)$. This value is then mapped to the Swiss-roll parameter range: $t(x) = t_{\min} + (t_{\max} - t_{\min})r(x)$, where we set $t_{\min} = \frac{3}{2}\pi$ and $t_{\max} = \frac{9}{2}\pi$. The final target samples are generated using the Swiss-roll transformation $y = (t \cos t, t \sin t) + z_{SR}$, where $z_{SR} \sim \mathcal{N}(0, \sigma_{SR}^2 I)$ is isotropic Gaussian noise added in the target space. We use $\sigma_{SR} = 0.8$ in all experiments. This construction preserves the characteristic geometry of the Swiss-roll distribution while providing an analytically known conditional ground-truth transport plan $\pi^*(\cdot|x)$. We use the same data setup as in the main text (§5.1), with $P = 128$ paired samples and $Q = R = 1024$ unpaired samples.

Metrics. For evaluation, we computed MMD (Gretton et al., 2012) and Sinkhorn divergence (Feydy et al., 2019) with regularization parameter $\varepsilon = 0.001$, which serves as an approximation of $\mathcal{W}_2$; see Appendix D.2.

Discussion. The results for this conditional transport plan are reported in Table 6. All methods struggle to accurately reconstruct the true conditional mapping, although they achieve partial success in recovering the target marginal distribution.

We believe that the main difficulty arises from the structure of the mapping itself: all points with the same radius $\|x\|$ are collapsed onto a single point on the Swiss-roll segment. This many-to-one transformation makes reconstruction of the conditional relationship from paired samples particularly challenging.

C.1.4. SENSITIVITY ANALYSIS

Setup. We performed an ablation study to investigate the sensitivity of EBiEOT-GMM to the number of Gaussian components M and N . Specifically, we evaluated all combinations on the grid $M, N \in \{8, 16, 32, 64\}$, including the symmetric setting $M = N$. The ground-truth conditional mapping was constructed using the same procedure as in §5.1, while varying the numbers of source and target Gaussian components according to the selected values of M and N .

Metrics. For evaluation, we computed MMD (Gretton et al., 2012) and Sinkhorn divergence (Feydy et al., 2019) with regularization parameter $\varepsilon = 0.001$, which serves as an approximation of $\mathcal{W}_2$; see Appendix D.2.

Discussion. The results are presented in Figure 8. Increasing the number of Gaussian components improves the unconditional metrics while degrading the conditional ones. This behavior is expected: a larger number of components increases the flexibility of the model and may lead to overfitting of the conditional distribution. At the same time, the EBiEOT-GMM parameterization remains computationally lightweight, making hyperparameter search relatively inexpensive. In practice, the number of components can be efficiently selected using automated optimization frameworks such as Optuna (Akiba

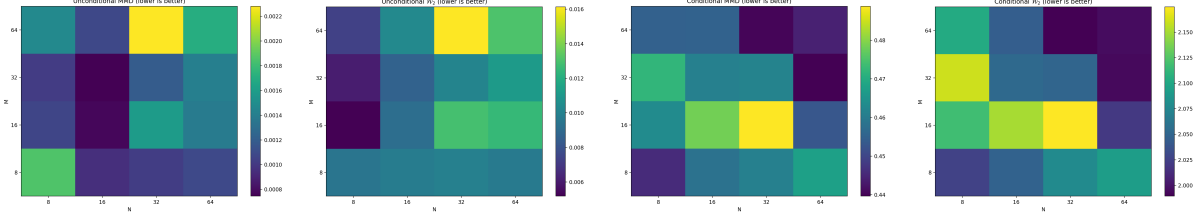


Figure 8. Results for sensitivity analysis described in Appendix C.1.4 for experiment §5.1.

et al., 2019).

C.2. Semi-supervised Classification

To demonstrate that our framework extends beyond domain translation, we apply the proposed objective (15) to a semi-supervised classification problem, following the discrete formulation sketch outlined in Appendix B.2. The goal is to learn the conditional distribution $\pi^*(y|x)$ from a small set of labeled examples together with additional unlabeled inputs.

In the classification setting, the target variable y takes values in a finite set of classes $\{y_1, \dots, y_K\}$. Using the Gibbs-Boltzmann parameterization (12) together with the energy decomposition (14), we obtain the discrete conditional model

$$\pi^\theta(y_k|x) = \frac{1}{Z^\theta(x)} \exp\left(\frac{f_k^\theta - c^\theta(x, y_k)}{\varepsilon}\right), \quad \log Z^\theta(x) = \log \sum_{k=1}^K \exp\left(\frac{f_k^\theta - c^\theta(x, y_k)}{\varepsilon}\right), \quad (34)$$

Here, $f^\theta \in \mathbb{R}^K$ is a learnable vector of class potentials, and $c^\theta(x, y_k)$ is a neural-network-based cost function. Substituting this parameterization into the general objective (15), the continuous integral over $\mathcal{Y}$ becomes a finite sum over classes:

$$\mathcal{L}(\theta) = \frac{1}{\varepsilon} \mathbb{E}_{(x,y) \sim \pi^*} [c^\theta(x, y)] - \frac{1}{\varepsilon} \mathbb{E}_{y \sim \pi_y^*} [f^\theta(y)] + \mathbb{E}_{x \sim \pi_x^*} [\log Z^\theta(x)]. \quad (35)$$

When the class prior is uniform, i.e. $\pi_y^*(y_k) = 1/K$, the second term simplifies to $-\frac{1}{K\varepsilon} \sum_{k=1}^K f_k^\theta$.

Setup. We conducted experiments on the MNIST dataset (Bottou et al., 1994), which contains ten classes corresponding to digits $0, \dots, 9$. We performed an ablation study by varying the number of labeled samples P and unlabeled samples Q . To obtain a balanced target marginal distribution π_y^* , we used the same number of labeled samples from each class.

Parameterization. We parameterize the joint cost $c(x, y)$ using embedding-based MLP and CNN architectures that evaluate all classes in a single forward pass. Each label y is represented by a learned embedding, which is combined with features of x to predict $c(x, y)$. The MLP uses flattened images and two 256-unit hidden layers, while the CNN uses two convolutional blocks followed by a 128-dimensional projection. In both cases, energies for all classes are computed

Method Order	Unconditional		Conditional	
	MMD ↓ [10 ⁻³]	$\mathcal{W}_2$ ↓ [10 ⁻²]	MMD ↓ [10 ⁻³]	$\mathcal{W}_2$ ↓ [10 ⁻¹]
Regression	18.35±0.00	48.42±0.00	88.91±5.80	24.94±0.86
CGAN	22.48±0.39	96.24±1.03	91.31±8.30	25.55±1.56
UGAN+ ℓ^2	9.55±0.01	30.20±0.01	91.96±8.25	25.76±1.58
CondNF	20.59±0.29	84.79±0.61	85.34±5.65	23.99±1.05
CondNF (SS)	14.77±0.16	69.69±0.28	84.94±5.22	23.87±0.88
OTCS	9.13±0.06	54.11±0.25	41.70 ±1.52	19.57±0.82
FSBM	30.88±0.16	126.17±0.47	43.63±2.91	17.68 ±0.71
EBiEOT-GMM (Ours)	0.03 ±0.04	0.36 ±0.16	68.84±1.59	22.08±1.01

Table 6. Comparison of methods on the new 128 dataset using MMD and $\mathcal{W}_2$ metrics for unconditional and conditional settings for setup described in Appendix C.1.3.

in parallel. Training uses the joint energies, a learnable class potential $f(y)$, and a log-sum-exp marginal over labels; predictions follow $p(y | x) \propto \exp((f(y) - c(x, y))/\varepsilon)$.

Metrics. We report classification accuracy. For each image x , we compute the Gibbs posterior over all classes, $p(y | x) \propto \exp((f(y) - c(x, y))/\varepsilon)$, and predict the label with the highest probability: $\hat{y} = \arg \max_y p(y | x)$. A prediction is counted as correct if $\hat{y}$ matches the ground-truth label. Accuracy is then computed as the fraction of correctly classified test images.

Discussion. The results are presented in Figure 9. The heatmap on test set show that adding unlabeled data generally improves accuracy, especially in the low-label regime ($P < 100$). This trend is consistent with the findings already reported in Table 1. At the same time, very large Q can hurt final performance, suggesting overfitting or reduced stability in that regime. We also observe that $P = 200$ paired samples is already sufficient to learn a strong classifier, while the test heatmaps indicate that some reduction in paired data can be compensated by increasing the amount of unlabeled data.

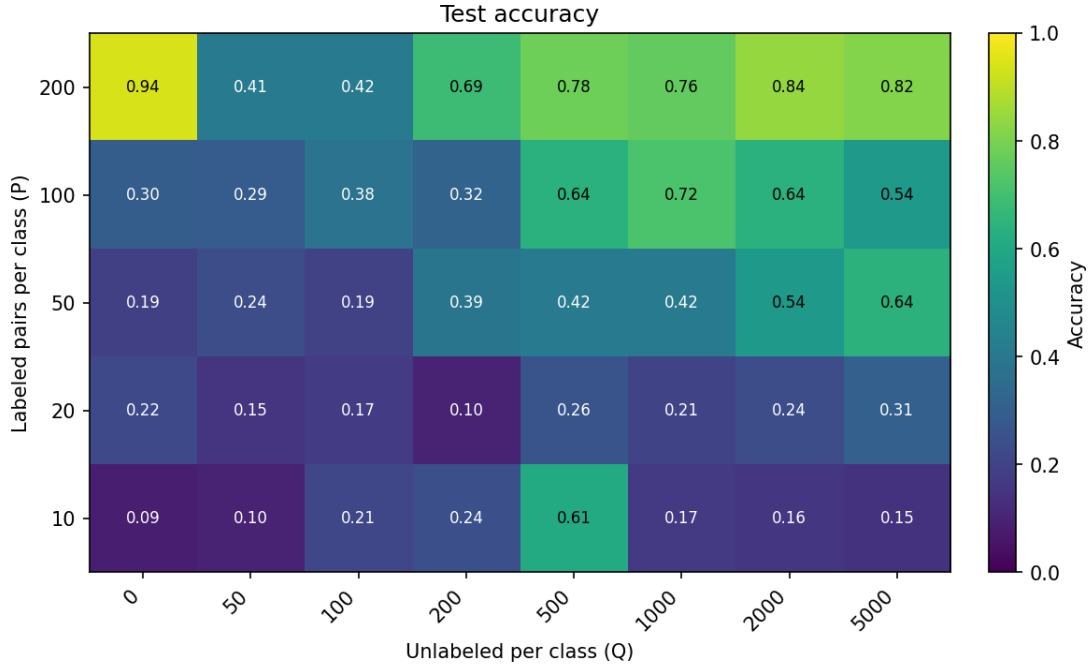


Figure 9. Heatmap for ablation of EBiEOT method for classification task described in Appendix C.2 for MNIST dataset.

D. Experimental Details

D.1. Baseline Details

This section details the loss functions employed by the baseline models, providing context and explanation for the data usage summarized in Table 7. Furthermore, it explains a straightforward adaptation of the log-likelihood loss function presented in (11) to accommodate unpaired data, offering a natural comparative approach to the method proposed in our work. Finally, it includes details about our reproduction of other methods and their discussion.

1. Standard generative & predictive models:

- **Regression Model** (MLP) uses the following simple ℓ^2 loss

$$\min_{\theta} \mathbb{E}_{(x,y) \sim \pi^*} \|y - G_{\theta}(x)\|^2,$$

where $G_{\theta} : \mathcal{X} \rightarrow \mathcal{Y}$ is a generator MLP with trainable parameters θ . Clearly, such a model can use only paired data. Furthermore, it is known that the optimal regressor G^* coincides with $\mathbb{E}_{y \sim \pi^*}(\cdot | x)y$, i.e., predicts the conditional expectation. Therefore, such a model will never learn the true data distribution unless all $\pi^*(\cdot | x)$ are degenerate.

- **Conditional GAN** uses the following min max loss:

$$\min_{\theta} \max_{\phi} \left[\underbrace{\mathbb{E}_{x,y \sim \pi^*} \log(D_{\phi}(y|x))}_{\text{Joint, requires pairs } (x,y) \sim \pi^*} + \underbrace{\mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{z \sim p_z} \log(1 - D_{\phi}(G_{\theta}(z|x)|x))}_{\text{Marginal, requires } x \sim \pi_x^*} \right],$$

where $G_{\theta} : \mathcal{Z} \times \mathcal{X} \rightarrow \mathcal{Y}$ is the conditional generator with parameters θ , p_z is a distribution on latent space $\mathcal{Z}$, and $D_{\phi} : \mathcal{Y} \times \mathcal{X} \rightarrow (0, 1)$ is the conditional discriminator with parameters ϕ . It is clear that the model can use not only paired data during the training, but also samples from π_x^* . The minimum of this loss is achieved when $G_{\theta}(\cdot|x)$ generates $\pi^*(\cdot|x)$ from p_z .

- **Unconditional GAN + ℓ^2 loss** optimizes the following min max objective:

$$\min_{\theta} \max_{\phi} \left[\lambda \underbrace{\mathbb{E}_{(x,y) \sim \pi^*} \mathbb{E}_{z \sim p_z} \|y - G_{\theta}(x,z)\|^2}_{\text{Joint, requires pairs } (x,y) \sim \pi^*} + \underbrace{\mathbb{E}_{y \sim \pi_y^*} \log(D_{\phi}(y))}_{\text{Marginal, requires } y \sim \pi_y^*} + \underbrace{\mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{z \sim p_z} \log(1 - D_{\phi}(G_{\theta}(x,z)))}_{\text{Marginal, requires } x \sim \pi_x^*} \right],$$

where $\lambda > 0$ is a hyperparameter and $G_{\theta} : \mathcal{X} \times \mathcal{Z} \rightarrow \mathcal{Y}$ is the stochastic generator. Compared to the unconditional case, the main idea here is to use the unconditional discriminator $D_{\phi} : \mathcal{Y} \rightarrow (0, 1)$. This allows using unpaired samples from π_y^* . However, using only GAN loss would ignore the paired information in any form, this is why the supervised ℓ^2 loss is added ($\lambda = 1$).

We note that this model has a trade-off between the target matching loss (GAN loss) and regression loss (which suffers from averaging). Hence, the model is unlikely to learn the true paired data distribution and can be considered as a heuristic loss for using both paired and unpaired data. Overall, we believe this baseline is representative of many existing GAN-based solutions (Tripathy et al., 2019, §3.5), (Jin et al., 2019, §3.3), (Yang & Chen, 2020, §C), (Vasluianu et al., 2021, §3), which use objectives that are *ideologically* similar to this one for paired and unpaired data.

- **Conditional Normalizing Flow** (Winkler et al., 2019) learns an explicit density model

$$\pi^{\theta}(y|x) = p_z(G_{\theta}^{-1}(y|x)) \left| \frac{\partial G_{\theta}^{-1}(y|x)}{\partial y} \right|$$

via optimizing log-likelihood (11) of the paired data. Here $G_{\theta} : \mathcal{Z} \times \mathcal{X} \rightarrow \mathcal{Y}$ is the conditional generator function. It is assumed that $\mathcal{Z} = \mathcal{Y}$ and $G_{\theta}(\cdot|x)$ is invertible and differentiable. In the implementation, we use the well-celebrated RealNVP neural architecture (Dinh et al., 2017). The optimal values are attained when the generator $G_{\theta}(\cdot|x)$ indeed generates $\pi^{\theta}(\cdot|x) = \pi^*(\cdot|x)$.

The conditional flow is expected to accurately capture the true conditional distributions, provided that the neural architecture is sufficiently expressive and there is an adequate amount of paired data available. However, as mentioned in §3.1, a significant challenge arises in integrating unpaired data into the learning process. For instance, approaches such as those proposed by (Atanov et al., 2019; Izmailov et al., 2020) aim to extend normalizing flows to a semi-supervised context. However, these methods primarily assume that the input conditions x are discrete, making it difficult to directly apply their frameworks to our continuous case. For completeness, below we discuss a variant of the log-likelihood loss (Atanov et al., 2019, Eq. 1) when both x, y are continuous.

2. Semi-supervised log-likelihood methods (Atanov et al., 2019; Izmailov et al., 2020):

- **Semi-supervised Conditional Normalizing Flows.** As noted by the authors, a natural strategy for log-likelihood semi-supervised training that leverages both paired and unpaired data is to optimize the following loss:

$$\max_{\theta} \left[\underbrace{\mathbb{E}_{(x,y) \sim \pi^*} \log \pi^{\theta}(y|x)}_{\text{Joint, requires pairs } (x,y) \sim \pi^*} + \underbrace{\mathbb{E}_{y \sim \pi_y^*} \log \pi^{\theta}(y)}_{\text{Marginal, requires } y \sim \pi_y^*} \right]. \quad (36)$$

This straightforward approach involves adding the unpaired data component, $\mathbb{E}_{y \sim \pi_y^*} \log \pi^{\theta}(y)$ to the loss function alongside the standard paired data component (11). While loss (36) looks natural, its optimization is *highly non-trivial* since the marginal log-likelihood $\log \pi^{\theta}(y)$ is not directly available. In fact, (Atanov et al., 2019; Izmailov et al., 2020) use this loss exclusively in the case when x is a discrete object, e.g., the class label $x \in \{1, 2, \dots, K\}$. In this case $\log \pi^{\theta}(y)$ can be analytically computed:

$$\log \pi^{\theta}(y) = \log \mathbb{E}_{x \sim \pi_x^*} \pi^{\theta}(y|x) = \log \sum_{k=1}^K \pi^{\theta}(y|x=k) \pi_x^*(x=k),$$

and $\pi^*(x = k)$ are known class probabilities. Unfortunately, in the continuous case $\pi_x^*(x)$ is typically not available explicitly, and one has to exploit *approximations* such as

$$\log \pi^\theta(y) = \log \mathbb{E}_{x \sim \pi_x^*} \pi^\theta(y|x) \approx \log \frac{1}{Q} \sum_{q=1}^Q \pi^\theta(y|x_q),$$

where x_q are train (unpaired) samples. However, such Monte-Carlo estimates are generally **biased** (because of the logarithm) and do not lead to good results, especially in high dimensions. Nevertheless, for completeness, we also test how this approach performs. In our 2D example (Figure 3j), we found there is no significant difference between this loss and the fully supervised loss (11): both models incorrectly map to the target and fail to learn conditional distributions.

- **Semi-supervised Conditional Gaussian Mixture Model.** Using the natural loss (36) for semi-supervised learning, one could also consider a (conditional) Gaussian mixture parameterization for $\pi^\theta(y|x)$ instead of the normalizing flow. For completeness, we include this baseline for comparison. Using the same Gaussian mixture parameterization (19) as in our method, we observed that this loss quickly overfits and leads to degenerate solutions, see Figure 3e.

3. **Semi-supervised Methods.** These methods are designed to learn deterministic OT maps with general cost functions and, as a result, cannot capture stochastic conditional distributions.

- **Neural optimal transport with pair-guided cost functional** (Asadulaev et al., 2024, GNOT). This method employs a general cost function for the neural optimal transport approach, utilizing a neural network parameterization for the mapping function and potentials. In our experiments, we focus on the paired cost function setup, enabling the use of both paired and unpaired data. We use the publicly available implementation³, which has been verified through toy experiments provided in the repository.
- **Differentiable cost-parameterized entropic mapping estimator** (Howard et al., 2024, DCPME). We obtained the implementation from the authors but were unable to achieve satisfactory performance. This is likely due to the deterministic map produced by their method based on the entropic map estimator from (Cuturi et al., 2023). In particular, scenarios where nearby or identical points are mapped to distant locations may introduce difficulties, potentially leading to optimization stagnation during training.
- **Parametric Pushforward Estimation With Map Constraints** (Panda et al., 2023, parOT)⁴. We evaluated this method using the ℓ_2 cost function, where it performed as expected. However, on our setup, the method proved unsuitable because it learns a fully deterministic transport map, which lacks the flexibility needed to model stochastic multimodal mapping. This limitation is visually evident in Figure 6m.
- **Optimal Transport-guided Conditional Score-based diffusion model** (Gu et al., 2023, OTCS). We evaluated this method on a two-dimensional example from their GitHub repository⁵, where it performed as expected. However, when applied to our setup (described in §5.1), the method failed to yield satisfactory results, even when provided with a large amount of training data (refer to Figure 6n and detailed in Appendix C.1.1).
- **Feedback Schrödinger Bridge Matching** (Theodoropoulos et al., 2024, FSBM). We first tested the method on a two-dimensional example from their GitHub repository⁶, where it performed as reported in the original paper. However, as shown in Figure 3o, the learned target distribution is very noisy with a small amount of data. With more samples (Figure 6o), the method approximates the target distribution better but still fails to capture the ground-truth conditional distribution, presumably due to misleading guidance from the key-points.

D.2. Metric Computation Details

In this section, we describe the evaluation metrics used in our experiments.

Sinkhorn divergence. The *Wasserstein-2 distance* is defined as

$$\mathcal{W}_2(\alpha, \beta) \stackrel{\text{def}}{=} \left(\min_{\pi \in \Pi(\alpha, \beta)} \mathbb{E}_{x, y \sim \pi} \|x - y\|_2^2 \right)^{\frac{1}{2}}, \quad (\mathcal{W}_2)$$

³ <https://github.com/machinestain/GNOT>

⁴ <https://github.com/natalieklein229/uq4ml/tree/parot>

⁵ <https://github.com/XJTU-XGU/OTCS/>

⁶ <https://github.com/panostheo98/FSBM>

i.e., the square root of the OT problem (OT) with quadratic cost $c^*(x, y) = \|x - y\|_2^2$. Computing OT exactly is computationally expensive: the complexity of solving the OT problem is $O(s^3 \log s)$ for sample size s (Cuturi, 2013). Moreover, OT suffers from the curse of dimensionality. In particular, for empirical measures

$$\hat{\alpha}_s = \frac{1}{s} \sum_{i=1}^s \delta_{X_i}, \quad \hat{\beta}_s = \frac{1}{s} \sum_{i=1}^s \delta_{Y_i},$$

the estimation error scales as (Weed & Bach, 2019)

$$\mathbb{E} \left| \text{OT}_{c^*}(\alpha, \beta) - \text{OT}_{c^*}(\hat{\alpha}_s, \hat{\beta}_s) \right| = O(s^{-1/D}), \quad (37)$$

where D is the data dimension. To address these issues, in practice it's common to use entropic regularized OT (1), discussed in detail in Appendix B.1. This formulation reduces the computational complexity to $O(s^2)$ (up to polylogarithmic terms) (Altschuler et al., 2017; Dvurechensky et al., 2018) and achieves improved sample complexity (Niles-Weed & Rigollet, 2022; Groppe & Hundrieser, 2024):

$$\mathbb{E} \left| \text{EOT}_{c^*, \varepsilon}(\alpha, \beta) - \text{EOT}_{c^*, \varepsilon}(\hat{\alpha}_s, \hat{\beta}_s) \right| \lesssim \frac{1}{\sqrt{s}}, \quad (38)$$

where $\lesssim$ denotes inequality up to a constant independent of s . However, entropic OT is biased, since $\text{EOT}_{c^*, \varepsilon}(\alpha, \alpha) \neq 0$. To remove this bias, Feydy et al. (2019) introduced the *Sinkhorn divergence*:

$$SD_{c^*, \varepsilon}(\alpha, \beta) \stackrel{\text{def}}{=} \text{EOT}_{c^*, \varepsilon}(\alpha, \beta) - \frac{1}{2} \text{EOT}_{c^*, \varepsilon}(\alpha, \alpha) - \frac{1}{2} \text{EOT}_{c^*, \varepsilon}(\beta, \beta). \quad (39)$$

Moreover, as $\varepsilon \rightarrow 0$, the Sinkhorn divergence converges to the OT distance (Feydy et al., 2019). In our experiments, we approximate $\mathcal{W}_2$ using Sinkhorn divergence with $\varepsilon = 0.001$, computed with the `GeomLoss` library (Feydy et al., 2019).

MMD. We compute the *Maximum Mean Discrepancy* (MMD) (Gretton et al., 2012) using its kernel formulation. Let $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ be a positive-definite kernel with associated reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ and feature map ϕ . The squared MMD between distributions ρ_k and $\hat{\rho}_k$ is defined as

$$\text{MMD}_K^2(\rho_k, \hat{\rho}_k) = \mathbb{E}_{x, x' \sim \rho_k} K(x, x') - 2\mathbb{E}_{x \sim \rho_k, y \sim \hat{\rho}_k} K(x, y) + \mathbb{E}_{y, y' \sim \hat{\rho}_k} K(y, y'). \quad (40)$$

Importantly, MMD can be computed directly through kernel evaluations without explicitly constructing feature maps. Given samples $\{x_i\}_{i=1}^N \sim \rho_k$ and $\{y_j\}_{j=1}^M \sim \hat{\rho}_k$, we use the unbiased estimator

$$\widehat{\text{MMD}}_K^2 = \frac{1}{N(N-1)} \sum_{i \neq i'} K(x_i, x_{i'}) - \frac{2}{NM} \sum_{i=1}^N \sum_{j=1}^M K(x_i, y_j) + \frac{1}{M(M-1)} \sum_{j \neq j'} K(y_j, y_{j'}). \quad (41)$$

The estimator has quadratic complexity in the number of samples. In all experiments, we use a uniform mixture of five Gaussian (RBF) kernels,

$$K(x, y) = \frac{1}{5} \sum_{r=1}^5 \exp\left(-\frac{\|x - y\|^2}{2\sigma_r^2}\right). \quad (42)$$

The bandwidths $\{\sigma_r\}_{r=1}^5$ are chosen on a geometric grid with ratio 2. First, we compute the median pairwise Euclidean distance over the pooled test samples $X_{\text{test}} \cup Y_{\text{test}}$ (the median heuristic). This value defines the central bandwidth. The remaining bandwidths are obtained by scaling it by factors $\{2^{-2}, 2^{-1}, 1, 2, 2^2\}$.

Estimation Protocol. As discussed above, empirical Wasserstein distances have poor sample complexity and therefore typically require a large number of samples for accurate estimation. However, in the two-dimensional setting this issue is less pronounced. We report both unconditional and conditional metrics. The unconditional metric evaluates how accurately a method reconstructs the target marginal distribution. For this evaluation, we use 1024 generated samples. The conditional metric evaluates how accurately the conditional transport plan is recovered. In this setting, we use 30 initial points and 1024 corresponding target samples.

D.3. General Implementation Details

Parameterization. The depth and number of hidden layers vary depending on the experiment.

For f^θ (18) we represent:

- w_n as $\log w_n$,
- b_n directly as a vector,
- the matrix B_n in diagonal form, with $\log(B_n)_{i,i}$ on its diagonal. This choice not only reduces the number of learnable parameters in θ_f but also enables efficient computation of B_n^{-1} with a time complexity of $\mathcal{O}(D_y)$.

For c^θ (17), we represent:

- $v_m(x)$ as a multilayer perceptron (MLP) with ReLU activations (Agarap, 2018) and a LogSoftMax output layer,
- $a_m(x)$ as an MLP with ReLU activations.

Optimizers. We employ two separate Adam optimizers (Kingma, 2014) with different step sizes for paired and unpaired data to enhance convergence.

Initialization.

- $\log w_n$ as $\log \frac{1}{n}$,
- b_n using random samples from π_y^* ,
- $\log(B_n)_{j,j}$ with $\log(0.1)$,
- for the neural networks, we use the default PyTorch initialization (Ansel et al., 2024),
- $\varepsilon = 1$ for all experiments, since the solver is independent of ε , as discussed in §2.2.

D.4. Gaussian to Swiss Roll Mapping Details

Generation process. To create the ground truth plan π^* , we utilize the following procedure: sample a mini-batch of size 64 and then determine the optimal mapping using the entropic Sinkhorn algorithm, as outlined in (Cuturi, 2013) and implemented in (Flamary et al., 2021). This process is repeated P times to generate the required number of pairs.

Cost Matrix. Let $x \in \mathbb{R}^2$ and $y \in \mathbb{R}^2$ be points from the source and target distributions, respectively. Define the rotated vectors as

$$y^{\pm\varphi} = R_{\pm\varphi}(y) = \begin{bmatrix} \cos(\pm\varphi) & -\sin(\pm\varphi) \\ \sin(\pm\varphi) & \cos(\pm\varphi) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix},$$

where φ is a given rotation angle, in our case, it's $\pm 90^\circ$. The corresponding elements of mini-batch OT cost matrices are then

$$C_{ij}^{+\varphi} = \|x_i - y_j^{+\varphi}\|_2, \quad C_{ij}^{-\varphi} = \|x_i - y_j^{-\varphi}\|_2,$$

and the final cost matrix is

$$C_{ij} = \min(C_{ij}^{+\varphi}, C_{ij}^{-\varphi}), \quad \forall i, j.$$

In other words, each $x_i \sim \pi_x^*$ is mapped to a point y_j on the opposite side of the Swiss Roll, rotated either by $+90^\circ$ or -90° , depending on which distance is smaller.

Implementation Details. We choose the parameters as follows: $N = 50$, $M = 25$, with learning rates $lr_{\text{paired}} = 3 \times 10^{-4}$ and $lr_{\text{unpaired}} = 0.001$. We utilize a two-layer MLP network for the function $a_m(x)$ and a single-layer MLP for $v_m(x)$. The experiments are executed in parallel on a 2080 Ti GPU for a total of 25,000 iterations, taking approximately 20 minutes to complete.

Method	Paired $(x, y) \sim \pi^*$	Unpaired $x \sim \pi_x^*$	Unpaired $y \sim \pi_y^*$
Regression	✓	✗	✗
UGAN + ℓ^2	✓	✓	✓
CGAN	✓	✓	✗
CondNF	✓	✗	✗
CondNF (SS)	✓	✓	✓
GNOT	✓	✓	✓
DCPEME	✓	✓	✓
parOT	✓	✓	✓
OTCS	✓	✓	✓
FSBM	✓	✓	✓
CGMM (SS)	✓	✓	✓
Our method	✓	✓	✓

Table 7. The ability to use paired/unpaired data by various models.

D.5. Image Translation via ALAE Details

Implementation Details. We largely follow the setup in Appendix D.4, setting $N = 10$, $M = 1$, and using 10K optimization steps. Our method employs a single-layer MLP to predict the parameters of a mixture of 10 Gaussians.

D.6. Weather Prediction Details

We select two distinct months from the dataset (Malinin et al., 2021; Rubachev et al., 2025) and translate the meteorological features from the source month (January) to the target month (June). To operate at the monthly scale, we represent a source data point $x \in \mathbb{R}^{188}$ as the mean and standard deviation of the features collected at a specific location over the source month. The targets $y \in \mathbb{R}^{94}$ correspond to individual measurements in the target month.

Pairs are constructed by aligning a source data point with the target measurements at the same location. Consequently, multiple target data points y may correspond to a single source point x and represent samples from conditional distributions $\pi^*(y|x)$. The measurements from non-aligned locations are treated as unpaired. Such unpaired data naturally arise because stations may not provide reliable measurements in both months, for example, due to maintenance, sensor failures, extreme weather, or connectivity issues.

We obtain 500 unpaired and 192 paired data samples. For testing, 100 pairs are randomly selected.

Implementation Details. In general, we consider the same setting as in D.4. Specifically, we set $N = 10$, $M = 1$ and the number of optimization steps to 30,000. The baseline uses an MLP network with the same number of parameters, predicting the parameters of a mixture of 10 Gaussians.

Extremely Low-Data Regimes Discussion. As is clear from Table 1, our method diverges when trained on very few samples (e.g., 5 paired and no unpaired). This is not surprising given the high dimensionality of the data ($D = 94$) and the number of learnable parameters ($|\theta| = 2668$). In such low-data regimes, the model likely overfits the cost function c^θ to the small paired dataset, which can cause instability. This issue could potentially be alleviated by simplifying the model, for instance by using a shallow or even linear parameterization of c^θ (Andrade et al., 2025). However, for consistency and fairness, we kept the architecture fixed across all experiments in the table.

E. Proofs

E.1. Loss Derivation

Below, we present a step-by-step derivation of the mathematical transitions, allowing the reader to follow and verify the validity of our approach. We denote as C_1, C_2 all terms that are not involved in learning the conditional plan $\pi^\theta(y|x)$, i.e., not dependent on θ or marginal distributions such as π_x^* . Starting from (6), we deduce

$$\begin{aligned}
 \text{KL}(\pi^* \|\pi^\theta) &= \mathbb{E}_{x,y \sim \pi^*} \log \frac{\pi_x^*(x) \pi^*(y|x)}{\pi_x^\theta(x) \pi^\theta(y|x)} = \mathbb{E}_{x \sim \pi_x^*} \log \frac{\pi_x^*(x)}{\pi_x^\theta(x)} + \mathbb{E}_{x,y \sim \pi^*} \log \frac{\pi^*(y|x)}{\pi^\theta(y|x)} = \\
 \text{KL}(\pi_x^* \|\pi_x^\theta) &+ \mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^*(\cdot|x)} \log \frac{\pi^*(y|x)}{\pi^\theta(y|x)} = \underbrace{\text{KL}(\pi_x^* \|\pi_x^\theta)}_{\text{Marginal}} + \underbrace{\mathbb{E}_{x \sim \pi_x^*} \text{KL}(\pi^*(\cdot|x) \|\pi^\theta(\cdot|x))}_{\text{Conditional}} = \\
 C_1 &+ \mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^*(\cdot|x)} \log \frac{\pi^*(y|x)}{\pi^\theta(y|x)} = C + \mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^*(\cdot|x)} [\log \pi^*(y|x) - \log \pi^\theta(y|x)] = \\
 C_1 &- \mathbb{E}_{x \sim \pi_x^*} H(\pi^*(\cdot|x)) - \mathbb{E}_{x,y \sim \pi^*} \log \pi^\theta(y|x) = C_2 - \mathbb{E}_{x,y \sim \pi^*} \log \pi^\theta(y|x) \stackrel{(12)}{=} \\
 C_2 &- \mathbb{E}_{x,y \sim \pi^*} \log \frac{\exp(-E^\theta(y|x))}{Z^\theta(x)} = C_2 + \mathbb{E}_{x,y \sim \pi^*} E^\theta(y|x) + \mathbb{E}_{x,y \sim \pi^*} \log Z^\theta(x) \stackrel{(14)}{=} \\
 C_2 &+ \mathbb{E}_{x,y \sim \pi^*} \frac{c^\theta(x,y) - f^\theta(y)}{\varepsilon} + \mathbb{E}_{x,y \sim \pi^*} \log Z^\theta(x) = \\
 C_2 &+ \varepsilon^{-1} \mathbb{E}_{x,y \sim \pi^*} [c^\theta(x,y)] - \varepsilon^{-1} \mathbb{E}_{x,y \sim \pi^*} f^\theta(y) + \mathbb{E}_{x,y \sim \pi^*} \log Z^\theta(x) = \\
 C_2 &+ \varepsilon^{-1} \mathbb{E}_{x,y \sim \pi^*} [c^\theta(x,y)] - \varepsilon^{-1} \mathbb{E}_{y \sim \pi_y^*} \mathbb{E}_{x \sim \pi^*(\cdot|y)} f^\theta(y) + \mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^*(\cdot|x)} \log Z^\theta(x) = \\
 C_2 &+ \varepsilon^{-1} \mathbb{E}_{x,y \sim \pi^*} [c^\theta(x,y)] - \varepsilon^{-1} \mathbb{E}_{y \sim \pi_y^*} f^\theta(y) \underbrace{\mathbb{E}_{x \sim \pi^*(\cdot|y)} 1}_{=1} + \mathbb{E}_{x \sim \pi_x^*} \log Z^\theta(x) \underbrace{\mathbb{E}_{y \sim \pi^*(\cdot|x)} 1}_{=1} =
 \end{aligned}$$

$$C_2 + \varepsilon^{-1} \mathbb{E}_{x,y \sim \pi^*} [c^\theta(x, y)] - \varepsilon^{-1} \mathbb{E}_{y \sim \pi_y^*} f^\theta(y) + \mathbb{E}_{x \sim \pi_x^*} \log Z^\theta(x).$$

The mathematical derivation presented above demonstrates that our defined loss function (15) is essentially a framework for minimizing KL-divergence. In other words, when the loss (15) equals to $-C_2$, it implies that we have successfully recovered the true conditional plan π^* in the KL sense.

E.2. Expressions for the Gaussian Parameterization

Proof of Proposition 3.1. Our parameterization of the cost c^θ (17) and the dual potential f^θ (18) gives:

$$\begin{aligned} \exp\left(\frac{f^\theta(y) - c^\theta(x, y)}{\varepsilon}\right) &= \exp\left(\log \sum_{n=1}^N w_n \mathcal{N}(y | b_n, \varepsilon B_n) + \log \sum_{m=1}^M v_m(x) \exp\left(\frac{\langle a_m(x), y \rangle}{\varepsilon}\right)\right) \\ &= \sum_{m=1}^M \sum_{n=1}^N \frac{v_m(x) w_n}{\sqrt{\det(2\pi\varepsilon^{-1} B_n^{-1})}} \exp\left(-\frac{1}{2}(y - b_n)^\top \frac{B_n^{-1}}{\varepsilon} (y - b_n) + \frac{\langle a_m(x), y \rangle}{\varepsilon}\right) \end{aligned}$$

We now rewrite the expression inside the exponent, scaled by -2ε , using the symmetry of B_n , to cast it into a Gaussian mixture form:

$$\begin{aligned} (y - b_n)^\top B_n^{-1} (y - b_n) - 2\langle a_m(x), y \rangle &= y^\top B_n^{-1} y - 2b_n^\top B_n^{-1} y + b_n^\top B_n^{-1} b_n - 2\langle a_m(x), y \rangle = \\ &= y^\top B_n^{-1} y - 2 \underbrace{(b_n + B_n a_m(x))^\top B_n^{-1} y}_{\stackrel{\text{def}}{=} d_{mn}^\top(x)} + b_n^\top B_n^{-1} b_n = \\ &= (y - d_{mn}(x))^\top B_n^{-1} (y - d_{mn}(x)) + b_n^\top B_n^{-1} b_n - d_{mn}^\top(x) B_n^{-1} d_{mn}(x). \end{aligned}$$

Afterwards, we rewrite the last two terms:

$$\begin{aligned} b_n^\top B_n^{-1} b_n - d_{mn}^\top(x) B_n^{-1} d_{mn}(x) &= b_n^\top B_n^{-1} b_n - (b_n + B_n a_m(x))^\top B_n^{-1} (b_n + B_n a_m(x)) = \\ &= \underbrace{b_n^\top B_n^{-1} b_n - b_n^\top B_n^{-1} b_n}_{=0} - \underbrace{b_n^\top B_n^{-1} B_n a_m(x)}_{=I} - \underbrace{a_m^\top(x) B_n B_n^{-1} b_n}_{=I} - \underbrace{a_m^\top(x) B_n B_n^{-1} B_n a_m(x)}_{=I} = \\ &= -a_m^\top(x) B_n a_m(x) - 2b_n^\top a_m(x). \end{aligned}$$

Finally, we get

$$\begin{aligned} \exp\left(\frac{f^\theta(y) - c^\theta(x, y)}{\varepsilon}\right) &= \sum_{m=1}^M \sum_{n=1}^N w_n v_m(x) \exp\left(\frac{a_m^\top(x) B_n a_m(x) + 2b_n^\top a_m(x)}{2\varepsilon}\right) \\ &\quad \cdot \underbrace{\frac{1}{\sqrt{\det(2\pi\varepsilon^{-1} B_n^{-1})}} \exp\left(-\frac{1}{2}(y - d_{mn}(x))^\top \frac{B_n^{-1}}{\varepsilon} (y - d_{mn}(x))\right)}_{\stackrel{\text{def}}{=} z_{mn}(x)} \\ &\quad \underbrace{= \mathcal{N}(y | d_{mn}(x), \varepsilon B_n)} \end{aligned}$$

and, since $\int_{\mathcal{Y}} \mathcal{N}(y | d_{mn}(x), \varepsilon B_n) dy = 1$, the normalization constant simplifies to the sum of $z_{mn}(x)$:

$$\begin{aligned} Z^\theta(x) &= \int_{\mathcal{Y}} \exp\left(\frac{f^\theta(y) - c^\theta(x, y)}{\varepsilon}\right) dy \\ &= \int_{\mathcal{Y}} \sum_{m=1}^M \sum_{n=1}^N z_{mn}(x) \mathcal{N}(y | d_{mn}(x), \varepsilon B_n) dy = \sum_{m=1}^M \sum_{n=1}^N z_{mn}(x). \end{aligned}$$

□

Proof of Proposition 3.2. Combining equations (12), (14) and derivation above, we seamlessly obtain the expression (19) needed for Proposition 3.2. □

E.3. Gradient of our Loss for Energy-Based Modeling

Proof of Proposition A.1. Direct differentiation of (15) gives:

$$\frac{\partial}{\partial \theta} \mathcal{L}(\theta) = \varepsilon^{-1} \mathbb{E}_{x, y \sim \pi^*} \left[\frac{\partial}{\partial \theta} c^\theta(x, y) \right] - \varepsilon^{-1} \mathbb{E}_{y \sim \pi_y^*} \left[\frac{\partial}{\partial \theta} f^\theta(y) \right] + \mathbb{E}_{x \sim \pi_x^*} \left[\frac{\partial}{\partial \theta} \log Z^\theta(x) \right]. \quad (43)$$

Recalling expression for the normalization constant, the last term can be expressed as follows:

$$\begin{aligned} \mathbb{E}_{x \sim \pi_x^*} \left[\frac{1}{Z^\theta(x)} \frac{\partial}{\partial \theta} Z^\theta(x) \right] &= \mathbb{E}_{x \sim \pi_x^*} \left[\frac{1}{Z^\theta(x)} \int_{\mathcal{Y}} \frac{\partial}{\partial \theta} \exp \left(\frac{f^\theta(y) - c^\theta(x, y)}{\varepsilon} \right) dy \right] = \\ &= \mathbb{E}_{x \sim \pi_x^*} \left[\frac{1}{Z^\theta(x)} \int_{\mathcal{Y}} \frac{\frac{\partial}{\partial \theta} (f^\theta(y) - c^\theta(x, y))}{\varepsilon} \exp \left(\frac{f^\theta(y) - c^\theta(x, y)}{\varepsilon} \right) dy \right] = \\ &= \varepsilon^{-1} \mathbb{E}_{x \sim \pi_x^*} \left[\int_{\mathcal{Y}} \frac{\partial}{\partial \theta} (f^\theta(y) - c^\theta(x, y)) \underbrace{\left\{ \frac{1}{Z^\theta(x)} \exp \left(\frac{f^\theta(y) - c^\theta(x, y)}{\varepsilon} \right) \right\}}_{\pi^\theta(y|x)} dy \right]. \end{aligned}$$

From equation above we obtain:

$$\begin{aligned} \frac{\partial}{\partial \theta} \mathcal{L}(\theta) &= \varepsilon^{-1} \left\{ \mathbb{E}_{x, y \sim \pi^*} \left[\frac{\partial}{\partial \theta} c^\theta(x, y) \right] - \mathbb{E}_{y \sim \pi_y^*} \left[\frac{\partial}{\partial \theta} f^\theta(y) \right] \right. \\ &\quad \left. + \mathbb{E}_{x \sim \pi_x^*} \mathbb{E}_{y \sim \pi^\theta(y|x)} \left[\frac{\partial}{\partial \theta} (f^\theta(y) - c^\theta(x, y)) \right] \right\}, \end{aligned}$$

which concludes the proof. $\square$

E.4. Universal Approximation

Our objective is to set up and use the very general universal approximation result in (Acciaio et al., 2024, Theorem 3.8). Hereinafter, we use the following notation that slightly abuse notation from the main text.

Intra-Section Notation. For any $D \in \mathbb{N}$ we denote the Lebesgue measure on $\mathbb{R}^D$ by λ_D , suppressing the subscript D whenever clear from its context, we use $L_+^1(\mathbb{R}^D)$ to denote the set of Lebesgue integrable (equivalence class of) functions $f : \mathbb{R}^D \rightarrow \mathbb{R}$ for which $\int f(x) \lambda(dx) = 1$ and $f \geq 0$ λ -a.e; i.e. Lebesgue-densities of probability measures. We use $\mathcal{P}_1^+(\mathbb{R}^D)$ to denote the space of all Borel probability measures on $\mathbb{R}^D$ which are absolutely continuous with respect to λ , metrized by the total variation distance d_{TV} . For any $D \in \mathbb{N}$, we denote the set of $D \times D$ positive-definite matrices by PD_D . Additionally, for any $N \in \mathbb{N}$, we define the N -simplex by $\Delta_N \stackrel{\text{def.}}{=} \{u \in [0, 1]^N : \sum_{n=1}^N u_n = 1\}$. We also denote floor operation for any $x \in \mathbb{R}$ as $\lfloor x \rfloor \stackrel{\text{def.}}{=} \max\{n \in \mathbb{Z} | n \leq x\}$.

Lemma E.1 (The Space $(\mathcal{P}_1^+(\mathbb{R}^D), d_{TV})$ is quantizable by Gaussian Mixtures). *For every $N \in \mathbb{N}$, let $D_N \stackrel{\text{def.}}{=} \frac{N}{2}((D^2 + 3D + 2))$ and define the map*

$$\begin{aligned} GMM_N : \mathbb{R}^{D_N} &= \mathbb{R}^N \times \mathbb{R}^{ND} \times \mathbb{R}^{\frac{N}{2}D(D+1)} \rightarrow \mathcal{P}_1^+(\mathbb{R}^D) \\ (w, \{b_n\}_{n=1}^N, \{B_n\}_{n=1}^N) &\mapsto \sum_{n=1}^N \text{Proj}_{\Delta_N}(w)_n \nu(b_n, \varphi(B_n)), \end{aligned}$$

where $\text{Proj}_{\Delta_N} : \mathbb{R}^N \mapsto \Delta_N$ is the ℓ^2 orthogonal projection of $\mathbb{R}^N$ onto the N -simplex Δ_N and $\nu(b_n, \varphi(B_n))$ is the Gaussian measure on $\mathbb{R}^D$ with mean b_n , and non-singular covariance matrix given by $\varphi(B_n)$ where $\varphi : \mathbb{R}^{D(D+1)/2} \rightarrow \text{PD}_D$ is given for each $B \in \mathbb{R}^{D(D+1)/2}$ by

$$\varphi(B) \stackrel{\text{def.}}{=} \exp \left(\begin{pmatrix} B_1 & B_2 & \dots & B_D \\ B_2 & B_3 & \dots & B_{2D-1} \\ \vdots & \ddots & & \vdots \\ B_D & B_{2D-1} & \dots & B_{D(D+1)/2} \end{pmatrix} \right), \quad (44)$$

where $\exp$ is the matrix exponential on the space of $D \times D$ matrices. Then, the family $(GMM_n)_{n=1}^\infty$ is a quantization of $(P_1^+(\mathbb{R}^D), d_{TV})$ in the sense of (Acciaio et al., 2024, Definition 3.2).

Proof. As implied by (Arabpour et al., 2024, Equation (3.10) in Proposition 7) every Gaussian measure $\mathcal{N}(m, \Sigma) := \mu$ on $\mathbb{R}^D$ with mean $m \in \mathbb{R}^D$, symmetric positive-definite covariance matrix Σ can be represented as

$$\mu = \mathcal{N}(m, \varphi(X)) \quad (45)$$

for some (unique) vector $X \in \mathbb{R}^{D(D+1)/2}$. Therefore, by definition of a quantization, see (Acciaio et al., 2024, Definition 3.2), it suffices to show that the family of Gaussian mixtures is dense in $(\mathcal{P}_1^+(\mathbb{R}^D), d_{TV})$.

Now, let $\nu \in \mathcal{P}_1^+(\mathbb{R}^D)$ be arbitrary. By definition of $\mathcal{P}_1^+(\mathbb{R}^D)$ the measure ν admits a Radon-Nikodym derivative $f \stackrel{\text{def}}{=} \frac{D\nu}{D\lambda}$, with respect to the D -dimensional Lebesgue measure λ . Moreover, by the Radon-Nikodym theorem, $f \in L^1_\mu(\mathbb{R}^D)$; and by since μ is a probability measure then $\nu \in L^1_+(\mathbb{R}^D)$.

Since compactly-supported smooth functions are dense in $L^1_+(\mathbb{R}^D)$ then, for every $\varepsilon > 0$, there exists some $\tilde{f} \in C_c^\infty(\mathbb{R}^D)$ with $\tilde{f} \geq 0$ such that

$$\|f - \tilde{f}\|_{L^1(\mathbb{R}^D)} < \frac{\varepsilon}{3}. \quad (46)$$

Since $C_c^\infty(\mathbb{R}^D)$ is dense in $L^1(\mathbb{R}^D)$ then we may without loss of generality re-normalize $\tilde{f}$ to ensure that it integrates to 1.

Since $\tilde{f}$ is compactly supported and approximates f , then (if f is non-zero, which it cannot be as it integrates to 1) then it cannot be analytic, and thus it is non-polynomial. For every $\delta > 0$, let φ_δ denote the density of the D -dimensional Gaussian probability measure with mean 0 and isotropic covariance δI_D (where I_D is the $D \times D$ identity matrix). Therefore, the proof of (Pinkus, 1999, Proposition 3.7) (or any standard mollification argument) shows that we can pick $\delta \stackrel{\text{def}}{=} \delta(\varepsilon) > 0$ small enough so that the convolution $\tilde{f} \star \varphi_\delta$ satisfies

$$\|\tilde{f} - \tilde{f} \star \varphi_\delta\|_{L^1(\mathbb{R}^D)} < \frac{\varepsilon}{3}. \quad (47)$$

Note that $\tilde{f} \star \varphi_\delta$ is the density of probability measure on $\mathbb{R}^D$; namely, the law of a random variable which is the sum of a Gaussian random variance with law $\mathcal{N}(0, \delta I_N)$ and a random variable with law μ . That is, $\tilde{f} \star \varphi_\delta \lambda \in L^1_+(\mathbb{R}^D)$. Together (46) and (47) imply that

$$\|f - \tilde{f} \star \varphi_\delta\|_{L^1(\mathbb{R}^D)} < \frac{2\varepsilon}{3}. \quad (48)$$

Recall the definition of the convolution: for each $x \in \mathbb{R}^D$ we have

$$\tilde{f}(x) \star \varphi_\delta \stackrel{\text{def}}{=} \int_{u \in \mathbb{R}^D} \tilde{f}(u) \varphi_\delta(x - u) \lambda(du). \quad (49)$$

Since $\tilde{f}, \varphi_\delta \in C_c^\infty(\mathbb{R}^D)$ then Lebesgue integral of their product coincides with the Riemann integral of their product; whence, there is an $N \stackrel{\text{def}}{=} N(\varepsilon) \in \mathbb{N}$ “large enough” so that

$$\left\| \int_{u \in \mathbb{R}^D} \tilde{f}(u) \varphi_\delta(x - u) \lambda(du) - \sum_{n=1}^N \tilde{f}(u_n) \varphi_\delta(x - u_n) \lambda(du) \right\|_{L^1(\mathbb{R}^D)} < \frac{\varepsilon}{3} \quad (50)$$

for some $u_1, \dots, u_N \in \mathbb{R}^D$. Note that, $\sum_{n=1}^N \tilde{f}(u_n) \varphi_\delta(x - u_n)$ is the law of a Gaussian mixture. Therefore, combining (48) and (50) implies that

$$\left\| f - \sum_{n=1}^N \tilde{f}(u_n) \varphi_\delta(x - u_n) \lambda(du) \right\|_{L^1(\mathbb{R}^D)} < \varepsilon. \quad (51)$$

Finally, recalling that the total variation distance between two measures with integrable Lebesgue density equals the $L^1(\mathbb{R}^D)$ norm of the difference of their densities; yields the conclusion; i.e.

$$d_{TV}(\nu, \hat{\nu}) = \left\| f - \sum_{n=1}^N \tilde{f}(u_n) \varphi_\delta(x - u_n) \lambda(du) \right\|_{L^1(\mathbb{R}^D)} < \varepsilon$$

where $\frac{D\hat{\nu}}{D\lambda} \stackrel{\text{def}}{=} \sum_{n=1}^N \tilde{f}(u_n) \varphi_\delta(x - u_n) \lambda(du)$. □

Lemma E.2 (The space $(\mathcal{P}_1^+(\mathbb{R}^D), d_{TV})$ is Approximate Simplicial). *Let $\hat{\mathcal{Y}} \stackrel{\text{def.}}{=} \bigcup_{N \in \mathbb{N}} \Delta_N \times [\mathcal{P}_1^+(\mathbb{R}^D)]^N$ and define the map $\eta : \hat{\mathcal{Y}} \mapsto \mathcal{P}_1^+(\mathbb{R}^D)$ by*

$$\eta(w, (r_n)_{n=1}^N) \stackrel{\text{def.}}{=} \sum_{n=1}^N w_n r_n.$$

Then, η is a mixing function, in the sense of (Acciaio et al., 2024, Definition 3.1). Consequentially, $(\mathcal{P}_1^+(\mathbb{R}^D), \eta)$ is approximately simplicial.

Proof. Let $\mathcal{M}^+(\mathbb{R}^D)$ denote the Banach space of all finite signed measures on $\mathbb{R}^D$ with finite total variation norm $\|\cdot\|_{TV}$. Since $\|\cdot - \cdot\|_{TV} = d_{TV}$ when restricted to $\mathcal{P}_1^+(\mathbb{R}^D) \times \mathcal{P}_1^+(\mathbb{R}^D)$ and since $\|\cdot\|_{TV}$ is a norm, then the conclusion follows from (Acciaio et al., 2024, Example 5.1) and since $\mathcal{P}_1^+(\mathbb{R}^D)$ is a convex subset of $\mathcal{M}^+(\mathbb{R}^D)$. $\square$

Together, Lemmata E.1 and E.2 imply that $(\mathcal{P}_1^+(\mathbb{R}^D), d_{TV}, \eta, Q)$ is a QAS space in the sense of (Acciaio et al., 2024, Definition 3.4), where $Q \stackrel{\text{def.}}{=} (GMM_M)_{M \in \mathbb{N}}$. Consequently, the following is a geometric attention mechanism in the sense of (Acciaio et al., 2024, Definition 3.5)

$$\begin{aligned} \hat{\eta} : \bigcup_{N \in \mathbb{N}} \Delta_N \times \mathbb{R}^{N \times D_M} &\rightarrow \mathcal{P}_1^+(\mathbb{R}^D) \\ \left(w, (v_m, (b_{mn})_{n=1}^N, (B_{mn})_{n=1}^N)_{m=1}^M\right) &\mapsto \sum_{n=1}^N w_n \sum_{m=1}^M \text{Proj}_{\Delta_M}(v_m)_n \nu(b_{mn}, \varphi(B_{mn})). \end{aligned}$$

Before presenting our main theorem, we first introduce several definitions of activation functions that will be used in the theorem. These definitions, which are essential for completeness, are taken from (Acciaio et al., 2024, Definitions 2.2-2.4).

Definition E.3 (Trainable Activation Function: Singular-ReLU Type). A trainable activation function σ is of *ReLU+Step* type if

$$\sigma_\alpha : \mathbb{R} \ni x \mapsto \alpha_1 \max\{x, \alpha_2 x\} + (1 - \alpha_1)[x] \in \mathbb{R}$$

Definition E.4 (Trainable Activation Function: Smooth-ReLU Type). A trainable activation function σ is of *smooth non-polynomial* type if there is a non-polynomial $\sigma^* \in C_c^\infty(\mathbb{R})$, for which

$$\sigma_\alpha : \mathbb{R} \ni x \mapsto \alpha_1 \max\{x, \alpha_2 x\} + (1 - \alpha_1)\sigma^*(x) \in \mathbb{R}$$

Definition E.5 (Classical Activation Function). Let $\sigma^* \in C_c^\infty(\mathbb{R})$ be non-affine and such that there is some $x \in \mathbb{R}$ at which σ is differentiable and has non-zero derivative. Then σ is a classical regular activation function if, for every $\alpha \in \mathbb{R}^2$, $\sigma_\alpha = \sigma^*$.

Further in the text, we assume that activation functions are applied element-wise to each vector $x \in \mathbb{R}^D$. We are now ready to prove the first part of our approximation theorem.

Proposition E.6 (Deep Gaussian Mixtures are Universal Conditional Distributions in the TV Distance). *Let $\pi : (\mathbb{R}^D, \|\cdot\|_2) \rightarrow (\mathcal{P}_1^+(\mathbb{R}^D), d_{TV})$ be Hölder. Then, for every compact subset $K \subseteq \mathbb{R}^D$, every approximation error $\varepsilon > 0$ there exists $M, N \in \mathbb{N}$ and a MLP $\hat{f} : \mathbb{R}^D \mapsto \mathbb{R}^{N \times N D_M}$ with activations as in Definitions E.3, E.4, E.5 such that the (non-degenerate) Gaussian-mixture valued map*

$$\hat{\pi}(\cdot|x) \stackrel{\text{def.}}{=} \hat{\eta} \circ \hat{f}(x)$$

satisfies the uniform estimate

$$\max_{x \in K} d_{TV}(\hat{\pi}(\cdot|x) \| \pi(\cdot|x)) < \varepsilon.$$

Proof. Since Lemmata E.2 and E.1 imply that $(\mathcal{P}_1^+(\mathbb{R}^D), d_{TV}, \eta, Q)$ is a QAS space in the sense of (Acciaio et al., 2024, Definition 3.4), then the conclusion follows directly from (Acciaio et al., 2024, Theorem 3.8). $\square$

Since many of our results are formulated in the Kullback-Leibler divergence, then our desired guarantee is obtained only under some additional mild regularity requirements of the target conditional distribution $\hat{\pi}$ being approximated.

Assumption E.7 (Regularity of Conditional Distribution). Let $\pi : (\mathbb{R}^D, \|\cdot\|_2) \rightarrow (\mathcal{P}_1^+(\mathbb{R}^D), d_{TV})$ be Hölder and, for each $x \in \mathbb{R}^D$, $\pi(\cdot|x)$ is absolutely continuous with respect to the Lebesgue measure λ on $\mathbb{R}^D$. Suppose that there exist some $0 < \delta \leq \Delta$ such that its conditional Lebesgue density satisfies

$$\delta \leq \frac{d\pi(\cdot|x)}{d\lambda} \leq \Delta \quad \text{for all } x \in \mathbb{R}^D. \quad (52)$$

Theorem E.8 (Deep Gaussian Mixtures are Universal Conditional Distributions). *Suppose that π satisfies Assumption E.7. Then, for every compact subset $K \subseteq \mathbb{R}^{D_x}$, every approximation error $\varepsilon > 0$ there exists $M, N \in \mathbb{N}$ such that: for each $m = 1, \dots, M$ and $n = 1, \dots, N$ there exist MLPs: $a_m : \mathbb{R}^{D_x} \mapsto \mathbb{R}^{D_y}$, $v_m : \mathbb{R}^{D_x} \mapsto \mathbb{R}^M$ with ReLU activation functions and w_n, B_n learnable parameters such that the (non-degenerate) Gaussian-mixture valued map*

$$\hat{\pi}(\cdot|x) \stackrel{\text{def.}}{=} \sum_{n=1}^N \sum_{m=1}^M z_{mn}(x) \nu(d_{mn}(x), \varphi(D_{mn}(x)))$$

satisfies the uniform estimate

$$\max_{x \in K} d_{TV}(\pi(\cdot|x), \hat{\pi}(\cdot|x)) < \varepsilon. \quad (53)$$

If, moreover, $\hat{\pi}$ also satisfies (52) (with $\hat{\pi}$ in place of π) then additionally

$$\max_{x \in K} \text{KL}(\pi(\cdot|x), \hat{\pi}(\cdot|x)) \in \mathcal{O}(\varepsilon), \quad (54)$$

where $\mathcal{O}$ hides a constant independent of ε and of the dimension D .

The proof of Theorem E.8 makes use of the *symmetrized Kullback-Leibler divergence* KL_{sym} which is defined for any two $\alpha, \beta \in \mathcal{P}(\mathbb{R}^D)$ by $\text{KL}_{\text{sym}}(\mu, \nu) \stackrel{\text{def.}}{=} \text{KL}(\alpha||\beta) + \text{KL}(\beta||\alpha)$; note, if $\text{KL}_{\text{sym}}(\alpha, \beta) = 0$ then $\text{KL}_{\text{sym}}(\alpha||\beta) = 0$. We now prove our main approximation guarantee.

Proof of Theorem E.8. To simplify the explanation of our first claim, we provide the expression for $\hat{\pi}(y|x)$ from (19):

$$\hat{\pi}(y|x) = \sum_{n=1}^N w_n \sum_{m=1}^M v_m(x) \exp\left(\frac{a_m^\top(x) B_n a_m(x) + 2b_n^\top a_m(x)}{2\varepsilon}\right) \mathcal{N}(y | d_{mn}(x), \varepsilon B_n)$$

Thanks to the wide variety of activation functions available from Definitions E.3, E.4, E.5, we can construct the map $\hat{f}$ and directly apply Proposition E.6. This completes the proof of the first claim.

Under Assumption E.7, $\pi(\cdot|x)$ and $\hat{\pi}(\cdot|x)$ are equivalent to the D -dimensional Lebesgue measure λ . Consequently, for all $x \in \mathbb{R}^{D_x}$:

$$\pi(\cdot|x) \ll \hat{\pi}(\cdot|x)$$

Therefore, the Radon-Nikodym derivative $\frac{\hat{\pi}(\cdot|x)}{\pi(\cdot|x)}$ is a well-defined element of $L^1(\mathbb{R}^{D_x})$, for each $x \in \mathbb{R}^{D_x}$; furthermore, we have

$$\frac{\pi(\cdot|x)}{\hat{\pi}(\cdot|x)} = \frac{\pi(\cdot|x)}{d\lambda} \frac{d\lambda}{\hat{\pi}(\cdot|x)}. \quad (55)$$

Again, leaning on Assumption (52) and the Hölder inequality, we deduce that

$$\begin{aligned} \sup_{a \in \mathbb{R}^D} \left| \frac{\pi(\cdot|x)}{\hat{\pi}(\cdot|x)}(a) \right| &= \sup_{a \in \mathbb{R}^D} \left| \frac{\pi(\cdot|x)}{d\lambda}(a) \frac{d\lambda}{\hat{\pi}(\cdot|x)}(a) \right| \\ &\leq \sup_{a \in \mathbb{R}^D} \left| \frac{\pi(\cdot|x)}{d\lambda}(a) \right| \sup_{a \in \mathbb{R}^D} \left| \frac{d\lambda}{\hat{\pi}(\cdot|x)}(a) \right| \\ &\leq \sup_{a \in \mathbb{R}^D} \left| \frac{\pi(\cdot|x)}{d\lambda}(a) \right| \frac{1}{\delta} \\ &\leq \frac{\Delta}{\delta} \end{aligned} \quad (56)$$

where the final inequality under the assumption that $\hat{\pi}$ also satisfies Assumption 52. Importantly, we emphasize that the right-hand side of (56) holds *independently of* $x \in \mathbb{R}^{D_x}$ (“which we are conditioning on”). A nearly identical estimate holds for the corresponding lower-bound. Therefore, we may apply (Sason, 2015, Theorem 1) to deduce that: there exists a constant $C > 0$ (independent of $x \in \mathbb{R}^{D_x}$ and depending only on the quantities $\frac{\Delta}{\delta}$ and $\frac{\delta}{\Delta}$; thus only on δ, Δ) such that: for each $x \in \mathbb{R}^{D_x}$

$$\text{KL}(\pi(\cdot|x), \hat{\pi}(\cdot|x)) \leq C d_{TV}(\pi(\cdot|x), \hat{\pi}(\cdot|x)). \quad (57)$$

The conclusion now follows, since the right-hand side of (57) was controllable by the first statement; i.e. since (53) holds we have

$$\text{KL}(\pi(\cdot|x), \hat{\pi}(\cdot|x)) \leq C d_{TV}(\pi(\cdot|x), \hat{\pi}(\cdot|x)) \leq C\varepsilon. \quad (58)$$

A nearly identical derivation shows that

$$\text{KL}(\hat{\pi}(\cdot|x), \pi(\cdot|x)) \leq C\varepsilon. \quad (59)$$

Combining (58) and (59) yields the following bound

$$\max_{x \in K} \text{KL}_{sym}(\pi(\cdot|x), \hat{\pi}(\cdot|x)) \in \mathcal{O}(\varepsilon). \quad (60)$$

Since $\text{KL}(\alpha||\beta) \leq \text{KL}_{sym}(\alpha, \beta)$ for every pair of Borel probability measures α and β on $\mathbb{R}^{D_x}$ then (60) implies (54). □



Figure 10. Extended visual comparisons between the FSBM (Theodoropoulos et al., 2024) method (3rd column) and our method (4th column) for Woman-to-Man translation are shown here. The task is described in §5.3. The first column shows the source image and the second column the target image.



Figure 11. Visual comparisons for the Old-to-Young translation task between the FSBM (Theodoropoulos et al., 2024) method (3rd column) and our method (4th column). The task is described in §5.3. The first column displays the source image, and the second column shows the target image.